“This is driving me dotty: A new experimental method for studying sequential statistical learning”.

Sofia Maria Tsitsopoulou

A thesis submitted in partial fulfilment of the requirements of Nottingham Trent University for the degree of:

Doctor of Philosophy

June 2021

The copyright in this work is held by the author. You may copy up to 5% of this work for private study, or personal, non-commercial research. Any re-use of the information contained within this document should be fully referenced, quoting the author, title, university, degree level and pagination. Queries or requests for any other use, or if a more substantial copy is required, should be directed to the author.

In memory of my beloved mother…

Acknowledgments

Firstly, I would like to express my sincere gratitude to my director of studies & thesis supervisor, Dr. Gary Jones and the other 2 members of my supervision team, Dr. Mark Torrance and Dr. Jens Roeser for their continuous support with my Ph.D study and related research, for their patience, motivation, and immense knowledge. Each one of them contributed to my personal development and academic growth in different ways and on multiple levels (theoretical and practical). Their guidance helped me throughout the experimental research and writing of this thesis.

Additionally, I would like to thank the NTU Doctoral School, for offering me a fully funded scholarship scheme, in order to financially support this academic achievement. Without this opportunity, I would have never been able to secure a Doctorate position at first place. Furthermore, I would like to express my deep appreciation at the Division of Psychology in NTU, for providing to me all the necessary physical and mental resources needed (laboratories/ participation pool/ technological equipment/ training) to complete my research aims in order to structure this thesis. Furthermore, I would like to thank the technicians in Taylor Building (NTU - Psychology Division) Mr. Roy Barson, Mr. Ben Thompson and Mr. Steven Scott that provided unlimited support with the different technical issues that occurred during my research.

I would also like to say a big thank you to my family in Greece (father and brother) and UK (brother - Dr. Vasileios Tsitsopoulos) for supporting me emotionally and financially in times of great need, throughout my PhD academic journey and always reminding me that I can achieve anything that I have in mind, as long as I don’t quit

or give up on my dreams. As my dad says: “When life gives you lemons, you make lemonade”. At this point, I would also like to thank my dearest friend and family member for the past 13 years, my dog Malou, for always being next me, keeping me company and making me feel happy and loved. She was a great help to me during my Ph.D journey, and I feel blessed that I had her with me to accompany and support me.

Finally, I would like to thank a list of good friends, colleagues and brilliant scientists that supported my Ph.D journey with their encouragement, constructive criticism and academic intellect: Dr. Sebastiano Costa, Dr. Alex Muhl-Richardson, Dr. XU Xinyuan, Dr. Melina Throuvala, Miss Charlotte Boatman, Mr. Francesco Cabiddu & Mr. James Myers.

Table of Contents

List of Abbreviations

2AFC: Two-Alternative Forced Choice

3AFC: Three-Alternative Forced Choice

ADHD: Attention Deficit Hyperactivity Disorder

AGL: Artificial Grammar Learning

AOI: Area Of Interest

ASD: Autism Spectrum Disorder

BDD: Body Dysmorphic Disorder

CAPPUCCINO: Comprehension And Production Performed Using Chunks Computed Incrementally, Non-categorically, and On-line

CV: Consonant -Vowel

EEG: Electroencephalogram

GLMMs: Generalised Linear Mixed Effects Models

HMM: Hidden Markov Model

ICR: Implicit Chunk Recognition

ISI: Inter-Stimulus Interval

ISL: Implicit Statistical Learning

MaxEnt: Maximum Entropy

NLP: Natural Language Processing

OCD: Obsessive- Compulsive Disorder

PD: Parkinson's Disease

RNNs: Recurrent Neural Networks

RTs: Reaction Times

SA: Social Anxiety

SICR: Statistically Induced Chunking Recall

SISL: Serial Interception Sequence Learning

SL: Statistical Learning

S-R: Stimuli-Response

SRN: Simple recurrent network

SRTT: Serial Reaction Time Task

SST: Stimulus Sampling Theory

TGG: Transformational Generative Grammar

TPs: Transitional Probabilities

TRACX: Truncated Recursive Autoassociative Chunk Extractor"

VSL: Visual Statistical Learning

***Note:** *All code in this thesis, can be found in electronic format on Github by requesting access to the folder by accessing this link:*
https://github.com/smtsitsopoulou/This_thesis_is_driving_me_dotty.git

Abstract

This thesis introduces a new experimental paradigm for exploring the cognitive mechanism of statistical learning (SL). SL refers to the ability of extracting statistical regularities and patterns implicitly from the sensory input and forming them into units of knowledge. Most of the methodologies used in the literature to investigate or assess the mechanism of SL, use tasks and measurements that assess the outcome of the learning process, rather than the process itself. Therefore, the new experimental paradigm introduced in this thesis, provides a new way of observing the SL mechanism while it operates, with the usage of a gaze contingent/time-displayed eye-tracking sequential SL task (on the visual domain). Once the new methodology is introduced and assessed across two different eye-trackers (Gazepoint GP3; EyeLink 1000 (SR Research Ltd., Mississauga, Canada)), it is applied to specific research contexts about the encoding process of sequences in SL and the effects of sequence length and mixtures of sequences lengths in sequential SL. The findings of this thesis support evidence for a hierarchical structure in the SL encoding process, where the last item of a sequence is better learned than the previous and so on. Additionally, it is suggested that sequences with shorter lengths (2 items) are learned faster than longer length sequences (4 items) in same length tasks. However, when the sequential SL occurs within mixed length sequences, longer sequences are facilitated from their coexistence with shorter sequences in the task, resulting in faster learning, while learning of shorter sequences is impeded by their coexistence with longer sequences in the task. A contextual evaluation of the two eye-tracking systems used in this thesis is described to justify the usage of low-cost equipment in experimental research. Finally, a critical discussion of the findings and its applications on educational and clinical areas is given.

Key Terms: *implicit learning mechanism, sequential statistical learning, eye-tracking, new experimental paradigm, information processing, transitional probabilities*

Thesis Summary

This mini chapter aims to introduce to the reader a summary of the thesis content and highlight the main key terms used in it. This thesis investigates the mechanism of statistical learning via a new experimental paradigm.

Chapter 2, contains a detailed literature review about SL. It aims to help the reader understand better the mechanism of SL as part of cognition and highlight the gaps in the literature this thesis aims to answer. SL is part of implicit cognition, and as such, it develops associatively, fast and automatically. Throughout the years many experimental tasks have been used to understand the SL mechanism, such as 2AFC tasks, tasks with familiarization and testing phases, tasks that use a reward system, etc. However, a common element across all those experimental methodologies was that they were inferring the SL mechanism by assessing the outcome of the learning process rather than the process itself. Therefore, the need for a new experimental paradigm was born, that would record meaningful data that describes the SL process rather the outcome/result of it. An extension of this rationale was to reflect on the current understanding of how information encoding occurs during SL, if it is an all or none process, if the learning occurring during SL relies merely on the extraction of transitional probabilities, or if there is a unit chunking mechanism and if so, how it works. Sequential SL was chosen as the mechanism investigated by the new experimental paradigm, since in sequential SL, each item of the sequence has a unique point in space and time within the sequence presentation and therefore could provide meaningful details about the learning of associations formed across each item of the sequence and the broader learning of the sequence as a unit. Furthermore, sequential SL allowed us to investigate sequence length effects and

mixture of length effects that are still unanswered and play an important role in understanding SL in processes such as language learning.

In Chapter 3, the research questions of this thesis are detailed. The research questions derive directly from the gaps in knowledge identified within the literature review. This thesis aims to (a) introduce a new experimental paradigm that records the process of SL rather than the outcome of it, (b) understand how sequential SL occurs and develops in time and what information processing theories fit better the data obtained and finally (c) understand how specific components of sequences such as sequence length or mixture of lengths can affect the sequential SL process.

Since the research aims/ questions of this thesis have been justified by the literature review, the next step was to introduce the new experimental paradigm and the development of it by the presentation of two designs that were used to investigate the sequential SL of 3 dot sequences on the visual domain with the use of eye-tracking in Chapter 4. Design A, is the design that was first created to capture the sequential SL process. During design A, participants were presented with an array of 16 locations on the screen (presented as 16 grey dots) and a moving single green dot on those locations. The movement of the dot wasn't random. Each dot represented an item in a sequence, and in total 4 sequences of 3 items each were used. The visual stream of the moving dot was continuous. Each trial consisted of a time gaze contingent period until the participant looked at the green dot, followed by a time displayed period of 275ms where the dot remained still (to secure that the participant has seen it), followed by a time displayed period of 750ms that the green dot disappeared and the participant had to guess where the next location would appear by looking at it and ending by the presentation of the next green dot that was the next item in the sequence. Participants had no information about the stream of sequences, or the length of the sequences. Eye-samples on the correct next location on the array during the blank period of 750ms were considered as learning and were

expected to increase as exposure to the sequence and its items increased. In order to record those data, we used a Gazepoint GP3 (60 Hz) eye-tracker that recorded approximately 45 eye-samples for each 750 ms time window. The big success of Design A was the fact that it was recording the learning process from time-point 0, when the 1st item of the sequence was presented on the screen until the presentation of the last item of the sequence at the end of the task. However, the visualisation of the data obtained from Design A, demonstrated large variability in individual difference learning scores, with some participants learning a little, most of them learning nothing and a few of them demonstrating high learning scores. That fact, lead to the creation of Design B that is presented in the first section of Chapter 5. Design B, we used a more powerful eye-tracker, EyeLink 1000 (SR Research Ltd., Mississauga, Canada, 1000Hz), in order to be able to record 750 eye-samples during the blank window of 750ms where learning occurs, while a feedback beep sound was introduced at the beginning of the presentation of each green dot, if the participant failed to attend to the location of the dot during the blank 750 ms period, as a form of negative feedback. That way it was secured that a powerful tool was used, that allowed the recording of the mechanism within the specific time limitations of the task, while negative feedback was introduced in task, as a form of attention grasping/motivation strategy for participants. The visualisation of the data obtained from Design B, again contained a lot of individual differences in learning scores across participants, however this time more participants demonstrated learning and the learning scores were greater numeric values. Since the first research aim of this thesis was successfully completed by creating a new experimental paradigm that captured the time course of the SL mechanism and was acknowledged that individual differences are part of the SL mechanism therefore can't be prevented, it was decided to continue with design B, as the main experimental design used to provide answers to the rest of the research questions of this thesis. The second section of Chapter 5 is devoted to the application of the experimental methodology explained in the first

section, in 6 different tasks. The tasks were split in two categories: (a) the same length tasks (2 dots, 3dots, 4 dots) that contained same length sequences within each task and (b) the mixed length tasks (2&3 dots, 2&4 dots, 3&4 dots) that contained mixed length sequences within each task. At first, we examined if learning was successful across all tasks. Next, an analysis was performed to examine how the different type of transitions within the new methodology reflect SL. The learning of the 1st position of a sequence differed from the learning observed within the 2nd, 3rd and 4th items of a sequence. This can be explained by the fact that the learning observed on the 1st item of a sequence was random at the beginning of the task (1/16 chances) and transformed during the task, as the learning of the within sequence positions affected the chances of guessing (1/3).

In Chapter 6, we used the 6 tasks described in Chapter 5 to answer if sequential SL is an all or none process, and provide information about how unit chunking occurs, by exploring the learning patterns within the items of a sequence. The results suggested a consistent learning pattern across all tasks, suggesting that as the position of the item in the sequence was increased, the learning of that item was better. That learning pattern within the items of a sequence, rejected the all or none learning approach in SL, while it suggested a hierarchical structure of learning, that derives directly from the SL mechanism and is unaffected by sequence properties such as sequence length or mixture of sequences.

Furthermore, Chapter 7 focused on answering the final research aim of this thesis, which relates to sequence length effects and mixture of lengths effects on SL performance, by using the 6 tasks as explained in Chapter 5. The results suggested that shorter length sequences are learned faster than longer length sequences, while longer sequences are better learned in mixed length tasks, because they are blended with shorter length sequences. These findings can be directly applied in educational psychology, while they can be used for further research investigation of

mechanisms such as language learning that rely on mixed length sequences (syllables forming words/ words forming sentences analogy).

After answering all the research questions of this thesis, Chapter 8 was created to highlight methodological issues for people who may want to use this or similar paradigms, showing how different tools such as eye-trackers can affect the quality of measurements of the tasks. Firstly, it gives a summary of the contribution of eye-tracking as a tool, in the field of cognitive and clinical psychology and then it moves onto detailing the main technical differences of the two eye-tracking systems that were used in the new experimental methodology suggested in this thesis (Gazepoint GP3 vs EyeLink 1000,SR Research) and explains how different types of analysis can lead to underestimation or overestimation of findings based on the powerfulness of the tools used. The final section of this chapter is devoted to evaluation of the two eye-tracking systems within specific research contexts.

The final chapter of this thesis (Chapter 9) is a discussion about the application of the new suggested experimental paradigm in different concepts, a reflection of previous findings and literature on the findings of the current thesis about sequence length effects, mixed length effects and information processing in SL. Additionally, future implications and limitations of the current findings and design are elaborated, and additional evidence about how the current design can be used to examine individual learning differences in SL patterns.

“Statistical Learning (SL) – a powerful but still unknown mechanism- Evidence from theory, research methods and practices in the field of cognitive psychology”

Chapter Summary

In the interest of understanding the gaps in knowledge around the cognitive mechanism of SL, we devoted this chapter in a literature review about SL theory. This literature review will help the reader understand better the mechanism of SL and its applications on different cognitive learning concepts. Statistical learning (SL) is one of the main cognitive mechanisms that infants use to learn the world and start forming their first words. Even though this ability to extract patterns and regularities from the sensory input and form them into a meaningful unit is crucial, there aren’t plenty of studies trying to unzip the basic operation system of that mechanism, neither in infants nor in adults. Some of the main themes that will be covered in this chapter refer (a) to the theory behind statistical learning, (b) the current knowledge about SL from research and the used methodologies, (c) the temporal component of sequential SL mechanism of how people from various psychological fields (cognitive, psycholinguistics, cognitive neuroscience) tried to define them through their psychological tasks and models, (d) the gap in knowledge around the field of SL. Understanding what SL is and the different approaches towards it, are key components for identifying the current gaps in the literature and determinant for the nature of the research aims of this thesis.

A. Statistical Learning Theory

A.1. General SL Theory

SL theory is a theoretical approach in which mathematical models are used to describe processes of learning. A cognitive conceptualisation of SL would define SL as the ability of extracting transitional probabilities and statistical patterns from the sensory input and forming them into a coherent unit of knowledge, which can be later used to retrieve the memory of the input or predict the next input (Aslin & Newport, 2012). For example, people may initially see A, B, and C as individual elements but if they appear often together in the same sequence, they will eventually learn them as a unit ABC. SL ability has been crucial for the survival of primates (Newport, Hauser, Spaepen & Aslin, 2004), since the ability to identify regularities and patterns helps them understand their environment, make reliable predictions about it and respond to familiar stimuli in it.

One of the first statistical approaches to learning, was introduced by Estes (1950) with Stimulus Sampling Theory (SST). SST aimed to provide a statistical explanation about how learning occurs. The theory suggested that the learning of a specific stimulus-response (S-R) association occurs on a single trial; but the overall process of learning contains accumulations of discrete S-R pairings and is continuous. This suggests that during any trial that learning takes place between an S-R, there are numerous Rs that can match the S, but only the portion of Rs that are effective can form associations with the S. Therefore, the Rs are a sample out of all the possible stimuli that the participant has been exposed to. Many types of psychological learning tasks and paradigms were built around this theory, such as free-recall, concept identification, operant conditioning, paired-associates, stimulus generalization, preferential choice, paired-associates etc.

SST has been used mostly in memory, language learning, and developmental experiments and it's based on two basic principles: (a)while the learning of a particular instance can be either all or none, the overall learning process is gradual, continuous and cumulative; and (b) fluctuations in the environment and procedural factors will cause variability in learning progress (Estes,1970). Some of these environmental and internal components were identified later in the literature as perceptual components, cognitive structures and verbal load (Saffran, 1996; Aslin & Newport, 2012).

A.2. Is SL a mechanism: Mechanist philosophy vs Cognitive science

In philosophy of science, there are many approaches of the concept of mechanism. A commonly used approach on understanding mechanist philosophy is the Levy's taxonomy (Levy, 2013). According to Levy's taxonomy (Levy, 2013), there are 3 central mechanistic theses that coexist in every so-called mechanism but they are not always well-marked as separate theses. These mechanisms are (a) *the causal mechanism* which is referring to the relationship between the causal relations of the external physical phenomena and their existence in virtue of underlying mechanisms, (b) *the explanatory mechanism* which highlights the need of citing mechanistic information in order to explain a phenomenon and (c) *the strategic mechanism* that suggests that certain phenomena are best handled mechanistically The strategic mechanism, according to Levy (2013), is an exploratory "strategy" about the scientific method. It's a way of understanding the nature of complex systems, by treating those systems mechanistically, and breaking them down to smaller simpler systems that have specific epistemic and cognitive features. This approach is largely used in cognitive sciences in methods such as AI and Computational Modelling.

Most philosophers tried to create distinct criteria in order to decompose the complex systems of mechanisms. They mostly emphasized on (a) the importance of *parts*, *operations and their*

organization (Bechtel & Abrahamsen 2005 ; Glennan, 1996), (b) *the hierarchical structure* of mechanisms and the distinct discrimination between levels and bottoming out in a mechanism. This means that the parts and operations of a mechanism are placed in a series of levels with different importance or status. The levels of each mechanism are distinctive and there is no overlap between them. The hierarchy in a mechanism can rely on either the importance of each part (most important →less important) or to the procedural order of each part in a mechanism (what part occurred, 1st , 2nd etc). (Glennan, 1996; Machamer, Darden & Craver, 2000), (c) *the causality and the notion of causal laws* (Glennan, 1996; Craver, 2007) and how causality allows the interaction between the different parts of a mechanism and finally (d) *the ability to create mechanistic explanations* in order to explain the mechanisms observed in the physical world (nature)(Craver, 2007; Bechtel & Abrahamsen, 2012).

However, in psychological mechanisms and especially in cognitive scientific mechanisms, we have an information processing approach rather a material approach (Bechtel, 2008). This information processing approach comes from observations of behavioural tasks, and most of the times we use scientific tools from neuroscience, biology and computer science to develop our explanatory, causal and strategic mechanism.

Statistical learning has been considered as a solid psychological mechanism in the literature of cognitive and experimental psychology. Many studies tried to understand the parts of this mechanism, the levels and the operational causality of this mechanism or even provide a mechanistic explanation that would break down SL to smaller cognitive specific operational parts (Creel, Newport & Aslin, 2004; Saffran, Johnson, Aslin & Newport, 1999; Fiser & Aslin, 2001; Turk-Browne, Scholl, Chun & Johnson, 2009; Saffran & Thiessen, 2007), however most of the times their findings have failed to compose a solid mechanistic approach with 4 theses and therefore created the issue “black box” of statistical learning mechanism (Figure 3.1).

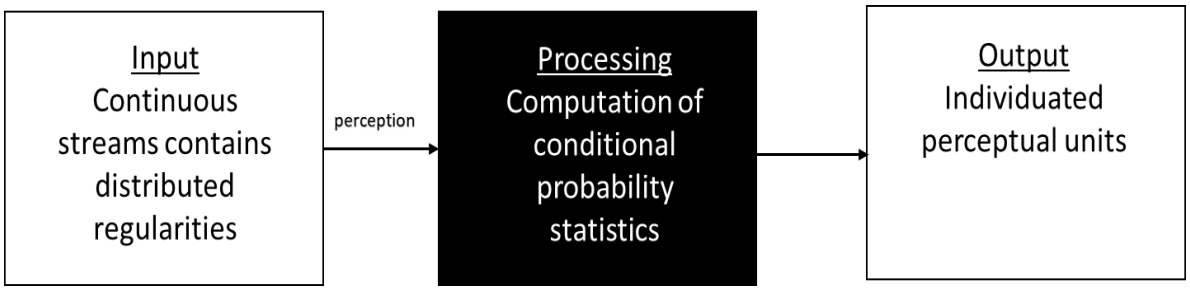


Figure 3.1 Mock diagram of the statistical learning mechanism (Betzler, 2016).

A.3. SL as part of implicit cognition.

According to Reingold and Ray (2006), implicit cognition consists of a set of unconscious influences, such as memory, knowledge and perception and those unconscious influences have the ability to change a person's behaviour. The idea that there are 2 fundamentally different sets of cognitive processes (implicit vs. explicit), has been widely used in the field of cognitive psychology and many researchers tried to create theories/models to understand how this "dual-processing" system works (M. Posner, Snyder & Solso, 1975; Schneider & Shiffrin, 1977). The implicit system is believed to be associative, fast and automatic and the explicit system is believed to be slow and reflective.

In 1967, Reber defined implicit learning as a mechanism that involved "automatic learning mechanisms that are used to extract regularities and patterns distributed across a set of exemplars, typically without conscious awareness of the regularities being learned" (p.114). This definition highlighted the main difference between conscious and unconscious processes, which relies on the fact that during implicit processes the individuals are unaware of manifesting the process (A. S. Reber, 1989, 1993).

According to De Houwer, Teige-Mocigemba, Spruyt and Moors (2009), "Implicit measures can be defined as outcomes of measurement procedures that are caused in an automatic manner by psychological attributes. To establish that a measurement outcome is an implicit measure,

one should examine (a) whether the outcome is causally produced by the psychological attribute it was designed to measure, (b) the nature of the processes by which the attribute causes the outcome, and (c) whether these processes operate automatically." (De Houwer, Teige-Mocigemba, Spruyt & Moors, 2009, p. 347). A great example of implicit learning task is the artificial grammar learning (AGL) test suggested by Reber (1969). During an AGL test, participants are exposed to novel stimuli that are structured according to a set of grammatical properties and rules. After the exposure to this test, participants are able to recognise the difference between stimuli that follow the grammatical structure and those that don't, without realising how they achieve to discriminate between those two, or even being aware that they are exposed to structured/non-structured stimuli.

Implicit learning is extremely crucial in processes such as real-life problem solving (Funke & Frensch, 2007), language learning (Shaffran et al., 1996; Gomez & Gerken, 1999), and may also be a predictor of educational attainment (Mackintosh, 2011). The SL paradigm is the most recently developed, and investigated, out of the implicit learning paradigms. It resembles a lot the AGL paradigm, however it was applied on different modality. The AGL paradigm used letter strings- presented on the visual domain, to represent the underlying formal grammatical structure generated by a Markov chain sequence and the grammatical structure as measurements, while Saffran et al. (1996), used simpler auditory speech stimuli (syllables stream) with the purpose of investigating if pre-verbal infants could extract statistical structure for auditory speech-like input (Reber et al., 2019; Safran et al., 1996).

A.4. Rule-based natural language processing vs. Statistical Learning (SL) in language processing (usage-based approach).

In order to understand the debate behind rule-based learning and statistical learning in language it is important to define Natural Language Processing (NLP). The term NLP is used to describe the interaction between computers and human language. It involves a set of processes such as

developing algorithms and models that enable computers to understand, interpret, and generate human language in a meaningful manner.

Rule-based NLP refers to an approach that uses predefined rules or patterns to analyse and understand language. In this approach linguistic rules are explicitly defined to perform various NLP tasks such as text parsing, information extraction, sentiment analysis, and question answering. Rule-based NLP has been inspired by Chomsky (1957) and his introduction to the concept of transformational generative grammar.

Transformational Generative Grammar (TGG) is a linguistic theory developed by Chomsky (1957). TGG describes the structure and the rules of human language that produce grammatically correct sentences. TGG consists of six main components: (a) *generative grammar* - TGG is a type of generative grammar that seeks to generate and analyse the grammatical structure of sentences. It focuses on describing the rules and principles that generate an infinite number of grammatically acceptable sentences in a language; (b) *transformational rules* - TGG introduces the concept of transformational rules, which are operations that transform one structure into another while preserving the meaning. These rules account for various transformations observed in language, such as passive voice, question formation, relative clauses, and negation; (c) *deep structure and surface structure*- TGG proposes the existence of deep structure and surface structure. Deep structure represents the underlying meaning and syntactic structure of a sentence, while surface structure represents the actual arrangement of words in a sentence; (d) *phrase structure rules* - TGG utilizes phrase structure rules to describe the hierarchical structure of sentences. These rules specify how constituents (e.g., nouns, verbs) can be combined to form larger units; (e) *generative capacity* - TGG aims to define the generative capacity of a language, which refers to the set of all grammatically possible sentences in that language. The theory seeks to capture the rules and constraints that generate grammatical sentences and exclude the ones with wrong grammar; (f)

universal grammar - Universal Grammar suggests that all human languages share a common underlying structure and a set of universal principles. The transformational rules and underlying structures proposed in TGG are considered part of this innate linguistic knowledge.

The main strength of rule-based NLP approaches is that they offer an explicit and transparent overview over the language processing tasks. It offers a clear and robust structure and set of rules on how NLP occurs. Models that have used this approach have allowed the researcher to manipulate the language rules based on their knowledge and understanding around language processing. The explicit and transparent nature of rule-based NLP has allowed to create models that can handle complex linguistic concepts and produce interpretable and robust findings. However, rule-based NLP's explicit nature suggests that these models might struggle to cover concepts such as linguistic ambiguity, linguistic exceptions and variations across large scale datasets. It is extremely difficult to develop a set of rules for all the possible language variations and nuances that can be observed in NLP. Therefore, it lacks ecological validity as there will always be an unaccounted variance that can't been interpreted based on the explicit rules.

Statistical learning was introduced in the context of language learning, syntax and grammar by Pinker (1979). While Pinker (1979) recognises the role of innate factors in language like Chomsky (1957), he believes that cognitive abilities and exposure to linguistic input are determinant to language acquisition (Pinker & Prince, 1988).

Statistical learning in language processing was introduced as an approach in the late 1980s and early 1990s. In their book Manning and Schutze (1999), discuss various statistical approaches to NLP including n-gram models, Hidden Markov Models and Max Entropy models. This period was crucial for the field of AI and computational linguistics, as it marked a shift from Chomsky's perspective on NLP towards using statistical models and machine learning techniques to understand NLP tasks.

For example, Brown et al., (1992) introduced the concept of n-gram models, which use statistics to estimate the probability of a word or sequence of words based on their context. The key idea behind n-gram models, is that language is represented as a sequence of n-grams, which are contiguous sequences of n items (e.g., words). The n-gram models are capturing statistical regularities in the dataset by calculating the probabilities of n-grams based on their frequencies in the training data. One main issue with n-gram models, is the "data sparsity", where n-grams that haven't been included in the training data, lead to zero probabilities and therefore poor generalisation.

Rabiner and Juang (1986) introduced Hidden Markov Models (HMMs) as an SL approach to investigate NLP in speech recognition. HMMs are statistical models that represent sequences of data that are generated by an underlying process with hidden states. In the context of speech recognition, HMMs are employed to model the relationship between spoken words or phonemes and the acoustic features extracted from the speech signal. There are in general 6 key principles in HMMs: (a) *states and transitions* - represents the assumption speech can be modelled as a sequence of hidden states. Each state represents a unique linguistic unit (e.g., phoneme, word). An HMM is representing the set of states and the in between state transitions; (b) *observation symbols* – In every state, the HMM creates an observation symbol representing the acoustic features (e.g., frequency) by extracting it from the speech signal; (c) *probability distributions* - Every state in the HMM has associated probability distributions over the observation symbols; (d) *model training* - The transitional probabilities and the observation distributions, are calculated based on a training dataset; (e) *Viterbi decoding* - The Viterbi algorithm, computes the most probable state sequence in speech by considering both the transition probabilities and the likelihood of the observation symbols; and (f) *language model integration*- HMMs incorporate a language model to further refine the recognition results and

improve the accuracy. HMMs, are in general powerful tools for speech recognition, as their probabilistic nature, allows to capture the temporal dynamics of speech.

Another statistical approach in NLP that was developed around that period were the Maximum Entropy (MaxEnt) models (Berger, Della Pietra & Della Pietra,1996; Ratnaparkhi, 1997). MaxEnt NLP models operate in nine simple steps: (a) *define the task* - define the specific NLP you are interested in exploring; (b) *define the features* – defining the syntactic, semantic and lexical properties of the input; (c) *collect labelled training data* - collect datasets where every input text has its corresponding output label; (d) *extract features* – extract the features from the training data; (e) *calculate the empirical feature expectations* – calculate the observed frequencies for each output label in the training data; (f) *define the MaxEnt model* - Construct the MaxEnt model, specifying the feature expectations derived from the training data as constraints; (g) *model training* - Use an optimization algorithm to estimate the model parameters that maximize the likelihood of the observed feature expectations while satisfying the constraints; (h) *create predictions* - make predictions on new instances; (i) *evaluate the model* - use NLP task appropriate metrics such as accuracy or precision to evaluate the model. MaxEnt NLP models seem to be very efficient at computing complex patterns and dependencies in the data, due to the fact that the models are being trained with labelled data and therefore the entropy of the probability distribution increases.

SL approaches in NLP seem to significantly benefit from the implicit and automatic nature of the mechanism. The main strength of SL approaches is that they can automatically learn patterns and rules from vast datasets and even generalise efficiently on examples that the models hasn't been exposed to. However, SL models have significant weaknesses. They are significantly less transparent in comparison to rule-base models (probability based) and the interpretation of the outcomes can vary based on the understanding of the researcher around language mechanisms. Additionally, the training data have a critical role in the operation of the

model and can affect the ecological validity of those models. The SL models usually require vast amounts of training data (costing both time and money), and they can still fail to capture linguistic rules or complex structures of language if those elements are not present in the training data. Even the most carefully selected and rich training data cannot capture the sheer extent of language that someone can encounter in NLP. Similarly, because the SL models rely entirely on the training data, they can be prone to the bias of the training data or overfitting, SL approaches are difficult to be tested in full extent.

Some researchers such as (Gomez-Perez, Denaux & Garcia-Silva, 2020; Dash, 2021) have tried to combine the two different versions of models by creating hybrid models that would benefit from the strengths of each approach. Hybrid models tend to incorporate explicit rules into SL models in order to improve accuracy, interpretability and generalisation. In recent years, SL approaches and models have gained more prominence in NLP as they can handle vast datasets and complex patterns. However, before choosing an approach it is crucial to consider the domain and the context. Rule-based approaches can be extremely valuable when exploring domains that explicit rules have already been predefined/generated in a robust manner, or when interpretability and control over the data is crucial.

In this thesis, we align with the SL approach, as our new suggested methodology mimics the language paradigm of SL by Saffran et al. (1996) on the visual domain, but without the linguistic load, as it doesn't include any language related stimuli. That allows us to observe how the SL approach would work for a language-like paradigm, where the complexity of the patterns can be analogous to language. Additionally, the SL approach allow us to make further investigation on how information processing occurs during SL, without explicit a priori rules and limitations in a simplistic, easy to interpret manner.

B. Methodologies used in the literature to examine SL.

One of the first methods that used to understand SL in the field of cognition, was transitional probabilities (TPs) (Saffran, Newport, & Aslin, 1996; Aslin, Saffran & Newport, 1998; Perruchet & Desaulty, 2008). Saffran et al., (1996) investigated SL and language acquisition mechanisms on 8 months old infants. In their first study they familiarized 24 8-month-old infants with 2 min of a continuous speech stream that consisted of four three-syllable nonsense words repeated in random order. In order to create the speech stream, they used a speech synthesizer with a monotone female voice that was producing 270 syllables per minute. In their first study, participants were firstly exposed to a continuous stream of visual or auditory stimuli, called as the familiarization phase. During that familiarization phase, the stimulus sequences were divided into triplets of co-occurring elements (e.g., in the continuous word stream string kupadilagotubidako, were included the three-syllable chunks/triplets kupadi, lagotu, and bidako). The order in which these triplets occurred was free and therefore, transitional probabilities (TPs) were structured such that TPs from one syllable to the next were higher for stimuli within a triplet (e.g., padi) than for those that span a triplet boundary (e.g., dila). Then the infants had to undergo through a preferential look process between a word that was presented in the familiarisation phase (e.g., kupadi) and a new one (e.g., gadilo). Two of the three-syllable strings were "words" of the artificial language that had been displayed from during the familiarization phase, and two were novel three-syllable "nonwords" that consisted of the same syllables heard during familiarization in different order. Infants demonstrated longer durations of listening the novel non-words. But serial order information isn't enough cue to word boundaries. That led them to design their second study to assess the ability of infants on extraction of relative frequencies of co-occurrence of sound pairs. In these pairs the transitional probabilities that signal the word boundaries were relatively low, and therefore the statistical computations were harder. In this second study they recruited again 24 8-month-old

infants and familiarized them with 2 min of a continuous speech stream consisting of three-syllable nonsense words similar in structure to the artificial language used in their first experiment. During the test phase the infants listened to two words and two "part-words." The part-words consisted of the last syllable of a word and first two syllables of another word. These part words were designed in such way that infants have heard them during the familiarization phase, however statistically they do not correspond to words. That detail in the design was crucial, as the only way that infants could judge those part words as novel was only if they had learned the words with great accuracy and they were able to acknowledge the word boundaries of a word. Similarly, infants demonstrated longer hearing time for the part-words rather than the actual words, indicating that they were able to identify them as novel stimuli. The interesting conclusion from this study was that word segmentation, one of the most basic processes in language acquisition is successfully accomplished by 8 months old infants, by using statistical relationships between the neighbour sounds. Infants' exposure to a constant sequential stream of sounds was enough to determine word boundaries only based on the extraction of TPs that differed within and without a word.

Fiser and Aslin (2002), used TPs in order to understand how SL operates in infants in the visual domain on objects recognition. More specifically, their aim was to examine if statistically optimal representations of scenes can be formed during early development (9-months old infants), by using a habituation paradigm. In total they used 12 shapes grouped into four base pairs and four "noise" elements. Each scene consisted of a "noise" element and a base pair that was located within a 2x2 grid. Each "noise" element was assigned to one base pair but could be located in 1 out of 4 possible locations and the screen, resulting in a total of 16 scenes. Again, this task was a preferential looking task with an observer initiating the presentation sequences and the infant's behaviour. The habituation phase consisted of an attention getter (pulsing checkboard pattern accompanied with sound effects, located in the centre of the

screen). Once the infant looked at the attention target, the habituation process started with the scenes appearing in random order in the centre of the screen. At each trial when the scene appeared at the centre of the screen, the size of the scene loomed over the course of 1.5s and then paused at its maximum size for 0.5s. Then the scene disappeared, and a new trial started with another scene going through the same looming loop. During the testing trial, infants were presented with a single display repeated over and over. In Experiment 1, all four scenes were displayed during the habituation phase while in Experiment 2 and Experiment 3, all four low-frequency base pairs were used, but only two high-frequency base pairs scenes were presented. The scenes with low-frequency pairs were presented twice more often that the scenes with high-frequency base pairs in the tasks. Their results showed higher looking times (RTs) on base pairs compared to frequency balanced non base pairs, suggesting that 9-month-old infants are sensitive to the statistical structure of multielement scenes.

Monaghan, Schoetensack and Rebuschat (2019) suggested a unified framework that integrates implicit and statistical learning, two processes that are often considered separate in cognitive science. Implicit learning refers to the acquisition of knowledge without conscious awareness, while statistical learning involves the extraction of regularities from the environment. Both of those mechanisms share common underlying mechanisms and therefore can be studied by using a single paradigm. In order to examine both implicit and statistical learning under the same setting, they created a between group design, cross-situational learning task of artificial grammar and a post-test questionnaire about participant's awareness of patterns in the task. This study was conducted on adult participants and each participant was randomly allocated to one of the two conditions (incidental exposure or instructed exposure). The cross-situational task consisted of 8 visual objects, whose shape was taken from Fiser and Aslin (2012), 8 types of object motions, 16 two-syllable non-words that were used as content words that could describe the process occurring to the visual object (either shape or motion) and 2 monosyllabic

pseudowords (“tha” and “noo”) that could mean either the shape or the motion. In experiment one, at the beginning of the task, participants were instructed that they would be presented with two scenes and would listen to a sentence. Their task was to match the sentence to a scene. After the instructions, both groups were presented with an example. During the example, a rectangular shape moved in a circle while the participants listened to the sentence “Tha trepier noo vinnoy”. The incidental group was instructed that the word “trepier” referred to the shape of the scene and that the word “vinnoy” referred to the circular movement. They were reminded again about this instruction midway through the task. The instructed group received explicit instructions about the words and the sentences that they would listen on top of the instructions that the incidental group received: “Each sentence contains the name of an object and the name of its motion. The object name is always preceded by the word tha, and the motion name is always preceded by the word noo.” (Monaghan et al., 2019, p. 540). In each trial, the participants observed two scenes foe 3 seconds; each scene containing an object that was performing a movement and then they listened to the sentence. Participants had to match the sentence to the right scene (left or right) as fast and as accurately as they could by pressing a keyboard button. There was a 500ms pause in between trials, and a total of 12 training blocks that contained 24 trials each. After the training phase participants moved to the testing phase without a break. Because participants could complete the task by selectively learning only the noun–object or verb–object pairing, two different testing blocks were introduced at the end of the training: (a) the testing block that was testing verb learning, where participants were exposed to two scenes with a novel object doing different movements and a word that referred to a specific movement (from training phase); participants were required to select the scene described by the word) and (b) the testing block for noun learning, where participants saw two objects and heard one word that referred to a specific object (from training phase); participants again had to select the object that was described by the word. There was a total of sixteen trials

per testing task. After completing the task, they completed a questionnaire to examine they were aware of any sentence structures, rules or patterns in the experimental task. In experiment 2, they used the same procedure as they did in experiment one that differed only in two aspects. The first difference was that they didn't provide the description of a trial before the training and the second difference was that they introduced a second decision making per trial during the training phase. Participants had to make an additional decision per trial, based on whether their choice for that trial was based on guess, intuition, recollection or rule knowledge, by pressing a keyboard button. The instructions about those judgements on their choices were clear. Participants had to choose: (a) guess - if their decision was random (as flipping a coin); (b) intuition - if they thought their answer was correct but couldn't justify/explain it; (c) recollection - if they had consciously recollected this part of the sequence or all the sequence information from their memory and (d) rule knowledge - if they followed a conscious rule that they could verbalize when making the decision. The results suggested that with increased exposure to the task, participants correct responses were increased. They suggested that their paradigm uses an artificial language that is used to pair visual scenes and sentences and that explicit language structures affect both the grammatical and the statistical learning of vocabulary.

The results suggested that with increased exposure to the task, participants correct responses were increased. They suggested that their paradigm uses an artificial language that is used to pairs visual scenes and sentences and that explicit language structures affect both the grammatical and the statistical learning of vocabulary.

Kirkham, Slemmer and Johnson (2002) also compared the statistical learning between infants who were one-, five-, and eight-months old with the use of TPs. Interestingly, they found that there were no significant differences between the 3 different age groups and their ability to

discriminate between speech stimuli that adhered or violated the statistical regularities. That paradigm, even though it doesn't simulate the complexity of formal linguistic structures, it provides evidence that even preverbal infants possess a sophisticated learning ability that could support key aspects of language learning (Reber et al., 2019) and it's independent of other learning mechanisms in infancy such as reinforcement (Kirkham et al., 2002).

The two-alternative forced-choice task (2AFC), is a commonly used task to measure SL. During a 2AFC the learners are presented with pairs of stimuli and are asked to select which of the two items can recall. A possible issue with the usage of 2AFC tasks is the fact that you rely an implicit process such as SL on an explicit response that relies on a "gut feeling" about implicitly acquired statistical regularities. Therefore, Franco, Eberlen, Destrebecqz, Cleeremans and Bertels (2015), suggested that 2AFC may be a better measurement tool for explicit decision-making processes rather than the actual implicit SL mechanisms. Additionally, 2AFC performance is likely to introduce error variance due to cognitive complexity, in a way that the scores are not reflective of the individual differences during the learning process.

The usage of 2AFC tasks in a SL learning paradigm has been questioned by Siegelman and Frost (2015), it fails to capture and explain individual differences in performance. Frost, Armstrong, Siegelman and Christiansen (2015) came up with a new theoretical approach to SL in order to explain individual differences. According to their approach, SL is a set of computational principles that operates on the general domain for different modalities, and therefore is affected by the characteristics and constraints of the modality. They argue that SL has both modality-specific constraints and domain-general principles and suggest that SL depends on modality specific neural networks and some partially shared neural networks. Those networks represent the domain general and domain specific components of SL that have

been observed in previous research. They are also suggesting that further exploration into these networks can be done by targeting individual differences in these networks. More recently the concept of TPs has been replaced by the concept of chunking and more specifically the Statistically Induced Chunking Recall (SICR) method. In a SICR task participants are exposed to an artificial language, using a standard statistical learning familiarisation process. Then, they are asked to perform a recall task on strings of syllables that either follow the statistical patterns of the familiarization phase or include the same syllables presented in different order (Isbilen, McCauley, Kidd & Christiansen, 2020).

Van Witteloostuijn, Lammertink, Boersma, Wijnen and Rispens (2019), that used a 2AFC and 3 AFC paradigm to examine SL performance on the visual domain, on early-school- aged children. Their aim was to introduce the concept of RT to explain part of the variability in individual differences in SL. The visual statistical learning (VSL) task contained triplets instead of pairs this time. During the familiarization phase participants were presented with an alien character on the screen and had to give a button response to proceed to the next alien. During the testing phase, participants, had either to choose the triplet that they have seen before, or complete a missing stimulus of the triplet. Half of the participants performed a cover task, while the other half did not. The researchers successfully measured the online sensitivity to the statistical structure by comparing the RTs for the predictable vs the unpredictable aliens, and the results suggested that RTs were significantly longer for the unpredictable (novel) than the predictable elements. That suggests that early school aged children are sensitive to TPs during exposure and that RTs are a good measure to assess that observation.

However, a crucial critique to all the above studies as stated, is the fact that they assess to learning solely at the end of the learning process (Saffran, Aslin & Newport, 1996; Fiser & Aslin, 2002), and not while it occurs and develops. Especially, in cases that we deal with

sequential SL, that has a temporal element, a continuity in the examination of the mechanism is necessary. By creating a new methodology that captures precisely the SL process of each element of the sequence from timepoint 0 up to the end of the learning process (full learning curve), could bring new insights to the current SL models and provide more information about the encoding and chunking processes that occur during sequential SL.

Additionally, it is important to highlight some ecological validity bias over the implicit nature of SL that these methodologies consist of due to the implicit nature of their task. For example, many researchers in order to secure that the learning process was implicit during their familiarisation tasks, instructed participant to either listen passively to the stream or perform a cover task that is unrelated to the statistical regularities presented to them (e.g., Arciuli and Simpson, 2011). Even though it sounds like an ideal experimental manipulation, it isn't as it shifts the attention from the actual implicit task and can lead to increased cognitive load. In real life situations, there isn't a what we call a "familiarisation phase" but learning starts and is assessed by the first exposure to the TPs of the sensory input. Therefore, choosing methods such as 2AFC tasks, imitation tasks etc, are limiting the ecological validity of the findings of SL. The SL ability should be assessed in real life conditions, from time-point one, and constantly have an interactive feedback from the learning process. Experimental methods should have internal validity (Andrade, 2018) in terms of the design and conduct, but also external validity (Andrade, 2018), so that the findings and set ups can be generalised to other concepts, and more specifically, in the case of ecological validity, in real life SL processes and settings.

Serial Reaction Time Task (SRTT) is a popular experimental paradigm that has been used over time to explore unconscious learning processes (Hunt & Aslin, 2001; Robertson, 2007; Lee, Beesley & Livesey, 2016; Zhao et al., 2020; Kaur & Balasubramaniam, 2022). Nissen and

Bullemer (1987) created the first SRTT as a way to overcome issues previously raised within AGL tasks around the implicit nature of the tasks and its suitability to measure implicit learning (Reber, Batterink & Reuveni, 2019). Typically, during a SRTT participants are asked to respond to a specific cue in a stream of repeated fixed stimuli (e.g., press a particular button (response) every time they see the target stimulus appear on the screen (cue)). Cues' occurrence in the stimuli distribution are defined by probabilities, therefore can be learned and predicted by the participants, leading to faster reaction times. That means that as time progresses in the task, the participants unconsciously and unintentionally learn those probabilities and respond faster to the cue stimulus. When it comes to testing sequential learning with SRTTs, a commonly used structure of the task is (a) to associate each stimulus item with a button (S-R association) and (b) to present particular sequences of items (fixed sequences) within random sequences of items, so that RTs gradually improve (faster) for the fixed sequences than random. For instance, a common visuospatial SRTT task would involve participants following a sequence pattern on the screen which they are instructed to follow and repeat. The sequence of patterns occurs multiple times and is determined by probabilities, however participants are not aware of it. Participants are expected to show better reaction times during time on sequences that are determined by probabilities rather than random sequences.

SRTT has been used by researchers to explore unconscious learning processes such as implicit motor skill learning (Robertson, 2007), visual long-term memory and attentional selection (Zhao & Vogel, 2022), sequence learning (Dennis, Howard & Howard, 2006; Lee, Beesley &

Livesey, 2016; Kaur & Balasubramaniam, 2022), and statistical learning (Hunt & Aslin, 2001). The literature claims that SRTT is ideal to measure implicit learning (Hunt & Aslin, 2001; Deroost & Soetens, 2006), because it can be set up in a way so that it reveals no explicit cues about the probabilistic structure of the stimuli to the participants. Hunt and Aslin (2001) used an SRTT paradigm on the visual domain to investigate the ability of accessing and processing two separable statistical cues during sequential learning. In the first experiment they tried to replicate the findings of Saffran, Newport and Aslin (1996) but on a visual non-language domain. In order to be able to replicate the complexity of the word task used in Saffran et al.,1996 (6 trisyllabic words each with unique syllables = 18 syllables in total), but avoid adding visuospatial complexity to the task (having an array of 18 unique dots that would each need to be associated to a unique button so that RTs could be established for each item), they came up with an alternative structure. They designed an array of 7 illuminated buttons on a screen positioned in a semicircle. Each sequence consisted of 3 elements, and each element consisted of a unique pair of lights in the stimulus stream. In total, they created 21 unique button pairings (elements) that they placed into a sequence and created 7 sequences that contained triplets of pairs. Note that: (1) 21 pairs using only 7 lights means elements duplicated lights (e.g., 1-7 and 4-1); and (2) each pair of lights was illuminated at the same time. Across their three experiments they controlled for the predictability of the within sequence transitions, and the ratio of between sequence transitions, and the conditional probability and joint probability combinations. Their

results showed that participants can access more than one source of statistical information in anticipation of learning.

According to Hunt and Aslin (2001), SRTT is a better suited experimental task than 2AFC to investigate sequential probabilistic learning, as it allows to assess the learning of statistical probabilities across time. The SRTT allowed the researchers to control the statistical information that was available to the learners and at the same time record the progress of the learning of those statistics over time. Due to the structure of the task, they were able to observe two different types of statistics: (a) the bigram joint probability and (b) conditional probability. To understand the nature of those probabilities we need to examine the structure of the task. If in the SRTT, the pair (1-2) occurs 100 times and the pairs (2-3) and (2-4) occur 50 times, participants will learn that element 2 follows element 1 more frequently than element 3 and element 4 follow element 2. In this example, the joint probability of a bigram would be .5 for the pair (1-2) pair (it occurs 100 times while there are 200 pairs in total) and .25 for the pairs (2-3) and (3-4) (because each occur 50 times and there are 200 pairs in total). On the other hand, the conditional probability represents the probability of an element based on the first element in the pair. For example, the conditional probability that element 2 will follow element 1 is 1.0 because 1 is always followed by 2. The conditional probability that element 3 or element 4 will follow element 2 is .50, because 2 is followed half the time by 3 and half the time by 4. Therefore, it is possible to observe and measure more than one statistical structure in one task.

However, while the tracking of learning on SRTT is “live” and we can observe differences in the reaction time from timepoint 0, it is also true that we cannot refer to the direct learning of associations between each item of the sequence. For example, in Hunt and Aslin’s paradigm we don’t realistically observe the learning of a 3-item sequence, but the associative learning of a pair of lights (element) with the next element. But we do not have any information about how the actual element was learned. For example, in accordance to the SRTT paradigm used by Hunt and Aslin (2001), if we have a sequence consisting of 3 items (A, B, C) and A consists of (1,2), B consists of (6,1) and C consists of (8,5), then the actual exposure stream that the participant has in terms of minimal structural element is H-1-2-H-6-1-H-8-5 (H being the home button that was used so that prior to a response, participants began from the same start point). However, learning is potentially confounded by the need to use the home button in between every element of a sequence and the presence of paired (and duplicated) elements such as in 1-2 and 6-1. This information is very useful; however, it doesn’t decode how information processing occurs as a S-R occurrence in a sequence as the elements end up being combined into one higher order element (e.g., the pairs 1-2, 6-1 and 8-5). Therefore, a new methodology is needed that will allow the exploration of the above implicit learning concept that will allow the observation of the implicit learning mechanism over time but will also allow the retrieval of information about how the smallest component of each sequence is learned. Ideally, this new method would allow to observe learning not only between the 3 items of the sequence A-B-C, but also between the transition of its subunits 1-2, 2-6, 6-1,1-8, 8-5.

Furthermore, given the importance that reaction time has in a SRTT and how it is determining the learning in a task, when it comes to reaction time responses, eye-movements are more accurate than motor responses like pressing a button. An SRTT-like paradigm that would use eye-tracking, could source more precise reaction times but also provide us with details about the visual processing of stimuli, where the participant was looking during the task, if he was attending the task, if he was engaging with the task by looking at visual latencies and visual speed (Lange et al., 2018).

Another issue with serial reaction time is that each item needs to be associated with a specific button to press (so that reaction time to a specific item can be recorded). This means, for example, 4 sets of tri-syllabic nonsense words, each containing unique syllables, would require 18 buttons – 1 per syllable. Learning the correspondence between each syllable and each button would therefore be quite onerous even before any sequence learning can be assessed (and hence why studies such as Hunt & Aslin (2001) used the same lights within different elements, such as 2-1 and 1-6). Therefore, what is needed is a paradigm that takes the advantages of both methods and is able to record RTs while participants complete the learning phase, while ensuring that the number of unique items can be potentially unlimited.

C. SL models and the temporal element of sequential leaning

This subsection is devoted on understanding the various models that have been developed during the times to explain how SL works. Some of the most popular cognitive models of SL in cognitive psychology suggest that all processes involved in SL such as encoding/retrieving

are based on transitional probabilities (TP). If that is true, and there is no temporal difference between the elements of a sequence, then the performance of SL on each element of the sequence should be independent from the performance of SL of the element before and after within the sequence. However, chunking theory suggests that while TPs are being used to retrieve the information from the environment, the actual information is being grouped in units of information. If that is true, then it is expected a correlation between the performance of the first item learned in a sequence and the preceding/following ones. During chunking, the information is being grouped/ organised into meaningful units called "chunks". Those chunks are supposed to facilitate information processing (encoding, storage, retrieval). In the context of statistical learning, chunking theory refers to the idea that individuals can learn and recognize patterns in information by identifying and extracting meaningful chunks. Therefore, the TPs are used to identify and define the boundaries of chunks within the data. If for example, I am learning the sequence A-B-C-D, I will eventually extract the chunk ABCD. The chunk's representation will be reinforced every time I validate this chunk through exposure. This suggests that every time I am exposed to A, I have an advantage of retrieving B (based on the chunk ABCD, and the fact that the element A from the chunk was just validated), so performance on B will be better than performance A. Similarly, when I am exposed to B, I have an advantage of retrieving C based on chunk ABCD, and the fact that part of the chunk (AB) has been validated), so performance on C will be better than performance on B. Finally, when I am exposed to C, I have an advantage of retrieving D (based on the chunk ABCD, and the fact that part of the chunk (ABC) has been validated), so performance on D will be better than performance on C. This would end up giving us something like A<B<C<D in terms of performance. This outcome would also align with the fact that chunking in sequential SL occurs in hierarchical manner, meaning that the elements of the sequence maintain their order in the chunk during information processing (encoding, storage, retrieval).

C.1. SL models: the "all or none" learning processing.

The Estes Models: Learning occurs in a non-continuous manner (all or none), accounting for continuous learning by introducing the factor of stimulus – response connection (S-R). The idea is that you can form or not an S-R connection during an experimental trial, however you need plenty of these connections in order to be able to produce a correct response (exposure factor). In these models a probability of a correct response on an experimental trial is estimated as a proportion of stimulus elements conditioned to that response on the specific trial.

The Pattern Model (Estes, 1959): This model focuses on the learning of paired associations between S-Rs. It assumes that learning occurs through the extraction/formation of patterns, which are specific combinations of S-Rs, which are later on stored in memory. When individual see a new stimulus, they activate a specific pattern in memory (that they have previously learned) that matches the new stimulus. This activation spreads throughout the network of connected patterns, influencing the likelihood of particular responses associated with the activated pattern. The model incorporates mathematical equations to describe the learning process. These equations specify how the strength of associations between stimuli and responses changes over time based on factors such as reinforcement, repetition, and the temporal order of stimuli and responses.This model had the power to produce explicit formulas for many statistics. In 1961, Bower validated the pattern model by presenting explicit formulas for more than 22 statistics that derived from the model. However, in a pattern model, a former identification of proportion of conditioned stimulus elements with probability of a response is not allowed. Only one pattern (stimulus element) is presented on each trial. This suggests that either this model was wrong, or learning has a non-continuous nature.

Generalization of Pattern Model (Suppes & Atkinson, 1960): This model suggested that individuals learn by forming representations of patterns based on the environment and the

conditions that they are exposed to. These patterns are later on used to build the general characteristics of a category. Therefore, on when individuals get exposed to new stimuli, they compare these new stimuli with the characteristics of that category (similarities & differences) to decide whether they are similar or belong in the same category. Therefore, the generalisation of the pattern occurs via comparing the characteristics of a category with the new stimuli. If the new stimulus matches the characteristics of the category it will be generalised as a member of that category and if not, it will be generalised differently or not at all. This model was introduced by Suppes and Atkinson (1960) to allow a greater number of parameters in the model. While in the basic pattern model one parameter represents the probability of the response on a trial to be correct, in this model a different parameter is possible for each response.

The Incremental Model (Atkinson, 1961): This model was introduced by Atkinson (1961) and had a completely different approach to the previous models. The fundamental idea behind the model was that a stimulus element can be in one of finite number of conditioning states on any experimental trial. That suggests that on a single-stimulus version of this model with different number of possible responses, the maximum number of states varies with the number of possible response and therefore is not a simple function of reinforcement or motivation.

C.2. Key Models of Implicit Statistical Learning (ISL): From Chunking to Transitional Probabilities (TPs).

The literature considers SL and implicit learning as two separate areas of research (Perruchet, 2019). In their paper Perruchet and Pacton (2006) highlighted the fact that both SL and implicit learning have a lot of similarities in terms of learning situations; however, there is a big gap between the favoured interpretations about the selection of chunks, the formation of chunk

boundaries and the statistical computations associated with TPs. Perruchet (2019), explains this debate mainly by examining on the one hand the efficiency of statistical computations of pairwise associations (e.g. A-B, A predicts B) in explaining and predicting ISL and on the other hand the efficiency of chunk-based models in predicting ISL. It needs to be noted that these models do not operate in the same way and the main differences rely on their conceptual frameworks.

An example that has been used a lot in language learning literature and word segmentation research, are the Bayesian models. Bayesian models work by calculating statistical probabilities that will predict the likelihood of a predicted outcome. A Bayesian approach means that probabilities can be assigned to events that are neither repeatable nor random. Goldwater, Griffiths and Johnson (2009) explored how infants learn to segment words by extracting statistical regularities from the speech stream, from a corpus of child-directed speech. To do so, they used computational modelling to explore how different assumptions that the infants make about the nature of the word affects and predicts the words' segmentation. These computational models were Bayesian models that assumed that either words are independent units or that words are units that enable the prediction of other units. Their results suggest that the assumption of independence between the words predictability, led to under-segmentation of 2-word and 3-word sequences. However, when words were predictive the accuracy in segmentation was significantly higher.

Another popular model of chunking is the PARSER model (Perruchet & Vinter, 1998). PARSER suggests that learners are not extracting statistical relations about the stimuli from stream input, but instead they extract and represent statistically coherent chunks of information. Miller (1956) examined the chunking hypothesis within an AGL task. He suggested that when the learner gets exposure to the stimuli set for the first time, each letter in the sequence is represented and coded individually (A-A-C-D). However, as exposure increases, the letters can be coded to fewer chunks (AA-CD) and potentially form one final chunk (AACD). Servan-Schreiber and Anderson (1990) used an AGL paradigm but instead of letters they were trained on sentences. The sentences were divided in three conditions depending on their grammatical structure (well-structured, unstructured and badly structured). Their results suggested that participants were significantly better at both rejecting incorrect grammars and accepting correct grammars when the sentences belonged to the well-structured condition, rather than the unstructured and badly structured condition. This process was labelled as "competitive chunking".

Simple recurrent network (SRN) models or connectionist models (Christiansen, Allen & Seidenberg, 1998; French, Addyman & Mareschal, 2011) suggest that segmentation relies on the learning of statistical relationships between the items of a sequence. However, in contrast to previous chunking models, these models suggest that the statistical relationships do not represent the segmented units (Slone & Johnson, 2015). SRNs are a category of recurrent neural networks (RNNs) and have been used to explain learning in domains such as natural

language processing (Christiansen & Chater, 1999) and speech recognition (Amberkar et al., 2018). An SRN works on a feedback loop. It has an operational hidden level, that is the memory of the network, where it stores previous inputs. This hidden level is updated every time by combining the current input with previous inputs. Then the updated hidden level, is used to predict new outputs. This way, the network allows to capture statistical dependencies and patterns in sequential data. SRNs, due to their gradual build-up of representations over time, fall more into the category of chunking than TPs and have shown that chunking can account for language phenomena.

French et al. (2011) invented a new mechanism called "implicit chunk recognition" (ICR) to understand how sequence segmentation and chunk extraction work in ISL. ICR is novel as it removes the element of prediction of upcoming items in the sequence as a predictor of learning and instead it suggests that learning can be predicted by the recognition of previous encountered chunks in the input (familiarity & exposure element). To operationalise ICR, they came up with a connectionist autoassociator model called "Truncated Recursive Autoassociative Chunk Extractor" (TRACX). TRACX operates by extracting chunks based on truncated recursion. The key idea behind TRACX, is the assumption that people encode and retrieve information in chunks rather than as individual elements. A chunk is a unit of information that represents a meaningful or familiar pattern of items. For example, in a sequence of numbers, the chunks could be digits that are part of a larger pattern or concept. TRACX uses RNNs to simulate the processes of chunk formation and activation. The model

consists of a network of interconnected units, where each unit represents a chunk. During learning, the model is exposed to a stream of information, and it automatically detects and encodes chunks by identifying regularities and repetitions in the input. TRACX has proven to outperform the robustness and accuracy in performance of other chunking-based models such as PARSER (Perruchet & Vintner, 1998) or SRN models (Cleeremans & McClelland, 1991) in matching human sequence segmentation data.

Another popular model that has been used to explore ISL, in language learning is the 'Comprehension And Production Performed Using Chunks Computed Incrementally, Non-categorically, and On-line' (CAPPUCCINO) model (McCauley & Christiansen, 2011). The CAPPUCCINO model was inspired by differences in perceptions of linguistic productivity by the generativists (Pinker, 1999) and the usage-based approaches to language (Tomasello, 2003). Linguistic productivity is the capacity that a language has to generate new expressions through its rules and structures. More specifically, it refers to the ability of the speakers of a language to generate and understand an unlimited number of novel utterances, that they haven't interacted with before (encountered or memorised) and have meaning (Pinker,1999; Diessel, 2017). But in order to understand better the argumentation behind linguistic productivity and CAPPUCCINO models, one must review the two most prominent approaches around language learning and linguistic productivity: (a) the generativist approach and (b) the usage-based approach.

On the one hand, the generativist approach (Chomsky, 1957; Pinker, 1999) focuses on the innate and rule-based nature of language and the creativity of speakers to generate and understand an infinite number of novel sentences (Chomsky, 1957). Generative linguists (Pinker, 1999; Pinker & Jackendoff, 2005) argue that speakers have the ability to combine and recombine a finite set of linguistic elements, such as words and grammatical structures, according to the rules of their language (transformational generative grammar). By applying these rules, speakers can generate an unlimited number of grammatically correct sentences, including those they have never encountered before. On the other hand, usage-based linguists like Tomasello (2003) and Goldberg (2019), suggest that productivity emerges from the usage patterns and frequency of linguistic constructions in everyday communication. According to them, linguistic productivity is not solely the result of innate knowledge or rule-based mechanisms but is heavily influenced by the cognitive processes that are involved in language use and the exposure that the speakers have to those specific constructions. The CAPPUCCINO model (McCauley & Christiansen, 2011) was designed to test the usage-based approach to children's language learning by focusing on stored chunks of information. According to McCauley and Christiansen (2011):

> "To this end, the model gradually builds up an inventory of chunks consisting of one or more words—a 'chunkatory'—used for both language comprehension and production. The model was further designed with several key psychological and computational properties in mind: a) incremental learning: at any given point

> in time, the model can only rely on the input seen so far (no batch learning); b) on-line processing: input is processed word-by-word as it is encountered; c) simple statistics: learning is based on computing backward transitional probabilities (which 8-month-olds can track; Pelucchi, Hay,& Saffran, 2009); d) comprehension: the model segments the input into chunks comparable to the output of a shallow parser; e) production: the model reproduces the child's actual utterances; f) naturalistic input: the model learns from child-directed speech; g) cross-linguistic coverage: the model is exposed to a typologically diverse set of languages (including Sesotho, Tamil, Estonian, and Indonesian)." (McCauley & Christiansen, 2011, p. 1619-1620)

One of the strengths of CAPPUCCINO model (McCauley & Christiansen, 2011) is the fact that is more realistic to natural language processing. It allows to observe language processes such as comprehension and production, as meaningful units of information (chunks) that are being processed gradually and are being integrated incrementally. This processing is done in a flexible, context-dependent manner, without rigid categorization. It occurs in real-time, allowing for continuous adjustment and updating as new information becomes available. McCauley and Christiansen (2011), managed to replicate the findings of Safrran (2002) of real child data on SL. Moreover, their findings supported a usage-based approach of language where distributional statistics are crucial for predicting linguistic outcomes in children. They found that word-based distributional information were better outcome predictors than word class statistics. This finding is in agreement with the previous findings in the field that support

a usage-based approach. For example, Monaghan and Christiansen (2008), found that both distributional and phonological information can significantly predict the learning outcomes of lexical categories and phrase structures. These findings are in agreement with the behavioural concepts that Tomasello (2003) proposes against the nativist approach to language, suggesting that language acquisition is a social-cognitive process that highly relies on the children's interactions with their caregivers and other language users. Language learning in children occurs via constructing and understanding utterances based on their communicative intentions and the linguistic input they receive.

C.3. Sequential SL and temporal information processing.

As the nature of statistical learning is by definition mathematical and based on the probabilistic nature of events, statistical learning has been used in numerous models trying to encode human behaviour. Some of the most popular models that used statistical learning in applied cognitive concepts, were the popular mathematical models of memory (Norman, 1970) and the multistore model of memory from Atkinson and Shiffrin (1968). In both models, stimulus sampling theory (SST) (Estes, 1950) and the probability of certain stimulus occurring in a certain time period and therefore the pairing of that stimuli with a given response, was the key element component.

More recently, there is a constructive debate around the area of SL focusing on the hierarchical component of it. It seems as there is a shift from understanding SL processes as sets of memory processes, to perceiving SL as a single mechanism. Many studies in the field suggested that statistical learning and rule-based learning are two different mechanisms since SL refers to the ability to learn stimuli that you have been exposed to, while rule learning can be generalised therefore be applied to novel stimuli and new combinations (Marcus, 2000; Endress & Bonatti, 2007). However, Aslin et al (2012), argued that this hierarchical perspective of learning is

wrong and that there is a single statistical learning mechanism that can account for both learning of input stimuli and generalisation of learned patterns to novel instances. The key element that differentiates the learning outcome during the learning process is the perceptual properties of the stimuli (for example verbal) and not the actual learning mechanism. They also underlined that there are two factors between instance-learning and generalization phase. These factors are the strength of perceptual and cognitive components of the structural regularities, and the consistency of elements' contexts (unique vs. overlapping).

In the field of applied cognitive linguistics, Arciuli (2018), examined whether reading can be thought as a process of learning statistical regularities and how that component of reading can be used and applied in improving teaching methods of reading in schools for both typical learners and those ones with developmental disabilities. This approach used a combination of theoretical, behavioural and computational concepts in order to demonstrate that implicit methods that are based on the SL principles can be used as a supplementary method to the explicit (rule-based) method with positive results on children's reading ability. It additionally highlights the idea of conceptualising learning as continuous process that has components in different perceptual and cognitive structure levels.

Du & Clark (2017), were aware of that gap in the literature, and the lack of clarity about the initial acquisition of sequences and if it's a result of chunking learning or TPs learning. Therefore, they decided to examine each sequence holistically, by focusing on the temporal dependencies of the entire sequence that could disclose the representations of chunks but also TPs. In their task, participants performed a serial reaction time (SRT) task under different stimulus interval conditions. During the SRT task a visual cue appeared on the screen, and the participant had to respond by moving his feet (stepping) on the appropriate location, then the visual cue disappears, ending the trial, and after a fixed delay, another visual cue appears marking the beginning of a new trial. The SRT task was performed under one of three stimulus

interval conditions (see Figure 3.2). In condition I, the trial consisted of each stimulus being presented for 700 ms and then the next stimulus appeared after an interval of 600 ms (700 + 600 ms), creating a period of 1300-ms-long interstimulus-interval (ISI). The time intervals were reduced at 900 ms for condition II (700 + 200 ms) and condition III (300 + 600 ms) for condition II and 300 + 600 ms for condition III. During the task participants did a 3-minute break after completing each block while no information or instruction was provided about the sequence presentation.

Their results suggested that sequence learning can be reflected by reaction time (RT) rather than the improvements represented by movement time. The temporal dependency of RT and movement time revealed that both RT and movement time displayed repetitive patterns caused by biomechanical effects of response locations and foot movements. Chunking was only noticed in the presence of the recurring RT or movement time and disappeared right after the foot was relocated, suggesting that the chunking observed was related to the biomechanical constraints rather than learning itself. The most important finding of that study was the fact that trial-to-trial associations were strengthened as learning progressed regardless of stimulus intervals. That finding could reflect internal cognitive representation of the first-order stimulus contingencies.

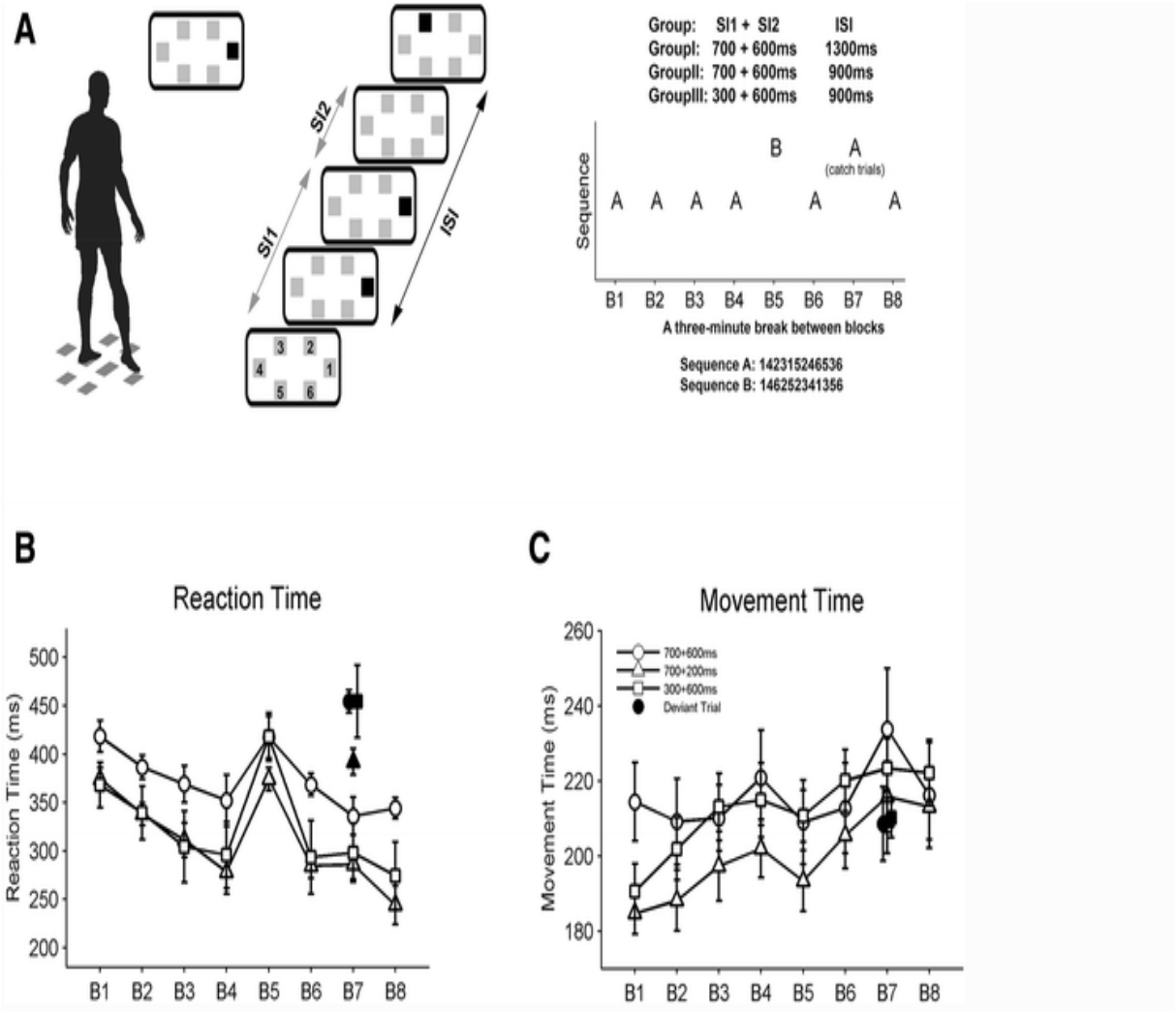


Figure 3.2 Figure from the Du & Clark (2017) method's section. **a** Experiment procedure. **b** Mean RT across learning blocks. **c** Mean MT across learning blocks. *Error bars* represent standard errors.

D. The importance of sequential SL as a cognitive component of learning ability.

Essential characteristics of skill learning include the need for encoding, representing, and producing structured sequences (Conway & Pisoni, 2008). In real- life situations the environment includes regular sensory stimulation, of sounds, objects and events. These sounds, objects and events have a specific place and duration in time, and in order for any organism to successfully adapt to its environment should first successfully learn these environmental structures (Conway & Pisoni, 2008). They suggest that the regular coherent sensory stimulation can be viewed as probabilistic structure patterns. These patterns are part of almost all aspects

of our sensory interactions with the environment and the real-world, and can be part of processes such as speaking, hearing, learning a new skill, or perceiving complex scenes (across modalities).

Lieberman, Chang, Chiao, Bookheimer and Knowlton (2004), suggested that this automatic knowledge of sequential patterns and regularities is beneficial for the learning process and facilitates the spontaneous activation of associated representations to be triggered by the mere presence of the sequential cues (Lieberman et al., 2004). As P. J. Reber (2013) states, "We should expect to find implicit learning and memory phenomena whenever perception and/or actions are repeated so that processing comes to reflect the statistical structure of experience" (p. 2029). However, understanding the process of how the information encoding occurs and uses these probabilistic structures constantly occurring in our environmental surroundings through both space and implicit Cognition remains an open challenge (Conway & Pisoni, 2008).

Before going deeper in understanding the components that can influence the SL process, it is important to acquire a better understanding of the nature of SL as a cognitive mechanism. That will allow to understand why certain factors affect SL and why others not. Statistical learning (SL) a domain-general mechanism which detects the underlying distributional properties of the input. In one hand the domain general approach, suggests that SL is unitary learning system (Bulf, Jonhson & Valenza, 2011) that operates the same way over similar types of perceptual input across different domains (Saffran and Thiessen, 2007) and species in the same way (Hauser & Aslin, 2001). On the other hand, plenty of studies demonstrate that SL has a domain specific component, as there are qualitative differences in SL learning outcome in the auditory, visual, and tactile modalities (Conway and Christiansen, 2005; Frost, Armstrong, Siegelman & Christiansen, 2015). However, Frost et al., (2018) , suggest that these observed differences rely on the fact that different modalities bring different knowledge and as a result they are bound to

give different learning outcome even if the mechanism is domain-general. As Frost, Armstrong, Siegelman & Christiansen (2015) suggest, future SL researchers should target into perceptual properties of the sensory information, in order to understand better the differences across the domains, since SL is not an unitary mechanism and therefore uses different domain general mechanisms and specific brain networks during SL and information processing in different domains.

A described above, most of the conflicts around the nature of statistical learning come from an approach that either ignores completely the perceptual properties of the sensory information or is looking for a minimalistic mechanistic explanation. In order to understand SL we need to understand each part and level of the mechanism and how each level of the mechanism interacts with each other and with the final statistical learning outcome. It is more than possible that SL is a mechanism that involves or interacts with other perceptual mechanisms on an early processing stage before moving into the regularity extractions, and therefore can have a dual nature. It can be domain general on the regularity extraction and creation of perceptual units, however it is also domain specific to modalities and stimuli affected from perceptual properties of the stimuli (exposure, cognitive load, length of the pattern regularities, linguistic information, etc) that might cause facilitation or delay on the "pre-statistical level" or "probabilistic level" of the mechanism that is purely perceptual and therefore can cause actual delay or slower performance on the statistical learning outcome unit.

E. Properties that affect SL: evidence from research

One recently popular computational theory in the field of SL, states that SL doesn't rely on TPs, but instead is using a mechanism of elements' extraction from the sensory input into the memory traces, that leads into an integration across these memory traces that highlight consistent information (Thiessen & Pavlik, 2013). This approach created a huge research

interest in the field of developmental psycholinguistics, cognitive linguistics, and cognitive psychology. A great number of studies have shown that SL is one of the basic mechanisms that are responsible for the language acquisition and vocabulary growth (Lew-williams & Saffran, 2013; Jones et al., 2018). Most of the research being done around the SL and language development, focused on infancy as it is the crucial age period that humans start acquiring their first words.

Similar to the Thiessen's and Pavlik's (2013) research, was conducted on SL and word segmentation on adults by Frank, Goldwater, Griffiths and Tenenbaum (2010). They tried to explore the SL ability on word segmentation while controlling for sentence length, exposure and number of word types. Even though the behavioural data showed a clear effect of those components on SL performance, suggesting that longer sequence length, less exposure (less frequent stimuli) and more language (greater diversity) make language learning harder, their computational proposals failed to replicate those findings. These findings could lead future research into two different research directions: either (a) focusing on creating more complicated SL models that will capture various perceptual components of stimuli (such as verbal information) or (b) creating simpler tasks to explain basic mechanism of SL, which will directly infer to implicit processes and include perceptually simpler stimuli.

A main corpus of the literature tried to provide a clearer view about which are the actual stimulus properties that are making SL harder. Most of them (Saffran et al, 1996; Saffran, 1996), concluded that SL operates the same way on visual and auditory input (Aslin et al, 2012), and that any differences observed in the learning outcome, are effects of exposure, familiarity, novelty of the stimulus. More specifically, Aslin (2017), suggested that the main implicit mechanism that infants develop their learning is SL and it is operating by mere exposure. In addition, this exposure can generalize to completely novel information, and enable the transfer of knowledge. Those findings suggest that since there are no differences observed

between visual and auditory SL tasks, any manipulation of exposure, familiarity and novelty will be equally affecting the SL on both domains. In other words, since SL is part of the language learning (Perruchet, 2018), any knowledge around the mechanism and restrictions of SL on the visual domain can equally reflect on language learning process. SL is a mechanism that has been associated with language learning and word segmentation across humans (Saffran et al., 1996) and non-human primates, such as tamarins (Hauser et al., 2001; Santolin & Saffran, 2018). However, SL is not a language specific mechanism. It's a domain general mechanism that operates across all modalities (auditory, visual, tactile) and can be observed in cognitive processes that involve the segmentation of continuous sensory input (Conway & Christiansen, 2005; Polyanskaya, 2021), and as it appears language learning relies strongly on this mechanism.

Frost and Monaghan (2016) suggested that the same SL principles that underlie speech segmentation, also underlie processes such as grammatical generalisation. In their study they used an artificial language paradigm, with language structures that relied on non-adjacent dependencies. Participants could successfully segment the words in the speech stream, but they were also able to generalise their knowledge about the structure of novel speech that they haven't been exposed to previously.

Robinson and Pascalis (2004), suggested that the development of visual recognition flexibility in infant's memory, relies on hippocampal development of infants and no differences in performance are observed after the age of 24 months old. To test so, they compared 4 different age groups: 6-monthsold, 12-months old, 18-months old and 24-months old infants. The task consisted of a familiarization where infants were presented with toys images with a specific background colour and a testing phase that the toys were presented on a different background colour. They found that recognition memory is impaired by a change in context at the early age of 6 and 12 months but remains unaffected after the age of 18 months, suggesting again that

there is a developmental constraint in a mechanism highly related to SL, supporting the theoretical approach of SL as a non- unitary mechanism (Frost, Armstrong, Siegelman & Christiansen, 2015), but also challenging all the developmental SL studies, about the complexity of the stimuli used. It could have been the case that Frost et al. (2018) highlights, that the previous knowledge to the stimuli used (e.g picture of toys) varied across the developmental spectrum of infants used for this study and that's where the differences in performance rely on.

Another important element that has been shown to affect SL outcome is the sequence length. Sequence length can be considered as a main property of perceptual and cognitive structure of the element. The most elementary example of sequence length effects on language learning is the fact that infants learn short words (one and two syllables words) faster and sooner than longer ones (three or more syllables) (Lew-Williams et al., 2011; Saffran, 1996).

Additional research in the field of SL and artificial languages, found that regular (most common) units length of verbal and nonverbal artificial languages were learned better than irregular (less common) units of length, suggesting an effect of prior knowledge/exposure of sequence length on the SL outcome (Hoch, Tyler & Tillman, 2013). However, there is some literature suggesting that the mechanisms supporting implicit sequence learning are not capacity-constrained by sequence length nor adversely affected by high rates of irrelevant sequences during training (Sanchez & Reber, 2012), but there is an obvious debate about if the learning processes involved in the tasks used were implicit and could be considered as good example of SL or it was a rule-based learning process.

Sanchez and Reber (2012), were interested in investigating information processing constraints such as sequence length, as they recognised that implicit learning system has distinct operating systems from the explicit learning system, and therefore relies on different brain areas. They

used a serial interception sequence learning (SISL) task with covertly embedded repeating sequences that were much longer than most previous studies (30-60 exp1, 60-90 items exp 2, 12-item repeating sequences were embedded among increasing amounts of irrelevant nonrepeating sequences). During the SISL task, participants were presented with circular cues that scrolled vertically across the screen towards one of the target zones that were marked as rings. In total there were 4 target zones. Participants had to give a key-response by pressing the D, F, J, or K on the keyboard and be as fast as possible in order to press the button while the cue moves through the target zone. Dual-button responses, were only used in Experiments 1 and 2, while in experiment 3, the targets were presented at the bottom of the screen and the circles scrolled downward. Their results suggested robust learning for sequences up to 80 items in length. However, due to the SISL task, their results could be interpreted as intermediate associations of 4 corresponding keys (D, F, J, or K) and not direct association learning of the items within the sequences.

In addition, Lew – William et al (2011), found that prior knowledge on sequences length (for the language domain- words length) can enhance learning of the input, only if the input is sharing same length properties. This is an extremely interesting approach if we consider its implications on real world conditions. In real world, verbal, auditory and visual information in the learning and sensory environment consist of sequences of stimuli with mixed lengths but humans still manage to learn those properties efficiently. Currently there isn't enough research that looks learning of mixed length sequences within a task. Johnson and Tyler (2010) investigated the concept of language learning and word segmentation with the use of a SICR task, that involved mixed and non-mixed length sequence tasks. More specifically, they investigated the sequence length factor on infant's language ability by using artificial language and demonstrated that 5,5 and 8 months old infants can successfully segment uniform length words (sequences of words with 2 syllables (CVCV)) but failed to segment the language that

contained words of varying length (2 syllables (CVCV) and 3 syllables (CVCVCV) mixed). However, because the task wasn't looking into the timecourse of the SL process and focused only on the SL outcome, it is unclear why this difference in performance was observed and if it relates to a specific chunking theory. Therefore, this paper creates a great opportunity for future research to look into mixed length effects on learning and how sequence length knowledge can affect the predicting ability and so on learning ability of mixed length sequences, while recording the actual learning time course of the SL process and not only the SL outcome.

F. Gaps in the literature

All the evidence above suggests that: (a) implicit learning is still an unknown mechanism, (b) most of the tasks that investigate SL are assessing the learning outcome rather than the process itself, (c) there is a temporal component in SL that is still unexplored, (d)there are factors that affect SL outcome that need further investigation such sequence length and sequential mixture types of SL tasks.

SL is part of the implicit learning mechanism of cognition and plays an important role in main aspects of everyday life of human and non-human species. Understanding how it operates could be beneficial for educational, developmental and AI reasons, as it could bring improvements with its applications in smart AI systems, to populations with learning difficulties or disabilities, and support the cognitive development of populations that have received unequal opportunities (e.g social factors - Low Socioeconomic Status, or disability- hearing impairment).

Focusing on understanding SL while it occurs, can give a clearer image about how information processing occurs, why some elements or the task are being learned better over others, and also provide us with growth learning curve that actually represent the learning process itself and not

other perceptual/ cognitive load factors. Additionally, by recognising the temporal element of SL and showing how each element in a series of events (sequence) affect the learning of the whole sequence, can help us understand better how language learning occurs during infancy, and help us develop specific tools to facilitate different SL processes in language such as word segmentation, reading and spelling.

Furthermore, environmental effects such as sequence length and sequential mixture type need to be further investigated and accounted for, in order to understand how SL works in natural and artificial languages. Language consists of different linguistic structures such as syllables, words, and sentences. If the aim is to understand how language is being learned, it is important to be able to understand how word learning is affected from longer and shorter words, and how sentences that are formed with mixed length words can be learned faster depending on their consistency.

In the chapter that follows (Chapter 3), the research aims of this thesis are being identified. The current gaps in knowledge around the area of SL, led to the need for the development of a new methodological paradigm for SL, that will be able to observe and assess, both the learning outcome and the learning process of SL mechanism. This new methodology will be fully detailed in Chapters 4 and 5, and will be used to answer specific questions in regard to how learning occurs across same length and mixed length sequences, and what are the current chunking mechanisms observed during SL.

Thesis Research Aims

The present thesis is devoted on developing a new methodological paradigm that investigates in depth the impact of factors such as sequence length and item positioning on sequential statistical learning (SL). SL is a strong but still unknown mechanism, that is widely used as a conceptual construct of research areas such as cognitive linguistics and language acquisition, learning and memory, visual and auditory perception, artificial intelligence, bioinformatics and machine learning. However, for the purposes of this thesis, SL will be investigated and perceived as a psychological cognitive learning mechanism.

In the following chapters the interest will be focused on providing a different experimental research approach on SL that will control for exposure and stimuli complexity, while we manipulate properties such as the length of the regularities and the mixture of regularities, to observe differences in the SL outcome performance. From now on and for the rest of this thesis the statistical regularity patterns will be called sequences.

More explicitly, the research aims of this thesis are:

1. **Can an efficient design be created that tap into the process, and the development of learning with exposure during SL?**

   We will see in the literature review in the next chapter that there are problems with current methods of examining SL, such as assessing the learning process only at the end point of learning or using tasks that can't fully capture the implicit nature of SL due to the experimental design (e.g. complexity of stimuli) or procedure (e.g. forced-choice task that goes against the implicit nature of the mechanism). The method proposed in this thesis will overcome these problems by creating a new, more ecologically valid paradigm of SL that uses minimal sensory input. That new method should be able to answer *(a) if sequence learning is achievable through a completely*

implicit task and (b) what is the minimum exposure needed for a sequence to be learned via a sequential SL task. More specifically, this task should be able to pick up about how different independent variables such as sequence length, and mixture of sequences effect human SL performance. It will record the development of learning as it occurs, from timepoint 0 - until the end of the task and will be able to tap into the actual procedural mechanism of learning during SL, rather than assessing solely learning outcomes on a trial-by-trial basis. The design should try and keep a minimalistic approach in terms of the sensory input across the tasks to avoid increased perceptual load and minimise any influence of prior knowledge on the task. Additionally, it should resemble a real- world learning paradigm.

2. **What is the time course of learning during a sequential SL task?**

   SL has by its definition a temporal nature. In the literature review we will see that many studies do not consider the time course of learning but the end product of it. Since sequential SL has a temporal component, the holistic perception of learning (all or none) of the parts of the sequence may not be valid, because each element of the sequence occurs in a unique timepoint and that timepoint is related with the presentation of the next item suggesting a serial process of learning. Therefore, *differences in the learning outcome are expected to be observed between the items of each sequence depending on their temporal order in the sequence*.

3. **SL has been examined in depth in the language domain, yet there are no clear answers to basic questions such as:**

   a) Does sequence length affect the performance in SL?

Again, as we will see in Chapter 2 while we will be reviewing the literature, sequence length is a basic experimental characteristic of the SL process, yet there is contradictory evidence about how it affects the actual learning process, depending on the nature of the sensory input and experimental design. A design as suggested above should be able to *identify length effects on SL performance and provide a clear answer about how the length of a sequence affects the SL outcome.* If SL is a solid mechanism that is not affected by such things as cognitive load or working memory capacity, then the performance across the different length sequences should be the same, and as a result the learning rate should be the same. For example, if SL has a memory capacity component and is affected by the properties of the stimuli then, sequence length effects will be observed, with shorter sequences being learned better than the longer ones due to less cognitive load. That would result into a faster and better learning rate of shorter over longer sequences.

b) How does SL occur in mixed length sequences?

Most of the current experimental tasks that are involving SL, use same length stimuli. However, that is problematic as it lacks of ecological validity. In real world situations we are called to perform SL processes in a wide range of sensory input that contains mixed length sequences (e.g word learning). This new methodology should be able due to its design to *(a) examine if mixed length sequences can be better learned than same length sequences, (b) demonstrate how mixing up the sequence lengths affects the learning of other length sequences* and *(c) highlight any limitations on the learning process that derive from sequence length*. Blending shorter with longer sequences can either facilitate the learning within the task, make it harder, or not affect the learning at all.

In order to investigate the proposed research aims, a novel eye-tracking task that enables eye-movement recordings throughout the experimental process will be designed. As Aslin & Newport (2012) suggested, there are no differences observed on the SL outcome between visual and auditory stimuli. The leading hypothesis that SL is unaffected by modalities lead to creating a series of visual SL tasks, to provide access on processes that occur during sequential SL, that would be harder to capture, if auditory stimuli were being used.

The suggested design that this thesis proposes, suggests that every observation that is captured during the visual SL tasks can be equally reflected on auditory SL tasks and more specifically language sounds. Language includes words, that are formed with syllables. If the design manages to capture the processes of SL during sequence learning of visual stimuli, it could be later used in future research to understand how sequences of sounds (e.g., syllables, words) are learned and get access on the basic language acquisition mechanism and be applied on educational methods, and artificial intelligence systems.

The new experimental paradigm used the visual modality, as it is the most research-wise investigated modality (Hutmacher, 2019), and tried to resemble a sequence learning paradigm that could occur in a language learning paradigm (Saffran et al.,1996) in the visual domain. The suggested design kept the nature of the task as implicit as possible and introduced the novelty of a gaze-contingent eye-movement paradigm on a “guess where the next dot will appear next task”, while recording a continuous stream of eye-movements from the beginning of the first trial until end of the task. This technical manipulation allowed to observe learning processes throughout a task, when exposure to the visual sequences start at time 0 and how the learning mechanism builds up knowledge about the visual sequences and pairing throughout the task while controlling for exposure. The main factors that were manipulated throughout these tasks were sequence length and mixture of sequence lengths within a task. In order to

validate and establish the efficiency of the new methodology suggested, only adult typical population with not known learning disabilities were recruited.

New experimental design: Primary design, piloting data and considerations.

Chapter Summary

In this chapter, the reader will be given a full examination of the new methodology suggested for investigating sequential statistical learning. In the following sections the reader will get a summary of the current methodological issues around sequential statistical learning and the proposed methodology that will overcome those issues. The methods of this chapter contain the primary experimental design (Design A), and relevant piloting data that led to the creation of Design A. A justification will be given about the various choices of experimental settings. Finally, in the discussion, an evaluation of Design A will be given. The critique of the primary design will lead to the creation of the final design that will be presented in the next Chapter (Chapter 5).

A. Introduction

As it was demonstrated in Chapter 2, statistical learning (SL) has been argued to be one of the main mechanisms for processes like language acquisition, implicit learning from the sensory input (visual and auditory modality) and sometimes it has been used as a measure of cognitive function. Additionally, SL has been used as a way to understand better learning disabilities such as dyslexia or ADHD. Throughout these fields, a set of experimental methods have been used to understand SL and how it operates, mainly by focusing on the outcome of the procedure, rather than the procedure itself. However, these methods can sometimes restrain the full examination of the SL mechanism, and often lack ecological validity.

Currently, there is an emergent need in the field to create a new method that will (a) examine SL during time course, (b) have an implicit character, (c) allow manipulations to the stimuli, such as to sequence length and mixture length effects in SL and finally (d) provide some insight about the information processing/chunking during SL in order to be able to access the mechanism itself rather than the outcome of the process. In line with the dominant view that SL is a domain general mechanism (Kirkham, Slemmer & Johnson, 2002; Thiessen, 2011), this new methodology was designed on the visual domain, by introducing a gaze-contingency eye-tracking paradigm during sequential learning task. However, the technical design (experimental coding) of the suggested method, allows the introduction of auditory stimuli too in the task, allowing the investigation of auditory SL in the future. But in order to avoid multisensory perception cognitive load, and intermediate associations between the visual and the auditory stimuli, it was decided that the visual domain was the best way to test this new method and see if it can capture the actual SL process. Therefore, the new experimental paradigm involves learning sequences of locations in a semi-random array (not a grid) of identical images (dots). Dots are highlighted in sequences and participants are asked to

anticipate and look toward the location of the next dot in the sequence. Sequences are presented in a continual stream, with no indication of when a new sequence starts, and participants are only instructed to "Guess where the dot will appear next" by looking at which dot in the location array they think will be illuminated next. These instructions secure that the implicit nature of the task remains implicit and that the participants are not affected by directional explicit rules about the stimuli or the learning process. No information about pattern extraction or learning of sequences is given to participants. Participants are instructed as if the task is a single stimulus response (SR) task on a trial basis, and they do not have to make associations about prior trials. Leaning is determined by studying eye movements from timepoint 0 until the last trial of each task. There's no separate familiarisation and testing phase in each task. Obviously, the new method is inspired by previous experimental paradigms, however it was constructed in such way to overcome previous problems and combine the benefits of previous methods.

One of the most common methods used to examine SL in infants is to measure the duration of "preferential looking" suggesting that looking longer at a stimulus suggests its novelty. Saffran, Aslin & Newport (1996), investigated SL and language acquisition mechanisms on 8 months old infants by using that "preferential look" method. Their task consisted of a familiarisation phase, where infants were exposed to a continuous speech stream string (e.g. kupadilagotubidako), that consisted of three-syllable chunks/triplets (e.g. kupadi, lagotu, and bidako). The order in which these triplets occurred was free and therefore, transitional probabilities (TPs) were structured such that TPs from one syllable to the next were higher for stimuli within a triplet (e.g., padi) than for those that span a triplet boundary (e.g., dila). During the testing/preferential look phase, infants were presented with a word that was included in the familiarisation phase (e.g., kupadi) (familiar triplet) and a new one (e.g., gadilo) (novel triplet). Longer reaction times (preferential look) indicate non-learned items, and in this case the novel

triplets had significantly longer RTs than the familiar triplet. Even though this methodology sounds ideal for conducting language learning SL tasks in young infants, it has its limitations. Firstly, it is restraining as it assesses the learning at the end of the process and doesn't allow any further details about how the learning occurs and if infants extract TPs or perform information chunking. Additionally, it is limiting the possible answers to two options, as any 2AFC task would do, by increasing error variance that can not explain facts such as individual differences in learning. However, it was used as an inspiration of the current suggested new methodology, by introducing eye-tracking as the main technical equipment that would be used in the new design. Eye-tracking is one of the most well-established methodologies used in the field of psychology, suggesting that there is strong justification for the outcomes of the study. The idea that eyes can provide us with details about different cognitive processes, is a key component for the new methodology suggested as eye-movements can allow an implicit experimental design without introducing explicit task introductions, just by observing how, when and where the eyes move. That indicates that processes such as SL can be observed within more ecologically valid experimental set ups.

However, this is not the first time that someone is suggesting an eye-tracking study to assess a mechanism as basic as SL. Memory for example, especially visual memory has been studied many times by the usage of eye-tracking in adults, children and infants (Oakes, Baumgartner, Barrett, Messenger & Luck, 2013; Chevalier, Blaye, Dufau & Lucenet, 2010). It is known that infants acquire knowledge from experience with the world by using their senses. According to Eichenbaum (1997) the memory retrieval has 'representational flexibility', suggesting that it occurs despite changes in the environmental cues. Plenty of studies have shown age related changes in flexibility in various tasks such as (a) mobile conjugate tasks in 3-6months old where infants learn contingency between actions. During a mobile conjugate task, the researchers tie the one side of a ribbon around the ankle of the infant and the other side of the

ribbon on a mobile that hangs on the top of the infant's crib. In this set up the infant learns that the kicking will cause the movement of the mobile. During this process the focus of eye-movements and the durations of the eye-movements of the children are being recorded (Rovee-Collier, 1999; Hayne, Greco, Earley, Griesler & Rovee-Collier, 1986; Rovee-Collier,Griesler & Earley, 1985), (b) the delayed imitation task with 6- to 24-month-olds: during this task, the researchers use actions such as manipulating objects or using body movements, and expect infants to imitate those behaviours. To do so, they record the eye-movements of the children to see where they fixate their attention during this imitation process. Then they test if infants can imitate the actions they observed after delay (Hanna & Meltzoff, 1993; Hayne,Boniface & Barr, 2000; Hayne, MacDonald & Barr, 1997), and (c) visual paired-comparison task with 6- to 24-month-olds: this task assesses visual preferences and discrimination abilities of infants. During the task, the researcher present pairs of visual stimuli to infants and measure their looking behaviour towards each stimulus (where they fixate, for how long etc), (Robinson & Pascalis, 2004). Based on these abilities Richmond and Nelson (2009) used a scene/face paradigm to access the nature of memory representations and memory encoding during infancy, using a Tobi eye-tracker to measure infant's fixations. Their results replicated previous findings in adults in terms of the magnitude and the time course of the preferential looking, despite the differences in stimulus exposure between infants and adults.

Even though eye-movements and gaze RTs provide strong evidence that eye-tracking can be a good way to examine basic cognitive mechanisms, across different age ranges in the developmental spectrum, also highlight the fact that certain experimental designs can be restrictive in the quality of information that they collect, even when using powerful technical methods such as eye-tracking. One of the main problems of the above methodological set ups, is that they focus solely on the outcome of the learning process, while ignoring any information about the learning time course, while claiming that the findings reflect an aspect of the learning

process. The preferential looking (spending more time on fixating on stimulus x rather than stimulus y, as x is recognised as novel) for example, assesses behaviour at the end of the learning process (or if there are intermediary preferential looking tasks, they take a snapshot of learning at different timepoints). That suggests, they do not examine how learning progresses but just capture parts of the learning process (usually at the end of a training phase). However, in processes such as sequential SL it is important to obtain a full image of the process to understand how each element of the sequence is associated with the previous one, and it's not a learned vs non- learned process as a "2AFC task" would suggest. That's why the current suggested experimental set up was designed to allow a gaze-contingent paradigm from the beginning of the task up to the end of the task, to be able to infer to learned or non-learned items of sequences but also to the relationship between the items of the sequence. Furthermore, the current design in future could be easily adjusted to examine eye movements during different experimental phases such as the training phase and the preferential looking (or testing) phase, to capture the holistic process of learning, but also allow comparisons between the different phases of learning.

Similarly, another methodology used in the SL literature is that of "familiarization" (Slone & Johnson, 2018; Siegelman, Bogaerts, Elazar, Arciuli & Frost, 2018). In this methodology, novelty is being treated as the key element of research focus and differences between familiar and novel stimuli can reveal differences between the conditions. However, this methodological paradigm on its own isn't efficient, as the exposure needed for each stimulus to be learned might differ depending on the perceptual complexity of the stimuli, the domain of the modality etc. However, it was used as a steppingstone in the new methodology suggested to examine learning of novel sequences from time point 0, capture the learning process of completely novel items while controlling for exposure.

After summarising the current methodologies used in SL, we can conclude that most of the experimental paradigms claim to reflect aspects/components of the SL process while they are actually studying the learning outcome of the process. That is limiting the actual information that we can get about the actual mechanism and therefore emerges the need for a new experimental design that captures the time course of the SL mechanism. Additionally, methodologies such as 2AFC and familiarization tasks, are better fit for understanding explicit learning mechanisms rather than implicit. SL is an implicit learning mechanism, and therefore it needs to be studied as an unconscious learning process, without participants being affected by instructional, or task procedural bias.

After reviewing the problematic uses of each methodology above and their relation to the current new suggested method, the aim of the following sections in this chapter are to:

(a) Provide a detailed description of the new experimental paradigm and the step-by-step development of a fully functional experimental task.

(b) Present data and analyses from different development steps of the new method to demonstrate the efficiency of the methods to show how learning over time occurs and how it overcomes prior limitations in SL methodologies.

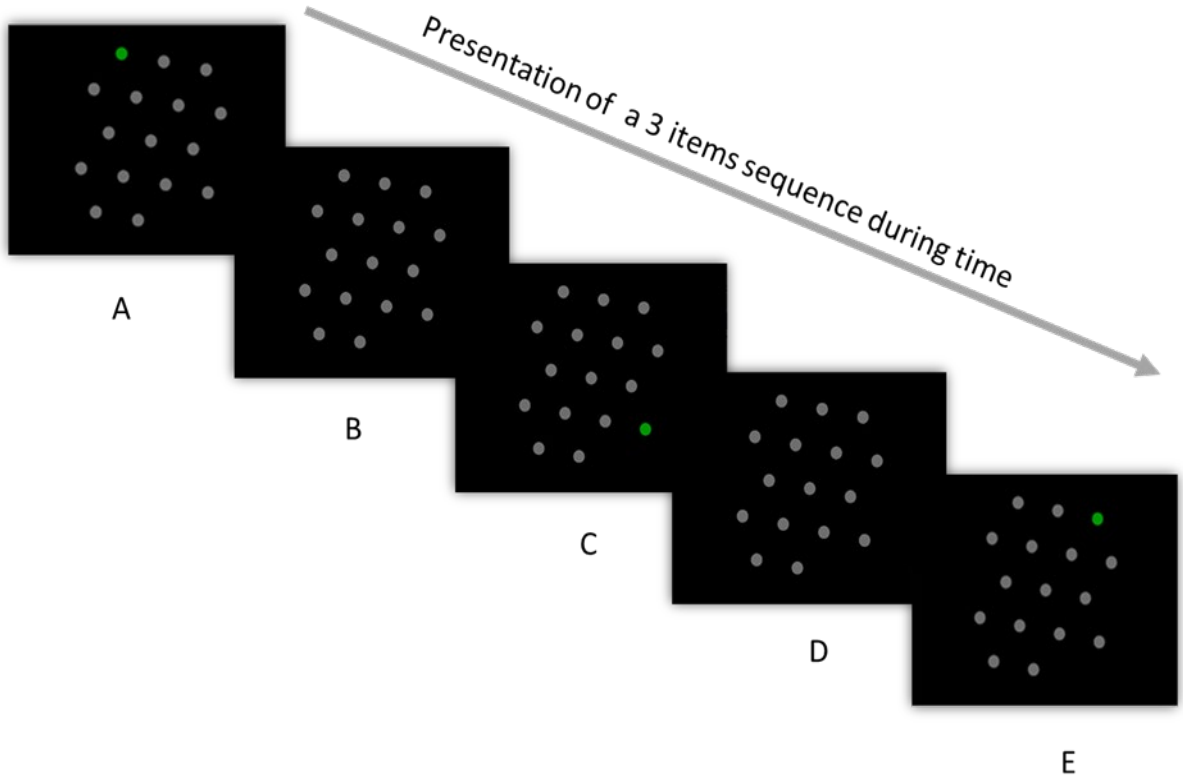
(c) Give a full explanation of how this current research methodology can be applied in other domains rather than the visual that have a temporal element, and what are the current design limitations.

B. Primary Experimental Design (Design A) & Piloting

The demonstrated design A was part of a larger study with more tasks, but for the purposes of introducing the new design, we will only present in this chapter the task that was created for SL of sequences of 3 dots.

Paradigm summary:

The suggested experimental paradigm involves the recording of eye-movements, during a sequential visual SL task. The sequential visual SL task involves learning sequences of dots that appear on locations in a semi-random array. Dots are highlighted in sequences and participants are asked to anticipate and look toward the location of the next dot in the sequence. Sequences are presented in a continual stream, with no indication of when a new sequence starts. Learning is determined by studying raw eye-samples during the task and there is no separate learning and testing phase. Figure 4.1 demonstrates the presentation of a 3 dots sequence within the task, with the details about eye-tracking recordings and time displays.



Phase A	**Beginning of 3 items sequence. 1st item of sequence is presented. Gaze contingent eye-tracking until gaze hits target. Time displayed for 275 ms after gaze hits target.**
Phase B, D	Time displayed for 750 ms of the blank period that guessing occurs.
Phase C	2nd item of sequence is presented. Gaze contingent eye-tracking until gaze hits target. Time displayed for 275 ms after gaze hits target.
Phase E	3rd item of sequence is presented. Gaze contingent eye-tracking until gaze hits target. Time displayed for 275 ms after gaze hits target.
Learning is quantified as the number of raw eye-samples on the target location during the 750ms guessing period.	

Figure 4.1 This figure represents the presentation of a 3 dots sequence in the new experimental design.

B.1. Ethics

This project has been approved by the Ethics Committee of Nottingham Trent University, (NTU Ethics Procedure Approval Code No 2018/218).

B.2. Participants

For this study 36 adults (30 female, 6 male) within the age range of 18-33 years old (M=21.06, SD=3.01), with not known learning disabilities or cognitive impairments were recruited. Figure 4.B.1 shows the participants age and gender distribution. All participants had normal or corrected vision. Participants were recruited via the SONA reward system from the participation pool of Nottingham Trent University.

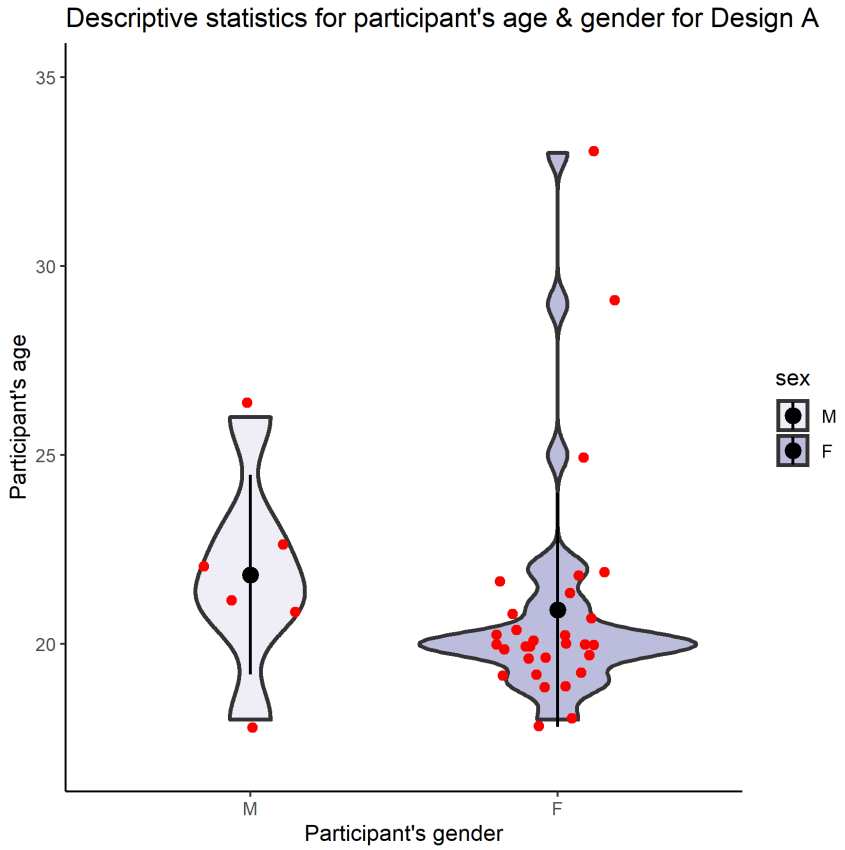


Figure 4.B.2.1 This violin plot represents the age and gender frequency distribution of participants for Design A. Each red dot represents a participant.

B.3. Materials

The visual stimuli used for this study was a green dot that moved on a 16 locations array of grey dots on a black background. The code that generates the location patterns of grey dots (Appendix D) and the code that generates the actual experimental task (Appendix B) were both

Python based scripts (Python Software Foundation. Python Language Reference, version 2.7. Available at http://www.python.org). In total, 3 different location sets (Table 4.B.3.1, Figure 4.B.3.2) were created and one of them was allocated to each participant. All the cartesian points within a location set, were designed to be equidistant to avoid internal distance bias during the eye-tracking. A visual representation of each location set can be found in Figure 4.A.2. The basic frameworks that were used to build up the operational scripts that connected the eye-tracker operating system with the computer operating system, were mainly ioHub (https://github.com/isolver/ioHub) from where we resourced the eye-tracker connection scripts in order to activate the eye-tracker during the task, but also perform the calibration process and PsychoPy2 (https://www.psychopy.org/) where we build the experimental sequence.

set	dot	x	y	set	dot	x	y	set	dot	x	y
LS1	1	-0.40528	-0.04712	LS2	1	-0.27688	-0.19392	LS3	1	0.31616	-0.14288
LS1	2	0.1156	-0.3372	LS2	2	-0.07264	-0.2892	LS3	2	0.29664	0.08168
LS1	3	0.26056	-0.16456	LS2	3	0.1316	-0.3844	LS3	3	0.27688	0.30616
LS1	4	0.40536	0.00808	LS2	4	-0.29656	0.03056	LS3	4	0.13168	-0.27208
LS1	5	0.18344	0.0472	LS2	5	-0.09232	-0.06464	LS3	5	0.11192	-0.0476
LS1	6	-0.10624	-0.29808	LS2	6	0.11192	-0.15992	LS3	6	0.09232	0.17688
LS1	7	0.03856	-0.12544	LS2	7	0.31616	-0.25512	LS3	7	0.07264	0.40136
LS1	8	-0.1156	0.29808	LS2	8	-0.31616	0.25512	LS3	8	-0.48112	0.0136
LS1	9	0.32824	0.21984	LS2	9	-0.11192	0.15984	LS3	9	-0.07264	-0.17688
LS1	10	-0.32824	-0.25888	LS2	10	0.09232	0.06464	LS3	10	-0.11192	0.27216
LS1	11	-0.18336	-0.08632	LS2	11	0.29656	-0.03064	LS3	11	-0.09232	0.0476
LS1	12	-0.03856	0.08632	LS2	12	-0.1316	0.38432	LS3	12	-0.25728	-0.30616
LS1	13	0.1064	0.25896	LS2	13	0.07264	0.28904	LS3	13	-0.27688	-0.0816
LS1	14	0.2512	0.4316	LS2	14	0.27688	0.19392	LS3	14	-0.29656	0.14288
LS1	15	-0.33744	0.3372	LS2	15	0.48112	0.09856	LS3	15	-0.31616	0.36736
LS1	16	-0.2604	0.12544	LS2	16	0.25728	0.41832	LS3	16	-0.46152	-0.21088

Table 4.B.3.1 Shows the x and y coordinates for the 3 different location sets that were created for design A. Each location set contained 16 different points, that were equidistant. Those cartesian points created the appropriate equidistant array for our visual stimuli to draw on the screen.

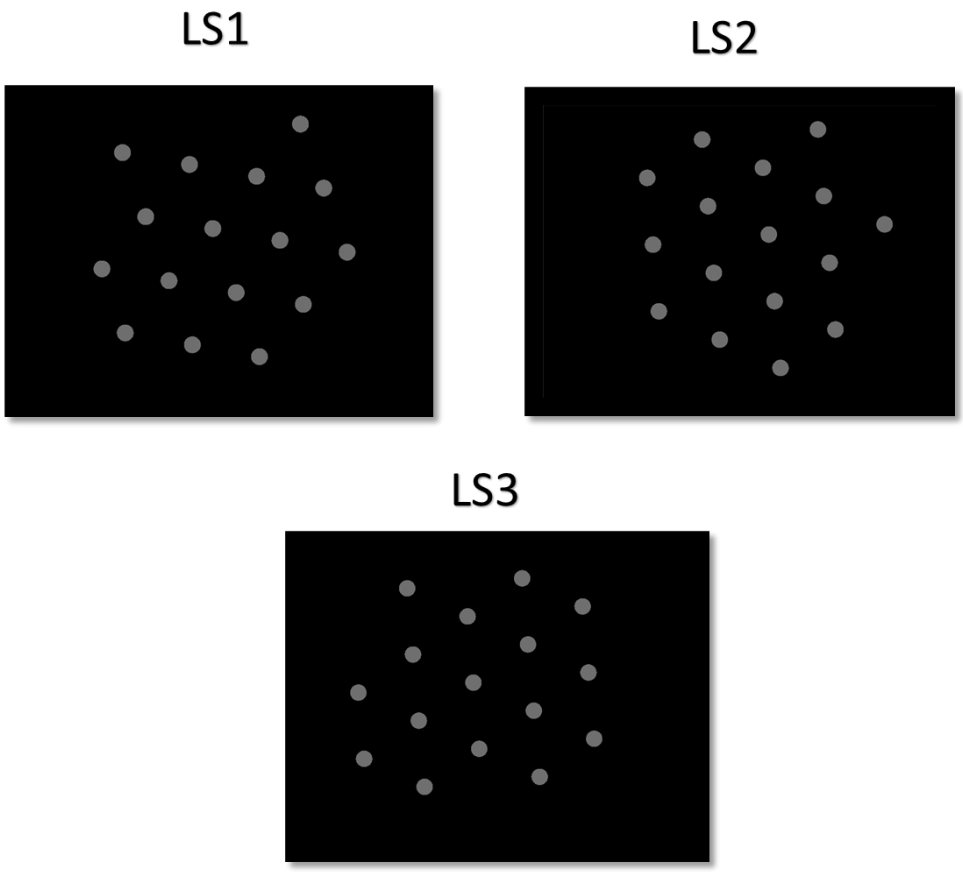


Figure 4.B.3.2 This figure is a visual representation of the 3 different location sets (LS1, LS2, LS3) that were used in Design A. Each location set contained 16 different points, that were equidistant.

B.4. Equipment

A Gazepoint GP3 (60 Hz) portable eye-tracker was used to record eye-movements during the experimental procedure. The eye-tracker has 0.5-1 points visual angle accuracy and can capture 25cm (horizontal), 11cm (vertical) movement with a ±15 cm range of depth movement and a sampling rate of 60 Hz resulting in a recording of 60 eye-samples per second. The task was presented on a 19" monitor with 1280 x 1024 pix resolution and its dimensions were 471 x 281 x 38 mm. The task was operating on a Windows x 64 Intel® core ASUS i7 laptop. A chinrest was placed on top of the desk, facing at the monitor at 70 cm distance.

B.5. Design

In the current design, the interest focuses on how learning (number of eye-samples on the target location) occurs in sequences of 3 dots. For this purpose, task 1 was created and contained 4 sequences (A, B, C, D) that consisted of 3 dots each one. Each trial began with a green dot been drawn on top of a grey dot - on an array of total 16 grey dot locations- until the participant looked at it and remained there for 275ms after which a blank period of the grey dot location array followed for 750ms (if learning of the location of the next dot occurs, then eye movements toward the dot would be expected during this blank period). A visual representation of a trial can be seen in Figure 4.B.5.1 The number of trials for the task was 480 (4 sequences * 40 repetitions each sequence * 3 items each sequence). The target location in each trial was defined as the next location in the sequence. If participants have successfully learned the sequences, after the visual presentation of the 1st item (dot) of the sequence, they were expected to look straight in the next location within the 750ms gap between one location being in green and the next location being in green. The process was gaze-contingent, meaning that the design secured that the participants had looked at the green dot before moving into the next trial (i.e., the next green dot being shown). Eye-samples in within the radius of 2.75 times the size of the dot, were counted as hits in the area of interest (AOI). A visual representation of the AOI can be found in Figure 4.B.5.2. The size of the AOI was decided so that there was no overlap between the AOIs of the other displayed dots and the target dot, but also ensuring that the AOI was big enough to capture eye-samples on the target dot. Specific timings about the trials are given in the procedure section. In Table 4.B.5.1. is presented the different dimensions of stimuli (dot/target), AOI and display, with the relevant centred visual degrees calculation for horizontal and vertical axis. The visual degrees were calculated with the use of the online SR Research calculator, (https://www.sr-research.com/visual-angle-calculator/). In Table 4.B.5.2, we can find the scores for the mean(M) and the standard deviation (SD) for the eye-samples that hit

the AOI during a trial in the 3 dots task, across all 36 participants. The total number of trials was 17244.

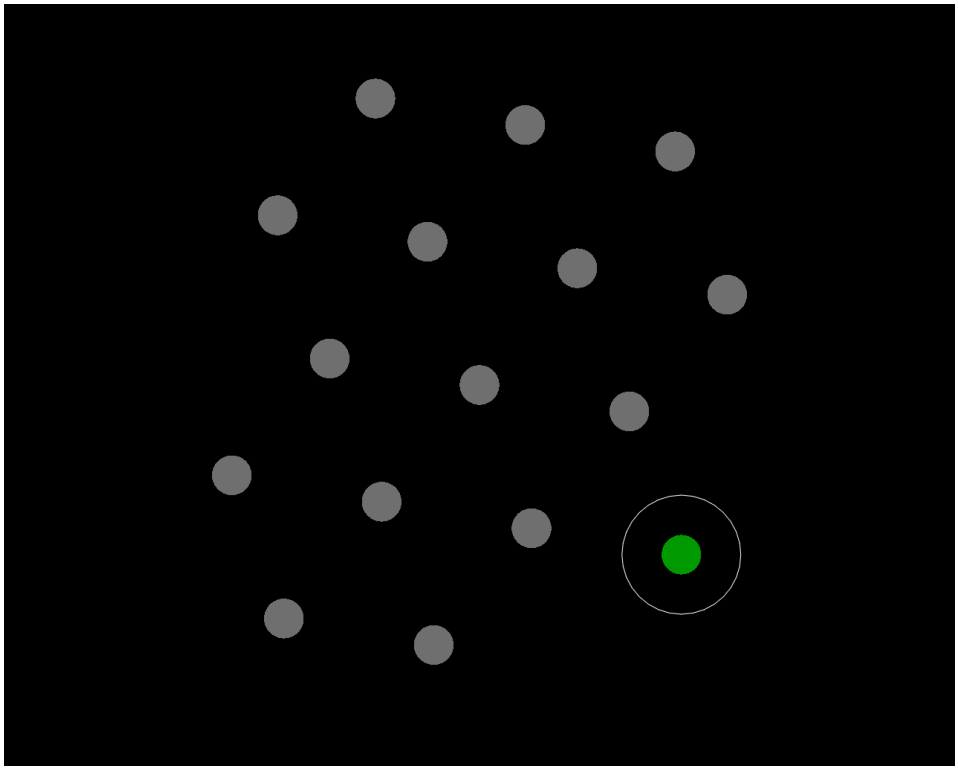


Figure 4.B.5.2. This figure shows that AOI around the green target dot (in a 1280x 1024 px resolution).

	Screen	**AOI**	**Target**
Dimensions	1280 x 1024 pix	Radius AOI =45.84 pix (2.75 times the radius of target/dot)	Diameter =33.33 pix Radius = 16.67 pix
Size in Pixels	1310720 pix (width * hight of monitor)	6601 pix (3.14 * Radius AOI)	873 pix (3.14* Radius Targer/ Dot)
Horizontal Visual Degrees	1 Horizontal Visual degree = 33.29 pix		
Vertical Visual Degrees	1 Vertical Visual degree = 44.64 pix		
Distance from Screen 700 mm (70 cm)			
Dimensions of monitor 471x 281 mm			

Table 4.B.5.1 Table of centred visual degrees, size in pixels, and dimensions for target, AOI and display.

Trial = Blank period of guessing(0-750ms) + Target appears on screen, participants look at it, it remains there for 275ms after participants have looked at it. (750ms-…)		
	M	**SD**
Blank period	2.44	6.79
Target appears on Screen	23.9	22.8

Table 4.B.5.2 This table shows the mean and the standard deviation of raw-eye-samples that hit the AOI during a trial. Scores were calculated based on a total sample of 17244 trials (36 participants-479 trials each participant).

Counterbalancing & randomization of trials.

All trials for the task were counterbalanced for location set per participant, while the sequence order and the locations allocation were randomized for each participant. Each sequence wasn't allowed to appear twice in a row. Each location of the array was allocated on a single sequence item, to avoid overlapping of locations across sequences. All trial files were generated using R.3.2.5 version (R Core Team, 2013).

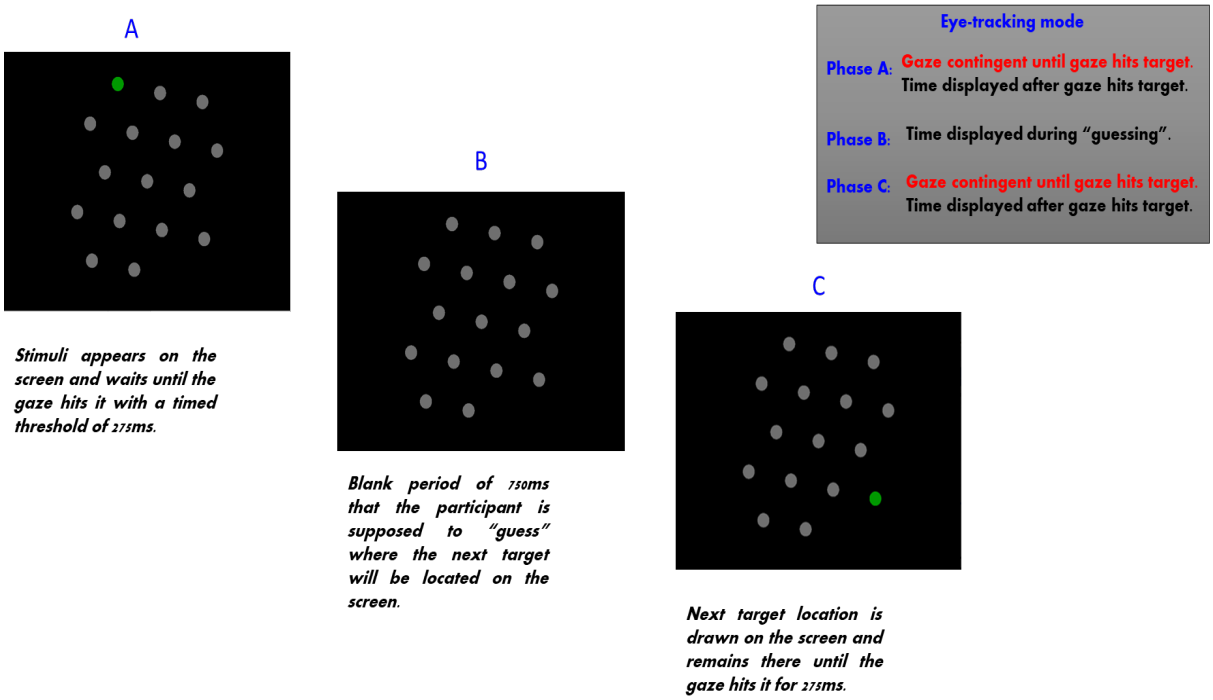


Figure 4.B.5.1 This figure represents an experimental trial in Design A.

B.6. Procedure

The experimental session procedure consisted of 5 different stages:

1. Participant's information, Consent form & Demographics: Since the participant entered in the eye tracking lab, he was provided with the Information Sheet about the study (Appendix A, Section A.1) and was given approximately 5 minutes to read through it. Then the participant was asked if he had any questions about the study (if so, was given further study details) and moved on signing the Consent Form (Appendix A, Section A.2). After the participant signed the Consent Form, he was allocated with an experimental ID number. The researcher opened the appropriate framework on the laptop and entered the experimental ID number, age and gender of the participant in the platform.

2. **Verbal instructions about the procedure & positioning for eye-tracking:** In this stage the participant was asked to turn off his mobile phone or any other electronic devices he might had with him and sit comfortably in front of the experimental monitor, so that he could be given

the verbal instructions about the task. The verbal instructions for the task were the following: "*In this task you will see a green dot moving in different locations. All you have to do is guess where the dot will appear next by looking at the location that you think the green dot will appear next*." Then the participant was asked if he had any questions about the instructions. If so, the researcher answered the question and moved on to the next step of positioning for the eye tracking. At this time point the participant was informed about the importance of keeping his head stable on the chinrest until the end of the task. Then the researcher explained the procedure of eye-tracking as follows: *"Before you begin the experiment, you will have to do a small task that last about a minute. All you must do here is look at the centre of a dot that will be drawn in different locations on the screen. At this task you are not asked to guess, but just look at the centre of the dot. After this task is finished, you will move to the experimental task, where you will start the guessing".* Then the researcher positioned the head of the participant on the chinrest and pressed enter on the open framework on the laptop. Then a window appeared asking if the participant was ready to be calibrated. The researcher pressed enter the calibration process began.

3. Calibration: As part of the procedure of conducting an eye-tracking experiment, participants did a 5-points calibration task, with a 20 pix calibration error. If the calibration was successful, the participant moved on the experimental task. Otherwise, the calibration process was repeated, until it was successful. A new calibration process was occurring before the beginning of each experimental task. Usually, the calibration process lasted less than 1 min. At the end of a successful calibration a window was drawn saying that the calibration was successful and giving the calibration error threshold. Then the researcher pressed enter and the experimental task began.

4. Experimental Task: In the task, each trial began with a green dot been drawn on top of a grey dot - on a array of total 16 grey dot locations- until the participant looked at it and remained

there for 275ms, after which a blank period of the grey dot location array followed for 750ms. The task was gaze contingent, so participants had to look at each green dot to progress further in the task. The green dot was considered as seen if there was a hit within a radius of 2.75 times the size of the dot. The 750ms "blank" period is the crucial experimental period that will show if SL occurs. At the end of the task a black window was drawn on the screen that thanked the participant for his participation in the experiment. The window remained there until the participant pressed "space" on the keyboard.

5. Debrief about the research aims of the study: At this stage the participant was verbally thanked for his participation by the researcher and given a Debrief Sheet (Appendix A, Section A.3) about the aims of the study. Then he was asked if he had any other questions relevant to the experiment and was given the SONA credits as a reward for his participation.

The total task procedure lasted about 15 mins.

B.7. Piloting prior to Primary Experimental Design- A justification of choices in experimental Design A.

1. Choices of Equipment (GP3 eye-tracker)

The GP3 (Gazepoint) eye-tracker was new experimental equipment that arrived at the department approximately at the same time as I started my PhD journey. Previous research that has been conducted with GP3 (Mannaru et al., 2017; Brand, Diamond, Thomas & Gilbert-Diamond, 2021), has shown that even though it is a low frequency tracker (60 Hz) compared to others (e.g., Tobii, Eye-Link SR), it can provide reliable eye-tracking data. The GP3 trackers also have the advantage of low purchase cost and portable functionality, that could allow data collection outside of an experimental lab (e.g., schools, nurseries, etc.) if needed. The GP3 was therefore chosen as the equipment for design A.

2. Choice of Blank Period timing & Piloting

Given the fact that Design A is an eye-tracking design, it was crucial to take into consideration the timing of basic eye-tracking movements such as saccades and fixations. A typical saccade can last from 20 ms to 40 ms and a fixation is on average 250ms. Considering that the responses in Design A were recorded by eye-movements (hits on target location) during the blank period, we had to ensure that the participants had enough time and space to perform at least one fixation on target, meaning that the minimum blank period tested was 500ms to allow saccades, around the different targets (distance of targets across the window) and at least one fixation. This would mean that with a 60Hz sampling rate we could have a maximum of 30 eye-samples (hits) on target for a blank period of 500ms. However, as it is shown in Figure 4.B.7.1, the piloting data for 3 participants (001, 002, 003) demonstrate that no learning was achieved during the 500ms period. Therefore, we increased the blank period time to 600 ms, that would allow a number of 36 eye-samples (hits) on target. Figure 4.B.7.2, shows the piloting data for 600ms for 3 participants (001, 002, 003). No learning is evident across the three participants for the blank period of 600ms. Then it was decided to set up the blank period window to 700ms, to allow more time to elicit any learning taking place. During the 700ms blank window period a maximum of 42 eye-samples (hits) on target could be achieved. In Figure 4.B.7.3, it is shown that some learning was achieved by some participants (001, 002, 004), while some participants (003, 005) did not learn. In order to avoid having too long trials, as it would make the task too long and exhausting, and not lose the temporal element of the task (too long intervals between stimuli (dots)), we decided to choose a blank period of 750 ms. This way we could increase the number of stimuli presented (task with sequences of 4 or 5 dots) and still have a task that would last 15-20 minutes. It is crucial to highlight that because of the continuous nature of the task it was impossible to introduce breaks during a task, as it would disrupt the learning.

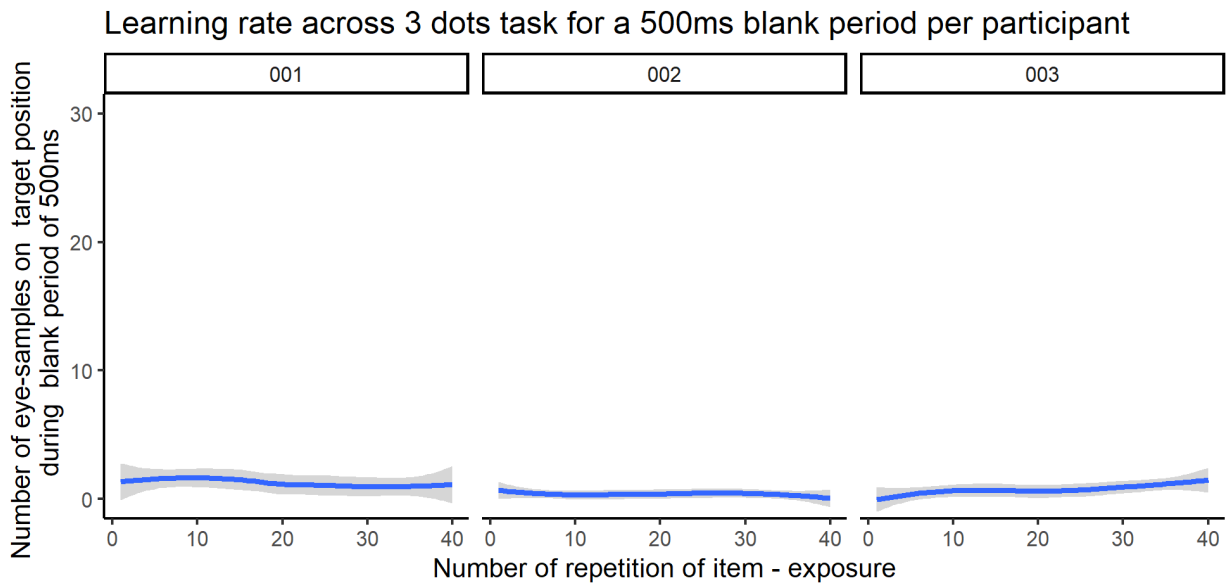


Figure 4.B.7.1. Piloting: Learning rate for 3 dots task during a blank period of 500ms per participant (001-003).

Error bars represent 95% CIs.

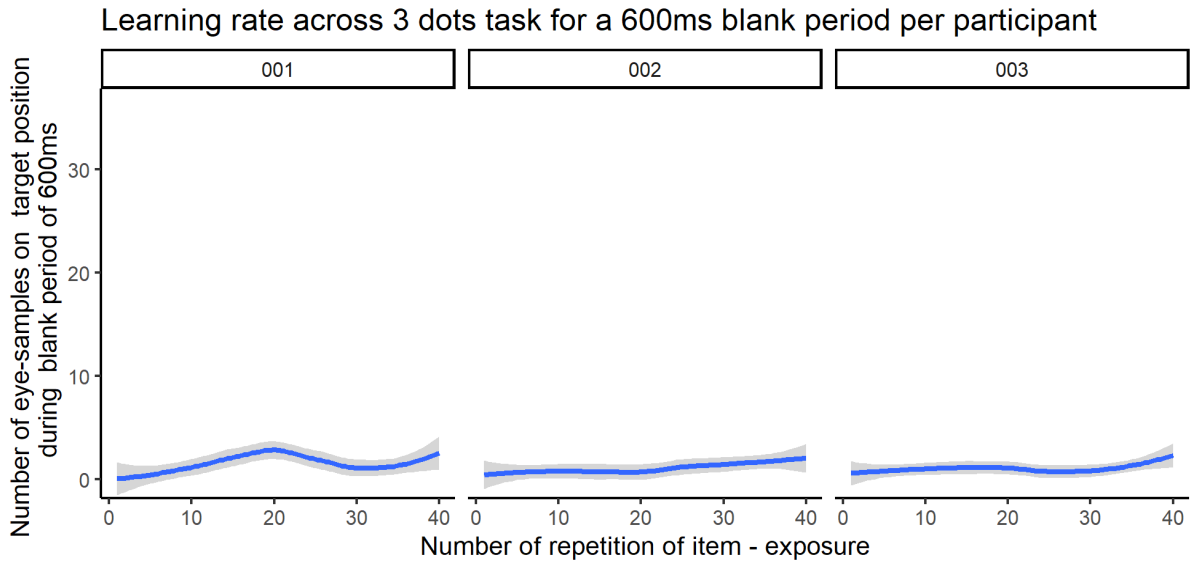


Figure 4.B.7.2. Piloting: Learning rate for 3 dots task during a blank period of 600ms per participant (001-003).

Error bars represent 95% CIs.

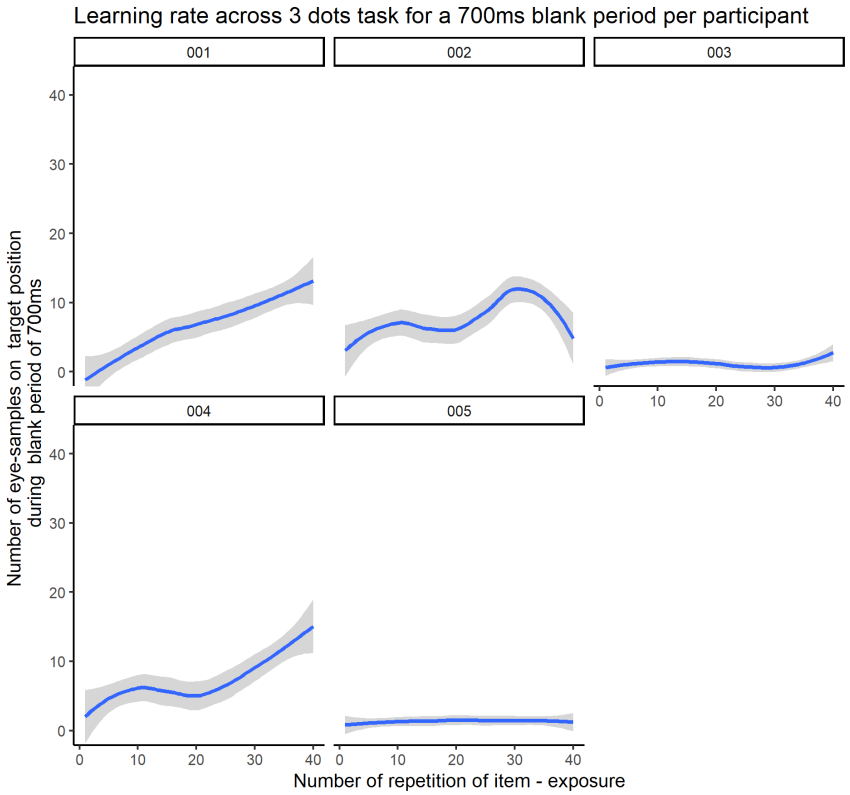


Figure 4.B.7.3. Piloting: Learning rate for 3 dots task during a blank period of 700ms per participant (001-005). Error bars represent 95% CIs.

3. Choice of sequence repetitions (occurrence).

As it has been stated previously, the total duration of each task was a serious concern of this design, since no breaks could be introduced midway through the task, as they would disrupt the learning process. Therefore, we had to ensure that all the experimental manipulations were the best option possible in terms of time efficiency and demonstration of some actual learning capacity. We decided that 40 repetitions of each sequence in a task, was enough exposure to each sequence for participants to be able to demonstrate learning. There was no piloting with more or less repetitions, but it is something that could be considered in future designs. Previous research in statistical learning (Saffran et al., 1996) has shown that an amount of 45 repetitions per word (4 three-syllabic words, on a speech stream of 270 syllables per minute, on a speech stream of total duration of 2 minutes), is sufficient to show learning in 8-months old infants.

Since we are testing adults, we considered that 40 repetitions would be a sufficient amount of exposure to observe learning, as piloting data can confirm (see previous Figure 4.B.7.3).

4. Choice of raw eye-samples over fixations.

For this experimental paradigm, we have decided to look at raw eye samples rather than fixations, to demonstrate learning. This approach allowed us to analyse patterns of eye-movements holistically throughout a task and record the process as a live learning procedure, with continuity from the 1st trial until the end, rather than observing learning on a trial-by-trial basis. Focusing only on fixations on target locations in each trial could provide a robust measure for learning on a trial-by-trial basis, but it would disrupt the continuity of the task and it would limit the visualisation of the learning curves (how many fixations can fit in 750 ms? - the more data we have for learning (fixations vs. raw eye-samples) the more precise/naturalistic the magnitude of the curve). To visually establish this fact, Figure 4.B.7.4 and Figure 4.B.7.5 were created. Figure 4.B.7.4 shows the learning of participant 005 based on raw eye-samples, and Figure 4.B.7.5 shows the learning of participant 005 based on fixations. As you can see from the graphs, the pattern of learning in the fixation graph mimics the pattern of learning in the raw eye-samples graphs (shape of curve), however it minimises the numeric variance that learning can get in this design, due to low learning rates and variance in learning across individuals.

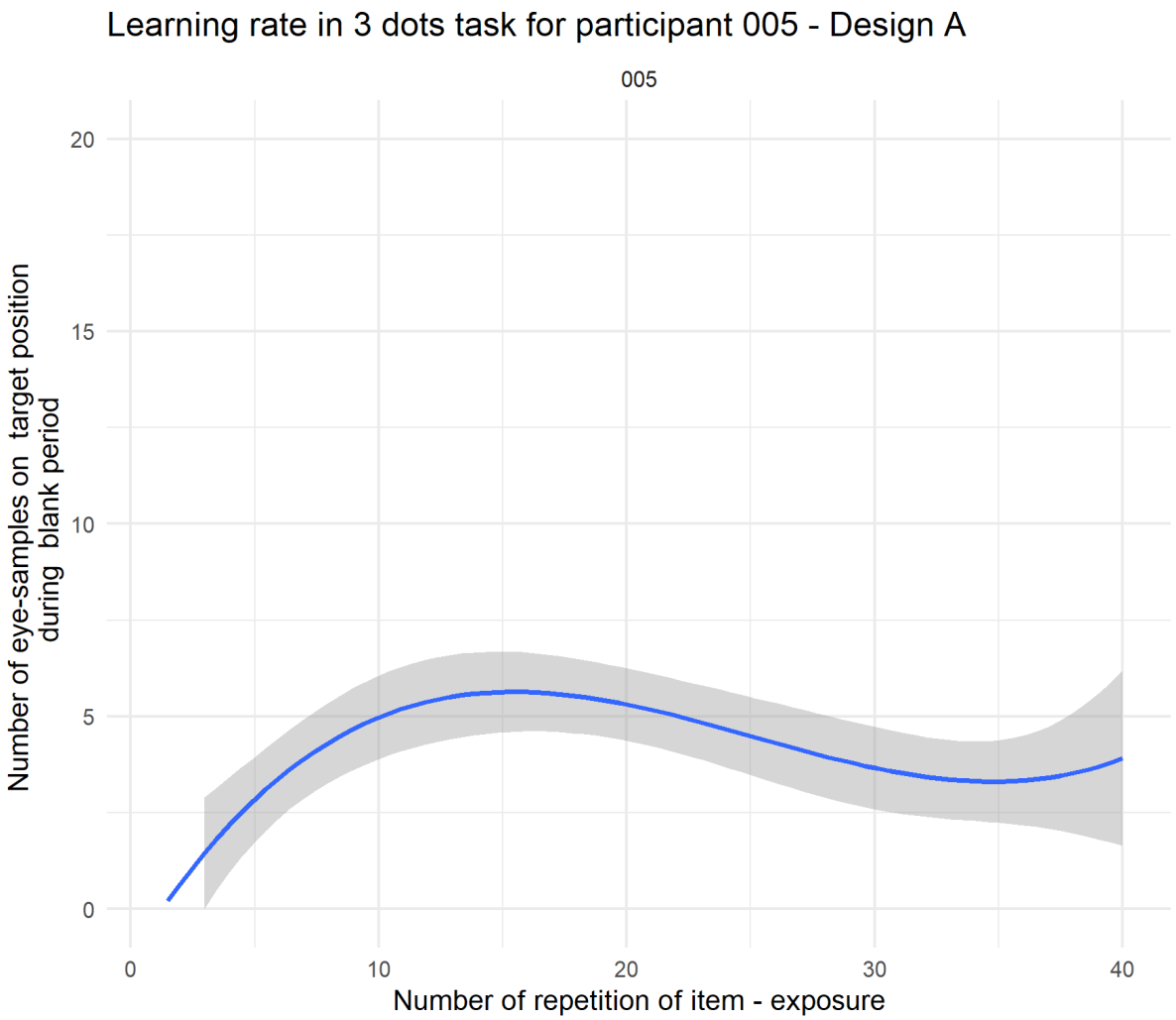


Figure 4.B.7.4. This plot shows the learning rate across sequences in the 3 dots task, for participant 005, for Design A. On x-axis is the number of occurrence / exposure and on the y- axis is the number of eye-samples on the target location during the 750ms blank period. Note – there is a maximum of 45 eye-samples per trial. Error bars represent 95% CIs.

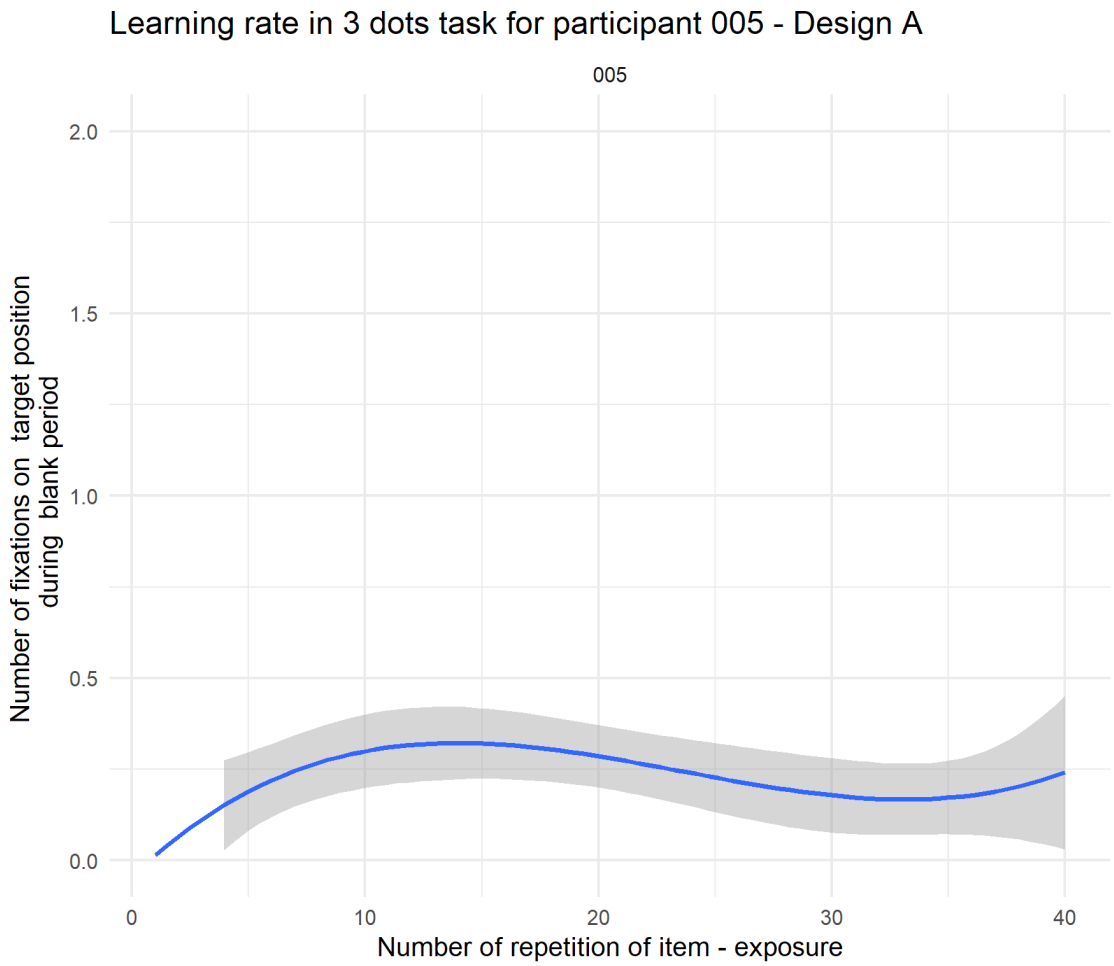


Figure 4.B.7.5. This plot shows the learning rate across sequences in the 3 dots task, for participant 005, for Design A. On x-axis is the number of occurrence / exposure and on the y- axis is the number of fixations on the target location during the 750ms blank period. Error bars represent 95% CIs.

Raw eye samples have the advantage of higher temporal resolution over fixations. This suggests that raw eye samples provide a more detailed representation of eye movements, capturing the precise position of the eye at each sample point. The high temporal resolution of raw eye-samples allows the recording of learning in a more naturalistic, continuous manner in comparison to fixations which represent relatively longer periods of stable gaze. On the other hand, raw eye-samples increased the complexity of the data analysis due to the volume of the data and the complexity of the computations (calculating eye-samples over a certain period of time per trial on target location). Furthermore, raw eye-sample are more open to interpretability

in comparison to fixations. Fixations offer a clear demonstration of where attention is focused, while raw eye-samples require additional processing in order to determine if the eye-movement patterns represent learning. This subjectivity could lead to false interpretations of the data and as a result we would label data that are random hits on the target as "learning". That said, random hits on the target – which would be potentially classed as learning – would account for very few samples of 'eye on target' and would therefore contribute little to the overall learning trajectories. That said, to mitigate any effect of random samples on target, we visualised the data that are presented in this chapter (design A) and in the next chapters (data that derive from design B) and ensured that the raw eye-samples that were representing learning had (a) a consistency of movement pattern during a trial and (b) a continuity in time during the critical blank period time. To examine those components, we created plots for each task, for each participant, for each sequence, that were demonstrating the eye-movements during each trial (from the moment that x target appeared to screen, during the blank-period, and up until the moment the next target appeared on the screen). These graphs were capturing the distance of the gaze from the AOI for the current trial and allowed us to observe the pattern of eye-movements of individuals in detail and assess their reliability. A reliable learning pattern during a trial that (for example) a pair of targets is presented (e.g., 1-2) and which examines the learning of target 2 would demonstrate the following patterns:

a. gaze away from AOI of target 2, as it should be looking at the target 1 for the first 275 ms.
b. gaze moving towards the AOI of target 2, during blank period of 750ms (guessing period) and remain there for at least a continuous time of 100 ms (thus ensuring that raw samples on AOI are not samples from saccades across the screen).

c. gaze moving towards the AOI of target 2, after the end of 750ms, since target 2 has appeared on screen.

An example of this graph for participant 005 can be found in the next section, in Figure 4.C.4.

C. Results and Critical Evaluation of Design A

This section consists of the evaluation of Design A, with some basic visualisation of the learning occurring during the task.

C.1. Outcomes of Design A

In Design A, a visualisation of the outcome was used to examine whether learning is achievable in a 3 items sequence. In order to do so, the data were cleaned data and only the raw eye-data that reflected learning were included in the data visualisation (scripts available in Appendix G). For this design demonstration, trials that consisted of eye samples to the first location of a new sequence (transition from the final position of the current sequence to the first position of the next sequence) were not removed from the data. However, it must be acknowledged that these trials are exogenous to the sequence learning process and in some way random (1/3 chances to guess correctly if learning within all sequences is successful due to only being 4 sequences with no sequence repeated immediately after itself and 1/16 chances to guess correctly if learning within all sequences is unsuccessful). This issue will be addressed in the next Chapter (Chapter 5), where a comparison between those exogenous to the sequence trial and the endogenous to the sequence trial will be compared. But for the purpose of this design evaluation, they were included in the figures and descriptives statistics. Learning was calculated as the averaged eye samples on the target location across the endogenous sequence positions for each task within the 750ms of the blank period. For the 3 dots task the eye samples in target locations of the second and third items of the sequences were categorized as learning

data. Finally, a polynomial regression (method lm, y ~ poly(x,3)) was fitted to the data to extract a first impression about the form of the data.

Figure 4.C.1 shows the percentage of learning in the 3 dots task from their 1st exposure until the 40th exposure aggregated across all 4 sequences. As it can be seen in Figure 4.C.1 the percentage score of learning is low, potentially indicating that very little learning took place (the number of eye samples is 45 because the 750 ms blank period equates to approximately 45 eye samples). This led to the creation of closeup visualisation of the same data for each specific sequence (Figure 4.C.2) and a look into individual plots to examine whether the low numeric values derive from individual variation on performance. As it can be seen in Figure 4.C.1 only an approximate of ~7,5% of eye samples manage to hit the target location, suggesting that the averaged results across participants showed that there are approximately 3 to 4 out of the 45 eye-samples (Figure 4.C.2) on the target location by the end of task. These low numeric values could have many explanations; therefore, a further visualisation of individual differences was needed to evaluate the method (Figure 4.C.3). As we can see in Figure 4.C.3 There are many participants that fail to learn (e.g., 019,007), but there are also some participants that manage to show some learning (e.g., 005,024).

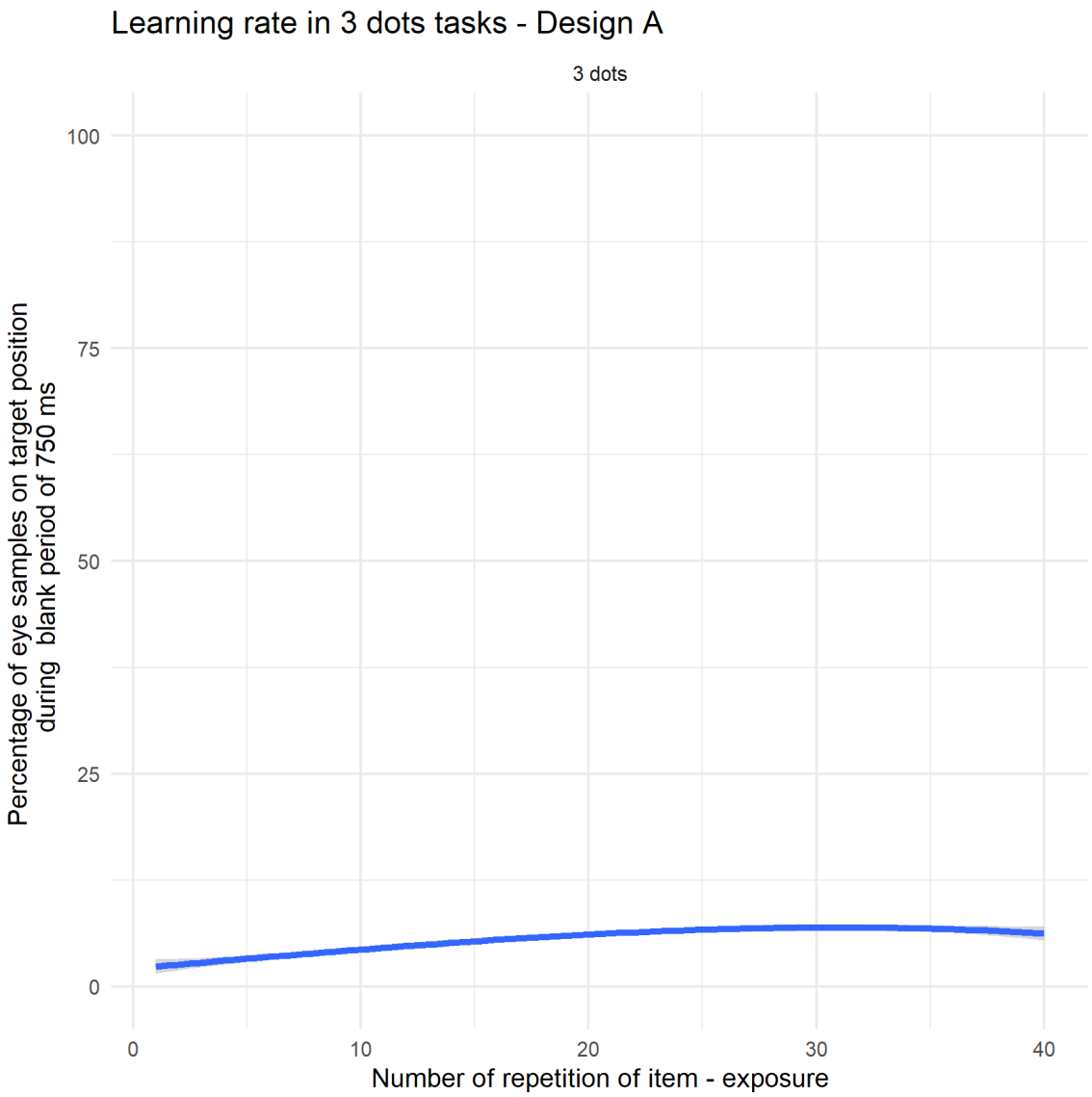


Figure 4.C.1 This plot visualises the learning rate across sequences in the 3 dots task for Design A. On x-axis is the number of occurrence / exposure and on the y- axis is the percentage (%) of eye-samples on the target location during the 750ms blank period. Note - there is a maximum of 45 eye-samples per trial. Error bars represent 95% CIs.

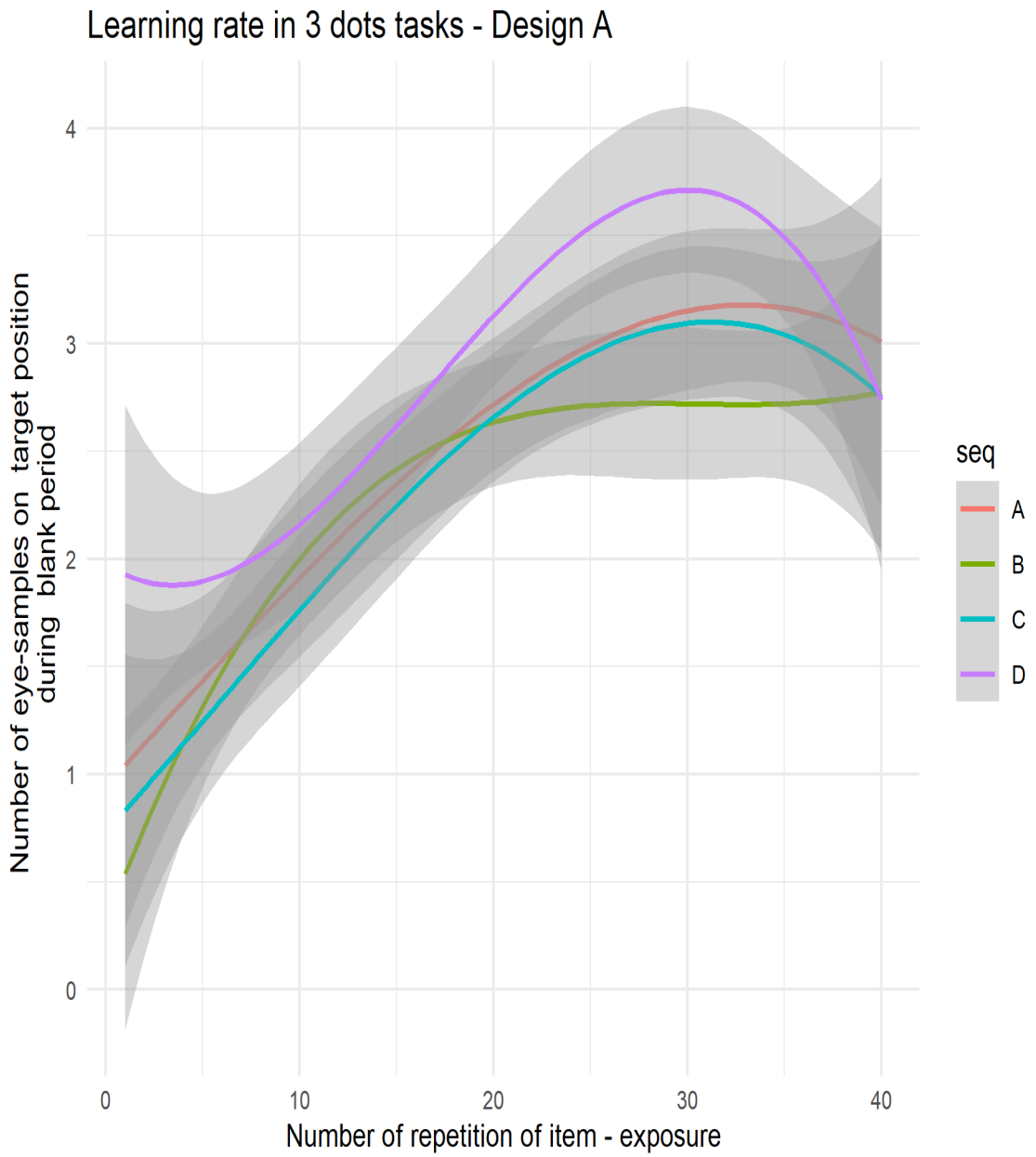


Figure 4.C.2 This plot provides a closer look into the effects across sequences in the 3 dots task for Design A, for each of the four 3 dot sequences (A, B, C, D). On x-axis is the number of occurrence / exposure and on the y-axis is the number of eye-samples on the target location during the 750ms blank period. Note – there is a maximum of 45 eye-samples per trial. Error bars represent 95% CIs.

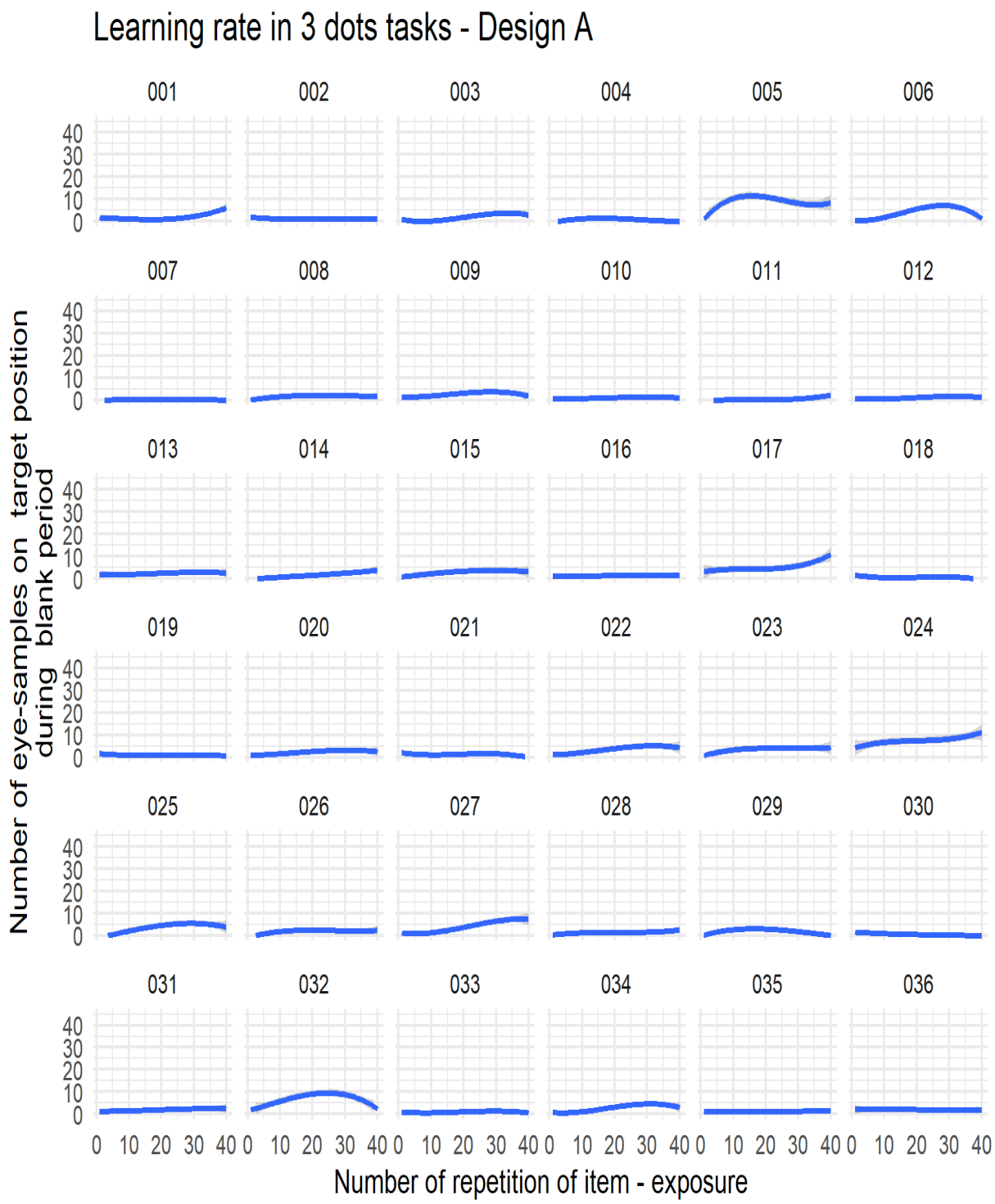


Figure 4.C.3 This plot shows the learning across sequences in the 3 dots task for Design A, for each individual subject (001-036). The x-axis is the number of occurrence / exposure and on the y- axis is the number of eye-samples on the target location during the 750ms blank period. Note there is a maximum of 45 eye-samples per trial. Horizontal lines at a y-axis of 0 indicate no learning (e.g., participant 019) with deviation from this indicating learning (e.g., participant 005). Error bars represent 95% CIs.

C.2. Critical Evaluation of Design A

Design A seemed to efficiently record the process of the task, however indicated low levels of learning that could question the validity of the suggested method. This raised concerns about the need of creating a more powerful design, Design B, that will be discussed in the next chapter (Chapter 5). But before moving into a new design, it is crucial to highlight that design A achieved one of the primary design goals. It managed to successfully capture the learning process of sequential SL from the beginning of the task until the end, without disruptions, and by simply capturing the implicit character of the SL procedure, without explicit instructions. The success of Design A can be seen in Figure 4.C.4, which is an example of a participant's performance from trial number 1 to trial 480 and shows the learning across time with the location of eye samples moving towards the next dot during the blank period, indicated by a shift in the red line towards 0 prior to the target being shown, (i.e., greater looking towards the target over time during the anticipatory blank period, indicating learning).

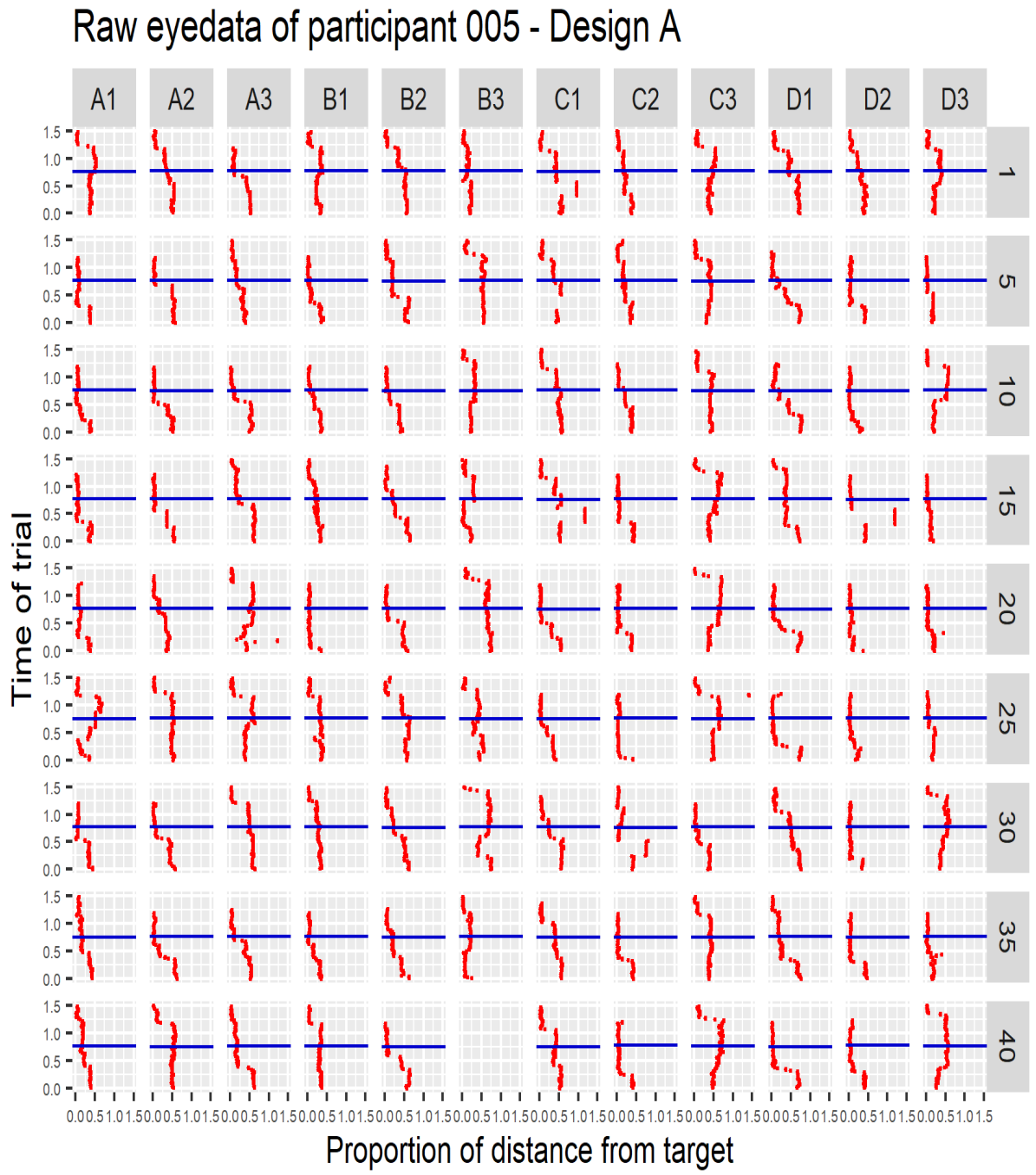


Figure 4.C.4 Plot with raw eye samples during a trial for participant 005. This plot demonstrates a 4 sequences – 3 dots length task. On x-axis is the distance of the eye from the target location (as a proportion), so red dots at 0 indicate the eye is on the target with increasing deviations from 0 meaning the eye is further and further away from the target. On y-axis is the time period of a trial, with critical period (the blank period where the target location is not yet indicated) being 0- 0.75s. A1, A2, A3 etc represent the transitions from a dot to another one within a sequence (so A,B,C,D stand for the 4 different sequences) and 1,5,10… 40 is the number of exposure to that item of the sequence. The blue line is representing when the stimulus is drawn at the next location on the screen – this is drawn at .75. The figures show both learning and lack of learning. For example, the location of the second dot of sequence A (column A2) has been learned at the 10th exposure (row 10) as indicated by the distance of the eye to the target location reducing over time (red plot moving leftward toward 0 on the x-axis prior to crossing the blue line) until the target is displayed (indicated by the blue line). This is not the case on the 5th presentation (row directly above, labelled '5') where the eye only shifts to the dot once the dot is displayed.

The GP3 (GazePoint) eyetracker has a maximum frequency of 60 Hz, suggesting that a maximum of 45 eye samples could be recorded during the 750 ms "blank period" of guessing. That low sampling rate might be responsible for the low numerical values in terms of number of eye samples, however there was a clear formulation of a learning curve. Therefore, the big question was created: "Are we supposed to perceive as learning a curve that goes from 0 to a maximum of 3 or 4 eye samples?" Those 3 or 4 eye samples could be on the right place at the right time due to a random saccade movement. Using a more powerful eye-tracker would provide more samples (eye locations) per second and alleviate this problem. Therefore, to improve the design a more powerful eye-tracker would increase the numerical power of the design, so that strong conclusions could be drawn for the effects observed in the data and the primary research aims. Additionally, that was a good reminder of always critically evaluating the efficiency of each equipment in a specific experimental set up. Sometimes the tools that are being used aren't powerful enough to capture the strength of the mechanism that needs to be explored. But how do we deal or interpret the findings of a less powerful design? What are the criteria that allow to assess the efficacy of a tool, and how low numerical values can be indicators of actual effect patterns? This topic will be fully discussed in Chapter 8, where a technical comparison of the 2 eye-tracking systems will be given.

Apart from the shift into a more powerful eye tracker, design A needs a methodological boost up. Although some individuals show learning (see Figure 4.C.3, 005, 032, 024, 027), there were other that showed none. That could be explained by having floor effects, due to the difficulty of the task, or lack of concentration of the participant, or simply individual differences in the learning process that are quite common in the field. If the task was really hard, with the usage of such low frequency eye-tracking (60Hz) we wouldn't be able to capture

any learning at all. This way, the possible explanations for floor effects are reduced to (a) low frequency sampling equipment, (b) individual differences in learning and (c) lack of attention and concentration in the task. Therefore, by using higher frequency equipment and maintaining the concentration of the participant in the task by introducing the beep sound as a negative feedback form, we aimed to improve the design and provide a clearer answer as for why we observe these low numeric values and if the data from Design A are actually meaningful in a context of interpretation.

To conclude, the coding software was easily adjustable and could be used with many eye-trackers while running the exact same processing routine. The experimental code was very flexible and allowed adding experimental elements such as negative feedback in the process with ease. All these factors led to the creation of Design B that will be presented in Chapter 5.

Statistical Sequential Learning: Final Experimental Design (Design B) and the debate around within sequence transitions and exogenous sequence transitions.

Chapter Summary

In this chapter, we will examine the final experimental design that has generated the experimental data that we will analyse in the following chapters to answer our research aims. Furthermore, an analysis of on the basis of within-sequence vs. outside-sequence transitions will be given to support the rationale of exclusion of outside-sequence (exogenous) transitions from the future analyses. More specifically, this chapter will be divided in two sections: section A, that will detail Design B and will provide a justification for the adaptations in it, in comparison to Design A (seen previous Chapter 4) and section B that will provide a theoretical and empirical rational for excluding between-sequence transitions from the future analysis, as they represent a form of learning that is different from the learning that occurs in within-sequence transitions.

A. Final Experimental Design (Design B): rationale, moderations and outcomes.

A.1. Rationale of Design B

In this chapter we will examine the alternations that we performed to Design A (previous chapter, Chapter 4), in order to maximise the learning outcome of the task. Design B is part of larger studies with more tasks. However, in order to compare Design B (final design) with Design A (primary design), in this section we will present only the task that was created for SL of sequences of 3 dots.

As we saw from the previous chapter (Chapter 4), Design A was created to examine sequential SL after a series of pilot studies. In that set up, SL of sequences of 3 dots was examined, under the usage of a GP3 (60Hz) eye-tracker. However, from the evaluation of design A, we made three conclusions: (a) SL was recorded within the task, (b) numeric values of learning were low, and (c) there were many participants that didn't demonstrate learning at all. Therefore, the need for an improved design arose. The design that will be presented in this chapter, Design B, is the improved version of Design A. In this set up, SL of sequences of 3 dots was examined under the usage of an Eye-link 1000 (1000Hz) eye-tracker which is a more powerful eye tracker (that can result in higher numeric values on the SL outcome - raw eye samples on target location) and additionally a negative reward manipulation was introduced in the task in order to motivate participants to do better in the task and increase the numeric values of the SL outcome. The negative reward system in Design B worked by introducing a beep sound as part of the procedure that worked as feedback. If the participant failed to orient their eyes to the next location within the 750 ms blank period (i.e., they had not learned the location of the next

dot) then a beep would sound. That would act as motivation to do well on the task (i.e., learn) to avoid hearing the beep sound. Every time that participant was correctly guessing the next location the sound wasn't heard. That would hopefully result in a task that would end up mostly silent after the 20^{th}-40^{th} repetition of the sequence for most of the participants that would concentrate on the task, while it would be consistently playing sound from the beginning until the end for the participants that were not learning during the task. It is expected that the sound feedback will have a negative reward effect (playing a sound when participants did not shift their eyes to the next location within the blank period) that will increase participant's motivation.

The "reward system" method is a method that has been used widely in the literature. Freedberg, Schacherer and Hazeltine (2016) used this different experimental paradigm to show how the associations between two stimuli (stimuli-response S-R) can be strengthened. In their study, they adapted the Seibel's (1963) chord task. During the task participants were presented with multiple stimuli simultaneously and had to respond correctly while they were rewarded for choosing correctly for some chords, but not for all. What is currently known about introducing a reward in a task in a specific domain, such as a motor task, is that it improves learning on that specific domain, and so for example responses relying on rewarded motor programs are produced faster (e.g., Abe, 2011; Wächter et al., 2009). One possible explanation is that participants become reward biased and therefore produce them faster (e.g., Rescorla,1968, 1988), however Freedberg et al. (2016) suggested that the reward strengthens the associations of S-R and therefore higher speed is expected due to stronger associations rather than the reward itself. This event is plausible if the S-R is on a one-to-one level or on a combination of stimulus and response match level (Freedberg et al., 2016). Similarly, the speed of performance can become faster when anticipating a reward (e.g., Haith, Reppert, & Shadmehr, 2012; Opris, Lebedev, & Nelson, 2011) but also rewarded stimuli are attended longer and perceived faster

than other stimuli that were associated with smaller reward (e.g., Anderson, Laurent, & Yantis, 2011; Roper, Vecera & Vaidya, 2014). However, reward systems can lack ecological validity since there is not always a reward system in real life situations, so the process doesn't imitate the actual real-life learning process and can lead to implications related to the implicit (non-biased) character of the task and the applicability of the method in a variety of tasks across domains. In the new final methodology suggested, a reverse or negative reward system was used during the eye-tracking task, that worked as negative feedback (bip-sound) every time the participants failed to learn an item of a sequence during the task (failed to move their eyes to the next dot location before the dot was illuminated). This way the task managed to keep the implicit character of the task, while it motivated the participants to attend to the task and want to learn to avoid the bip-sound.

A.2. Design B

A.2.1. Ethics

This project has been approved by the Ethics Committee of Nottingham Trent University, (NTU Ethics Procedure Approval Code No 2018/218).

A.2.2. Participants

For this study 36 adults (30 female, 6 male) within the age range of 18-23 years old (M=19.53, SD=1.11), with no known learning disabilities or cognitive impairments were recruited. Figure 5.A.1 shows the participants age and gender distribution. All participants had normal or corrected vision. Participants were recruited via the SONA reward system from the participation pool of Nottingham Trent University. Participants that have already taken part in Design A (Chapter 4) were excluded to avoid familiarity to the task bias.

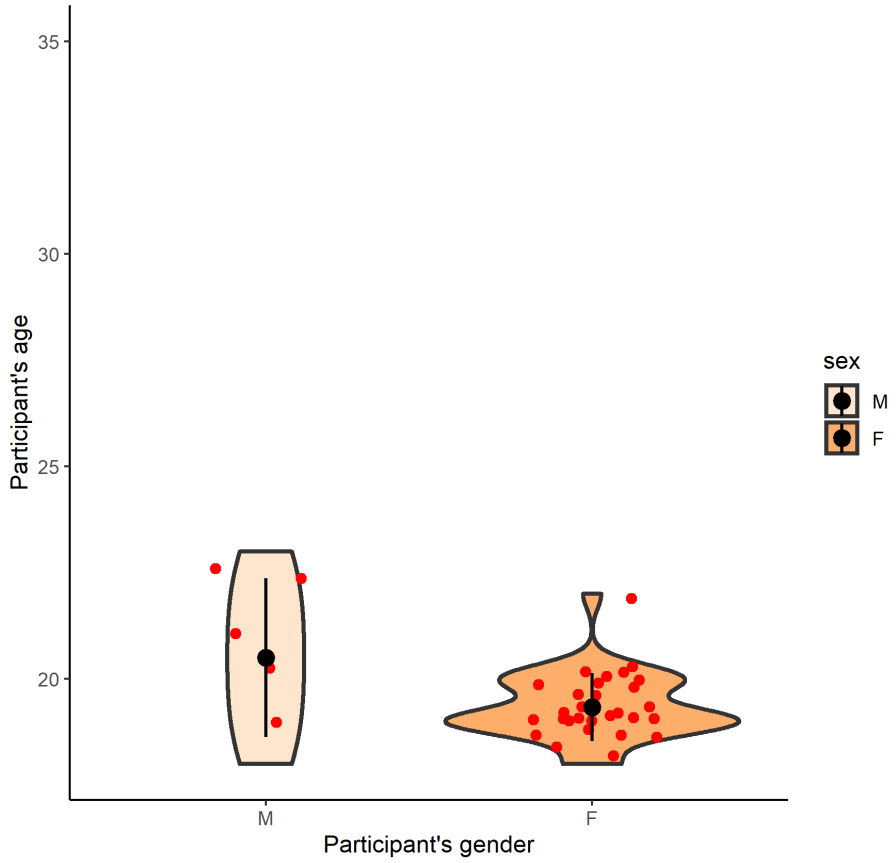


Figure 5.A.1 This violin plot represents the age and gender frequency distribution of participants for design B. Each red dot represents a participant.

A.2.3. Materials

The visual stimuli used for this study was a green dot that moved on a 16 location array of grey dots on a black background. The code that generates the location patterns was the same as used in previous studies. For the needs of the design, 6 different locations sets were used (Figure 5.A.2). The exact coordinates of each location set can be found in Table 5.B.1. All the cartesian points within a location set, were designed to be equidistant to avoid internal distance bias during the eye-tracking. It is important to mention that the same experimental coding and software that was used in Design A with Gazepoint GP3 was now used on Design B with EyeLink 1000 (SR Research).

set	dot	x	y	set	dot	x	y
LS1	1	-0.25712	-0.30644	LS2	1	0.08964	-0.3346
LS1	2	-0.06016	-0.34116	LS2	2	0.28284	-0.28284
LS1	3	-0.32552	-0.11844	LS2	3	-0.24492	-0.24492
LS1	4	-0.12856	-0.15316	LS2	4	-0.05172	-0.19316
LS1	5	0.0684	-0.18796	LS2	5	0.14148	-0.1414
LS1	6	0.26536	-0.22268	LS2	6	0.3346	-0.08964
LS1	7	-0.19696	0.03476	LS2	7	-0.38636	-0.10356
LS1	8	1.11E-17	4.00E-05	LS2	8	-0.19316	-0.0518
LS1	9	0.19696	-0.03468	LS2	9	4.00E-05	-4.00E-05
LS1	10	-0.26536	0.22268	LS2	10	0.19324	0.05172
LS1	11	-0.0684	0.18796	LS2	11	0.38636	0.10356
LS1	12	0.12856	0.15324	LS2	12	-0.3346	0.08964
LS1	13	0.32552	0.11852	LS2	13	-0.1414	0.1414
LS1	14	-0.1368	0.37588	LS2	14	0.0518	0.19316
LS1	15	0.06016	0.34116	LS2	15	0.245	0.24492
LS1	16	0.25712	0.30644	LS2	16	-0.28284	0.28284
set	dot	x	y	set	dot	x	y
LS3	1	-0.24492	-0.245	LS4	1	0.20848	-0.27668
LS3	2	-0.3346	0.08964	LS4	2	0.36816	-0.15628
LS3	3	-0.19316	-0.0518	LS4	3	0.0244	-0.19852
LS3	4	-0.05172	-0.19324	LS4	4	0.18408	-0.07812
LS3	5	0.08964	-0.3346	LS4	5	0.34384	0.0422
LS3	6	-0.28284	0.28284	LS4	6	-0.31944	-0.24076
LS3	7	-0.1414	0.1414	LS4	7	-0.15976	-0.12036
LS3	8	4.00E-05	-4.00E-05	LS4	8	0	-4.00E-05
LS3	9	0.1414	-0.14148	LS4	9	0.15976	0.12036
LS3	10	0.28284	-0.28284	LS4	10	0.31944	0.24068
LS3	11	-0.08964	0.3346	LS4	11	-0.34384	-0.0422
LS3	12	0.0518	0.19316	LS4	12	-0.18408	0.07812
LS3	13	0.19316	0.05172	LS4	13	-0.0244	0.19852
LS3	14	0.3346	-0.08964	LS4	14	0.13536	0.31884
LS3	15	0.10356	0.38636	LS4	15	-0.20848	0.27668
LS3	16	0.38636	0.10348	LS4	16	-0.04872	0.397
set	dot	x	y	set	dot	x	y
LS5	1	0.37588	-0.1368	LS6	1	-0.0302	0.34508
LS5	2	0.34116	0.06016	LS6	2	-0.22948	0.32772
LS5	3	0.22268	-0.26536	LS6	3	0.483	0.21612
LS5	4	0.18796	-0.0684	LS6	4	0.28372	0.19876
LS5	5	0.15324	0.12856	LS6	5	0.08452	0.18132
LS5	6	0.06948	-0.39392	LS6	6	-0.11476	0.16388
LS5	7	0.03476	-0.19696	LS6	7	-0.31396	0.14644
LS5	8	4.00E-05	0	LS6	8	0.39844	0.03492
LS5	9	-0.03468	0.19696	LS6	9	0.19924	0.01748
LS5	10	-0.0694	0.39392	LS6	10	-4.00E-05	4.00E-05
LS5	11	-0.11844	-0.32544	LS6	11	-0.19924	-0.0174
LS5	12	-0.15316	-0.12856	LS6	12	-0.39852	-0.03484
LS5	13	-0.18788	0.0684	LS6	13	0.31396	-0.14636
LS5	14	-0.22268	0.26536	LS6	14	0.11468	-0.1638
LS5	15	-0.30636	-0.25704	LS6	15	-0.08452	-0.18124
LS5	16	-0.37588	0.13688	LS6	16	-0.2838	-0.19868

Table 5.A.1 Shows the x and y coordinates for the 6 different location sets that were created for study 4. Each location contained 16 different points, that were equidistant. Those cartesian points created the appropriate equidistant array for our visual stimuli to draw on. A visual representation of each location set can be found in Appendix E (Figures E.1-E.6).

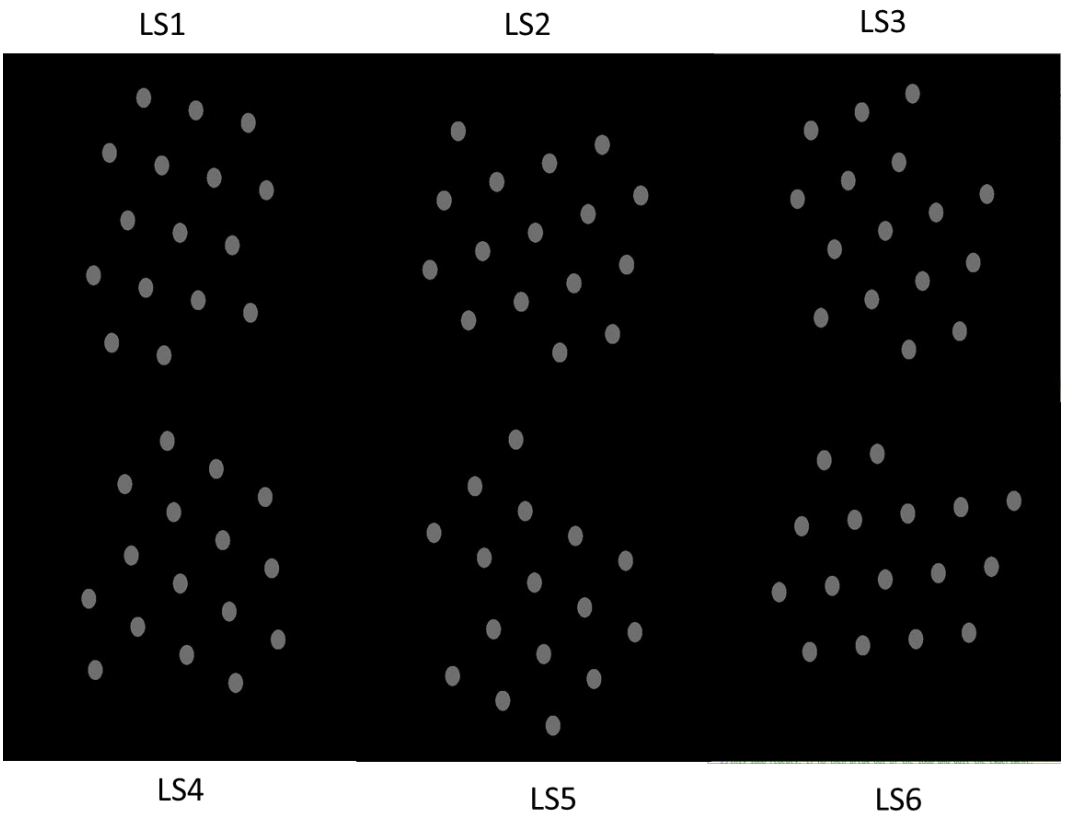


Figure 5.A.2. This figure is a visual representation of the 6 different location sets (LS1, LS2, LS3, LS4, LS5, LS6) that were used in Design B. Each location set contained 16 different points, that were equidistant.

A.2.4. Equipment

An EyeLink 1000 (SR Research Ltd., Mississauga, Canada) eye-tracker was used, on a monocular chinrest recording mode (right eye), with a sampling rate of 1000 Hz, resulting in 1000 eye samples per second. The task was presented on a 24" monitor with 1280 x 1024-pixel resolution and it's dimensions were 533 x 299 x 38 mm. The task was operating on a Windows x 64 Intel® core i7 laptop. A chinrest was placed on top of the desk, facing at the monitor at 50 cm distance. Finally, external to the laptop's system speakers were used that were always used at a sound level of 50%. In Table 5.A.2. is presented the different dimensions of stimuli (dot/target), AOI and display, and the relevant visual angles. The visual degrees were calculated with the use of the online SR Research calculator, (https://www.sr-research.com/visual-angle-calculator/).

	Screen	AOI	Target
Dimensions	1280 x 1024 pix	Radius AOI =45.84 pix (2.75 times the radius of target/dot)	Diameter =33.33 pixels Radius = 16.67 pixels
Size in Pixels	1310720 pix (width * hight of monitor)	6601 pix (3.14 * Radius AOI)	873 pix (3.14* Radius Targer/ Dot)
Horizontal Visual angle	53.01	4.5	1.63
Vertical Visual angle	39.24	2.89	1.05
Distance from Screen 500 mm (50 cm)			
Dimensions of monitor 533x 299 mm			

A.2.5. Design

Same as in design A, but this time with a small but crucial experimental addition in the trial. *The crucial difference between design A and this design B is that an additional beep sound was used in the design as a form of feedback.* If the participant had correctly guessed where the next dot would appear then the beep sound wouldn't be heard. Otherwise, the beep sound would be heard at the same time the next stimulus was drawn to the location. Eye-samples in within the radius of 2.75 times the size of the dot were counted as hits in the area of interest (AOI). Specific timings about the trials are given in the procedure section (Figure 5.A.3).

Counterbalancing & randomization of trials.

All trials for the task were counterbalanced for location set per participant, while the sequence order and the locations allocation were randomized for each participant. Each sequence wasn't allowed to appear immediately after itself. Each location of the array was allocated on a single

sequence item, to avoid overlapping of locations across sequences. All trial files were generated using R.3.2.5 version (R Core Team, 2013).

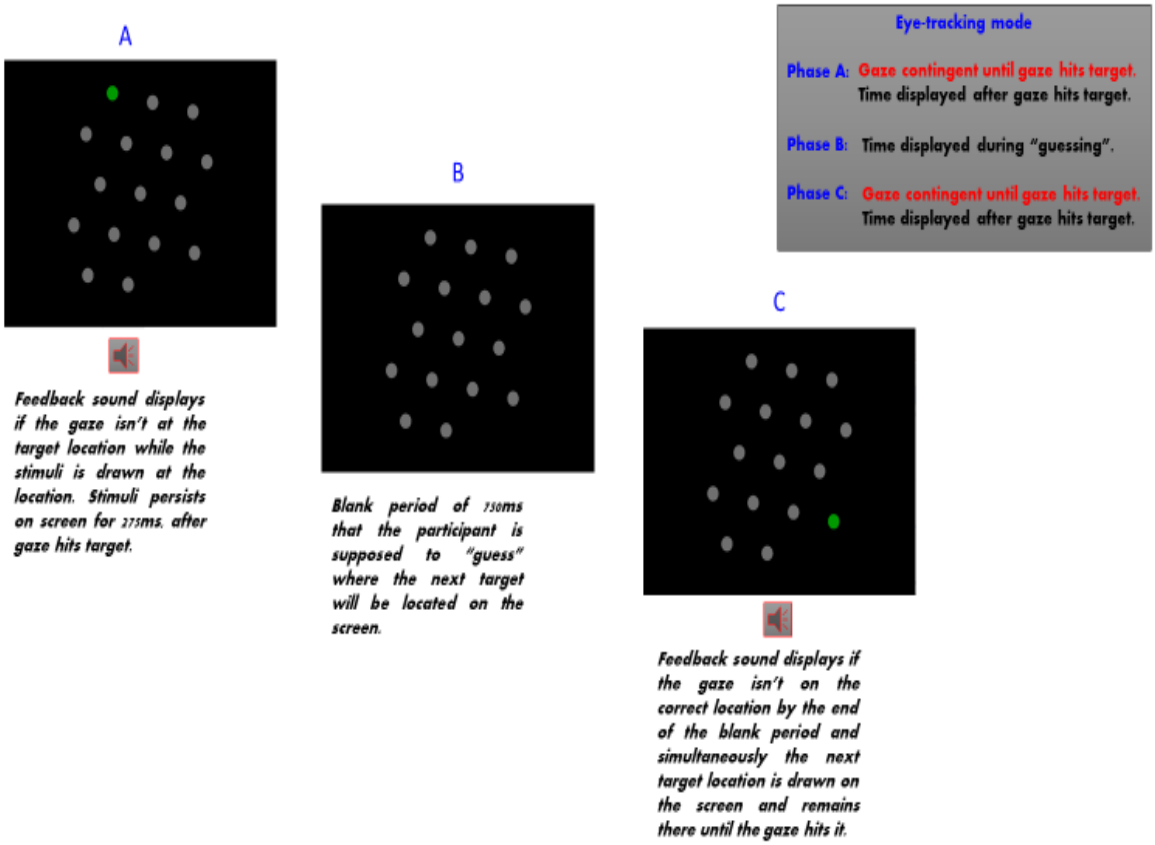


Figure 5.A.3 This figure represents an experimental trial in Design B.

A.2.6. Procedure

The experimental session procedure consisted of 5 different stages:

1. Participant's information, Consent form & Demographics: Same as Design A.

2. **Verbal instructions about the procedure & positioning for eye-tracking:** In this stage the participant was asked to turn off his mobile phone or any other electronic devices he might had with him and sit comfortably in front of the experimental monitor, so that he could be given the verbal instructions about the task. The verbal instructions for the task were the following:

"*In this task you will see a green dot moving in different locations. Every time the green dot appears in a location you will listen to a beep sound. All you have to do is guess where the dot will appear next by looking at the location that you think the green dot will appear next. The better you become at the guessing, the less often you will hear the beep sound*". Then the participant was asked if he had any questions about the instructions. If so, the researcher answered the question and moved on to the next step of positioning for the eye tracking. At this time point the participant was informed about the importance of keeping his head stable on the chinrest until the end of the task. Then the researcher explained the procedure of eye-tracking as follows: *"Before you begin the experiment, you will have to do a small task that last about a minute. All you must do here is look at the centre of a dot that will be drawn in different locations on the screen. At this task you are not asked to guess, but just look at the centre of the dot. After this task is finished, you will move to the experimental task, where you will start the guessing".* Then the researcher positioned the head of the participant on the chinrest and pressed enter on the open framework on the laptop. Then a window appeared asking if the participant was ready to be calibrated. The researcher pressed "enter" the calibration process began.

3. Calibration/ Validation: As part of the procedure of conducting an eye-tracking experiment, participants did a 9-points calibration task, with a maximum 0.8 degrees calibration error. If the calibration was successful, the participant moved on the experimental task. Otherwise, the calibration process was repeated, until it was successful. A new calibration/validation process was occurring before the beginning of each experimental task. Usually the calibration/validation process lasted less than 1 min. At the end of a successful calibration a window was drawn saying that the calibration was successful indicating that the validation process could start. At the end of the validation process, the researcher pressed "esc" and the experimental task began.

4. Experimental Task: In each task, each trial began with a green dot been drawn on top of a grey dot - on a array of total 16 grey dot locations- until the participant looked at it and remained there for 275ms while a blank period of the grey dot location array followed for 750ms. If the participant had looked at the next location that the next green dot would appear during the blank period (before the actual stimulus is drawn), then the beep sound wasn't displayed. Otherwise, the sound was displayed at the same time that next stimulus was drawn on the screen. The time between the beginning of the blank period until the first eye-sample on the next location target, the task was gaze contingent, securing that the dots have been seen otherwise the task wasn't progressing. The green dot was considered as seen if there was a hit within a radius of 2.75 times the size of the dot. The 750ms "blank" period is the crucial experimental period that will show if SL occurs. At the end of the task, a black window was drawn on the screen that thanked the participant for his participation in the experiment. The window remained there until the participant pressed "space" on the keyboard.

5. Debrief about the research aims of the study: Same as design A.

The total task procedure lasted about 15 mins.

A.3 Results and Critical Evaluation of Design B

This section consists of a basic visualisation of the learning occurring during Design B, and additionally provides some critical evaluation about the current flaws and potential implications of the new methodology in different experimental set ups.

A.3.1. Outcomes of Design B

In Design B, a visualisation of the outcome was used to examine whether learning is achievable in a 3 items sequence, and if the new additions to the design have added power to the outcome

(R scripts for this chapter can be found in Appendix G). To observe the learning in the task we had to clean the data and select the eye-samples that reflected learning. For this design demonstration, trials that consisted of eye samples to the first location of a new sequence (transition from the final position of the current sequence to the first position of the next sequence) were not removed from the data. However, it must be acknowledged that these trials are exogenous to the sequence learning process and in some way random (1/3 chances to guess correctly if learning within all sequences is successful due to only being 4 sequences with no sequence repeated immediately after itself and 1/16 chances to guess correctly if learning within all sequences is unsuccessful). This issue will be addressed in the next section (section B), where a comparison between those exogenous to the sequence trials and the endogenous to the sequence trials will be compared. But for the purpose of this design evaluation, they were included in the figures and descriptive statistics. Learning was calculated as the averaged eye samples on the target location across the endogenous sequence positions for each task. For the 3 dots task the eye samples in target locations of the second and third items of the sequences were categorized as learning data. Then the percentage of learning was calculated as an overall of the eye-samples on the target location divided by the maxim possible number of eye-samples that could occur during the blank period of 750 ms, which was 750. Similar to design A, a polynomial regression (method lm, y ~ poly(x,3)) was fitted to the data to extract a first impression about the form of the data.

As we can see in Figure 5.A.4 there is an increase to the learning outcome in comparison to Design A, with the percentage of eye-sample on the target location increasing from 7.5% to 12.5%. That increase, in terms of pure numeric eye-sample values, is also more powerful (100 samples, Figure 5.A.5) in Design B, as a more powerful equipment was used that increased the sampling power. It is important to mention at this point that although the numeric values still seem low in terms of percentage, there is a period where the participant has to do the eye-

movement planning (250ms), so it could be easily argued that the actual percentage would be a proportion of (hits on target/500) rather than (hits on target /750) as it is presented here. However, since the movement planning differs and can start at different points for each individual during the trial, it was decided that it's clearer to keep as the crucial time window all the guessing phase of 750ms, even if that lowers our numeric values. Now we are certain that the number of eye-samples on the target location at the 40th repetition, can't be due to randomness as they are ~100 and therefore we are overpassing problems such as the effects of low numeric values of Design A.

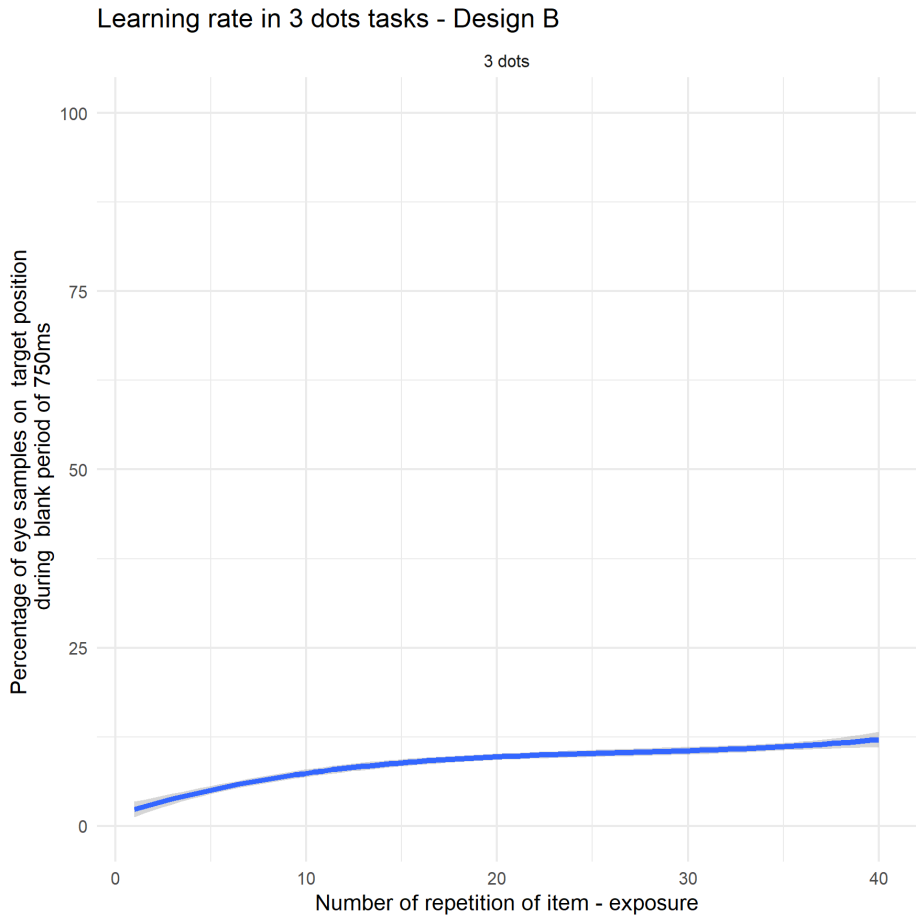


Figure 5.A.4. This plot visualises the learning rate across sequences in in the 3 dots task for Design B. On x-axis is the number of occurrence / exposure and on the y- axis is the percentage (%) of eye-samples on the target location during the 750ms blank period. Reminder that there is a maximum of 750 eye-samples per trial. Error bars represent 95% CIs.

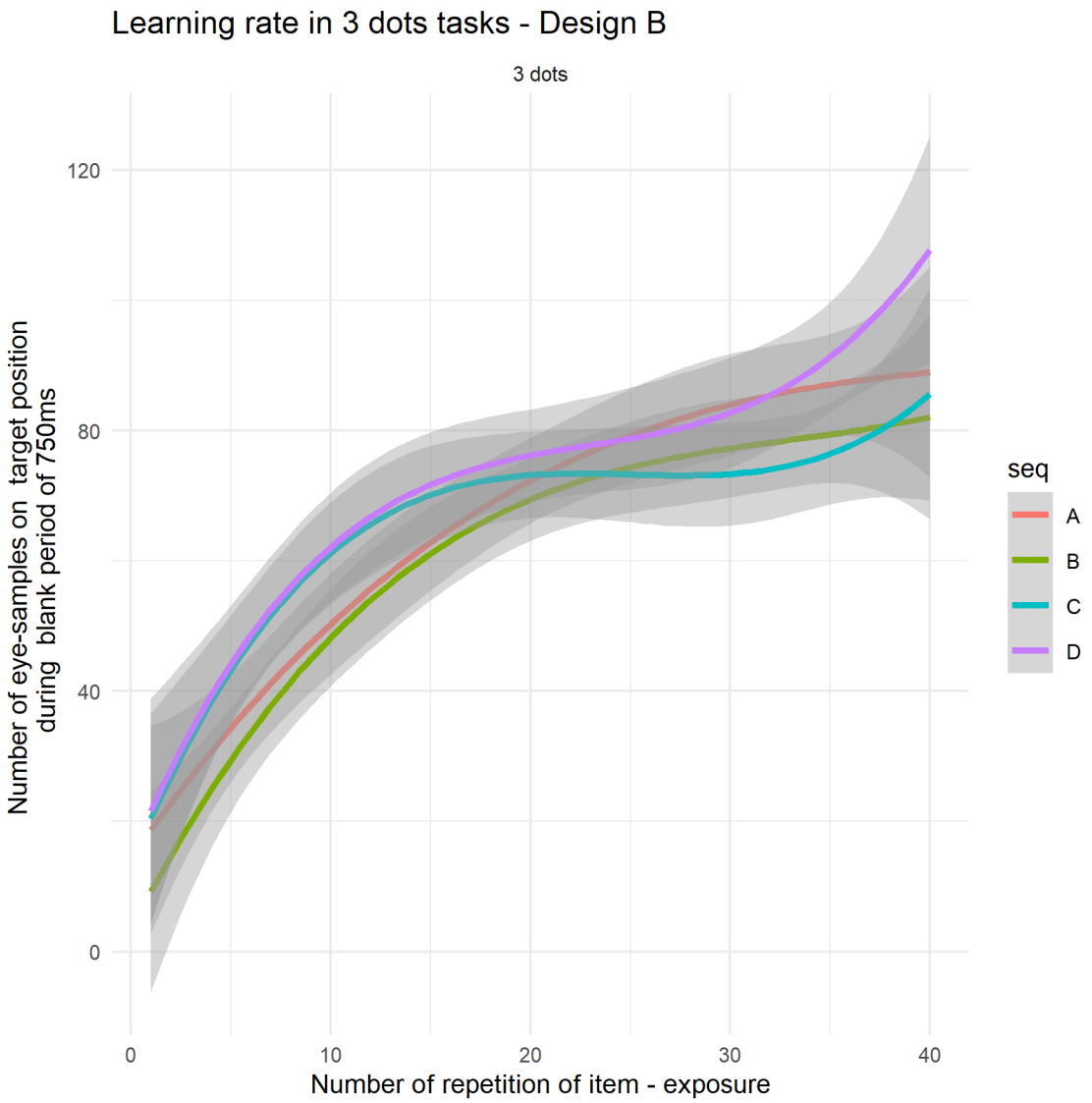


Figure 5.A.5 This plot provides a closer look into the effects across sequences in the 3 dots task for Design B, for each individual sequence (A, B, C, D). On x-axis is the number of occurrence / exposure and on the y- axis is the number of eye-samples on the target location during the 750ms blank period. Reminder that there is a maximum of 750 eye-samples per trial. Error bars represent 95% CIs.

As it can be seen in Figure 5.A.6 there is still a variance on participants performance, however only a few participants fail to demonstrate learning (022, 033) and still their values are non-zero. Additionally, there are plenty of participants that demonstrate great learning (007, 016, 024) but their learning curves have different peaks and shapes demonstrating that exposure and tiredness can affect differently individuals.

Figure 5.A.6. This plot shows the learning across sequences in the 3 dots task for Design B, for each individual subject (001-036). On x-axis is the number of occurrence / exposure and on the y- axis is the number of eye-samples on the target location during the 750ms blank period. Reminder that there is a maximum of 750 eye-samples per trial. Error bars represent 95% CIs.

A.3.2. Critical evaluation of Design B

This improved design B, suggests that the new methodology can give powerful results but also overcome all the problems that were highlighted in the introduction of this chapter with previous methodologies. Previous methodologies used to assess learning solely at the end of the learning process (Saffran, Aslin & Newport, 1996; Slone & Johnson, 2018), but this novel methodology captures the learning while it occurs and develops. In other words, it shows the learning during the time course of the process, for each of the items of the sequential learning. The precise recording of raw eye-samples from the first exposure until the last exposure of each sequence, allowing us to infer processes that occur within sequences and across the task,

and by adding a continuity into the mechanism rather than examining it as a learned / non-learned process. The precise timing recordings within the transitions of each sequence can be used to refer back to encoding and chunking processes that occur during sequential SL but also enlighten us about the effectiveness of positioning an item in a specific order in a sequence.

Something else that could have potentially explained the improvement in performance is the nature of the instructions. The participants in this task didn't have to "simply guess" where the next dot will appear. The instructions in Design B were "*In this task you will see a green dot moving in different locations. Every time the green dot appears in a location you will listen to a beep sound. All you have to do is guess where the dot will appear next by looking at the location that you think the green dot will appear next. The better you become at the guessing, the less often you will hear the beep sound".* Even though the instructions had an implicit nature (not explicitly mentioning the word learning in the instructions or mentioning any information about the existence of patterns or sequences), they could still prompt the participant by hinting that the guessing isn't random and therefore improvement in the performance can be noticed by receiving less beeps.

Additionally, it minimises effects common in the literature, such as the effect of stimuli complexity (MacKenzie, Aslin & Fiser 2011). The usage of a green dot on an array of grey dots, is actually minimising the perceptual load of stimuli in the task and provides an understanding of the actual SL mechanism on a minimally loaded perceptual format. However, that doesn't exclude the usage of more complex stimuli (different domain- audio, or load-perceptually complex stimuli) in future research manipulations. The findings on the current thesis can be used as the baseline of stimuli complexity for future research and allow further exploration of SL during more complex stimuli in different domains.

Previous studies included bias of reward learning processes (Abe et al., 2011; Freedberg et al., 2016), bias deriving from the perceptual complexity of the stimuli used to explore the SL process (Robinson & Pascalis, 2004; Saffran et al., 1996) and sometimes bias over the implicit nature of the task in imitation paradigm tasks (Hayne, Boniface & Barr, 2000) or familiarization paradigm tasks (Arciuli & Simpsons, 2011). The new paradigm that we suggest in this chapter, due to its minimal stimulus complexity and the nature of the instructions (implicit learning, non-explicit instructions, simple guessing of dots, no information about temporal sequences) manages to overcome these issues. More specifically, this new methodology has internal validity (Andrade, 2018) as the design (gaze contingent eye-tracking, guessing process, minimal stimuli complexity), conduct (implicit instructions) and analysis (eye-samples on target location during time course of learning) provide a clear understanding of how SL occurs without bias. Similarly, it has external validity (Andrade, 2018), as it can be generalised to other concepts, and more precisely it demonstrates high ecological validity as the experimental procedure imitates the real-life SL processes, therefore, can be generalized to real-life settings. The negative feedback sound was introduced in the task to increase participants' motivation (Burgers, Eden, van Engelenburg, & Buningh, 2015) and therefore engage more with the task. However, it should be acknowledged that the multisensory input of the beep sound and visual dot could have caused an increase in the cognitive load (Broadbent et al., 2018; Brünken, Steinbacher, Plass & Leutner, 2002), and this is something that future work may want to examine.

To conclude, before moving into the next conceptual research questions of the current thesis, it is important to highlight the importance of this design for the general field of cognitive psychology and SL. It is a method that it's programming and structure allow multiple experimental manipulations, on many eye-tracking systems, across different domains, age ranges (infants, children, adults) and could be used as part of further investigation of SL

mechanism in special populations such as autistic (Jones, Tarpey , Hamo , Carberry , Brouwer & Lord, 2018) or ADHD (Parks & Stevenson, 2018; Tenev, Markovska-Simoska, Kocarev, Pop-Jordanov, Müller & Candrian, 2014) population that have demonstrated differences on SL tasks in comparison to typical population. Additionally, further explorations and improvements could be added to increase the ecological validity of the task, depending on the specific experimental set up and research question.

B. Sequential Statistical Learning: Which transitions represent purely sequential SL in Design B?

B.1. Rationale

In this section, we will examine sequential SL with the use of Design B across different sequence length tasks (2 dots, 3 dots, 4 dots, 2&3 dots, 2&4 dots, 3&4 dots). In the previous section, we looked at the data only from 1 task, the task with the 3 dot sequences. In this section will preview all 6 tasks and we will examine how the transitions in each sequence represent sequential SL.

B.2. Methods

B.2.1. Ethics

Same as stated above in Section A.2.1

B.2.2. Participants

Same as in stated above in, Section A.2.2.

B.2.3. Materials

Same as stated above in, Section A.2.3.

B.2.4. Equipment

Same as stated above in, section A.2.4.

B.2.5. Design

The outcome variable in this study was the number of eye-samples on the target location. The predictor variables in this study were the sequences length (2 dots, 3 dots or 4 dots) and the type of task (mixed sequences learning task or non- mixed sequences learning task). Six conditions were created:

(a) Task 1 contained 4 sequences (A, B, C, D) that consisted of 2 dots each (2 dots task),

(b) Task 2 contained 4 sequences (A, B, C, D) that consisted of 3 dots each (3 dots task),

(c) Task 3 contained 4 sequences (A, B, C, D) that consisted of 4 dots each (4 dots task),

(d) Task 4 contained 4 sequences (A, B, C, D) in total, 2 of which consisted of 3 dots and 2 of which consisted of 2 dots (2&3 dots task),

(e) Task 5 contained 4 sequences (A, B, C, D) in total, 2 of which consisted of 2 dots and 2 of which consisted of 4 dots (2&4 dots task),

(f) Task 6 contained 4 sequences (A, B, C, D) in total, 2 of which consisted of 3 dots and 2 of which consisted of 4 dots (3&4 dots task).

Task 1, 2 and 3 involve sequences with the same length only (i.e.non-mixed length). Tasks 4, 5 and 6 are mixed length conditions, because they involve sequences of different lengths (2 of one length, 2 of another). Every task contained 40 occurrences (repetitions) of each sequence, and each sequence was not allowed to be presented twice in a row. Each task contained a different number of trials as shown in Table 5.B.1, because it consisted of different sequence lengths and mixtures of lengths.

Each trial (Figure 5.B.1) began with a green dot – on an array of 16 dot locations – until the participant looked at it and rested their gaze for 275ms. It was then followed by a blank grey dot period for 750ms. And then the next in order sequence item-green dot was presented on the screen. The number of trials for each task can be found in Table 5.B.1. The target location in each trial was defined as the next location in the sequence. If participants have successfully learned the sequences, after the visual presentation of the 1st item (dot) of the sequence, they were expected to look immediately to the location of the next item in the sequence. Participants were required to look at the green dot before moving into the next trial and a beep sound was displayed as feedback if the participant failed to anticipate the dot location prior to it changing green. The beep sound was omitted if the participant correctly anticipated the location of the next dot prior to it being illuminated green. Eye-samples within the radius of 2.75 times the size of the dot were counted as hits in the area of interest (AOI). Specific timings about the trials are given in the procedure section. Visual angles are same as in section A.2.6 (Chapter 5).

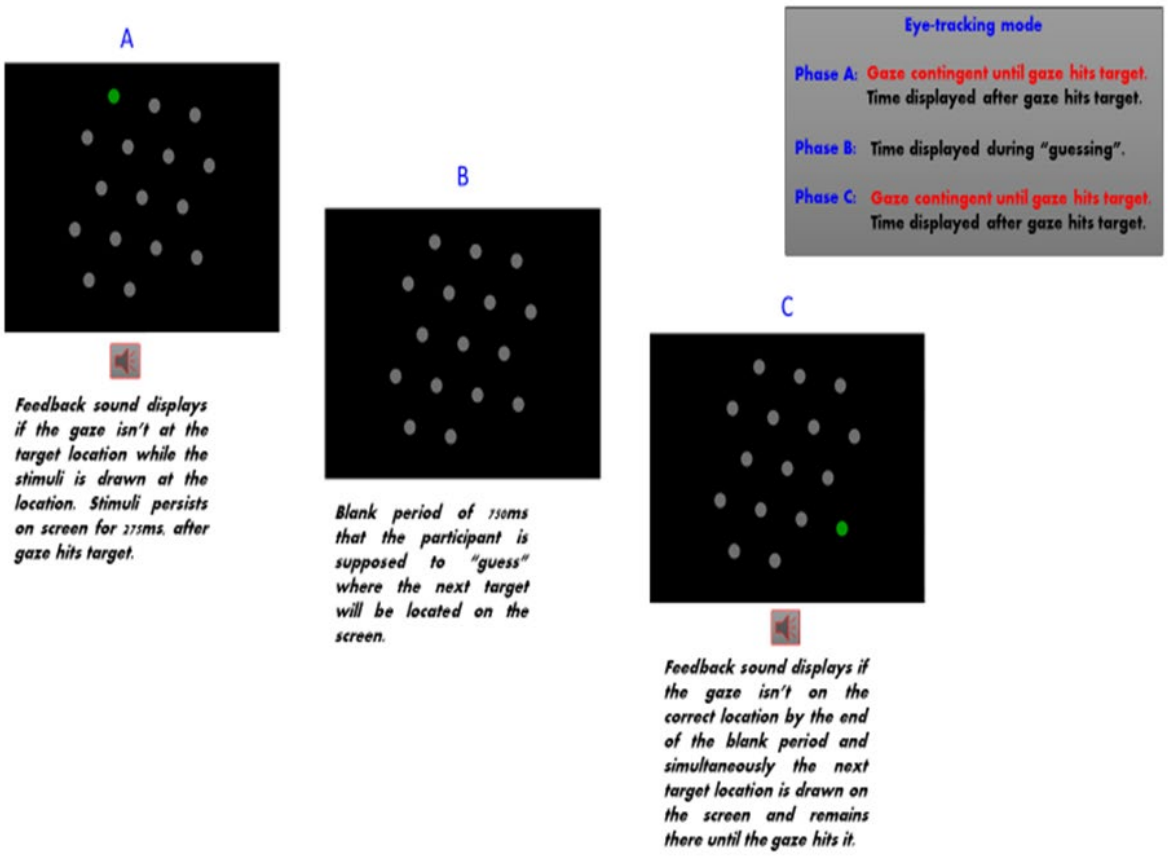


Figure 5.B.1 This figure is an example trial in the experiment.

Task	Total trials	Calculation
Task 1- 2 dots	320	4 sequences * 40 repetitions each sequence * 2 dots each sequence
Task 2- 3 dots	480	4 sequences * 40 repetitions each sequence * 3 dots each sequence
Task 3- 4 dots	640	4 sequences * 40 repetitions each sequence * 4 dots each sequence
Task 4- 2&3 dots	400	(2 sequences * 40 repetitions each sequence * 2 dots each sequence) + (2 sequences * 40 repetitions each sequence * 3 dots each sequence)

Task 5- 2&4 dots	480	(2 sequences * 40 repetitions each sequence * 2 dots each sequence) + (2 sequences * 40 repetitions each sequence * 4 dots each sequence)
Task 6- 3&4 dots	560	(2 sequences * 40 repetitions each sequence * 3 dots each sequence) + (2 sequences * 40 repetitions each sequence * 4 dots each sequence)

Table 5.B.1 This table shows the number of trials for each experimental task in this study.

All trials were counterbalanced for location set per participant. We created a total of 6 different potential matches of location set (location sets) – task (6 tasks) for all 36 subjects, in order to ensure that the learning observed is not related to a specific visual component of the visual allocation of the locations array but it is purely driven by the task. Additionally, the order of the tasks was semi-counterbalanced to avoid certain tasks reflecting order task effects. The sequence order and the locations allocation were randomized for each participant. No sequence was followed immediately by itself. Each location of the array was allocated to a single sequence item, to avoid overlapping of locations across sequences.

B.2.6. Procedure

Same as stated above in, Section A.2.6., with the only difference that in between tasks, an approximate 5-minute break was introduced. The total duration of the experiment was 2 hours.

B.3. Analysis

B.3.1 Data Visualization- Is there evidence of learning across the 6 tasks?

Before conducting any statistical analysis to the data, we were interested in exploring the learning curves of the data with locally estimated smoothing functions. To do so, we created

with the use of R Studio (R Core Team, 2013) a descriptive plot that fitted a locally estimated smooth linear regression (method loess y ~ x) to our data to get a first impression about the learning rate of for each task (Figure 5.B.1). All plotting and analysis code for this chapter can be found in Appendix N. Learning was calculated as the averaged eye samples (of all 36 participants) on the target location across positions for each task. Specific details about the rationale of data extraction/selection for each research question differs and will be given prior the statistical modelling of that research question.

This indicates that learning was rated as the average of eye samples on the specific target location across positions. For 2 dots task, we averaged the eye samples that were in the correct location for the second positioned item of the sequence (position 2), for the 3 dots task, we averaged the eye samples that were in the correct location for the second and third positioned items of the sequences (position 2 and position 3) and similarly we did for the 4 dots task. For the mixed length sequences task (2&4, 3&4, 2&3) we had 2 different approaches depending on our research question.

In this section the term transition will be used. A transition in this data analysis represents the raw eye-samples during blank period (750ms) during the transitioning between the current dot (1) and the next dot (2). If we place this context in the actual experimental set up of sequences, this means that some transitions will occur within a sequence (internal, or within-sequence transitions) and some will occur between transitions (exogenous or between-sequence transitions). In the case of a 3-dot task for example, where sequence A is followed by sequence B, and each dot represents an ordered item of a sequence, the participants will see the dots A1, A2, A3, B1, B2, B3. The transitions: A1-A2, A2-A3, B1-B2, B2-B3 are internal (within sequence transitions) as they all take place within the same sequence. The transition A3-B1 is an exogenous (between sequence) transition as it occurs between two different sequences. Therefore, the raw eye-samples that are recorded during the blank period on every 1st target

position, are representing a 1st transition to a sequence, which is labelled as exogenous. When we looked at learning within task, and how well participants learned, we averaged the eye samples that were in the correct location for each position/transition, for each task.

As we can see in Figure 5.B.1, learning can be demonstrated across all six tasks, and across all target positions. However, the learning for the 1st position of each target (represents the transition to the 1st item of a new sequence (between-sequence)) is lower than the within-sequence transitions (learning o2 position 2, position 3 & position 4). That is something that can be predicted by the nature of the task, since eye samples on position 1 of a sequence (transition from the final position of the current sequence to the first position of the next sequence) are exogenous to the within sequence learning process and in some way random (1/3 chances to guess correctly if learning within all sequences is successful due to only being 4 sequences with no sequence repeated immediately after itself and 1/16 chances to guess correctly if learning within all sequences is unsuccessful). Therefore, it was decided to perform further exploration of the data and perform a comparison between the learning rate of within-sequence transitions (position 2, position 3 and position 4) vs. between sequence transitions (position 1). The new factorial variable was named type of transition and consisted of two levels, (a) the between-sequence transition (learning on position 1) and (b) the within-sequence transition (learning on position 2, position3, position 4). In figure 5.B.2, the learning for between-sequence transitions and within-sequence transitions has been aggregated across all sequences and all tasks.

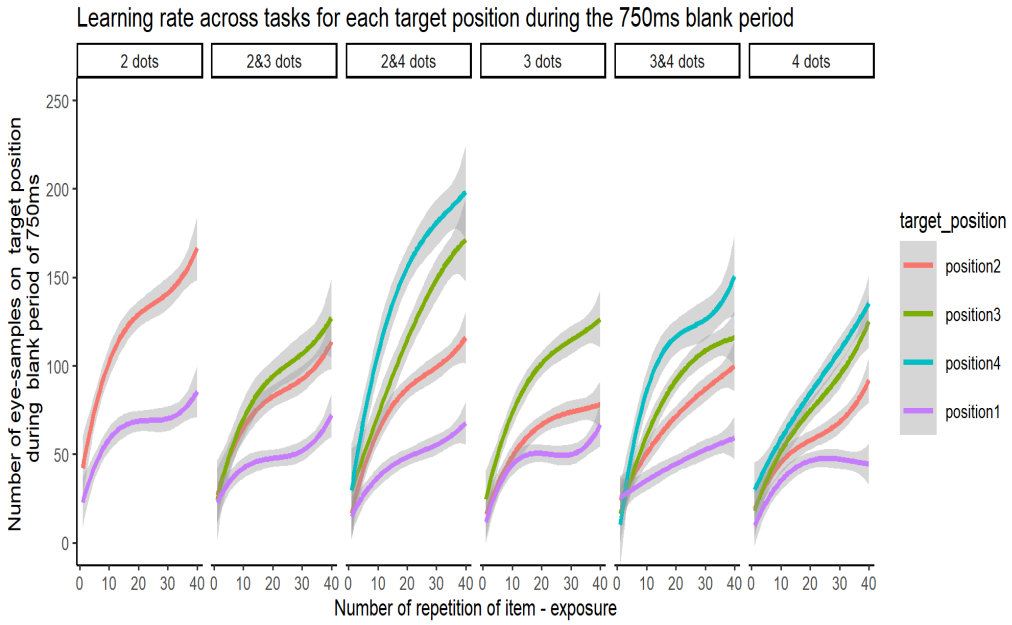


Figure 5.B.1 This plot shows the learning rate across the 6 different tasks (2 dots, 2&3dots, 2&4 dots, 3 dots, 3&4 dots, 4 dots) for each target position within a sequence. The x-axis is the number of occurrence / exposure and the y- axis is the number of eye-samples on the target location during the 750ms blank period. Note: there is a maximum of 750 eye-samples per trial. Error bars represent 95% CIs.

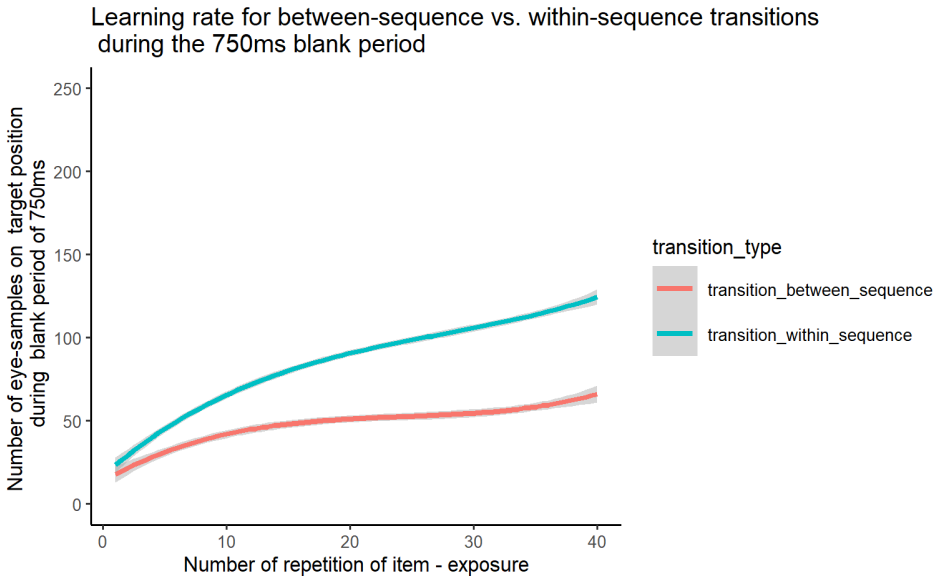


Figure 5.B.2 This plot shows the aggregated learning rate across all tasks for the different types of transitions (transition between-sequences (1st transition to a new sequence) vs. transition within sequence). On the x-axis is the number of occurrence / exposure and on the y- axis is the number of eye-samples on the target location during the 750ms blank period. Note: there is a maximum of 750 eye-samples per trial. Error bars represent 95% CIs.

As it is shown in Figure 5.B.2, the learning across the 2 types of transitions seem to differ. Therefore, further statistical analysis was designed to explore this relationship.

B.3.2. Statistical Modelling Approach

All data processing (scripts in Appendix C) and analysis (scripts in Appendix N) were conducted in R version 3.6.3 (R Core Team, 2013) with main tools the lme4 (Bates, Meachler, Bolker, & Walker, 2015) and tidyverse (Wickham et al., 2019) packages. After research aims were set, it was decided that the best statistical approach was the usage of Generalised Linear Mixed Effects Models- GLMMs (McCullagh & Nelder, 1989; Skrondal & Rabe-Hesketh, 2004). GLMMs provide flexibility to explore possible interactions of our independent variables (task, sequence length, position in sequence, occurrence- exposure, type of task (mixed length/ non-mixed)), provide the ability to predict the exact relationship of each independent variable with the expected outcome (learning rate) but also understand sources of random variability in the outcome. The main fixed effect that is being examined in this section is the type of transition: 1st transition to new sequence (exogenous transitions) vs. in-sequence transitions (internal transitions). The random effects in our data derive from the variable of sequences that it is clustered within participants. However, the models presented below, did not include random slopes. This was intentionally done, since the models including random slopes couldn't be computationally resolved. Random slopes are important for GLMMS, as they are used to model how the dependent variable (learning) changes as a function of the independent variables (occurrence, condition, etc.) while accounting for the variability introduced by the random effects (participants). These random effects capture the variations between different groups or clusters within our data. If the random slopes were included in the model, that would allow the model to estimate the individual-specific or group-specific relationships between the independent and dependent variables. This means that the relationship between learning (dependent variable) and occurrence, condition, etc. (independent variables) can vary across

different levels of the random effects (participants). By not including the slopes, we have lost this flexibility of the model to account for this variance and therefore we have a less accurate and realistic representation of the modelled data. The choice of removing the random slopes from the model, simplifies the model and allows it to be resolved. This is a common practice that has been recommended by other researchers such as Walker (2018). All the future models that will follow in this thesis, are in line with this practice and for computational reasons, they will not include random slopes.

Again, in this analysis, the outcome variable is the number of samples on the location of the next item in the sequence at each presentation of the sequence. Where there is no learning then the value will be zero. The triggering in the descriptive plots of the data in the methodological chapter 3 and earlier on in this chapter in the data visualisation, is the fact that there are many zero inflated values (eye samples), so it is necessary to adjust the primary GLMMs by introducing a Poisson transformation (Lambert, 1992; Casals, Langohr, Carrasco, Rönnegård, 2015) into the statistical model. This will account for the probability distribution of count data. Since the zero-inflated values represent the "non-learned" items and are meaningful for the interpretation of the data and the modelling, as they are part of the learning curve growth, no further manipulation was on dealing with them was applied in the statistical modelling. Similarly, the component of time was modelled and tested as a polynomial function, for the reasons explained in Chapter 4 (nature of learning growth curves). For this reason, 3 polynomial models (Table 5.B.2) were used to examine which occurrence- exposure function explains better the learning rate: (a) ~poly(occurrence,1) which refers to a linear function (x1), (b) ~poly(occurrence,2) which refers to a quadratic function (x2), and finally (c) poly(occurrence,3) which refers to a cubic function (x3). The cubic function [poly(occurrence,3)] provides a significantly better fit (ANOVA tested, see Table 5.B.3) in all

tasks and therefore was used as the basic structural component for the synthesis of more complex models.

Model type	*R model*	*Curve Shape*	*Meaning*
Baseline model	glmer(on_target_pre_total ~ 1 + (1|subno) + (1|seq:subno), data, family = poisson())	Linear shape	Learning isn't related to exposure
Shape model 1	glmer(on_target_pre_total ~ poly(occurrence,1) + (1|subno) + (1|seq:subno), data , family = poisson())	Linear shape	Exposure can predict learning in a linear function.
Shape model 2	glmer(on_target_pre_total ~ poly(occurrence,2) + (1|subno) + (1|seq:subno), data , family = poisson())	Quadratic shape	Exposure can predict learning in a quadratic function.
Shape model 3	glmer(on_target_pre_total ~ poly(occurrence,3) + (1|subno) + (1|seq:subno), data , family = poisson())	Cubic shape	Exposure can predict learning in cubic function.

Table 5.B.2 This table demonstrates the baseline statistical model and the testing curve shape models. The cubic shape model was significantly a better fit to the data for all tasks, and therefore was chosen to be the baseline model for the rest predictor factors in the next research questions.

Model Comparison	**$\chi 2$**	**P-value**
Baseline Model vs Shape Model 1	640013	<.001
Shape Model 1 vs Shape Model 2	80142	<.001
Shape Model 2 vs Shape Model 3	33958	<.001

Table 5.B.3 This table shows the ANOVA results from the statistical comparison of shape models in Table 5.B.2.

The above model comparison revealed that shape model 3 was the best fit to the data and therefore will be used as the baseline learning model to answer the following research

questions. From now on learning will be estimated as cubic function of occurrence. The code for these models can be found in Appendix N, section N.1.

B.3.3. Hypothesis modelling testing- Is there a difference in the learning depending on the type of the transition (1st transition (exogenous) vs. internal to the sequence transitions)? Is the learning of the 1st transition into the 1st item of a new sequence statistically different from the learning observed across transitions within the same sequence (e.g., for 3 dots task transitions to the 2nd and 3rd dot for each sequence)?

To answer this research question, we created a baseline and 2 hypothesis models, and then the model with the best fit (ANOVA comparison) was selected as the model describing better the data. The full R scripts for this analysis can be found in Appendix N, section N.2.

Model type	R model	Meaning
Baseline model	glmer(on_target_pre_total ~ poly(occurrence,3) + (1|subno) + (1|sequence:subno), data, family = poisson())	Exposure can predict learning in a cubic function.
Hypothesis model 1	glmer(on_target_pre_total ~ poly(occurrence,3) + transition_type+ (1|subno) + (1|sequence:subno), data, family = poisson())	Exposure can predict learning in a cubic function & the type of transition differentiates the learning outcome.
Hypothesis model 2	glmer(on_target_pre_total ~ poly(occurrence,3) * transition_type + (1|subno) + (1|sequence:subno), data, family = poisson())	Exposure can predict learning in a cubic function & the type of transition differentiates the learning outcome and that changes during exposure.

Table 5.B.4 This table demonstrates the baseline learning model and the testing hypothesis models.

Model Comparison	$\chi 2$	P-value
Baseline Model vs Hypothesis Model 1	492254	<.001

Hypothesis Model 1 vs Hypothesis Model 2	13206	<.001

Table 5.B.5. This table shows the ANOVA results from the statistical comparison of shape models in Table 5.B.4.

The above results suggest that the learning demonstrated by the 1st transitions in these data is significantly different from the learning of the in-sequence transitions and that relationship is changing over exposure. A table with the specific residuals can be found in Appendix N, section N.3, Table N.1.

B.4 Conclusions

These findings make sense since the learning of the 1st transition to a new sequence relies on the learning of the in-sequence transitions in the first place. The trials that consisted of eye samples to the first location of a new sequence (transition from the final position of the current sequence to the first position of the next sequence) are exogenous to the within sequence learning process and in some way random (1/3 chances to guess correctly if learning within all sequences is successful due to only being 4 sequences with no sequence repeated immediately after itself and 1/16 chances to guess correctly if learning within all sequences is unsuccessful). Therefore, the learning observed in the 1st transition of a sequence, could be either considered a baseline for the task since it represents "random" learning in the task or it could be considered a result of the learning that takes place within the sequence in a semi-random manner (The more sequences an individual learns, the better chances he has to guess correctly the 1st transition to a new sequence). For example, in the case of the 3 dots task, that consists of 4 sequences (A, B, C, D), an individual must have successfully segmented all 4 sequences (A, B, C, D) and all the within sequence transitions (A1-A2, A2-A3) to be able to reach the chance of 1/3 of guessing correctly for between sequence transitions. This means that the nature of the learning of 1st transitions changes over time within the task, as it starts as a random guess (1/16 locations on array) and evolves into a semi-dependent guessing that relies on the in-sequence

learning. Therefore, since the nature of the learning of the 1st transitions is distinguished from the learning that takes places within the within-sequence transitions (part of SL learning), the 1st transitions have been excluded from future data analysis in the next chapters. This decision is in-line with the thesis focus, that is fixating on within-sequences SL learning and relevant effects.

The temporal element of SL: Positioning effects on Statistical Learning (SL) during a sequential visual task- is SL an all or none process?

Chapter Summary

In this chapter we will examine the temporal element of SL during a sequence learning task, by examining positioning effects within sequences. As we saw in Chapter 2, most of the studies do not consider the time course of learning but the end product of it. Since sequence SL has a temporal component, the holistic perception of learning (all or none) of the parts of the sequence may not be valid, because each element of the sequence occurs in a unique timepoint and that timepoint is related with the presentation of the next item suggesting a serial process of learning. Firstly, we will examine the theories about the SL process and the different predictions that they would suggest on the learning outcome for each item positioned within a sequence. For the examination of this chapter, we will be using the data from Chapter 5. Then in the results section, we will perform the appropriate Generalized Linear Mixed Effects Models (GLMMs) to answer questions such as (a) Is there a consistent pattern of positioning effects across tasks? And if so (b) how do the positioning effects operate on mixed length sequential learning tasks? The end of the chapter discusses the implications of the current findings on current SL models and real-life applications of the findings on educational paradigms.

A. Introduction

As we saw in the literature review in Chapter 2, SL can be described as an implicit cognitive learning mechanism, where the learning occurs by the ability of extracting transitional probabilities and statistical patterns from the sensory input and forming them into a coherent unit of knowledge, which can be later used to retrieve the memory of the input or predict the next input (Aslin & Newport, 2012). That in other words suggests that in a sequence with items A, B, C, D, at the beginning of the SL process, people perceive these items as individual elements, however after a certain amount of exposure to the sequence, people will form a unit about the sequence ABCD and will be able to perform predictions about what element of the sequence will follow. So, if for example the have been exposed to the AB part of the unit they can successfully guess that the CD part will follow.

However, it isn't clear in the literature how this unit is being formed. There have been many theories developed to explain the SL process and multiple SL computational models that tried to clarify how SL occurs within a sequence and how it can affect the learning of each individual item of the sequence. In the following paragraphs we will present some of the most popular of these theories and models and their predictions about the forming of learning the unit ABCD.

One of the first people that tried to understand the SL process and forming of learning units was Estes. The Estes models (1959), suggest an all-or-none-learning understanding of SL. According to these models, learning occurs in a non-continuous manner (all or none), accounting for continuous learning by introducing the factor of stimulus – response connection (S-R). The idea is that you can form or not an S-R connection during an experimental trial, however you need plenty of these connections in order to be able to produce a correct response (exposure factor). In these models a probability of a correct response on an experimental trial is estimated as a proportion of stimulus elements conditioned to that response on the specific

trial. This conceptualisation of the learning process was innovative at the time that it was developed and widely used and developed during that period (Suppes & Atkinson, 1960; Estes, Hopkins, & Grothers 1960). These models were called pattern models (Estes, 1959; Suppes & Atkinson, 1960), and their main characteristic was that only one pattern (stimulus element) was allowed to be presented on each trial. That suggested that for example, in an A-B-C-D sequence, the learning would take place as pairs of SR associations, for AB, BC, CD, and therefore the unit of ABCD wasn't possible. This all or none learning approach was reflection on the S-R associations and not on higher level units (sequences of triplets etc.).

A new approach on the understanding of SL learning came by Atkinson (1961) and the incremental model. The fundamental idea behind this model was that a stimulus element can be in one of a finite number of conditioning states on any experimental trial. That suggests that on a single-stimulus version of this model with different number of possible responses, the maximum number of states varies with the number of possible response and therefore is not a simple function of reinforcement or motivation. That theory could have direct implications for the sequential SL paradigm as suggested in our methodological paradigm (Chapter 4). In our methodological paradigm, stimuli have a unique position in time and within a sequence, but response choices are constantly 16 (16 location array) suggesting that in a sequence of ABCD items we can consider AB, BC and CD as S-R pairs. Since ABCD can be allocated on an array of 16 locations, once some of those locations is learned, and some S-R associations are made, that affects the learning of the future S-R associations. In other words, it suggests a hierarchical structure that allows more flexibility between stimulus-response connection, recommending that in our ABCD sequence paradigm, the learning of the AB unit can affect the learning of BC unit (e.g., because the location of B is now known and so the remaining 15 locations can be discounted) and the learning of the BC unit can affect the learning of the CD unit.

Because SL is by definition a theory that uses a mathematical concept and relies on the probabilistic nature of events, it has been applied to numerous cognitive learning models that tried to encode human behaviour. Some of them are the mathematical models of memory (Norman, 1970) and the multistore model of memory from Atkinson and Shiffrin (1968). In both models, stimulus sampling theory (SST) (Estes, 1950) and the probability of certain stimulus occurring in a certain time period and therefore the pairing of that stimuli with a given response, was the key element component. Again, in these models we can observe the hierarchical element of SL, but also a shift to the focus on the timing/serial component by giving a whole different perspective on the meaning of SL procedure. Sequential SL has a temporal component because each element of the sequence occurs in a unique timepoint and that timepoint is related to the presentation of the previous and next items suggesting a serial process of learning. That could be reflected in our ABCD unit learning, as the ability of a person to learn parts of the sequence (e.g., learn the BCD unit without A, or learn the CD unit without B). The important element here is that we can either have one unit of knowledge ABCD if all 3 subunits are learned successfully (AB, BC, CD) but we can also allow the knowledge of subunits AB and CD as 2 individual sequences if the learning of the BC unit is unsuccessful.

In a series of experiments Fiser and Aslin (2005) investigated the ability to generalize knowledge that derived from a visual SL (multi-element shapes) task on new stimuli, in order to understand better how the encoding of information occurs during SL, by comparing the learning of units in triplets and pairs based on familiarity (random vs familiar). In their first experiment they found that base triplets were chosen more often than random triplets, but also that base pairs that were embedded in the triplets, didn't significantly differ from random pairs. In experiment 2 they wanted to examine whether the results of experiment one, were deriving from participant's inability to learn pairs. However, when participants were asked to perform a pair learned and pair novel task, participants preferred the known pairs over random pairs. In

experiment 3, they found that participants would more often select familiar pairs and familiar triplets over random shape combinations when scenes composed from only triplets or only pairs were randomly intermixed during the familiarization phase. Then they decided to use pairs and quadruples to examine the holistic or pair chunking encoding process of SL. The findings in experiment 4 suggested that both familiar quadruples and familiar pairs were preferred significantly over random combinations of elements. However, participants were unable to distinguish pairs embedded in the quadruples over random pairs, even after doubling the duration of familiarization. In their experiment 5, they used 3 types of trials: (a) the single trials that were noise elements with high-frequency that were later compared to the low-frequency shapes, (b) the quadruple trials, that were subparts of the two sextuples and were compared with random quadruples (c) the key test trials that strong vs. weak pairs were compared with random non familiar shape pairs. In all these tasks, participants were choosing more frequently the familiar rather than novel or high frequency elements, while strong pairs were selected more frequently than weak pairs. These findings suggest a completely new approach to encoding during SL, as they suggest that extraction of independent parts is possible (with highly coherent subsets of elements, of different levels of complexity from the scene in parallel and without an explicit task), but there were differences in performance, with groupings being remembered to a different extent, depending on whether they were part of a larger or complex cluster of elements. Furthermore, stronger pairs are encoded significantly better than weak pairs even when the weak pairs are presented more often during familiarisation. This study, even though it presents solid evidence about the encoding of visual SL, it doesn't account for the element of time and sequential SL, a context that is commonly used in language SL processes, since a lot of the language structures (e.g., syllables form into a word unit) come as temporal sequences. In all the series of the experiments the pairs, triplets and quadruplets were presented simultaneously, and not serially. That creates a gap in the literature about how

learning occurs in time when stimuli are presented serially. A great paradigm of serial SL is language learning and how infant learns to form their first words based on exposure of specific ordered syllables over time.

Additionally, even if we can retrieve information about the encoding of information, we don't have any information about the actual temporal process that led to the unit of knowledge, since all the tasks described, were capturing the outcome of the learning process rather than the process itself. That suggests that it could be possible to have an anticipation of pairs vs triplets during the encoding phase of a ABC unit, but this wasn't captured due to assessing the process by its outcome. That methodology didn't provide any information at all about the stages of building for example the triplet ABC, instead validating that ABC was learned as a unit at the end. It could have been the case that if B wasn't learned then the ABC unit couldn't be formulated, because the subunits (if they exist) of AB and BC that lead to the learning of ABC couldn't be formulated. Similarly, the quadruplet unit of ABCD could have occurred by learning AB, BC, CD, or AB, CD as 2 independent pairs (non-continuous) and then binding them into the unit of ABCD.

In order to cover this gap in the literature, we decided to investigate how visual statistical learning occurs during a serial sequence learning task, where items in the sequence occur one after another in a set order and record the learning of each item of the sequence individually from the beginning until the end of the task. The main idea behind this decision was to understand the role of each item in sequence and how does that affect the learning of units during the encoding phase.

The focus now of this chapter will be on whether the positioning of an item in a sequence can affect the learning outcome for that item, and if there are other factors such as sequence length, that can affect this positioning effect and why. Firstly, we are interested in investigating

whether there is a consistent learning pattern between the learned items of a sequence. That translates in the paradigm of the ABCD sequence, that we are interested to examine if there is a consistent performance learning pattern between AB, BC, CD. If that's the case, we can directly suggest that sequential SL is not an all or none learning process but has a continuity in time and that continuity could potentially reflect a hierarchical structure of unit learning in sequential SL.

Finally, once we identify if there are any consistent learning patterns, we will have to consider which factors drive those performance patterns, and basically if the consistency of the learning patterns relies on factors such as sequence length or mixture type of sequences (mixed sequences or non-mixed sequences tasks) in order to be able to extract some meaningful information about which factors affect the learning of unit ABCD in the current set up.

In order to answer the above questions, we used the experimental paradigm that we suggested in our methodological paradigm in Chapter 5 with sequences of 2 dots, 3 dots and 4 dots in mixed length and non-mixed length tasks. More details about the method and the design can be found in the section below.

B. Methods

Same as reported in Chapter 5, section B.2.

C. Results

C.1. Statistical Modelling Approach

All data processing (scripts in Appendix C) and analysis (scripts in Appendix H) was conducted in R version 3.6.3 (R Core Team, 2013) with main tools the lme4 (Bates, Meachler, Bolker, & Walker, 2015) and tidyverse (Wickham et al., 2019) packages. After research aims were set, it

was decided that the best statistical approach was the usage of Generalised Linear Mixed Effects Models- GLMMs (McCullagh & Nelder, 1989; Skrondal & Rabe-Hesketh, 2004). GLMMs provide flexibility to explore possible interactions of our independent variables (task, sequence length, position in sequence, occurrence- exposure, type of task (mixed length/ non-mixed)), provide the ability to predict the exact relationship of each independent variable with the expected outcome (learning rate) but also understand sources of random variability in the outcome. The main fixed effect that is being examined in this chapter is the item position. The random effects in our data derive from the variable of sequences that it is clustered within participants.

Again, in this analysis, the outcome variable is the number of samples on the location of the next item in the sequence at each presentation of the sequence. Where there is no learning then the value will be zero. The triggering in the descriptive plots of the data in Chapter 4 with the GP3 data and in Chapter 5 with Eye-Link data, is the fact that there are many zero inflated values (eye samples), so it is necessary to adjust the primary GLMMs by introducing a Poisson transformation (Lambert, 1992; Casals, Langohr, Carrasco, Rönnegård, 2015) into the statistical model. This will account for the probability distribution of count data. Since the zero-inflated values represent the "non-learned" items and are meaningful for the interpretation of the data and the modelling, as they are part of the learning curve growth, no further manipulation was on dealing with them was applied in the statistical modelling. Similarly, the component of time was modelled and tested as a polynomial function, for the reasons explained in Chapter 4 (nature of learning growth curves). For this reason, 3 polynomial models (Table 6.1) were used to examine which occurrence- exposure function explains better the learning rate: (a) ~poly(occurrence,1) which refers to a linear function (x1), (b) ~poly(occurrence,2) which refers to a quadratic function (x2), and finally (c) poly(occurrence,3) which refers to a cubic function (x3). The cubic function [poly(occurrence,3)] provides a significantly better fit

(ANOVA tested, see Table 6.1) in all tasks and therefore was used as the basic structural component for the synthesis of more complex models.

Model type	*R model*	*Curve Shape*	*Meaning*
Baseline model	glmer(on_target_pre_total ~ 1 + (1|subno) + (1|seq:subno), data, family = poisson())	Linear shape	Learning isn't related to exposure
Shape model 1	glmer(on_target_pre_total ~ poly(occurrence,1) + (1|subno) + (1|seq:subno), data , family = poisson())	Linear shape	Exposure can predict learning in a linear function.
Shape model 2	glmer(on_target_pre_total ~ poly(occurrence,2) + (1|subno) + (1|seq:subno), data , family = poisson())	Quadratic shape	Exposure can predict learning in a quadratic function.
Shape model 3	glmer(on_target_pre_total ~ poly(occurrence,3) + (1|subno) + (1|seq:subno), data , family = poisson())	Cubic shape	Exposure can predict learning in cubic function.

Table 6.1 This table demonstrates the baseline statistical model and the testing curve shape models. The cubic shape model was significantly a better fit to the data for all tasks, and therefore was chosen to be the baseline model for the rest predictor factors in the next research questions.

Model Comparison	$\chi 2$	P-value
Baseline Model vs Shape Model 1	564589	<.001
Shape Model 1 vs Shape Model 2	73576	<.001
Shape Model 2 vs Shape Model 3	23359	<.001

Table 6.2 This table shows the ANOVA results from the statistical comparison of shape models in Table 6.1.

The above model comparison revealed that shape model 3 was the best fit to the data and therefore will be used as the baseline learning model to answer the following research questions. From now on learning will be estimated as cubic function of occurrence.

C.2. Hypothesis modelling testing- Is there a hierarchical structure in the SL process in our data? Does position of an item, affect its learning outcome?

To answer this research question, we created a baseline and 4 hypothesis models, and then the model with the best fit (ANOVA comparison) was selected as the model describing better the data.

Model type	R model	Meaning
Baseline model	glmer(on_target_pre_total ~ poly(occurrence,3) + (1|subno) + (1|sequence:subno), data, family = poisson())	Exposure can predict learning in a cubic function.
Hypothesis model 1	glmer(on_target_pre_total ~ poly(occurrence,3) + positions + (1|subno) + (1|sequence:subno), data, family = poisson())	Exposure can predict learning in a cubic function & the item position differentiates the learning outcome.
Hypothesis model 2	glmer(on_target_pre_total ~ poly(occurrence,3) * positions + (1|subno) + (1|sequence:subno), data, family = poisson())	Exposure can predict learning in a cubic function & the item position differentiates the learning outcome and that changes during exposure.
Hypothesis model 3	glmer(on_target_pre_total ~ poly(occurrence,3) * positions + type + (1|subno) + (1|sequence:subno), data, family = poisson())	Exposure can predict learning in a cubic function, the item position differentiates the learning outcome and that changes during exposure, and the type of task (mixed length or non-mixed length) can affect the learning outcome.
Hypothesis model 4	glmer(on_target_pre_total ~ poly(occurrence,3) * positions * type + (1|subno) + (1|sequence:subno), data, family = poisson())	Exposure can predict learning in a cubic function, the item position differentiates the learning outcome and that changes during exposure, and the type of task (mixed length or non-mixed length) interacts with the item position over time, affecting the learning outcome.

Table 6.3 This table demonstrates the baseline learning model and the testing hypothesis models.

Model Comparison	$\chi 2$	P-value
Baseline Model vs Hypothesis Model 1	115186	<.001
Hypothesis Model 1 vs Hypothesis Model 2	6237.7	<.001
Hypothesis Model 2 vs Hypothesis Model 3	1.8651	0.172
Hypothesis Model 2 vs Hypothesis Model 4	12100	<.001

Table 6.4. This table shows the ANOVA results from the statistical comparison of shape models in Table 6.3.

The above results suggest that there are position effects in our data, that are describing the learning process and they change over exposure. Additionally, they suggest that the learning rate is different across the type of tasks and that can be reflected on positions. For example, the scores for the 4th position are greater in the 2&4 dots task than they are in the 3&4 dots task and similarly the scores for the 4th position are greater in the 3&4 dots task than they are in the 4 dots task. A table with the specific residuals can be found in Appendix E, Table E.1. However, within each task, there is a persistent pattern with the last item of the sequences being learned better than the previous and so on.

From the results of this analysis, we can see a hierarchical structure in the learning process with the learning performance increasing as the position of the item increases in the sequence. In other words, learning rate of the 1st association < learning rate of the 2nd association< learning rate of the nth association (see Figure 6.1, Figure 6.2). There is a clear positioning pattern of the learning process, that can be used as evidence to reject the all or none hypothesis of learning and partially support evidence for a hierarchical structure in sequential SL. That effect is persistent across tasks and sequence lengths indicating that it is not associated with the properties of the sequence but rather the mechanism itself.

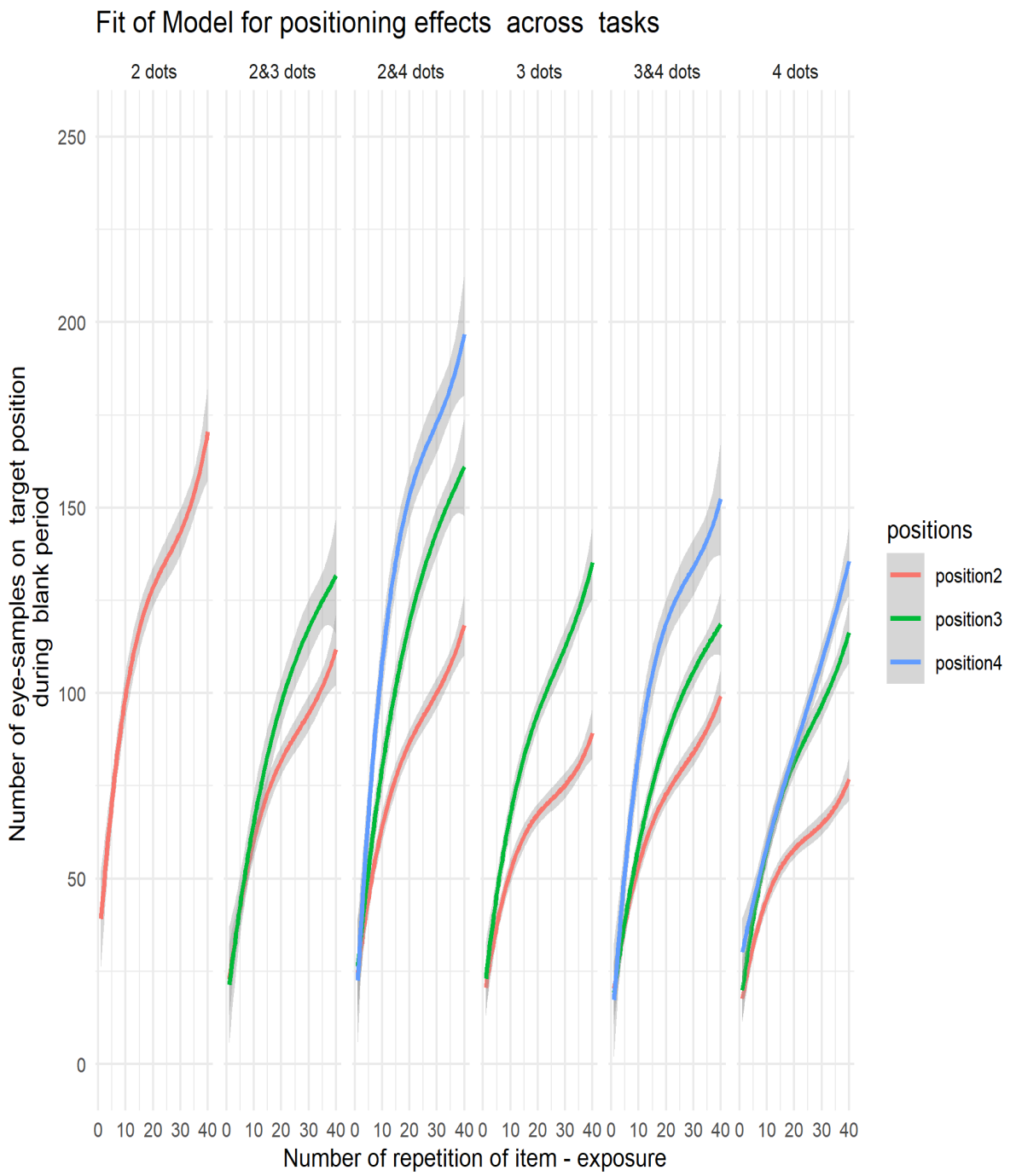


Figure 6.1. This figure shows the fitted values of the hypothesis model 2, for the 6 tasks (2 dots,2&3 dots, 2&4 dots, 3dots, 3&4 dots, 4 dots). On the x-axis is the number of repetitions of item or in other words exposure to that specific sequence (how many times that stimulus has been shown) and on the y-axis is the mean number of eye samples on the target during the blank period (750ms). The colour of the line represents the learning rate for each individual item position within the task. Error bars represent 95% CIs.

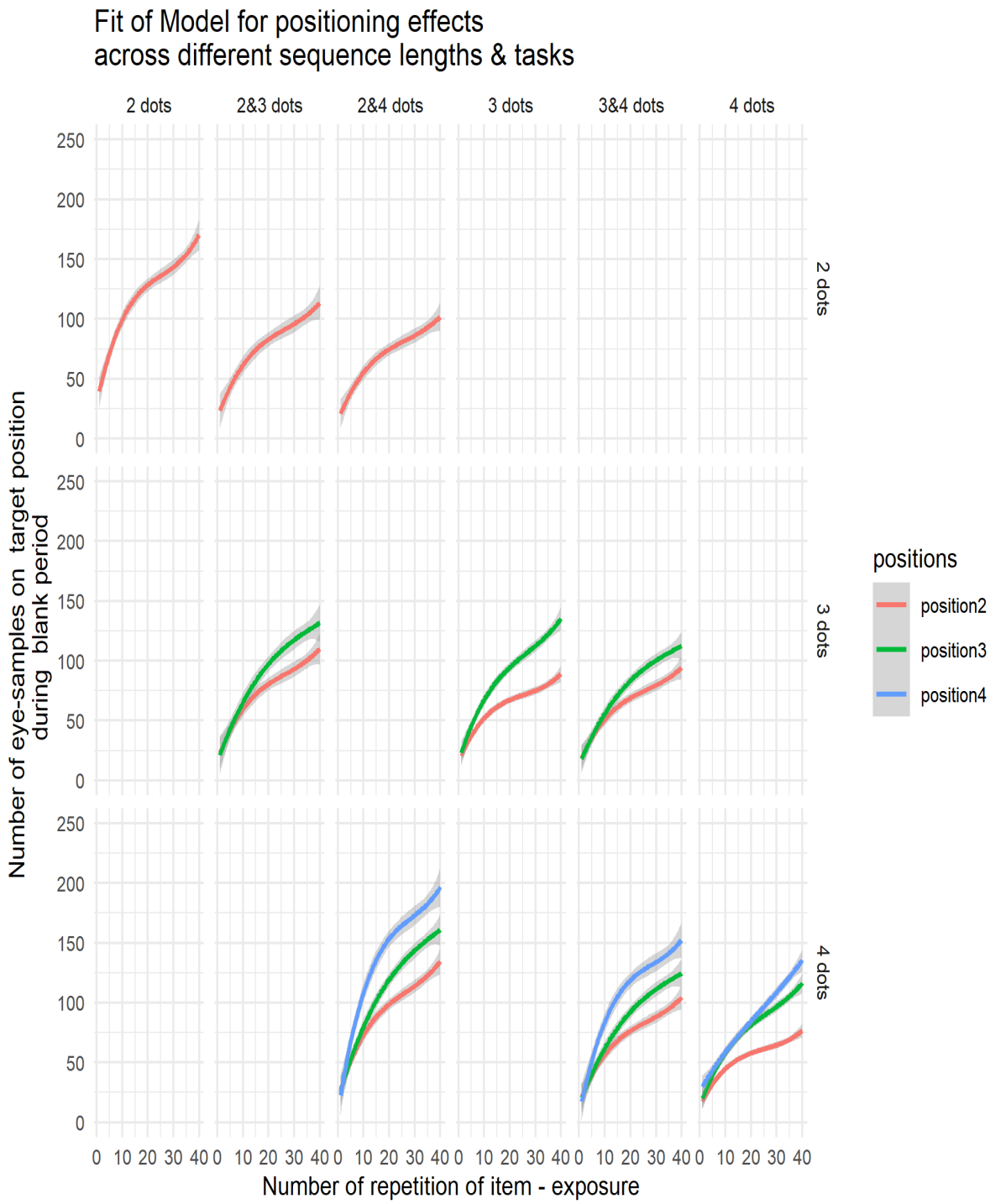


Figure 6.2. This figure shows the fitted values of the hypothesis model 2, for the 6 tasks (2 dots,2&3 dots, 2&4 dots, 3dots, 3&4 dots, 4 dots) and each individual sequence length within task (2 dots, 3 dots, 4dots). On the x-axis is the number of repetitions of item or in other words exposure to that specific sequence (how many times that stimulus has been shown) and on the y-axis is the mean number of eye samples on the target during the blank period (750ms). The colour of the line represents the learning rate for each individual item position within the task. Error bars represent 95% CIs.

D. Discussion

The positioning effects that were found in this chapter reject the "all or none learning" nature of SL as they suggest clear event related relationship between the associations of the sequence formed. There is a consistent pattern in the positioning effect, with the first association of the sequence being learned more slowly and the final item more quickly. Since that effect is consistent and persistent across sequence lengths and types of tasks (mixed length/ same length), it suggests that it is not related to the properties of the stimuli or the sequence itself, but rather on the actual SL mechanism. Therefore, it suggests a hierarchical nature of SL that operates hierarchically on the temporal domain, rather than the actual structural sequential domain, supporting an encoding process that aligns with the principles of the models suggested by Atkinson and Shiffrin (1968) and Norman (1970).

The finding about the speed of learning for each of the different positions can reveal how the encoding of information occurs. For example, in the sequence paradigm of A, B, C, D items, if learning CD occurs before the BC, and the BC before the AB, then we are dealing with a different way of chunking information. Furthermore, if instead of pairs we observe triplets in the sequences of 3 dots but pairs in the sequences of 4 dots, that implies that there are 2 distinctive mechanisms for chunking information. Additionally, we could have same pair and triplet effects only presented in mixed length tasks suggesting that the SL mechanism is being affected by sequential properties of the sequences that are mixed with the sequence A, B, C, D (e.g E, F, G-length of 3 items sequence). Finally, it could be the case, that there isn't a case of faster learned but better learned, that could be explained as an effect of procedural facilitation within a sequence.

At this stage, we need to acknowledge that the numeric values that demonstrate learning across all tasks are low (100-200 eye-samples on target) in comparison to the maximum possible

recording value of 750 eye samples on target. However, that doesn't reflect poor learning in total, but individual differences in learning, since these results are aggregated data of both participants that learned and didn't learn the task. Individual differences are expected to be part of the observed mechanism and therefore exclusion of data from non- learners was not an option. Additionally, it is important to understand that this sequential SL process that we recorded occurred with specific time limitations per trial, that can be limiting for the time course of eye-movements. Even though we allowed a 750ms window for learning to occur during the blank window period of guessing, if we include the eye-movements planning and the anticipation of multiple target locations (16 in total) (Wu & Kowler, 2013), that reduces the actual reaction time substantially (e.g., if eye movement planning and anticipation of location take 500ms, only 250 ms is left, equating to 250 eye-samples on target per total, for learning that develops during a trial). However, once the sequence is learned, the planning can occur immediately once the 1st item of a sequence occurs, and the gaze can be directed immediately on the position of the next items in the sequence as a reflection of a learned association of S-R with a maximum of 750 eye samples on target location.

A direct implication of those findings could be applied in educational paradigms. Many researchers use the paradigm of SL to improve vocabulary growth (Stokes, Kern, Dos Santos, 2011; Stokes, 2010) in infants and reading ability in children (Arciuli, 2018; Elleman, Steacy, Compton, 2018). By selecting to position specific learning targets in SL task-games we could improve the results of the current applications used and facilitate both teachers and the children and their families. Furthermore, the findings can be applied in special education intervention programs or in improving current teaching methods of reading in schools for both typical learners and those ones with learning disabilities (Arcuili, 2018; Gabay, Thiessen & Holta, 2015). Currently it is known that developmental dyslexia is associated with impaired statistical learning ability (Gabay, Thiessen, Holta, 2015) but also dysgrapgia (McCloskey & Rapp, 2017)

However, there is evidence (Arcuili, 2018; Protopapas et al., 2017) that the SL principles can be used as a supplementary method to the explicit (rule-based) method with positive results on children's reading ability and orthographic learning.

The main limitation of the current design is the fact that it focuses on the encoding of the information and not the generalisation part. So it might be the case that while we are building up the unit of knowledge during the SL task we form the units in an X way, but when we retrieve that information in order to be able to generalise it for the creation of new units we use a different mechanism Y. Future research, could be easily adapted to the current paradigm to examine whether the mechanism X as identified in the encoding phase is the same that applies for the generalisation phase too. If these mechanisms differ, that suggests that SL operates with hierarchical structures within the mechanism.

To conclude, the findings above suggest a temporal hierarchical encoding process of sequential SL on the visual domain. These hierarchical effects are persistent across sequence properties and tasks properties suggesting that they reflect actual part of the SL mechanism rather than procedural or perceptual effects.

Effects of sequence length and sequence mixture type on SL rate on the visual domain.

Chapter Summary

In this chapter we are demonstrating the application of the methodology presented in Chapter 5 to investigate the effect of sequence length and the effect of same-length sequence sets vs. mixed-length sequence sets on the rate of sequence learning (SL). Sequences of 2, 3 and 4 items were used to test whether shorter sequences facilitate greater learning than longer sequences. Additionally, we predicted that if the length of the sequences is mixed in a task, the tasks that contain shorter length sequences will be learned better than for longer length sequences, since the amount of information that will have to be processed will be smaller. Finally, a difference in performance of sequences with specific lengths (2, 3 or 4) will be observed between mixed length tasks and same length tasks, hypothesising that longer sequences are better learned when they are mixed with shorter sequences rather than when they are presented in a same length task, again due to the amount of information contained in each task. As stated in the literature review in Chapter 2, sequence length is an issue broadly investigated in language learning in particular but also visual SL cognitive psychology in general. The application of the new experimental paradigm suggested in Chapter 5, aims to clarify how sequential SL occurs in the visual domain, and how the length of the sequences impacts on the sequential SL rate on mixed length and non-mixed length tasks.

A. Introduction

Sequence length has been used by various researchers in the field as one of the factors that affect the SL process, in processes such as language learning, machine learning and deep learning, and neuronal memory processes. Part of the literature suggests that SL can be affected by sequence length and others suggest that SL has unlimited capacity and therefore is not affected by sequence length. Furthermore, naturalistic language involves mixed lengths of words, but infants have shown inability to segment mixed lengths words by the age of 8 months-old. Therefore, we decided to test the methodology developed in Chapter 5, to examine how sequence length affects the SL rate on a sequential SL task in the visual domain. Additionally, we decided to further investigate whether the learning rate of mixed-length sequences differs from the learning rate of non-mixed length sequences and the potential sequence length effects of those differences.

Heimbauer, Conway, Christiansen, Beran and Owren (2018) investigated the impact of sequence length and grammar complexity on artificial grammar learning (AGL) on nonhuman primates (i.e. macaques). It is well established that humans and nonhuman primates use implicit sequential learning and pattern extraction abilities to categorise and organise environmental stimuli (Conway, Christiansen & Morten, 2001). Heimbauer et al. (2018) showed that over the training period, macaques responded faster to sequences that were generated from the artificial grammar in comparison to random sequences, validating their ability to perform a sequential SL task. Additionally, they were able to generalise the learned pattern to novel sequences of the same grammar. Furthermore, macaques were found to learn and generalise the grammar for sequences with up to eight items.

Milne, Petkov and Wilson (2018) further investigated sequential learning in humans and monkeys in the visual and auditory domain, using an AGL paradigm. Both groups

demonstrated sensitivity to item order in both modalities. Interestingly, both species demonstrated similar response patterns to the visual and auditory sequences, suggesting that SL is an evolutionary mechanism that passed on to human species from primates through evolution, but also that it operates comparably across different modalities. However, this sequence length is a feature of explicit sequential learning, and it is already known that longer sequences are harder to be remembered than shorter ones (Ebbinghaus, 1964/1885).

Jacoby, Woloshyn, and Kelly (1989) suggested that explicit memory has limited capacity while implicit memory has no such limitations. Therefore Howard (Howard & Howard, 1992; Howard, Mutter, & Howard, 1992) investigated whether sequence length limits performance in implicit tasks. In her studies she reported that short sequences are learned better than longer sequences in implicit learning tasks. Pascual-Leone et al. (1993) agreed with Howard that longer sequences are more difficult to learn and therefore decrease the probability of procedural learning. He suggested a comparison between an 8-items sequence and a 12-items sequence paradigm, by saying "*In the case of the 8-item sequence, the subject has to retain in the memory buffer at least the previous 8 asterisk positions in order to identify the pattern. In the case of the 12-item sequence the minimal required storage is 12 items, which in some individuals may exceed their declarative short-term memory. An essential demand on the memory buffer is the appropriate temporal indexing of the occurrence of asterisk positions so that the sequence can be stored and retrieved as a sequence*" (Pascual-Leone et al., 1993, p. 600). Pascual-Leone et al. (1993) concludes that longer sequences are harder to learn and introduced the idea of temporal indexing of the occurrence of the sequence items.

Stadler and Neely (1997), inspired by the research of Pascual-Leone et al. (1993) and Howard et al. (1992) investigated sequence length effects on implicit serial learning tasks. They proposed that the structure of the sequence might be more important than the sequence length for the implicit learning process. Sequential structure can be coded and described in many ways

(e.g., Attneave, 1959; Vitz & Todd, 1969). Stadler and Neely (1997) used redundancy as their information metric and sequence structure component: in particular, they used a paradigm from the sequence learning paradigm of Nissen and Bullemer (1987). “In Nissen and Bullemer's (1987) sequence of DBCACBDCBA, the level can be one trial (D, B, C, etc.), pairs of trials (DB, BC, CA, etc.), triplets (DBC, BCA, CAC, etc.), and so on. If every possible run of a given level occurs in the sequence with equal frequency, redundancy is zero” (Stadler and Neely, 1997, p.15). They attribute the learning outcome to an interaction of learning mechanism and short-term memory, rather than reducing length effects entirely to a consequence of short-term memory capacity limitations.

Spiegel and McLaren (2006) performed a series of studies in which they compared the human performance in Serial Reaction Time (SRT) tasks with the predictions that the simple recurrent network (SRN) model of associative sequence learning (by Elman ,1990) produced. The researchers observed that the predictions of the SRN were similar to the human performance, and therefore suggesting that sequence learning occurs in an associative manner and not in a rule-based manner. Additionally, they noticed that previous models of associative sequence learning such as simple associative chaining models failed to match the performance of the SRN models. This could be due to the fact that simple associative chain models do not rely on the extraction of statistical regularities of the sequences. Therefore, they concluded that human sequence learning occurs in an associative manner and relies on the extraction of statistical regularities from the sequences.

In language learning, sequential learning has been used in the format of syllables to study word learning, usually with sequences of 2, 3 or 4 items (syllables) to resemble to language as close as possible. Saffran, Aslin and Newport (1996) investigated SL and language acquisition mechanisms on 8 months old infants by using transitional probabilities (TPs). Word boundaries were determined after exposure in the stream of sounds on the basis of transitional probabilities

that differed internally and externally of a word. They found successful segmentation for artificial words with 3 syllables and artificial words with 3 syllables after mere exposure to a continuous stream of sounds. The conclusion is that word segmentation, a basic process involved in language acquisition, is successfully accomplished by 8 months old infants. Word segmentation is achieved by using statistical relationships between neighbouring sounds.

Witteloostuijn, Lammertink, Boersma, Wijnen and Rispens (2019) investigated sequential SL in the visual domain, with sequences of 3 items. They used a 2AFC and 3 AFC paradigm to examine SL performance on the visual domain, on early-school-aged children. Their aim was to introduce the concept of RT to explain part of the variability in individual differences in SL. The visual statistical learning (VSL) task contained triplets. During the familiarization phase participants were presented with an alien character on the screen and had to give a button response to proceed to the next alien. During the testing phase, participants, had either to choose the triplet that they have seen before, or complete a missing stimulus of the triplet. Half of the participants performed a cover task, while the other half did not. The researchers successfully measured the online sensitivity to the statistical structure by comparing the RTs for the predictable vs the unpredictable aliens, and the results suggested that RTs were significantly shorter for the predictable than the unpredictable elements. That suggests that early school aged children are sensitive to TPs during exposure and that RTs are a good measure to assess that observation.

Frank, Goldwater, Griffiths and Tenenbaum (2010), decided to explore word segmentation and sentence length via a sequential SL task. More specifically they examined SL of word segmentation while controlling for sentence length, exposure and number of word types. Even though the behavioural data showed a clear effect of those components on SL performance, suggesting that longer sequence length, less exposure (less frequent stimuli) and more language

(greater diversity) make language learning harder, their computational proposals failed to replicate those findings.

Thiessen (2017) approached the memory limitations suggested by previous researchers in a different way. His computational framework suggests that statistical learning arises from a set of processes, that are present in the mechanism of memory and consist of activation, interference, integration of information and forgetting (Perruchet & Vinter, 1998; Thiessen et al, 2013). According to this approach statistical learning does not involve explicit computation of statistics and TPs but is the result of fundamental memory processes.

Johnson and Tyler (2010) used TPs to examine how infants perform in mixed length streams of sequences. They highlight that the paradigms of artificial languages being used in past methodologies (Saffran et al., 1996; Heimbauer et al., 2018) to examine language learning have been using same length of sequences and therefore they significantly differ from naturalistic languages. They compared the SL ability between two age group of infants (5,5 months-old vs 8months old) of segmenting word boundaries, by exposing the two groups in streams of artificial language that contained four same length words (all CVCV) or four mixed length words (two CVCV, two CVCVCV), with equal TPs across word types. Their findings suggested that both infant groups performed equally well in the same length words task, and successfully identified the word boundaries, however both groups failed to demonstrate learning for the mixed length words.

It is well established that infants use a sequential SL mechanism to learn language, and they are particularly good at it, while adults seem to have memory capacity limits when dealing with longer sequences. As temporal indexing seems to be crucial for the sequential SL process as shown in Chapter 5, using mixed length of sequences in the same task, could affect the learning rate of mixed length sequences since the mechanism of unit learning (pairs vs triplets) might

facilitate a task of 2& 4 dots (using pairs for both) or impede a task of 3&4 or 2&3 dots task (if it uses pairs for the sequences of 2 and 4 dots and triplets for the sequences of 3 dots).

The aim of the current chapter is to use the experimental data from Chapter 5, in order to examine sequential SL effects on adults the visual domain, by comparing their performance in sequences of 2, 3 and 4 items, in mixed and non-mixed sequence length tasks. We predicted that shorter sequences will be learned better than longer ones, and that longer sequences will be learned better when they are presented with shorter ones in mixed sequence length task, rather than when they are presented in same length tasks.

B. Methods

Same as stated in Chapter 5 (section B) & Chapter 6.

C. Results

C.1. Data Visualisation

Before conducting any statistical analysis to the data, we were interested in to exploring the learning curves of the data with locally estimated smoothing functions. To do so, we created with the usage of R Studio (R Core Team, 2013) a descriptive plot that fitted a locally estimated smooth linear regression (method loess y ~ x) to our data to get a first impression about the learning rate for each task (Figure 7.1). All plotting and analysis code for this chapter can be found in Appendix I. Trials that consisted of eye samples to the first item of a new sequence (transition from the final position of the current sequence to the first position of the next sequence) were removed within task, as they were considered exogenous to the sequence learning process and in some way random (1/3 chances to guess correctly if learning within all sequences is successful due to limited locations (16) and 1/16 chances to guess correctly if

learning within all sequences is unsuccessful). Learning was calculated as the averaged eye samples (of all 36 participants) on the target location across positions for each task. Specific details about the rationale of data extraction/selection for each research question differs and will be given prior the statistical modelling of that research question.

This indicates that learning was rated as the average of eye samples on the specific target location across positions. For 2 dots task, we averaged the eye samples that were in the correct location for the second positioned item of the sequence (position 2), for the 3 dots task, we averaged the eye samples that were in the correct location for the second and third positioned items of the sequences (position 2 and position 3) and similarly we did for the 4 dots task. For the mixed length sequences task (2&4, 3&4, 2&3) we had 2 different approaches depending on our research question.

When we looked at learning within task, and how well participants learned a mixed length task, we averaged the eye samples that were in the correct location for each position, for each sequence length type. For example, in the 2&3 dots we averaged (a) the eye samples in the target location for the second positioned item of the sequence for the sequences with 2 dots sequences length and (b) the eye samples in the target location for the second and third positioned item of the sequences for the sequences with 3 dots sequence length. For the same research question performed our analysis similarly for the rest of the mixed length tasks (2&4, 3&4).

However, when we were interested in how positioning of the item in a mixed length sequence is affecting the learning, we only kept the common positions across the different sequence lengths tasks.

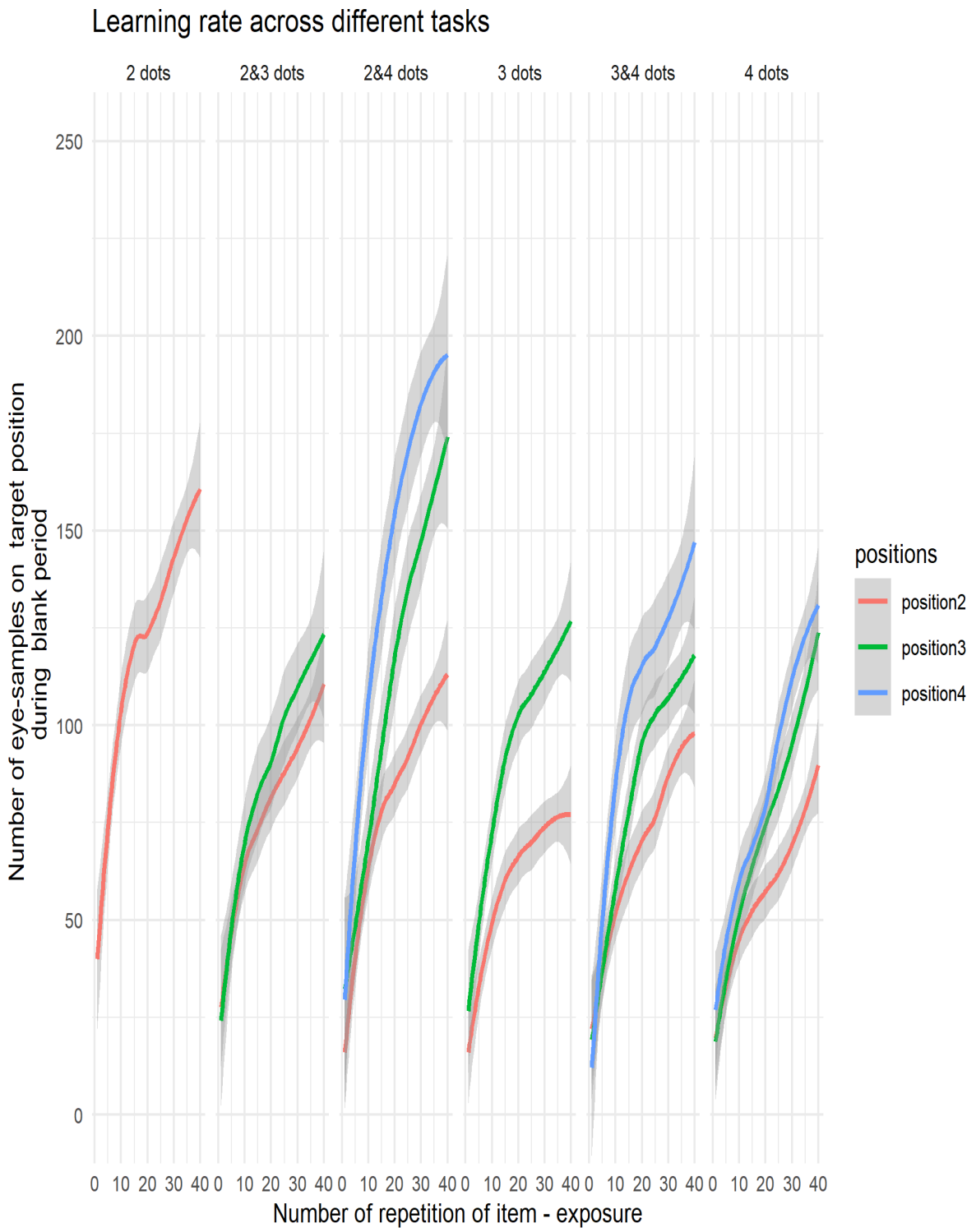


Figure 7.1 Demonstrates the learning rate for all the participants, across the 6 different tasks (2 dots, 3 dots, 4 dots, 2&3 dots, 2&4 dots, 3&4 dots), for each item position within the sequence. On the x-axis is the number of repetitions of an item or in other words exposure to that specific sequence (how many times that stimulus has been shown) and on the y- axis is the mean number of eye samples on the target location during the blank period (750ms). Since our eye-tracker is recording at 1000Hz, the maximum value that the y-axis can take is 750 samples.

The different line colours represent the position of the item within the sequence; for 2 dots, only position 2 can be shown (i.e., learning the next dot in a 2-dot sequence) whereas for four dots, three can be shown (learning the positions of the second, third, and fourth dots). Error bars represent 95% CIs.

C.2. Statistical Modelling

All data analysis was conducted in the statistical programming environment *R* (R Core Team, 2013). Data were modelled using Generalised Linear Mixed Effects Models (GLMMs) (McCullagh & Nelder, 1989; Skrondal & Rabe-Hesketh, 2004) using the R-package lme4 Bates, Meachler, Bolker, & Walker, 2015) and R-package tidyverse (Wickham et al., 2019). The code for the analysis of this chapter can be found in Appendix I. Predictors were included for sequence length, position in sequence, occurrence- exposure, type of sequence (mixed length, non-mixed) and interactions of occurrence, sequence length and type of sequence, provide the ability to predict the relationship of each independent variable with the expected outcome (learning rate) but also understand sources of random variability in the outcome. The fixed effects that are being examined in this chapter is the sequence length (how many dots had the sequence) and the type of task (mixed length task or same length task). The random effects in our data derive from the variable of sequences (4 sequences each task, A, B, C, D) that it is clustered within participants (36 in total).

A baseline GLMM (Table 7.1) was created to define the best baseline model for the data, before accounting for the variability derived from sequence length effects. The main component that affected the structure of the GLMMs was the structure of function of occurrence- exposure, suggesting that as exposure increases, the eye-samples increase and therefore learning occurs. Exposure occurs in time and many studies have shown that learning during time is not linear but rather quadratic or polynomial. The psychological factors that can affect the non-linear nature of learning during time, can be tiredness or general natural learning limitations (Miller, 1956; Mastorakis, 2018; Endress & Szabó, 2017). The general natural learning limitations

suggest that once the individual reaches his individual learning threshold, they can't exceed the learning rate and the curve begins to flatten at its pick point. That threshold varies across individuals and is usually being used as a psychometric tool for cognitive scientists (Gold, Law, Connolly & Bennur, 2010). Since the SL process in the current design requires processes such as retrieval of the location and saccade planning, a maximum of 750 ms during the learning phase is hard to occur, unless learning has been automated. As Ghahghaei and Preeti Verghese (2015) suggest, efficient saccade planning requires time, and in this paradigm the time is being deducted from the actual guessing time (the 750 ms blank period). The data contained a large number of zero eye samples on the target, but zero values represented no learning in the tasks, therefore were considered as a meaningful part of the learning growth curves and were not removed or manipulated by some function. In total 4 different models were created to select the shape of the learning curve. A cubic function of occurrence provided the best fit (Table 7.2) to our modelling data and therefore was chosen as the baseline model for the rest of the analysis in this chapter. The models used in the comparison of fits can be found in Table 7.1 and were representing 3 different hypotheses for the curve shape (linear, quadratic, cubic).

Model type	R model	Curve Shape
Baseline model	glmer(on_target_pre_total ~ 1 + (1|subno) + (1|seq:subno), data, family = poisson())	Linear shape
Shape model 1	glmer(on_target_pre_total ~ poly(occurrence,1) + (1|subno) + (1|seq:subno), data , family = poisson())	Linear shape
Shape model 2	glmer(on_target_pre_total ~ poly(occurrence,2) + (1|subno) + (1|seq:subno), data , family = poisson())	Quadratic shape
Shape model 3	glmer(on_target_pre_total ~ poly(occurrence,3) + (1|subno) + (1|seq:subno), data , family = poisson())	Cubic shape

Table 7.1 This table demonstrates the baseline statistical model and the testing curve shape models. The cubic shape model was a significantly better fit to the data for all tasks, and therefore was chosen to be the baseline model for the remaining research questions.

Model Comparison	$\chi 2$	P-value
Baseline Model vs Shape Model 1	564589	<.001
Shape Model 1 vs Shape Model 2	73576	<.001
Shape Model 2 vs Shape Model 3	23359	<.001

Table 7.2 This table shows the ANOVA results from the statistical comparison of shape models in Table 7.1.

C.3 Is sequence length affecting the learning rate on same length tasks? Are shorter sequences learned faster and better than longer ones?

In order to answer this question, we extracted all the "meaningful" eye-samples on target positions for all the same length tasks (2 dots, 3dots, 4dots) and run the models as shown in Table 7.3. The "meaningful" eye-samples on target positions, are the eye-samples on the target positions that the participants were expected to learn the locations for. Therefore, all eye-samples on the first item of a sequence, in every task were excluded, as they were not indicators of learning but random guessing (since any sequence is followed by one of three other sequences), while the within sequence positions were classified as learning data. A cubic polynomial function was fitted.

The baseline model predicted that learning was a cubic function of occurrence, suggesting that learning relies merely on exposure and no other factor. Hypothesis model 1 adds on the baseline model, with sequence length as a predictor of learning in same length sequences, on an intercept level. Hypothesis model 2 adds on baseline model the sequence length as an interaction predictor variable (on a slope level). The model fit comparison can be found in Table 7.4. Hypothesis model 2 was proven to be a better fit to the data.

Model type	R model	Interpretation
Baseline model	glmer(on_target_pre_total ~ poly(occurrence,3) + (1|subno) + (1|sequence:subno), data, family = poisson())	Learning can be explained as a cubic function of occurrence (exposure)
Hypothesis model 1	glmer(on_target_pre_total ~ poly(occurrence,3)+ sequence_length + (1|subno) + (1|sequence:subno), data, family = poisson())	Learning can be explained as a cubic function of occurrence and sequence length (on an intercept level)
Hypothesis model 2	glmer(on_target_pre_total ~ poly(occurrence,3)* sequence_length + (1|subno) + (1|sequence:subno), data, family = poisson())	Learning can be explained as cubic function of occurrence and interaction sequence length (on a slope level)

Table 7.3. This table shows the models that were used in R to examine whether sequence length has an impact on learning rate in same length sequence tasks. An ANOVA was performed between the hypothesis and baseline model, that demonstrated that the hypothesis model is explaining the data significantly better than the baseline model.

Model Comparison	$\chi 2$	P-value
Baseline Model vs Hypothesis Model 1	1.7854	0.409
Baseline Model vs Hypothesis Model 2	8591.5	<.001

Table 7.4. Shows model fits comparison of models stated in Table 7.3

Figure 7.2 and Figure 7.3 show better learning for the 2 dots compared to the 3 dots task and the 4 dots task, confirming the hypothesis that shorter sequences are learned both faster and more robustly compared to the slope of curves in different tasks in Figure 7.2).

The odds ratios of Hypothesis model 2 can be found in Appendix E, Table E.2.The model suggests that 2 dots sequences are learned significantly better than 3 dots and 4 dots sequences,

however 3 dots sequences and 4 dots sequence perform at similar scores their curves differ in shape.

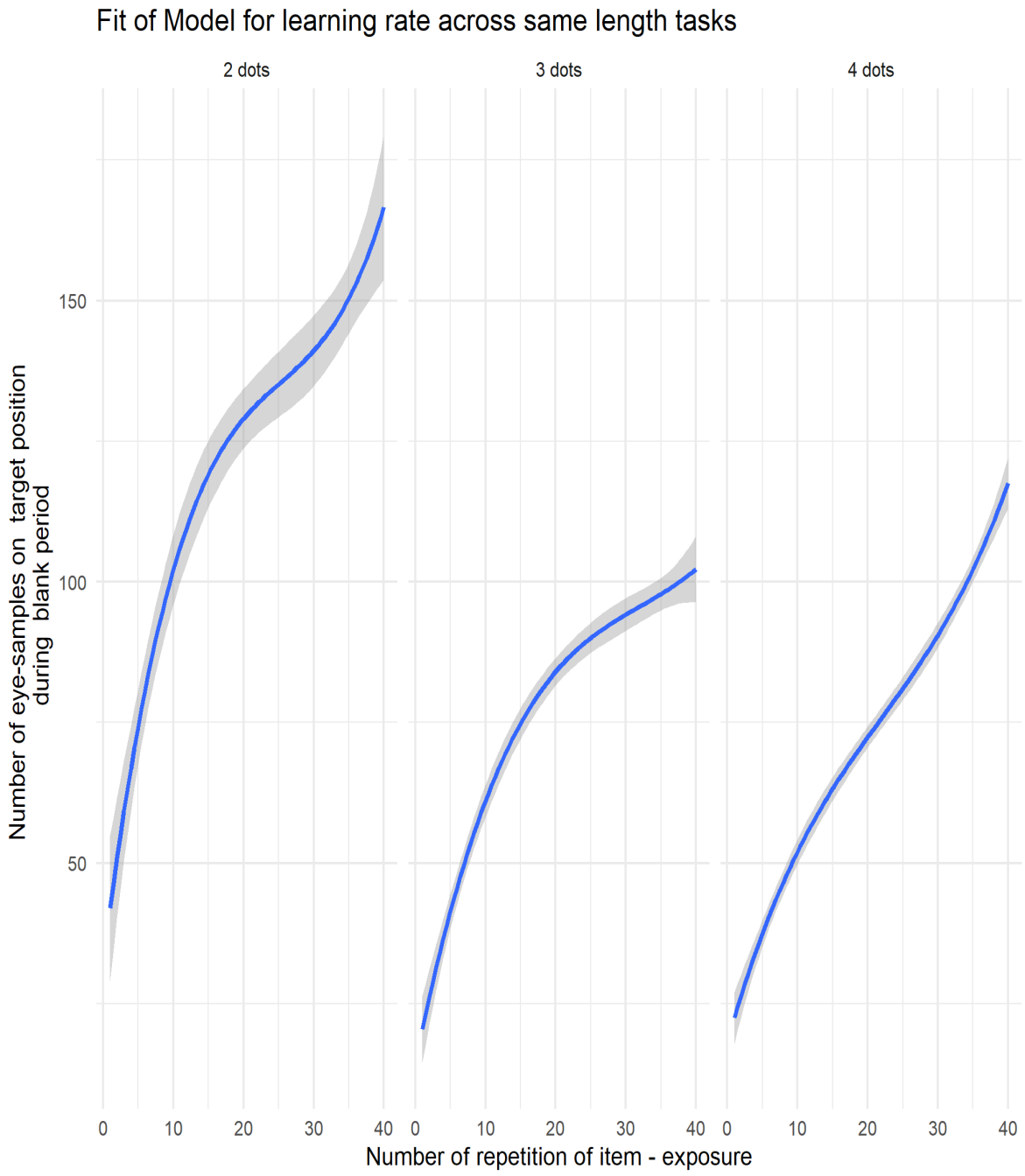


Figure 7.2. This figure shows the fitted values of the hypothesis model 2 (line) for the 3 same length tasks (2 dots, 3 dots, 4 dots). On the x-axis is the number of repetitions of item or in other words exposure to that specific sequence (how many times that stimulus has been shown) and on the y- axis is the mean number of eye samples on the target –location area during the blank period of (750ms). Error bars represent 95% CIs.

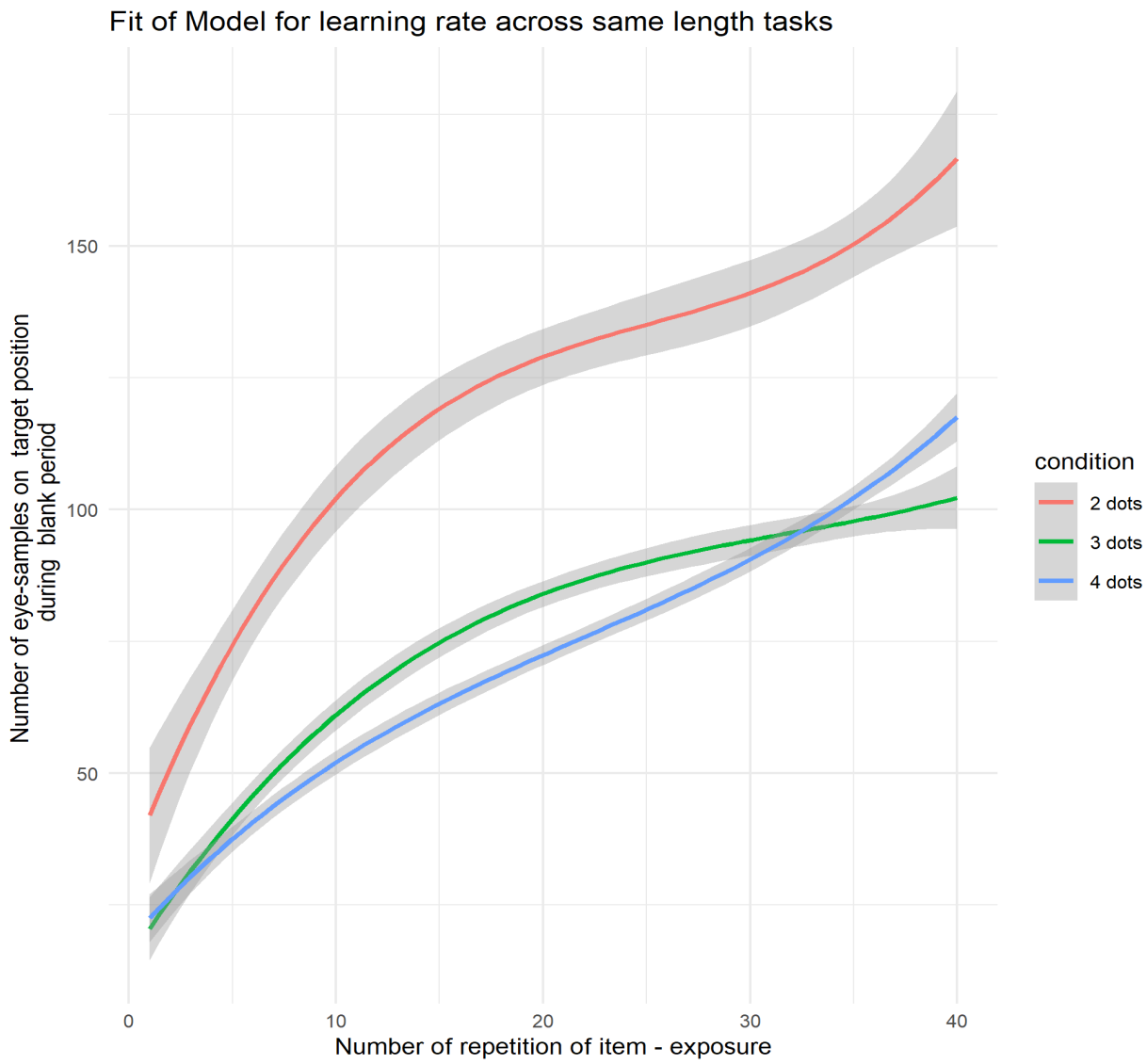


Figure 7.3. This figure shows the fitted values of the hypothesis model(line) for the 3 same length tasks (2 dots, 3 dots, 4 dots) on the same graph. On the x-axis is the number of repetitions of item or in other words exposure to that specific sequence (how many times that stimulus has been shown) and on the y- axis is the mean number of eye samples on the target during the blank period (750ms). Error bars represent 95% CIs.

C.4. Is sequential learning achievable in mixed length tasks? If so, how does sequence length affect performance in mixed length tasks?

In order to answer this question, we extracted all the "meaningful" eye-samples on target positions for all the mixed length tasks (2&3 dots, 2&4dots, 3&4dots) and run the models as shown in Table 7.5. The "meaningful" eye-samples on target positions, are the eye-samples on the target positions that the participants were expected to learn the locations for. Therefore, all eye-samples on the first item of a sequence, in every task were excluded, as they were not

indicators of learning but pseudo-random guessing, while the within sequence positions were classified as learning data. A cubic polynomial function was fitted. The baseline model predicted that learning was a cubic function of occurrence, suggesting that learning relies merely on exposure and no other factor. Hypothesis model 1 adds on the baseline model, the sequence length as a predictor of learning in same length sequences, on an intercept level. Hypothesis model 2 adds on baseline model the sequence length as an interaction predictor variable (on a slope level). Hypothesis model 2 was found to be a significantly better fit to the data (Table 7.6). Figure 7.4 shows the fit of the model plotted for each task for each sequence length, while Figure 7.5 shows the total learning across sequence lengths within each mixed length task.

Model type	R model	Interpretation
Baseline model	glmer(on_target_pre_total ~ poly(occurrence,3) + (1|subno) + (1|sequence:subno), data, family = poisson())	Learning can be explained as a cubic function of occurrence (exposure)
Hypothesis model 1	glmer(on_target_pre_total ~ poly(occurrence,3)+ sequence_length + (1|subno) + (1|sequence:subno), data, family = poisson())	Learning can be explained as a cubic function of occurrence and sequence length (on an intercept level)
Hypothesis model 2	glmer(on_target_pre_total ~ poly(occurrence,3)* sequence_length + (1|subno) + (1|sequence:subno), data, family = poisson())	Learning can be explained as cubic function of occurrence and interaction sequence length (on a slope level)

Table 7.5. This table shows the models that were used in R to examine whether sequence length has an impact on learning rate in same length sequence tasks. An ANOVA was performed between the hypothesis and baseline model, that demonstrated that the hypothesis model is explaining the data significantly better than the baseline model.

Model Comparison	χ2	P-value
Baseline Model vs Hypothesis Model 1	20.099	<.001
Hypothesis Model 1 vs Hypothesis Model 2	2981.4	<.001

Table 7.6. This table shows the models fits scores of the models tested in Table 7.5.

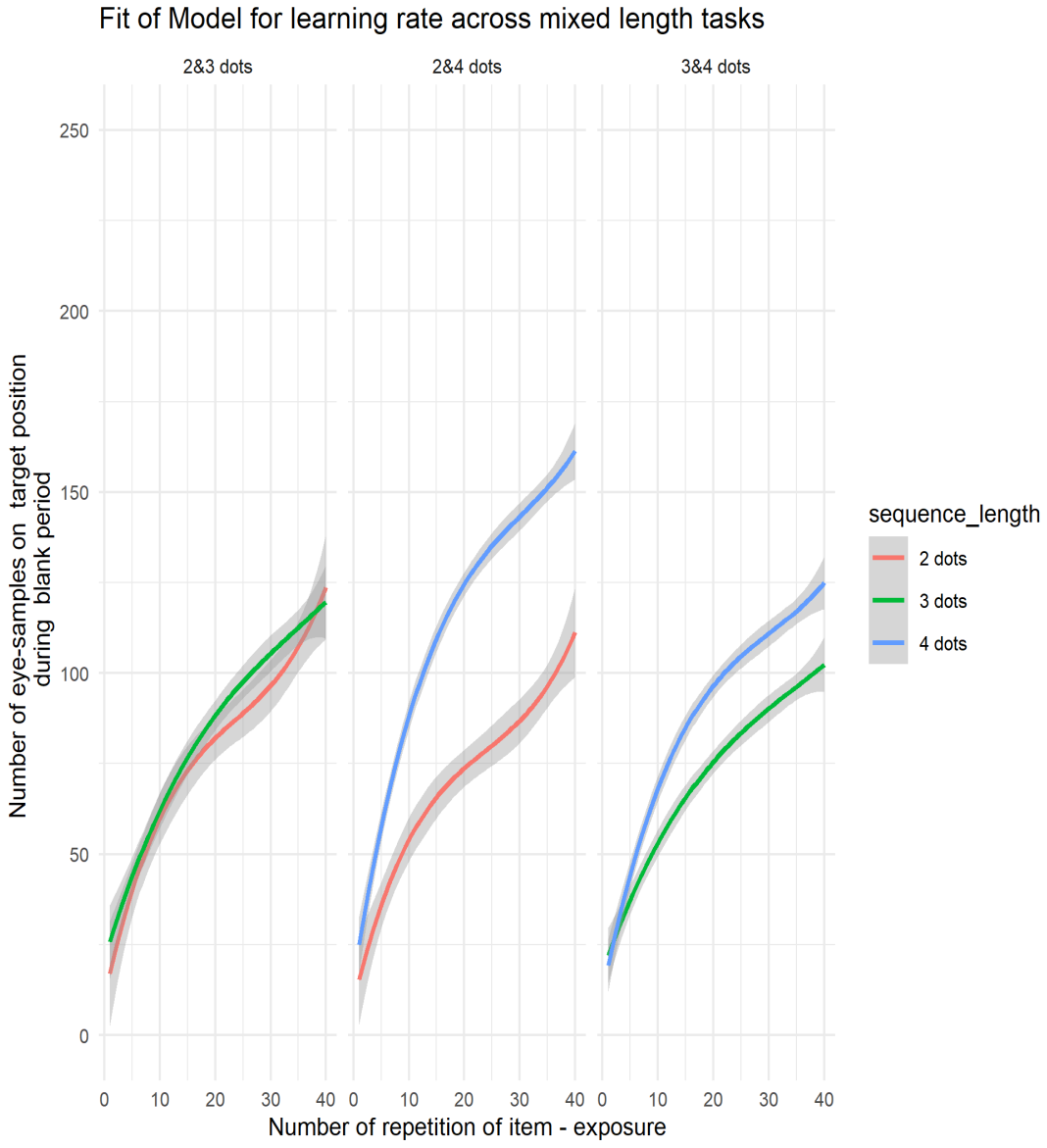


Figure 7.4. This figure shows the fitted values of the hypothesis model 2(line) for the 3 mixed length tasks (2&3 dots, 2&4 dots, 3&4 dots). On the x-axis is the number of repetitions of item or in other words exposure to that specific sequence (how many times that stimulus has been shown) and on the y- axis is the mean number of eye samples on the target during the blank period (750ms). The colour of the line represents the learning rate for different sequence lengths within the task. Error bars represent 95% CIs.

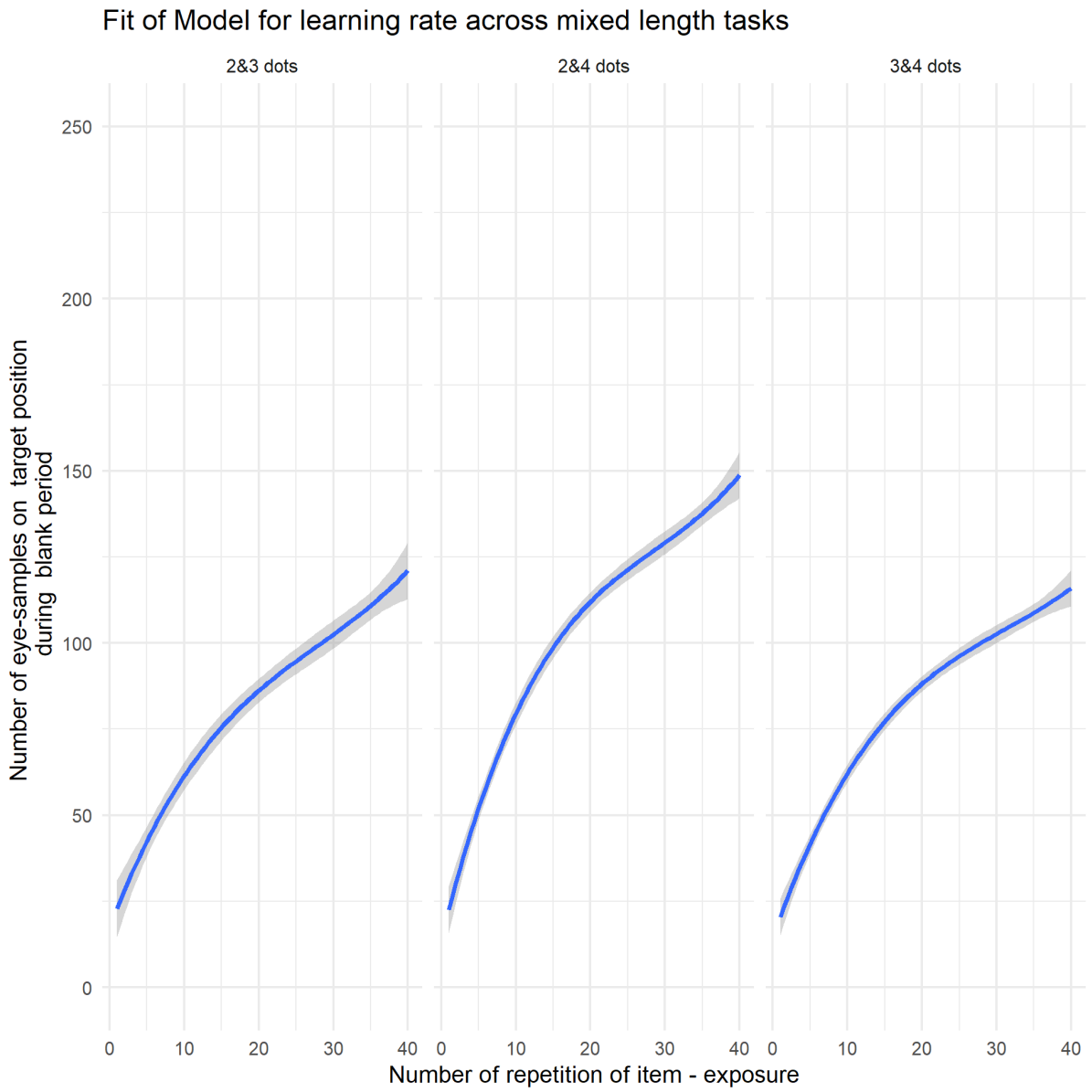


Figure 7.5. This figure shows the fitted values of the hypothesis model 2 (line), for the 3 mixed length tasks (2&3 dots, 2&4 dots, 3&4 dots). On the x-axis is the number of repetitions of item or in other words exposure to that specific sequence (how many times that stimulus has been shown) and on the y- axis is the mean number of eye samples on the target during the blank period (750ms). Error bars represent 95% CIs.

The analysis results of the models suggest that learning in mixed length tasks can be explained as a cubic function of occurrence and its interaction with sequence length. As we can see in Figure 7.4 and Figure 7.5, the mixed length tasks that contain even numbers of sequence lengths (2&4 dots) performed better than the tasks that contained a mixture of even and odd sequence lengths (2&3 dots, 3&4 dots), suggesting evidence towards a distinctive coding

mechanism between different sequence lengths that operates with both pairs and triplets and therefore causing facilitation on even length mixed sequences due to a shared chunking strategy. The odds ratios of Hypothesis model 2 can be found in Appendix E, Table E.3.

C.5. Is sequence length affected by the type of task (mixed/non-mixed)? If so, how does the type of task affect sequence length learning?

To answer this question, we extracted all the "meaningful" eye-samples on target positions for all tasks (2 dots, 2&3 dots, 2&4 dots, 3 dots, 3&4 dots, 4 dots) and run the models as shown in Table 7.7. The "meaningful" eye-samples on target positions, are basically the eye-samples on the target positions that the participants were expected to learn the locations for. Therefore, all the eye-samples of the first item of a sequence, in every task were excluded, as they were not indicators of learning but pseudo-random guessing, while the within sequence positions were classified as learning data. The baseline model was predicting that learning in these tasks occurs on cubic shape during exposure. The hypothesis fits were then compared through an ANOVA. The ANOVA showed that hypothesis model 4 was a significantly better fit for the data (Table 7.8). The fits of the model can be seen in Figure 7.6, demonstrating that sequence length effects differ between mixed and same length tasks.

Model type	R model	Interpretation
Baseline model	glmer(on_target_pre_total ~ poly(occurrence,3) + (1|subno) + (1|sequence:subno), data, family = poisson())	Learning can be interpreted as a cubic function of occurrence (exposure)
Hypothesis model 1	glmer(on_target_pre_total ~ poly(occurrence,3) + type + (1|subno) + (1|sequence:subno), data=df_learning, family = poisson())	Learning can be predicted as a cubic function of occurrence and type of task (mixed/non-mixed) is a predictor of learning (intercept level).

Hypothesis model 2	glmer(on_target_pre_total ~ poly(occurrence,3) * type + (1|subno) + (1|sequence:subno), data=df_learning, family = poisson())	Learning can be predicted as a cubic function of occurrence and type of task (mixed/non-mixed) is an interaction predictor of learning (slope level).
Hypothesis model 3	glmer(on_target_pre_total ~ poly(occurrence,3) *type + sequence_length + (1|subno) + (1|sequence:subno), data=df_learning, family = poisson())	Learning can be predicted as a cubic function of occurrence, type of task (mixed/non-mixed) is an interaction predictor of learning (slope level) and sequence length is a predictor of learning (intercept level).
Hypothesis model 4	glmer(on_target_pre_total ~ poly(occurrence,3) * type * sequence_length + (1|subno) + (1|sequence:subno), data=df_learning, family = poisson())	Learning can be predicted as a cubic function of occurrence, type of task (mixed/non-mixed) is an interaction predictor of learning (slope level). And sequence length is an interaction predictor of learning (slope level).

Table 7.7. This table shows the models that were used in R to examine whether sequence length has an impact on learning rate in same length sequence tasks. An ANOVA was performed between the hypothesis and baseline model, that demonstrated that the hypothesis model is explaining the data significantly better than the baseline model.

Model Comparison	**χ2**	**P-value**
Baseline Model vs Hypothesis Model 1	1.8396	.175
Baseline Model vs Hypothesis Model 2	1926.3	<.001
Hypothesis Model 2 vs Hypothesis Model 3	19.287	<.001
Hypothesis Model 3 vs Hypothesis Model 4	10177	<.001

Table 7.8. This table shows the models comparison fits for the models used in Table 7.7

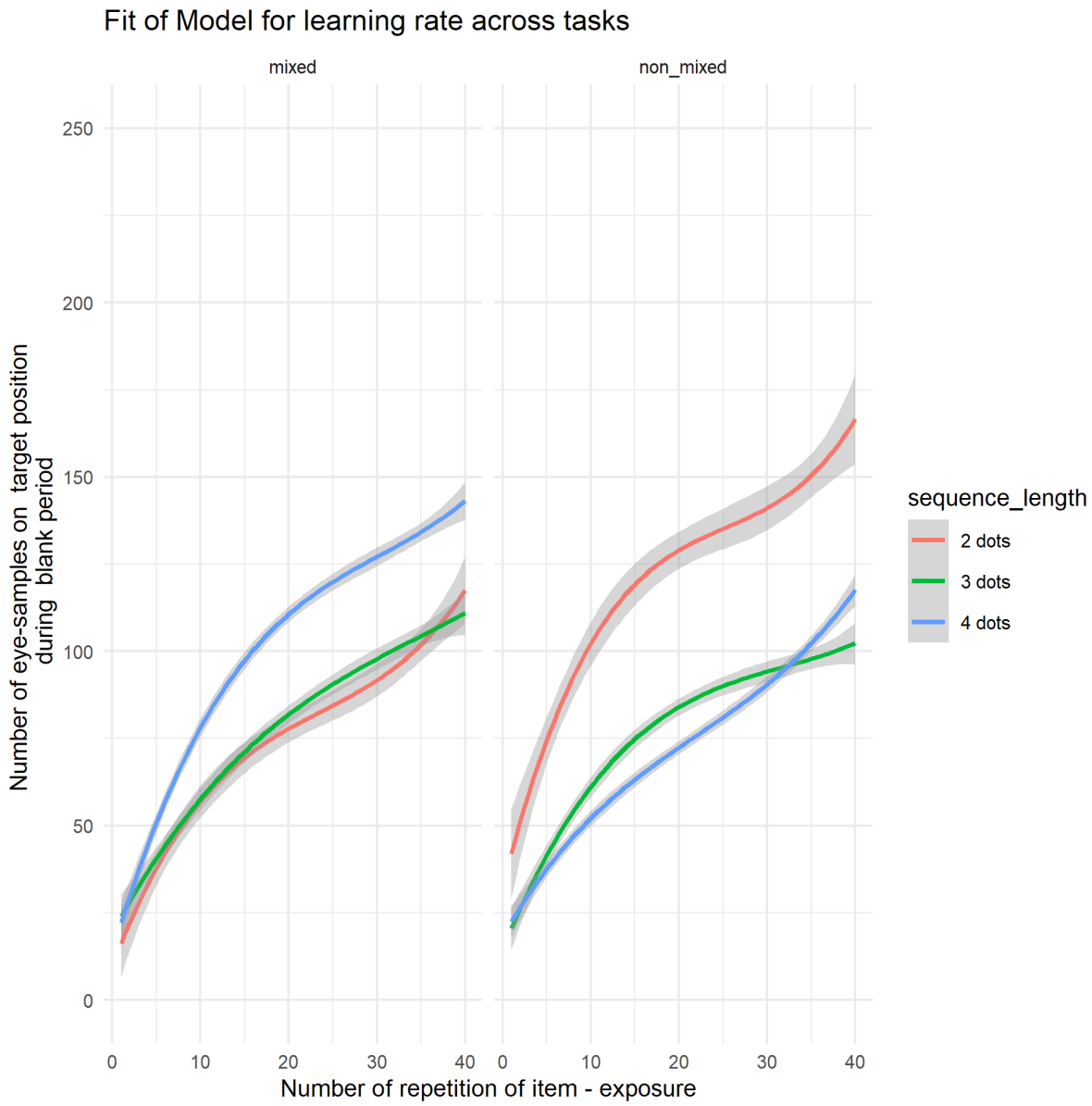


Figure 7.6. This figure shows the fitted values of the hypothesis model 4 (line), for the mixed length tasks and non-mixed tasks. On the x-axis is the number of repetitions of item or in other words exposure to that specific sequence (how many times that stimulus has been shown) and on the y- axis is the mean number of eye samples on the target during the blank period (750ms). Error bars represent 95% CIs.

The results of hypothesis model 4, suggest that sequence length doesn't have a robust effect on the SL mechanism, but in contrast it depends on the type of task and the environment that surrounds a sequence. As we can see in Figure 7.6 shorter sequences of 2 items are learn worse and slower when they are mixed with long length sequences of 4 items rather than when they are presented on their own. Similarly, longer sequences of 4 dot items are better learned within mixed length tasks rather than same length tasks. Finally, sequences of 3 items perform equally

across mixed and non-mixed length tasks. The odds ratios of Hypothesis model 4 can be found in Appendix E, Table E.4.

Discussion

Our results suggest that sequential SL is achievable with the new methodology across same and mixed length tasks and sequences with length of 2, 3 and 4 items. Our findings confirm partially the findings that previous literature suggest, with shorter sequences (2 items) being learned faster than longer ones (3 items, 4 items) in same length tasks. Additionally, we found that when longer sequences of 4 items are mixed with shorter sequences of 2 items, the learning of the longer sequences improves. But it is important to mention that the learning of 2 dots items drops significantly when they are combined with long sequences (4 items sequences) rather than when they are presented in a same length task. Furthermore, we found that sequence length effects are not consistent across mixed and non-mixed tasks. Only the 3 dots task seems to behave similarly in mixed and non-mixed tasks. 4 dots sequences seem to be learned better in the mixed length tasks, as they were paired with shorter sequences, and 2 dots items were learned better on non-mixed length, as they were paired with shorter sequences. That suggests supportive evidence towards a binary mechanism of chunking that operates with pairs and triplets and therefore facilitates the learning of even length sequences (2&4 dots) task, as they share a common chunking mechanism.

Previous research in the field of cognitive linguistics and language processing has found that chunking and predictability are two factors crucial for the learning of grammatical dependencies. Wang, Zevin and Mintz (2019) found that the learning of non-adjacent dependencies in artificial language can be influenced by factors such as the frequency and the predictability of the target words. The term non-adjacent dependencies is used to describe the

grammatical relationships between words in a sentence that are not immediately adjacent to each other. According to their findings, participants demonstrated better learning when the dependencies were more frequent and when the target words were more predictable based on the preceding context. This assumes that learners have the ability to extract and generalise non- adjacent dependencies from a continuous stream and this ability is heavily influenced by the statistical properties of the steam.

Furthermore, Wang, Zevin, Trueswell and Mintz (2020) investigated the chunking mechanism in language processing and how it affects the learning of adjacent dependencies. In this instance, the term chunking represents the process of grouping words into phrases. The researchers claimed that the process of chunking could influence the way we perceive and learn those adjacent dependencies, and therefore words that are grouped together in a coherent chunk are easier to learn and be recalled and as a result it maximises the learning of the dependencies between those words. To test this hypothesis, they used the artificial language learning paradigm in a series of experiments. During each experiment, participants were exposed to sentences containing adjacent dependencies in an artificial language and were asked to learn these patterns. The sentences were manipulated to create different grouping conditions: some sentences had strong cues for grouping the adjacent words together, while others had weak cues or no cues at all. The results of these experiments showed that the adjacent dependencies were better learned when there were strong grouping cues compared to weak or no cues. Therefore, they concluded that chunking occurs in a top-down grouping manner and that this chunking can facilitate the learning of the adjacent dependencies.

Our findings, contradict the findings of Johnson and Tyler (2010), that suggested that mixed length words can't be segmented successfully by 5,5 and 8 months old infants. However, it is important to mention that we examine adult population and that our task was performed on the visual domain rather than the auditory. A possible explanation for the different findings could

be the fact that during infancy, the chunking mechanisms are still developing therefore mixture of lengths impedes the learning, while same length words are easier to be learned. Even in our results, there was successful learning across all mixtures of task, however the learning rate of mixed sequences that contained odd and even number of lengths was lower. Our research findings apply on the visual domain, so the results found can be applied on the sequential SL on the visual domain. But infants learn language mainly from the auditory environmental input or multisensory input (audio-visual mapping and associative learning). Further research using the same methodology should focus on replicating these findings on the auditory domain and expanding the tasks into sequences of 5 and 6 dots lengths to examine how the chunking mechanism operates in those.

It is already known that adults can learn up to 12 item sequences, therefore it would be interesting to replicate the same learning patterns with mixed and not mixed longer sequences. This way the ecological validity of the task could be improved, since it could imitate approximately better processes such as visual SL or word segmentation, and vocabulary growth that is related to language exposure (syllables, words, sentences level). SL is a process that occurs implicitly and is suggested by the literature that has no capacity limitations. By expanding the current methodology in longer sequences, we could imitate real-life learning situations and manage to understand how sequential statistical learning occurs during time.

To conclude, the data show that during a sequential SL task, shorter sequences are learned better than longer ones in same length tasks. In mixed length tasks, sequences that contain even numbers of sequence length are better learned than sequences that contain a mixture of even and odd numbers of sequence lengths. This finding supports the concept of a binary chunking mechanism that operates with pairs and triplets. If it was just pairs, we wouldn't see a difference between even and odd number mixed length sequences (a 3 items sequence would be learned as AB, BC). It is obvious that there is something in the 3 dots task that is affecting the learning

rate. Future research should focus on examining differences in performance between mixed length tasks that contain even and odd number of sequences, in order to understand if the differences observed in the current study are due to the fact that chunking operates only with pairs and therefore learning of triplets impedes the learning process, or whether there are two mechanisms one that works with pairs and one that works with triplets and when combined, learning is slower. The next chapter will evaluate some of the current limitations of this design, such as low numeric values, by introducing the eye-movements component in the evaluation of the design, in order to explain why low numeric values, represent learning in the current set up and how different experimental set ups improve their design by using lower cost and frequency equipment such as Gazepoint GP3 (60 Hz).

A critical evaluation of eye-tracking in the field of experimental psychology & a technical comparison between a high frequency (EyeLink 1000 - 1000 Hz, SR Research Ltd., Mississauga, Canada) and low frequency (Gazepoint GP3- 60 Hz,) eye-tracker based on the thesis experimental paradigm.

Chapter Summary

This chapter will discuss the contribution of eye-tracking in the field of experimental psychology, with main focus on the fields of clinical and cognitive psychology. After evaluating the current applications of eye-tracking in research, a technical comparison between the two eye-tracking systems (Gazepoint GP3 60 Hz vs EyeLink 1000Hz) will be presented based on the applications of both systems on the methodology suggested in the current thesis. Finally, the implications and limitations of each system in applied research will be discussed, by highlighting the reliability of certain equipment specifications such as sampling frequency in research methods, in specific research contexts.

A. The contribution of eye-tracking in the field of experimental psychology.

Eye-tracking is a widely used experimental tool in the field of cognitive psychology (e.g., visual attention, social interaction, body perception, reading, etc.) and clinical psychology (autism spectrum disorder (ASD), ADHD, Parkinson's disease, etc.) across the age spectrum (infants, children, adults, elders). Technically speaking, eye-tracking is the process of measuring eye movements by examining the duration and the location of the eye-movements. By determining facts such as where the individual is looking (x,y coordinates), what is the individual looking at, or how long the individual is looking at something, we can extract meaningful interpretations about the cognitive processes that occur during or prior (planning) to the eye movement period.

The most common eye-tracking movements studied in the field of experimental psychology, are fixations and saccades. A fixation occurs when the visual gaze is fixated on a location with x and y coordinates and usually lasts from 150ms up to 300ms while saccades are the movements prior and after a fixation occurs (duration less than 100ms) and are used to change the location of the fixation. Additionally, Purves, et al. (2001) stated that there are four types of eye-movements: (a) the saccades, that are fast ballistic movements that eyes make when they change their point of fixation and usually last less than 100ms, (b) the smooth pursuit movements, that are slower eye movements that eyes use to track moving objects that usually start 100-150ms after the movements of the moving object, (c) the vergence movements that are aligning the fovea of each eye when the target is located at different distances for the individual (disjunctive movements) and finally (d) the vestibulo-ocular movements that stabilise the eyes on a stimuli accounting for head movements.

Frazier et al. (2016) used eye-tracking as a new tool for creating a novel autism risk index based on eye movements. In the past, abnormal social attentional patterns have been observed

in infants with ASD, when engaging in fixations on other people's eyes (Jones & Klin, 2013) or social scenes (Chawarska, Macari & Shic, 2013), by reporting reduced fixation times. Similar abnormalities in visual social cues have been observed in preschool children (Vivanti, Trembath & Dissanayake, 2014), older children (Magrelli et al., 2013) and adults (Rice, Moriuchi, Jones & Klin, 2012) with ASD. Therefore, Frazier et al. (2016) decided to compare duration of fixations on a series of social attention tasks (static facial affect, dynamic vs. naturalistic scenes, etc.) that have been used previously in the field and compare those durations across non-ASD and ASD children. The researchers used eye-tracking technology to observe and analyse eye movements in infants and young children, specifically focusing on patterns of visual attention. They hypothesized that infants and children with ASD will exhibit distinctive eye movement patterns in comparison to the neurotypical children. The data from the study were analysed with a machine learning algorithm and created the Autism Risk Index (ARI). ARI consisted of various eye movement parameters, such as gaze duration, saccade length, and fixation count, and could distinguish between ASD and non – ASD children with high accuracy.

Navarro, González and Molina (2018) used eye-tracking in students with or without attentional difficulties, across a range of tasks including images, text and videos. They found persistent gaze differences between the two groups when the stimuli presented were images, however when the stimuli were text or videos those differences were only present at the first exposure with the stimuli and then disappeared. Rutledge, Schweitzer, Guyer and Young (2010), aimed to further investigate decision-making in people with ADHD with the use of eye-tracking. They used a utility tone-discrimination task across adults with or without ADHD. During the decision-making task, the participant's eye-movements were recorded. Their findings showed decision durations were faster in individuals without ADHD, while there was a big variability on decision durations across individuals with ADHD. However, gaze durations are not the

only eye measurements that have been used to investigate ADHD. Aya, Toshinobu, Haruhisa, Akira, Shigenobu, Nobumasa (2020), used pupillometry to investigate alertness deficits in ADHD individuals. They hypothesised that the pupil diameter reflects the firing of norepinephrine neurons in the locus coeruleus, so differences in pupil dilation can reflect deficits in the norepinephrine - locus coeruleus neuromodulatory system that is responsible for alertness. They monitored the kinetics of pupil diameter of ADHD and typically developing adults during an auditory continuous performance task. They found that adults with ADHD had larger tonic pupil diameter in comparison with typically developing adults, while they exhibited a suppressed stimulus-evoked phasic pupil dilation. That is, atypical pupil behaviour according to Aya et al. (2020) is reflecting the hyperactive norepinephrine - locus coeruleus system that results in the alertness deficits of ADHD individuals.

Lee (2017) used eye-tracking to understand sentence formulation while speaking in patients with Parkinson's Disease (PD). To do so, Lee (2017) used two competing models for sentence production in PD patients. The first one suggested that PD patients have increased demands during speech production due to high levels of buffering of words, to minimise speech disfluencies while the second one suggests that word planning occurs one word at a time resulting in compromised performance on speech accuracy and fluency. During the task participants had to give a description of a scene with 3 items and formulate a sentence explaining the position of the three items in relation to the others (e.g., "the A and the B are above the C"), while the name codability of the objects varied. Gaze durations were recorded for each one of the objects, during sentence production. PD patients demonstrated word by word planning, providing supporting evidence for the second sentence production model while objects with low codability contained more disfluencies and word-finding errors than high codability objects.

Eye-tracking has been used in the field of clinical psycholinguistics to examine not only speech production as stated above, but also as a screening tool for processes like reading in individuals with dyslexia. Benfatto, Seimyr, Ygge, Pansell, Rydberg, Agneta et al. (2016), highlighted that an early intervention is important for dyslexia, and therefore identifying high risk individuals from an early age is needed. They recorded the eye-movements of high risk and low risk children with reading difficulties at the age of 9-10 years while they were reading. This new risk assessment as suggested by Benfatto et al. (2016) is not using the typical screening methods for dyslexia that involve oral and written tests but using eye-tracking that allows the recording of the reading process in real-time. Finally, they concluded that even though dyslexia is mainly a language-based learning disability, creating assessments that do not measure straight verbal responses, like eye-tracking, can be an efficient way to identify high risk dyslexic children with reading difficulties at early stages.

Another research area that eye-tracking has been frequently used is body perception and body dysmorphic disorder (BDD). According to Greenberg, Reuman, Hartmann, Kasarskis and Wilhelm (2014), individuals with BDD have negative attention bias and as a result they are overfocusing mainly on negative attributes. Similarly, Toh, Castle and Rossell (2015), investigated facial affect recognition between BBD and Obsessive- Compulsive Disorder (OCD) individuals with the use of eye-tracking. During the experimental procedure, participants were viewing pictures of Facial Affect (Ekman & Friesen, 1975), while their eye-movements were being recorded. The study findings suggest that BDD participants exhibited fewer fixations of shorter durations to facial areas such as the eyes, nose and mouth, while participants with severe BDD/OCD demonstrated lower accuracies, fewer fixations and greater saccade amplitudes.

Liang, Tsai and Hsu (2017) used eye-tracking to measure sustained visual attention in individuals with social anxiety (SA), by assessing the time course of attentional processing. In

their study they recorded the eye-movements of individuals with and without SA during a multiple emotional stimuli free-viewing paradigm. During the task a stream of angry, sad, happy and neutral faces was presented for 10s per trial. Individuals with SA showed stronger engagement with the thread at the early stages of the processing, while it was harder for them to disengage their attention from the thread once they had fixated on it and paid less attention to positive stimuli at a later stage than the individuals without SA.

However, apart from purely clinical applications of eye-tracking in the field of psychology, eye-tracking has been one of the main experimental tools for investigating learning processes and environments. Jamet (2014) used eye-tracking in order to examine the effects of visual cues on multimedia learning in typical populations. In this study, individuals were exposed to multimedia computer learning environments that either contained or not visual cues and their eye-movements were recorded. The visual cue was the change of colour of an item, when the item was verbally evoked by the environment. Individuals that were exposed to visual cues spent less time fixating on irrelevant areas in the multimedia environment and allowed more time for the synchronization of auditory and visual processing leading to better learning.

Koc-Januchta, Höffler, Prechtl, and Leutner (2017), used eye-tracking as a way to measure cognitive learning style in college students. The students were split into two groups: the visualizers and the verbalizers, based on some questionnaires that the students answered about their visual or verbal cognitive style and during the task they were asked to learn about two different topics, by means of verbal (text) and visual (image) combinations, while their eye-movements were being recorded. The results show that visualisers spent significantly more time on the images than verbalizers, and similarly verbalizers spent more time processing the text format of the information. Both types of learners learn actively from the information contained in their type of source information, supporting their cognitive learning style, but visualisers performed better in comprehending the information given rather than verbalisers.

After exhibiting some examples of the research conducted in the field of psychology, that use eye-tracking as a tool of understanding better attentional processing and learning processes in clinical and non-clinical populations, it is clear that eye-tracking requires precision and accuracy. During the development of the new experimental paradigm that this thesis suggested as a new way of examining sequential SL in Chapters 4 and 5, 2 different eye- trackers have been used (Gazepoint GP3 and EyeLink 1000). These two eye-tracking systems have basic technical differences, but both of them can be used to examine concepts like the ones explained above (reading, attentional processing, visual cues etc) with specific set ups and requirements, to avoid implementing type I and type II errors in the findings that would derive from an overpowered or underpowered design. In the next section, the limitations of the new sequential SL experimental paradigm for each of the two eye-trackers will be explained and evaluated through a technical and numeric comparison of the learning outcomes in the data. Gazepoint GP3 is a relatively new eye-tracker, not as popular as EyeLink 1000, with very little evaluation of the system by experimental research. This chapter is adding to the current knowledge by evaluating the performance of the two eye-tracking systems on the same task.

B. A technical comparison of two eye-tracking systems based on the thesis experimental paradigm: Gazepoint GP3 vs EyeLink 1000.

As it was stated in the previous chapters EyeLink 1000 (SR Research Ltd., Mississauga, Canada) is a more powerful tool, that samples in higher frequency, with greater precision and accuracy than Gazepoint GP3 (60 Hz). In Table 7.1 are contained some of the basic technical differences between the two eye-tracking systems. However, precision and accuracy at that high level is not necessarily always the requirement for every design. Gazepoint GP3 (60 Hz) is a low-cost eye-tracker that can be easily used in non-laboratory environments such as

schools, nurseries, hospitals etc. That fact increases the research value of the specific equipment, as it allows easy data collection, across different age groups, locations and set ups.

	Gazepoint GP3	EyeLink 1000- SR Reserarch
Frequency	60Hz	250 Hz, 500Hz, 1000Hz, 2000 Hz
Recording Type	Binocular	Monocular, Binocular
Calibration Points	5-point, 9-point	3-point, 5-point, 9-point
Portable	Yes	No
System Requirements	Intel Core i5 – 8th generation or faster, 8 GB RAM, Windows 7, 8.1 or 10. Mac and Linux are not supported.	External Host PC, Windows (2000, XP), Mac or Linux.
Glasses Compatibility	Yes	Yes
Set ups	Laptop Mount, Monitor Mount, mini tripod	Tower Mount, Primate Mount, Arm Mount
Depth movement	±15 cm range of depth movement	±10 mm depth/ 20 cm (Arm Mount -Remote)
Horizontal/Vertical Movement	25cm (horizontal) x 11cm (vertical) movement	±25 mm horizontal or vertical or 22cm (horizontal) x 18cm (vertical) (Arm Mount-Remote)
Visual Angle accuracy	0.5° –1 ° of visual angle accuracy	down to 0.15° (0.25° to 0.5° typical) 0.5° (Arm Mount- Remote)

Table 7.1 This table demonstrates the technical specifications of Gazepoint GP3and EyeLink 1000.

As we have already seen in Chapter 4 and in Chapter 5, the new experimental eye-tracking method that we suggest uses eye-tracking on different modes (time displayed and gaze contingent) and assesses learning as the raw eye-samples on the target location during the guessing “blank period” of 750ms. During that time period a Gazepoint GP3 can maximally record 45 eye-samples while EyeLink 1000 can record 750 eye- samples. At first, it seems an obvious assumption to say that EyeLink 1000 is a better option for the design, however that happens because we evaluate the comparison based on numeric values and not as a proportion of percentage learned. It is a priori defacto that the numeric values of eye-samples on target of

Gazepoint GP3 will be lower than the ones of EyeLink 1000 due to the huge difference in the sampling rate. A fair evaluation of both systems in the current design, would calculate learning as a proportion of eye-samples on target, divided by the number of maximum eye-samples that each eye-tracker can record during the 750ms "blank period" of guessing. In order, to evaluate the efficiency of the eye-trackers Figure 7.1 has been created and demonstrates the percentage of learning rates for the 2 dots task from 2 different experiments, one run with EyeLink 1000 and one with Gazepoint GP3, that was calculated as explained above and fitted as a smooth cubic function of occurrence.

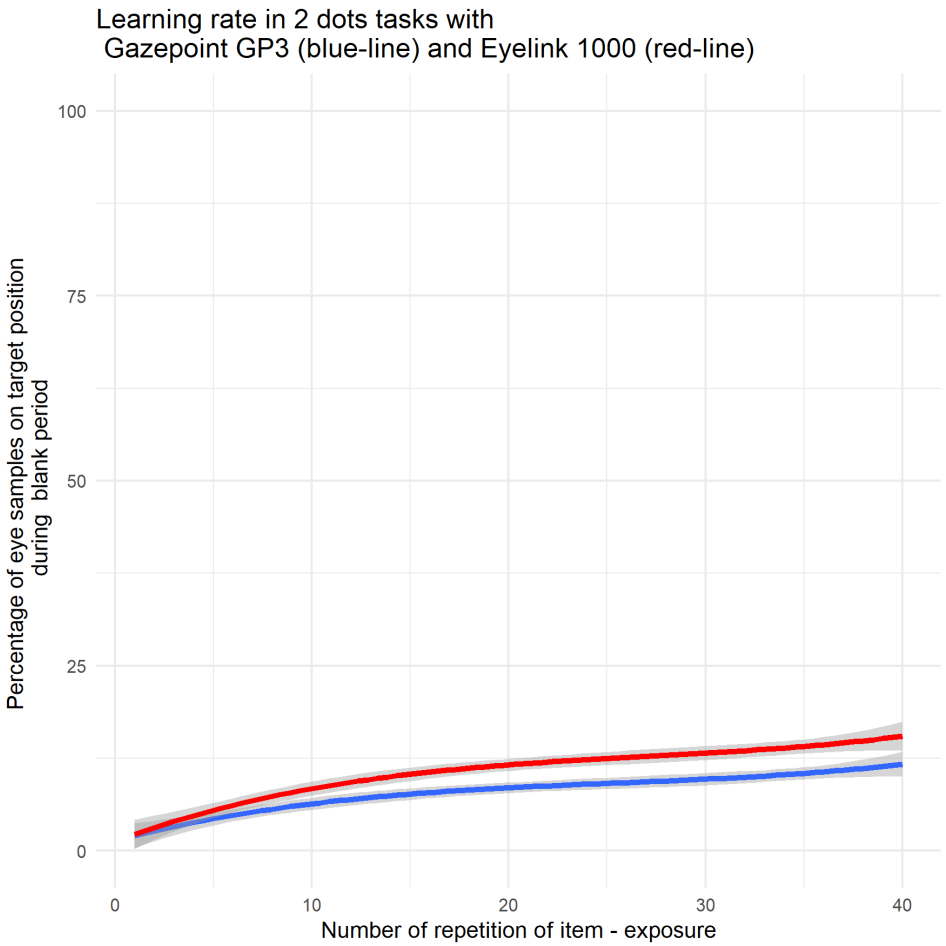


Figure 7.1. This figure shows the percentage learning scores for each eye-tracker. The linear fit of the curves has been estimated as a smoothed cubic function of occurrence over the observed percentage scores of learning in the Gazepoint GP3 and EyeLink 1000 experiment for the 2 dots task. Error bars represent 95% CIs.

As we can see from Figure 7.2, when the numeric values are transformed into percentages, the gap of performance between Gazepoint GP3 and EyeLink 1000 reduces significantly. At this point, it is also important to mention that the experiment conducted on the EyeLink 1000 had a methodological addition of feedback sound, that could explain the difference in performance across the tasks. Arguably, the percentage scores of learning remain low, for both eye-trackers, but if we take into consideration the eye-movement planning that occurs during the 750ms "blank period" of guessing, and the fact that each item of the sequence is presented only 40 times in total in the task, the performance can be argued to be not low at all. Additionally, both scores are results of aggregated scores across participants that learned and participants that didn't learn. There was no point to exclude the non-learners from the tasks as they evidence individual differences in learning patterns, and since the suggested methodology is new, it requires to be shown at its full extent in order to be fairly evaluated.

Spatial Acuity comparison of GP3 and Eye-Link 1000

Process of Spatial Acuity Analysis for GP3 and Eye-Link 1000

Selection of eye-tracking data (definition of stimuli & eye-tracking movements): For this analysis, I used the 2 dots task data that was collected from Design A (GP3) and Design B (Eye-Link). As it is common practice in this type of analysis I selected fixations and their durations instead of raw samples. More specifically, I will be using the fixations and their characteristics on target locations, after the target has appeared. I did not want to choose the blank period of guessing as observations about less fixations or smaller durations during this period could be due to poorer learning rather than actual differences between the two eye-trackers. By using the fixations on the targets, after the targets have appeared on screen, we guarantee that what we measure reflects the acuity differences between the eye-trackers and

not necessarily differences in learning or participant's skills. Another element that needs to be highlighted is the fact that the two data sets come from different participants and have some small design differences (Design B had a beep sound as negative feedback, while Design A didn't). Therefore, the data might be biased. Trials that had no fixations on target location were removed from the data.

Focus of analysis: number of fixations on target locations, duration of fixations on target locations, accuracy of fixations on target locations (distance from target).

Analysis

Before conducting any sample comparisons, we looked at the descriptive statistics for the number of fixations per trial for each eye-tracker (Table 7.2) and the descriptive statistics for the duration and distance from target location for each fixation, for each eye-tracker (Table7.3)

	M	**SD**	**Number of trials that had fixations (N)**
Number of Fixations per trial (GP3)	2.38	1.17	2992
Number of Fixations per trial (Eye-Link 1000)	1.11	0.39	4524

Table 7.2. This table shows the descriptive statistics for the number of fixations on target location during the 2 dots task, across all 36 participants, for the two different eye-trackers.

	M	**SD**	**Number of total fixations across 11484 trials (N)**
Duration of fixation (GP3)	0.313 (ms)	0.16	7108

Distance from target (GP3)	453.59 (pixels)	190.28	7108
Duration of fixation (Eye-Link 1000)	0.263(ms)	0.20	5011
Distance from target (Eye-Link 1000)	401.82 (pixels)	166.93	5011

Table 7.3. This table shows the descriptive statistics for the duration and the distance from target for each fixation on target location during the 2 dots task, across all 36 participants, for the two different eye-trackers.

An independent-samples t-test was conducted to compare the number of fixations per trial in the GP3 eye-tracker and Eye-Link 1000 eye-tracker. There was a significant difference in the amount of fixations for GP3 (M=2.38, SD=1.17) and Eye-Link 1000 (M=1.11, SD=0.39); $t(7514)=67.66$, $p < .001$. This suggests that GP3 had significantly more fixations per trial on target locations than Eye-Link 1000.

An independent- samples t-test was conducted to compare the duration of fixations in the GP3 eye-tracker and Eye-Link 1000 eye-tracker (as shown in Figure 7.2). There was a significant difference in fixation duration for GP3 (M=0.31, SD=0.16) and Eye-Link 100 (M=0.26, SD=0.20); $t(12117)= 15.330$, $p < .001$. This suggests that GP3 had significantly longer

fixations on target than Eye-Link 1000.

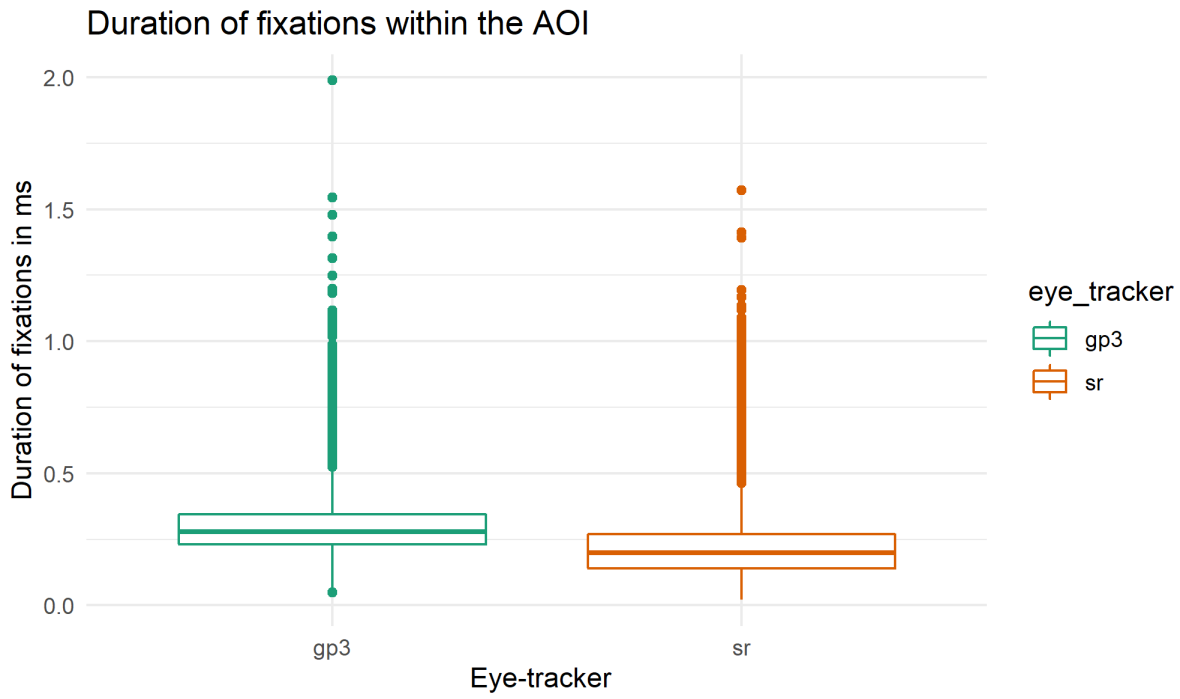


Figure 7.2. This boxplot shows the scores for duration for each fixation on target location during the 2 dots task, across all 36 participants, for the two different eye-trackers.

An independent- samples t-test was conducted to compare the distance from target on the fixations that were within the AOI for the GP3 eye-tracker and Eye-Link 1000 eye-tracker (as shown in Figure 7.3). There was a significant difference in the distance from target for GP3 (M=453.59, SD=190.28) and Eye-Link 1000 (M=401.82, SD=166.93); t(12117)= 15.51, $p < .001$. This suggests that Eye-Link 1000 was significantly closer to the target location.

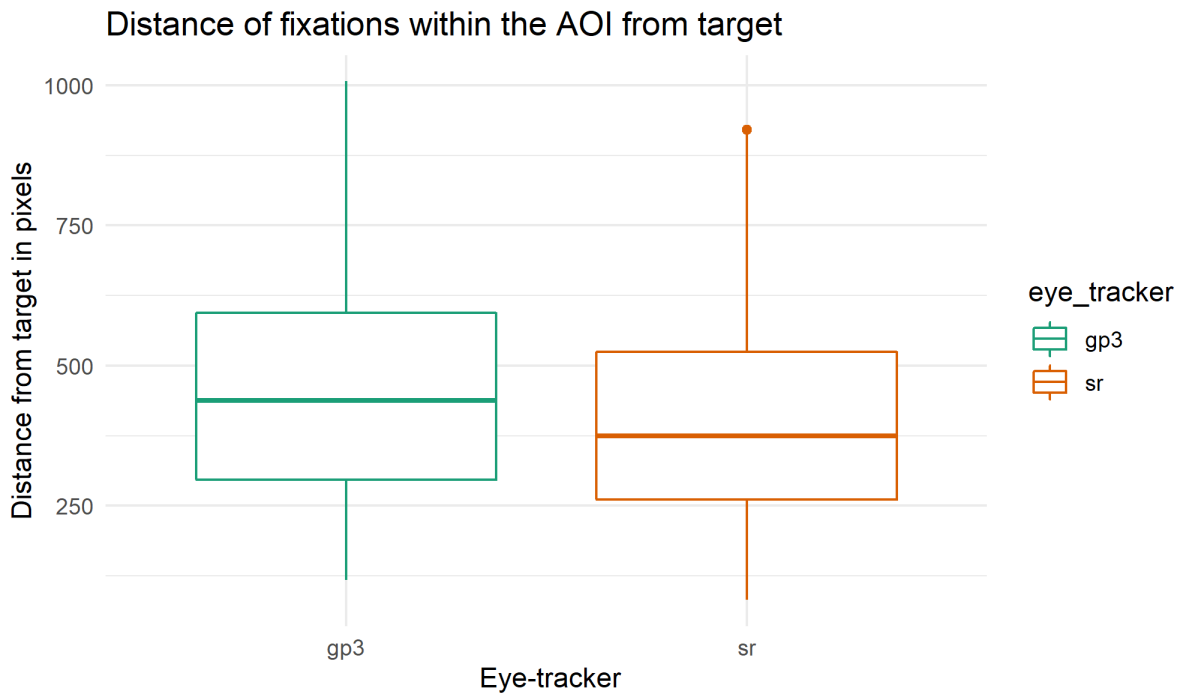


Figure 7.3. This boxplot shows the scores for the distance from target for each fixation on target location during the 2 dots task, across all 36 participants, for the two different eye-trackers.

Scripts for this analysis can be found in Appendix J (section J.2).

Conclusions

After the analysis of the empirical data between the 2 eye-trackers, we can assume that GP3 collected more fixations on target per trial, and those fixations lasted significantly longer in comparison to the Eye-Link 1000. Additionally, Eye-Link seemed to be more precise in terms of spatial resolution, as the fixations appeared to be significantly closer to the target than the ones collected from the GP3. Therefore, we can assume that in an experimental paradigm like the one detailed in this thesis, accuracy and precision on target location is crucial. This was proven to be significant in the eye-tracking data collected by the Eye-Link 1000 eye-tracker. Additionally, the duration of the fixations and the number of fixations on target are not crucial for our design since we focused on raw eye-samples for the data interpretation/analysis. This suggests that future experiments, that will use the same methodology, should use Eye-Link 1000, as it provides better spatial accuracy.

C. Evaluating Gazepoint GP3 and Eye-Link 1000 use in different research contexts.

On the one hand, Gazepoint GP3 can be an ideal eye-tracker for a design that demands data collection in non-laboratory-based environments, as it is cheap, small, light, portable and requires only a laptop to operate. For example, studies that require data collection in nurseries, schools, retirement houses, hospitals, art centres, or even online could use Gazepoint GP3. Similarly, any design that contains large visual stimuli or wide areas or involves the recording of processes that their duration can be easily captured by 60 samples per second, can be efficiently run with a Gazepoint GP3. Gazepoint GP3 is also useful in designs similar to mine that are counting raw eye-samples on target locations, or fixations/saccades rather than focusing on pupil size differences. The only concern when dealing with count eye-data, is that the count data should be transformed as a percentage proportion of the estimate outcome variable, in order for the results to be reliable and reflect the actual process recorded. On the other hand, there are research areas in clinical psychology, that use eye-tracking as a risk assessment criterion for learning disabilities and clinical disorders. In these cases, high precision and accuracy are pre-requirements of the method, so usage of a low frequency eye-tracker like Gazepoint GP3 could be problematic. Designs that require maximum spatial and temporal precision of the recording of eye-movements should use EyeLink 1000.

EyeLink 1000 is one of the most commonly used eye-trackers in the field of psychology while Gazepoint GP3 is a relatively new eye-tracker that is trying to establish its worth in the field. In the following paragraphs I will preview some of the research contexts where the two eye-trackers have been used.

Murphy and Connaughton (2017) used EyeLink 1000 to examine automatic visual processing of social information in individuals with schizophrenia. They recruited 20 individuals with schizophrenia and 20 control individuals that were exposed to passive natural scenes that contained social (people) and non-social objects while their eye-movements were recorded. Their results suggested that individuals with schizophrenia spent significantly less time looking at the social objects, and they fixated more on non-social objects than the control group. In a design like this, the focus about choosing equipment would be based on the duration of fixations and their latency. From our results, we know that the GP3 had significantly longer fixation durations than Eye-Link 1000, therefore it could be potentially used for a similar design.

Eye-Link 1000 has been also used to examine bilingualism and word parsing. Tremblay (2011) compared the performance of native and non-native speakers of French on a parsing recognition task of misaligned syllables and word boundaries in a context of liaison. Participants had their eye-movements recorded while they had to recognise the auditory stimulus (word) in one out of four words that they were given as options per trial. The proportion of fixations on those four words during the trial duration was measured. Their results suggest that non-native speakers recognised harder liaison-initial real words, and consonant-initial words slower than liaison-initial nonsense words. However, native speakers recognised faster consonant-initial real and nonsense words than liaison-initial ones. In this design, the number of fixations and their duration is crucial, however the key element is the precision of location of the eye-movements. The stimuli are words, therefore have a very precise and small AOI. Based on our findings Eye-Link 1000 is the most suitable tracker as it can provide better spatial precision of the eye-movements.

Wu, Filipe, Leek and Thierry (2013) used EyeLink 1000 (SR Research Ltd., Mississauga, Canada) to investigate lexical access processing in bilingualism. In total they recruited 20

Chinese -English bilinguals and 20 native speakers of English. During the task, participants were presented with (a) the filler trials that contained one string of shapes (circles or squares) and three English words that worked as distractors, on a 4-item grid, and (b) the experimental trials, that consisted of four locations with English words, including a critical word that phonologically overlapped with the Chinese word for the shape of circle or square when translated into Chinese. At the end of each trial participants had to give a key response by clicking one out of 3 buttons, depending on if they say squares, circles, or a word. Their eye-movements were recorded throughout the task. Their results suggested that during the experimental trials bilingual participants looked more often and for longer durations the critical word that overlapped phonologically, while native speakers didn't. They concluded that lexical representations for both languages are activated in bilinguals, even for tasks that do not require explicit language processing but contain incidental word processing. This design focused on the duration of fixations and differences in them, but also in the spatial precision of the fixations. Therefore, a higher frequency tracker like an Eye-Link 1000, that has better spatial precision of eye-movements, is the most suitable equipment.

Sibley, Foroughi, Brown and Coyne (2018), used Gazepoint GP3 (60Hz) in order to examine whether this low-cost equipment could replicate previous research findings between working memory capacity the resting pupil size. To do so, they recruited a total number of 79 Navy and Marine Corps student pilots that firstly gave access to the experimenters to their aviation selection test scores, and then performed two tasks: (a) a colour change task where participants had the baseline of their pupil size measured based on the changes of the screen luminance and (b) an automated operation span test where participants were presented with a series of 75 math calculation tasks and 75 letter recall tasks. The Gazepoint GP3 eye-tracking system on binocular mode, detected the light flex of the pupil across all participants during the colour change task, however it failed to replicate previous findings about resting pupil size.

Additionally, a negative correlation instead of a positive correlation between the partial Operation Span scores and the resting pupil size measured in millimetres was observed. The researchers concluded that Gazepoint GP3 system's millimetre pupil size measurements should not be used as measurements of individual differences across participants. In our analysis, we didn't perform any type of analysis around pupillometry. However, there is no indicator of why GP3 wouldn't be suitable for the task, since it outperformed Eye-Link 1000 on recording more fixations per trial and on recording longer fixations.

Gazepoint GP3 (60Hz), has been used in the literature to investigate online language learning in combination with electroencephalogram (EEG) and behavioural data, by Notaro and Diamond (2018). They monitored the eye-movements, while they were recording EEG and mouse tracking (cursor movements and clicks), while adult participants performed online classes with only visual content of German language on Duolingo. The creation of this multimodal dataset aimed to be used for future analysis and further methods development. However, the fact that Gazepoint GP3 was used in a multimodal data collection set up, demonstrates one of its many potentials in future research and demonstrates how it's portability and accuracy can become handy in such designs. The fact that GP3 is small, portable and easy to adapt to any laptop or screen, gives the advantage in multi-equipment experimental set-ups like this one. However, one could argue that a higher frequency eye-tracker like Eye-Link 1000 would record more precisely temporal changes in eye-movements, therefore would provide a more precise eye-tracking filter for eye-movement artifacts in the EEG data.

Furthermore, Gazepoint GP3 (60 Hz) has been used by Costescu and Rosan (2019) for educational purposes on individuals with ASD, in order to create an assessment protocol. The idea behind this research was that since individuals with ASD struggle with written social information, eye-tracking could be used to identify differences in visual patterns while processing social information between individuals with low and high autism spectrum traits.

They concluded that individuals with high autism spectrum traits had different visual attention patterns and social vignettes than individuals with low autism spectrum traits and suggested that those could be applied in intervention techniques for social skills in individuals with ASD. This research provides evidence that GP3 has been used in clinical experimental set ups, and that it was able to record all the components that differed between the low and high autism spectrum traits. However, in a set up like this one, accuracy in spatial and temporal resolution is crucial. Since Eye-Link 1000 has better temporal resolution (higher frequency) and spatial resolution (higher precisions on locations of eye-movements), it could be used for this design, to provide more reliable results, especially if they are intended to be used to form clinical assessments.

D. Conclusions

To conclude, before deciding in favour of or against a technical equipment such as an eye-tracker, we should evaluate the design requirements and how the design measurements and outcome variables can be affected by the technical equipment manipulation. Not all designs require high efficiency and accuracy precision in order to capture the mechanisms suggested, and therefore low frequency equipment can be found handy for such designs and data collection in non-laboratory-based environments. Gazepoint GP3 has been found to be non-restrictive in terms of the quality of data obtained, so it is highly recommended for set ups like mine that assess implicit learning, even on time ranges smaller or equal to 750ms.

An evaluation of thesis outcomes and application of thesis findings in different research contexts.

Chapter summary

This Chapter reflects the new experimental paradigm on previous methodologies that have been used in the literature and highlights limitations of the current design and how they can be overcome in future designs. Additionally, a critical evaluation of the findings about the sequence length effects and the information processing findings is given, in relation to previous theories and findings. Finally, an application of these research findings is suggested within educational and clinical concepts and further research directions are proposed.

A. A critical evaluation of the new experimental paradigm on previous methodologies used to investigate SL, limitations and implications on future research.

The current thesis developed a new experimental paradigm that is observing learning in real time rather than assessing the sequential SL process at the end of it. It also demonstrates evidence for a hierarchical structure of sequential SL in information processing rather an all-or-none processing in learning. The positioning effects found in the tasks of this thesis, are similar across all sequence lengths and mixtures of lengths, suggesting that the learning of an item increases as its position in the sequence increases. In other words, in a sequence of ABCD, the learning follows the pattern of AB<BC<CD. The sequence length effects demonstrated differences in learning between same length and mixed length tasks for the sequences with 2 items and 4 items, while performance was similar across types of tasks (mixed/non-mixed lengths) for the 3 items sequence. Sequences of 2 dots were learned significantly better when they were presented in a same length task, than when they were mixed with longer sequences, while sequences of 4 dots were learned better when they were mixed with shorter sequences of 2 dots.

Th element of novelty is introduced in this thesis by the creation of a new experimental paradigm in sequential SL. This new experimental paradigm offers a new perspective on how the SL mechanism should be observed and measured. As we have seen from previous chapters, most of the methodologies used to observe or assess the SL mechanism have been measuring the learning outcome at the end of the SL process (Saffran et al.,1996; Saffran, Johnson, Aslin & Newport, 1999; Perruchet & Vinter, 1998; Newport & Aslin, 2004; Johnson & Tyler, 2010), rather than evaluating and observing the process while it occurs, and the learning develops. That was indeed problematic, since many procedural and perceptual effects that related to the stimuli of the task or the procedure of the task, were classified as effects that derived from the SL mechanism, since there was little option to discriminate between those two. Additionally,

processes such as information encoding in SL, that are widely used in artificial language contexts (Monaghan et al., 2019; Hoch, Tyler & Tillmann, 2013) or language learning models, were unable to be accessed while they were occurring due to task designs and therefore, they were evaluated as the results of the process.

The main contribution of the suggested new experimental paradigm is that provides a solution to the above problem and allows the mechanism of SL to be observed during its time course in order to provide more details about how SL occurs and develops over time. Furthermore, the task doesn't include any prior familiarisation/ exposure to the task or explicit instructions about the form of the task and therefore allows for the observation of the SL mechanism across all timepoints, when exposure to the stimuli/procedure is 0 and the learning process is completely implicit (non-biased from instructions- "Guess where the dot will appear next"). This fact implies that the new experimental paradigm is ecologically valid (due to its implicit nature) but is also construct and content valid since it measures an implicit SL mechanism, from the start of the process with an implicit task.

The experimental coding of the new paradigm allows the usage of two eye-tracking systems (Gazepoint GP3; EyeLink 1000, SR Research Ltd., Mississauga, Canada) and mouse tracking (hasn't been used in an experimental set up in this thesis) via open source/access frameworks promoting this way the concept of Open Access/Source Science and allowing future experimental manipulations to be created fast and efficiently across different equipment and modalities. All experimental set up variables such as durations of gaze-contingent and time displayed windows, sizes of AOIs, allocation and number of location arrays on the monitor, etc, can be easily controlled by changing the numeric values in the experimental run or set up file. This way, this new experimental methodology, can not only be applied on the research aims of the current thesis, but can also be used by future researchers that investigate sequential SL in different concepts and under different experimental specifications.

The most important obstacle that this new design overcame, was the fact that it achieved to capture sequential SL developing during time, by observing the time course of the mechanism rather than the resulting outcome on an external measure of SL. Future research can use the finding of the current thesis, and further investigate sequential SL in in reading (Arciuli, 2018), word segmentation (Johnson & Tyler, 2010) and speech perception (Franco et al., 2015).

Furthermore, the current paradigm offers a clear observation of individual differences in learning patterns during SL across participants. Individual differences in SL, were not part of the thesis research aims, however they are a huge research topic in the literature (Misyak & Christiansen, 2012; Siegelman, Bogaerts & Frost, 2017) and therefore individual learning rates have been plotted and attached in Appendix K (sections K.1- K.6), from the 6 sequential SL tasks of 2 dots, 3 dots, 4 dots, 2&3 dots, 2&4 dots and 3&4 dots. The plots were generated with an R script that is in Appendix L (Section L.1), by fitting a smoothed cubic function of occurrence on the raw data. Learning was measured as raw eye-samples on target location during the guessing period of 750ms (blank period). Eye-samples on the first item of a sequence, were excluded since they were not representing part of the learning process, but they were random across the location array during the task (the sequences were presented on a continuous stream, so there was no clue about when or which the next sequence would be). By understanding better individual differences in learning mechanisms such as SL, that are highly involved in processes such as language learning, we can understand better why there is such variability in learning performance in infants during language learning and vocabulary growth, but also how exposure can facilitate the learning process and improve the learning outcome in educational intervention programmes.

Similarly, the individual differences in SL also recommend that not everyone is equally good at SL, but with a combination of both explicit and implicit rules as Arcuili et al. (2018) suggested, greater learning can be achieved. Even though, SL is a domain general mechanism

as examined by the literature, domain specific differences in performance occur due to differences in processing the modalities (stimuli processing) and previous exposure to these modalities (expertise in modalities- exposure to stimuli). Further investigation across different types of learners (verbal vs visual) and modalities in SL (auditory visual) would be able to provide a solution of how SL performance can be improved by using the new experimental paradigm with sequences of visual – as it is now, auditory non-verbal and auditory verbal stimuli. During the experimental auditory tasks, in each trial each location would be associated with an auditory stimulus (verbal or non-verbal) and it would be presented on the visual array with a green dot, suggesting a multimodal sequential SL paradigm, where the association of the location and the auditory stimuli is mediated by the association of the location and the visual stimuli (green dot) on the monitor. That experimental set up would provide more information about SL across different domains and during multisensory input, but would also increase the ecological validity, since multisensory input is common in everyday learning environments. For example, infants in their everyday life, when they are playing with toys or eating are being exposed to an auditory verbal stimulus(word) and a visual stimulus (object in the scene) and are expected to learn that the specific object is associated with the verbal label in order to successfully build a mental representation.

However, as every experimental paradigm, it comes with its limitations. One of the main limitations of this paradigm is that it observes the learning within a very specific time window (blank period of guessing) that is only 750 ms. Given the circumstances within our experimental set up the exposure was a pre-set factor of 40 occurrences (experimental time limitations) for each item of a sequence within a task. In terms of the actual learning process, that could imply that participants haven't had enough exposure to the sequences, or enough time to plan eye-movements while they are still learning the sequence. Furthermore, even when a sequence is learned, and individuals can plan in advance of the blank period their eye-

movements, it is not necessary to attend to the target/correctly guessed location for longer than 200-250 ms. This fact was evident in the thesis results by the low means of hits on target-learning (aggregated across participants), since the learning was calculated as a count of raw eye-samples on the target location during the blank period of 750ms, and the percentage of learning rate was calculated as the proportion of the raw eye-samples on target, divided by the maximum recording of eye-samples number that each eye-tracker allowed (depending on its frequency). That suggests that the observed count data that were interpreted as "bad scores" of learning, could have actually been misinterpreted, either because participants didn't have enough time to look more on the target location due to anticipation of eye—movement planning on that window, or due to the fact that participants did not have to attend for 750ms even when they learned the location. In order to deal with this problem, all the modelling that occurs in chapters 6 and 7 has been replicated in a binomial format in Appendix M, where learning was calculated as a binary variable of 0s and 1s. The threshold for defining a location as learned was calculated by getting the mean of eye-samples on the first item, from each of the 4 sequences, across all 6 tasks and all 36 subjects. That threshold was representing the hit on target by chance process. That threshold was 13,21 eye-samples on target and because it was aggregated across participants that did and didn't learn, it was rounded up to 25 eye-samples to make the threshold criterion stricter. Any target with equal or greater than 25 eye-samples on it was classified as learned (1), and every target with less than 25 eye-samples was classified as non-learned (0). The binary models replicate the findings of the Poisson distribution models in every aspect but with higher learning scores. Since they don't add additional information about the data, they were not included as part of the analysis in this thesis but are demonstrated in Appendix M to show how different mathematic calculations can account for low numeric values and validate that the results of this thesis are accurate.

Finally, another limitation of the current experimental set up, was the fact that learning was aggregated across learners and non-learners in this thesis. Since, the new experimental design was under examination for its validity, it was important to include in the data the participants that didn't learn and try to understand why this happened. Part of the non-learners is reflecting individual differences in SL, as 40 occurrences might not be enough exposure for them to learn the task. Another part of the non-learners represents the participants that were not motivated to do the task, the participants that lost their attentional focus on the task and were confused by the implicit nature of the task or the participants that were exhausted by the duration of the total experiment. That parameter was controlled in Design B and in the data presented in the analysis chapters, by introducing into the design the negative feedback beep sound that motivated participants to perform well in the task (guess correctly) and concentrate, and by adding breaks after each task. Future experiments that use the suggested new experimental paradigm are highly recommended to increase the amount of occurrence within a task, so that they can allow more exposure within a task that could potentially lead to greater learning.

B. A reflection of the findings on information processing during sequential SL.

The positioning effects in Chapter 6, suggest a hierarchical structure in the SL process. In more details, this thesis has provided clear evidence that across all 6 tasks, an increased learning rate is observed as the item order in the sequence increases. In other words, the learning of the location of the n*th* dot is less than the learning of the n+1*th* dot etc where n is the order of the item in a sequence. These findings suggest evidence against the all or none understanding of learning. According to the all or none theory of learning in our experimental paradigm a sequence of ABCD items can either be learned as a whole (that doesn't allow differences in scores between the S-R of AB, BC and CD) or it can't be learned at all.

These findings are additionally suggesting evidence for a unit chunking mechanism of SL (Franco & Destrebecqz, 2012; Isbilen, Mccauley, Kidd & Christiansen, 2020), rather than simple TPs (Perruchet, 2018; Perruchet & Desault, 2008; Newport, Hauser, Spaepen & Aslin. 2004). If the information processing relied merely on TPs, then no differences in the learning of the items within the sequence would be expected, since TPs were even across all items within a sequence. However, future research should use the same experimental paradigm to investigate how unit chunking information occurs during SL (pairs or triplets) and if it is solid or depends on the length of a sequence (even or odd number). That hypothesis could be easily added by introducing sequences of greater lengths in the tasks, and by creating tasks that contain part-known sequences and part-new sequences and compare the learning across the different tasks.

Also, the same experimental design could be used to examine information recall and generalisation, by introducing greater exposure of the task until the learning of the pattern-implicit rule (sequence length, e.g 3 dots) was successful on an 75% level and above and then introduce a task that contained the same pattern-implicit rule (sequence length, e.g.3dots task) on a different location array. A comparison between the two learning rates would show how once the learning of a pattern is successful, it can be generalised to facilitate the learning of novel stimuli that contain the same pattern-implicit rule.

These findings can be used by AGL (Beran & Qwren, 2018; Lieberman et al., 2004) models, or artificial language models, to provide a deeper understanding on how SL occurs and improve current methodologies and models that are being used to understand language learning or visual SL in adults and infants. Furthermore, studies that allow simultaneous data collection of EEG and eye-tracking data with the usage of this experimental paradigm and the portable Gazepoint GP3 eye-tracker, can identify neuronal networks that are activated during the encoding and the

retrieval information phase, and differences in activation frequencies when the knowledge is generalised to novel stimuli.

C. A reflection of the findings on sequence length and mixture length effects during sequential SL.

Sequence length is one of the main components of sequential SL (Saffran et al., 1999; Sanchez & Reber, 2018; Slone & Johnson, 2018; Stadler & Neely, 1997; Heimbauer, Conway, Christiansen, Beran & Owren, 2018) and it is necessary to understand how it works, in order to understand how processes such as language learning, vocabulary growth, and word segmentation occur. The findings of this thesis, suggest that longer sequences are harder to learn than shorter sequences, however when sequence lengths are mixed within a task the learning of longer sequences is facilitated and the learning of shorter sequences is impeded, in comparison to the performance in the same length tasks.

This kind of information is extremely useful if we consider that infants use SL as their main mechanism to learn language and form their verbal units in stages of syllables, words and sentences. These verbal units coexist in multiple lengths, and most of the times are blended within a stream of multiple lengths that is presented in specific order during time. For example, a syllable consists from 1, 2, 3…etc letters or sounds, a word can consist of 1, 2, 3, …etc. syllables, and a sentence can consist of 1, 2, 3...etc. lengths of words. Understanding how the mixture of lengths within these verbal units facilitates or impedes learning can be crucial for educational and clinical developmental reasons.

Interestingly, it is still unclear if the difference in performance across the difference lengths when lengths are mixed and not, occurs due to procedural cognitive load. It could be the case that in a same length task that consists of 4 sequences each with a sequence length of 4 dots, participants need to learn 16 different locations, but in a mixed length task of 2 sequences of 2

dots and 2 sequences of 4dots participants have to learn only 12 locations and therefore less information processing. However, it could be the case that due to the way encoding occurs (pairs or triplets), even for mixed length sequence tasks (2&4 dots) there is facilitation from applying the rule of chunking of one sequence length (e.g. pairs) to the other (4 dots) that leads to faster learning. If the observed differences are actually due to procedural cognitive load, then the longer sequences in same length task the harder the learning would be, while when those long sequences are mixed with shorter their learning will increase.

Furthermore, sequences of mixed lengths have shown that the difference in performance observed between mixed and non-mixed length task, might not rely on the sequence length itself, but on the way that information encoding happens during sequential SL. We saw that participants performed significantly better in the 2&4 dots task rather than the 2&3 and 3&4 dots tasks. If the difference in performance relied purely on the amount of information that needs to be processed, it would be expected to see the learning rate of 2&3 dots task to be higher than the learning rate of 2&4 dots task. Also, the fact that 3 dots task performed equally well across mixed and non-mixed tasks should be taken into consideration.

One possible explanation that could explain all the sequence length and mixture findings could be that chunking mechanism operates with both triplets and pairs. Therefore, when an even number of sequence lengths is blended (2&4) dots, it is automatically facilitated due to the usage of the same chunking mechanism (pairs). However, when sequences with odd and even lengths are blended, the learning is impeded, as the chunking mechanism anticipates between the usage of pairs and triplets. In order to be able to answer that question, sequences with 5 items and 6 items should be introduced to the experimental paradigm. By comparing performance between mixed length tasks, with different consistencies in lengths (e.g. 2&5 dots, 3&5 dots, 2&4 dots, 5& 4 dots), we would be able to understand how the chunking occurs and if the impediment of learning is clearly due procedural cognitive load.

D. Application of thesis findings in applied educational and clinical concepts & future research directions.

The findings of this thesis have direct implications in the research field as explained in the sections above, but also in applied educational psychology, clinical psychology and psycholinguistics.

Since SL is the main mechanism of language learning (written and verbal format), direct implications of these findings could be applied in education. By creating online task/ games that implicitly contain linguistic structures and by formulating and ordering the structures in a way that facilitates and maximises the learning, we could examine how typically developed and atypical children learn language (vocabulary, reading, orthography-spelling), while maintaining a pleasant character in the educational task.

A series of studies (Arciuli, 2018; Arciuli & Conway, 2018; Arciuli & Simpson, 2011) have shown that SL should be applied in educational contexts in order to facilitate learning across typically and atypically developed children. Since SL is a mechanism highly involved in processes such as language learning (Saffran et al.,1996), vocabulary learning (Chan & Monaghan, 2019), reading (Arciuli, 2018), orthography (Protopapas et al., 2017), and word segmentation (Perruchet & Vinter ,1998; Perruchet & Desaulty, 2008), the experimental paradigm of this thesis must be applied in future research in developmental projects, in order to provide a clearer understanding of differences observed in SL performance between different developmental age groups and across modalities.

Additionally, the experimental design suggested above could be used in clinical populations with developmental disabilities (dyslexia, dyspraxia, ASD, ADHD), in order to understand why SL is harder, how information is being processed during sequential SL and how we can intervene in order to facilitate the learning in the process. As it was shown in Chapter 8, eye-

movements have always been a solid method of diagnostics and performance comparisons across clinical populations. What this new design is suggesting, is not only the observation of differences in performance across typical vs atypical, but an identification of the problem during the learning procedure when targeted to the problem solution.

Jones, Tarpey, Hamo, Carberry, Brouwer and Lord (2018) investigated the mechanism of SL on young autistic children and suggested that poor verbal abilities of ASD children can be explained by differences observed in visual SL performance in a task between typical and ASD children. Jones & Klin (2013) also highlighted the importance of early assessment of ASD, since visual attentional decline can be evident on ASD infants from the early age of 2-6 months-old. A methodology like the one suggested in the thesis, could provide useful information about the differences in SL processing between typical individuals and ASD individuals from a very young age, in order to tackle the potential cognitive deficits in advance.

Since SL performance across the visual and auditory domains have been shown to differ in individuals with ADHD (Kaitlyn & Stevenson, 2018), the new paradigm could be used across individuals with ADHD and typical individuals to provide more details about how information encoding and pattern recognition occurs in ADHD (Joao, Marcelo, Efujita & Luis, 2012). Additional expansion of the developmental spectrum of the task is also recommended, since SL has been widely investigated in infants and children to understand processes such as language learning (Saffran et al.,1999; Slones & Johnson, 2018; Arciuli, 2018; Arciuli & Simpson, 2011).

Areas such as computational models of language, that have been using TPs as the main mechanism of SL, can be directly affected by the implementations of this thesis. Reforming currently existing models, in accordance to the encoding of information via chunking units and not TPs, could improve models performance, increase the fit of their predictions and result in

models that imitate better the actual language learning process. The same experimental paradigm could be used so that the sequential learning occurs within sequences of verbal information (written or auditory), to validate suggested orthographic (Protopapas et al., 2017) and speech production (Lee, 2017) theories that use SL as their main mechanism.

E. Conclusions

To conclude, this thesis is suggesting a new strong experimental paradigm that investigates the SL mechanism during its time course, rather than assessing its outcome at the end of the process. That allows the observation of the learning while it is developing from time-point zero and requests zero prior exposure to the stimuli or the experimental procedure, contrast with other methods such as artificial language tasks that were exposing participants to already known streams of syllables (Saffran et al., 1996; Aslin & Newport, 2012). The items positioning effects shown in this thesis, provide strong evidence that SL learning used a unit chunking information processing that has a hierarchical structure, rather than a mechanism of extraction of TPs. Sequence length effects on the other hand, showed that while 3 items dots are learned in same tasks and mixed task with a same rate, 4-item sequences are learned better when they are mixed with 2-item sequences, than when being learned in a same length task. However, 2-item sequences are learned better when they are presented in non-mixed tasks. Furthermore, performance across mixed length sequence tasks revealed that tasks that contain sequences with even sequence length number (2&4), the learning rate of the task is significantly better than in tasks with mixtures of odd and even sequence lengths (2&3, 3&4). Finally, findings of this thesis can be implemented in future research about information processing in sequential SL and language learning in both typical and atypical populations.

References

Abe, J. A. A. (2011). Positive emotions, emotional intelligence, and successful experiential learning. *Personality and Individual Differences, 51*(7), 817-822. doi:10.1016/j.paid.2011.07.004

Amberkar, A., Awasarmol, P., Deshmukh, G., & Dave, P. (2018, March). Speech recognition using recurrent neural networks. In 2018 international conference on current trends towards converging technologies (ICCTCT) (pp. 1-4). IEEE.

Anderson, B. A., Laurent, P. A., & Yantis, S. (2011). Learned value magnifies salience-based attentional capture. *Plos One, 6*(11). doi:10.1371/journal.pone.0027926

Andrade, C. (2018). Internal, external, and ecological validity in research design, conduct, and evaluation. *Indian Journal of Psychological Medicine, 40*(5), 498-499. doi:10.4103/IJPSYM.IJPSYM_334_18

Arciuli, J. (2018). Reading as statistical learning. *Language, Speech & Hearing Services in Schools (Online), 49*(3), 634-643. doi:10.1044/2018_LSHSS-STLT1-17-0135

Arciuli, J., & Conway, C. M. (2018). The promise—and challenge—of statistical learning for elucidating atypical language development. *Current Directions in Psychological Science : A Journal of the American Psychological Society, 27*(6), 492-500. doi:10.1177/0963721418779977

Arciuli, J., & Simpson, I. C. (2011). Statistical learning in typically developing children: The role of age and speed of stimulus presentation. *Developmental Science, 14*(3), 464. doi:10.1111/j.1467-7687.2009.00937.x

Aslin, R. (2017). Statistical learning: A powerful mechanism that operates by mere exposure. Wiley Interdisciplinary Reviews, 8(1-2), N/a.

Aslin, R. N., & Newport, E. L. (2012). Statistical learning: From acquiring specific items to forming general rules. Current directions in psychological science, 21(3), 170–176. https://doi.org/10.1177/0963721412436806

Aslin, R. N., & Newport, E. L. (2012). Statistical learning: From acquiring specific items to forming general rules. *Current Directions in Psychological Science : A Journal of the American Psychological Society, 21*(3), 170-176. doi:10.1177/0963721412436806

Atkinson, R. C., & Shiffrin, R. M. (1968). Human memory: A proposed system and its control Processes.*Psychology of Learning and Motivation, 2*, 89-195. doi:https://doi.org/10.1016/S0079-7421(08)60422-3

Aya Shirama, Toshinobu Takeda, Haruhisa Ohta, Akira Iwanami, Shigenobu Toda, & Nobumasa Kato. (2020). Atypical alert state control in adult patients with ADHD: A pupillometry study. *PloS One, 15*(12), E0244662.

Bates, D., Mächler, M., Bolker, B., Walker,S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software, 67*(1) doi:10.18637/jss.v067.i01

Bechtel, W., & Abrahamsen, A. (2005). Explanation: A mechanist alternative. *Studies in History and Philosophy of Science.Part C, Studies in History and Philosophy of Biological and Biomedical Sciences, 36*(2), 421-441. doi:10.1016/j.shpsc.2005.03.010

Bechtel, W., 2008. Mechanisms in Cognitive Psychology: What Are the Operations? *Philosophy of Science*, 75(5), pp.983–994. 10.1086/594540.

Bechtel, W., Abrahamsen, A., 2012. Diagramming Phenomena for Mechanistic Explanation [online]. Available at: https://escholarship.org/uc/item/1cs113rv.

Bechtel, W.. (2012). Mental Mechanisms: Philosophical Perspectives on Cognitive Neuroscience. Mental Mechanisms: Philosophical Perspectives on Cognitive Neuroscience. 1-308. 10.4324/9780203810095.

Berger, A., Della Pietra, S. A., & Della Pietra, V. J. (1996). A maximum entropy approach to natural language processing. Computational linguistics, 22(1), 39-71.

Betzler, R. J. (2016). Is statistical learning a mechanism? *Null, 29*(6), 826-843. doi:10.1080/09515089.2016.1167179

Broadbent, H. J., Osborne, T., Rea, M., Peng, A., Mareschal, D., & Kirkham, N. Z. (2018). Incidental category learning and cognitive load in a multisensory environment across childhood. Developmental Psychology, 54(6), 1020.

Brown, P. F., Della Pietra, V. J., Desouza, P. V., Lai, J. C., & Mercer, R. L. (1992). Class-based n-gram models of natural language. Computational linguistics, 18(4), 467-480.

Brünken, R., Steinbacher, S., Plass, J. L., & Leutner, D. (2002). Assessment of cognitive load in multimedia learning using dual-task methodology. Experimental psychology, 49(2), 109.

Brysbaert, M., Mandera, P., & Keuleers, E. (2018). The word frequency effect in word processing: An updated review. *Current Directions in Psychological Science,* 27(1), 45-50.

Bulf, H., Johnson, S. P., & Valenza, E. (2011). Visual statistical learning in the newborn infant. *Cognition, 121*(1), 127-132. doi:https://doi.org/10.1016/j.cognition.2011.06.010

Burgers, C., Eden, A., van Engelenburg, M. D., & Buningh, S. (2015). How feedback boosts motivation and play in a brain-training game. Computers in Human Behavior, 48, 94-103.

Chan, J., & Monaghan, P. (2019). Simulating bilingual word learning: Monolingual and bilingual adults' use of cross-situational statistics. *Proceedings of the 41st Cognitive Science Society Conference.*

Chawarska, K., Macari, S., & Shic, F. (2013). Decreased Spontaneous Attention to Social Scenes in 6-Month-Old Infants Later Diagnosed with Autism Spectrum Disorders. *Biological Psychiatry (1969), 74*(3), 195-203.

Chevalier, N., Blaye, A., Dufau, S., & Lucenet, J. (2010). What visual information do children and adults consider while switching between tasks? eye-tracking investigation of cognitive flexibility development. *Developmental Psychology, 46*(4), 955. doi:10.1037/a0019674

Chomsky, N. (1957). Syntactic structures. Mouton.

Chomsky, N. (1957). Syntactic Structures. The Hague: Mouton.

Christiansen, M. H., & Chater, N. (1999). Connectionist natural language processing: The state of the art. Cognitive science, 23(4), 417-437.

Cleeremans, A., & McClelland, J. L. (1991). Learning the structure of event sequences. Journal of Experimental Psychology: General, 120(3), 235.

Conway, C. M., & Christiansen, M. H. (2001). *Sequential learning in non-human primates* doi:10.1016/S1364-6613(00)01800-3

Conway, C. M., & Christiansen, M. H. (2006). Statistical learning within and between modalities: Pitting abstract against stimulus-specific representations. Psychological science, 17(10), 905-912.

Conway, C. M., & Pisoni, D. B. (2008). Neurocognitive basis of implicit learning of sequential structure and its relation to language processing. *Annals of the New York Academy of Sciences, 1145*(1), 113. doi:10.1196/annals.1416.009

Conway, C.M. and Christiansen, M.H. (2005) Modality-constrained statistical learning of tactile, visual, and auditory sequences. *J. Exp. Psychol. Learn. Mem. Cogn.* 31, 24–39

Costescu, C., & Rosan, A. (2019). Developing an assessment protocol to identify the characteristics of ASD using eye-tracking for educational purpose. *Journal of Applied Technical and Educational Sciences, 9*(4), 70-87.

Craver, Carl and James Tabery, "Mechanisms in Science", *The Stanford Encyclopedia of Philosophy* (Summer 2019 Edition), Edward N. Zalta (ed.), URL= https://plato.stanford.edu/archives/sum2019/entries/science-mechanisms/ .

Creel, S.C., Newport, E.L., Aslin, R.N., (2004). Distant Melodies: Statistical Learning of Nonadjacent Dependencies in Tone Sequences. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 30(5), pp.1119–1130. 10.1037/0278-7393.30.5.1119.

Danner, D., Hagemann, D., & Funke, J. (2017). Measuring individual differences in implicit learning with artificial grammar learning tasks: Conceptual and methodological conundrums. *Zeitschrift Für Psychologie, 225*, 5-19. doi:10.1027/2151-2604/a000280

Dash, B. (2021). A hybrid solution for extracting information from unstructured data using optical character recognition (OCR) with natural language processing (NLP).

Dennis, N. A., Howard, J. H., & Howard, D. V. (2006). Implicit sequence learning without motor sequencing in young and old adults. Experimental brain research, 175, 153-164.

Diessel, H. (2017). Usage-based linguistics. In Oxford research encyclopedia of linguistics.

Du, Y., & Clark, J. (2017). New insights into statistical learning and chunk learning in implicit sequence acquisition. *Psychonomic Bulletin & Review*, 24(4), 1225-1233.

Ebbinghaus, H. (2013). Memory: A contribution to experimental psychology. *Annals of Neurosciences, 20*(4), 155. doi:10.5214/ans.0972.7531.200408

Ekman, P., & Friesen, W. V. (1975). *Unmasking the face: A guide to recognizing emotions from facial clues.* Prentice-Hall.

Elman, J. L. (1991). Distributed representations, simple recurrent networks, and grammatical structure. Machine learning, 7, 195-225.

Endress, A. D., & Szabó, S. (2017). Interference and memory capacity limitations. *Psychological Review, 124*(5), 551-571. doi:10.1037/rev0000071

Endress, A. D., Slone, L. K., & Johnson, S. P. (2020). Statistical learning and memory. *Cognition, 204* doi:10.1016/j.cognition.2020.104346

Estes, W. K. (1950). Toward a statistical theory of learning. *Psychological Review, 57*(2), 94–107. https://doi.org/10.1037/h0058559

Estes, W.K. (1970). Learning Theory and Mental Development. New York: Academic Press.

Farkas, B. C., Janacsek, K., & Nemeth, D. (2022). The reliability of the alternating serial reaction time task. PsyArXiv. February, 7.

Fiser, J., & Aslin, R. N. (2001). Unsupervised statistical learning of higher-order spatial structures from visual scenes. *Psychological Science, 12*(6), 499-504.

Franco, A., & Destrebecqz, A. (2012). Chunking or not chunking? how do we find words in artificial language learning? *Advances in Cognitive Psychology, 8*(2), 144-154. doi:10.2478/v10053-008-0111-3

Franco, A., Eberlen, J., Destrebecqz, A., Cleeremans, A., & Bertels, J. (2015). Rapid serial auditory presentation: A new measure of statistical learning in speech segmentation. *Experimental Psychology, 62*(5), 346. doi:10.1027/1618-3169/a000295

Frank, M. C., Goldwater, S., Griffiths, T. L., & Tenenbaum, J. B. (2010). Modeling human performance in statistical word segmentation. *Cognition, 117*(2), 107-125. doi:10.1016/j.cognition.2010.07.005

Frazier, T., Klingemier, Beukemann, Speer, Markowitz, Parikh, . . . Strauss. (2016). Development of an Objective Autism Risk Index Using Remote Eye Tracking. *Journal of the American Academy of Child and Adolescent Psychiatry, 55*(4), 301-309.

Freedberg, M., Schacherer, J., & Hazeltine, E. (2016). Incidental learning of rewarded associations bolsters learning on an associative task. *Journal of Experimental Psychology.Learning, Memory, and Cognition, 42*(5), 786. doi:10.1037/xlm0000201

French, R. M., Addyman, C., & Mareschal, D. (2011). TRACX: a recognition-based connectionist framework for sequence segmentation and chunk extraction. Psychological review, 118(4), 614.

Frost, R. L. A., & Monaghan, P. (2016). Simultaneous segmentation and generalisation of non-adjacent dependencies from continuous speech. *Cognition, 147*, 70-74. doi:10.1016/j.cognition.2015.11.010

Frost, R., Armstrong, B. C., Siegelman, N., & Christiansen, M. H. (2015). Domain generality versus modality specificity: The paradox of statistical learning. *Trends in Cognitive Sciences, 19*(3), 117-125. doi:10.1016/j.tics.2014.12.010

Funke, J., & Frensch, P. (2007). Complex problem solving: The european Perspective—10 years after. (pp. 25-47) doi:10.4324/9781315091938-2

Ghahghaei, S., & Verghese, P. (2015). Efficient saccade planning requires time and clear choices. *Vision Research (Oxford), 113*, 125-136. doi:10.1016/j.visres.2015.05.006

Gina M. Notaro, & Solomon G. Diamond. (2018). Simultaneous EEG, eye-tracking, behavioral, and screen-capture data during online German language learning. *Data in Brief, 21*, 1937-1943.

Glennan, S.S. (1996) Mechanisms and the nature of causation. *Erkenntnis* 44**,** 49–71. https://doi.org/10.1007/BF00172853

Gobet, Lane, Croker, Cheng, Jones, Oliver, & Pine. (2001). Chunking mechanisms in human learning. *Trends in Cognitive Sciences*, 5(6), 236-243.

Gold, J., Law, C. T., Connolly, P., & Bennur, S. (2010). Relationships between the threshold and slope of psychometric and neurometric functions during perceptual learning: Implications for neuronal pooling. *Journal of Neurophysiology, 103*(1), 140-154. doi:10.1152/jn.00744.2009

Goldberg, A. E. (2019). Explain me this: Creativity, competition, and the partial productivity of constructions. Princeton University Press.

Goldwater, S., Griffiths, T. L., & Johnson, M. (2009). A Bayesian framework for word segmentation: Exploring the effects of context. Cognition, 112(1), 21-54.

Gomez-Perez, J. M., Denaux, R., & Garcia-Silva, A. (2020). A Practical Guide to Hybrid Natural Language Processing: Combining Neural Models and Knowledge Graphs for NLP (pp. 1-268). Springer.

Greenberg, J., Reuman, L., Hartmann, A., Kasarskis, I., & Wilhelm, S. (2014). Visual hot spots: An eye tracking study of attention bias in body dysmorphic disorder. *Journal of Psychiatric Research, 57*(1), 125-132.

Gullberg, M., Roberts, L., Dimroth, C., Veroude, K., & Indefrey, P. (2010). Adult Language Learning After Minimal Exposure to an Unknown Natural Language. *Language Learning*, 60, 5-24.

Haith, A. M., Reppert, T. R., & Shadmehr, R. (2012). Evidence for hyperbolic temporal discounting of reward in control of movements. *The Journal of Neuroscience : The Official Journal of the Society for Neuroscience, 32*(34), 11727. doi:10.1523/JNEUROSCI.0424-12.2012

Hanna, E., & Meltzoff, A. N. (1993). Peer imitation by toddlers in laboratory, home, and day-care contexts: Implications for social learning and memory. *Developmental Psychology, 29*(4), 701. doi:10.1037/0012-1649.29.4.701

Hauser, M.D. et al. (2001) Segmentation of the speech stream in a nonhuman primate: statistical learning in cotton-top tamarins. *Cognition* 78, B53–B64

Hayne, H., Boniface, J., & Barr, R. (2000). The development of declarative memory in human infants: Age-related changes in deferred imitation. *Behavioral Neuroscience, 114*(1), 77-83. doi:10.1037/0735-7044.114.1.77

Hayne, H., Greco, C., Earley, L., Griesler, P., & Rovee-Collier, C. (1986). Ontogeny of early event memory: II. encoding and retrieval by 2- and 3-month-olds. *Infant Behavior and Development, 9*(4), 461-472. doi:https://doi.org/10.1016/0163-6383(86)90018-4

Hayne, H., MacDonald, S., & Barr, R. (1997). Developmental changes in the specificity of memory over the second year of life. *Infant Behavior & Development, 20*(2), 233-245. doi:10.1016/S0163-6383(97)90025-4

Heimbauer, L., Conway, C., Christiansen, M., Beran, M., & Owren, M. (2018). Visual artificial grammar learning by rhesus macaques (macaca mulatta): Exploring the role of grammar complexity and sequence length. *Animal Cognition, 21*(2), 267-284. doi:10.1007/s10071-018-1164-4

Hoch, L., Tyler, M. D., & Tillmann, B. (2013). Regularity of unit length boosts statistical learning in verbal and nonverbal artificial languages. *Psychonomic bulletin & review*, *20*(1), 142–147. https://doi.org/10.3758/s13423-012-0309-8

Houwer, J., Teige-Mocigemba, S., Spruyt, A., & Moors, A. (2009). Implicit measures: A normative analysis and review. *Psychological Bulletin, 135*(3), 347-368. doi:10.1037/a0014211

Howard, D. V., & Howard, J. H. (1992). Adult age differences in the rate of learning serial patterns: Evidence from direct and indirect tests. *Psychology and Aging, 7*(2), 232–241. https://doi.org/10.1037/0882-7974.7.2.232

Howard, J. H., Mutter, S. A., & Howard, D. V. (1992). Serial pattern learning by event observation. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 18*(5), 1029–1039. https://doi.org/10.1037/0278-7393.18.5.1029

Hunt, R. H., & Aslin, R. N. (2001). Statistical learning in a serial reaction time task: access to separable statistical cues by individual learners. Journal of Experimental Psychology: General, 130(4), 658.

Hutmacher F. (2019). Why Is There So Much More Research on Vision Than on Any Other Sensory Modality?. Frontiers in psychology, 10, 2246. https://doi.org/10.3389/fpsyg.2019.02246

Isbilen, E. S., Mccauley, S. M., Kidd, E., & Christiansen, M. H. (2020). Statistically induced chunking recall: A memory-based approach to statistical learning. *Cognitive Science, 44*(7) doi:10.1111/cogs.12848

Jacoby, L. L., Kelley, C., Brown, J., & Jasechko, J. (1989). Becoming famous overnight: Limits on the ability to avoid unconscious influences of the past. *Journal of Personality and Social Psychology, 56*(3), 326-338. doi:10.1037/0022-3514.56.3.326

Jamet, E. (2014). An eye-tracking study of cueing effects in multimedia learning. *Computers in Human Behavior, 32*, 47-53.

Joao, R. S., Marcelo, Q. H., Efujita, A., & Luis, A. R. (2012). Evaluation of pattern recognition and feature extraction methods in ADHD prediction. *Frontiers in Systems Neuroscience, 6*(2012) doi:10.3389/fnsys.2012.00068

Johnson, Elizabeth K., & Tyler, Michael D. (2010). Testing the Limits of Statistical Learning for Word Segmentation. *Developmental Science*, 13(2), 339-345.

Jones, R., Tarpey, T., Hamo, A., Carberry, C., Brouwer, G., & Lord, C. (2018). Statistical learning is associated with autism symptoms and verbal abilities in young children with autism. *Journal of Autism and Developmental Disorders, 48*(10), 3551-3561. doi:10.1007/s10803-018-3625-7

Jones, W., & Klin, A. (2013). Attention to eyes is present but in decline in 2-6-month-old infants later diagnosed with autism. *Nature, 504*(7480), 427-31.

Kaitlyn, M. A. P., & Stevenson, R. A. (2018). Auditory and visual statistical learning are not related to ADHD symptomatology: Evidence from a research domain criteria (RDoC) approach. *Frontiers in Psychology, 9* doi:10.3389/fpsyg.2018.02502

Kihlstrom, J. (1994). *Implicit learning and tacit knowledge - an essay on the cognitive unconscious. - reber,as*

Kirkham, Natasha Z., Slemmer, Jonathan A., & Johnson, Scott P. (2002). Visual Statistical Learning in Infancy: Evidence for a Domain General Learning Mechanism. *Cognition*, 83(2), B35-B42.

Koc-Januchta, M., Höffler, T., Thoma, G., Prechtl, H., & Leutner, D. (2017). Visualizers versus verbalizers: Effects of cognitive style on learning with texts and pictures - An eye-tracking study. *Computers in Human Behavior, 68*, 170-179.

Lange, B., Hunfalvay, M., Murray, N., Roberts, C. M., & Bolte, T. (2018). Reliability of computerized eye-tracking reaction time tests in non-athletes, athletes, and individuals with traumatic brain injury. Optometry & Visual Performance, 6(3).

Lee, J. (2017). Time Course of Lexicalization During Sentence Production in Parkinson's Disease: Eye-Tracking While Speaking. *Journal of Speech, Language, and Hearing Research: JSLHR, 60*(4), 924-936.

Lee, J. C., Beesley, T., & Livesey, E. J. (2016). Sequential effects and sequence learning in a three-choice serial reaction time task. Acta Psychologica, 170, 168–176. https://doi.org/10.1016/j.actpsy.2016.08.004

Levy, A., 2013. Three kinds of new mechanism. *Biology & Philosophy*, 28(1), pp.99–114. 10.1007/s10539-012-9337-z.

Lew-Williams, C., & Saffran, J. R. (2012). All words are not created equal: Expectations about word length guide infant statistical learning. *Cognition, 122*(2), 241-246. doi:10.1016/j.cognition.2011.10.007

Liang, C., Tsai, J., & Hsu, W. (2017). Sustained visual attention for competing emotional stimuli in social anxiety: An eye tracking study. *Journal of Behavior Therapy and Experimental Psychiatry, 54*, 178-185.

Lieberman, M. D., Chang, G. Y., Chiao, J., Bookheimer, S. Y., & Knowlton, B. J. (2004). An event-related fMRI study of artificial grammar learning in a balanced chunk strength design. *Journal of Cognitive Neuroscience, 16*(3), 427-438. doi:10.1162/089892904322926764

Lisa, M. O., Heidi, A. B., Frederick, S. B., Ian, M. M., & Steven, J. L. (2013). Developmental changes in visual short-term memory in infancy: Evidence from eye-tracking. *Frontiers in Psychology, 4* doi:10.3389/fpsyg.2013.00697

Machamer, P., Darden, L., & Craver, C. F. (2000). Thinking about mechanisms. *Philosophy of Science, 67*(1), 1-25. doi:10.1086/392759

Mackenzie, K. J., Aslin, R. N., & Fiser, J. (2011). The interaction between chunking and stimulus complexity in infant visual statistical learning. *Journal of Vision, 11*(11), 459. doi:10.1167/11.11.459

Magrelli, Jermann, Noris, Ansermet, Hentsch, Nadel, & Billard. (2013). Social orienting of children with autism to facial expressions and speech: A study with a wearable eye-tracker in naturalistic settings. *Frontiers in Psychology, 4*, 840.

Manning, C., & Schutze, H. (1999). Foundations of statistical natural language processing. MIT press.

McCauley, S. M., & Christiansen, M. H. (2011). Learning simple statistics for language comprehension and production: The CAPPUCCINO model. In Proceedings of the annual meeting of the cognitive science society (Vol. 33, No. 33).

McCullagh, P. and Nelder, J.A. (1989) Generalized Linear Models. 2nd Edition, Chapman and Hall, London. http://dx.doi.org/10.1007/978-1-4899-3242-6

Miller, G. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *The psychological review*, 63, 81-97.

Misyak, J., & Christiansen, M. (2012). Statistical Learning and Language: An Individual Differences Study. Language Learning, 62(1), 302-331.

Monaghan, P., & Christiansen, M. H. (2008). Integration of multiple probabilistic cues in syntax acquisition. Trends in corpus research: Finding structure in data, 139-63.

Monaghan, P., Schoetensack, C., & Rebuschat, P. (2019). A single paradigm for implicit and statistical learning. *Topics in Cognitive Science, 11*(3), 536-554. doi:10.1111/tops.12439

Murphy, C., & Connaughton, E. (2017). M37. Automatic Visual Processing of Social Information in Schizophrenia. *Schizophrenia Bulletin, 43*(Suppl1), S225.

Navarro, O., Gonzalez, A., & Molina, A. (2018). Experience of use of eye tracking technology with children who have attention problems. *2018 International Symposium on Computers in Education (SIIE),* 1-6.

Newport, E. L., Hauser, M. D., Spaepen, G., & Aslin, R. N. (2004). Learning at a distance II. statistical learning of non-adjacent dependencies in a non-human primate. *Cognitive Psychology, 49*(2), 85-117. doi:10.1016/j.cogpsych.2003.12.002

Newport, E., & Aslin, R. N. (2004). Learning at a distance I. statistical learning of non-adjacent dependencies. *Cognitive Psychology, 48*(2), 127-162. doi:10.1016/S0010-0285(03)00128-2

Newport, Hauser, Spaepen, & Aslin. (2004). Learning at a distance II. Statistical learning of non-adjacent dependencies in a non-human primate. *Cognitive Psychology*,49(2), 85-117.

Nilsson Benfatto, M., Öqvist Seimyr, G., Ygge, J., Pansell, T., Rydberg, A., Jacobson, C., & Lappe, M. (2016). Screening for Dyslexia Using Eye Tracking during Reading. *PLoS ONE, 11*(12), E0165508.

Nissen, M. J., & Bullemer, P. (1987). Attentional requirements of learning: Evidence from performance measures. *Cognitive Psychology, 19*(1), 1–32. https://doi.org/10.1016/0010-0285(87)90002-8

Norman, D. (1970). Models of Memory. New York: Academic Press. Niemark, E.D. & Estes, W.K. (1967). Stimulus Sampling Theory. San Francisco: Holden-Day.

Opris, I., Lebedev, M., & Nelson, R. (2011). Motor planning under unpredictable reward: Modulations of movement vigor and primate striatum activity. *Frontiers in Neuroscience, 5* doi:10.3389/fnins.2011.00061

Orena, Theodore, & Polka. (2015). Language exposure facilitates talker learning prior to language comprehension, even in adults. *Cognition*, 143, 36-40.

Pascual-Leone A, Grafman J, Clark K, Stewart M, Massaquoi S, Lou JS, Hallett M. Procedural learning in Parkinson's disease and cerebellar degeneration. Ann Neurol. 1993 Oct;34(4):594-602. doi: 10.1002/ana.410340414. PMID: 8215247

Perruchet, P. (2018). What Mechanisms Underlie Implicit Statistical Learning? Transitional Probabilities Versus Chunks in Language Learning. *Topics in Cognitive Science, Topics in cognitive science*, 19 December 2018.

Perruchet, P., & Desaulty, S. (2008). A role for backward transitional probabilities in word segmentation? *Memory & Cognition, 36*, 1299-305. doi:10.3758/MC.36.7.1299

Perruchet, P., & Pacton, S. (2006). Implicit learning and statistical learning: One phenomenon, two approaches. Trends in cognitive sciences, 10(5), 233-238.

Perruchet, P., & Vinter, A. (1998). PARSER: A Model for Word Segmentation. *Journal of Memory and Language,* 39(2), 246-263.

Pinker, S. (1979). Formal models of language learning. Cognition, 7(3), 217-283.

Pinker, S. (1999). Words and Rules. New York. NY: Basic Books.

Pinker, S., & Jackendoff, R. (2005). The faculty of language: what's special about it?. Cognition, 95(2), 201-236.

Pinker, S., & Prince, A. (1988). On language and connectionism: Analysis of a parallel distributed processing model of language acquisition. Cognition, 28(1-2), 73-193.

Polyanskaya, L. (2021). Cognitive mechanisms of statistical learning and segmentation of continuous sensory input. Memory & Cognition, 1-18.

Prasada, S., & Pinker, S. (1993). Generalisation of regular and irregular morphological patterns. Language and cognitive processes, 8(1), 1-56.

Protopapas, A., Mitsi, A., Koustoumbardis, M., Tsitsopoulou, S. M., Leventi, M., & Seitz, A. R. (2017). Incidental orthographic learning during a color detection task. Cognition, 166, 251-271. doi:https://doi.org/10.1016/j.cognition.2017.05.030

Purves D, Augustine GJ, Fitzpatrick D, et al., editors. Neuroscience. 2nd edition. Sunderland (MA): Sinauer Associates; 2001. Types of Eye Movements and Their Functions. Available from: https://www.ncbi.nlm.nih.gov/books/NBK10991/

Rabe-Hesketh, S., Skrondal, A., & Pickles, A. (2004). Generalized multilevel structural equation modeling. *Psychometrika, 69*(2), 167-190. doi:10.1007/BF02295939

Rabiner, L., & Juang, B. (1986). An introduction to hidden Markov models. ieee assp magazine, 3(1), 4-16.

Ratnaparkhi, A. (1997). A simple introduction to maximum entropy models for natural language processing. IRCS Technical Reports Series, 81.

Reber, A. (1969). Transfer of syntactic structure in synthetic languages. *Journal of Experimental Psychology, 81*(1), 115-119. doi:10.1037/h0027454

Reber, A. (1989). Implicit learning and tacit knowledge. *Journal of Experimental Psychology-General, 118*(3), 219-235. doi:10.1037/0096-3445.118.3.219

Reber, P. J., Batterink, K., & Reuveni, B. (2019). Implicit Learning: History and applications. In A. Cleeremans, V. Allakhverdov, & M. Kuvaldina (Eds.), Implicit Learning: 50 Years

Reber, P. J., Batterink, K., & Reuveni, B. (2019). Implicit Learning: History and applications. In A. Cleeremans, V. Allakhverdov, & M. Kuvaldina (Eds.), Implicit Learning: 50 Years On. London Routledge https://doi.org/10.4324/9781315628905-2

Rescorla, R. A. (1988). Behavioral studies of pavlovian conditioning. *Annual Review of Neuroscience, 11*(1), 329-352. doi:10.1146/annurev.ne.11.030188.001553

Rice, K., Moriuchi, J., Jones, W., & Klin, A. (2012). Parsing Heterogeneity in Autism Spectrum Disorders: Visual Scanning of Dynamic Social Scenes in School-Aged

Children. *Journal of the American Academy of Child and Adolescent Psychiatry, 51*(3), 238-248.

Richmond, J., & Nelson, C. A. (2009). Relational memory during infancy: Evidence from eye tracking. *Developmental Science, 12*(4), 549. doi:10.1111/j.1467-7687.2009.00795.x

Robertson, E. M. (2007). The serial reaction time task: implicit motor skill learning?. Journal of Neuroscience, 27(38), 10073-10075.

Robinson, A. J., & Pascalis, O. (2004). Development of flexible visual recognition memory in human infants. *Developmental Science, 7*(5), 527.

Roper, Z. J. J., Vecera, S. P., & Vaidya, J. G. (2014). Value-driven attentional capture in adolescence. *Psychological Science, 25*(11), 1987-1993. doi:10.1177/0956797614545654

Rovee-Collier, C. (1999). Infants memory processing of a serial list: List length effects. *Journal of Experimental Child Psychology, 73*(4), 72-91. doi:10.1016/S0022-0965(02)00169-8

Rovee-Collier, C., Griesler, P. C., & Earley, L. A. (1985). Contextual determinants of retrieval in three-month-old infants. *Learning and Motivation, 16*(2), 139-157. doi:10.1016/0023-9690(85)90009-8

Rutledge, K., Schweitzer, Julie B., Guyer, Amanda, & Young, Gregory. (2010). *Attention to Reward Information Preceding Decision Making in Adults with Attention-Deficit / Hyperactivity Disorder: An Eye-Tracking Study,* ProQuest Dissertations and Theses.

Saffran, J. R. (2002). Constraints on statistical language learning. Journal of Memory and Language, 47, 172-196.

Saffran, J. R., Johnson, E. K., Aslin, R. N., & Newport, E. L. (1999). Statistical learning of tone sequences by human infants and adults. *Cognition, 70*(1), 27-52. doi:10.1016/S0010-0277(98)00075-4

Saffran, J., Aslin, R., & Newport, E. (1996). Statistical learning by 8-month-old infants. Science (New York, N.Y.), 274(5294), 1926-8.

Saffran, J.R. and Thiessen, E.D. (2007) Domain-general learning capacities. In Blackwell Handbook of Language Development (Hoff,E. and Shatz, M., eds), pp. 68–86, Wiley

Sanchez, D. J., & Reber, P. J. (2012). Operating characteristics of the implicit learning system supporting serial interception sequence learning. *Journal of Experimental Psychology: Human Perception and Performance, 38*(2), 439–452. https://doi.org/10.1037/a0026347

Santolin, C., & Saffran, J. R. (2018). Constraints on statistical learning across species. *Trends in Cognitive Sciences*, *22*(1), 52-63.

Santolin, C., & Saffran, J. R. (2018). Constraints on Statistical Learning Across Species. Trends in cognitive sciences, 22(1), 52–63. https://doi.org/10.1016/j.tics.2017.10.00

Schmalz, X., Altoè, G., & Mulatti, C. (2017). Statistical learning and dyslexia: A systematic review. *Annals of Dyslexia, 67*(2), 147-162. doi:10.1007/s11881-016-0136-0

Servan-Schreiber, D., Cleeremans, A., & McClelland, J. L. (1991). Graded state machines: The representation of temporal contingencies in simple recurrent networks. Machine Learning, 7, 161-193.

Servan-Schreiber, E., & Anderson, J. R. (1990). Learning artificial grammars with competitive chunking. Journal of Experimental Psychology: Learning, Memory, and Cognition, 16(4), 592.

Sibley, C., Foroughi, C., Brown, N., & Coyne, J. (2018). Low Cost Eye Tracking: Ready for Individual Differences Research? *Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 62*(1), 741-745.

Siegelman, N., Bogaerts, L., & Frost, R. (2017). Measuring individual differences in statistical learning: Current pitfalls and possible solutions. Behavior Research Methods, 49(2), 418-432.

Siegelman, N., Bogaerts, L., Elazar, A., Arciuli, J., & Frost, R. (2018). Linguistic entrenchment: Prior knowledge impacts statistical learning performance. *Cognition, 177*, 198-213. doi:10.1016/j.cognition.2018.04.011

Slone, L. K., & Johnson, S. P. (2015). Infants' statistical learning: 2-and 5-month-olds' segmentation of continuous visual sequences. Journal of experimental child psychology, 133, 47-56.

Slone, L. K., & Johnson, S. P. (2018). When learning goes beyond statistics: Infants represent visual sequences in terms of chunks. *Cognition, 178*, 92-102. doi:10.1016/j.cognition.2018.05.016

Spiegel, R., & Mclaren, I. P. L. (2006). Associative sequence learning in humans. *Journal of Experimental Psychology.Animal Behavior Processes, 32*(2), 150. doi:10.1037/0097-7403.32.2.150

Stadler, M., & Neely, C. (1997). Effects of sequence length and structure on implicit serial learning. *Psychological Research, 60*(1-2), 14-23. doi:10.1007/BF00419677

Steele, F. (2008a). Book review: Skrondal A and rabe-hesketh S 2004: Generalized latent variable modelling: Multilevel, longitudinal and structural equation models. boca raton, FL: Chapman & hall)CRC. 508 pp £50 (HB), ISBN 1-58488-000-7. *Stat Methods Med Res, 17*(1), 119-120. doi:10.1177/09622802080170010702

Tenev, A., Markovska-Simoska, S., Kocarev, L., Pop-Jordanov, J., Müller, A., & Candrian, G. (2014). Machine learning approach for classification of ADHD adults. *International Journal of Psychophysiology, 93*(1), 162-166. doi:10.1016/j.ijpsycho.2013.01.008

Thiessen, E. D. (2017). What's statistical about learning? insights from modelling statistical learning as a set of memory processes. *Philosophical Transactions of the Royal Society B: Biological Sciences, 372*(1711) doi:10.1098/rstb.2016.0056

Thiessen, E., & Pavlik, P. (2013). IMinerva: A Mathematical Model of Distributional Statistical Learning. *Cognitive Science*, 37(2), 310-343.

Toh, W., Castle, D., & Rossell, S. (2015). Facial affect recognition in body dysmorphic disorder versus obsessive-compulsive disorder: An eye-tracking study. *Journal of Anxiety Disorders, 35*, 49-59.

Tomasello, M. (2003). Constructing a language: A usage-based theory of language acquisition. Cambridge, MA: HUP.

Tremblay, A. (2011). Learning to parse liaison-initial words: An eye-tracking study *. *Bilingualism: Language and Cognition, 14*(3), 257-279.

Turk-Browne, N., Scholl, B. J., Chun, M. M., & Johnson, M. K. (2009). Neural evidence of statistical learning: Efficient detection of visual regularities without awareness. *Journal of Cognitive Neuroscience, 21*(10), 1934-1945. doi:10.1162/jocn.2009.21131

Van Witteloostuijn, M., Lammertink, I., Boersma, P., Wijnen, F., & Rispens, J. (2019). Assessing Visual Statistical Learning in Early-School-Aged Children: The Usefulness of an Online Reaction Time Measure. *Frontiers in psychology*, *10*, 2051. https://doi.org/10.3389/fpsyg.2019.02051

Vitz, P. C., & Todd, T. C. (1969). A coded element model of the perceptual processing of sequential stimuli. *Psychological Review, 76*(5), 433-449. doi:10.1037/h0028113

Vivanti, G., Trembath, D., & Dissanayake, C. (2014). Atypical monitoring and responsiveness to goal-directed gaze in autism spectrum disorder. *Experimental Brain Research, 232*(2), 695-701.

Wächter, T., Lungu, O. V., Liu, T., Willingham, D. T., & Ashe, J. (2009). Differential effect of reward and punishment on procedural learning. *The Journal of Neuroscience : The Official Journal of the Society for Neuroscience, 29*(2), 436. doi:10.1523/JNEUROSCI.4132-08.2009

Walker, J. A. (2018). *Applied Statistics for Experimental Biology*. https://www.middleprofessor.com/files/applied-biostatistics_bookdown/_book/

Wang, F. H., Zevin, J. D., Trueswell, J. C., & Mintz, T. H. (2020). Top-down grouping affects adjacent dependency learning. *Psychonomic Bulletin & Review*, *27*, 1052-1058.

Wang, F. H., Zevin, J., & Mintz, T. H. (2019). Successfully learning non-adjacent dependencies in a continuous artificial language stream. *Cognitive Psychology*, *113*, 101223.

Wickham et al., (2019). Welcome to the Tidyverse. Journal of Open Source Software, 4(43), 1686, https://doi.org/10.21105/joss.01686

Wu, Y., Cristino, F., Leek, C., & Thierry, G. (2013). Non-selective lexical access in bilinguals is spontaneous and independent of input monitoring: Evidence from eye tracking. *Cognition, 129*(2), 418-425.

Yang, C., Crain, S., Berwick, R. C., Chomsky, N., & Bolhuis, J. J. (2017). The growth of language: Universal Grammar, experience, and principles of computation. Neuroscience & Biobehavioral Reviews, 81, 103-119.

Zhao, C., & Vogel, E. (2022). Visual long-term memory guides attentional selection during serial reaction time task. Journal of Vision, 22(14), 3226-3226.

Zhao, F., Gaschler, R., Nöhring, D. O., Röttger, E., & Haider, H. (2020). Sequential modulation of across-task congruency in the serial reaction time task. Acta Psychologica, 205, 103043.

Appendix A – Study Participation Documents

A.1 Information Sheet and Consent Form.

Department of Psychology
School of Social Sciences
50 Shakespeare Street, Nottingham, NG1 4FQ. Tel. +44(0)115 941 8418 www.ntu.ac.uk/s3

Participant Information Sheet
Explaining individual differences in associative learning ability.

Dear Potential Participant,

Would you like to participate in a study about associative learning? It is already known that we acquire knowledge about the world by extracting patterns and regularities from our environment. This ability to extract patterns and associate two distinguished events or objects is referred to as associative learning. Plenty of studies suggest that the rate of associative learning is increasing during development as basic cognitive mechanisms such as attention and memory improve. However, there are contradicting evidence suggesting that apart from those mechanisms, other factors such as exposure and familiarity of stimuli are influencing the associative learning ability. This research seeks to investigate exposure and general domain processes during development.

We have a short set of tasks that will enable us to properly assess whether or not associative learning increases with age, familiarity and exposure, and these tasks will take approximately 10-15 minutes. The tasks are similar to one another and involve several presentations of items (e.g., familiar shapes), with participants simply being asked to look at where the item appears on the screen. These tasks have been used widely with adults and should not be more stressful than any task the participants experience in their everyday life. Extra forms with demographic information and vocabulary checklists will be obtained. Our researchers are well trained and are able to identify children's emotion changes. In case your child is distressed, though very unlikely, the session will be terminated immediately, and s/he will be comforted and reassured. Child eye-movements will be recorded for later data analysis.

You will participate in the study if you give your consent. In case you become tired or uninterested, we will stop the session or take a break. Participation is entirely voluntary, you are free to terminate your participation and withdraw from the project at any time until 15 days after the date of session without prejudice to you.

All data will be anonymised and confidential with all children being given a unique identifier (e.g., P26) rather than any names being used. You will therefore *not* be able to be identified by name in any of the data we will hold[1]. All data will be stored securely in locked cabinet and password protected computers and only the research team will have access to them. All data will be destroyed within 5 years after research publication.

If you are interested in helping with this study, please sign the 'I wish to participate' form below. This study is undertaken by Sofia Tsitsopoulou (Nottingham Trent University) under the supervision of Dr. Gary Jones, Dr. Mark Torrance, and Dr. Kate Ellis-Davies (all Nottingham Trent University). If you have any questions about the research, please feel free to contact us at 0115 8482422 or gary.jones@ntu.ac.uk.

Thank you very much for your support for this study.

Sincerely yours,

_____
Department of Psychology
Nottingham Trent University

[1] [1] Until the withdrawal cut-off date, we will keep a securely-held list of child names alongside their identifiers in case you may wish to withdraw your child's data. This list will be kept in a locked cabinet and will be destroyed after the withdrawal cut-off date.

NOTTINGHAM TRENT UNIVERSITY

Department of Psychology
School of Social Sciences
50 Shakespeare Street, Nottingham, NG1 4FQ. Tel. +44(0)115 941 8418 www.ntu.ac.uk/s3

Consent Form

Explaining individual differences in associative learning ability.

I wish to participate

Participant's name:_____ Participant's Sex:_____M / F_____

Participant's date of birth (dd/mm/yyyy):_____

1. I consent to participate in the above project, the particulars of which have been explained to me.

2. I acknowledge that:

(a) the possible effects of the tasks or procedures have been explained to me to my satisfaction;

(b) The project is for the purpose of research and not for treatment;

(c) I have read and understood the information sheet for this study;

(d) I have been informed that all data will be treated confidentially and that any information that participants provide will be safeguarded subject to any legal requirements;

Unique identifier:_____

Participant's signature:_____

Signature of researcher: _____

Date:_____

NOTTINGHAM TRENT UNIVERSITY

A.2 Debrief Form.

Department of Psychology
School of Social Sciences
50 Shakespeare Street, Nottingham, NG1 4FQ. Tel. +44(0)115 941 8418 www.ntu.ac.uk/s3

Debrief Form

Associative Learning Ability & Vocabulary Growth

Thank you for participating in this study! We hope you enjoyed the experience. This form provides background about our research and will help you understand better the purposes of the tasks.

The aim of this study was to investigate the associative learning mechanisms that support the language learning process. Infants and children learn language quickly, easily, and without effort or formal teaching. A crucial mechanism that supports language learning is associative learning. Within the literature it is suggested that associative learning may operate generally (i.e., regardless of whether the stimuli are language-related or not, one's learning ability remains constant) or it may operate within specific domains (i.e., one's ability to learn within the language domain is different to that of other domains).

On the one hand, general domain associative learning refers to the ability of extracting and learning regularities/ patterns from the sensory input, while on the other hand, language specific associative learning refers to the ability of extracting verbal regularities from the auditory modality. In addition, vocabulary learning is influenced by language exposure. However, language exposure (i.e., the amount of language one hears) inevitably increases with age because children hear language on a daily basis and therefore with age there is more opportunity to learn. This means that in terms of associative learning, it is important to examine performance over time because learning and exposure are inextricably linked to one another. The tasks that you were involved with, included general domain and language specific domain processes that will be used to explain developmental differences in associative learning ability and vocabulary growth within different age groups and different exposure levels.

As you know, your participation in this study is voluntary. If you so wish, you may withdraw after reading this debriefing form, at which point all records of your participation will be destroyed. You will not be penalized if you withdraw. You also have right to withdraw your data, 15 days after the data collection date. If that's the case, feel to contact me at sofia.tsitsopoulou@ntu.ac.uk. Also, for any further questions don't hesitate to contact us at 0115 8482422 or at sofia.tsitsopoulou@ntu.ac.uk.

NOTTINGHAM TRENT UNIVERSITY

Appendix B – Python Experimental scripts

B.1 Experimental Set Up Script.

```
from __future__ import division
from psychopy import visual, sound, core
from psychopy.iohub.client import launchHubServer
from collections import OrderedDict as od
import pandas as pd

import os
import sys

sys.dont_write_bytecode = True # just during testing


class alearn(object):
    '''
    sets up window and provides stimuli for Sophia's associative
    learning (alearn) experiments
    '''

    def __init__(self, session_info, outfile, tracker_type = 'mouse'):
        '''tracker_type must be mouse, SR or GP3'''


        self.tracker_type = tracker_type
        self.session_info = session_info
        self.outfile = outfile

        self.quitnow = False

        self.io_config_files = {'mouse':'configs/iohub_config.yaml',
                                'SR':'configs/iohub_config_sr_1.yaml',
                                'GP3':'configs/iohub_config_gp3.yaml'}


        # launch iohub and set devices
        self.io_config = {'experiment_code':'alearn',
                          'session_info': self.session_info,
                          'datastore_name':self.outfile,
                          'iohub_config_name':
self.io_config_files[self.tracker_type]}

        self.io = launchHubServer(**self.io_config)

        self.keyboard = self.io.devices.keyboard
        self.display = self.io.devices.display

        if self.tracker_type == 'mouse':
            self.mouse = self.io.devices.mouse
        else:
            self.tracker = self.io.devices.tracker

        self.stimdir = "stimuli"

        self.win = 'none'

        self.winsize = (0,0)
```

```
56
57          self.showAOI = False #set to false from run
58
59          # globals used for playing sequences of sounds
60          # these are initialised with get_sounds()
61          self.si = 0
62          self.sstart = 'none'
63
64          self.gazedot_visible = False
65
66
67          # this defines various different sets of locations
68          # the keys must correspond to the location lables used in the trial
69  file
70          # The values below are just for testing. Normally created at
71  runtime with
72          # the get_locations function below.
73          self.locations_data = od([
74                  ('A', ((-.3,.3),
75                         (0,0),
76                         (0,.3)))
77                  ,('B',((-.3,.3),
78                         (0,0),
79                         (0,.3),
80                         (-.1,.2)))
81                  ])
82
83
84      def setup(self, background_colour = (0,0,0)):
85          '''
86          get a window and create some display object parameters
87          '''
88
89          # launch window, automatically detecting full screen display size
90          self.win = visual.Window(pos = (0,0),
91                                   units = 'pix',
92                                   color = background_colour,
93                                   fullscr=True,
94                                   allowGUI = False)
95
96          print "Window size: %d by %d"%(self.win.size[0],self.win.size[1])
97
98
99          #just for testing
100         # al.locations should be specified at runtime using
101         # al.locations =
102 al.location_sets(al.win.size[1],al.locations_data)[k]
103         # here k is a key from al.locations.data
104 #        y = self.win.size[1]
105 #        self.locations = od([
106 #                            (1, (-y/3,y/3)),
107 #                            (2, (0,y/3)),
108 #                            (3, (y/3, y/3)),
109 #                            (4, (-y/3,0)),
110 #                            (5, (0,0)),
111 #                            (6, (y/3,0)),
112 #                            (7, (-y/3,-y/3)),
113 #                            (8, (0,-y/3)),
114 #                            (9, (y/3,-y/3))
115 #                            ])
116
```

```
117             gazedot_opacity = 1
118             if self.gazedot_visible:
119                 gazedot_opacity = 1
120 
121             self.gazedot = visual.GratingStim(self.win, tex=None,mask='gauss',
122                             pos=[0, 0],size=[25,25],
123                             opacity=gazedot_opacity, units='pix')
124 
125 
126         def get_locations_data(self, loc_filename, sheet = 'locs1'):
127             '''takes excel file
128             loc_filename = path. Sheet has columns for set-label, x, y
129             '''
130             Df =  pd.read_excel(loc_filename,sheet_name = sheet,
131                 na_values = [], keep_default_na = False)
132 
133             dct = od()
134 
135             for a,c,d in zip(Df.location_set,Df.x,Df.y):
136                 dct.setdefault(a,[])
137                 dct[a].append((c,d))
138 
139             l = ((i[0],tuple(i[1])) for i in dct.items())
140 
141             dct = od(l)
142 
143             for k in dct.keys():
144                 dct[k] = sorted(dct[k], key = lambda x: (x[1]), reverse = True)
145 
146             return dct
147 
148         def location_sets(self, y, locdat):
149             '''returns dictionary of different sets of locations
150             locdat -- locations data dictionary
151             y -- display height
152             '''
153 
154             def get_pixel_coords(y,locs):
155                 '''returns dict of x,y screen coordinates/n
156                 y -- display height (for win.size)/n
157                 locs = list of x,y coordinates expressed as proportion of
158     screen height
159                 varying from -.5 to +.5 because locations are centre-
160     anchored'''
161 
162                 d = od([])
163                 i = 1
164                 for loc in locs:
165                     d[i] = (loc[0]*y,loc[1]*y)
166                     i += 1
167                 return d
168 
169             d = od([])
170             for k in locdat.keys():
171                 d[k] = get_pixel_coords(y,locdat[k])
172 
173             return d
174 
175         def get_locations(self, loc_filename, sheet = 'locs1'):
176             '''winy = display height'''
```

```
177             return
178     self.location_sets(self.win.size[1],self.get_locations_data(loc_filename,
179     sheet))
180 
181         def image_stim(self,imagefilename, size = 1/12, position = (0,0)):
182             """
183             Create PsychoPy image objects.
184 
185             imagefilename -- file name for image (has to be png?)
186 
187             size -- x and y multiplier (one value), as proportion of screen
188     width.
189             Image will be forced to square
190             """
191             winx,winy = self.win.size
192 
193             x = winx*size
194             y = (winy*size*(winx/winy))
195 
196             im = visual.ImageStim(self.win, units='pix'
197                                   , image =
198     os.path.join('.',self.stimdir,imagefilename)
199                                   , size = (x,y)
200                                   , pos = position
201                                   )
202 
203             return im
204 
205         def get_aoi(self, imx, imy, scale = 2, ratio = 1):
206 
207             """
208             Create rectangle around image (probably).
209             Visible if alearn.showAOI == True
210 
211             imx,imy -- dimensions of associated image or row of images, in
212     pixels
213 
214             scale -- multiplier on y of image (e.g. 2 gives aoi twice as high
215     as image)
216 
217             ratio -- proportion of (y aoi - y image) to add to x of aoi
218             """
219 
220             y = imy*scale
221             x = imx + (y-imy)*ratio
222 
223             return visual.Circle(self.win, radius=0.5, edges=32,
224                                  units='pix',
225                                  size = (x,y),
226                                  lineWidth = 1,
227                                  opacity = int(self.showAOI))
228 
229 
230         def get_sounds(self,soundfilename):
231             """
232             initialises sound sequence play globals
233             returns one or more psychopy sound stim
234             """
235 
236             self.sstart = core.getTime()
237             self.si = 0
```

```
238
239             sounds = []
240
241             if 'none' not in soundfilename:
242                 if type(soundfilename) in (str, unicode):
243                     soundfilename = [soundfilename]
244
245                 sounds = [sound.Sound(os.path.join('.',self.stimdir,sname)) for
246     sname
247                                       in soundfilename]
248
249             return sounds
250
251
252         def q(self):
253             '''close gracefully'''
254             self.win.close()
255             self.io.quit()
256
257         def ioq(self):
258             '''quit just iohub server - useful when testing'''
259             self.io.quit()
260
261
262         def escaped(self):
263             '''test for quit key'''
264             if 'escape' in self.keyboard.getKeys():
265                 self.io.sendMessageEvent("escape key pressed to quit",
266                                          category = 'Experiment')
267                 print "Escape key pressed to quit at %.3f\n"%(core.getTime())
268                 return True
269
270
271         def set_image_aoi(self,target_image, foil_image, target_location,
272                           image_size = 1/12, aoi_scale = 2, aoi_ratio = 1):
273             """
274             Places an image at each of the locations in self.locations
275             target_image is placed at target_location
276             creates aoi image around target
277             foil is placed at each of the other locations defined in
278             self.locations
279
280             target_image -- target image filename\n
281             foil_image -- foil (non target) image filename\n
282             target_location -- must be integer and be a key in self.locations\n
283             image_size -- a screen size multiplier \n
284             aoi_scale -- aoi proportion of image height \n
285             aoi_ratio -- proportion of aoi height to add to width
286
287             returns list of psychopy images objects and an aoi image
288             with correct positions already set
289
290             """
291
292             # get list of positions
293             positions = self.locations.values()
294
295             # make list of image files
296             image_files = [foil_image for p in positions]
297
298             # set image file at target position
```

```
299             image_files[positions.index(self.locations[target_location])] =
300     target_image
301
302             # make psychopy images
303             images = [self.image_stim(im,
304                                       size = image_size,
305                                       position = p) for (p,im) in
306     zip(positions,image_files)]
307
308             # make aoi
309             imx,imy =
310     images[positions.index(self.locations[target_location])].size
311
312             aoi = self.get_aoi(imx, imy, scale = aoi_scale, ratio = aoi_ratio)
313
314             aoi.pos = self.locations[target_location]
315
316             return images, aoi
317
318
319         def draw_stim(self, images, aoi, locs = [], sounds = [], sound_interval
320     = 0, draw_image = True):
321             '''
322             Draws an psychopy image or images, optionally modifying their
323     location
324             Draws gazedot and aoi.
325             Optionally plays one or more sounds, with sound_interval pause
326     between each sound.\n
327
328             This is not called directly by the experiment run script but used
329     by the
330             display functions below.
331             '''
332
333             self.quitnow = self.escaped() # check for escape key
334
335             if self.tracker_type != 'mouse':
336                 if self.tracker.getLastGazePosition():
337                     gpos = self.tracker.getLastGazePosition() # returns 202 if
338     runtime gaze location not supported
339                 else:
340                     gpos = (-999,-999)
341                     print 'could not get tracker last position'
342             else:
343                 gpos = self.mouse.getPosition()
344
345             self.gazedot.pos = gpos
346             gaze_in_aoi = False
347
348             # draw stuff
349             if draw_image:
350                 if locs:
351                     for image,loc in zip(images,locs):
352                         image.pos = loc
353                         image.draw()
354                         #print "image dimensions (draw stim) = ",image.size
355                 else:
356                     for image in images:
357                         image.draw()
358
359             aoi.draw()
```

```
360
361         self.gazedot.draw()
362
363         #play sequence of sounds
364         if sounds:
365             if core.getTime() >= self.sstart and self.si < len(sounds):
366                 sounds[self.si].play()
367                 print "%d, %s played at %.3f"%(self.si,
368 sounds[self.si].fileName, core.getTime())
369                 self.sstart += sound_interval
370                 self.si += 1
371
372
373         gaze_in_aoi = self.gazedot.overlaps(aoi)
374
375         flip_time = self.win.flip()
376
377         return {'gaze_overlap': gaze_in_aoi,
378                 'time':flip_time,
379                 'gaze_position': gpos}
380
381
382     def timed_display(self,imagefilename = ['foil.png','foil.png'],
383                       loc = 1,
384                       locs = [],
385                       image_size = 1/12, #proportion of display width
386                       aoi_scale = 2, #aoi height = image height * aoi_scale
387                       aoi_ratio = 1, #porportion of aoi extra height to add to
388 aoi width
389                       dur = 2.000, #duration
390                       label = 'timed_display',
391                       soundfilename = [],
392                       sound_interval = 1.000, #time between sounds in more than
393 one (s)
394                       draw_nothing = False,
395                       blank_on_exit = True):
396
397         '''
398         Shows a shape (and aoi) for a period dur at location with keyword
399 loc
400         imagefilename can be list of filenames or a single string
401         '''
402
403         out = od([])   # dictionary to collect variable to output
404
405         if not self.quitnow:
406
407             foilpng,targetpng = imagefilename
408
409             images,aoi = self.set_image_aoi(targetpng, foilpng, loc,
410                       image_size = image_size, aoi_scale = aoi_scale,
411 aoi_ratio = aoi_ratio)
412
413             sounds = []
414             if soundfilename:
415                 sounds = self.get_sounds(soundfilename)
416
417             flip = self.draw_stim(images, aoi, sounds, sound_interval,
418 locs,
419                                   draw_image = not draw_nothing)
420
```

```
421             onset = flip['time']
422             gaze_pos = flip['gaze_position']
423 
424             out['onset_time'] = onset
425             out['onset_gaze_pos'] = gaze_pos
426 
427             while core.getTime() - onset <= dur and not self.quitnow:
428                 flip = self.draw_stim(images, aoi, sounds, sound_interval,
429 locs,
430                                       draw_image = not draw_nothing)
431 
432             if blank_on_exit:
433                 self.win.flip()
434 
435             out['end_time'] = flip['time']
436             out['end_gaze_pos'] = flip['gaze_position']
437 
438             print '%s onset = %.3f'%(label,onset)
439 
440             return out
441 
442 
443     def gaze_contingent_display(self,imagefilename = ['foil.png',
444 'target.png'],
445                                 label = 'gaze contingent display',
446                                 loc = 1,
447                                 locs = [], # see note
448                                 image_size = 1/12, #proportion of display
449 width
450                                 aoi_scale = 2, #aoi height = image height *
451 aoi_scale
452                                 aoi_ratio = 1, #porportion of aoi extra
453 height to add to aoi width
454                                 #dur = 3.000, #duration
455                                 nofix_sound = '', # if wav file then its
456 played if no fix on location at onset
457                                 soundfilename = [],
458                                 sound_interval = 1.000, #time between
459 sounds if more than one (s)
460                                 threshold = .3,
461                                 persist_after_gaze = 1.0,
462                                 draw_nothing = False,
463                                 blank_on_exit = False):
464 
465         '''
466         shows a shape until a minimum gaze duration is reached
467         threshold is time that gaze needs to be in aoi before progress
468         persist after gaze is time in ms that display remains on screen
469         after gaze duration threshold met/n/n
470 
471         locs is not normally needed. It overides locations defined in
472 self.locations
473         and must be same length and self.locations.
474         '''
475 
476         out = od([])    # dictionary to collect variables to output
477 
478         if not self.quitnow:
479             i = 0
480             switch = 0
481 
```

```
482             foilpng,targetpng = imagefilename
483
484             images,aoi = self.set_image_aoi(targetpng, foilpng, loc,
485                       image_size = image_size, aoi_scale = aoi_scale,
486   aoi_ratio = aoi_ratio)
487
488             images_2,aoi= self.set_image_aoi(foilpng, foilpng, loc,
489                       image_size = image_size, aoi_scale = aoi_scale,
490   aoi_ratio = aoi_ratio)
491             sounds = []
492             if soundfilename:
493                 sounds = self.get_sounds(soundfilename)
494
495             flip = self.draw_stim(images, aoi, sounds, sound_interval,
496   locs,
497                                   draw_image = not draw_nothing)
498
499             if not flip['gaze_overlap'] and nofix_sound:
500
501   sound.Sound(os.path.join('.',self.stimdir,nofix_sound)).play()
502
503             onset = flip['time']
504             gaze_pos=flip['gaze_position']
505
506             out['onset_time'] = onset
507             out['onset_gaze_pos'] = gaze_pos
508             out['location_x'] = self.locations[loc][0]
509             out['location_y'] = self.locations[loc][1]
510
511             while switch == 0 and not self.quitnow:
512                 flip = self.draw_stim(images, aoi, locs, sounds,
513   sound_interval,
514                                       draw_image = not draw_nothing)
515
516                 if flip['gaze_overlap']:
517                     on_target_start = flip['time']
518                     #on_target_first_position=flip['gaze_position']
519
520                     i += 1
521
522                     while flip['gaze_overlap'] and not self.quitnow:
523                         flip = self.draw_stim(images, aoi, locs, sounds,
524   sound_interval,
525                                               draw_image = not
526   draw_nothing)
527
528                         if core.getTime() - on_target_start >= threshold:
529                             thresh_met_time = flip['time']
530                             last_position_in_aoi = flip['gaze_position']
531                             duration = thresh_met_time - onset
532
533                             out['gaze_dur_threshold_reached'] =
534   thresh_met_time
535                             out['gaze_dur_threshold_reached_pos'] =
536   last_position_in_aoi
537                             out['entries_before_gaze'] = i
538
539                             switch = 1
540
541                             print ('\n%s onset = %.3f, duration = %.3f,
542   entries = %d\n'
```

```
543
544   %(label,onset,duration,i))
545
546                               while core.getTime() - thresh_met_time <=
547   persist_after_gaze and not self.quitnow:
548                                   flip = self.draw_stim(images_2, aoi, locs,
549   sounds, sound_interval,
550                                                         draw_image = not
551   draw_nothing)
552
553                               break
554
555
556
557                   out['end_time'] = core.getTime()
558
559                   if switch:
560                       break
561
562               if blank_on_exit:
563                   self.win.flip()
564
565               return out
566
567
568       def yes_to_question(self,msg_text):
569           msg = visual.TextStim(self.win,msg_text)
570           msg.draw()
571           self.win.flip()
572           core.wait(.2)
573           self.keyboard.clearEvents()
574           if 'y' in self.keyboard.waitForKeys():
575               return True
576           else:
577               return False
578
579
580       def pause_until_anykey(self, msg_text = "Press any key to
581   continue..."):
582           msg = visual.TextStim(self.win,msg_text)
583           msg.draw()
584           self.win.flip()
585           core.wait(.2)
586           self.keyboard.clearEvents()
587           keys = self.keyboard.waitForKeys()
588           print("Key press detected: {}".format(keys))
589           self.win.flip()
590
591
592
593   #%%
594
595   # just for testing
596   if __name__ == '__main__':
597
598       from aseq_setup import alearn as _alearn
599   #    from calibration_setup import calibration
600
601       info = od([('Subject Name','Mark'),
602                           ('Session Code','todays date'),
603                           ('Tracker type, mouse / SR / GP','GP3'),
```

```
                        ('Location', 'school or nursery name'),
                        ('Age, months',0),
                        ('Sex: M,F','F'),
                        ('baseline / main', 'baseline'),
                        ('familiar / unfamiliar', 'familiar'),
                        ('image / speech', 'image'),
                        ('Comments', 'I have no comments')
                    ])

    sess_info = od([('code',info['Session Code']),
                            ('name',info['Subject Name']),
                            ('comments',info['Comments']),
                            ('user_variables',{})
                            ])

    outfile = 'test'

    al = _alearn(sess_info,outfile, tracker_type = 'mouse')

    def testbed():
         al.gazedot_visible = False
         al.setup()
         al.locations =
al.location_sets(al.win.size[1],al.locations_data)['B']
         #al.q()

    testbed()
```

B.2 Calibrartion Set Up Script.

```
"""
Created on Thu Jun 21 10:42:47 2018

@author: psychlaptop
"""

from __future__ import division, print_function, absolute_import

from psychopy import core, visual
from psychopy.gui.qtgui import infoDlg, warnDlg
from psychopy.iohub.client import launchHubServer


def calibration(mytracker):
    GP_CAL_ERR_THRESHOLD = 20.0
    infoDlg("Eye Tracker Setup",
        "Press OK to start\neye tracker setup / calibration procedure.")
    run_cal = True
    while run_cal is True:
          r = mytracker.runSetupProcedure()
          if isinstance(r, dict):
        # iohub-GP3 interface setup call returns the GP3 cal results as a
dict.
        # Parse GP3 calibration results dict
            cal_avg_err = r.get('SUMMARY',{}).get('AVE_ERROR')
            if cal_avg_err is None:
                # Not a GP3 tracker??
                run_cal = False
                continue
            num_calpt = len([k for k in r.keys() if k.startswith('RV')])
            num_val_calpt = r.get('SUMMARY',{}).get('VALID_POINTS')
```

```
            if num_calpt == num_val_calpt and cal_avg_err <
GP_CAL_ERR_THRESHOLD:
                    infoDlg("Calibration Successful", "Calibration
Complete. Average Error: %.2f"%(cal_avg_err))
                    run_cal = False
            else:
                warnDlg("Calibration Failed", "Calibration Incomplete (%d
of %d valid). Average Error: %.2f\nPress OK to restart
Calibration."%(num_val_calpt, num_calpt, cal_avg_err))
          else:
                 run_cal = False
```

B.3 Experimental Run Script.

```
# -*- coding: utf-8 -*-

from __future__ import division, absolute_import
from psychopy import gui, core
from psychopy.data import importConditions, TrialHandler
from psychopy import visual, sound, core
from datetime import datetime
from collections import OrderedDict
from aseq_setup_sound import alearn
from calibration_setup import calibration
import os
import sys



#%% Get Session-level information via dialogue box

# for real
show_dialogue = True
ismouse = False #forces mouse if true

# for testing
#show_dialogue = False
#ismouse = True


# The information
sys.dont_write_bytecode = True # just during testing

working_dir = os.getcwd()
print "\nworking directory = ", working_dir

try: scriptname = os.path.basename(__file__)
except: scriptname = ''

info = OrderedDict([('Experiment name','aseq-exp4')
                    ,('Subject Number (integer from 1 to 36)','1')
                    ,('Trial file name','exp2_very_short.csv')
                    ,('Condition',('C1','C2','C3','C4','C5','C6','ALL'))
                    ,('Date',(datetime.now().strftime('%Y%m%d_%H%M')))
                    ,('Tracker type: mouse/SR/GP3',('mouse','SR','GP3'))
                    ,('Age',0)
                    ,('Sex',('F','M'))
                    ])

if ismouse: info['Tracker type: mouse/SR/GP3'] = 'mouse'

```

```
48  # run the diaglogue box
49  if show_dialogue:
50      exp_dlg = gui.DlgFromDict(info, title = '', order = info.keys())
51
52  # the session info takes a user_variables dictionary
53  # this is not unpacked in the hdf5
54  # but I've made an R script to handle this
55  user_vars = OrderedDict([('age',info['Age']),
56                           ('sex',info['Sex']),
57                           ('trialfile',info['Trial file name']),
58                           ('tracker_type', info['Tracker type:
59  mouse/SR/GP3'])
60                           ])
61
62  # this is read in by iohub as part of the config
63  sess_info = OrderedDict([('name',str(info['Subject Number (integer from 1
64  to 36)']).rjust(2,'0')),
65                           ('user_variables',user_vars),
66                           ('condition', info['Condition']),
67                           ('date',info['Date'])])
68
69  # get file name (io automatically adds extension)
70  outf = '%s_%s_%s'%(info['Experiment
71  name'],sess_info['name'],sess_info['condition']) #add tracker file
72  index = len([f for f in os.listdir('./output') if outf in f])
73  outf = os.path.join('.','output','%s'%(outf))
74  if index:
75      outf = '%s_v%d'%(outf,index+1) #add version if filename exists
76
77  print 'data saved to %s.hdf5'%(outf)
78
79  #%% run functions
80
81  def calibrate():
82      al.pause_until_anykey('''
83      If this experiment inolves eyetracking then you will now be asked to
84  calibrate.
85
86      Press any key to continue...
87      ''')
88      if iseyetracking:
89          al.tracker.setRecordingState(False)
90          al.win.close()#to have access on  calibration windows
91          calibration(al.tracker)
92          al.setup(background_colour=(-1,-1,-1))
93          #al.win = visual.Window(pos = (0,0), #new bit
94                                 #units = 'pix',
95                                 #color = (-1,-1,-1),
96                                 #fullscr=True,
97                                 #allowGUI = False)
98          #al.win.open()
99          al.tracker.setRecordingState(True)
100
101 def run_trial(trial, trial_count):
102
103         al.io.sendMessageEvent("Trial_start",category = 'trial')
104
105         trial['trial_start_time'] = core.getTime()
106
107         dat = al.gaze_contingent_display(imagefilename =['foil.png',
108 'target.png'],
```

```
                                    loc = trial['location'],
                                    image_size = 1/20, #proportion of
display width
                                    aoi_scale = 2.75, #aoi height = image
height * aoi_scale
                                    aoi_ratio = 1, #porportion of aoi extra
height to add to aoi width
                                    threshold = 0.275,
                                    nofix_sound = 'tone2.wav',
                                    persist_after_gaze = 0.750)

        if not al.quitnow:
            trial['onset_time'] = dat['onset_time']
            trial['location_x'] = dat['location_x']
            trial['location_y'] = dat['location_y']
            trial['onset_gaze_pos'] = str(dat['onset_gaze_pos'])

trial['gaze_dur_threshold_reached']=dat['gaze_dur_threshold_reached']
            trial['gaze_dur_threshold_reached_pos']
=str(dat['gaze_dur_threshold_reached_pos'])
            trial['entries_before_gaze'] = dat['entries_before_gaze']

        al.io.sendMessageEvent("Trial_end",category = 'trial')

        trial['trial_order_id'] = trial_count
        trial['trial_end_time'] = core.getTime()

        al.io.addTrialHandlerRecord(trial.values())

def prepare_block(trial):
    # assumes same location set througout block
    al.locations =
al.get_locations('./configs/locations_file.xlsx')[trial['location_set']]
    print "using location set",trial['location_set']
    print al.locations

    calibrate()
    #al.win = visual.Window(pos = (0,0), #new bit
                           # units = 'pix',
                           # color = (-1,-1,-1),
                            #fullscr=True,
                            #allowGUI = False)
    return 0

def run_experiment(trials):

    blocking_variable = 'condition'
    prev_trial = ''
    trial_count = 0

    for trial in trials:

        # actions at the very start of the experiment
        if not prev_trial:
            prepare_block(trial)

        # actions at the start of each block
        elif trial[blocking_variable] != prev_trial[blocking_variable]:
            prepare_block(trial)

        #actions for each trial
```

```
            run_trial(trial,trial_count)
            prev_trial = trial
            trial_count += 1

        al.pause_until_anykey('''
        That's it. Thank you. And goodbye.

        Press any key to finish...
        ''')

        al.q()
        print "\nExperiment run ended successfully"


#%% do it all


#Note that the csv file that gives trials also includes empty columns for
DVs etc
TrialFilename = os.path.join('.','trials',user_vars['trialfile'])

trials, ecnames = importConditions(TrialFilename, returnFieldNames=True)

# to maintain order in output - create ordered rather than normal dict for
each trial
trials = [OrderedDict([(name,trial[name]) for name in ecnames]) for trial
in trials]

# get just trials for specific subject

if sess_info['condition']==(str('ALL')):
    trials = [trial for trial in trials if
(str(trial['subno']).rjust(2,'0') == sess_info['name'])]
else:
    trials = [trial for trial in trials if
(str(trial['subno']).rjust(2,'0') == sess_info['name']) and
(str(trial['condition'])==sess_info['condition'])]

# intiate alearn class and start ioHub
al = alearn(sess_info,outf, tracker_type = user_vars['tracker_type'])
iseyetracking = user_vars['tracker_type'] in ("SR","GP3")

# create trial handler object
trials = TrialHandler(trials, nReps = 1, method= 'sequential')
al.io.createTrialHandlerRecordTable(trials, cv_order = ecnames)

# run the experiment
al.setup(background_colour=(-1,-1,-1))
run_experiment(trials)
```

Appendix C – Data Extraction R scripts

C.1 R script to get Hdf5 file.

```r
###### filename = get_hdf5.R
library(tidyverse)
library(stringr)
library(hdf5r)

#this is where the magic happens
h5f <- H5File$new(filename, mode = "r")

# get the names of the individual datasets in your hdf5 file
getDatasetName = function(x){tail(strsplit(x,'/')[[1]],1)}

# write all datasets in hdf5 file to separate data frames
for(dset in list.datasets(h5f)){
  sname = getDatasetName(dset)
  assign(sname,h5f[[dset]][])
  }

if('user_variables' %in% names(session_meta_data)){
  source("./distance/get_session_uservars.R")}

h5f$close_all()
rm(dset,filename,h5f,sname,getDatasetName)
```

C.2 R script to get session variables for each participant from Hdf5 file.

```r
#######filename= get_session_uservars
library(stringr)

# io hub uservars appear as a column in session_meta_data
# with values like this {"age": 0, "sex": "M"}
# There isn't a sensible way of fixing this in PyschoPy
# so this turns these into separate, names, columns
# it will work for whatever user variables you collect
# (i.e. not just age and sex)
# should be sourced in the main get_hdf5 script

d1 = session_meta_data %>%
  mutate(uv = as.character(user_variables),
         # remove quotes and { }
         uv = str_replace_all(uv,"\\}|\\{|\"",''),
         # split
         uv = str_split(uv,"\\: |\\, ")) %>%
  select(uv)

uv = d1$uv
# get alternate values (i.e. the dict keys) from first item in vector
nms = unlist(uv[1])[c(T,F)]
# remove these names from all items in vector to leave just the dict values
d1 = data.frame(lapply(uv, Filter, f = function(x) !(x %in% nms)))
# transpose
d1 = data.frame(t(d1))
# name columns
names(d1) = nms
#add back in
session_meta_data = cbind(session_meta_data,d1)
session_meta_data$user_variables = NULL
```

```
rm(uv,d1,nms)
```

C.3 R script to wrangle hdf5 files from path and sorting out the trials.

```
'
This assumes one file per subject per session.
So that each trial has unique and sequential start times.
'

library(magrittr)

# get filename of most recently created file

Getfilefrompath <- function()
  {fs=list.files(path,full.names=T)
  for (i in fs){
    filename=i[length(i)]
  return (filename)}}

GetMostRecentFilename <- function()
    {fs = list.files("../output/new_pilot",full.names = T)
    fs = fs[order(file.info(fs)$ctime)]
    filename = fs[length(fs)]
    cat(paste0("\nMost recent data = ",filename,'\n'))

    return(filename)}



MatchTrialsToSamples <- function(trials,samples)
  {
  samples$trial_order_id = 'none'
    for (r in 1:nrow(trials))
      {
      start = trials[r,]$trial_start_time
      end = trials[r,]$trial_end_time
      trialno = trials[r,]$trial_order_id
      samples[samples$time >= start & samples$time <= end,]$trial_order_id
<- trialno
    }

  samples <-  merge(samples, trials,
                        all.x = T, by = 'trial_order_id')

  return(samples)
}
```

C.4 R script that loops overs Hdf5files and extracts session variables and eye-movements for GP3, using the above C.1, C.2, and C.3 scripts.

```
library(tidyverse)
library(magrittr)
library(hdf5r)
#----
# Functions
dist <- function(x1,x2,y1,y2){
  sqrt((x1-x2)**2 + (y1-y2)**2)
}

# ----
```

```
# get data


path="./exp3"
files <- list.files(path, full.names=T)
files_list<-data.frame(files)
for (x in files){
  filename=x
  #remove hdf5 ending
  gsub(x, pattern=".hdf5$", replacement="")
  source("./distance/get_hdf5.R")

  rm(list=Filter(function(x) {c("data.frame") %in% class(get(x)) &
      !(x %in% c("EXP_CV_1","BinocularEyeSampleEvent"))}, ls()))

  EXP_CV_1 %<>%
    group_by(condition) %>% #add ST

    #make numberic
    mutate_at(vars(onset_time,
                   trial_start_time,
                   gaze_dur_threshold_reached,
                   trial_end_time,
                   location_x,
                   location_y,
                   occurrence),
              funs(as.numeric)) %>%
    #add columns with next target location coordinates and next stimuli
onset
    mutate(location_x_next = lead(location_x,1),
           location_y_next = lead(location_y,1),
           next_trial_start_time = lead(trial_start_time,1),

next_gaze_dur_threshold_reached=lead(gaze_dur_threshold_reached,1))%>%
    ungroup()#add ST

  #clean the last trial of each condition that has NAs
  EXP_CV_1<-na.omit(EXP_CV_1)

  summary(EXP_CV_1)

  #subset into smaller data frames based on condition
  df1<-subset(EXP_CV_1, condition=="C1")

  df2<-subset(EXP_CV_1, condition=="C2")



  df3<-subset(EXP_CV_1, condition=="C3")


  df1 %<>%
    group_by(seq, occurrence) %>%
    mutate(transitions = 1,
           transitions = cumsum(transitions),
           seqID_trans = paste0(seq,transitions))
  df2 %<>%
    group_by(seq, occurrence) %>%
    mutate(transitions = 1,
           transitions = cumsum(transitions),
           seqID_trans = paste0(seq,transitions))
```

```
  df3 %<>%
    group_by(seq, occurrence) %>%
    mutate(transitions = 1,
           transitions = cumsum(transitions),
           seqID_trans = paste0(seq,transitions))
  ############################################################
  #match to eyesamples trials
  MatchTrialsToSamples <- function(trials,samples)
  {
    samples$trial_order_id = 'none'
    for (r in 1:nrow(trials))
    {
      start = trials[r,]$gaze_dur_threshold_reached; print
      end = trials[r,]$next_gaze_dur_threshold_reached
      trialno = trials[r,]$trial_order_id
      samples[samples$time >= start & samples$time < end,]$trial_order_id
<- trialno
    }

    samples <-  merge(trials, samples,
                      all.x = T, by = 'trial_order_id') %>%
      arrange(time)


    return(samples)
  }

  #match the subsets(it works only with small chunks of code)
  dfall_1<-MatchTrialsToSamples(df1,BinocularEyeSampleEvent)

  dfall_2<-MatchTrialsToSamples(df2,BinocularEyeSampleEvent)

  dfall_3<-MatchTrialsToSamples(df3,BinocularEyeSampleEvent)

  #for each data set max distance 1
  dms<- unique(with(df1,expand.grid(location_x,location_y))) %>%
    rename(x = Var1, y = Var2) %>%
    unite(x_y,x,y) %>%
    mutate(x_y2 = x_y)

  dms2 <-  unique(with(dms, expand.grid(x_y,x_y2))) %>%
    separate(Var1, c("x1","y1"), sep = '_') %>%
    separate(Var2, c("x2","y2"), sep = '_') %>%
    mutate_all(funs(as.numeric)) %>%
    mutate(dist = dist(x1,x2,y1,y2)) %>%
    summarise(max_disp_dist = max(dist))


  dfall_1$maxdist<- as.numeric(c(dms2["max_disp_dist"]))



  dfall_1 %<>%
    mutate(event_sample_time = time - gaze_dur_threshold_reached,
           trial_duration = trial_end_time - trial_start_time,
           onset_time_of_2nd = next_trial_start_time -
gaze_dur_threshold_reached,
           threshold_reached_for_trial = gaze_dur_threshold_reached -
trial_start_time,
           distance_sample_to_next_target =
dist(right_gaze_x,location_x_next,right_gaze_y,location_y_next),
```

```
            distance_current_target_to_next_target =
dist(location_x,location_x_next,location_y,location_y_next),
            # distance from next target as proportion of distance between
previous target to next target
            proportion_distance_2 = distance_sample_to_next_target /
distance_current_target_to_next_target)
  dfall_1 <- dfall_1 %>%
    mutate(
      event_sample_time=time-gaze_dur_threshold_reached,
      trial_duration=next_gaze_dur_threshold_reached-
gaze_dur_threshold_reached,
      onset_time_of_2nd=next_trial_start_time-gaze_dur_threshold_reached)



  # distance from next target as proportion of maximum array dimension
(distance between two most distant points)
  dfall_1%<>%
    mutate(proportion_distance = distance_sample_to_next_target /maxdist)


  dfall_1 <-dfall_1%>%
    select(subno,condition,location, seq, location_set, target, seqID_trans
,transitions,trial_order_id,occurrence,time,

trial_start_time,trial_end_time,trial_duration,event_sample_time,threshold_
reached_for_trial,
            onset_time_of_2nd,event_sample_time,
            onset_time,onset_gaze_pos,gaze_dur_threshold_reached,
            gaze_dur_threshold_reached_pos,location_y,location_x,
right_gaze_x, right_gaze_y, proportion_distance_2,
            location_x_next, location_y_next, next_trial_start_time,
proportion_distance,
            distance_current_target_to_next_target,
distance_sample_to_next_target )




################################################################################
################################
  #for each data set max distance 2
  dms<- unique(with(df2,expand.grid(location_x,location_y))) %>%
    rename(x = Var1, y = Var2) %>%
    unite(x_y,x,y) %>%
    mutate(x_y2 = x_y)


  dms2 <-  unique(with(dms, expand.grid(x_y,x_y2))) %>%
    separate(Var1, c("x1","y1"), sep = '_') %>%
    separate(Var2, c("x2","y2"), sep = '_') %>%
    mutate_all(funs(as.numeric)) %>%
    mutate(dist = dist(x1,x2,y1,y2)) %>%
    summarise(max_disp_dist = max(dist))



  dfall_2$maxdist<- as.numeric(c(dms2["max_disp_dist"]))
```

```
  dfall_2 %<>%
    mutate(event_sample_time = time - gaze_dur_threshold_reached,
           trial_duration = trial_end_time - trial_start_time,
           onset_time_of_2nd = next_trial_start_time -
gaze_dur_threshold_reached,
           threshold_reached_for_trial = gaze_dur_threshold_reached -
trial_start_time,
           distance_sample_to_next_target =
dist(right_gaze_x,location_x_next,right_gaze_y,location_y_next),
           distance_current_target_to_next_target =
dist(location_x,location_x_next,location_y,location_y_next),
           # distance from next target as proportion of distance between
previous target to next target
           proportion_distance_2 = distance_sample_to_next_target /
distance_current_target_to_next_target)




  dfall_2 <- dfall_2 %>%
    mutate(
      event_sample_time=time-gaze_dur_threshold_reached,
      trial_duration=next_gaze_dur_threshold_reached-
gaze_dur_threshold_reached,
      onset_time_of_2nd=next_trial_start_time-gaze_dur_threshold_reached)
  # distance from next target as proportion of maximum array dimension
(distance between two most distant points)
  dfall_2%<>%
    mutate(proportion_distance = distance_sample_to_next_target /maxdist)
  dfall_2 <-dfall_2%>%
    select(subno,condition,location, seq, location_set, target, seqID_trans
,transitions,trial_order_id,occurrence,time,

trial_start_time,trial_end_time,trial_duration,event_sample_time,threshold_
reached_for_trial,
           onset_time_of_2nd,event_sample_time,
           onset_time,onset_gaze_pos,gaze_dur_threshold_reached,
           gaze_dur_threshold_reached_pos,location_y,location_x,
right_gaze_x, right_gaze_y, proportion_distance_2,
           location_x_next, location_y_next, next_trial_start_time,
proportion_distance,
           distance_current_target_to_next_target,
distance_sample_to_next_target )



##########################################################################
###################################
  #for each data set max distance 3
  dms<- unique(with(df3,expand.grid(location_x,location_y))) %>%
    rename(x = Var1, y = Var2) %>%
    unite(x_y,x,y) %>%
    mutate(x_y2 = x_y)


  dms2 <-  unique(with(dms, expand.grid(x_y,x_y2))) %>%
    separate(Var1, c("x1","y1"), sep = '_') %>%
    separate(Var2, c("x2","y2"), sep = '_') %>%
    mutate_all(funs(as.numeric)) %>%
    mutate(dist = dist(x1,x2,y1,y2)) %>%
    summarise(max_disp_dist = max(dist))
```

```
255
256
257
258     dfall_3$maxdist<- as.numeric(c(dms2["max_disp_dist"]))
259
260
261     dfall_3 %<>%
262       mutate(event_sample_time = time - gaze_dur_threshold_reached,
263              trial_duration = trial_end_time - trial_start_time,
264              onset_time_of_2nd = next_trial_start_time -
265   gaze_dur_threshold_reached,
266              threshold_reached_for_trial = gaze_dur_threshold_reached -
267   trial_start_time,
268              distance_sample_to_next_target =
269   dist(right_gaze_x,location_x_next,right_gaze_y,location_y_next),
270              distance_current_target_to_next_target =
271   dist(location_x,location_x_next,location_y,location_y_next),
272              # distance from next target as proportion of distance between
273   previous target to next target
274              proportion_distance_2 = distance_sample_to_next_target /
275   distance_current_target_to_next_target)
276
277     dfall_3 <- dfall_3 %>%
278       mutate(
279         event_sample_time=time-gaze_dur_threshold_reached,
280         trial_duration=next_gaze_dur_threshold_reached-
281   gaze_dur_threshold_reached,
282         onset_time_of_2nd=next_trial_start_time-gaze_dur_threshold_reached)
283     # distance from next target as proportion of maximum array dimension
284   (distance between two most distant points)
285     dfall_3%<>%
286       mutate(proportion_distance = distance_sample_to_next_target /maxdist)
287     dfall_3 <-dfall_3 %>%
288       select(subno,condition,location, seq, location_set, target, seqID_trans
289   ,transitions,trial_order_id,occurrence,time,
290
291   trial_start_time,trial_end_time,trial_duration,event_sample_time,threshold_
292   reached_for_trial,
293              onset_time_of_2nd,event_sample_time,
294              onset_time,onset_gaze_pos,gaze_dur_threshold_reached,
295              gaze_dur_threshold_reached_pos,location_y,location_x,
296   right_gaze_x, right_gaze_y, proportion_distance_2,
297              location_x_next, location_y_next, next_trial_start_time,
298   proportion_distance,
299              distance_current_target_to_next_target,
300   distance_sample_to_next_target )
301
302
303
304   ##########################################################################
305   ##################
306     ###### up to here is ok############
307     # # square diagonal
308     # dms <- EXP_CV_1 %>%
309     #   summarise(minx = min(location_x),
310     #             miny = min(location_y),
311     #             maxx = max(location_x),
312     #             maxy = max(location_y)) %>%
313     #   mutate(max_disp_dist = dist(minx,maxx,miny,maxy)); dms
314
315
```

```
  dfall<-dffinal<-rbind(dfall_1,dfall_2,dfall_3)
  #save dataframe in csv file
  write.csv(dfall, paste0('./outcome/',basename(x),'.csv'))}

path="folder_outcome_csv_x"

filenames <- list.files(path,full.names=TRUE)
All <- lapply(filenames,function(i){
  read.csv(i, header=TRUE)})
df <- do.call(rbind.data.frame, All)
save(df,file="experiment_x.Rdata")
load("experiment_x.Rdata")
```

C.5 R script that loops overs Hdf5files and extracts session variables and eye-movements for SR EyeLink, using the above C.1, C.2, and C.3 scripts.

```
#----
# Setup
library(tidyverse)
library(magrittr)
library(dplyr)
library(hdf5r)

pythag <- function(x1,x2,y1,y2){
  sqrt((x1-x2)**2 + (y1-y2)**2)
}


#path="C://Users/psychlaptop/Desktop/"
path="C://Users/psychlaptop/Desktop/data_analysis_thesis/exp2"


files <- list.files(path, full.names=T)

Df_all = tibble()

# ----
filename = list.files(path, full.names=T)[1] #for testing

for (filename in files){


  message(paste("\n\nProcessing",filename))

  #gsub(x, pattern=".hdf5$", replacement="")
  h5f <- H5File$new(filename, mode = "r")

  # get the names of the individual datasets in your hdf5 file
  getDatasetName = function(x){tail(strsplit(x,'/')[[1]],1)}

  # write all datasets in hdf5 file to separate data frames
  for(dset in list.datasets(h5f)){
    sname = getDatasetName(dset)
    assign(sname,h5f[[dset]][]) }

  if('user_variables' %in% names(session_meta_data)){
    source("get_session_uservars.R")}

  h5f$close_all()

  rm(list=Filter(function(x) {c("data.frame") %in% class(get(x)) &
```

```
       !(x %in% c("EXP_CV_1","MonocularEyeSampleEvent", "Df_all"))}, ls()))

  MonocularEyeSampleEvent %<>%
    select(time, gaze_x, gaze_y)

# compute various trial-level variables

  EXP_CV_1 %<>%
    group_by(condition) %>%
    mutate_at(vars(onset_time,
                   trial_start_time,
                   gaze_dur_threshold_reached,
                   trial_end_time,
                   trial_order_id,
                   location_x,
                   location_y,
                   occurrence,
                   subno),
              list(as.numeric)) %>%
    #add columns with next target location coordinates and next stimuli
onset
    mutate(location_x_next = lead(location_x,1),
           location_y_next = lead(location_y,1),
           next_trial_start_time = lead(trial_start_time,1),

next_gaze_dur_threshold_reached=lead(gaze_dur_threshold_reached,1)
           ,subno = sprintf("%03d",subno))%>%
    ungroup()

  # get the maximum distance across locations (note that for Experiment 1
the
  # locations were the same in all three conditions?)
  #remove empty rows from part data

  # add transition indexes
  EXP_CV_1 %<>%
    group_by(condition, occurrence, seq) %>%
    mutate(transitions = 1,
           transitions = cumsum(transitions),
           seqID_trans = paste0(seq,transitions),
           seq_len = max(transitions)) %>%
    ungroup()


  # get the maximum distance across locations (note that for Experiment 1
the
  # locations were the same in all three conditions?)
  #remove empty rows from part data

  EXP_CV_1<-na.omit(EXP_CV_1, cols='onset_time')

  EXP_CV_1 %<>%
    group_by(condition) %>%
    filter(length(occurrence)>= 120) %>%
    ungroup()

   source('get_location_dists.R')
  # merge trials and samples
  EXP_CV_1 %<>% mutate(ID = 1,
                       ID = cumsum(ID))

```

```
  MonocularEyeSampleEvent$ID <-
findInterval(x=MonocularEyeSampleEvent$time,
                              vec=EXP_CV_1$gaze_dur_threshold_reached)



  Df <-left_join(MonocularEyeSampleEvent,EXP_CV_1, by = 'ID') %>%
    arrange(time) %>%
    filter(time < next_gaze_dur_threshold_reached)

  rm(MonocularEyeSampleEvent)
  gc(verbose = F) #might help with freeing up memory

  #rm(list=Filter(function(x){!(x %in% c("Df",'files','pythag', 'Df_all',
'EXP_CV_1'))}, ls()))


  Df %<>%
    mutate(event_sample_time=time-gaze_dur_threshold_reached
           ,distance_sample_to_next_target =
pythag(gaze_x,location_x_next,gaze_y,location_y_next)
           ,distance_sample_to_current_target =
pythag(gaze_x,location_x,gaze_y,location_y)
    )

  # AOI around target is circle with radius .5 * minimum distance between
locations.
  AOI_radius = .5*min(Df$min_loc_dist)
  message(paste('AOI radius = ',AOI_radius))


  # fixations on target. pre is for period before target appears. post is
period after it appears.
  Df_on_targ <- Df %>%
    mutate(on_target_pre_total = ifelse(distance_sample_to_next_target <
AOI_radius &
                                        event_sample_time <.75 &
                                        event_sample_time >=0
                                      ,1,0),
           on_target_pre = ifelse(distance_sample_to_next_target <
AOI_radius &
                                          event_sample_time <.75 &
                                          event_sample_time >=.5
                                        ,1,0),
           on_target_post = ifelse(distance_sample_to_next_target <
AOI_radius &
                                        event_sample_time >.75 &
                                        event_sample_time < 1
                                      ,1,0),
           on_target_preandpost=ifelse(distance_sample_to_next_target <
AOI_radius &
                                          event_sample_time >=0 &
                                          event_sample_time < 1
                                        ,1,0)) %>%
    group_by(ID) %>%
    summarise(on_target_pre = sum(on_target_pre),
              on_target_pre_total = sum(on_target_pre_total)
              ,on_target_post = sum(on_target_post),
              on_target_preandpost=sum(on_target_preandpost))

    EXP_CV_1 %<>% left_join(Df_on_targ, by = 'ID') %>%
```

```
168       select(subno,condition,location_set,seq,target,location,
169              occurrence,trial_order_id,ID,seqID_trans,transitions,seq_len,
170   on_target_pre_total,
171              on_target_pre,on_target_post,on_target_preandpost)
172
173       rm(Df,Df_on_targ)
174       gc(verbose = F)
175       Df_all <- rbind(Df_all,EXP_CV_1)
176
177   }
178
179   #Df_all <- rbind(Df_on_target,EXP_CV_1)
180   save(Df_all,file="data/exp2_final.Rdata")
181
182
183   rm(list=Filter(function(x){!(x %in% c('Df_all'))}, ls()))
184
```

Appendix D – Equidistant Points Python Scripts

D.1 Script that generates equal size circles of a given radius into an arbitrary shape.

```
1   from __future__ import division
2
3   import numpy as np
4   from scipy import misc
5   import matplotlib.pyplot as plt
6
7
8   def plot_points(points):
9       """
10      : points are list of x,y tuples
11      """
12
13      plt.plot([p[0] for p in points],[p[1] for p in points],'ro')
14      plt.show()
15      plt.clf()
16
17  def print_points(points):
18      for p in points:
19          print '\t'.join([str(c) for c in p])
20
21  #1/6->9 points,1/ 7->14,15 points
22  #16 points 1/9
23  #seelect to remove some
24  def tessalate(imagefile,radius = 1/6.5):
25      """
26      : radius is radius if circle as proportion image height / width
27      : image must be png (?) of RGB figure with transparent background
28
29      I created the image in powerpoint - encolsed shape with no border -
30  then
31      copied into Irfanview, set background to black, and saved as png with
32      background as transparent.
33      """
34
35      image = misc.imread(imagefile)
36      image = misc.imresize(image,(1000,1000))
37
38      ys, xs, _ = image.shape
39      x, y = np.meshgrid(np.arange(xs), np.arange(ys), sparse=True)
40      radius = xs*radius
41
42      points = []
43
44      for i in range(int(-0.5 * ys / radius), int(xs / radius)):
45          for j in range(int(ys / radius)):
46              x0 = 2 * radius * (i + 0.5 * j)
47              y0 = 2 * radius * np.sqrt(3)/2 * j
48
49              r = np.sqrt((x - x0) ** 2 + (y - y0) ** 2)
50
51              indicator = r < radius
52
53              if np.any(image[indicator, 0] != 0):
54
55                  if x0 <= 1000 and y0 <= 1000:
56                      points.append((round(((x0-500)/1000),4),round(((500-
57  y0)/1000),4)))
```

```
58
59                      #image[r < radius] = [150, 0, 0, 250]
60
61          plt.imshow(image)
62          plt.show()
63          plt.clf()
64
65          plot_points(points)
66
67          return points
68
69      def rotate(point, angle, origin = (0,0)):
70          """
71          Rotate a point counterclockwise by a given angle around a given origin.
72
73          The angle should be given in degrees
74          """
75
76          angle = np.deg2rad(angle)
77          ox, oy = origin
78          px, py = point
79
80          qx = ox + np.cos(angle) * (px - ox) - np.sin(angle) * (py - oy)
81          qy = oy + np.sin(angle) * (px - ox) + np.cos(angle) * (py - oy)
82
83
84          return (round(qx,4), round(qy,4))
85
86      def rotate_all(points, angle, origin = (0,0), scale = 1):
87          points = [rotate(p,angle,origin) for p in points]
88
89          # centre points
90          xs = [p[0] for p in points]
91          ys = [p[1] for p in points]
92          xs = [x -((max(xs)+min(xs))/2) for x in xs]
93          ys = [y -((max(ys)+min(ys))/2) for y in ys]
94
95          points = zip(xs,ys)
96
97          #scale
98          points = [(scale*p[0],scale*p[1]) for p in points]
99
100         plot_points(points)
101         print_points(points)
102         return points
103
104     # get 4 rows of 3 packed points
105     points = tessalate("images/blob5.png",radius=1/6)
106     points.remove((0.5, -0.0774))
107     points.remove((0.5, 0.5))
108     plot_points(points)
109
110     # variations
111     points1 = rotate_all(points, 17, scale = .8)
112     points2 = rotate_all(points, 97, scale = .8)
113     points2 = rotate_all(points, 343, scale = .8)
114
```

D.2 Script that generates coordinates for equidistant points (for the array of locations used in experiments) by using function tessalate from D.1.

```
# -*- coding: utf-8 -*-

from collections import OrderedDict as od


imfile = './images/blob5.png'
# this image looks like this

plt.imshow(misc.imread(imfile))

# pack with circles (probably too big)
points = tessalate(imfile,radius=1/9.4)
points
plot_points(points)

# rotate points by 37 degress, and reduce scale to .6 of display area
points2 = rotate_all(points, 76, scale = 0.6)

# these look like this...
plot_points(points2)

##using these######
#generate 3 backgroungs for my 2, 3, 2_3mixed seq experiments
#create more than 12 ..then delete the ones that I don't need
points3=rotate_all(points, 110, scale = .6)
points3
plot_points(points3)


points4=rotate_all(points, 35, scale = .6)
points4

plot_points(points4)


points5=rotate_all(points, 155, scale = .6)
plot_points(points5)


locations_for_tasks = od([('location_set1',points3),
    ('location_set2',points4),
    ('location_set3',points5)
    ])
locations_for_tasks
```

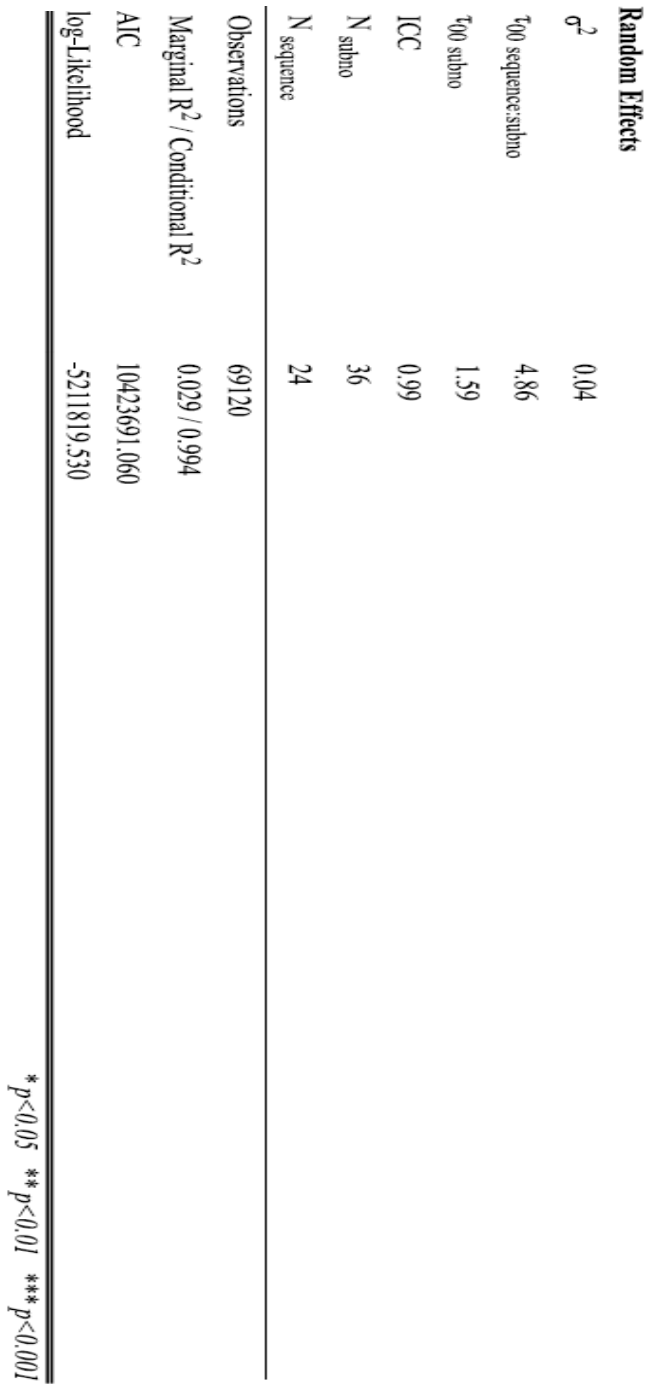
Appendix E- Tables

	on_target_pre_total		
Predictors	*Incidence Rate Ratios*	*CI*	*Statistic*
(Intercept)	18.89 ***	12.74 – 28.01	14.63
poly(occurrence, 3)1	9963670195914200644088686246402024626 46.00 ***	67081990921581739800688482440804268 4000.00 – 14799012731897032058606242802088004860 48.00	444.88
poly(occurrence, 3)1:positionsposition3	87833991.30 ***	58193287.09 – 132572164.48	87.08
poly(occurrence, 3)1:positionsposition3:typenon_mixed	23.96 ***	14.45 – 39.71	12.32
poly(occurrence, 3)1:positionsposition4	1674088.97 ***	980151.44 – 2859327.42	52.47
poly(occurrence, 3)1:positionsposition4:typenon_mixed	9686.54 ***	5630.17 – 16665.40	33.15
poly(occurrence, 3)1:typenon_mixed	0.00 ***	0.00 – 0.00	-44.27
poly(occurrence, 3)2	0.00 ***	0.00 – 0.00	-225.54
poly(occurrence, 3)2:positionsposition3	0.00 ***	0.00 – 0.00	-44.95
poly(occurrence, 3)2:positionsposition3:typenon_mixed	96.47 ***	50.87 – 182.95	13.99
poly(occurrence, 3)2:positionsposition4	0.00 ***	0.00 – 0.00	-91.04
poly(occurrence, 3)2:positionsposition4:typenon_mixed	182277500931 8.63 ***	11053286953 53.50 – 3005901094003.50	110.62
poly(occurrence, 3)2:typenon_mixed	10.09 ***	6.79 – 15.00	11.43
poly(occurrence, 3)3	44404379.67 ***	32695847.99 – 60305789.74	112.75

Table E.1 Part 1 This table shows the residuals of the model of hypothesis model 4, in Chapter 5, Section C.2 describing positioning effects in learning across all tasks (fixed effects).

poly(occurrence, 3)3:positionsposition3	0.15 ***	0.11 – 0.22	-10.67
poly(occurrence, 3)3:positionsposition3:typenon_mixed	3.05 ***	1.98 – 4.72	5.02
poly(occurrence, 3)3:positionsposition4	5440.62 ***	2449.38 – 12084.85	21.12
poly(occurrence, 3)3:positionsposition4:typenon_mixed	0.00 ***	0.00 – 0.00	-30.22
poly(occurrence, 3)3:typenon_mixed	3.97 ***	2.60 – 6.06	6.37
position2	*Reference*		
position3	1.16 ***	1.16 – 1.17	103.92
position4	1.46 ***	1.45 – 1.46	213.44
positionsposition3:typenon_mixed	1.19 ***	1.18 – 1.19	79.51
position_next	*Reference*		
positionsposition4:typenon_mixed	1.04 ***	1.04 – 1.05	15.26
typenon_mixed	1.17	0.88 – 1.56	1.08

Table E.1 Part 2 This table shows the residuals of the model of hypothesis model 4, in Chapter 5, Section C.2 describing positioning effects in learning across all tasks (fixed effects).



Random Effects	
σ^2	0.04
$\tau_{00\ sequence:subno}$	4.86
$\tau_{00\ subno}$	1.59
ICC	0.99
N_{subno}	36
$N_{sequence}$	24
Observations	69120
Marginal R^2 / Conditional R^2	0.029 / 0.994
AIC	10423691.060
log-Likelihood	-5211819.530

*$p<0.05$ **$p<0.01$ ***$p<0.001$

Table E.1 Part 3 This table shows the residuals of the model of hypothesis model 4, in Chapter 5, Section C.2 describing positioning effects in learning across all tasks (random effects).

	on_target_pre_total		
Predictors	*Incidence Rate Ratios*	*CI*	*Statistic*
(Intercept)	29.02 ***	18.41 – 45.74	14.51
poly(occurrence, 3)1	256302693093848855344 20.00 ***	185917502158927864704 64.00 – 353334515171169790868 02.00	314.99
poly(occurrence, 3)1:sequence_length3 dots	44416.34 ***	30403.13 – 64888.43	55.33
poly(occurrence, 3)1:sequence_length4 dots	1182511919.45 ***	787327700.24 – 1776051368.71	100.66
poly(occurrence, 3)2	0.00 ***	0.00 – 0.00	-125.34
poly(occurrence, 3)2:sequence_length3 dots	0.00 ***	0.00 – 0.00	-36.56
poly(occurrence, 3)2:sequence_length4 dots	67.93 ***	43.98 – 104.93	19.02
poly(occurrence, 3)3	768731.59 ***	552494.31 – 1069600.63	80.42
poly(occurrence, 3)3:sequence_length3 dots	0.53 *	0.33 – 0.88	-2.47
poly(occurrence, 3)3:sequence_length4 dots	0.03 ***	0.02 – 0.05	-16.64
sequence_length3 dots	0.73	0.50 – 1.08	-1.58
sequence_length4 dots	0.96	0.66 – 1.41	-0.20
Random Effects			
σ^2	0.04		
$\tau_{00\ sequence:subno}$	4.35		
$\tau_{00\ subno}$	1.72		
ICC	0.99		
$N_{\ subno}$	36		
$N_{\ sequence}$	12		
Observations	34560		
Marginal R^2 / Conditional R^2	0.026 / 0.994		
AIC	5238223.789		
log-Likelihood	-2619097.895		

** p<0.05 ** p<0.01 *** p<0.001*

Table E.2 This table shows the residuals of the model of hypothesis model 2, in Chapter 6, Section C.3 describing learning in same length sequences.

	on_target_pre_total		
Predictors	*Incidence Rate Ratios*	*CI*	*Statistic*
(Intercept)	9.62 ***	5.72 – 16.19	8.52
poly(occurrence, 3)1	603161753354245322384242024068 28.00 ***	419302202126216656304026840628 40.00 – 867641760202004853120488206600 82.00	394.46
poly(occurrence, 3)1:sequence_length3 dots	0.00 ***	0.00 – 0.01	-24.06
poly(occurrence, 3)1:sequence_length4 dots	0.08 ***	0.05 – 0.12	-11.84
poly(occurrence, 3)2	0.00 ***	0.00 – 0.00	-137.31
poly(occurrence, 3)2:sequence_length3 dots	6.98 ***	4.31 – 11.31	7.90
poly(occurrence, 3)2:sequence_length4 dots	0.00 ***	0.00 – 0.00	-27.61
poly(occurrence, 3)3	19568178.47 ***	12659459.41 – 30247232.23	75.56
poly(occurrence, 3)3:sequence_length3 dots	0.00 ***	0.00 – 0.00	-27.95
poly(occurrence, 3)3:sequence_length4 dots	0.12 ***	0.07 – 0.20	-8.25
sequence_length3 dots	2.02 **	1.26 – 3.23	2.92
sequence_length4 dots	4.78 ***	3.12 – 7.33	7.18
Random Effects			
σ^2	0.04		
$\tau_{00\ sequence:subno}$	4.50		
$\tau_{00\ subno}$	1.97		
ICC	0.99		
N_{subno}	36		
$N_{sequence}$	12		
Observations	34560		
Marginal R^2 / Conditional R^2	0.075 / 0.994		
AIC	5306874.140		
log-Likelihood	-2653423.070		

** p<0.05 ** p<0.01 *** p<0.001*

Table E.3 This table shows the residuals of the model of hypothesis model 2, in Chapter 6, Section C.4 describing learning in mixed length sequences.

Predictors	*Incidence Rate Ratios*	*CI*	*Statistic*
	on_target_pre_total		
(Intercept)	9.68 ***	6.98 – 13.43	13.62
poly(occurrence, 3)1	879886408569733014326202244220682640820646464.00 ***	68699661469414070996248886486440888422860422 8.00 – 1126934362450135714324024240884426666408488446.00	819.65
poly(occurrence, 3)1:sequence_length3 dots	0.00 ***	0.00 – 0.00	-36.74
poly(occurrence, 3)1:sequence_length4 dots	0.03 ***	0.02 – 0.04	-21.80
poly(occurrence, 3)1:typenon_mixed	0.00 ***	0.00 – 0.00	-157.36
poly(occurrence, 3)1:typenon_mixed:sequence_length3 dots	10995991761.01 ***	6608187199.62 – 18297277476.54	88.99
poly(occurrence, 3)1:typenon_mixed:sequence_length4 dots	24355325537390 2.31 ***	14213256243663 9.50 – 41734411303298 9.25	120.55
poly(occurrence, 3)2	0.00 ***	0.00 – 0.00	-129.58
poly(occurrence, 3)2:sequence_length3 dots	15.60 ***	7.26 – 33.53	7.03
poly(occurrence, 3)2:sequence_length4 dots	0.00 ***	0.00 – 0.00	-30.00
poly(occurrence, 3)2:typenon_mixed	848.75 ***	501.71 – 1435.84	25.14
poly(occurrence, 3)2:typenon_mixed:sequence_length3 dots	0.00 ***	0.00 – 0.00	-37.41
poly(occurrence, 3)2:typenon_mixed:sequence_length4 dots	2252640.70 ***	1391658.02 – 3646291.00	59.53
poly(occurrence, 3)3	20453848006.49 ***	13535376493.52 – 30908626625.42	112.71

Table E.4 Part 1 This table shows the residuals of the model of hypothesis model 4 , in Chapter 6, Section C.5 describing learning in different types of tasks and lengths of sequences.

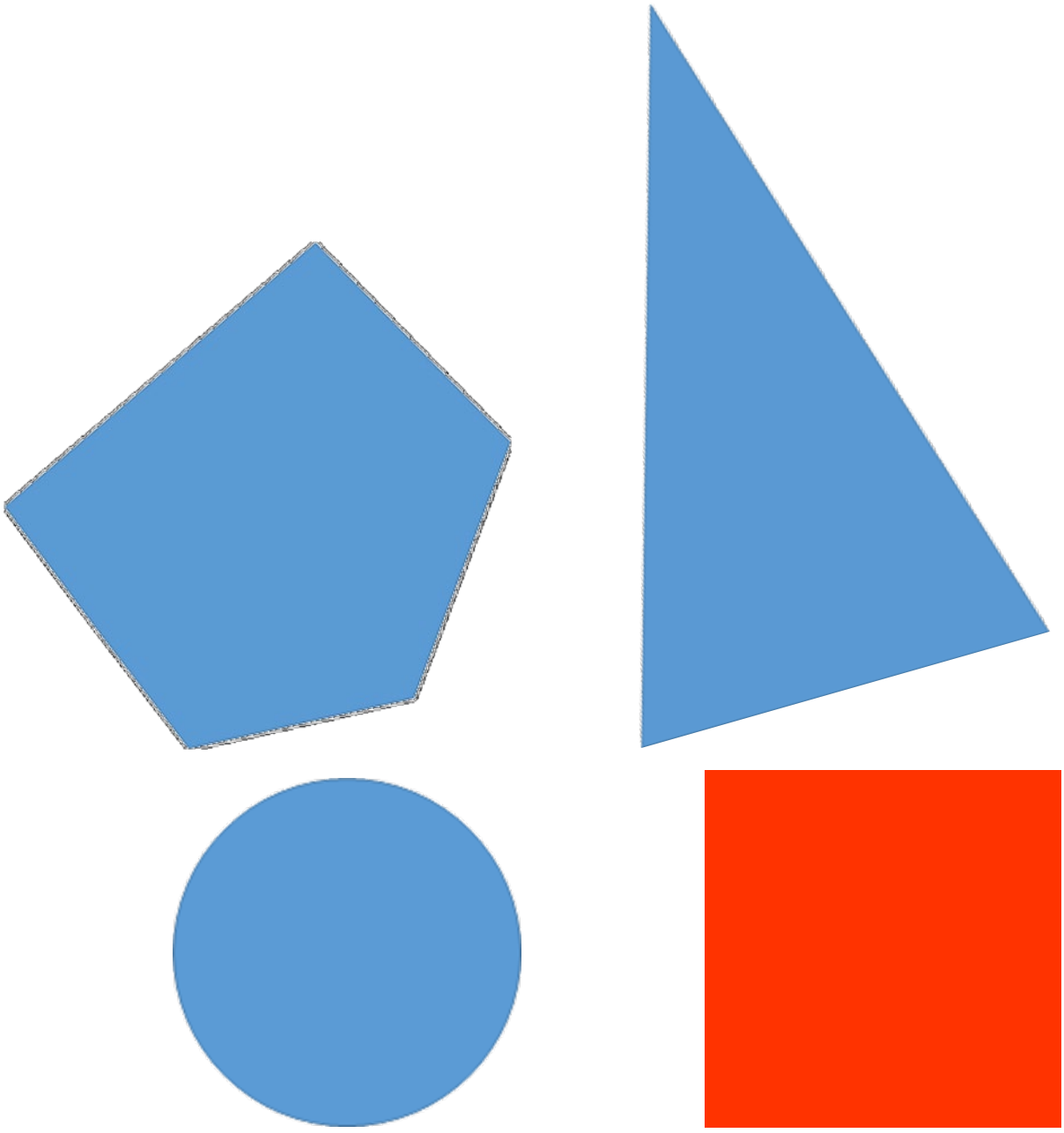
poly(occurrence, 3)3:sequence_length3 dots	0.00 ***	0.00 – 0.00	-45.03
poly(occurrence, 3)3:sequence_length4 dots	0.05 ***	0.03 – 0.09	-10.21
poly(occurrence, 3)3:typenon_mixed	0.01 ***	0.01 – 0.02	-16.26
poly(occurrence, 3)3:typenon_mixed:sequence_length3 dots	27114.24 ***	14384.78 – 51108.35	31.56
poly(occurrence, 3)3:typenon_mixed:sequence_length4 dots	0.14 ***	0.06 – 0.33	-4.48
sequence_length3 dots	2.00 ***	1.54 – 2.60	5.17
sequence_length4 dots	4.75 ***	3.60 – 6.28	10.97
typenon_mixed	2.97 ***	2.29 – 3.86	8.14
typenon_mixed:sequence_length3 dots	0.37 ***	0.26 – 0.52	-5.58
typenon_mixed:sequence_length4 dots	0.20 ***	0.15 – 0.28	-10.14
Random Effects			
σ^2	0.04		
$\tau_{00\ sequence:subno}$	4.68		
$\tau_{00\ subno}$	1.60		
ICC	0.99		
N_{subno}	36		
$N_{sequence}$	24		
Observations	69120		
Marginal R^2 / Conditional R^2	0.052 / 0.994		
AIC	1054092.205		
log-Likelihood	-527520.103		

Table E.4 Part 2 This table shows the residuals of the model of hypothesis model 4, in Chapter 6, Section C.5 describing learning in different types of tasks and lengths of sequences

Appendix F – Stimuli

F.1 Stimuli used in experimental task.

F.2 Shape images that were used to generate equidistant points in scripts attached in Appendix D.



Appendix G- Data visualisation R scripts for Chapter 4 and Chapter 5

G.1 R script for extracting demographic information of participants for Design A and Design B.

```
library(dplyr)
library(tidyverse)
library(magrittr)
#functions
#transform factor into numeric
as.numeric.factor<- function(x){as.numeric(levels(x))[x]}
#transform factor into numeric

demographics_total<-tibble()
#exp3
load("data/exp3_dem.Rdata")
######
Df_all%>%
  select(age,sex,name)-> df;df
sapply(df,class)
df$age<-as.numeric.factor(df$age)
sapply(df,class)
#################

male<-nrow(filter(df,sex=='M'))
female<-nrow(filter(df,sex=='F'))
mean_age<-mean(df$age)
range_age<- list(range(df$age))
min_age<-min (df$age)
max_age<-max (df$age)
sd_age<-sd(df$age)
########################
demographic_3<-data.frame(female, male,mean_age, sd_age,min_age, max_age)

demographic_3 %<>%
  mutate(experiment= 'exp3')

demographics_total<-rbind(demographics_total,demographic_3)

#-------------------------------------------------------------------------
--------
#exp4
load("data/exp4_dem.Rdata")
######
Df_all%>%
  select(age,sex,name)-> df;df
df[df$age=="0"]<-"19"
sapply(df,class)
df$age<-as.numeric.factor(df$age)
sapply(df,class)
#################

male<-nrow(filter(df,sex=='M'))
female<-nrow(filter(df,sex=='F'))
mean_age<-mean(df$age)
range_age<- list(range(df$age))
min_age<-min (df$age)
max_age<-max (df$age)
```

```
sd_age<-sd(df$age)
########################
demographic_4<-data.frame(female, male,mean_age, sd_age,min_age, max_age)

demographic_4 %<>%
  mutate(experiment= 'exp4')

demographics_total<-rbind(demographics_total,demographic_4)


write.csv(demographics_total,"./summary_stats_demographics.csv", row.names
= FALSE)
```

G.2 R script for plotting demographic information of participants for Design A and Design B.

```
library(tidyverse)
library(magrittr)
library(dplyr)
library(ggplot2)

load("data/exp3_dem.Rdata")
df3<-Df_all

load("data/exp4_dem.Rdata")
df4<-Df_all

df<-rbind(df3,df4)

data_summary <- function(x) {
  m <- mean(x)
  ymin <- m-sd(x)
  ymax <- m+sd(x)
  return(c(y=m,ymin=ymin,ymax=ymax))}


sapply(df,class)
as.numeric.factor<- function(x){as.numeric(levels(x))[x]}
df$age<-as.numeric.factor(df$age)
sapply(df,class)



#--------------------

df3$age<-as.numeric.factor(df3$age)

df3%>%
  ggplot(aes(y= age, x=sex, fill=sex))+
  geom_violin(trim=TRUE ,size=1)+
  geom_jitter(colour=2, size=2, position=position_jitter(0.2))+
  scale_fill_brewer(palette="Purples")+
  stat_summary(fun.data=data_summary, colour=1, size=0.7)+
  theme_classic()+
  ylim(17,35)+
  labs(title="Descriptive statistics for participant's age & gender for
Design A", x= "Participant's gender", y="Participant's age")

ggsave('exp3_participants.png',height = 15, width = 15, units = 'cm')

#--------------------------------------
df4$age<-as.numeric.factor(df4$age)
```

```
df4$sex <-factor(df4$sex, levels=c("M", "F"), labels=c("M", "F"))


df4%>%
  ggplot(aes(y= age, x=sex, fill=sex))+
  geom_violin(trim=TRUE ,size=1)+
  geom_jitter(colour=2, size=2, position=position_jitter(0.2))+
  scale_fill_brewer(palette="Oranges")+
  stat_summary(fun.data=data_summary, colour=1, size=0.7)+
  theme_classic()+
  ylim(18,35)+
  labs(title="Descriptive statistics for participant's age & gender for
Design B", x= "Participant's gender", y="Participant's age")

ggsave('exp4_participants.png',height = 15, width = 15, units = 'cm')
```

G.3 R script for experimental plots for design A.

```
library(tidyverse)
library(reshape2)
library(magrittr)
library(dplyr)


load("./data/exp3_final.Rdata")
df1 <- Df_all %>% as_tibble()
names(df1)
df3 <- subset(df1, condition == "C1")
# transform df3 into percentage for meth chap
df3$percentage_on_target<-(df3$on_target_pre_total/45)*100
####
df3 %>%
  ggplot(aes(y = percentage_on_target, x = occurrence) )+
  geom_smooth(size= 1.2, method='lm', formula = y ~ poly(x,3))+
  facet_grid(~condition)+
  theme_minimal(base_size = 10)+
  ylim(0,100)+
  labs(title ="Learning rate in 3 dots tasks - Design A")+
  xlab("Number of repetition of item - exposure")+
  ylab("Percentage of eye samples on target position \n during  blank
period")+
  ggsave('../plots/exp3/ percentage.png', height = 15, width = 15, units =
'cm')

#####
#individual differences

df3 %>%
  ggplot(aes(y = on_target_pre_total, x = occurrence) )+
  geom_smooth(size= 0.8, method='lm', formula = y ~ poly(x,3))+
  facet_wrap(~subno)+
  theme_minimal(base_size = 10)+
  ylim(0,45)+
  labs(title ="Learning rate in 3 dots tasks - Design A")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank
period")+
  ggsave('../plots/exp3/ meth_3.png', height = 13, width = 15, units =
'cm')

###### per sequence learning
```

```
df3 %>%
  ggplot(aes(y = on_target_pre_total, x = occurrence, colour=seq) )+
  geom_smooth(size= 0.8, method='lm', formula = y ~ poly(x,3))+
  theme_minimal(base_size = 10)+
  labs(title ="Learning rate in 3 dots tasks - Design A")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank
period")+
  ggsave('../plots/exp3/ meth_3.png', height = 13, width = 15, units =
'cm')
###############


######raw samples distance from target

library(tidyverse)
library(magrittr)
library(hdf5r)
#----
# Functions
dist <- function(x1,x2,y1,y2){
  sqrt((x1-x2)**2 + (y1-y2)**2)
}

# ----
# get data


path="./exp3"
files <- list.files(path, full.names=T)
files_list<-data.frame(files)
for (x in files){
  filename=x
  #remove hdf5 ending
  gsub(x, pattern=".hdf5$", replacement="")
  source("./distance/get_hdf5.R")

  rm(list=Filter(function(x) {c("data.frame") %in% class(get(x)) &
      !(x %in% c("EXP_CV_1","BinocularEyeSampleEvent"))}, ls()))

  EXP_CV_1 %<>%
    group_by(condition) %>% #add ST

    #make numberic
    mutate_at(vars(onset_time,
                   trial_start_time,
                   gaze_dur_threshold_reached,
                   trial_end_time,
                   location_x,
                   location_y,
                   occurrence),
              funs(as.numeric)) %>%
    #add columns with next target location coordinates and next stimuli
onset
    mutate(location_x_next = lead(location_x,1),
           location_y_next = lead(location_y,1),
           next_trial_start_time = lead(trial_start_time,1),

next_gaze_dur_threshold_reached=lead(gaze_dur_threshold_reached,1))%>%
    ungroup()#add ST

```

```
105     #clean the last trial of each condition that has NAs
106     EXP_CV_1<-na.omit(EXP_CV_1)
107
108     summary(EXP_CV_1)
109
110     #subset into smaller data frames based on condition
111     df1<-subset(EXP_CV_1, condition=="C1")
112
113     df2<-subset(EXP_CV_1, condition=="C2")
114
115
116
117     df3<-subset(EXP_CV_1, condition=="C3")
118
119
120     df1 %<>%
121       group_by(seq, occurrence) %>%
122       mutate(transitions = 1,
123              transitions = cumsum(transitions),
124              seqID_trans = paste0(seq,transitions))
125     df2 %<>%
126       group_by(seq, occurrence) %>%
127       mutate(transitions = 1,
128              transitions = cumsum(transitions),
129              seqID_trans = paste0(seq,transitions))
130     df3 %<>%
131       group_by(seq, occurrence) %>%
132       mutate(transitions = 1,
133              transitions = cumsum(transitions),
134              seqID_trans = paste0(seq,transitions))
135     ############################################################
136     #match to eyesamples trials
137     MatchTrialsToSamples <- function(trials,samples)
138     {
139       samples$trial_order_id = 'none'
140       for (r in 1:nrow(trials))
141       {
142         start = trials[r,]$gaze_dur_threshold_reached; print
143         end = trials[r,]$next_gaze_dur_threshold_reached
144         trialno = trials[r,]$trial_order_id
145         samples[samples$time >= start & samples$time < end,]$trial_order_id
146   <- trialno
147       }
148
149       samples <-  merge(trials, samples,
150                         all.x = T, by = 'trial_order_id') %>%
151         arrange(time)
152
153
154       return(samples)
155     }
156
157     #match the subsets(it works only with small chunks of code)
158     dfall_1<-MatchTrialsToSamples(df1,BinocularEyeSampleEvent)
159
160     dfall_2<-MatchTrialsToSamples(df2,BinocularEyeSampleEvent)
161
162     dfall_3<-MatchTrialsToSamples(df3,BinocularEyeSampleEvent)
163
164
165
```

```
166
167
168     #for each data set max distance 1
169     dms<- unique(with(df1,expand.grid(location_x,location_y))) %>%
170       rename(x = Var1, y = Var2) %>%
171       unite(x_y,x,y) %>%
172       mutate(x_y2 = x_y)
173
174     dms2 <-  unique(with(dms, expand.grid(x_y,x_y2))) %>%
175       separate(Var1, c("x1","y1"), sep = '_') %>%
176       separate(Var2, c("x2","y2"), sep = '_') %>%
177       mutate_all(funs(as.numeric)) %>%
178       mutate(dist = dist(x1,x2,y1,y2)) %>%
179       summarise(max_disp_dist = max(dist))
180
181
182     dfall_1$maxdist<- as.numeric(c(dms2["max_disp_dist"]))
183
184
185
186     dfall_1 %<>%
187       mutate(event_sample_time = time - gaze_dur_threshold_reached,
188              trial_duration = trial_end_time - trial_start_time,
189              onset_time_of_2nd = next_trial_start_time -
190   gaze_dur_threshold_reached,
191              threshold_reached_for_trial = gaze_dur_threshold_reached -
192   trial_start_time,
193              distance_sample_to_next_target =
194   dist(right_gaze_x,location_x_next,right_gaze_y,location_y_next),
195              distance_current_target_to_next_target =
196   dist(location_x,location_x_next,location_y,location_y_next),
197              # distance from next target as proportion of distance between
198   previous target to next target
199              proportion_distance_2 = distance_sample_to_next_target /
200   distance_current_target_to_next_target)
201     dfall_1 <- dfall_1 %>%
202       mutate(
203         event_sample_time=time-gaze_dur_threshold_reached,
204         trial_duration=next_gaze_dur_threshold_reached-
205   gaze_dur_threshold_reached,
206         onset_time_of_2nd=next_trial_start_time-gaze_dur_threshold_reached)
207
208
209
210     # distance from next target as proportion of maximum array dimension
211   (distance between two most distant points)
212     dfall_1%<>%
213       mutate(proportion_distance = distance_sample_to_next_target /maxdist)
214
215
216     dfall_1 <-dfall_1%>%
217       select(subno,condition,location, seq, location_set, target, seqID_trans
218   ,transitions,trial_order_id,occurrence,time,
219
220   trial_start_time,trial_end_time,trial_duration,event_sample_time,threshold_
221   reached_for_trial,
222              onset_time_of_2nd,event_sample_time,
223              onset_time,onset_gaze_pos,gaze_dur_threshold_reached,
224              gaze_dur_threshold_reached_pos,location_y,location_x,
225   right_gaze_x, right_gaze_y, proportion_distance_2,
```

```
            location_x_next, location_y_next, next_trial_start_time,
proportion_distance,
            distance_current_target_to_next_target,
distance_sample_to_next_target )

#############################################################################
################################
  #for each data set max distance 2
  dms<- unique(with(df2,expand.grid(location_x,location_y))) %>%
    rename(x = Var1, y = Var2) %>%
    unite(x_y,x,y) %>%
    mutate(x_y2 = x_y)


  dms2 <-  unique(with(dms, expand.grid(x_y,x_y2))) %>%
    separate(Var1, c("x1","y1"), sep = '_') %>%
    separate(Var2, c("x2","y2"), sep = '_') %>%
    mutate_all(funs(as.numeric)) %>%
    mutate(dist = dist(x1,x2,y1,y2)) %>%
    summarise(max_disp_dist = max(dist))



  dfall_2$maxdist<- as.numeric(c(dms2["max_disp_dist"]))



  dfall_2 %<>%
    mutate(event_sample_time = time - gaze_dur_threshold_reached,
           trial_duration = trial_end_time - trial_start_time,
           onset_time_of_2nd = next_trial_start_time -
gaze_dur_threshold_reached,
           threshold_reached_for_trial = gaze_dur_threshold_reached -
trial_start_time,
           distance_sample_to_next_target =
dist(right_gaze_x,location_x_next,right_gaze_y,location_y_next),
           distance_current_target_to_next_target =
dist(location_x,location_x_next,location_y,location_y_next),
           # distance from next target as proportion of distance between
previous target to next target
           proportion_distance_2 = distance_sample_to_next_target /
distance_current_target_to_next_target)



  dfall_2 <- dfall_2 %>%
    mutate(
      event_sample_time=time-gaze_dur_threshold_reached,
      trial_duration=next_gaze_dur_threshold_reached-
gaze_dur_threshold_reached,
      onset_time_of_2nd=next_trial_start_time-gaze_dur_threshold_reached)
  # distance from next target as proportion of maximum array dimension
(distance between two most distant points)
  dfall_2%<>%
    mutate(proportion_distance = distance_sample_to_next_target /maxdist)
  dfall_2 <-dfall_2%>%
    select(subno,condition,location, seq, location_set, target, seqID_trans
,transitions,trial_order_id,occurrence,time,
```

```
285
286   trial_start_time,trial_end_time,trial_duration,event_sample_time,threshold_
287   reached_for_trial,
288              onset_time_of_2nd,event_sample_time,
289              onset_time,onset_gaze_pos,gaze_dur_threshold_reached,
290              gaze_dur_threshold_reached_pos,location_y,location_x,
291   right_gaze_x, right_gaze_y, proportion_distance_2,
292              location_x_next, location_y_next, next_trial_start_time,
293   proportion_distance,
294              distance_current_target_to_next_target,
295   distance_sample_to_next_target )
296
297
298
299   ##########################################################################
300   ####################################
301     #for each data set max distance 3
302     dms<- unique(with(df3,expand.grid(location_x,location_y))) %>%
303       rename(x = Var1, y = Var2) %>%
304       unite(x_y,x,y) %>%
305       mutate(x_y2 = x_y)
306
307
308     dms2 <-  unique(with(dms, expand.grid(x_y,x_y2))) %>%
309       separate(Var1, c("x1","y1"), sep = '_') %>%
310       separate(Var2, c("x2","y2"), sep = '_') %>%
311       mutate_all(funs(as.numeric)) %>%
312       mutate(dist = dist(x1,x2,y1,y2)) %>%
313       summarise(max_disp_dist = max(dist))
314
315
316
317     dfall_3$maxdist<- as.numeric(c(dms2["max_disp_dist"]))
318
319
320     dfall_3 %<>%
321       mutate(event_sample_time = time - gaze_dur_threshold_reached,
322              trial_duration = trial_end_time - trial_start_time,
323              onset_time_of_2nd = next_trial_start_time -
324   gaze_dur_threshold_reached,
325              threshold_reached_for_trial = gaze_dur_threshold_reached -
326   trial_start_time,
327              distance_sample_to_next_target =
328   dist(right_gaze_x,location_x_next,right_gaze_y,location_y_next),
329              distance_current_target_to_next_target =
330   dist(location_x,location_x_next,location_y,location_y_next),
331              # distance from next target as proportion of distance between
332   previous target to next target
333              proportion_distance_2 = distance_sample_to_next_target /
334   distance_current_target_to_next_target)
335
336     dfall_3 <- dfall_3 %>%
337       mutate(
338         event_sample_time=time-gaze_dur_threshold_reached,
339         trial_duration=next_gaze_dur_threshold_reached-
340   gaze_dur_threshold_reached,
341         onset_time_of_2nd=next_trial_start_time-gaze_dur_threshold_reached)
342     # distance from next target as proportion of maximum array dimension
343   (distance between two most distant points)
344     dfall_3%<>%
345       mutate(proportion_distance = distance_sample_to_next_target /maxdist)
```

```
  dfall_3 <-dfall_3 %>%
    select(subno,condition,location, seq, location_set, target, seqID_trans
,transitions,trial_order_id,occurrence,time,

trial_start_time,trial_end_time,trial_duration,event_sample_time,threshold_
reached_for_trial,
           onset_time_of_2nd,event_sample_time,
           onset_time,onset_gaze_pos,gaze_dur_threshold_reached,
           gaze_dur_threshold_reached_pos,location_y,location_x,
right_gaze_x, right_gaze_y, proportion_distance_2,
           location_x_next, location_y_next, next_trial_start_time,
proportion_distance,
           distance_current_target_to_next_target,
distance_sample_to_next_target )



################################################################################
#################
  ###### up to here is ok############
  # # square diagonal
  # dms <- EXP_CV_1 %>%
  #   summarise(minx = min(location_x),
  #             miny = min(location_y),
  #             maxx = max(location_x),
  #             maxy = max(location_y)) %>%
  #   mutate(max_disp_dist = dist(minx,maxx,miny,maxy)); dms



  dfall<-dffinal<-rbind(dfall_1,dfall_2,dfall_3)
  #save dataframe in csv file
  write.csv(dfall, paste0('./outcome/',basename(x),'.csv'))}
###############
###plot distance

dfall%>%
  filter(condition=="C1") -> df2;df2


Df2 <- df2%>%

select(event_sample_time,proportion_distance,occurrence,transitions,onset_t
ime_of_2nd, seqID_trans) %>%
  filter_all(any_vars(!is.na(.)))


Df2 %>%
  filter(occurrence  %in% c(1, 5,10,15,20,25,30,35,40))%>%
  ggplot(aes(y=event_sample_time, x=proportion_distance))+
  labs(x = "Proportion of distance from target")+
  labs(y = "Time of trial")+
  geom_point(colour='red',alpha = 1, size = 0.2,na.rm =FALSE) +
  geom_hline(aes(yintercept = onset_time_of_2nd),colour='blue3', linetype =
'solid') +
  facet_grid(occurrence~seqID_trans)+
  ylim(0,1.5) +
  theme(axis.text = element_text(size = 5))+
  xlim(0,1.5)+
  labs(title ="Raw eyedata of participant 005 - Design A")+
  ggsave('templ1.png', width = 15, height = 12.5, units = 'cm')
```

G.4 R script for experimental plots for design B.

```
library(tidyverse)
library(reshape2)
library(magrittr)
library(dplyr)

load("../data/exp4_final.Rdata")

df1 <- Df_all %>% as_tibble()
names(df1)

recode_if <- function(x, condition, ...) {
  if_else(condition, recode(x, ...), x)
}

#transform data in usable format
#transform data in usable format
df1$sequence<-as.factor(paste(df1$seq, df1$condition, sep = "_"))

df2<-df1 %>%
  mutate(type = ifelse(condition %in% c("C1", "C4", "C5"), "non_mixed",
                               ifelse(condition %in% c("C2", "C3", "C6"),
"mixed", NA)),
                condition = recode(condition,
                                   C1 = "4 dots",
                                   C2 = "2&4 dots",
                                   C3 = "3&4 dots",
                                   C4 = "3 dots",
                                   C5 = "2 dots",
                                   C6 = "2&3 dots"),
                sequence_length=recode(seq_len,
                                       "4"="4 dots",
                                       "2"="2 dots",
                                       "3"="3 dots")) %>%

         mutate(positions=factor(paste0("position",transitions+1)),
                target_position=positions)


df2 <- df2 %>%
  mutate(target_position = recode_if(target_position, sequence_length == "4
dots" & positions == "position5", "position5" = "position_next"))


df2$target_position[df2$sequence_length == '3 dots' & df2$positions ==
'position4'] <- "position_next"

df2$target_position[df2$sequence_length == '2 dots' & df2$positions ==
'position3'] <- "position_next"


df2<- df2%>%
  mutate(type_transition= ifelse(positions %in% c("position_next"),
"1st_transition",
                                 ifelse(positions %in% c("position2",
"position3", "position4"), "in_sequence_transition",NA )))

df2<-df2%>%
  mutate(  target_position=recode(target_position,
                                  "position_next"="position1",
```

```
                                    "position2"="position2",
                                    "position3"="position3",
                                    "position4"="position4"))
df2<- df2%>%
  select(on_target_pre_total,positions,seq, target_position,
sequence_length,occurrence,condition,subno,sequence)
write.csv(df2, "ready_data_exp4.csv")

##############################################################################
#############################################

df6 <- subset(df2, condition == "3 dots")
df6$percentage_on_target<-(df6$on_target_pre_total/750)*100
df6 %>%
  ggplot(aes(y = percentage_on_target, x = occurrence) )+
  geom_smooth(size= 1.2, method='lm', formula = y ~ poly(x,3))+
  facet_grid(~condition)+
  theme_minimal(base_size = 10)+
  ylim(0,100)+
  labs(title ="Learning rate in 3 dots tasks - Design B")+
  xlab("Number of repetition of item - exposure")+
  ylab("Percentage of eye samples on  target position \n during  blank
period")+
  ggsave('../plots/exp4/percentage.png', height = 15, width = 15, units =
'cm')


df6 %>%
  ggplot(aes(y = on_target_pre_total, x = occurrence, colour=seq) )+
  geom_smooth(size= 1.2, method='lm', formula = y ~ poly(x,3))+
  facet_grid(~condition)+
  theme_minimal(base_size = 10)+
  labs(title ="Learning rate in 3 dots tasks - Design B")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank
period")+
  ggsave('../plots/exp4/ meth_2.png', height = 15, width = 15, units =
'cm')


#individual differences

df6 %>%
  ggplot(aes(y = on_target_pre_total, x = occurrence) )+
  geom_smooth(size= 0.8, method='lm', formula = y ~ poly(x,3))+
  facet_wrap(~subno)+
  theme_minimal(base_size = 10)+
  ylim(0,NA)+
  labs(title ="Learning rate in 3 dots tasks - Design B")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank
period")+
  ggsave('../plots/exp4/ meth_3.png', height = 13, width = 15, units =
'cm')
```

G.5 R scripts for piloting data

```
library(lme4)
```

```
library(dplyr)
library(tidyverse)
library(reshape2)
library(magrittr)
library(effects)
library(sjPlot)
library(sjmisc)
library(sjlabelled)

#500ms
load("data/piloting_500.Rdata")
str(Df_all)
##piloting data gp3

df<-Df_all%>%
  filter(condition == "C1") %>%
  mutate( condition = recode(condition,
                             C1 = "3 dots"))

df%>%
  ggplot(aes(x=occurrence, y=on_target_pre_total),group=subno) +
  geom_smooth( method = "loess")+
  theme_classic(base_size = 10)+
  facet_wrap(~subno)+
  coord_cartesian(ylim = c(0, 30))+
  labs(title ="Learning rate across 3 dots task for a 500ms blank period
per participant")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank period
of 500ms")+
  ggsave('./new_plots/piloting_500.png', height = 8,width = 16.6, units =
'cm')


#600ms
load("data/piloting_600.Rdata")
str(Df_all)
##piloting data gp3

df<-Df_all%>%
  filter(condition == "C1") %>%
  mutate( condition = recode(condition,
                             C1 = "3 dots"))

df%>%
  ggplot(aes(x=occurrence, y=on_target_pre_total),group=subno) +
  geom_smooth( method = "loess")+
  theme_classic(base_size = 10)+
  facet_wrap(~subno)+
  coord_cartesian(ylim = c(0, 36))+
  labs(title ="Learning rate across 3 dots task for a 600ms blank period
per participant")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank period
of 600ms")+
  ggsave('./new_plots/piloting_600.png', height = 8,width = 16.6, units =
'cm')

#700ms
load("data/piloting_700.Rdata")
str(Df_all)
```

```
##piloting data gp3

df<-Df_all%>%
  filter(condition == "C1") %>%
  mutate( condition = recode(condition,
                             C1 = "3 dots"))

df%>%
  ggplot(aes(x=occurrence, y=on_target_pre_total),group=subno) +
  geom_smooth( method = "loess")+
  theme_classic(base_size = 10)+
  facet_wrap(~subno)+
  coord_cartesian(ylim = c(0, 42))+
  labs(title ="Learning rate across 3 dots task for a 700ms blank period")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank period
of 700ms")+
  ggsave('./new_plots/piloting_700.png', height = 8,width = 13, units =
'cm')
```

G.6 R script for fixation and eye-sample graphs in Chapter 4.

```

load("data/Rdata_exp3.Rdata")
df<- Df_all%>%
  filter(condition %in% c("C1"))%>%
  mutate(condition=recode(condition, C1= "3 dots" ))
df %>%
  filter(subno %in% c("005"))%>%
  ggplot(aes(y = on_target_pre_total, x = occurrence) )+
  geom_smooth(size= 0.8, method='lm', formula = y ~ poly(x,3))+
  facet_wrap(~subno)+
  theme_minimal(base_size = 10)+
  ylim(0,20)+
  labs(title ="Learning rate in 3 dots task for participant 005 - Design
A")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank
period")+
  ggsave('./new_plots/ eye_sample_gp3.png', height = 13, width = 15, units
= 'cm')

load("./fixations_gp3.Rdata")
df<- Df_all%>%
  filter(condition %in% c("C1"))%>%
  mutate(condition=recode(condition, C1= "3 dots" ))
df %>%
  filter(subno %in% c("005"))%>%
  ggplot(aes(y = on_target_pre_total, x = occurrence) )+
  geom_smooth(size= 0.8, method='lm', formula = y ~ poly(x,3))+
  facet_wrap(~subno)+
  theme_minimal(base_size = 10)+
  ylim(0,2)+
  labs(title ="Learning rate in 3 dots task for participant 005 - Design
A")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of fixations on  target position \n during  blank period")+
  ggsave('./new_plots/ fixation_gp3.png', height = 13, width = 15, units =
'cm')

```

Appendix H – Analysis R Scripts for Chapter 6

H.1 R script for models and model fit for positioning effects.

```
###POSITIONING#######

library(lme4)
library(tidyverse)
library(reshape2)
library(magrittr)
library(effects)
library(sjPlot)
library(sjmisc)
library(sjlabelled)
load("data/exp4_final.Rdata")
df1 <- Df_all %>% as_tibble()
names(df1)

#transform data in usable format
#transform data in usable format
df1$sequence<-as.factor(paste(df1$seq, df1$condition, sep = "_"))
df1 %>% mutate(type = ifelse(condition %in% c("C1", "C4", "C5"),
"non_mixed",
                             ifelse(condition %in% c("C2", "C3", "C6"),
"mixed", NA)),
               condition = recode(condition,
                                  C1 = "4 dots",
                                  C2 = "2&4 dots",
                                  C3 = "3&4 dots",
                                  C4 = "3 dots",
                                  C5 = "2 dots",
                                  C6 = "2&3 dots"),
               sequence_length=recode(seq_len,
                                      "4"="4 dots",
                                      "2"="2 dots",
                                      "3"="3 dots")) %>%


  mutate(positions=factor(paste0("position",transitions+1)),
         positions=recode(positions,
                          position5="position_next"),
         trans = factor(paste0("trans", transitions)))%>%

  dplyr::select(on_target_pre_total,positions,sequence_length, type,
occurrence, transitions, trans,condition, subno, sequence) -> df2;df2

#learning for condition-task 4 dots
df3 <- subset(df2, transitions %in% 1:3 & condition == "4 dots")
#learning for condition-task 3 dots
df4<- subset(df2, transitions %in% 1:2 & condition == "3 dots")
#learning for condition-task 2 dots
df5<- subset(df2, transitions %in% 1:1 & condition == "2 dots")



#learning for condition-task 2&3 dots
df6 <- subset(df2, sequence_length == "3 dots"  & condition == "2&3 dots" &
transitions %in% 1:2 )
df7<-subset(df2, sequence_length == "2 dots"  & condition == "2&3 dots" &
transitions %in% 1:1 )
```

```
df8<-rbind(df6,df7)

#learning for condition-task 2&4 dots
df9 <- subset(df2, sequence_length == "4 dots"  & condition == "2&4 dots" &
transitions %in% 1:3 )
df10<-subset(df2, sequence_length == "2 dots"  & condition == "2&4 dots" &
transitions %in% 1:1 )
df11<-rbind(df9,df10)


#learning for condition-task 2&4 dots
df12 <- subset(df2, sequence_length == "4 dots"  & condition == "3&4 dots"
& transitions %in% 1:3 )
df13<-subset(df2, sequence_length == "3 dots"  & condition == "3&4 dots" &
transitions %in% 1:2 )
df14<-rbind(df12,df13)

library(data.table)
#bind the learnind data in one df
df_learning<-rbind(df14,df11,df8,df5,df4,df3)
setDT(df_learning)[ , new_subno := .GRP, by = .(subno)]

rm(list=Filter(function(x){!(x %in% c('df2','df_learning'))}, ls()))
#########################################################################
formula_3<- glmer(on_target_pre_total ~ poly(occurrence,3)  + (1|subno) +
(1|sequence:subno), data=df_learning, family = poisson())
formula_2<- glmer(on_target_pre_total ~ poly(occurrence,2)  + (1|subno) +
(1|sequence:subno), data=df_learning, family = poisson())
formula_1<- glmer(on_target_pre_total ~ poly(occurrence,1)  + (1|subno) +
(1|sequence:subno), data=df_learning, family = poisson())
formula_0<- glmer(on_target_pre_total ~ 1  + (1|subno) +
(1|sequence:subno), data=df_learning, family = poisson())

anova(formula_0, formula_1)
anova(formula_1, formula_2)
anova(formula_2, formula_3)

formula_4<- glmer(on_target_pre_total ~ poly(occurrence,3)+positions  +
(1|subno) + (1|sequence:subno), data=df_learning, family = poisson())
formula_5<- glmer(on_target_pre_total ~ poly(occurrence,3)*positions  +
(1|subno) + (1|sequence:subno), data=df_learning, family = poisson())
formula_6<- glmer(on_target_pre_total ~ poly(occurrence,3)*positions + type
+ (1|subno) + (1|sequence:subno), data=df_learning, family = poisson())
formula_7<- glmer(on_target_pre_total ~ poly(occurrence,3)*positions * type
+ (1|subno) + (1|sequence:subno), data=df_learning, family = poisson())
anova(formula_3, formula_4)
anova(formula_4,formula_5)
anova(formula_5,formula_6)
anova (formula_5, formula_7)
tab_model(formula_7,p.style = c( "both"))
summary(formula_7)
df_learning$fitted_model<-fitted(formula_7)

df_learning%>%
  ggplot(aes(x=occurrence, y=fitted_model, colour=positions),group=subno) +
  geom_smooth( method = "lm", formula = y ~ poly(x,3))+
  theme_minimal(base_size = 10)+
  facet_grid(~condition)+
  coord_cartesian(ylim = c(0, 250))+
  labs(title ="Fit of Model for positioning effects  across  tasks")+
  xlab("Number of repetition of item - exposure")+
```

```
  ylab("Number of eye-samples on  target position \n during  blank
period")+
  ggsave('./plots/position_condition.png', height = 16.6,width = 16.6,
units = 'cm')

df_learning%>%
  ggplot(aes(x=occurrence, y=fitted_model, colour=positions),group=subno) +
  geom_smooth( method = "lm", formula = y ~ poly(x,3))+
  theme_minimal(base_size = 10)+
  facet_grid(sequence_length~condition)+
  coord_cartesian(ylim = c(0, 250))+
  labs(title ="Fit of Model for positioning effects \nacross different
sequence lengths & tasks")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank
period")+
  ggsave('./plots/position_condition per seq.png', height = 16.6,width =
16.6, units = 'cm')
```

Appendix I – Analysis R Scripts for Chapter 7

I.1 R Script for visualisation of learning across tasks and fit of curve in Chapter 7, Section C.1 and C.2.

```
library(lme4)
library(tidyverse)
library(reshape2)
library(magrittr)
library(effects)
library(sjPlot)
library(sjmisc)
library(sjlabelled)
load("data/exp4_final.Rdata")
df1 <- Df_all %>% as_tibble()
names(df1)

#transform data in usable format
#transform data in usable format
df1$sequence<-as.factor(paste(df1$seq, df1$condition, sep = "_"))
df1 %>% mutate(type = ifelse(condition %in% c("C1", "C4", "C5"),
"non_mixed",
                             ifelse(condition %in% c("C2", "C3", "C6"),
"mixed", NA)),
               condition = recode(condition,
                                  C1 = "4 dots",
                                  C2 = "2&4 dots",
                                  C3 = "3&4 dots",
                                  C4 = "3 dots",
                                  C5 = "2 dots",
                                  C6 = "2&3 dots"),
               sequence_length=recode(seq_len,
                                      "4"="4 dots",
                                      "2"="2 dots",
                                      "3"="3 dots")) %>%


  mutate(positions=factor(paste0("position",transitions+1)),
         positions=recode(positions,
                          position5="position_next"),
         trans = factor(paste0("trans", transitions)))%>%

  dplyr::select(on_target_pre_total,positions,sequence_length, type,
occurrence, transitions, trans,condition, subno, sequence) -> df2;df2

#learning for condition-task 4 dots
df3 <- subset(df2, transitions %in% 1:3 & condition == "4 dots")
#learning for condition-task 3 dots
df4<- subset(df2, transitions %in% 1:2 & condition == "3 dots")
#learning for condition-task 2 dots
df5<- subset(df2, transitions %in% 1:1 & condition == "2 dots")



#learning for condition-task 2&3 dots
df6 <- subset(df2, sequence_length == "3 dots"  & condition == "2&3 dots" &
transitions %in% 1:2 )
df7<-subset(df2, sequence_length == "2 dots"  & condition == "2&3 dots" &
transitions %in% 1:1 )
```

```
df8<-rbind(df6,df7)

#learning for condition-task 2&4 dots
df9 <- subset(df2, sequence_length == "4 dots"  & condition == "2&4 dots" &
transitions %in% 1:3 )
df10<-subset(df2, sequence_length == "2 dots"  & condition == "2&4 dots" &
transitions %in% 1:1 )
df11<-rbind(df9,df10)


#learning for condition-task 3&4 dots
df12 <- subset(df2, sequence_length == "4 dots"  & condition == "3&4 dots"
& transitions %in% 1:3 )
df13<-subset(df2, sequence_length == "3 dots"  & condition == "3&4 dots" &
transitions %in% 1:2 )
df14<-rbind(df12,df13)

library(data.table)
#bind the learnind data in one df
df_learning<-rbind(df14,df11,df8,df5,df4,df3)
setDT(df_learning)[ , new_subno := .GRP, by = .(subno)]

rm(list=Filter(function(x){!(x %in% c('df2','df_learning'))}, ls()))
#########################################################################

df_learning%>%
  ggplot(aes(x=occurrence, y=on_target_pre_total,
colour=positions),group=subno) +
  geom_smooth( method = "loess")+
  theme_minimal(base_size = 10)+
  facet_grid(~condition)+
  coord_cartesian(ylim = c(0, 250))+
  labs(title ="Learning rate across different tasks")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank
period")+
  ggsave('./plots/learning across conditions.png', height = 16.6,width =
16.6, units = 'cm')

########## shape of curve during occurence
formula_0<- glmer(on_target_pre_total ~ 1  + (1|subno) +
(1|sequence:subno), data=df_learning, family = poisson())
formula_1<- glmer(on_target_pre_total ~ poly(occurrence,1)  + (1|subno) +
(1|sequence:subno), data=df_learning, family = poisson())
anova(formula_0, formula_1)
formula_2<- glmer(on_target_pre_total ~ poly(occurrence,2)  + (1|subno) +
(1|sequence:subno), data=df_learning, family = poisson())
formula_3<- glmer(on_target_pre_total ~ poly(occurrence,3)  + (1|subno) +
(1|sequence:subno), data=df_learning, family = poisson())

anova(formula_1, formula_2)
anova(formula_3, formula_2)
```

I.2 R Script for modelling learning in same length tasks in Chapter 7, Section C.3.

```
library(lme4)
library(tidyverse)
library(reshape2)
library(magrittr)
library(effects)
library(sjPlot)
library(sjmisc)
library(sjlabelled)
load("data/exp4_final.Rdata")
df1 <- Df_all %>% as_tibble()
names(df1)

#transform data in usable format
df1$sequence<-as.factor(paste(df1$seq, df1$condition, sep = "_"))
df1 %>% mutate(type = ifelse(condition %in% c("C1", "C4", "C5"),
"non_mixed",
                             ifelse(condition %in% c("C2", "C3", "C6"),
"mixed", NA)),
               condition = recode(condition,
                                  C1 = "4 dots",
                                  C2 = "2&4 dots",
                                  C3 = "3&4 dots",
                                  C4 = "3 dots",
                                  C5 = "2 dots",
                                  C6 = "2&3 dots"),
               sequence_length=recode(seq_len,
                                      "4"="4 dots",
                                      "2"="2 dots",
                                      "3"="3 dots")) %>%


  mutate(positions=factor(paste0("position",transitions+1)),
         positions=recode(positions,
                          position5="position_next"),
         trans = factor(paste0("trans", transitions)))%>%

  dplyr::select(on_target_pre_total,positions,sequence_length, type,
occurrence, transitions, trans,condition, subno, sequence) -> df2;df2
############################################################



df3 <- subset(df2,type == 'non_mixed')
df4<-subset(df3, transitions %in% 1:1 & condition == "2 dots")
df5<-subset(df3, transitions %in% 1:2 & condition == "3 dots")
df6<-subset(df3, transitions %in% 1:3 & condition == "4 dots")
dfmain<-rbind(df4,df5,df6)


###########
formula_0<- glmer(on_target_pre_total ~ 1  + (1|subno) +
(1|sequence:subno), data=dfmain, family = poisson())
formula_1<- glmer(on_target_pre_total ~ poly(occurrence,1)  + (1|subno) +
(1|sequence:subno), data=dfmain, family = poisson())
anova(formula_0, formula_1)
formula_2<- glmer(on_target_pre_total ~ poly(occurrence,2)  + (1|subno) +
(1|sequence:subno), data=dfmain, family = poisson())
formula_3<- glmer(on_target_pre_total ~ poly(occurrence,3)  + (1|subno) +
(1|sequence:subno), data=dfmain, family = poisson())

anova(formula_1, formula_2)
```

```
anova(formula_3, formula_2)


formula_4<- glmer(on_target_pre_total ~ poly(occurrence,3)+sequence_length
+ (1|subno) + (1|sequence:subno), data=dfmain, family = poisson())
formula_5<- glmer(on_target_pre_total ~ poly(occurrence,3)*sequence_length
+ (1|subno) + (1|sequence:subno), data=dfmain, family = poisson())
anova(formula_3,formula_4)
anova(formula_3,formula_5)
tab_model(formula_5,p.style ="asterisk",    show.intercept = TRUE,
          show.est = TRUE,
          show.ci = 0.95,
          show.se = NULL,
          show.std = NULL,
          show.p = TRUE,
          show.stat = TRUE,
          show.df = FALSE,
          show.zeroinf = TRUE,
          show.r2 = TRUE,
          show.re.var = TRUE,
          show.ngroups = TRUE,
          show.fstat = TRUE,
          show.aic = TRUE,
          show.loglik = TRUE,
          show.obs = TRUE,
          show.reflvl = TRUE)


dfmain$fitted_model<-fitted(formula_5)

dfmain%>%
  ggplot(aes(x=occurrence, y=fitted_model),group=subno) +
  geom_smooth( method = "lm", formula = y ~ poly(x,3))+
  theme_minimal(base_size = 10)+
  facet_wrap(~condition)+
  labs(title ="Fit of Model for learning rate across same length tasks")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank
period")+
  ggsave('./plots/fit  same length_condition.png', height = 16.6,width =
16.6, units = 'cm')

dfmain%>%
  ggplot(aes(x=occurrence, y=fitted_model, colour=condition),group=subno) +
  geom_smooth( method = "lm", formula = y ~ poly(x,3))+
  theme_minimal(base_size = 10)+
  labs(title ="Fit of Model for learning rate across same length tasks")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank
period")+
  ggsave('./plots/fit  same length_condition.png', height = 16.6,width =
16.6, units = 'cm')
```

I.3 R Script for modelling learning in mixed length tasks in Chapter 7, Section C.4.

```
library(lme4)
library(tidyverse)
```

```
library(reshape2)
library(magrittr)
library(effects)
library(sjPlot)
library(sjmisc)
library(sjlabelled)
load("data/exp4_final.Rdata")
df1 <- Df_all %>% as_tibble()
names(df1)

#transform data in usable format
df1$sequence<-as.factor(paste(df1$seq, df1$condition, sep = "_"))
df1 %>% mutate(type = ifelse(condition %in% c("C1", "C4", "C5"),
"non_mixed",
                             ifelse(condition %in% c("C2", "C3", "C6"),
"mixed", NA)),
               condition = recode(condition,
                                  C1 = "4 dots",
                                  C2 = "2&4 dots",
                                  C3 = "3&4 dots",
                                  C4 = "3 dots",
                                  C5 = "2 dots",
                                  C6 = "2&3 dots"),
               sequence_length=recode(seq_len,
                                      "4"="4 dots",
                                      "2"="2 dots",
                                      "3"="3 dots")) %>%


  mutate(positions=factor(paste0("position",transitions+1)),
         positions=recode(positions,
                          position5="position_next"),
         trans = factor(paste0("trans", transitions)))%>%

  dplyr::select(on_target_pre_total,positions,sequence_length, type,
occurrence, transitions, trans,condition, subno, sequence) -> df2;df2
############################################################


#learning for condition-task 2&3 dots
df6 <- subset(df2, sequence_length == "3 dots"  & condition == "2&3 dots" &
transitions %in% 1:2 )
df7<-subset(df2, sequence_length == "2 dots"  & condition == "2&3 dots" &
transitions %in% 1:1 )
df8<-rbind(df6,df7)

#learning for condition-task 2&4 dots
df9 <- subset(df2, sequence_length == "4 dots"  & condition == "2&4 dots" &
transitions %in% 1:3 )
df10<-subset(df2, sequence_length == "2 dots"  & condition == "2&4 dots" &
transitions %in% 1:1 )
df11<-rbind(df9,df10)
#learning for condition-task 2&4 dots
df12 <- subset(df2, sequence_length == "4 dots"  & condition == "3&4 dots"
& transitions %in% 1:3 )
df13<-subset(df2, sequence_length == "3 dots"  & condition == "3&4 dots" &
transitions %in% 1:2 )
df14<-rbind(df12,df13)
df_mixed<-rbind(df14,df11, df8)
formula_3<- glmer(on_target_pre_total ~ poly(occurrence,3)  + (1|subno) +
(1|sequence:subno), data=df_mixed, family = poisson())
```

```
formula_4<- glmer(on_target_pre_total ~ poly(occurrence,3)+sequence_length
+ (1|subno) + (1|sequence:subno), data=df_mixed, family = poisson())

formula_5<- glmer(on_target_pre_total ~ poly(occurrence,3)*sequence_length
+ (1|subno) + (1|sequence:subno), data=df_mixed, family = poisson())

anova(formula_3,formula_4)
anova(formula_4,formula_5)

summary(formula_5)
tab_model(formula_5,p.style ="asterisk",    show.intercept = TRUE,
          show.est = TRUE,
          show.ci = 0.95,
          show.se = NULL,
          show.std = NULL,
          show.p = TRUE,
          show.stat = TRUE,
          show.df = FALSE,
          show.zeroinf = TRUE,
          show.r2 = TRUE,
          show.re.var = TRUE,
          show.ngroups = TRUE,
          show.fstat = TRUE,
          show.aic = TRUE,
          show.loglik = TRUE,
          show.obs = TRUE,
          show.reflvl = TRUE)


df_mixed$fitted_model<-fitted(formula_5)

df_mixed%>%
  ggplot(aes(x=occurrence, y=fitted_model,
colour=sequence_length),group=subno) +
  geom_smooth( method = "lm", formula = y ~ poly(x,3))+
  theme_minimal(base_size = 10)+
  facet_wrap(~condition)+
  coord_cartesian(ylim = c(0, 250))+
  labs(title ="Fit of Model for learning rate across mixed length tasks")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank
period")+
  ggsave('./plots/ 2 mixed length_condition.png', height = 16.6,width =
16.6, units = 'cm')

df_mixed%>%
  ggplot(aes(x=occurrence, y=fitted_model),group=subno) +
  geom_smooth( method = "lm", formula = y ~ poly(x,3))+
  theme_minimal(base_size = 10)+
  facet_wrap(~condition)+
  coord_cartesian(ylim = c(0, 250))+
  labs(title ="Fit of Model for learning rate across mixed length tasks")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank
period")+
  ggsave('./plots/ 1 mixed length_condition.png', height = 16.6,width =
16.6, units = 'cm')
```

I.4 R Script for modelling learning in mixed length tasks vs same length tasks in Chapter 7, Section C.5.

```
library(lme4)
```

```
library(tidyverse)
library(reshape2)
library(magrittr)
library(effects)
library(sjPlot)
library(sjmisc)
library(sjlabelled)
load("data/exp4_final.Rdata")
df1 <- Df_all %>% as_tibble()
names(df1)

#transform data in usable format
#transform data in usable format
df1$sequence<-as.factor(paste(df1$seq, df1$condition, sep = "_"))
df1 %>% mutate(type = ifelse(condition %in% c("C1", "C4", "C5"),
"non_mixed",
                             ifelse(condition %in% c("C2", "C3", "C6"),
"mixed", NA)),
               condition = recode(condition,
                                  C1 = "4 dots",
                                  C2 = "2&4 dots",
                                  C3 = "3&4 dots",
                                  C4 = "3 dots",
                                  C5 = "2 dots",
                                  C6 = "2&3 dots"),
               sequence_length=recode(seq_len,
                                      "4"="4 dots",
                                      "2"="2 dots",
                                      "3"="3 dots")) %>%


  mutate(positions=factor(paste0("position",transitions+1)),
         positions=recode(positions,
                          position5="position_next"),
         trans = factor(paste0("trans", transitions)))%>%

  dplyr::select(on_target_pre_total,positions,sequence_length, type,
occurrence, transitions, trans,condition, subno, sequence) -> df2;df2

#learning for condition-task 4 dots
df3 <- subset(df2, transitions %in% 1:3 & condition == "4 dots")
#learning for condition-task 3 dots
df4<- subset(df2, transitions %in% 1:2 & condition == "3 dots")
#learning for condition-task 2 dots
df5<- subset(df2, transitions %in% 1:1 & condition == "2 dots")



#learning for condition-task 2&3 dots
df6 <- subset(df2, sequence_length == "3 dots"  & condition == "2&3 dots" &
transitions %in% 1:2 )
df7<-subset(df2, sequence_length == "2 dots"  & condition == "2&3 dots" &
transitions %in% 1:1 )
df8<-rbind(df6,df7)

#learning for condition-task 2&4 dots
df9 <- subset(df2, sequence_length == "4 dots"  & condition == "2&4 dots" &
transitions %in% 1:3 )
df10<-subset(df2, sequence_length == "2 dots"  & condition == "2&4 dots" &
transitions %in% 1:1 )
df11<-rbind(df9,df10)
```

```
#learning for condition-task 3&4 dots
df12 <- subset(df2, sequence_length == "4 dots"  & condition == "3&4 dots"
& transitions %in% 1:3 )
df13<-subset(df2, sequence_length == "3 dots"  & condition == "3&4 dots" &
transitions %in% 1:2 )
df14<-rbind(df12,df13)

library(data.table)
#bind the learnind data in one df
df_learning<-rbind(df14,df11,df8,df5,df4,df3)
setDT(df_learning)[ , new_subno := .GRP, by = .(subno)]
##############
formula_3<- glmer(on_target_pre_total ~ poly(occurrence,3)  + (1|subno) +
(1|sequence:subno), data=df_learning, family = poisson())
formula_4<- glmer(on_target_pre_total ~ poly(occurrence,3) +type +
(1|subno) + (1|sequence:subno), data=df_learning, family = poisson())
formula_41<- glmer(on_target_pre_total ~ poly(occurrence,3) *type +
(1|subno) + (1|sequence:subno), data=df_learning, family = poisson())
anova(formula_3,formula_4)
anova(formula_41, formula_3)
formula_5<- glmer(on_target_pre_total ~ poly(occurrence,3) *type +
sequence_length + (1|subno) + (1|sequence:subno), data=df_learning, family
= poisson())
formula_6<- glmer(on_target_pre_total ~ poly(occurrence,3) *type *
sequence_length + (1|subno) + (1|sequence:subno), data=df_learning, family
= poisson())
anova(formula_4,formula_41)

anova(formula_41, formula_5)
anova( formula_5, formula_6)


tab_model(formula_6,p.style = "asterisk",    show.intercept = TRUE,
          show.est = TRUE,
          show.ci = 0.95,
          show.se = NULL,
          show.std = NULL,
          show.p = TRUE,
          show.stat = TRUE,
          show.df = FALSE,
          show.zeroinf = TRUE,
          show.r2 = TRUE,
          show.re.var = TRUE,
          show.ngroups = TRUE,
          show.fstat = TRUE,
          show.aic = TRUE,
          show.loglik = TRUE,
          show.obs = TRUE,
          show.reflvl = TRUE)

summary(formula_6)
df_learning$fitted_model<-fitted(formula_6)


df_learning%>%
  ggplot(aes(x=occurrence, y=fitted_model,
colour=sequence_length),group=subno) +
  geom_smooth( method = "lm", formula = y ~ poly(x,3))+
  theme_minimal(base_size = 10)+
```

```
124     facet_wrap(~type)+
125     coord_cartesian(ylim = c(0, 250))+
126     labs(title ="Fit of Model for learning rate across tasks")+
127     xlab("Number of repetition of item - exposure")+
128     ylab("Number of eye-samples on  target position \n during  blank
129   period")+
130     ggsave('./plots/ nonmixed mixed length_condition.png', height =
131   16.6,width = 16.6, units = 'cm')
132
133   rm(list=Filter(function(x){!(x %in% c('df2','df_learning'))}, ls()))
134
```

Appendix J - R Script for plot in Chapter 8

J.1 R script that generates in one graph the percentage of the learning rate in the task of 2 dots for the GP3 and the EyeLink 1000.

```
library(tidyverse)
library(reshape2)
library(magrittr)
library(dplyr)


load("./data/exp3_final.Rdata")
df1 <- Df_all %>% as_tibble()
names(df1)

load("./data/exp4_final.Rdata")
df2 <- Df_all %>% as_tibble()
names(df2)
#transform data in usable format
str(df2)
df2$seq_len<-as.factor(df2$seq_len)
df2 %>% mutate(type = ifelse(condition %in% c("C1", "C2"), "non_mixed",
                             ifelse(condition %in% c("C3"), "mixed", NA)),
               condition = recode(condition,
                                  C1 = "3 dots",
                                  C2 = "2 dots",
                                  C3 = "2&3dots"),
               sequence_length=recode(seq_len,
                                      "2"="2 dots",
                                      "3"="3 dots")) %>%


  mutate(positions=factor(paste0("position",transitions+1)),
         positions=recode(positions,
                          position5="position_next"),
         trans = factor(paste0("trans", transitions)))%>%

  select(on_target_pre_total,positions,sequence_length, occurrence,
transitions, trans,condition, subno, seq,type) -> df4;df4

##################
df1$sequence<-as.factor(paste(df1$seq, df1$condition, sep = "_"))
df1 %>% mutate(sequence_length=recode(seq_len,
                                      "4"="4 dots",
                                      "2"="2 dots",
                                      "3"="3 dots")) %>%


  mutate(positions=factor(paste0("position",transitions+1)),
         positions=recode(positions,
                          position5="position_next"),
         trans = factor(paste0("trans", transitions)))%>%

  dplyr::select(on_target_pre_total,positions,sequence_length, occurrence,
transitions, trans,condition, subno) -> df3;df3
#########################################################

library(data.table)
setDT(df3)[ , new_subno := .GRP, by = .(subno)]
```

```
setDT(df4)[ , new_subno := .GRP, by = .(subno)]

dfgp3<-subset(df3, transitions %in% 1:1 & condition == "C2")
dfsr<-subset(df4, transitions %in% 1:1 & condition == "2 dots")


dfgp3$percentage_on_target<-(dfgp3$on_target_pre_total/45)*100

dfsr$percentage_on_target<-(dfsr$on_target_pre_total/750)*100


ggplot(data=dfgp3,aes(y = percentage_on_target, x = occurrence) )+
  geom_smooth(size= 1.2, method='lm', formula = y ~ poly(x,3))+
  geom_smooth(data=dfsr,size= 1.2, method='lm', formula = y ~ poly(x,3),
colour= 'red')+
  theme_minimal(base_size = 10)+
  ylim(0,100)+
  labs(title ="Learning rate in 2 dots tasks with \nGP3(blue-line)and
Eyelink 1000 (red-line)")+
  xlab("Number of repetition of item - exposure")+
  ylab("Percentage of eye samples on target position \n during  blank
period")

ggsave(' percentage 2 dots across eye trackers.png', height = 15, width =
15, units = 'cm')
```

J2.R script that generates t-tests and graphs for spatial acuity analysis.

```

#load GP3
load('./fixations_gp3_comp.Rdata')

gp3<-Df_all

#load sr
load('./fixations_sr_comp.Rdata')
sr<-Df_all



#filter data
gp3<-gp3%>%filter(condition %in% c("C2"))%>%
  mutate(condition=recode(condition, C2= "2 dots" ))%>%
  filter(on_target_post != 0)

sr<-sr%>%filter(condition %in% c("C5"))%>%
  mutate(condition=recode(condition, C5= "2 dots" ))%>%
  filter(on_target_post != 0)

########
all = bind_rows(
  mutate(gp3, eye_tracker = "gp3"),
  mutate(sr, eye_tracker = "sr")
)

##################################################


```

```
 gp3%>%
  summarise(mean = mean(on_target_post), n = n(), sd=sd(on_target_post))

 sr%>%
   summarise(mean = mean(on_target_post), n = n(), sd=sd(on_target_post))



 ##duration
 gp3%>%
   summarise(mean = mean(duration), n = n(), sd=sd(duration))

sr%>%
   summarise(mean = mean(duration), n = n(), sd=sd(duration))

 ###distance from target
gp3%>%
  summarise(mean = mean(distance_sample_to_current_target), n = n(),
sd=sd(distance_sample_to_current_target))

sr%>%
  summarise(mean = mean(distance_sample_to_current_target), n = n(),
sd=sd(distance_sample_to_current_target))
###########
#visualise
library(psyntur)
t_test(data=all,duration~eye_tracker)

t_test(data=all,distance_sample_to_current_target~eye_tracker)
# distance of fixation from target
ggplot(all, aes(y = distance_sample_to_current_target,  x = eye_tracker,
color= eye_tracker)) +
  geom_boxplot()+theme_minimal()+
  scale_color_brewer(palette="Dark2")+
  labs(title ="Distance of fixations within the AOI from target")+
  xlab(" Eye-tracker")+
  ylab("Distance from target in pixels")+
  ggsave('./distance from target.png', height = 10,width = 17, units =
'cm')

# duration of fixation from target

ggplot(all, aes(y = duration,  x = eye_tracker, color= eye_tracker)) +
  geom_boxplot()+theme_minimal()+
  scale_color_brewer(palette="Dark2")+
  labs(title ="Duration of fixations within the AOI")+
  xlab(" Eye-tracker")+
  ylab("Duration of fixations in ms")+
  ggsave('./duration from target.png', height = 10,width = 17, units =
'cm')



 #####check per trial amount of fixations


gp1<-gp3%>%
  drop_na()%>%
  group_by(subno,condition, seq,target,occurrence)%>%
  mutate(total_post_fixations=sum(on_target_post),
         total_duration= sum(duration)/total_post_fixations)%>%
```

```
92
93  select(subno,condition,target,trial_order_id,occurrence,ID,total_post_fixat
94  ions,
95           total_duration)%>%
96    ungroup()%>%
97    distinct()
98
99  sr1<-sr%>%
100   drop_na()%>%
101   group_by(subno,condition, seq,target,occurrence)%>%
102   mutate(total_post_fixations=sum(on_target_post),
103          total_duration= sum(duration)/total_post_fixations,
104
105 total_distance=sum(distance_sample_to_current_target)/total_post_fixations)
106 %>%
107
108 select(subno,condition,target,trial_order_id,occurrence,ID,total_post_fixat
109 ions,
110          total_duration,total_distance)%>%
111   ungroup()%>%
112   distinct()
113
114 gp1%>%
115   summarise(mean = mean(total_post_fixations), n = n(),
116 sd=sd(total_post_fixations))
117
118 sr1%>%
119   summarise(mean = mean(total_post_fixations), n = n(),
120 sd=sd(total_post_fixations))
121
122 all1 = bind_rows(
123   mutate(gp1, eye_tracker = "gp3"),
124   mutate(sr1, eye_tracker = "sr")
125 )
126
127 t_test(data=all1, total_post_fixations~eye_tracker)
```

Appendix K- Individual Differences in learning performance in 6 tasks

K.1 Figure of individual differences in 4 dots task.

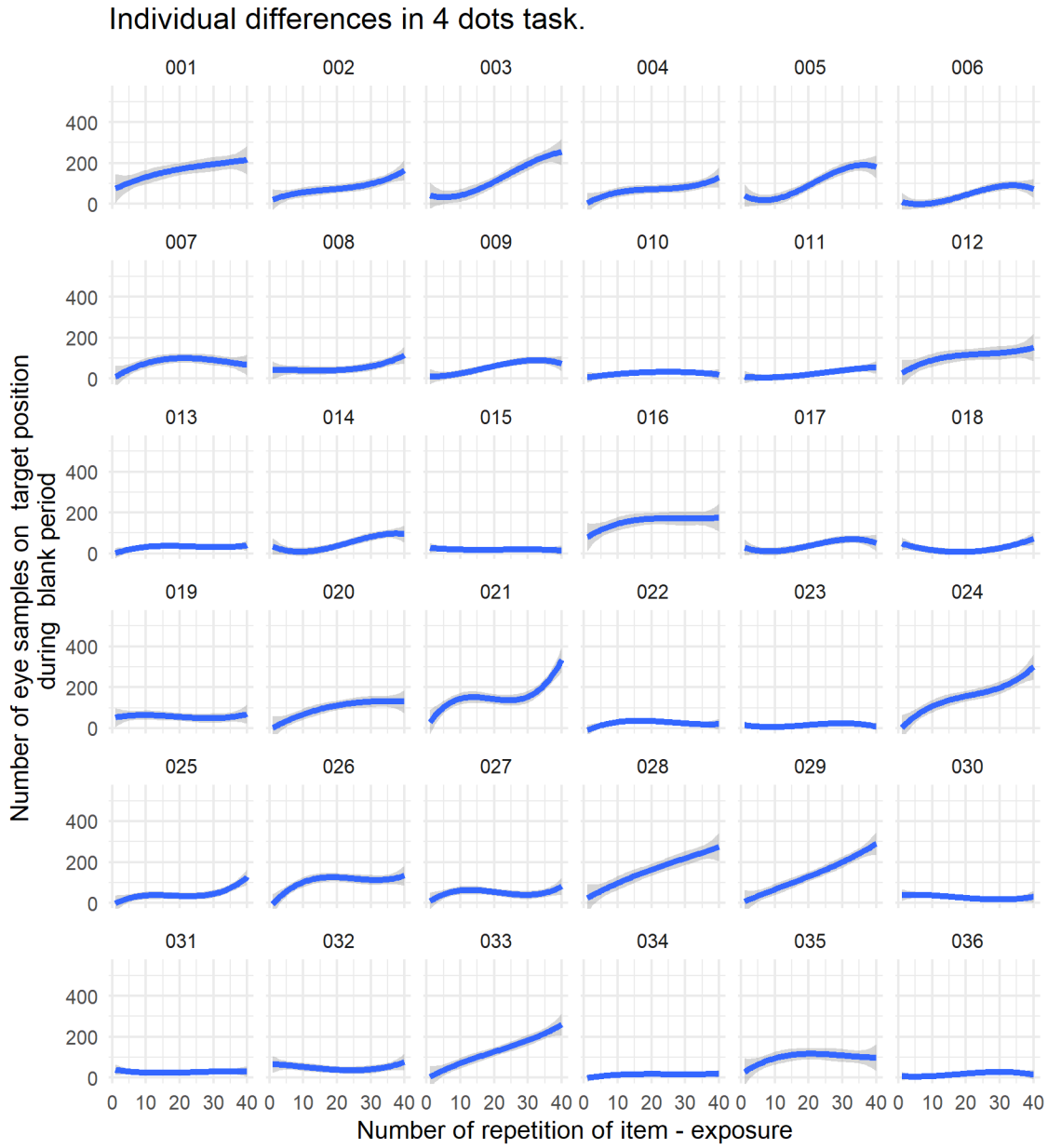


Figure K.1 This plot shows the learning across sequences in the 4 dots task, for each individual subject (001-036). On x-axis is the number of occurrence / exposure and on the y- axis is the number of eye-samples on the target location during the 750ms blank period. Reminder that there is a maximum of 750 eye-samples per trial.

K.2 Figure of individual differences in 3 dots task.

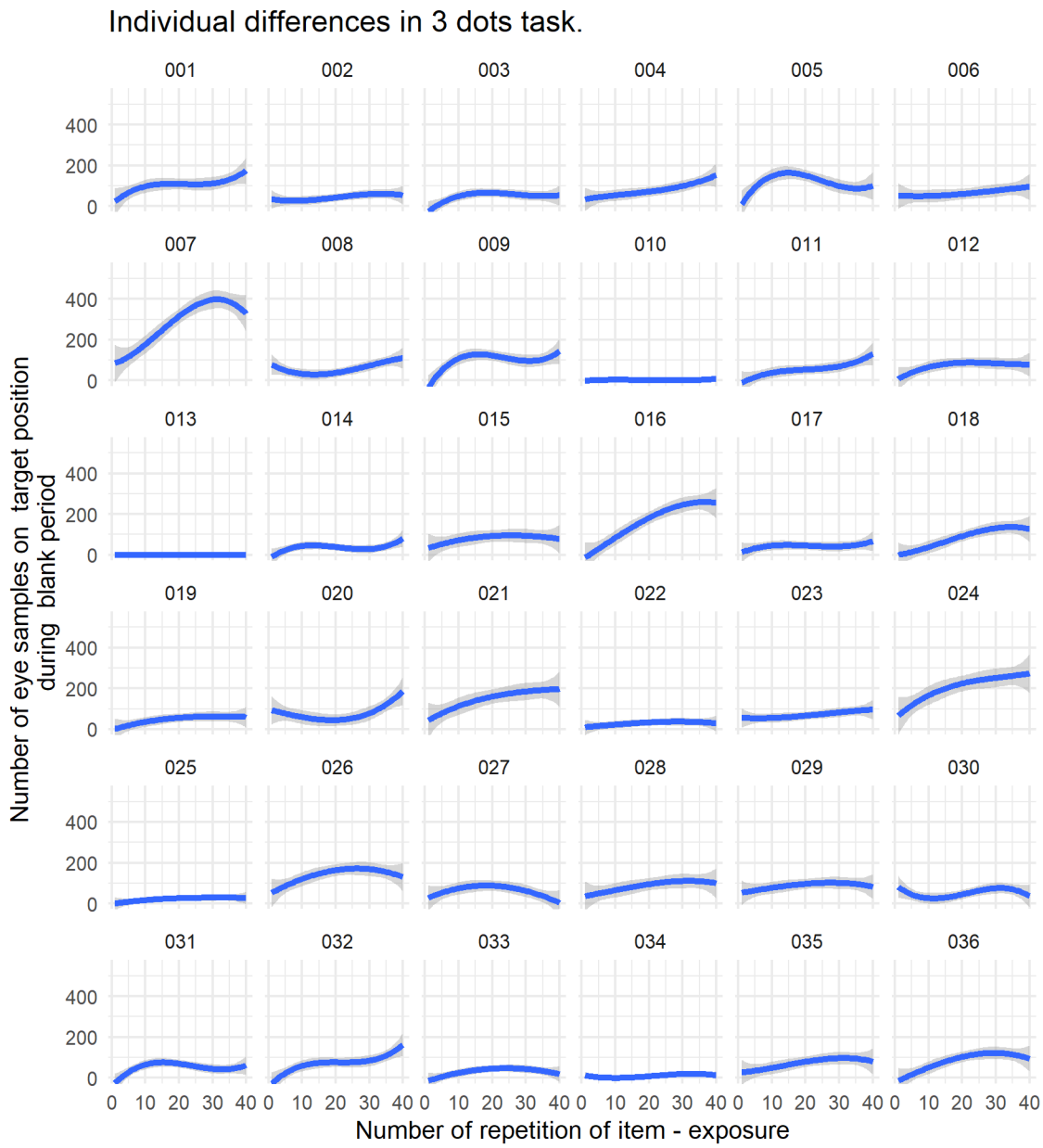


Figure K.2 This plot shows the learning across sequences in the 3 dots task, for each individual subject (001-036). On x-axis is the number of occurrence / exposure and on the y- axis is the number of eye-samples on the target location during the 750ms blank period. Reminder that there is a maximum of 750 eye-samples per trial.

K.3 Figure of individual differences in 3&4 dots task.

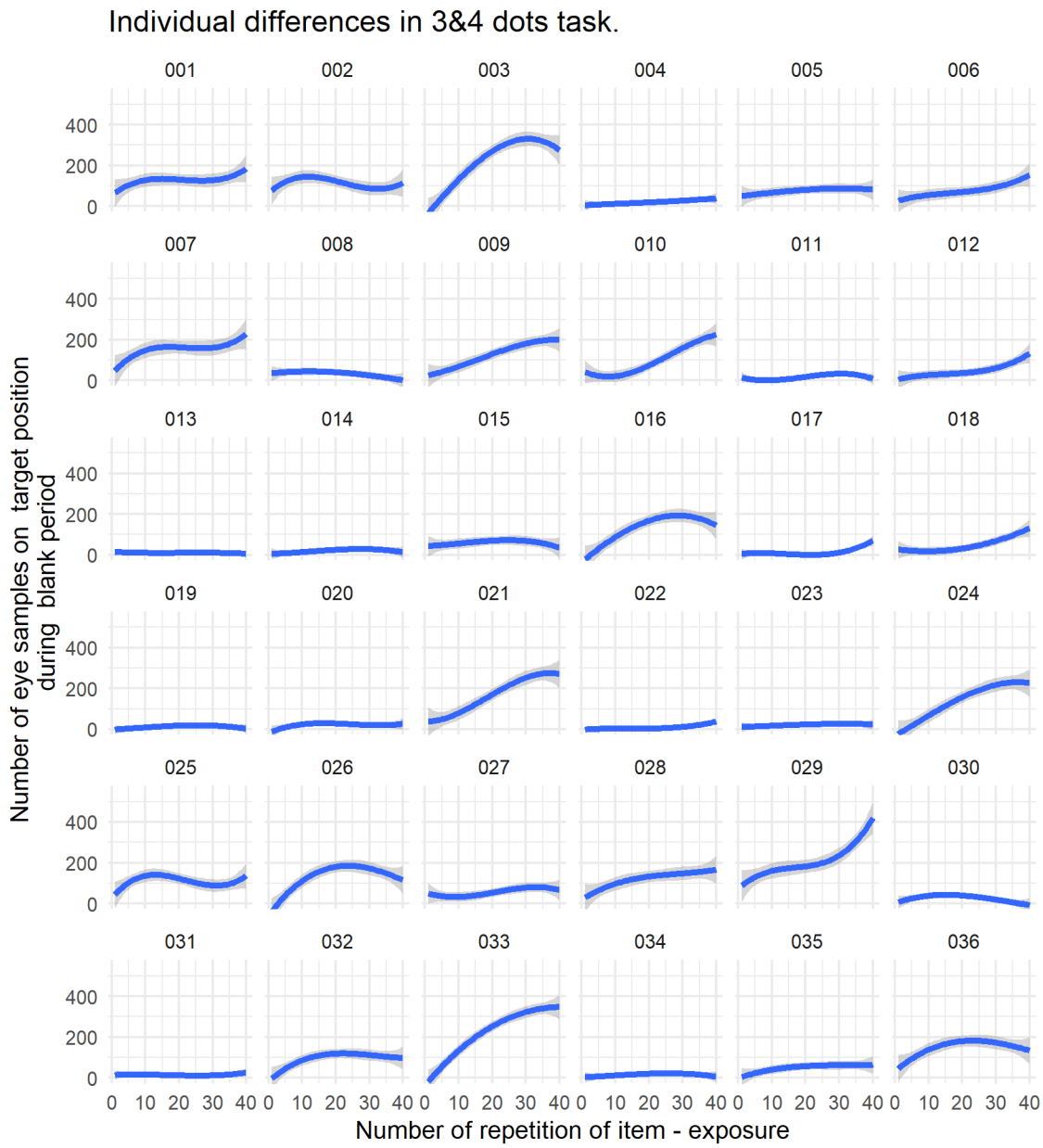


Figure K.3 This plot shows the learning across sequences in the 3&4 dots task, for each individual subject (001-036). On x-axis is the number of occurrence / exposure and on the y- axis is the number of eye-samples on the target location during the 750ms blank period. Reminder that there is a maximum of 750 eye-samples per trial.

K.4 Figure of individual differences in 2 dots task.

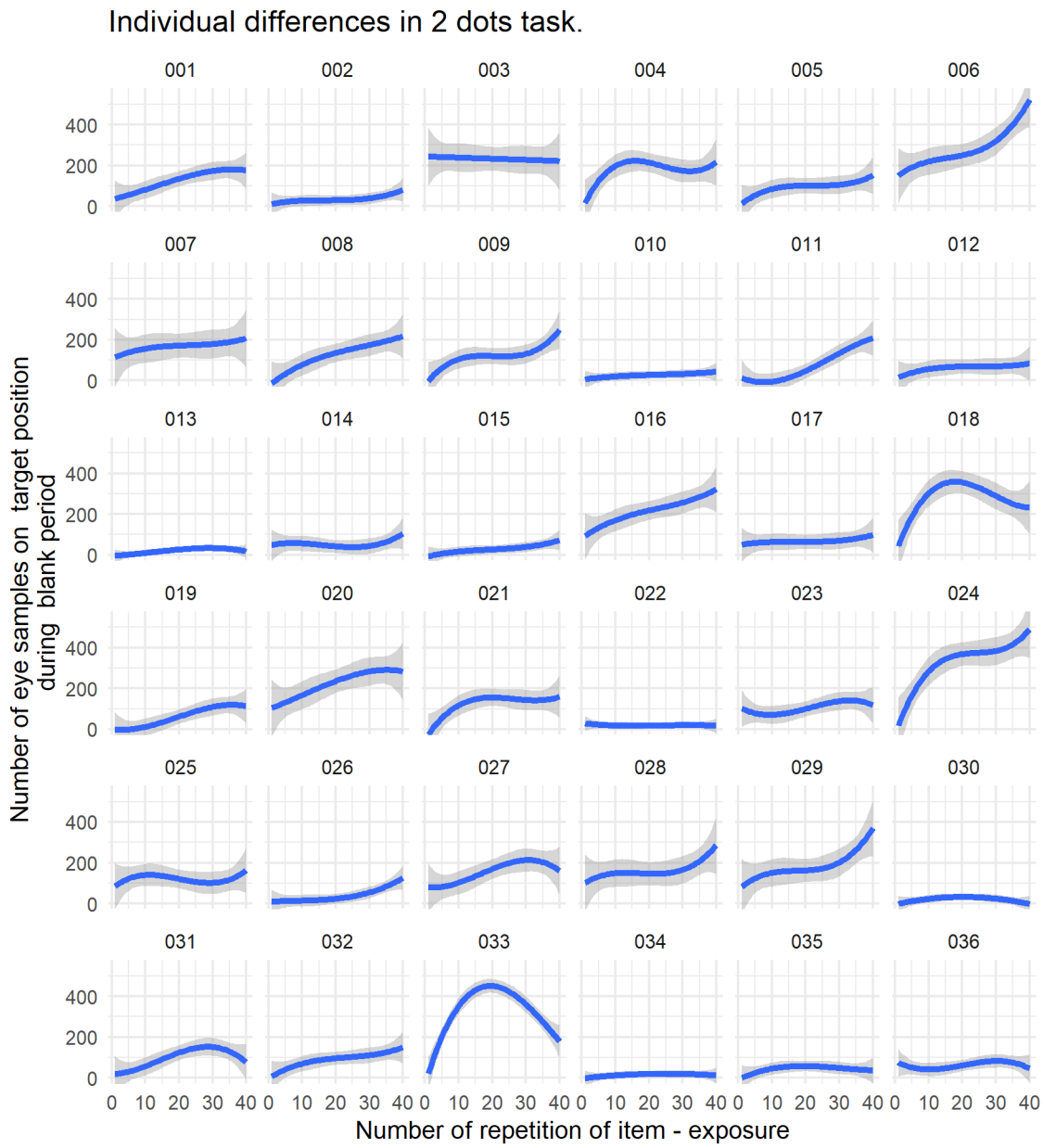


Figure K.4 This plot shows the learning across sequences in the 2 dots task, for each individual subject (001-036). On x-axis is the number of occurrence / exposure and on the y- axis is the number of eye-samples on the target location during the 750ms blank period. Reminder that there is a maximum of 750 eye-samples per trial.

K.5 Figure of individual differences in 2&3 dots task.

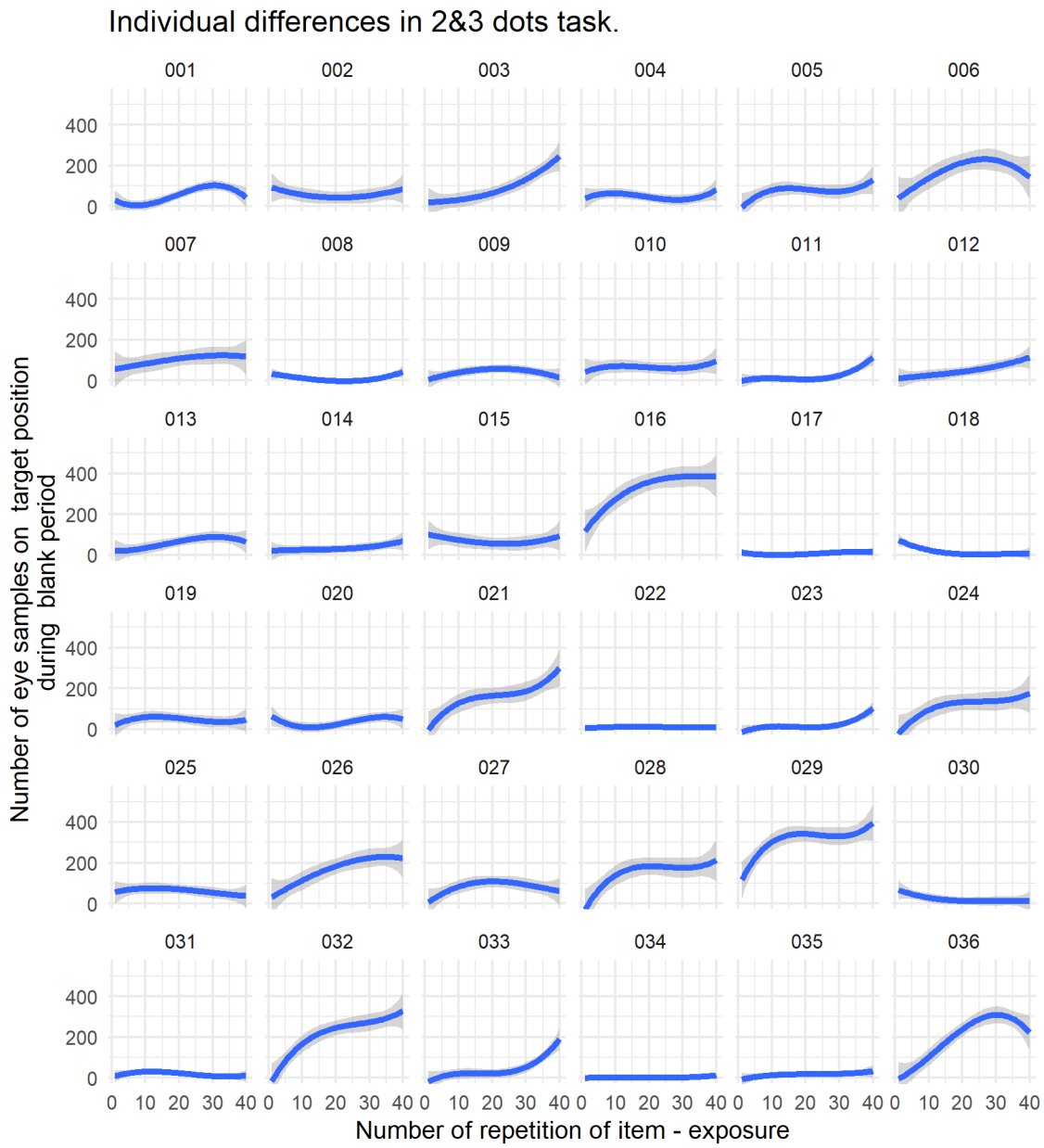


Figure K.5 This plot shows the learning across sequences in the 2&3 dots task, for each individual subject (001-036). On x-axis is the number of occurrence / exposure and on the y- axis is the number of eye-samples on the target location during the 750ms blank period. Reminder that there is a maximum of 750 eye-samples per trial.

K.6 Figure of individual differences in 2&4 dots task.

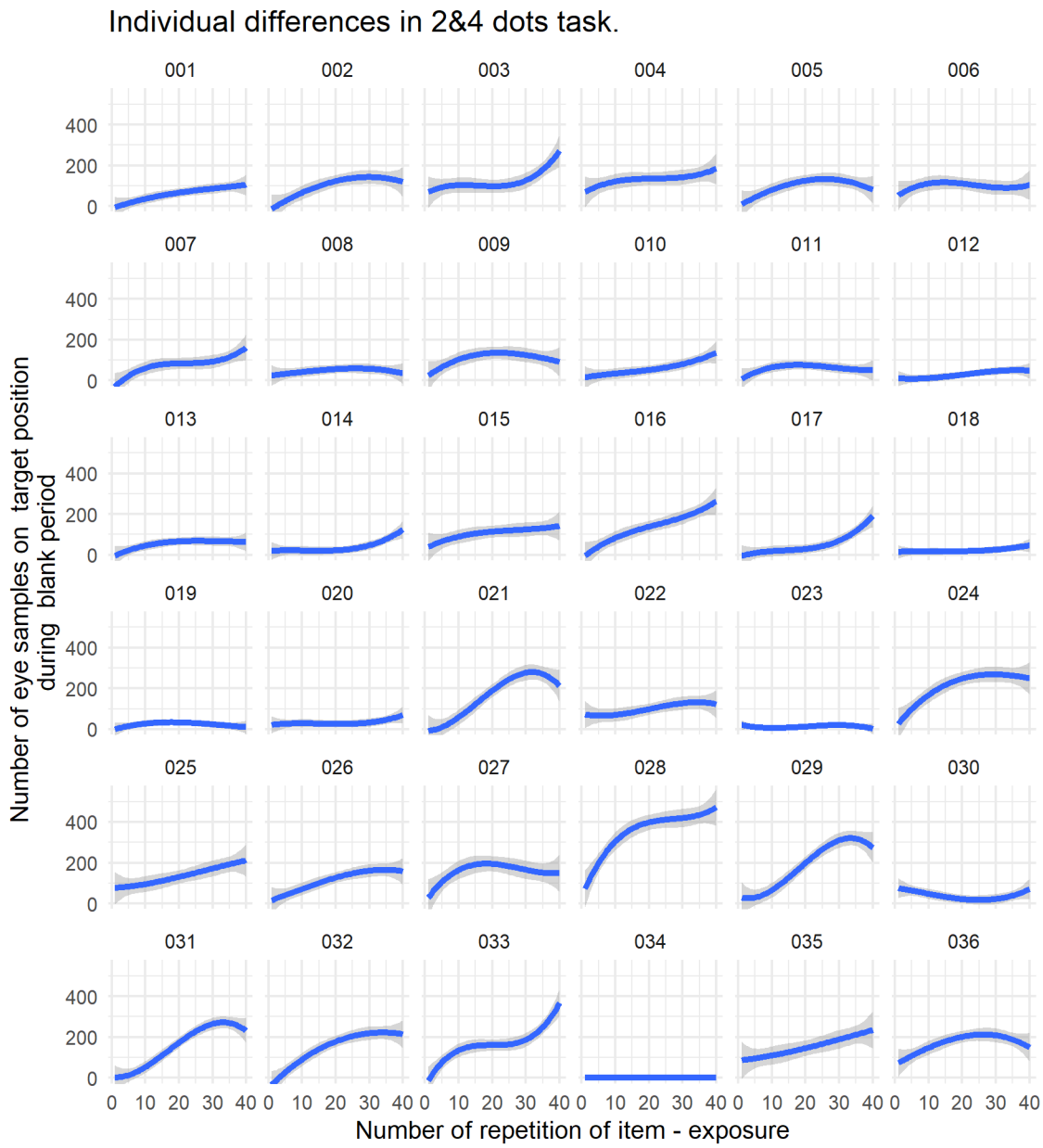


Figure K.6 This plot shows the learning across sequences in the 2&4 dots task, for each individual subject (001-036). On x-axis is the number of occurrence / exposure and on the y- axis is the number of eye-samples on the target location during the 750ms blank period. Reminder that there is a maximum of 750 eye-samples per trial.

Appendix L - R Script for individual differences in Appendix K

L.1 R Script for the plot of individual differences for the 6 different tasks.

```
#Individual differences plots in 6 tasks
library(tidyverse)
library(reshape2)
load("data/exp4_final.Rdata")
df1 <- Df_all %>% as_tibble()
names(df1)

#transform data in usable format
df1$sequence<-as.factor(paste(df1$seq, df1$condition, sep = "_"))
df1 %>% mutate(type = ifelse(condition %in% c("C1", "C4", "C5"),
"non_mixed",
                             ifelse(condition %in% c("C2", "C3", "C6"),
"mixed", NA)),
               condition = recode(condition,
                                  C1 = "4 dots",
                                  C2 = "2&4 dots",
                                  C3 = "3&4 dots",
                                  C4 = "3 dots",
                                  C5 = "2 dots",
                                  C6 = "2&3 dots"),
               sequence_length=recode(seq_len,
                                      "4"="4 dots",
                                      "2"="2 dots",
                                      "3"="3 dots")) %>%


  mutate(positions=factor(paste0("position",transitions+1)),
         positions=recode(positions,
                          position5="position_next"),
         trans = factor(paste0("trans", transitions)))%>%

  dplyr::select(on_target_pre_total,positions,sequence_length, type,
occurrence, transitions, trans,condition, subno, sequence) -> df2;df2
options(scipen=999)

# sequence length on same length tasks
df_nonmixed <- subset(df2,type == 'non_mixed')
# sequence length on mixed length tasks
df_mixed <-subset(df2,type == 'mixed')

#select meaningful transitions
sl_data<-df_nonmixed %>% filter((transitions %in% 1:1 & condition == "2
dots") |
                                  (transitions %in% 1:2 & condition == "3
dots")|
                                  (transitions %in% 1:3 & condition == "4
dots"))

ml_data <- df_mixed %>% filter((sequence_length == "3 dots"  & condition ==
"2&3 dots" & transitions %in% 1:2 ) |
                                 (sequence_length == "2 dots"  & condition
== "2&3 dots" & transitions %in% 1:1 ) |
                                 (sequence_length == "4 dots"  & condition
== "2&4 dots" & transitions %in% 1:3 ) |
```

```
                                    (sequence_length == "2 dots"  & condition
== "2&4 dots" & transitions %in% 1:1 ) |
                                    (sequence_length == "4 dots"  & condition
== "3&4 dots" & transitions %in% 1:3 ) |
                                    (sequence_length == "3 dots"  & condition
== "3&4 dots" & transitions %in% 1:2 ))

#data for 6 tasks, only learning transitions included
df_learning<-rbind(sl_data,ml_data)

#individual differences for 2 dots
df_learning %>%
  filter(condition == "2 dots")%>%
  ggplot(aes(y = on_target_pre_total, x = occurrence) )+
  geom_smooth(size= 1.2, method='lm', formula = y ~ poly(x,3))+
  facet_wrap(~subno)+
  theme_minimal(base_size = 10)+
  coord_cartesian(ylim=c(0,550))+
  labs(title ="Individual differences in 2 dots task.")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye samples on  target position \n during  blank
period")+
  ggsave('sub 2.png', height = 16.5, width = 15, units = 'cm')


#individual differences for 3 dots
df_learning %>%
  filter(condition == "3 dots")%>%
  ggplot(aes(y = on_target_pre_total, x = occurrence) )+
  geom_smooth(size= 1.2, method='lm', formula = y ~ poly(x,3))+
  facet_wrap(~subno)+
  theme_minimal(base_size = 10)+
  coord_cartesian(ylim=c(0,550))+
  labs(title ="Individual differences in 3 dots task.")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye samples on  target position \n during  blank
period")+
  ggsave('sub 3.png', height = 16.5, width = 15, units = 'cm')

#individual differences for 4 dots
df_learning %>%
  filter(condition == "4 dots")%>%
  ggplot(aes(y = on_target_pre_total, x = occurrence) )+
  geom_smooth(size= 1.2, method='lm', formula = y ~ poly(x,3))+
  facet_wrap(~subno)+
  theme_minimal(base_size = 10)+
  coord_cartesian(ylim=c(0,550))+
  labs(title ="Individual differences in 4 dots task.")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye samples on  target position \n during  blank
period")+
  ggsave('sub 4.png', height = 16.5, width = 15, units = 'cm')

#individual differences for 2&3 dots
df_learning %>%
  filter(condition == "2&3 dots")%>%
  ggplot(aes(y = on_target_pre_total, x = occurrence) )+
  geom_smooth(size= 1.2, method='lm', formula = y ~ poly(x,3))+
  facet_wrap(~subno)+
  theme_minimal(base_size = 10)+
  coord_cartesian(ylim=c(0,550))+
```

```
116     labs(title ="Individual differences in 2&3 dots task.")+
117     xlab("Number of repetition of item - exposure")+
118     ylab("Number of eye samples on  target position \n during  blank
119   period")+
120     ggsave('sub 2&3.png', height = 16.5, width = 15, units = 'cm')
121
122   #individual differences for 2&4 dots
123   df_learning %>%
124     filter(condition == "2&4 dots")%>%
125     ggplot(aes(y = on_target_pre_total, x = occurrence) )+
126     geom_smooth(size= 1.2, method='lm', formula = y ~ poly(x,3))+
127     facet_wrap(~subno)+
128     theme_minimal(base_size = 10)+
129     coord_cartesian(ylim=c(0,550))+
130     labs(title ="Individual differences in 2&4 dots task.")+
131     xlab("Number of repetition of item - exposure")+
132     ylab("Number of eye samples on  target position \n during  blank
133   period")+
134     ggsave('sub 2&4.png', height = 16.5, width = 15, units = 'cm')
135
136   #individual differences for 3&4 dots
137   df_learning %>%
138     filter(condition == "3&4 dots")%>%
139     ggplot(aes(y = on_target_pre_total, x = occurrence) )+
140     geom_smooth(size= 1.2, method='lm', formula = y ~ poly(x,3))+
141     facet_wrap(~subno)+
142     theme_minimal(base_size = 10)+
143     coord_cartesian(ylim=c(0,550))+
144     labs(title ="Individual differences in 3&4 dots task.")+
145     xlab("Number of repetition of item - exposure")+
146     ylab("Number of eye samples on  target position \n during  blank
147   period")+
148     ggsave('sub 3&4.png', height = 16.5, width = 15, units = 'cm')
149
```

Appendix M - Binomial Analysis (R Scripts and results)

M.1 R Script for creating the learning threshold for binomial modelling.

```
library(lme4)
library(tidyverse)
library(reshape2)
library(magrittr)
library(effects)
library(sjPlot)
library(sjmisc)
library(sjlabelled)
load("data/exp4_final.Rdata")
df1 <- Df_all %>% as_tibble()
names(df1)

#transform data in usable format
df1$sequence<-as.factor(paste(df1$seq, df1$condition, sep = "_"))
df1 %>% mutate(type = ifelse(condition %in% c("C1", "C4", "C5"),
"non_mixed",
                             ifelse(condition %in% c("C2", "C3", "C6"),
"mixed", NA)),
               condition = recode(condition,
                                  C1 = "4 dots",
                                  C2 = "2&4 dots",
                                  C3 = "3&4 dots",
                                  C4 = "3 dots",
                                  C5 = "2 dots",
                                  C6 = "2&3 dots"),
               sequence_length=recode(seq_len,
                                      "4"="4 dots",
                                      "2"="2 dots",
                                      "3"="3 dots")) %>%


  mutate(positions=factor(paste0("position",transitions+1)),
         positions=recode(positions,
                          position5="position_next"),
         trans = factor(paste0("trans", transitions)))%>%

  dplyr::select(on_target_pre_total,positions,sequence_length, type,
occurrence, transitions, trans,condition, subno, sequence) -> df2;df2
############################################################
#get  eyesamples on first occurence for 1st transition for every
#sequence (ABCD) and all 36 subjects (864 in totall)

df3<-df2 %>% filter(occurrence== 1 & transitions ==1)
mean(df3$on_target_pre_total)
# mean =13.21 so threshold 25 samples

#create data frame for binomial Glmms
df2<-df2%>% mutate(learning = ifelse(on_target_pre_total>= 25, "1", #yes
                          ifelse(on_target_pre_total<25, "0", NA))) #no
```

M.2 R Script for binomial modelling of positioning effects in learning.

```
#run the getting threshold.R script first

options(scipen=999)

#use the df_learning

p1<-glmer(on_target_pre_total ~ poly(occurrence,3)  + (1|subno) +
(1|sequence:subno), data=df_learning, family = binomial)
p2<-glmer(on_target_pre_total ~ poly(occurrence,3) + positions  + (1|subno)
+ (1|sequence:subno), data=df_learning, family = binomial)
p3<-glmer(on_target_pre_total ~ poly(occurrence,3) * positions  + (1|subno)
+ (1|sequence:subno), data=df_learning, family = binomial)
p4<-glmer(on_target_pre_total ~ poly(occurrence,3) * positions + condition
+ (1|subno) + (1|sequence:subno), data=df_learning, family = binomial)
p5<-glmer(on_target_pre_total ~ poly(occurrence,3) * positions * condition
+ (1|subno) + (1|sequence:subno), data=df_learning, family = binomial)

anova(p1,p2)
anova(p1,p2)
anova(p1,p2)
anova(p1,p2)

df_learning$fitted_model_trans<-fitted(m63)

df_learning %>%
  ggplot(aes(x=occurrence, y=
fitted_model_trans,colour=positions),group=subno) +
  geom_smooth( method = "lm", formula = y ~ poly(x,3))+
  theme_minimal(base_size = 10)+
  facet_wrap(~condition)+
  labs(title ="Fit of Model for learning rate across mixed length tasks")+
  xlab("Number of repetition of item - exposure")+
  ylab("Learning Rate (0-1)")+
  ylim(0,1)+
  ggsave('./plots/ transitions 1.png', height = 16.6,width = 16.6, units =
'cm')
```

M.3 R Script for binomial modelling of sequence length effects in learning.

```

#run the getting threshold.R script first
options(scipen=999)

# sequence length on same length tasks
df_nonmixed <- subset(df2,type == 'non_mixed')
# sequence length on mixed length tasks
df_mixed <-subset(df2,type == 'mixed')

#select meaningful transitions
sl_data<-df_nonmixed %>% filter((transitions %in% 1:1 & condition == "2
dots") |
                                  (transitions %in% 1:2 & condition ==
"3 dots")|
                                (transitions %in% 1:3 & condition == "4
dots"))

```

```
ml_data <- df_mixed %>% filter((sequence_length == "3 dots"  & condition ==
"2&3 dots" & transitions %in% 1:2 ) |
                                 (sequence_length == "2 dots"  & condition
== "2&3 dots" & transitions %in% 1:1 ) |
                                 (sequence_length == "4 dots"  & condition
== "2&4 dots" & transitions %in% 1:3 ) |
                                 (sequence_length == "2 dots"  & condition
== "2&4 dots" & transitions %in% 1:1 ) |
                                 (sequence_length == "4 dots"  & condition
== "3&4 dots" & transitions %in% 1:3 ) |
                                 (sequence_length == "3 dots"  & condition
== "3&4 dots" & transitions %in% 1:2 ))

#sequence length and type of task effects
df_learning<-rbind(sl_data,ml_data)
##########################################################################
#######
###SEQUENCE LENGTH EFFECTS ON SAME LENGTH TASKS
#check the polynomial on occurrence that is still valid
sl_data$learning<-as.numeric(sl_data$learning)
m01<-glmer(learning ~ poly(occurrence,1) + (1|subno) + (1|sequence:subno),
sl_data, family = binomial)
m02<-glmer(learning ~ poly(occurrence,2) + (1|subno) + (1|sequence:subno),
sl_data, family = binomial)
m03<-glmer(learning ~ poly(occurrence,3) + (1|subno) + (1|sequence:subno),
sl_data, family = binomial)

anova(m01,m02)
anova(m03,m02)
sl_data$learning<-as.numeric(sl_data$learning)
m0<-glmer(learning ~ poly(occurrence,3) + (1|subno) + (1|sequence:subno),
sl_data, family = binomial)
m1<-glmer(learning ~ poly(occurrence,3) + sequence_length  + (1|subno) +
(1|sequence:subno), sl_data, family =binomial)
m21<-glmer(learning ~ poly(occurrence,3) * sequence_length  + (1|subno) +
(1|sequence:subno), sl_data, family =binomial)

anova(m0,m1)
anova(m1,m21)
sl_data$fitted_model_sl<-fitted(m21)

slsum<- summary(m21)
slsum
sl_data$learning<-as.numeric(sl_data$learning)
#plot
sl_data%>%
  ggplot(aes(x=occurrence, y= fitted_model_sl),group=subno) +
  geom_smooth( method = "lm", formula = y ~ poly(x,3))+
  theme_minimal(base_size = 10)+
  facet_wrap(~condition)+
  labs(title ="Fit of Model for learning rate across same length tasks")+
  xlab("Number of repetition of item - exposure")+
  ylab("Learning Rate (0-1)")+
  ylim(0,1)+
  ggsave('./plots/ sl-1.png', height = 16.6,width = 16.6, units = 'cm')
##########################################################################

#####  SEQUENCE LENGTH EFFECTS IN MIXED LENGTH TASKS

ml_data$learning<-as.numeric(ml_data$learning)
```

```
m01<-glmer(learning ~ poly(occurrence,1) + (1|subno) + (1|sequence:subno),
ml_data, family = binomial)
m02<-glmer(learning ~ poly(occurrence,2) + (1|subno) + (1|sequence:subno),
ml_data, family = binomial)
m03<-glmer(learning ~ poly(occurrence,3) + (1|subno) + (1|sequence:subno),
ml_data, family = binomial)

anova(m01,m02)
anova(m03,m02)

m<-glmer(learning ~ poly(occurrence,3) + (1|subno) + (1|sequence:subno),
ml_data, family = binomial)
m12<-glmer(learning ~ poly(occurrence,3) + sequence_length  + (1|subno) +
(1|sequence:subno), ml_data, family =binomial)
m2<-glmer(learning ~ poly(occurrence,3) * sequence_length  + (1|subno) +
(1|sequence:subno), ml_data, family =binomial)

anova(m,m12)
anova(m12,m2)
ml_sum<-summary(m12)
ml_sum
# m12 best fit    ###################################
ml_data$fitted_model_ml<-fitted(m12)
ml_data$learning<-as.numeric(ml_data$learning)
ml_data%>%
  ggplot(aes(x=occurrence, y=
fitted_model_ml,colour=sequence_length),group=subno) +
  geom_smooth( method = "lm", formula = y ~ poly(x,3))+
  theme_minimal(base_size = 10)+
  facet_wrap(~condition)+
  labs(title ="Fit of Model for learning rate across mixed length tasks")+
  xlab("Number of repetition of item - exposure")+
  ylab("Learning Rate (0-1)")+
  ylim(0,1)+
  ggsave('./plots/ sl-2.png', height = 16.6,width = 16.6, units = 'cm')

##################################

##################type of task & sequence length
  df_learning$learning<-as.numeric(df_learning$learning)
  m0001<-glmer(learning ~ poly(occurrence,1)  + (1|subno) +
(1|sequence:subno), data=df_learning, family = binomial)
  m0002<-glmer(learning ~ poly(occurrence,2)  + (1|subno) +
(1|sequence:subno), data=df_learning, family = binomial)
  m0003<-glmer(learning ~ poly(occurrence,3)  + (1|subno) +
(1|sequence:subno), data=df_learning, family = binomial)
  anova(m0001,m0002)
  anova(m0002,m0003)

z1<-glmer(learning ~ poly(occurrence,3) + type + (1|subno) +
(1|sequence:subno), data=df_learning, family = binomial)
z2<-glmer(learning ~ poly(occurrence,3) * type + (1|subno) +
(1|sequence:subno), data=df_learning, family = binomial)
z3<-glmer(learning ~ poly(occurrence,3) *type + sequence_length + (1|subno)
+ (1|sequence:subno), data=df_learning, family = binomial)
z4<-glmer(learning ~ poly(occurrence,3) * type * sequence_length +
(1|subno) + (1|sequence:subno), data=df_learning, family = binomial)

anova(z1,z2)
anova(z3,z2)
anova(z3,z4)
```

```
anova(z4,z2)
df_learning$fitted_model_type<-fitted(z4)

df_learning %>%
  ggplot(aes(x=occurrence, y=
fitted_model_type,colour=sequence_length),group=subno) +
  geom_smooth( method = "lm", formula = y ~ poly(x,3))+
  theme_minimal(base_size = 10)+
  facet_wrap(~type*condition)+
  labs(title ="Fit of Model for learning rate across mixed length tasks")+
  xlab("Number of repetition of item - exposure")+
  ylab("Learning Rate (0-1)")+
  ylim(0,1)+
  ggsave('./plots/ sl-3.png', height = 16.6,width = 16.6, units = 'cm')

```

M.4 Model fits of sequence length effects in learning.

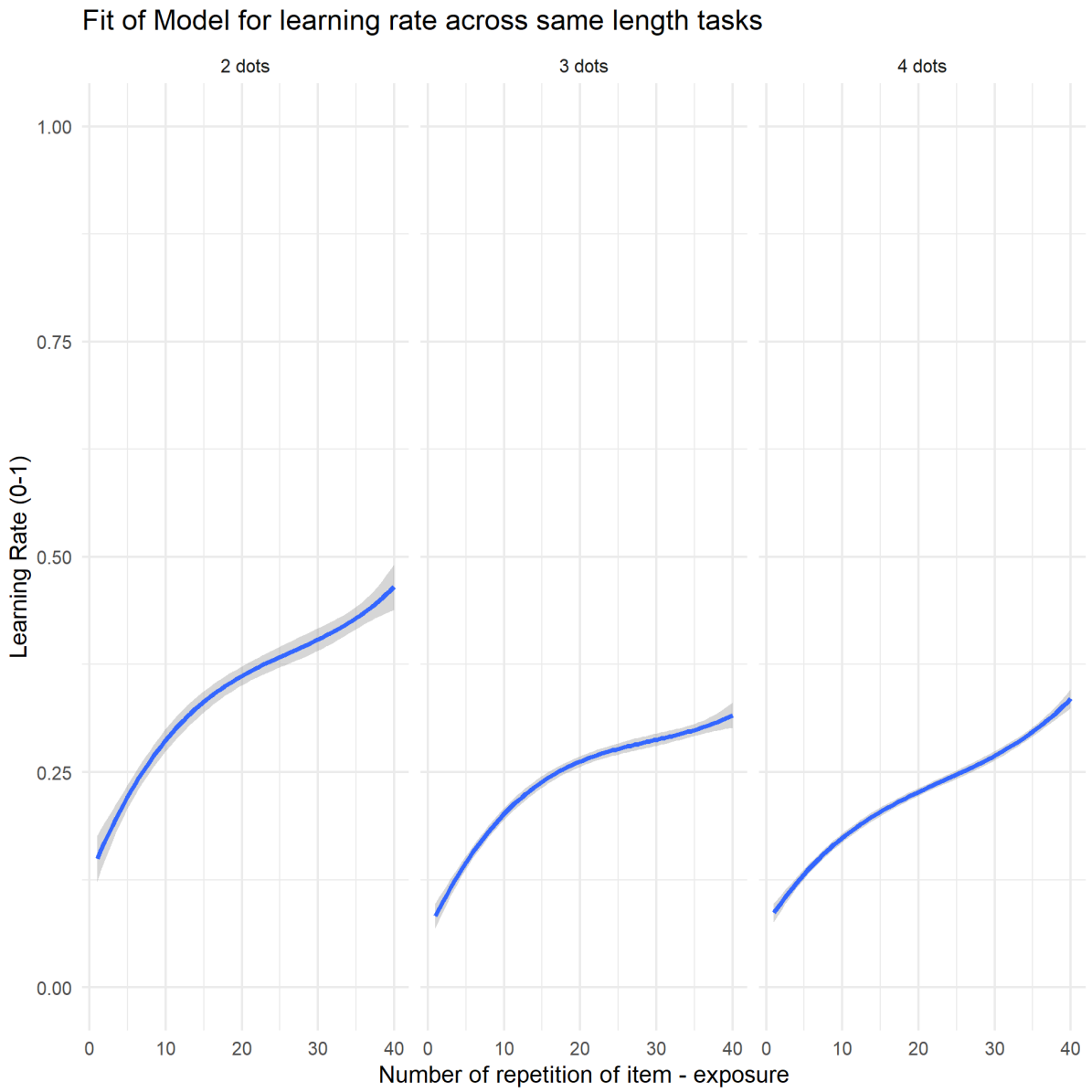


Figure M.1 demonstrates the fits of the binomial transformation of hypothesis model 2 (Chapter 6- Section C.3) for the same length tasks (2 dots, 3 dots, 4 dots) On the x-axis is the number of repetitions of item or in other words exposure to that specific sequence (how many times that stimulus has been shown) and on the y- axis is the mean rate of from (0-1) for the target –location area during the blank period of (750ms). Any target with greater or equal 25 eye-samples on it is considered as learned (1), while any target with less than 25 eye-samples on it is considered not learned (0).

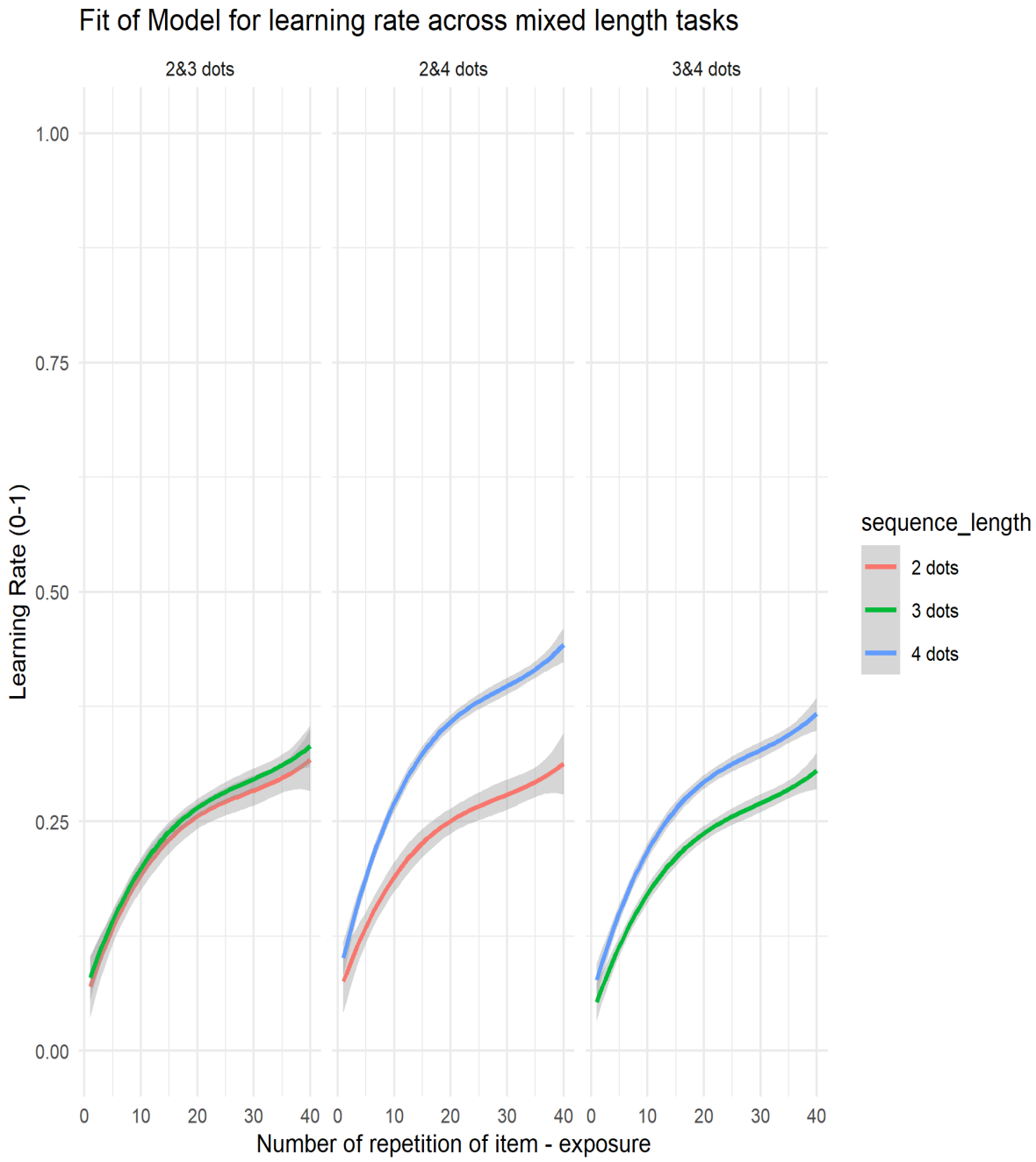


Figure M.2 demonstrates the fits of the binomial transformation of hypothesis model 2 (Chapter 6- Section C.4) for the mixed length tasks (2&3 dots, 2&4 dots, 3&4 dots). On the x-axis is the number of repetitions of item or in other words exposure to that specific sequence (how many times that stimulus has been shown) and on the y-axis is the mean rate of from (0-1) for the target –location area during the blank period of (750ms). Any target with greater or equal 25 eye-samples on it is considered as learned (1), while any target with less than 25 eye-samples on it is considered not learned (0).

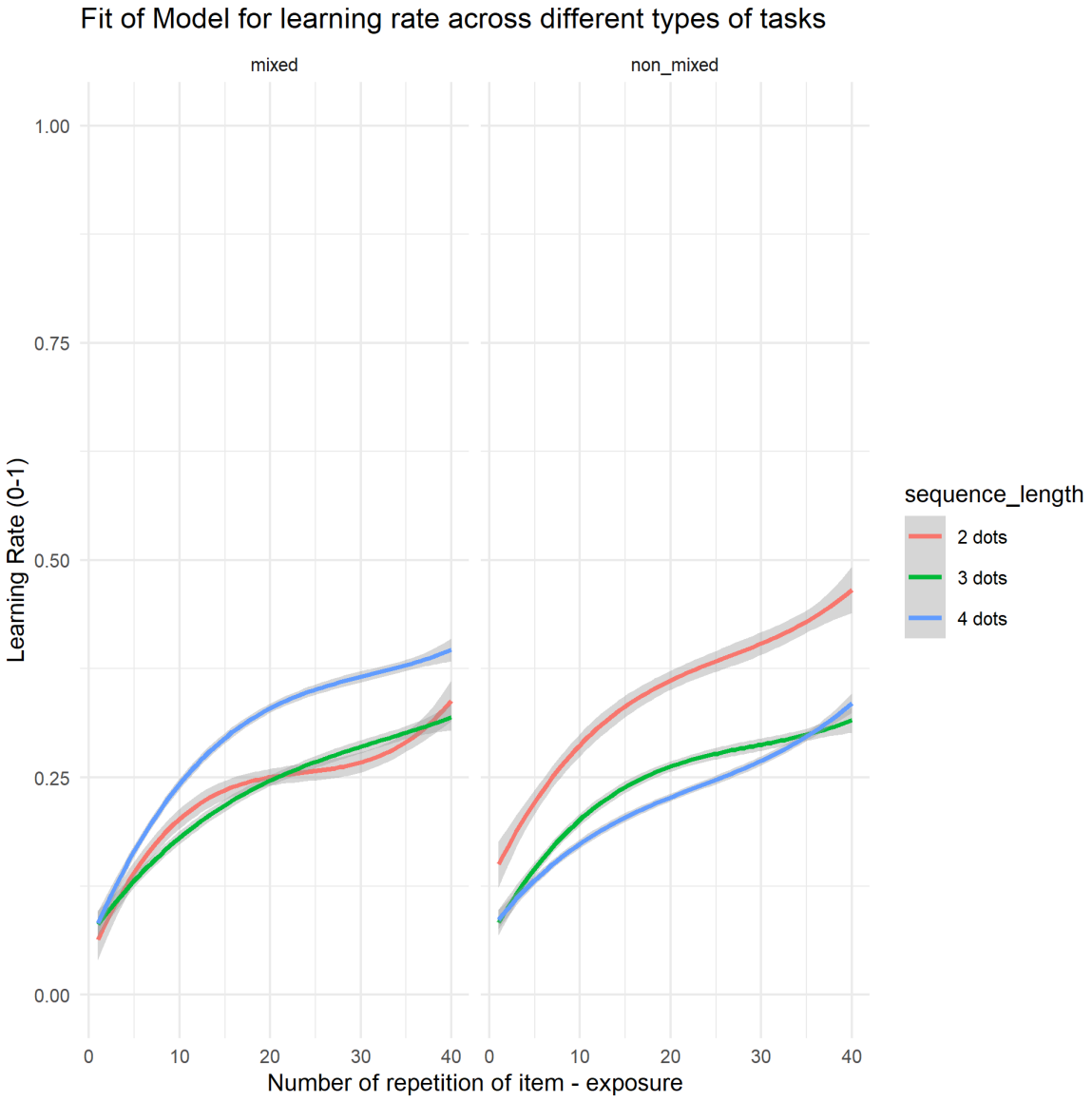


Figure M.3 demonstrates the fits of the binomial transformation of hypothesis model 4 (Chapter 6- Section C.5) for sequence length effects and sequence mixture effects across all tasks (2 dots, 3 dots, 4 dots, 2&3 dots, 2&4 dots, 3&4 dots). On the x-axis is the number of repetitions of item or in other words exposure to that specific sequence (how many times that stimulus has been shown) and on the y- axis is the mean rate of from (0-1) for the target –location area during the blank period of (750ms). Any target with greater or equal 25 eye-samples on it is considered as learned (1), while any target with less than 25 eye-samples on it is considered not learned (0).

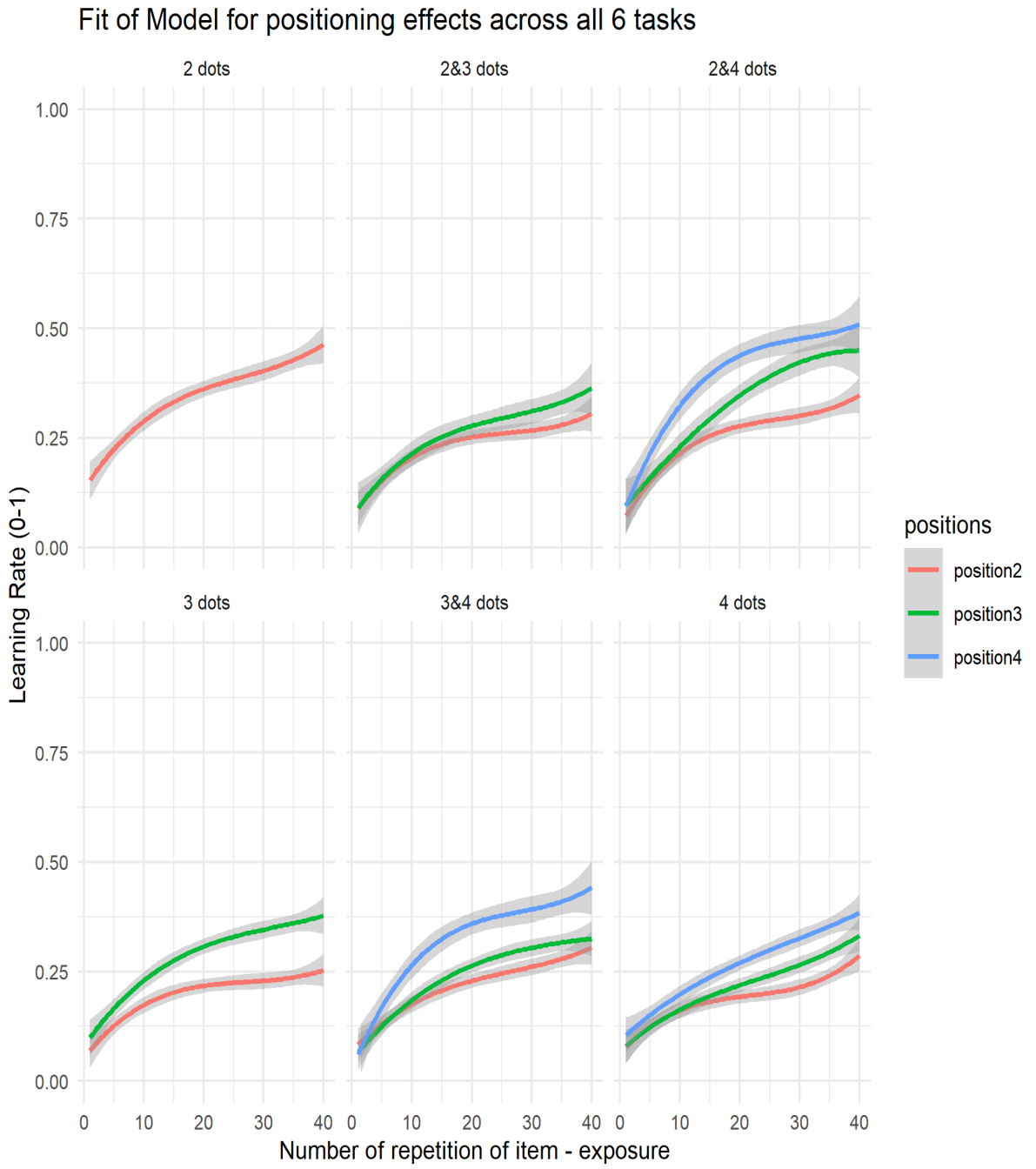


Figure M.4 demonstrates the fits of the binomial transformation of hypothesis model 4 (Chapter 5 - Section C.2) for position effects across all tasks (2 dots, 3 dots, 4 dots, 2&3 dots, 2&4 dots, 3&4 dots). On the x-axis is the number of repetitions of item or in other words exposure to that specific sequence (how many times that stimulus has been shown) and on the y- axis is the mean rate of from (0-1) for the target –location area during the blank period of (750ms). Any target with greater or equal 25 eye-samples on it is considered as learned (1), while any target with less than 25 eye-samples on it is considered not learned (0).

Appendix N – R scripts Analysis for Chapter 5 section B

N.1 R scripts for data visualisation and Shape of curve modelling.

```
##### 1st transition vs insequence
#create variable for type of transition

df2<-df2 %>%
  mutate(transition_type = ifelse(target_position %in% c("position1"),
"transition_between_sequence",
                          ifelse(target_position %in%
c("position2","position3","position4"), "transition_within_sequence", NA)))

#visualise
df2%>%
  ggplot(aes(x=occurrence, y=on_target_pre_total,
colour=transition_type),group=subno) +
  geom_smooth(method = "lm", formula = y ~ poly(x,3))+
  theme_classic(base_size = 8)+
  coord_cartesian(ylim = c(0, 250))+
  labs(title ="Learning rate for between-sequence vs. within-sequence
transitions \n during the 750ms blank period")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank period
of 750ms")
  ggsave('./new_plots/5Bvisual.png', height = 8,width = 13, units = 'cm')

df2<-df2 %>%
  mutate(transition = ifelse(target_position %in% c("position_1"), "1st",
                          ifelse(target_position %in% c("position2"), "2nd",
                           ifelse(target_position %in% c("position3"), "3rd",
                                  ifelse(target_position %in%
c("position4"),"4th",NA)))))



df2%>%
  ggplot(aes(x=occurrence, y=on_target_pre_total, colour=
target_position),group=subno) +
  geom_smooth(method = "lm", formula = y ~ poly(x,3))+
  theme_classic(base_size = 8)+
  facet_grid(~condition)+
  coord_cartesian(ylim = c(0, 250))+
  labs(title ="Learning rate across tasks for each target position during
the 750ms blank period")+
  xlab("Number of repetition of item - exposure")+
  ylab("Number of eye-samples on  target position \n during  blank period
of 750ms")+
  ggsave('./new_plots/5Bvisual3.png', height = 8,width = 17, units = 'cm')



###########
####### shape of curve fitting 1st transition in####
m0<-glmer(on_target_pre_total ~ 1 + (1|subno) + (1|seq:subno), data=df2,
family = poisson())
m1<-glmer(on_target_pre_total ~ poly(occurrence,1) + (1|subno) +
(1|seq:subno), data=df2, family = poisson())
```

```
m2<-glmer(on_target_pre_total ~ poly(occurrence,2) + (1|subno) +
(1|seq:subno), data=df2 , family = poisson())
m3<-glmer(on_target_pre_total ~ poly(occurrence,3) + (1|subno) +
(1|seq:subno), data=df2 , family = poisson())


anova(m0, m1)
anova(m1, m2)
anova(m2, m3)
```

N.2 R script for models: 1st transitions vs. in-sequence transitions.

```
m0<-glmer(on_target_pre_total ~ poly(occurrence,3) + (1|subno) +
(1|sequence:subno), data=df2, family = poisson())
m1<-glmer(on_target_pre_total ~ poly(occurrence,3)+transition_type +
(1|subno) + (1|sequence:subno), data=df2, family = poisson())
m2<-glmer(on_target_pre_total ~ poly(occurrence,3)*transition_type +
(1|subno) + (1|sequence:subno), data=df2, family = poisson())
anova(m0,m1)
anova(m1,m2)
tab_model(m2,p.style = c( "both"))


```

N.3 Residual for models: 1st transitions vs. in-sequence transitions.

	on_target_pre_total		
Predictors	*Incidence Rate Ratios*	*CI*	*p*
(Intercept)	23.96 ***	19.48 – 29.47	<0.001
occurrence [1st degree]	7833543345747779937848284006404000680.00 ***	617777493266673688044022844882888866.00 – 9933091123994765633882402048082440 8.00	<0.001
occurrence [2nd degree]	0.00 ***	0.00 – 0.00	<0.001
occurrence [3rd degree]	98394082351.38 ***	76829290923.53 – 126011776568.47	<0.001
transition_type [transition_within_sequence]	1.77 ***	1.76 – 1.77	<0.001
occurrence [1st degree] * transition_type [transition_within_sequence]	33392784895009 51.00 ***	259738966986773 6.50 – 429307198677338 6.50	<0.001
occurrence [2nd degree] * transition_type [transition_within_sequence]	0.00 ***	0.00 – 0.00	<0.001
occurrence [3rd degree] * transition_type [transition_within_sequence]	0.03 ***	0.02 – 0.03	<0.001
Random Effects			
σ^2	0.03		
$\tau_{00\ sequence:subno}$	1.73		
$\tau_{00\ subno}$	0.54		
ICC	0.99		
N_{subno}	36		
$N_{sequence}$	24		
Observations	103464		
Marginal R^2 / Conditional R^2	0.081 / 0.990		

*p<0.05 **p<0.01 ***p<0.001

Table N.1 This table shows the residuals of the model of hypothesis model 3, in Chapter 5, Section B.3 describing transition type effects in learning across all tasks (fixed effects).