DYNAMIC CONSTRAINTS IN STATISTICAL LEARNING

Joshua R. de Leeuw

Submitted to the faculty of the University Graduate School

in partial fulfillment of the requirements

for the degree

Doctor of Philosophy

in the Cognitive Science Program and

Department of Psychological and Brain Sciences,

Indiana University

August, 2016

Accepted by the Graduate Faculty, Indiana University, in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Doctoral Committee

_____

Robert L. Goldstone, Ph.D.

_____

Peter M. Todd, Ph.D.

_____

John K. Kruschke, Ph.D.

_____

Chen Yu, Ph.D.

July 18th 2016

Acknowledgements

This dissertation would not exist without the many forms of support that I have been fortunate enough to receive. I am deeply appreciative of the intellectual guidance and mentorship provided by Rob Goldstone and Peter Todd. John Kruschke and Chen Yu provided valuable insights and critiques that not only improved the work that existed but pushed me in new directions that I would not have considered otherwise. Members of the Percepts and Concepts Lab, in particular Paulo Carvalho, Ryan Best, and Tyler Marghetis shaped this work through many conversations. I would also like to thank Cheryl Hodaba and Lydia de Leeuw for assistance with data coding. Financial support for this research came from the NSF Graduate Research Fellowship Program and the NSF IGERT Program. This research was also supported in part by Lilly Endowment, Inc., through its support for the Indiana University Pervasive Technology Institute, and in part by the Indiana METACyt Initiative.

Finally, Tina de Leeuw has supported me in more ways than I can possibly describe here. In addition to the countless forms of support she provided as a partner, she also contributed substantially to the editing of this dissertation. I hope that her red pen has brought as much clarity to this work as she has brought to my life.

Joshua R. de Leeuw

DYNAMIC CONSTRAINTS IN STATISTICAL LEARNING

Statistical learning is a ubiquitous cognitive phenomenon in which learners extract the probabilistic regularities that generate the sensory environment. Characterizing the mechanisms that enable this kind of learning requires describing the constraints that shape learning. In this dissertation, I describe how some constraints on statistical learning may change on very short timescales as the learner acquires new information. In a series of experiments, I show that learning part of the probabilistic structure of a sequential pattern substantially improves learning of the other (statistically independent) parts of the probabilistic structure. This suggests that the learning process alters its own constraints. Furthermore, I demonstrate that the learning curves for individual items show rapid step-like changes as learners discover the statistical structure of the sequence. Taken together, these results suggest limits on the kinds of computational mechanisms that can explain statistical learning. A successful computational account needs to capture both the dependence between learning different parts of the probabilistic structure and the sudden, rather than gradual, changes in behavioral evidence of learning. I propose a general accumulator model to handle cases where people are learning multiple items simultaneously with or without mutual dependence between the learning rates. The model predicts when people will show the step-like change in behavior as they learn a sequential structure. The model is able to capture the results of the experiments only with high dependence between the learning rates of different items.

_____

Robert L. Goldstone, Ph.D.

_____

Peter M. Todd, Ph.D.

_____

John K. Kruschke, Ph.D.

_____

Chen Yu, Ph.D.

Table of Contents

List of Appendices

List of Figures

1 Introduction

Theories about learning are necessarily theories about constraints. Learning algorithms that are efficient for some problems are provably inefficient for others (Wolpert & Macready, 1997; Wolpert, 1996); any model of learning will be constrained in the kinds of problems that it can solve effectively. Cognitive science has long recognized the crucial role of constraints on learning, though there has been considerable debate about how constraints come to be and what the relevant constraints are (e.g., Elman, 1993; Elman et al., 1996; Gold, 1967; Goldstone & Landy, 2010; Keil, 1990; Newport, 1990; Shepard, 1984; Spelke & Kinzler, 2007).

In this dissertation, I consider constraints on statistical learning, a phenomenon of central importance to a diverse set of abilities (for recent reviews, see Aslin & Newport, 2012; Frost, Armstrong, Siegelman, & Christiansen, 2015; Schapiro & Turk-Browne, 2015; Turk-Browne, 2012). Statistical learning is the ability to discover regularities in sensory input. The distinction between a regularity and spurious co-occurrence is only possible to draw by integrating information across multiple instances. The necessity of learning across time makes statistical learning a particularly interesting domain for investigating constraints on learning because constraints, in theory, could change during learning.

1.1 How Should We Conceptualize Statistical Learning?

A canonical example of statistical learning is segmenting speech into words. Word boundaries are often ambiguous in unfamiliar languages, even with prosodic cues – just recall the last time you heard a rapid conversation in an unfamiliar foreign language. But with repeated exposure and the ability to make inferences based on the statistical

relationships between syllables, words can be extracted from speech even in the complete absence of any non-statistical cues like pauses or prosody (Saffran, Aslin, & Newport, 1996). For example, a single encounter with the spoken syllables `pa/di/ku/te/ba/mu` is insufficient to discover which, if any, of the syllables combine to form words. However, if `ku/te` is a frequent occurrence across many examples but `te/ba` and `di/ku` are not, then `ku/te` is likely a word.

While the above example is clearly an *instance* of statistical learning, there are different perspectives on what statistical learning actually *is*. What is the right level of description to understand statistical learning? One possibility is that statistical learning is a domain-general mechanism (Perruchet & Pacton, 2006). This suggests that a model of this singular mechanism is the best explanation of statistical learning. A domain-general account has a difficult time dealing with modality-specific effects in statistical learning (e.g., Conway & Christiansen, 2005). Yet the generality of statistical learning – the fact that it shows up in a variety of situations and domains – makes a domain-general account theoretically appealing. One possible resolution is to conceptualize statistical learning as a set of domain-general learning mechanisms that operate on domain-specific input (Frost et al., 2015). In this view, statistical learning is best understood as the process(es) these mechanisms share in common. Even broader perspectives have treated statistical learning as a "family of processes" (Bays, Turk-Browne, & Seitz, 2016) that includes basic cognitive processes, like perception, memory, and attention. In this view, the interaction of different process generates the outcome of statistical learning. "Statistical learning" would then describe a kind of representational and behavioral change, but not necessarily the mechanisms that produce them.

The evidence for these different perspectives is mixed. Occasionally statistical learning can transfer from one task domain to another (Altmann, Dienes, & Goode, 1995; Turk-Browne & Scholl, 2009), though other studies find that learning is modality-specific or even stimuli-specific (Christiansen & Conway, 2006; Conway & Christiansen, 2005). Neuroimaging evidence implicates a common set of brain regions – the medial temporal lobe and hippocampus – in a variety of statistical learning tasks, yet other brain regions are involved in only some kinds of tasks or in some modalities (Schapiro & Turk-Browne, 2015). Siegelman and Frost (2015) found no evidence for a correlation between statistical learning in different modalities within the same individuals. Direct evidence for the multiple mechanisms account comes from Bays et al. (2016), who found that different behavioral measures of statistical learning within an individual for the same item can produce opposite results. They measured statistical learning for individual items and found that items that show learning in search tasks are the same items that fail to show learning in detection tasks. One account of these results is there are multiple competing processes that contribute to statistical learning. The substantial variety of mechanisms used by models of statistical learning – associations, hypothesis testing, learning from prediction error, chunking, and so on – serves as a proof-of-concept that multiple kinds of learning processes can support statistical learning.

1.2 How Should We Conceptualize Constraints on Statistical Learning?

If statistical learning emerges from the interaction of basic cognitive processes like perception, memory, and attention, then these processes should shape and constrain learning (Endress, Nespor, & Mehler, 2009). Perceptual cues like similarity, continuity, and gestalt groupings can influence what visual structures are learned (Baker, Olson, &

Behrmann, 2004; Creel, Newport, & Aslin, 2004; Fiser, Scholl, & Aslin, 2007). Selective attention modulates learning – structures in sequences that are observed but not attended to are not learned – even though the learning process itself can occur without conscious awareness or effort (Toro, Sinnett, & Soto-Faraco, 2005; Turk-Browne, Jungé, & Scholl, 2005). Memory constraints have been implicated in performance on statistical learning tasks in several ways. Reducing memory demands by presenting information simultaneously, as opposed to sequentially, improves learning of a variety of different kinds of statistical structure (Frank & Gibson, 2011). Incorporating memory constraints into models of statistical learning can improve their fit to human performance (Frank, Goldwater, Griffiths, & Tenenbaum, 2010). Finally, working memory capacity is positively correlated with the ability to learn implicit statistical structure, suggesting that individual variation in statistical learning can be predicted, in part, by differences in participants' abilities to remember what they have seen (Karpicke & Pisoni, 2004).

Learning is also constrained by biases that are not directly attributable to a generic cognitive process. These biases are often described as priors on the kinds of statistical structures that learners use to make sense of the data. A well-known case in early word learning is mutual exclusivity (Merriman & Bowman, 1989). Adult learners exhibit a related parsimony bias. They prefer segmentations that result in simple and consistent structures, such as words that have the same length (Frank, Tily, Arnon, & Goldwater, 2010) or the minimal generative vocabulary necessary for reproducing a scene (Orbán, Fiser, Aslin, & Lengyel, 2008). Biases also help learners deal with the possibility of statistical structures changing over time. When confronted with a statistical structure that covertly changes after some initial exposure, learners stick with their initial interpretation

of the structure unless there is a cue that the structure may have changed (Gebhart, Aslin, & Newport, 2009; R. Q. Yu & Zhao, 2015), suggesting a strong – yet flexible – prior on stationary statistical structure (Aslin, 2014). These priors might be acquired through discovery of the kinds of structures that can be used to parsimoniously interpret the data (Tenenbaum, Kemp, Griffiths, & Goodman, 2011). For example, when exposed to a novel language, infants show a bias towards segmenting a stream of syllables into words that are the same length as previously heard words, but this bias can be adjusted to different lengths depending on exposure (Lew-Williams & Saffran, 2012).

Constraints like those described above are often framed as constraints *on* the learning process (e.g., Emberson, Liu, & Zevin, 2013; Frank & Gibson, 2011; Saffran, 2002; Turk-Browne et al., 2005). Yet, statistical learning affects the basic cognitive processes that constrain it (Turk-Browne, 2012). Attention is captured by statistical structure that is neither too predictable nor too random (Kidd, Piantadosi, & Aslin, 2012; Zhao, Al-Aidroos, & Turk-Browne, 2013), and can affect what features of a scene are encoded (Umemoto, Scolari, Vogel, & Awh, 2010; Zhao, Ngo, McKendrick, & Turk-Browne, 2011). Working memory capacity for a set of items improves as the statistical regularities of the set are learned (Brady, Konkle, & Alvarez, 2009). Perceptual sensitivity to the presence of an object is improved when the object is predictable due to learned regularities (Barakat, Seitz, & Shams, 2013; Bays et al., 2016).

If statistical learning results from the interaction of cognitive processes and learning can shape these processes then constraints caused by these processes are best understood as emergent – depending crucially on the experience, knowledge, and goals of the learner. This is not to argue that constraints will necessarily be fleeting, but rather that

constraints exist on different timescales depending on the processes that create them. Some constraints are stable over a lifetime because they are, for example, caused by stable properties of sensory systems (Emberson et al., 2013). Others change slowly via mechanisms like perceptual learning (Gerganov, Grinberg, Quinn, & Goldstone, 2007), and some might change within a few exposures based on the discovery of new structural forms or encoding of new relationships.

1.3 Overview

In this dissertation, I focus on constraints that change on very short timescales as learning occurs. I tackle this topic through variations of a canonical sequential statistical learning task (Fiser & Aslin, 2002) and models of statistical learning. Throughout, I test several instantiations of two broad model classes: models that predict learning to segment one portion of the sequence will affect learning of the other portions and models that predict independence of learning between different items. I show that only models that predict dependence between items can account for the results of several experiments.

In Chapter 2, I investigate the relationship between memory constraints and statistical learning. In two versions of a sequential segmentation task, I find that learning of a novel sequential pattern is more likely when the surrounding sequence contains patterns that the subject can either learn (Experiment 2.1) or already knows (Experiment 2.2). I test several process models of statistical learning on these experiments, and show that the pattern of results is only accounted for by models in which the learning process and memory constraints interact. In these models, learning creates compressed chunks, which require fewer memory resources to encode (Gobet et al., 2001b). This improved efficiency boosts learning by improving memory for items that have yet to be chunked.

This demonstration shows a specific example of how learning can alter the processes that constrain it on very short timescales.

Chapter 3 builds on these results by measuring learning as it happens in a variant of the serial response time task (Nissen & Bullemer, 1987). I develop a hierarchical Bayesian model to characterize the shape of the learning curve for individual items. This analysis strategy avoids well-documented but often overlooked problems of averaging learning curves (Estes, 1956). Obtaining item-level curves for individual participants enables the contrast of cognitive models that predict steady improvement and those that predict sudden and rapid improvement. I also investigate the relationship between learning and explicit knowledge (e.g., Mathews et al., 1989), showing that the two are highly correlated and that measuring learning at the item level for individual participants is more powerful than group-level analyses to assess the relationship between the two.

Chapter 4 introduces an accumulator model of learning multiple items simultaneously. This model generalizes the findings from Chapter 2 by demonstrating that a generic dependency mechanism between items is sufficient to account for the results of all of the experiments presented in Chapters 2 and 3. Memory constraints might be one way to produce dependency, but other processes like attention to sequential dependencies (Zhao et al., 2011) or learned expectancies about word length (Lew-Williams & Saffran, 2012) can also explain the results. By parameterizing the model in a way that maps onto key features of learning – the rate, stability, and the dependencies between items – general conclusions can be drawn about what learning looks like in a single trial. These insights can be used to refine models that implement specific kinds of learning mechanisms.

All of the raw data, analysis scripts, models, and model output presented in the dissertation are available in an Open Science Framework repository at https://osf.io/t5ahe/.

2 Memory Constraints and Statistical Learning

One way in which memory and learning interact is through the formation of chunks. Chunks are a mechanism for compressing correlated inputs, in a manner that reduces the computational resources required to store/represent/encode the raw bits of information (for a review, see Gobet et al., 2001a). The concept of chunking has been used to explain a wide range of psychological phenomena, including the advantages that expert chess players have in remembering the positions of chess pieces on a board (Chase & Simon, 1973; Gobet & Simon, 1998), differences in the speed of retrieving successive letters of the alphabet (Klahr, Chase, & Lovelace, 1983), the ability to remember more words when the words are part of familiar phrases (Simon, 1974), and many other feats of memory in which there is extensive experience within the domain (see Ericsson & Kintsch, 1995 and many references within). Working memory capacity is frequently conceptualized in terms of a number of chunks (Cowan, 2010; Miller, 1956). In working memory tasks, a set of items is more likely to be remembered if the set can be compressed though chunking mechanisms (Brady et al., 2009; Mathy & Feldman, 2012). Models using chunking mechanisms have successfully accounted for many of these phenomena (Gobet, Lane, & Lloyd-Kelly, 2015)

A canonical sequential statistical learning task is to present the participant with a sequence of tokens at a steady rate. The tokens are typically syllables for auditory tasks (Aslin, Saffran, & Newport, 1998; Frank, Goldwater, et al., 2010; Frank, Tenenbaum, & Gibson, 2013; Saffran et al., 1996) or abstract shapes for visual tasks (Fiser & Aslin, 2001, 2005; Turk-Browne et al., 2005; Zhao et al., 2013). The tokens are grouped together such that the transitional probability of within-group transitions is very high

compared to the between-group transitions. For example, in an experiment with twelve tokens, A–L, and four equal-sized groups of three, the tokens A, B, and C would form one group. After every A in the sequence a B would occur, and then a C. This kind of statistical structure affords compression because the sequence contains redundant information, and it would be reasonable to expect that participants in the task would learn an ABC chunk.

Several models of statistical learning processes use the formation of new representational units to explain performance on statistical learning tasks (French, Addyman, & Mareschal, 2011; Orbán et al., 2008; Perruchet & Vinter, 1998; Robinet, Lemaire, & Gordon, 2011). While the implementation varies from model to model, the general computational principle is that the learning process creates new units that compress the information stream by grouping highly correlated inputs together. For example, the MDLChunker model learns a lexicon of items that reduce the overall description length of the sequence (Robinet et al., 2011).

Given that a consequence of acquiring chunks is an increase in working memory capacity for individual tokens, and memory capacity acts as a constraint on statistical learning, a probable hypothesis is that there is a cyclical interaction between learning and memory capacity in statistical learning tasks. If this hypothesis is correct, then as people learn the structure of a sequence, they should be able to remember more of the sequence by chunking the primitives into larger units. This should make it easier to learn additional aspects of the structure. Because learning is accelerated, new chunks are quickly learned, and the cycle repeats.

In this chapter, I present two experiments investigating this possible interaction between memory mechanisms and statistical learning processes. In the first experiment, participants viewed a sequence of abstract shapes and then attempted to recognize a target set of three shapes that always appeared in the same order. In one condition, the rest of the sequence was compressible, providing an opportunity to learn chunks. In the other condition, the rest of the sequence was unpredictable. If chunking begets learning begets chunking, participants should be more likely to recognize the correct order of the target shapes when the rest of the sequence is also compressible. This is what I found. In the second experiment, I tested whether the differences in statistical structure alone can explain this effect, or if a learner's internal representations (chunks) are crucial by using two sequences with identical statistical structure but different familiarity to the learner.

I tested a set of models on these experiments, all of which use a chunking-based strategy to learn the input. However, the models varied on whether learning new chunks affected memory capacity. Some of the models predict that learning chunks should accelerate learning other chunks via reduced memory demands, while others do not. The models that contain this interaction were able to match the qualitative patterns of results observed in the experiment, while the models that lacked this interaction showed no difference in performance between the two different kinds of sequence. This suggests that memory mechanisms and learning processes interact during statistical learning and constrain the learning.

2.1 Experiment 2.1

2.1.1 Method

2.1.1.1 Participants

40 separate HIT assignments were posted on Amazon Mechanical Turk. 41 people participated in the study, due to one person completing the experiment without submitting a HIT.

2.1.1.2 Materials

Each participant viewed a unique randomly generated sequence of shapes. There were twelve distinct shapes, modeled after the shapes used by Fiser and Aslin, (2002). Each shape appeared 25 times, making the sequence 300 shapes long. Shapes were presented using the same method as in Fiser and Aslin (2002) – a vertically-oriented black bar remained in the center of the screen throughout the sequence. Shapes moved horizontally from "behind" the black bar until they were fully visible, and then moved back to the center of the screen, disappearing behind the bar. The next shape appeared on

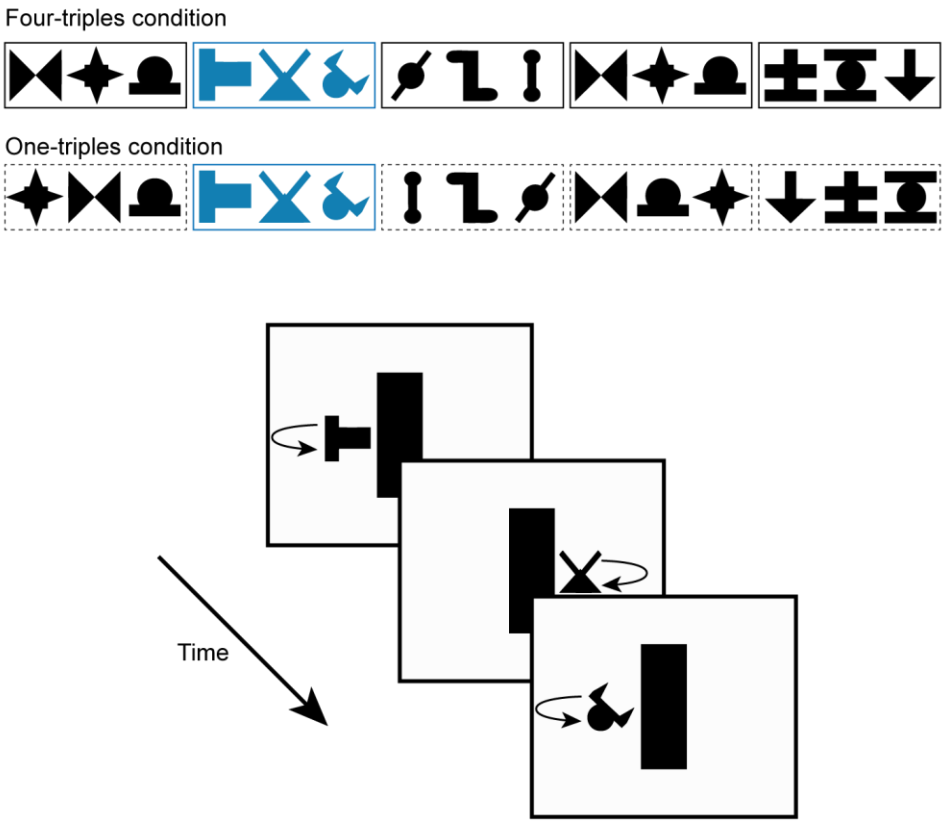

Figure 1. Experiment design for four-triples versus one-triple sequences. The upper half shows a short portion of a possible sequence from both conditions. The boxes highlight the triple-based structure. Solid boxes are drawn around the triples. Dashed boxes are drawn around the randomized sets of three. Note that in the four-triples condition, the first and fourth triples occur in the same order. In the one-triple condition, the order of the three shapes is shuffled in the first and fourth set of three. The lower portion illustrates how the sequence was shown to the participant. Shapes were presented in an animated sequence, moving horizontally back and forth across the screen. Shapes changed to the next item in the sequence when they were occluded by the black rectangle.

the other side of the black bar. The perceptual effect is that the shapes change identity while occluded by the black bar (Figure 1). Each shape was visible for a total of one second[1].

Participants were randomly assigned to either the *four-triples* or *one-triple* condition. In the four-triples condition, the twelve shapes were randomly grouped into four sets of three shapes (hereafter: a *triple*). The sequence of 300 shapes was formed by

[1]An example animation is available in the OSF repository: https://osf.io/p6x5f/.

randomly ordering 25 instances of each triple, with two constraints: (1) A triple could not occur twice in a row, and (2) a triple could not occur more than twice before every other triple occurred at least once. In the one-triple condition, the sequence was constructed in a similar fashion, except that, for three of the four triples, the presentation order of the items within the triple was randomly shuffled for each occurrence of the triple. The remaining triple maintained the same order throughout the sequence.

The experiment was built in JavaScript using the jsPsych software library (de Leeuw, 2015). Participants completed the experiment using a computer and web browser of their choice.

2.1.1.3 Procedure

Upon loading the experiment webpage, participants read a brief set of instructions. The instructions told the participants that they would be watching a five-minute long animation involving simple shapes and that they would be tested on what they saw. However, the instructions did not describe the precise nature of the test. The only viewing instruction given was to attend to the shapes. There was no other task for the participants to perform during this exposure phase.

After viewing the animation sequence, participants began the testing phase of the experiment. The test consisted of 32 two-alternative forced-choice questions. For each test item, two three-item sequences played using the same animation technique as used in the exposure phase. There was a blank screen displayed for one second between each sequence. After both sequences were finished, participants indicated whether the first or second sequence occurred more often during the exposure phase.

In the four-triples condition, each test pair contained one triple and one set of three shapes that never occurred sequentially during the sequence although each of the individual shapes did occur. There were four different triples and four different foil items. Each possible triple/foil pair was tested twice, with the triple occurring first once and the foil occurring first once, for a total of 32 trials. In the one-triple condition, the four targets included the lone triple and the three shuffled triples, each presented in the same order during every test trial. The four foil items were constructed in the same way as the four-triples condition. There were 32 trials in the one-triple condition, but only 8 of the trials contained an actual triple. The extra 24 trials were included to ensure that the frequency of triples during the testing phase was balanced, so that participants could not learn the correct triples during the testing phase. However, the data from these 24 trials were not included in the analysis.

2.1.2 Results

There were 21 participants in the four-triples condition and 20 in the one-triple condition. Participants in the four-triples condition correctly identified the triple 73.4% of the time, while participants in the one-triple condition correctly identified the triple 58.8% of the time (Figure 2); participants in the four-triples condition were 14.6 percentage points more accurate, on average.

I used a hierarchical model and Bayesian methods to estimate the group-level difference between the two conditions. Bayesian estimation techniques have several conceptual and practical advantages over frequentist hypothesis testing. In particular, interpretation of the posterior distribution of a parameter is intuitive – especially when compared with its frequentist counterpart, the confidence interval (Hoekstra, Morey,

Rouder, & Wagenmakers, 2014; Morey, Hoekstra, Rouder, Lee, & Wagenmakers, 2015) – and the distribution explicitly describes the uncertainty of the estimate (Kruschke & Liddell, 2015; Kruschke, 2011).

The analysis model for this experiment estimates the difference in the probability of a correct response in the two-alternative forced choice task. The full hierarchical model and fitting procedure is described in Appendix A. Briefly: the model treated each individual participant's response set as being generated by a binomial distribution. The probability-of-success parameter of the binomial distribution was estimated separately for each participant, with a hierarchical group-level estimate for each condition. The main parameter of interest is the difference between the group-level estimates of success. A desirable feature of this model is that it seamlessly accounts for the differences in the number of critical trials between conditions. Data from the one-triple condition is some number of correct judgments out of 8, while data from the four-triples condition is out of 32.

The Bayesian analysis estimates the relative probability of parameter values given the data and priors. The parameter estimates can be summarized with a 95% highest-density interval (HDI), which is the range of parameter values that contains 95% of the distribution of estimated values in which all the values inside the range are more probable than those outside. The main parameter of interest was the difference in the probability of success between the two conditions. The 95% HDI for this parameter was 10.7% to 43.9% with a mode of 26.6%. Thus, the model estimates that the probability of success is between 10.7 and 43.9 percentage points higher in the four-triples condition than the one-triple condition.

2.2 Modeling Experiment 2.1

The experiment found evidence that successful learning of a chunk depends on the overall structure of the sequence, not just the statistical properties of the chunk itself. In both conditions, the chunk had equivalent statistical properties (transition probabilities of 1.0 within chunk; equivalent number of presentations across the sequence; equivalent relative frequency of the target chunk and non-target-chunk items). In this section, I compare the performance of three process models of chunk learning on the task completed by the participants. While all three models describe a process for acquiring chunks, the models use very different strategies for learning. I start with a brief summary of each of the three models, and then explore the results produced by each.

2.2.1 Description of Models

2.2.1.1 PARSER (Perruchet & Vinter, 1998)

PARSER is a symbolic model that constructs an internal lexicon of potential chunks that can be used to segment the sequence. Each chunk in the lexicon is assigned a weight, which represents the chunk's strength of encoding. As PARSER encounters instances of a chunk, the weight of the chunk is increased. Chunk weights decrease during every step of the model, so evidence must be encountered frequently enough to prevent PARSER from forgetting the chunk. PARSER can be conceptualized as a competitive learning process, in which candidate chunks are created in a semi-random process and only those chunks that are repeatedly encountered in the sequence will persist.

PARSER creates candidate chunks by randomly selecting to simultaneously process the next 1, 2, or 3 chunks of the sequence. If 2 or 3 chunks are processed at a

time, then a new chunk is added to the lexicon that is the combination of the 2 or 3 chunks processed on that step. This is PARSER's only mechanism for generating new chunks. Early on in learning, when PARSER has no chunks in the lexicon, chunks are equivalent to the primitive elements of the sequence. Once PARSER acquires sufficient evidence for a chunk, it will process the sequence using that chunk. For example, the first step of PARSER, given the sequence of `ABCDEFG` and a random decision to process 3 items in the step, will be to form a new chunk `ABC`. Later on in learning, if PARSER has the chunks `ABC` and `DEF` in its lexicon, it would process `ABCDEFG` in one step (assuming that it was processing 3 chunks at once), because `ABC` and `DEF` are each one chunk. This would result in the creation of a new candidate chunk `ABCDEFG`.

New chunks are added to the lexicon with some initial weight (the value is a free parameter in the model). When a chunk that is already in the lexicon is encountered, its weight is increased. Chunks with weights above a shaping threshold (an additional free parameter in the model) are considered part of the active lexicon used for processing the sequence. Chunks with weights below the shaping threshold are not used for processing the sequence. On each step of the model, a forgetting process causes all chunks to decrease in weight. There is also an interference process, in which chunks with any of the same primitives as the chunk being processed on the current step have their weight decremented.

2.2.1.2 MDLChunker (Robinet et al., 2011)

MDLChunker, like PARSER, is a lexicon-based model with an explicit representation of the chunks that it has acquired. MDLChunker creates chunks via an application of the minimum description length (MDL) principle. The MDL principle is

based on the idea that a compressed representation of a data set can be generated when regularities exist in that data. MDLChunker applies this principle by creating new chunks to encode a sequence of primitives only when the addition of the chunk will result in a shorter description (in terms of the number of bits) of the sequence of items *and* the lexicon of chunks. Adding a chunk to the lexicon increases the number of bits required to encode the lexicon, so this cost must be offset by an at least equally large compression of the sequence in order to add a new chunk.

I used the memory-limited version of MDLChunker, which contains two structures: the lexicon and a memory buffer. The memory buffer contains the set of primitives recently seen by the model, with their sequential order intact. The lexicon contains the set of chunks used to encode the primitives in the memory buffer. On each step of the model, new primitives from the input are added to the memory buffer and an optimization routine checks to see if adding new chunks could decrease the overall description length of the model. If new chunks are found that decrease the description length, then they are added to the lexicon. The memory buffer has a limited capacity expressed in bits. Thus, the maximum number of items in memory depends on how compressible the items are.

Each item in MDLChunker's lexicon has an associated bit length. This is a measurement of the amount of information needed to encode instances of the chunk. Chunks with a high relative frequency will have a short bit length. In this application of MDLChunker, I take the bit length to be representative of the strength of encoding. As the number of bits associated with a sequence of primitives increases, I assume that the likelihood of successfully retrieving it from memory decreases because each of the

increased likelihood of failing to retrieve one or more of the bits. Therefore, shorter bit lengths are associated with more strongly-encoded chunks.

2.2.1.3 TRACX (French et al., 2011)

TRACX is a connectionist model of chunk learning. TRACX is an auto-associative three-layer network. The input layer represents two adjacent items (either primitives or chunks) from the sequence (called the left- and right-hand items, with the left-hand item occurring temporally before the right-hand item), the hidden layer forms a compressed representation of the input, and the output layer recreates the input. The input and output layers are fully connected to the hidden layer. The hidden layer has fewer units than the input and output layers (exactly half, in our implementation), which requires the hidden layer to form a lower-dimensional representation of the sequence. The network is exposed to pairs of items sequentially, and back-propagation is used to adjust the weights so that the output layer better matches the input.

The key innovation that enables TRACX to learn chunks is that the network will use the hidden layer activations as the left-hand item in the next input step when the error in reconstruction is low. Low reconstruction error occurs when the input is familiar to the network, and thus is likely to be a chunk. The distributed pattern of activity on the hidden layer is a representation of the chunk. Initially, TRACX will learn only two-primitive chunks, but as these chunks are learned and subsequently become part of the input, then longer chunks can also be learned.

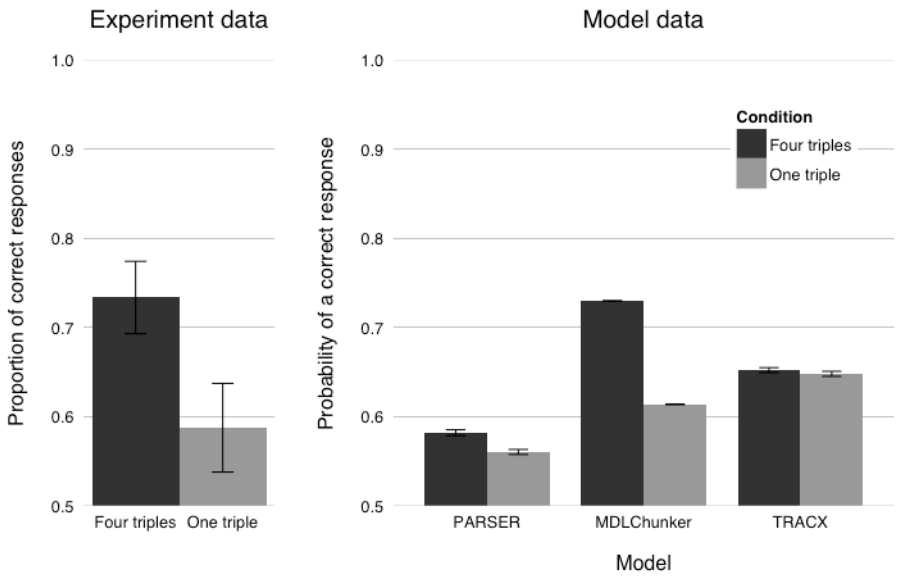

Figure 2. Data and model fits for four-triple and one-triple sequences. Model data is generated from 1,000 randomly generated sequences in each condition. Error bars are one standard error of the mean.

2.2.2 Model Evaluation

2.2.2.1 Model Implementation

I used a publicly available implementation of TRACX (Addyman, 2015), which required no modifications. I created my own implementations of PARSER and MDLChunker[2]. All three models are implemented in JavaScript, and available in the OSF repository: https://osf.io/p6x5f/

I used the default parameters for all models. For PARSER, the forgetting rate was 0.05, the interference rate was 0.005, the reinforcement rate was 0.5, the shaping threshold was 1, and the initial weight for new chunks was 1. For MDLChunker, the

[2] I thank Vivien Robinet for providing C++ source code for MDLChunker, which aided the development of the JavaScript implementation.

perceptual span was 25 bits, and the maximum memory size was 150 bits. For TRACX, the learning rate was 0.04, the recognition criterion was 0.4, and the reinforcement probability was 0.25.

2.2.2.2 Procedure

I tested each of the models with 1,000 different sequences in both the four-triples and one-triple conditions. The training sequences were created in the exact same manner as for the human experiment, except that a unique letter was used to represent each different shape. Each model generated two values, indicating the strength of encoding for the target triple and an impossible test triple. For PARSER, these values were the weight of the triple in the lexicon. If the item had a weight below the shaping threshold (the value at which the model interprets the lexicon item as a chunk), then the item was treated as having a weight of zero. For MDLChunker, the negative code length (in bits) for the triple was used. For TRACX, the negative network recognition error (the difference between the input and output layers) of the triple was used. For both MDLChunker and TRACX, negative values were used so that larger numbers represented stronger encodings.

To convert from the model generated strength-of-encoding values to the probability of a correct response in the two-alternative forced choice task I used a softmax function:

$$p(correct) = e^{\gamma a}/(e^{\gamma a} + e^{\gamma b})$$

In the above equation, a represents the strength of encoding for the target, b represents the strength of encoding for the foil, and γ is a parameter that modulates the degree of randomness of the response. High γ values result in very deterministic

behavior, where any difference in the strength of encoding for the two items will result in an extreme probability (near 0 or 1) for the response. Low γ values, on the other hand, produce more random behavior. The strength-of-encoding values are all on very different scales for each model. To allow each model the opportunity to fit the experimental data despite the differences in scales, I fit γ separately for each model using a hierarchical model and Bayesian estimation. This model and procedure is described in Appendix A.

The hierarchical model generated a distribution of credible γ values for each model. To generate simulated experimental data, I randomly sampled a γ value from the distribution for each run of the model. I then computed the probability of a correct response for that model run with the softmax function, using the strength-of-encoding values generated by the model and the randomly sampled γ value.

2.2.2.3 Results

MDLChunker and PARSER replicated the effect observed in the experiment: the target triple was correctly identified more often in the four-triples condition than in the one-triple condition (Figure 2; MDLChunker: $t(1998) = 205.9$, $p < 0.0001$, 95% CI: .115 to .117; PARSER: $t(1998) = 5.0$, $p < 0.0001$, 95% CI: 0.013 to 0.030). There was no evidence that TRACX showed a difference in performance between the two conditions, $t(1998) = 1.04$, $p = 0.29$, 95% CI: -0.003 to 0.012. Though both PARSER and MDLChunker predicted the correct qualitative pattern of results, MDLChunker quantitatively fits the data better than PARSER.

The best-fitting γ values varied for each model. MDLChunker required the lowest γ values to fit the experimental data (95% HDI: 0.0597 to 0.0604). PARSER also

required relatively low γ values (95% HDI: 0.165 to 0.168), while the γ values for TRACX were substantially higher (95% HDI: 1.02 to 1.03).

2.2.3 Discussion

MDLChunker clearly provides the best overall fit to the experimental data, so it is worth briefly exploring why this particular model predicts a difference in performance based on the sequence structure. The strength of encoding measure for MDLChunker is the code length of the target and foil. Code length is a measure of the relative frequency of the chunk in the sequence and lexicon, given the other chunks that the model possesses. With optimal learning of the chunk structure, the relative frequency of the target in the four-triples condition will be about 1 in 4, since there are four triples that appear with equal frequency. In the one-triple condition, the target will have a relative frequency of only about 1 in 10, since there are nine singletons and one triple. In short, MDLChunker predicts that the target triple is more strongly encoded in the four-triples condition because the surrounding information is compressible, which increases the relative frequency of the target.

PARSER also fits the qualitative pattern of results, and the underlying reason is similar to MDLChunker. In PARSER, each chunk is assigned a weight, which increases every time PARSER is exposed to the chunk. The chunk's weight decreases every step of the model that PARSER is not exposed to the chunk. In order for PARSER to strongly encode a chunk, it must be exposed to the chunk frequently enough that the reinforcement process outpaces the forgetting process. Thus, as with MDLChunker, the relative frequency of a chunk matters a great deal in how strongly the chunk will be

encoded. And, as with MDLChunker, the compressibility of the surrounding information will affect the relative frequency of the target chunk.

To illustrate this, imagine PARSER is presented with the sequence ABCGHIDEFABC. If PARSER contains no chunks, and randomly selects to see 3 chunks during the next processing step, then the input on that step will be A/B/C. But, if PARSER has already learned the chunks ABC, GHI, and DEF, then the input would be ABC/GHI/DEF. In both cases, the chunk ABC will be reinforced, increasing its weight in memory. However, on the next step, the version with no chunks will see the input G (supposing that PARSER randomly selects 1 unit as the input), and the ABC chunk will decay slightly in memory. The version with chunks will see ABC again because it has already processed the first nine primitives in the sequence, reinforcing ABC even further. When PARSER is able to chunk the input sequence, it can process the input in fewer model steps, as shown by this toy example. This has the effect of accelerating the exposure rate of chunks, or, equivalently, decelerating the rate at which chunks are forgotten. Given that the decay rate of items in memory is fixed to the number of model steps, an individual chunk will experience less decay between successive presentations when the intermediate sequence is compressible. PARSER, in essence, behaves like it has a longer lasting memory in terms of primitive elements when the input sequence is compressible than when it is not.

TRACX, in contrast to both MDLChunker and PARSER, has neither explicit memory storage nor any explicit forgetting parameter. TRACX processes a sequence at a rate of one primitive per step regardless of previous learning. Memory constraints in TRACX depend on interference during the learning of connection weights. The same

number of weight updates happen regardless of whether the model has learned to chunk the input or not. Therefore, TRACX lacks the kind of compressible-memory mechanism that I hypothesize is responsible for the observed effect.

2.2.4 Variations on the PARSER Model

A reasonable hypothesis, given the modeling results, is that a compressible memory mechanism is responsible for the observed difference in learning. In this section, I investigate that hypothesis further by examining what happens in the PARSER model when the model can remember everything or nothing at all, and when the model has a non-compressible memory. I explored the PARSER model in more detail because while MDLChunker fits the experimental data better than PARSER, MDLChunker is less modular in its design. PARSER can be modified in a conceptually straightforward way to remove the compressible memory function, while MDLChunker would require a more convoluted modification that would likely disrupt other aspects of the model as well.

2.2.4.1 The Forgetting Rate Parameter.

PARSER has a parameter to adjust the rate at which chunks in the lexicon are forgotten. On each step of the model, every item in the lexicon has its weight decreased by this amount. Small values of this parameter give PARSER the ability to remember more candidate chunks in the lexicon, while large values cause PARSER to forget all candidate chunks too quickly to form stable chunks. The default value for this parameter is 0.05 (Perruchet & Vinter, 1998), and experiments that fit human data with PARSER typically use values in the range of 0.025 to 0.085 for the forgetting rate (Frank, Goldwater, et al., 2010; Giroux & Rey, 2009; Perruchet & Peereman, 2004; Perruchet & Tillmann, 2010; Perruchet, Vinter, Pacteau, & Gallego, 2002; Perruchet & Vinter, 1998).

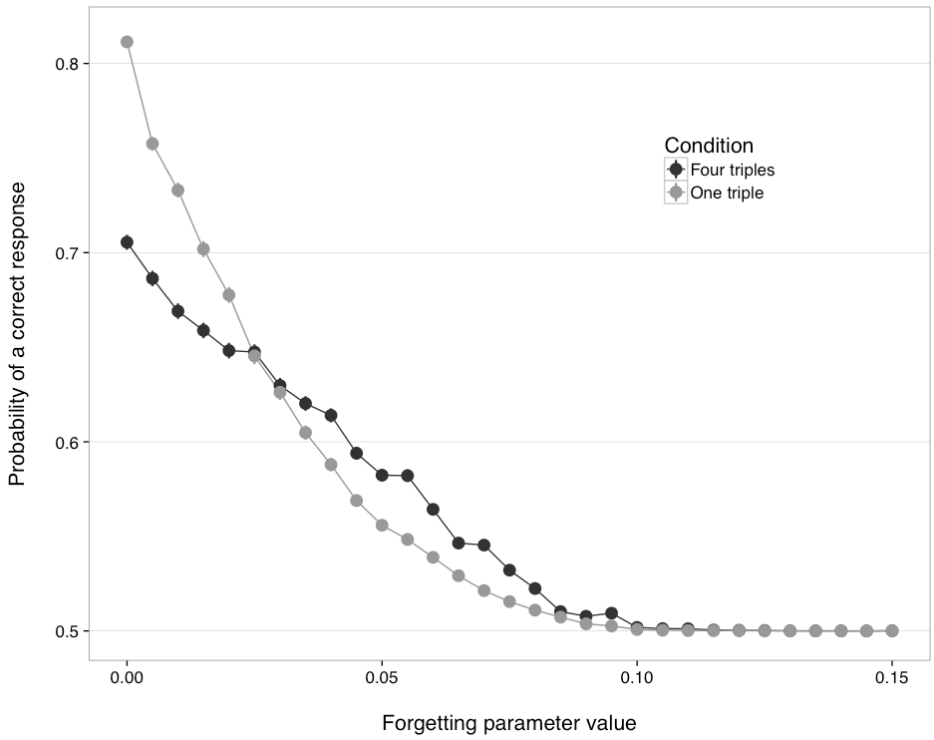

Figure 3. The effect of variations in PARSER's forgetting parameter. Each point is the mean of 1,000 simulations, and the error bars are one standard error of the mean. At low values of forgetting rate, PARSER performs better in the one-triple condition. At moderate values of the forgetting rate, PARSER performs better in the four-triples condition. At high values, PARSER performs at chance (0.5) in both conditions.

We can therefore think of the values in this range as representing a typical human level of forgetting.

I tested how variations in the forgetting rate affect PARSER's relative performance in the two experimental conditions. I started with the forgetting rate set to 0 (PARSER remembers everything) and increased it in steps of 0.005 until PARSER was unable to remember anything long enough to add items into the lexicon. This happened around a value of 0.10, and I continued the stimulation to a value of 0.15 to be sure that performance was asymptotic. I ran 1,000 simulations in each condition at each level of the forgetting parameter. I used a γ value of 0.166 for the softmax rule, based on the previous model fits (however, variations in γ do not change the direction of the effect,

just the scale, so this parameter is inconsequential for this demonstration). As shown in Figure 3, low forgetting rates result in a reversal of the empirically observed effect: the target is learned better in the one-triple than four-triples condition. High forgetting rates prevent PARSER from learning anything, and performance is equivalent in both conditions. Moderate forgetting rates, in the range typically used for modeling human data, generate the observed effect.

Why does PARSER learn the target triple better in the one-triple condition than in the four-triple condition when the forgetting rate is very low? As the forgetting rate decreases sufficiently, then even very low frequency patterns are likely to be encoded as chunks. In the four-triples condition, there are only three possible transitions after each triple. With a low forgetting rate, PARSER remembers each of these transitions, and as a result forms long chunks that contain multiple triples. Once these long chunks are formed, the smaller chunks are forgotten, because the stream is processed using the longer chunks and no reinforcement is given to the shorter chunks. The formation of extended chunks is less likely in the one-triple condition, where there are nine possible transitions after the triple. This makes it less likely that PARSER will encounter the same pattern enough times to memorize a pattern longer than a chunk. The end result is that PARSER forms a stronger encoding of the target triple in the one-triple condition, because it is more likely to encode the true chunk as opposed to a chunk that is made up of multiple triples.

2.2.4.2 Non-compressible Memory

The standard version of PARSER has a compressible memory mechanism. As PARSER learns to chunk the input, it needs fewer steps to process the same number of

input tokens. Because the forgetting function occurs once per step of the model, chunking the input reduces the forgetting rate of other chunks. As described in the previous section, this is likely the reason that PARSER shows differences in learning a triple when that triple is embedded in four-triple versus one-triple sequences.

To test this more directly, I created a modified version of PARSER with a forgetting function that occurs at a constant rate regardless of how compressible the input is. I implemented this by multiplying the forgetting rate by the number of primitives processed during that step of the model. No matter how many primitives PARSER is able to process in a single step, the forgetting rate is constant in terms of the number of primitives seen. Note that this modification increases the overall amount of forgetting substantially; lower values of the forgetting rate parameter are necessary for PARSER to be able to remember anything.

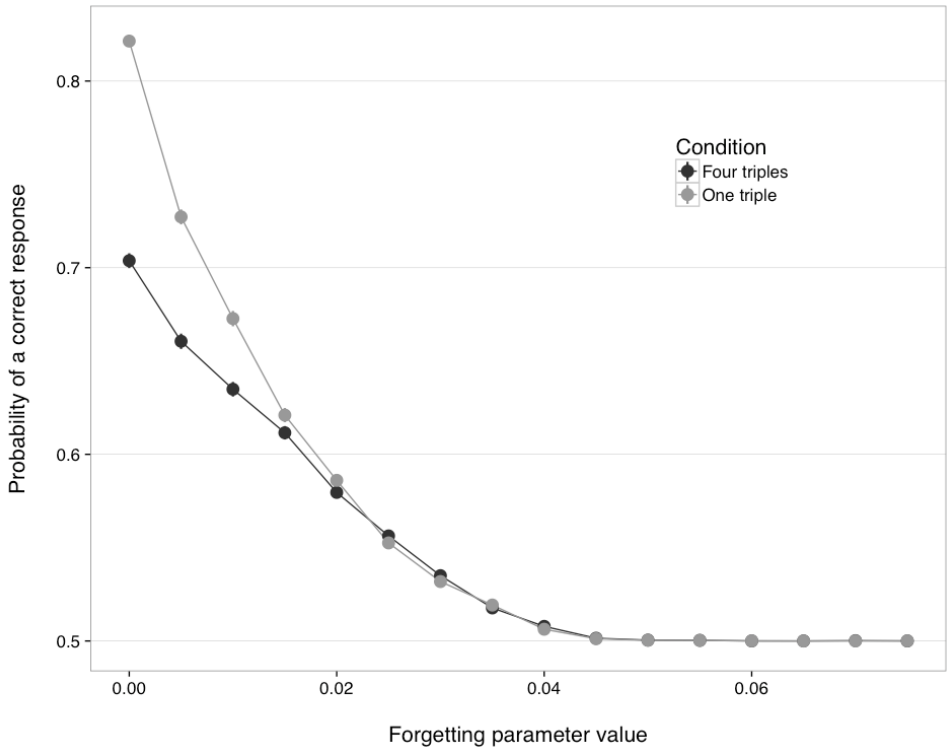

Figure 4. PARSER model with non-compressible memory. Each point is the mean of 1,000 simulations, and the error bars are one standard error of the mean. There is no parameter value at which the modified version of PARSER performs better in the four-triple condition.

I tested the modified version of the model with the same procedure as the standard version, stopping at a forgetting rate of 0.075. As shown in Figure 4, there is no parameter value in which learning is better in the four-triples than one-triple condition. In order for PARSER to mimic the difference in learning observed in the experiment, the forgetting rate of items in the lexicon must depend on the number of chunks that PARSER is exposed to, not the number of primitives that it has seen.

2.3 Experiment 2.2

The behavioral and modeling results from Experiment 2.1 suggests that learning the statistical structure of a sequence enables the formation of compressed chunks freeing up working memory resources for learning other chunks in the sequence. If this account

is right, then a similar effect should occur when the statistical structure is identical in both conditions, but participants already know some of the chunks in one condition but not the other. In other words, what should matter is the chunks that the learner possesses, not the statistical properties of the sequence itself.

To test this idea, I created a variation of Experiment 2.1 in which both sequences were built using twelve letters arranged into four triples. I used letters instead of shapes to allow people to use their pre-existing lexical knowledge as a way of manipulating whether the chunks are known or novel. In one condition, three of the four triples formed familiar words and in the other condition the same letters were arranged into novel words. This way of manipulating chunk-based knowledge seemed better than a training manipulation, because it allowed the underlying task to remain covert.

2.3.1 Method

2.3.1.1 Participants

178 people, recruited through Mechanical Turk, participated in the experiment. Five participants answered the test questions twice, presumably by refreshing the experiment in their browser and completing it a second time, and were removed from the analysis, making the final number of participants 173.

2.3.1.2 Procedure

Participants completed the experiment in a setting of their choice using a web browser of their choice. The entire experiment took approximately four minutes to complete. Upon loading the experiment page, participants saw the following instructions: "You are going to see a series of letters, presented one at a time. Watch the letters

carefully. After viewing all the letters, which takes about 2 minutes, we will ask you a few questions about what you remember. When you are ready to start, press any key."

Participants then viewed a sequence of 240 letters. Each letter was on the screen for 400 milliseconds, with a 100 millisecond blank screen between each letter. The letters were presented in a large font; though exact display size is impossible to determine because it varied among participants due to the experiment running online.

The sequence consisted of twelve different letters, each presented 20 times. The twelve letters were grouped into four sets of three letters. Letters within a group always appeared in the same order. The groups of three letters were all high-frequency three-letter English words: CUT, DAY, HER, and WIN. Each word followed a consonant-vowel-consonant pattern. Importantly, no other permutation of any of these words is itself a word.

Participants were randomly assigned to either the known-words or the novel-words condition. In the known-words condition, one of the four words was randomly selected to be flipped into a non-word by swapping the first and third letter, e.g., CUT becomes TUC. The other three words remained the same. In the novel-words condition, all four words were flipped. Thus, in both conditions there is at least one target word that the participant is unfamiliar with (e.g., TUC), that is either surrounded by either words the participant knows or other unfamiliar words.

The sequence was built by concatenating the words together in a random order, using the following algorithm: Randomly shuffle the order of the four words. If the first word in the random order is not the same as the last word in the current sequence, then concatenate the four words onto the sequence. Otherwise, shuffle the words again until

the above condition is met, and then append them onto the sequence. This algorithm achieved two purposes: (1) a word could not immediately repeat and (2) the words were distributed equally throughout the sequence.

After viewing the sequence of letters, participants answered four test questions. Each question simultaneously presented all six permutations, in a randomly ordered list, of one of the four words that the participant saw during the sequence (e.g., CUT, CTU, TUC, TCU, UCT, and UTC). Participants attempted to select the order of letters that matched the order that they saw in the sequence.

2.3.2 Results

The primary comparison of interest is the proportion of correct responses to flipped words in the known-words condition versus the novel-words condition. In the known-words condition, there was only a single trial where the target word was a flipped word. In the novel-words condition, all four trials were flipped words. Thus, in the known-words condition the datum for a single participant is either 0 or 1 correct responses out of 1 attempt. In the unknown-words condition, it is N correct trials out of 4 attempts. To unify these conditions despite the different number of trials, the analysis model treats each participant's data being generated by a binomial distribution with N equal to 1 in the known-words condition and 4 in the unknown-words condition. As with experiment 2.1, the model estimates the probability of success for each subject and generates a hierarchical group-level estimate. The difference in the probability of success at the group level is the parameter of theoretical interest. The analysis model and fitting procedure are described in Appendix A.

The 95% HDI for the difference in probability of success between conditions was 0.0597 to 0.501, with a mode of 0.22. Because the HDI is reliably above 0 and values reasonably close to 0, the data indicate that participants were better at picking the correct order of novel words in the known-words condition.

2.4 Modeling Experiment 2.2

This section compares PARSER and MDLChunker on the task from Experiment 2.2. In Experiment 2.1, PARSER and MDLChunker both predicted that a target chunk is more strongly encoded when the surrounding sequence is compressible. While both models share a similar commitment to the role of compressibility of the contextual sequence, they implement this process in different ways.

MDLChunker is primarily influenced by the statistical structure of the sequence regardless of prior exposure. At each processing step, new chunks are formed and code lengths are updated based on a relatively lengthy memory of the sequence. The only significant influence of the internal lexicon of the model is that more of the sequence can be remembered when MDLChunker has several chunks, since the short-term memory buffer capacity is defined in bits. In contrast, PARSER has a great deal of dependency on prior learning. There is no short-term memory buffer like there is in MDLChunker, and the encoding weights that represent previous experience depend heavily on the stochastic processes that govern which candidate chunks PARSER experiences on each step of the model. This difference in how the models treat the role of prior experience should influence how successful they are at predicting the empirical data from Experiment 2.2.

2.4.1 Procedure

2.4.1.1 Comparing PARSER and MDLChunker

The testing and fitting procedure was similar to Experiment 2.1 with a few minor modifications. Most important was to implement "prior knowledge" into the models. For PARSER, the lexicon was seeded with the three known words as tokens with a high enough weight (100) that the item would not decay below the shaping threshold over the course of the experiment. To seed MDLChunker, the model was exposed to a pre-training sequence of the word tokens intermixed with the individual letters from the target word. For example, if the known words are represented as `DEF`, `GHI`, and `JKL`, and the target word is `ABC`, then the pre-training sequence would be something like: `DEFBGHIAJKLJKLAGHIADEFDEFGHIBJKLGHI`…. The pre-training sequence contained 300 lexical items, where a lexical item is either a known word or a letter from the target word. The sequence was generated by sampling with replacement from the six possible lexical items. The pre-training sequence was used for MDLChunker because the model recalculates the bit length of each chunk in the lexicon after each step. Seeding the lexicon with the three chunks would have no effect on subsequent steps. The pre-training approach seeds MDLChunker's memory buffer, which affects the bit length calculations during the early stages of training, persisting until the pre-training sequence is overwritten in memory.

PARSER and MDLChunker were each tested 1,000 times in both the known-words and novel-words conditions. For each run, the model was trained on a randomly generated 300-item sequence, assembled in the same manner as the four-triples condition in Experiment 2.1. Learning was measured in a similar manner to Experiment 2.1,

measuring the weight (PARSER) or bit length (MDLChunker) of the target and foil words. The weights for PARSER were calculated by taking the total weight of the three individual letters that made up the word, plus the weight of the three-letter chunk (which was always 0 for the foil, because PARSER cannot learn spurious chunks). For MDLChunker, the average bit length of the five foil words was used as the foil weight. These weights were multiplied by -1, so that smaller bit lengths had a greater value in the decision making process.

The decision process was again modeled using a softmax rule, adjusted from the version used in Experiment 2.1 to reflect the six alternative choices. The portion of the denominator that reflects the foil weight was multiplied by 5 to account for the five foils.

$$p(correct) = e^{\gamma a}/(e^{\gamma a} + 5e^{\gamma b})$$

As in Experiment 2.1, γ controls the stochasticity of the decision process, and *a* and *b* are the target and foil weights, respectively. The value of γ was fit using Bayesian estimation. For details of this procedure, see Appendix A. After fitting the value of γ, simulated experimental data was generated by randomly sampling a γ value from the posterior distribution for each of the 1,000 runs of the model. I then computed the probability of a correct response for that model run with equation 1, using the strength-of-encoding values generated by the model and the randomly sampled γ value.

2.4.1.2 PARSER Variations

To test whether the compressible-memory hypothesis is consistent with the observed results, I tested the PARSER model using the same two variations as in Experiment 2.1. First, I tested the model with different values for the forgetting rate parameter. The parameter varied between 0 and 0.4, tested in increments of 0.002. For

each parameter value, the model was run 1,000 times in both conditions. The predicted probability of a correct response was generated using the softmax rule with a γ value of 0.226, which was the mode of the posterior distribution from fitting γ to the experimental data.

The non-compressible memory version of PARSER, developed in section 2.2.4.2, was also tested with a range of values for the forgetting parameter (0 to 0.07 in increments of 0.01). The same procedure was used to generate the predicted probability of a correct response.

2.4.2 Results

Only PARSER replicated the observed advantage for known words over novel words (Figure 5). PARSER predicted an 0.085 increase in the probability of success for the known words condition over the novel words condition, $t(1877.4) = 12.341$, $p < 0.0001$. MDLChunker predicted a miniscule difference in the opposite direction, an

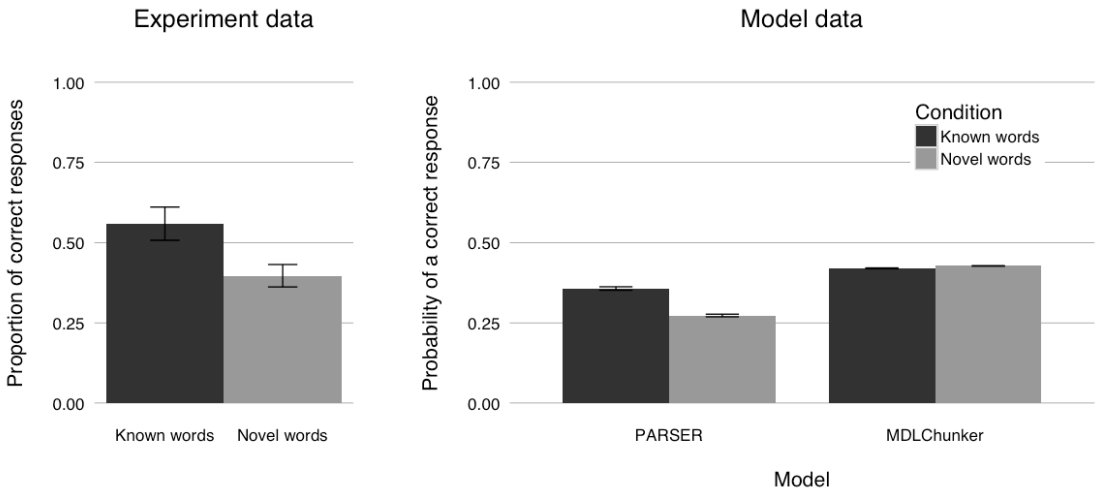

Figure 5. Experiment 2.2 Results and Model Fits. Model data was generated from 1,000 randomly generated sequences in each condition. Error bars are one standard error of the mean.

increase of less than 0.01 in the probability of success for novel words, $t(1725.2) = -4.953$, $p < 0.0001$.

PARSER predicted an advantage for the known words condition over the novel words condition at all small-to-moderate values of the forgetting parameter. At larger values, performance is at chance levels in both conditions, as PARSER cannot remember any of the chunks it sees. The difference between the conditions is largest at intermediate levels of forgetting when performance in the novel words condition drops to chance levels, but performance in the known words condition remains well-above chance.

PARSER with non-compressible memory also predicts an advantage for the known words condition over the novel words condition, though this difference is smaller than the corresponding predictions in the standard PARSER model. For example, when modified-PARSER predicts a 0.297 probability of a correct response in the known words condition, the prediction for the novel words condition is 0.262. When standard PARSER predicts a similar level of performance in the known words condition (0.301), the prediction for the novel words condition is only 0.172, a difference of 0.129.

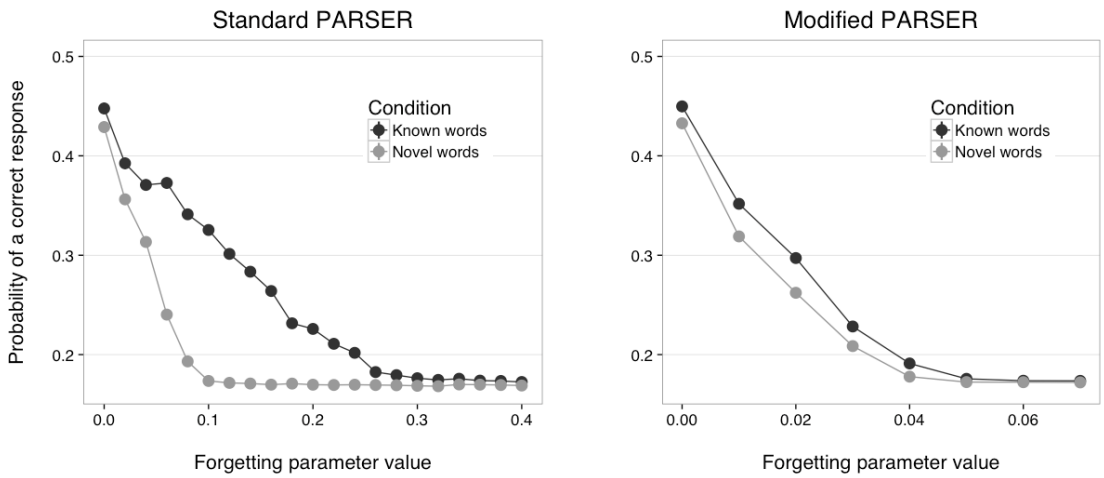

Figure 6. Variations of the PARSER model tested on Experiment 2.2. Each point is the mean of 1,000 simulations, and the error bars are one standard error of the mean. The left panel shows an unmodified version of PARSER. The right panel shows a version of PARSER that lacks a compressible memory. See section 2.2.4.2 for a description of modified PARSER.

2.4.3 Discussion

Only PARSER predicts that learning a novel triple will be easier in the context of known words. PARSER gets an advantage for encoding the novel target in the context of known words because processing the portions of the sequence that contain known words causes less interference for the target word. Modifying the PARSER model to remove the compressible memory mechanism greatly reduces, but does not eliminate, the advantage of the known-words context. This is likely because there is an additional way that PARSER can use the known-words: to segment the input into more probable chunks. PARSER randomly chooses how to segment the input to form candidate chunks, but does this in the context of pre-existing chunks, which are always kept together. PARSER is more likely to segment the input in a way that is consistent with the underlying statistical structure when it already knows some of the chunks. However, the benefit of this appropriate segmentation is minor compared to the benefit of compressible memory.

In MDLChunker, previous knowledge affects how many items can be stored in a short-term memory buffer. In some ways, this constraint plays a similar functional role to PARSER's compressible memory, yet MDLChunker does not predict a difference between the known- and novel-words conditions. This is because MDLChunker has relatively little state dependence – the only thing that matters for computing the representational strength of an item in MDLChunker's lexicon at time *t* is the relative frequency of that item in the memory buffer and lexicon.

Thus, the main theoretical difference of these two models is how the elements of a sequence are processed and stored. PARSER's strategy suggests that items are processed into chunks at encoding. If PARSER perceives the letters `A B C` as an `AB` chunk plus isolated `C` during encoding, that cannot be overwritten later by evidence that the more frequent chunk is `BC`. Processing decisions about the input must happen in the moment and are not subject to later revision; Christiansen and Chater (2016) called this the *now-or-never bottleneck*. MDLChunker, on the other hand, stores the raw sequence and runs a compression algorithm on the whole sequence at each step of the model, reinterpreting evidence that it had previously seen in light of new evidence. This makes it difficult for MDLChunker to capture effects of preexisting chunk knowledge; MDLChunker may overwrite its entire knowledge base in a single model step if it improves compression.

2.5 General Discussion

In this chapter, I investigated the way that memory constraints interact with statistical learning. Prior work found that limited memory capacity is a constraint on statistical learning (Frank & Gibson, 2011; Frank, Goldwater, et al., 2010). These previous characterizations of the role of memory constraints in statistical learning have

focused on how memory constraints limit statistical learning, or how statistical learning affects memory constraints (Brady et al., 2009). I found evidence that the interaction between memory processes and statistical learning forms a dynamic constraint. Statistical learning of structure in a sequence allows for chunking of the sequence, which reduces the impact of memory constraints and promotes further learning, which allows for further chunking, and so on.

In Experiment 2.1, I found that it is easier for adults to learn a subsequence of three shapes when the shapes are embedded in a sequence that has a relatively compressible (or chunk-able) structure than when it is embedded in a sequence that is less compressible. Computational models of statistical learning predicted this result if they had memory constraints that were sensitive to chunk-based compression. They failed to predict the result if they lacked memory constraints or if the memory constrains were not sensitive to chunks (i.e., chunking did not improve memory capacity). Furthermore, learning was enhanced in a particular way in the models. Chunking of the sequence led directly to a relatively better memory for other items in the sequence, regardless of whether those items were already chunked. This promoted the acquisition of new chunks because memory for instances of the non-chunked items was relatively stronger when surrounding information was chunked.

Despite their similar functional behavior with compressible structures, MDLChunker and PARSER have different algorithmic implementations of this process. The PARSER approach is similar to retroactive interference: PARSER forgets previously seen chunks as it sees new chunks. However, since the unit of interference is the chunk instead of the primitive, learning to chunk decreases the overall amount of interference.

MDLChunker's implementation is more like a competitive search in memory: the relative strength of any particular chunk depends on how frequent that chunk is in memory. When the model learns to chunk new items, the relative frequency of a particular chunk is increased since chunking reduces the total number of items in memory by grouping together primitives into chunks. A major difference in these two approaches is that MDLChunker predicts that learning chunks *after* exposure will result in relative increases in strength of encoding, but PARSER's implementation depends on acquiring chunks during learning in order to show the same effect. This difference was apparent in MDLChunker's failure to predict that prior knowledge of some chunks would lead to better learning of a new chunk in Experiment 2.2.

The observation that memory for non-chunked items can be improved by forming chunks for contextual information is not new. Bower (1969) investigated the role of chunking in verbal free-recall experiments. He presented participants with either a list of words that contained short idiomatic expressions that were familiar to the participants (e.g. *ice/cream/cone, happy/new/year*) with additional words not embedded in phrases or a list of unrelated words. Participants were more likely to recall words from the list with idiomatic phrases, even if the word did not appear in one of the phrases. Bower concluded that chunks, not individual items, act as the unit of interference in free-recall tasks. This observation closely matches the computational processes implemented by the models that predicted the empirical results and adds further evidence that this is a plausible mechanism for explaining the difference in learning between the two types of sequences. The results from this chapter extend this finding by demonstrating how

compressible-memory mechanisms influences the learning of new information in a task that is not explicitly requiring participants to memorize items.

One kind of objection to these results is that I only tested a handful of models and other models may predict the experiment result. This is certainly likely, and two different cases would be particularly interesting. One hypothesis that follows from the results is that any model with compressible memory will show the effect. This is a claim of *sufficiency*. If a counterexample were found, then it would refine our understanding about what kinds of processes are needed to replicate the experiment data. The other hypothesis – which is much more constraining and less likely – is that memory compressibility is *necessary* to replicate the experiment data. A successful replication by a model without memory compressibility would suggest there are multiple possible accounts of why learners have an easier time learning chunks embedded in compressible sequences. One possibility along these lines is that attention could explain the findings. Previous work has shown that sequences with chunks of a consistent size are easier to learn than sequences with varying chunk sizes (Johnson & Tyler, 2010). Learners may preferentially attend to certain chunk sizes based on exposure, and there is increased exposure to chunks of size 3 in the four-triple condition. This would suggest that the dynamic interaction between cognitive and learning processes is between attention and learning instead of memory and learning. It is likely that attention and memory both affect statistical learning constraints in dynamic ways. Further empirical work is necessary to determine the nature of the constraints.

There is a long history of research on how previous experiences in statistical-learning contexts can affect subsequent learning (e.g. Gebhart et al., 2009; Lany &

Gómez, 2008; Lew-Williams & Saffran, 2012). I have shown that one possible mechanism for experience-dependent effects on learning is the dynamic interplay between memory and learning. Instead of viewing memory as a constraint on learning, we should view memory and learning as interactive processes that generate dynamic constraints.

Acknowledgment. Portions of the data and modeling presented in this chapter initially appeared in de Leeuw & Goldstone (2015).

3 Learning Curves in Sequential Statistical Learning

Much of the foundational work on statistical learning from sequential information was concerned with whether variations in the distribution of transitional or joint probabilities between items in a sequence was sufficient to extract the units from the sequence (Aslin et al., 1998; Fiser & Aslin, 2001, 2002; Saffran et al., 1996; Saffran, Johnson, Aslin, & Newport, 1999; Turk-Browne et al., 2005). As in the experiments presented in Chapter 2, studies about the sufficiency of variations in statistical structure for chunk extraction typically use a learn-then-test approach. This approach is appropriate when the goal is to see *if* people can learn to segment a particular sequence, but it provides little information about the learning process. Some models of statistical learning do not attempt to describe the process of learning, instead focusing on explaining and predicting which units people will learn in given contexts (Brent & Cartwright, 1996; Goldwater, Griffiths, & Johnson, 2009; Orbán et al., 2008). For these models, the learn-then-test approach is a good method for model evaluation. Other models, like those described in Chapter 2, do try to describe the learning process, and therefore make predictions about the time course of learning. While some evaluations of these models can be made using a learn-then-test strategy, the most powerful way to evaluate models that make predictions about the learning process is to measure learning as it happens.

To illustrate this point further, consider the results from Experiments 2.1 and 2.2. All of the tested models correctly predict learning. In fact, all of the models predict a relatively strong encoding of the target relative to the foils in all of the experimental conditions. This is expected because the foils were groupings that the model never saw. PARSER, for example, never encoded the foil as a potential triple, and only exhibited less than perfect performance because (a) the softmax rule to fit the data used a very

noisy decision rule, and (b) sometimes the model did not learn the target in the time allotted. The second point is a critical one. Given enough exposure, PARSER would have learned the target every time in all conditions. The difference in performance (both model and human) across conditions is partially attributable to the limited exposure to the sequence. The pattern of results may have been different had participants and the model been tested after a different length of exposure to the sequence.

Measuring performance over time (generating a learning curve) solves this problem. The importance of learning curves for model selection has long been recognized in several areas of psychology (Estes, 2002). One of the central themes of this literature is the difference between the learning curves at the group and individual levels (Ackerman, 1987; Anderson & Tweney, 1997; Heathcote, Brown, & Mewhort, 2000; Sidman, 1952; Wixted & Ebbesen, 1997). Any particular average learning curve may have been generated by an infinitely large variety of individual curves. The inferences that can be drawn about individual behavior from the group average depend on the mathematical relationship between the functional form of the individual and group curves (Estes, 1956). The averaging process often obscures the shape of the individual-level curves. For example, when the slopes of individual-level curves contain sufficient variability, the group-level learning curve resembles a power law, even if the individual-level curves are linear, logarithmic, exponential, or even discontinuous, which may explain the ubiquity of power law shaped learning curves (Anderson, 2001; Haider & Frensch, 2002; Murre & Chessa, 2011; Myung, Kim, & Pitt, 2000).

The average learning curve appears gradual even in cases where learning is sudden if learners arrive at the solution at different points in time. Studies of many forms

of associative learning have found that learning at the level of individual items for individual participants may be sudden and fast (Estes, 1960; Gallistel, Fairhurst, & Balsam, 2004; Rock & Heimer, 1959; Rock, 1957; Trueswell, Medina, Hafri, & Gleitman, 2013) – "insightful" (Köhler, 1925, 1959)– even when patterns at the group level are steady and gradual. On the other hand, there are also many cases in which the learning curves for individual participants and items follow a gradual learning curve (Heathcote et al., 2000) or a hybrid of gradual improvement with occasional sudden improvement (Donner & Hardy, 2015).

Even though "learning curves of almost every conceivable shape have been found" (Mazur & Hastie, 1978, p. 1257) we might consider it useful to (over)simplify the possible space of sequential statistical learning models into two general classes: models that predict gradual improvement over learning and models that predict a sudden transition during learning. Whether learners exhibit sudden transitions or gradual improvement during statistical learning is a fundamental question about the underlying mechanisms of statistical learning, and one for which there is relatively little data. Many studies of sequential statistical learning are not designed to investigate the rate at which people learn, and among those that do (e.g., Buchner, Steffens, & Rothkegel, 1998; Gobel, Sanchez, & Reber, 2011; Gureckis & Love, 2010; Hunt & Aslin, 2001; Jiménez, 2008; Perlman, Pothos, Edwards, & Tzelgov, 2010; Sanchez, Gobel, & Reber, 2010; Shanks, Rowland, & Ranger, 2005) the reported learning curves are typically averaged across participants, items, and/or trials. These averaging processes, while completely appropriate for the research aims of the studies, obscure the trial-level resolution of learning curves for individual items. As this is precisely the level that process models of

statistical learning make predictions about, analyzing the data at the item and trial level allows for the most informative evaluations of the models.

The prototypical example of a gradual statistical learning process is the simple recurrent network (SRN; Elman, 1990). The SRN learns through error-driven backpropagation, which results in incremental decreases to the overall error of the network over a long period of training. The SRN has been widely used to model sequential learning tasks (e.g., Boucher & Dienes, 2003; Cleeremans & McClelland, 1991; French et al., 2011; Gureckis & Love, 2010; Misyak, Christiansen, & Tomblin, 2010). Many of these applications of the SRN have focused on whether the SRN learned the appropriate structure at all, rather than on the learning curve. One criticism of the SRN as a model of sequential learning is that it often takes much more training – an order of magnitude or more – to display comparable performance to humans (Gureckis & Love, 2010). While the SRN is a powerful model in terms of the varieties of structures it can learn, its learning process may be too gradual to be an accurate model of the kinds of sequential learning that are accomplished in short experimental sessions.

An alternative gradualist model is the linear associative shift-register (LASR; Gureckis & Love, 2010). LASR uses error-driven associative learning (Rescorla & Wagner, 1972) coupled with a limited and decaying short-term memory of previous events to make predictions about the next item in a sequence. While the model is substantially less powerful than an SRN due to the single-layer design, it does a better job of modeling human performance in short sequential learning tasks (Gureckis & Love, 2010). The limited flexibility makes it substantially faster than the SRN, though learning is still gradual with steady improvement after each exposure.

TRACX, described in section 2.2.1.3, is another version of gradual learning (French et al., 2011). Unlike the SRN and LASR, TRACX does not predict the next response in the sequence; it learns to regenerate the current input in an auto-associative manner, compressing it through a hidden layer representation. TRACX is an interesting case because the learning process involves a sharp transition between when something is considered a chunk and when it is not – though this sharp boundary was relaxed in a later version of the model, TRACX 2.0 (French & Cottrell, 2014). Even though TRACX's learning mechanism involves a sudden shift, the performance of the model – measured as the reconstruction error on the output layer – follows a slow gradual curve. Like the SRN, TRACX takes orders of magnitude more trials to learn than humans (French et al., 2011).

Relatively few models of sequential statistical learning predict sudden learning. MDLChunker (Robinet et al., 2011) is the clearest case. Section 2.2.1.2 describes MDLChunker in detail. The critical feature of the model that causes it to predict sudden learning is that chunks are created in a single time step. Unlike the gradual models, the evidence acquisition process in MDLChunker does not affect the output of the model much. MDLChunker stores a perfect, but limited capacity memory store of the previous x bits of information it has seen. The minimum description length algorithm checks if any new chunks should be created on each step to reduce the bit length of the memory store. This results in the sudden creation of chunks at the point in time where the memory store contains enough instances of the new chunk.

Complicating the classification of models as gradual or sudden learners is the fact that none of these models has a clear mapping from the model representation to behavior. PARSER (Perruchet & Vinter, 1998) illustrates this point well. The learning process in

PARSER has both gradual and sudden elements (see Section 2.2.1.1 for a full description of the model). The observable learning behavior of the model could be considered gradual or sudden, depending on how the internal lexicon is mapped to behavior. PARSER involves a gradual accumulation of evidence for a chunk. Like TRACX, PARSER has a parameter that describes a threshold at which a non-chunk becomes a chunk. If the weight of the chunk alone is converted to a measure of performance, such as through the softmax procedure used to model Experiments 2.1 and 2.2, then learning would seem relatively gradual, though not quite as gradual as the various neural network models because of the random process that PARSER uses to generate candidate chunks. On the other hand, the threshold in PARSER is intended to be an actual *perceptual* threshold (Perruchet & Vinter, 1998) and another way of mapping the model to behavior is to ignore anything that is below this threshold and only allow chunks that have passed the threshold to influence behavior. This would cause the model to exhibit sudden learning as chunks crossed the threshold.

The underspecified mapping between model representations and behavior is of course not limited to PARSER. Any model that specifies the output as the weight/error/strength of some internal representational without specifying how that representation produces behavior could be interpreted in multiple ways. The standard approach that makes the fewest additional assumptions about the architecture of the model is a linear mapping between internal state and response time (e.g., Cleeremans & McClelland, 1991; Gureckis & Love, 2010; Sun, Slusarz, & Terry, 2005), but this mapping is not strictly specified by the model and other reasonable choices, such as including a threshold, could be made. Which choice is most keeping with the spirit of the

model is sometimes clear, but – as the example with PARSER illustrates – not always obvious.

A way to (temporarily) sidestep the problem of mapping model performance to behavior while still making progress on describing the mechanism of learning is to use a descriptive model to characterize the shape of the learning curve. Donner and Hardy (2015) follow this approach to characterize the shape of learning in a series of cognitive tasks. They developed a piecewise power law model, which allowed them to flexibly fit learning curve data with one or multiple piecewise power curves. The piecewise model fit data from a large set of learners (N > 25,000) substantially better than a single power law, suggesting that learners in several basic cognitive tasks exhibit sudden improvements in performance. From this evidence, Donner and Hardy draw conclusions about the mechanisms of learning at a high level, which can be further refined and unpacked by cognitive models.

In this chapter I focus on (a) characterizing the shape of the learning curve at the item-level in a sequential statistical learning task and (b) further investigating the question raised in Chapter 2 – How does learning of the surrounding sequence affect learning of the target? – using methods that will allow us to understand not just whether learning occurred after some fixed time frame, but how quickly learning progressed and when learning happened. I develop a hierarchical learning curve model that characterizes the shape of learning in a serial reaction time task. The model is flexible enough to describe both gradual and sudden learning, as well as variants in between, such as delayed-but-gradual learning. I use the model to test how previous knowledge of chunks and chunks acquired during learning affect the learning curve of a target chunk.

3.1 Experiment 3.1

In Experiment 2.1, participants learned the target triple faster when it was surrounded by other triples as opposed to random orders. In Experiment 2.2, participants learned the target triple faster when it was surrounded by known words as opposed to novel words. This experiment tests the prediction that there should be a clear ordering to the speed of learning. Learning should be fastest when surrounded by known words, followed by novel words, and finally scrambled words. As in Experiment 2.2, this experiment uses a sequence of letters to leverage people's pre-existing lexical chunks without explicitly making participants aware of the statistical structure in the sequence.

To characterize the shape of the learning curve, performance must be measured throughout the task. A common experimental paradigm for online measurement of sequence learning is the serial response time (SRT) task. In the original SRT experiments (Nissen & Bullemer, 1987), participants pressed one of four keys in response to a corresponding stimulus. The participants' response times to a repeated sequence sped up over time, while response times to random sequence sped up only marginally as participants adapted to the task. The large decrease in response time for structured sequences but not random sequences is evidence that participants learned the structure of the repeating sequence.

The SRT has become a standard task for investigating implicit sequence learning (for recent reviews, see Abrahamse, Jiménez, Verwey, & Clegg, 2010; Schwarb & Schumacher, 2012). Research on statistical learning vis-à-vis segmentation based on statistical cues (e.g., Fiser & Aslin, 2002; Saffran et al., 1996) and implicit sequence learning behavior measured in tasks like the SRT originally had different aims in the

kinds of behaviors and processes they were trying to describe but are now generally viewed as sharing common underlying mechanisms (Perruchet & Pacton, 2006). In this experiment, a variant of the SRT is used as a way to measure learning of individual items on a trial-by-trial basis.

3.1.1 Method

3.1.1.1 Participants

142 students enrolled in an introductory psychology course at Indiana University participated in partial fulfillment of a course requirement. One participant was removed from the analysis because they did not record a single correct response throughout the experiment.

3.1.1.2 Procedure

Participants completed the experiment online. Participants had total control over the computer they used for the experiment and where they completed the experiment. The experiment was developed using the jsPsych JavaScript library for browser-based experiments (de Leeuw, 2015). As the primary dependent variable in the study is response time, it is worth mentioning that several experiments have found that JavaScript's response-time measurement capabilities are comparable to Psychtoolbox and other widely used laboratory software (de Leeuw & Motz, 2016; Hilbig, 2016; Neath, Earle, Hallett, & Surprenant, 2011; Reimers & Stewart, 2014).

Upon loading the experiment website, participants completed an informed consent document. After indicating agreement to participate, participants viewed the following instructions: "In this experiment, we are measuring how quickly you can type different

characters. You will see one letter at a time, and you should press the corresponding key as quickly as you can." On the following screen, participants practiced the task with 24 trials, using a different set of letters than used in the test phase of the experiment. There was no pattern in the practice phase.

The participant's task was to respond to a displayed letter as quickly as possible. Each letter appeared in a square window in the center of the screen with a font size of 72px. The window was 300 x 300px, with a light grey border. The path of the letter was animated so that it seemingly appeared from behind the right side of the window and moved to the left at a constant rate until disappearing behind the left side of the window (See Figure 7). Letters were visible for 2 seconds. The exact display size of the letter varied across participants depending on the properties of the display that each participant used to complete the task. If the participant pressed the correct key while any part of the letter was in view, then the border of the window turned green. If the participant responded incorrectly, the border of the window turned red. The response time for each letter was measured from when the letter first started appearing on the right side of the window. This was well before the full letter was visible.

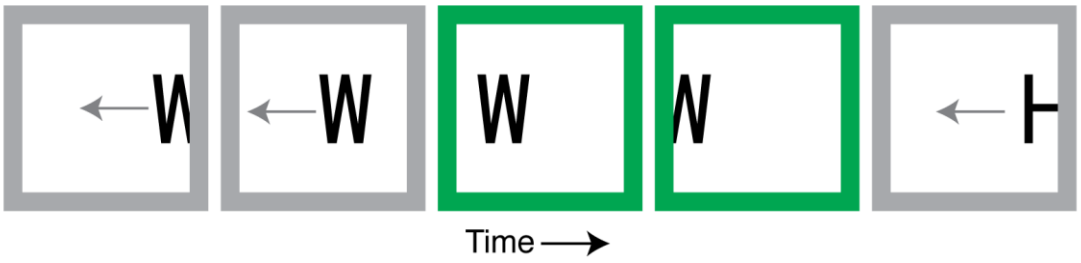

Figure 7. Illustration of Experiment 3.1 task. Participants saw a single gray square in the center of the screen. Letters appeared within the square and moved smoothly from right to left. If the participant pressed the corresponding key before the letter traveled off the screen, then the gray box turned green to indicate a correct response. The box turned red in the event of an incorrect response.

There were three different types of sequences of letters, and participants were randomly assigned to one of the three possible types. In each sequence, there were twelve different letters. The letters were selected by choosing three high-frequency English words with no overlapping letters – CUT, DAY, and HER – and a novel word that followed the same consonant-vowel-consonant pattern, NIW. In each condition, the letters N, I, and W always appeared in that exact sequence on every presentation of those three letters. The primary research aim is to understand how the responses to these three letters change as a result of varying the presentation of the remaining three words.

In the known-words condition, CUT, DAY, and HER were always shown in their normal order, i.e., C always came before U, which always came before T. In the unknown-words condition, each word was shown in the reverse order: TUC, YAD, and REH. Finally, in the scrambled-words condition, the order of letter presentation within each word was varied randomly each time the word was presented, with the exception that the letters could never appear in their proper English ordering.

Participants completed three blocks of 12 repetitions of each of the 4 words (144 letters per block). The sequence of letters within each block was created by randomly shuffling the order of the four words, and then concatenating the twelve letters to the end

of the sequence. If the concatenation would result in a repetition of a word, then a new random order was generated before concatenation. After each block, participants saw their average response time for that block and the percentage of correct responses that they made during the block.

Finally, after completing all three blocks, participants completed a structured debriefing designed to assess explicit knowledge for the patterns within the sequence. There were three questions, and participants had to respond to each question before viewing the next. The questions were: "What do you think the purpose of the experiment was?", "Did you notice any patterns in the sequence of letters?", and "Some letters occurred in the same order every time they were presented. If you noticed any of these orders, write them in the box below, with one order per line. For example, if you noticed that X was always followed by Z, you would write XZ on a line. If the pattern was longer, for example X followed by Z followed by Q, then write XZQ."

3.1.2 Results

3.1.2.1 Group Level

The first question that can be asked of these data is whether learning curves of the target triple NIW varied across contexts. Each letter appeared 36 times over the course of the experiment. Though the letters appeared in slightly different orders for different participants, the constraints of the sequence generation process ensure that the N^{th} presentation of a letter will occur at roughly the same time in the experiment for each learner. Therefore, the number of times the letter has appeared previously is a good marker for time, and will be referred to as *t*. I first found the average response time across all participants in the same condition for each of the three target letters (target triple

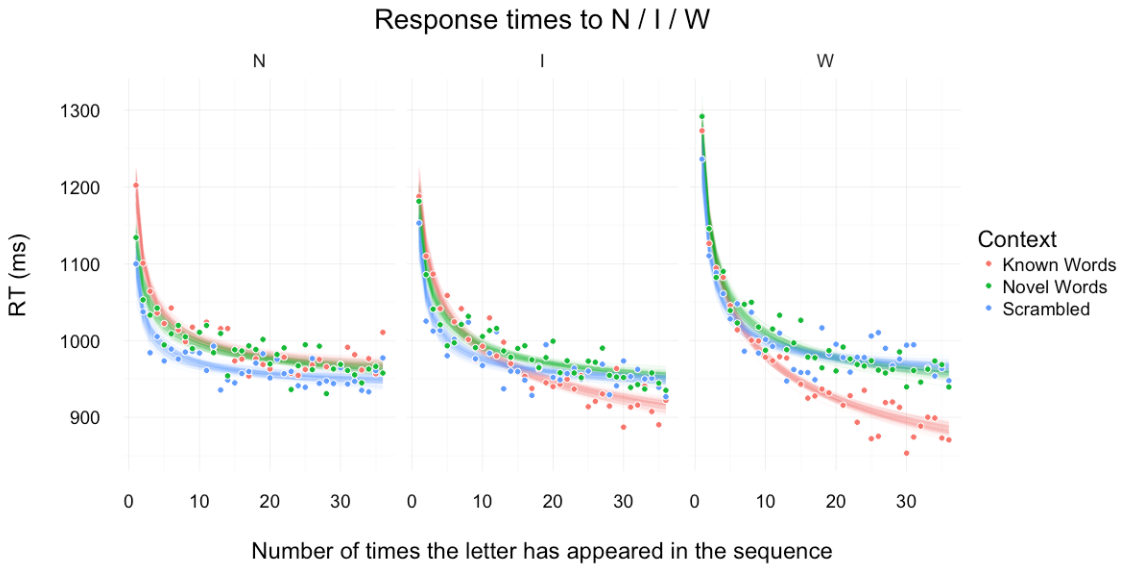

Figure 8. Group-level data and model fits for Experiment 3.1. The panels show the response times for each of the three elements in the target triple. Colors indicate the context that the target triple appeared in. Each data point is the mean response time for all participants who viewed the target in the corresponding context. The lines are posterior predictions of the group-level model. Twenty-five random samples from the posterior distribution were selected, and the corresponding power curve was plotted for each sample. The goal of this approach is to show the range of curve fits that the model views as probable descriptions of the relationship between time and response time.

position 1, position 2, and position 3) at each time *t*. I fit this data using a hierarchical model describing the response times as a function of *t* with a power curve, $\text{RT} \sim \alpha(1 + \beta[t^{-\gamma} - 1])$. The model found separate parameters for each context-letter combination. A hierarchical approach created shrinkage of the parameter estimates. Appendix B describes the full model and details of the fitting procedure.

This parameterization of the power curve means that β is interpretable as an amount of learning, scaled to the range 0-1, with 1 indicating that response times asymptote at 0ms and 0 indicating no learning relative to the starting point, α. Given sufficient time for learning, it would be reasonable to expect β to be similar in all three conditions, given that the item to be learned is identical in each condition. Once the statistical structure of the target word is learned there is no reason to expect that response

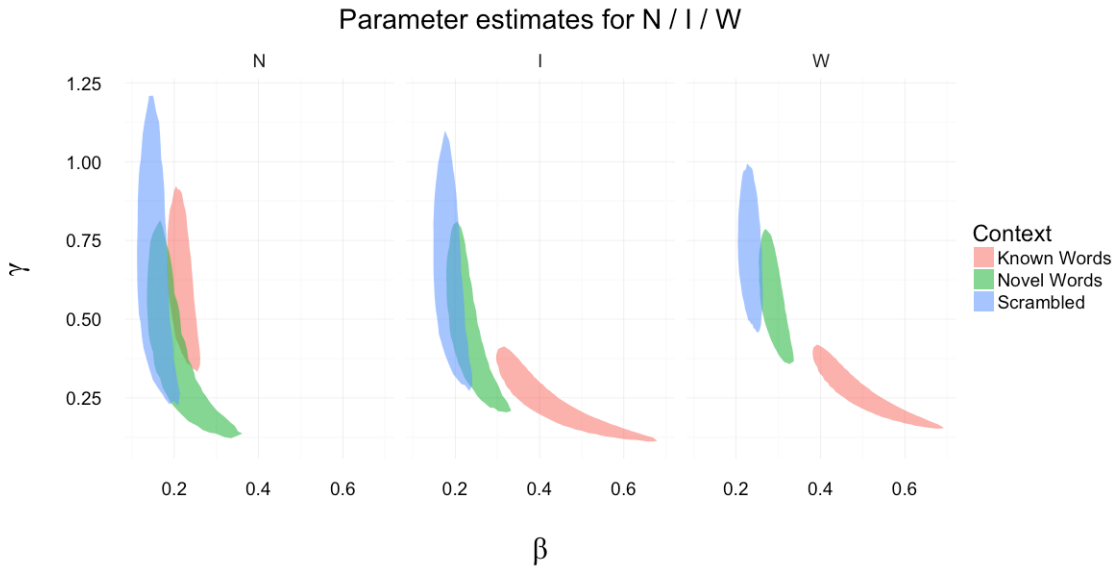

Figure 9. Group-level learning curve parameter estimates for Experiment 3.1. Each colored region is the 95% highest density region for the joint probability of $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$. HDR calculation was done using the emdbook R package (Version 1.3.9; Bolker, 2016).

times would vary based on context. The γ parameter is the steepness of the learning curve, with larger values indicating faster learning. If learning is faster in contexts with more known chunks, then it would be reasonable to expect a clear ordering of γ with the largest values in the known-words context and the smallest values in the scrambled context.

Figure 8 shows the data and model predictions. While there is clear evidence that the curves for the predictable letters – I in the second position of the target triple and W in the third position – are different in the known-words context, the difference is not consistent with the idea that learning is *faster* in the contexts where better learning is expected. Instead of faster learning towards a similar asymptote, there is slower learning towards a lower asymptote. Figure 9 shows the 95% HDIs for the joint posterior density of the β and γ parameters in each context-letter combination. There are clear correlations between these two parameters. The general pattern is that learning is faster in the

scrambled condition, but more learning occurs in the known-words condition, with the novel-words condition somewhere in-between.

3.1.2.2 Individual Level

One potential problem with interpreting the group-level fits is that the group average is the result of two kinds of learning. The first is general task *adaptation*; participants speed up as they become familiar with the task, the location of the response keys, and the kinds of symbols presented. This presence of this process is evident in the group-level model fits because response times to the first element in the target triple speed up in all conditions despite this symbol being unpredictable. In addition to adaptation, there is potentially *learning* of the statistical structure. This is also clearly present in the group-level fits. Response times to predictable elements of the target triple become reliably faster in the known-words condition. However, because these two processes are described by the same power curve, the parameters of the power curve are difficult to interpret.

The individual-level model separates *adaptation* and *learning* and fits the data on a participant-by-participant basis with (some) group-level hierarchical constraints. The model simultaneously describes the response times to the first (unpredictable) and third (predictable) element of the target triple. The power curve for the first element is assumed to describe pure adaptation. The curve for the third element is a mixture of two power curves: the adaptation curve, which is shared with the first element, and a learning power curve that has its own parameters. The learning curve can be offset from the start of the experiment, to allow for cases in which the response times to both elements of the target triple are best fit by a single adaptation curve in early trials, but then by two curves

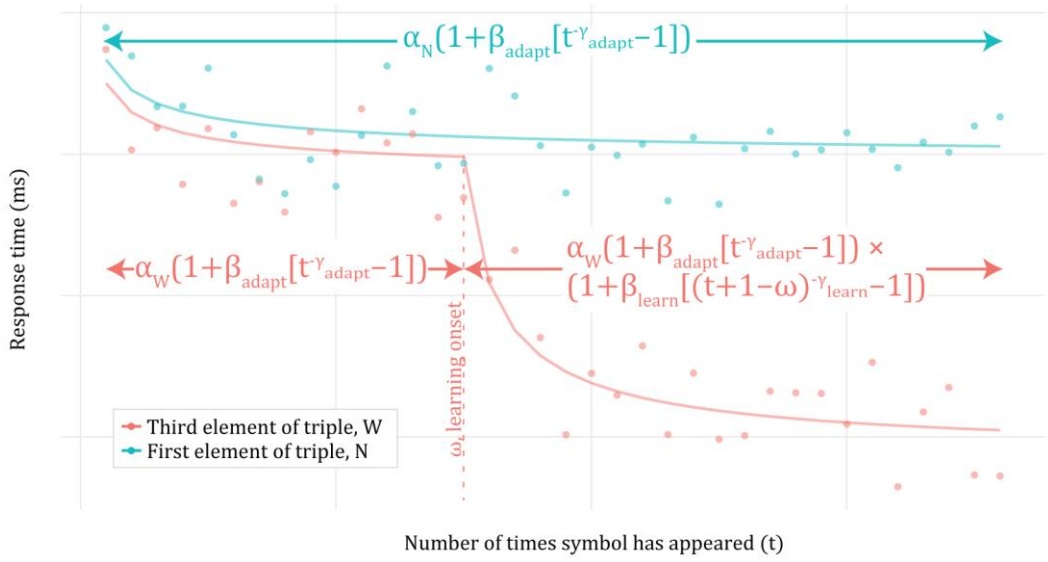

Figure 10. Individual-level model schematic. The individual model fits a curve for the response times to the unpredictable element N, shown in blue, and the predictable element W, shown in red. Both curves contain a common adaptation component described by the parameters $\boldsymbol{\beta_{adapt}}$ and $\boldsymbol{\gamma_{adapt}}$. The starting points of the curves are described separately by the $\boldsymbol{\alpha_N}$ and $\boldsymbol{\alpha_W}$ parameters. If there is additional learning of the statistical structure beyond task adaptation and the response times for W speed up at a rate different than the response times for N, then a second power curve is used to fit the data for responses to W. The onset of this curve is $\boldsymbol{\omega}$ and the shape of the curve is described by $\boldsymbol{\beta_{learn}}$ and $\boldsymbol{\gamma_{learn}}$. The data points shown are simulated.

as the participant learns the statistical structure of the sequence. A sketch of the model is shown in Figure 10. A description of the full model and details of the fitting procedure are described in Appendix C.

Before investigating how learning varied across conditions, it is important to look at how many participants showed evidence of statistical learning. Only participants who show evidence of statistical learning are useful for the comparison of learning across conditions. For each sample of the MCMC chain, the model classifies a participant as a learner or a non-learner. If a participant is classified as a non-learner, then the adaptation curve is used to fit response times for both symbols. The non-learner model is nested within the learner model – they are equivalent when β_{learn} is 0 – but the non-learner model has fewer parameters and is preferred when the additional flexibility of the learner

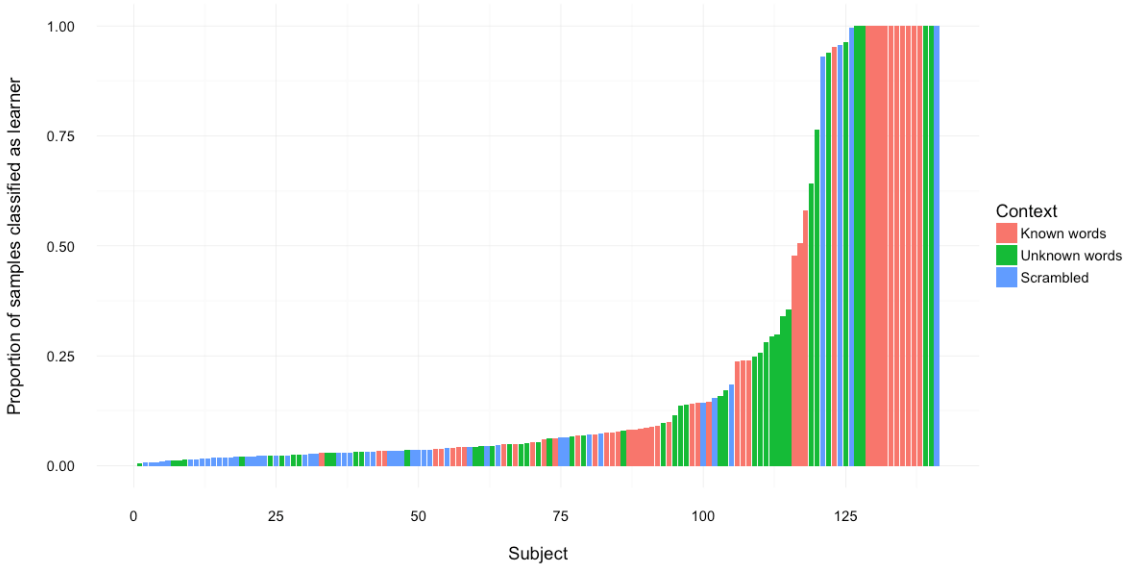

Figure 11. Proportion of MCMC samples each participant was classified as a learner. Participant are arranged in increasing probability of being classified as a learner.

model does not improve the likelihood of the data. Figure 11 shows the proportion of MCMC samples in which each participant was classified as a learner. Only 17.7% (25 of 141) of participants were classified as a learner more than 50% of the time, and only 15.6% (22 of 141) were classified as a learner at least 75% of the time. Figure 12 shows a selection of participants that were classified as a learner with low, moderate, and high probability to illustrate the differences in learning.

Among participants who did learn, the onset of learning was consistently delayed from the start of the experiment. As shown in the right-most panel of Figure 13, the onset parameter was never 0 for any participant who was reliably classified as a learner. The fastest onsets were around 5, which means that those participants saw approximately 60 total letters before they started to show any evidence of statistical learning. Most participants who learned did not show evidence of learning until about halfway through the experiment.

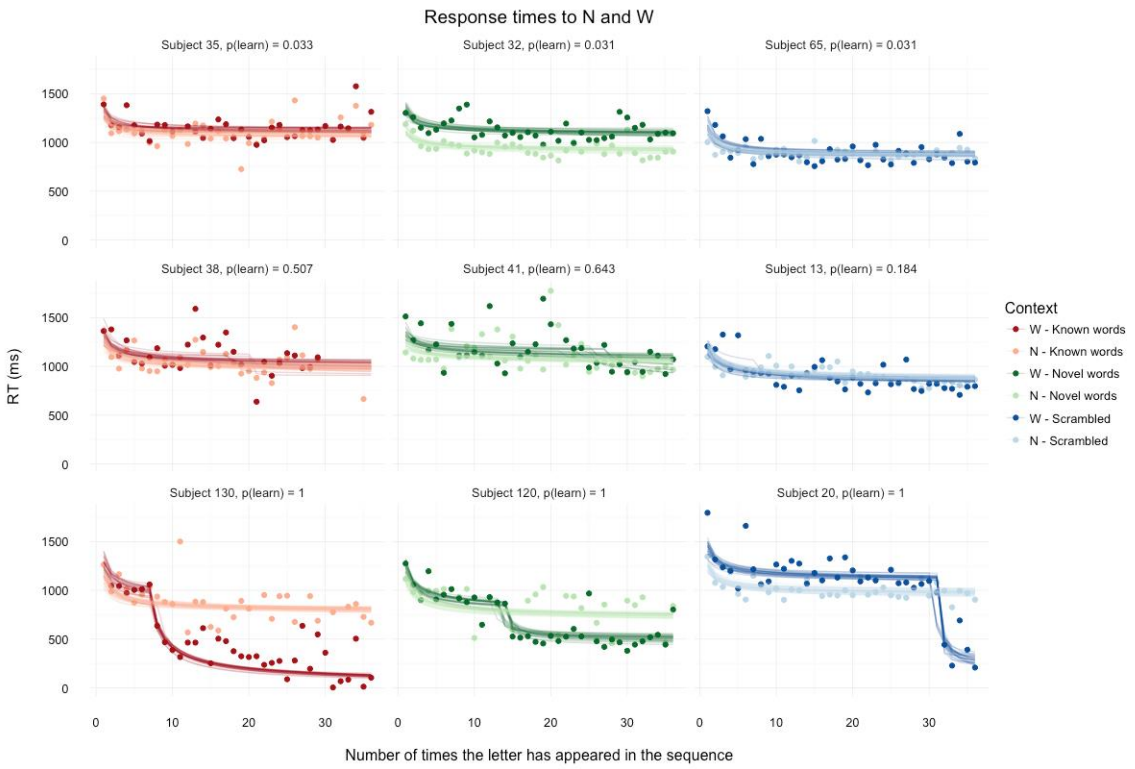

Figure 12. Participant data and model fits for Experiment 3.1. Each panel shows the response times and posterior predictions of the model for a single participant. Data and model fits for the unpredictable element N are shown in a lighter shade. The darker shade is used for the predictable element W. Model fits are generated by taking 25 random samples from the posterior distribution and using the parameters to reconstruct the curve. The variation in the fits shows the uncertainty in the posterior distribution. Participants in the top row were reliably classified as non-learners; those in the middle row were occasionally classified as learners; participants in the bottom row were always classified as learners.

With so few participants that reliably learned the target word, it is impossible to make robust comparisons of the shape of the learning curve in different contexts, which was the original aim of the experiment. Figure 13 shows the 95% HDIs for the parameters that determine the shape of the learning curve, and there are no obvious differences across contexts. Though the data are too sparse to make reliable conclusions, one possibility raised by these data is that the shape of the learning curve does not vary across context, but the onset of learning does. In other words, once a participant begins to learn the target triple the context in which they are learning it is largely irrelevant, but whether any learning occurs at all and when it occurs depends heavily on the context. In

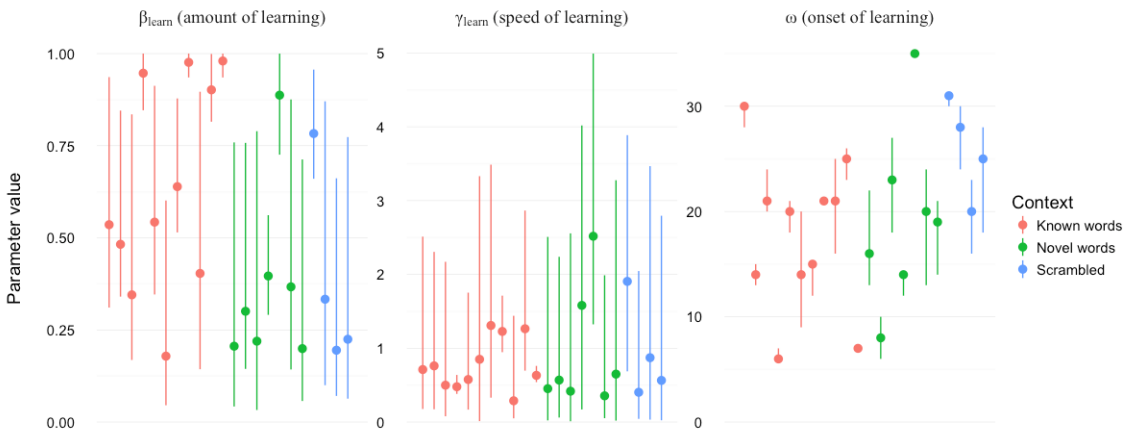

Figure 13. HDIs for learning-related individual-level parameters in Experiment 3.1. The 95% HDIs are plotted individually for each participant who was classified as a learner in at least 75% of the posterior samples. The dot shows the median of the HDI.

the individual-level model, the onset and probability of learning were modeled hierarchically with estimates at the context-level as well. Figure 14 shows the HDIs for these context-level estimates. Again, the data are too sparse to draw any robust conclusions, but the trend is that learning appears to be most likely when people already know chunks and least likely when the target is the only chunk. Similarly, learning begins earlier in a context with pre-existing chunks.

3.1.2.3 Explicit Knowledge of the Target Word

Participants' responses to the structured debriefing were coded by two people blind to the study aims. Responses were coded for correct identification of each of the possible words, taking the participant's assigned condition into account. If a participant reported a word that was not in the sequence a false positive response was coded.

Only 12 of 141 participants correctly reported the presence of the target word. I analyzed the relationship between behavioral evidence of learning and explicit report of learning by conducting a logistic regression with the individual level model's estimate of

the probability of successful learning as the predictor variable and correct report of the target triple as the predicted variable. Behavioral evidence of learning was predictive of the report of explicit knowledge, $z = 4.304$, $p = 0.0000168$. However, there were several participants who correctly reported the target word without showing behavioral evidence of learning, and there were many participants who showed behavioral evidence of learning without correctly reporting explicit knowledge. The logistic regression estimated that participants who were classified as a learner 100% of the time by the model had only a 40.17% chance of reporting explicit knowledge, and participants who were never classified as a learner by the model had only a 1.99% chance of reporting explicit knowledge.

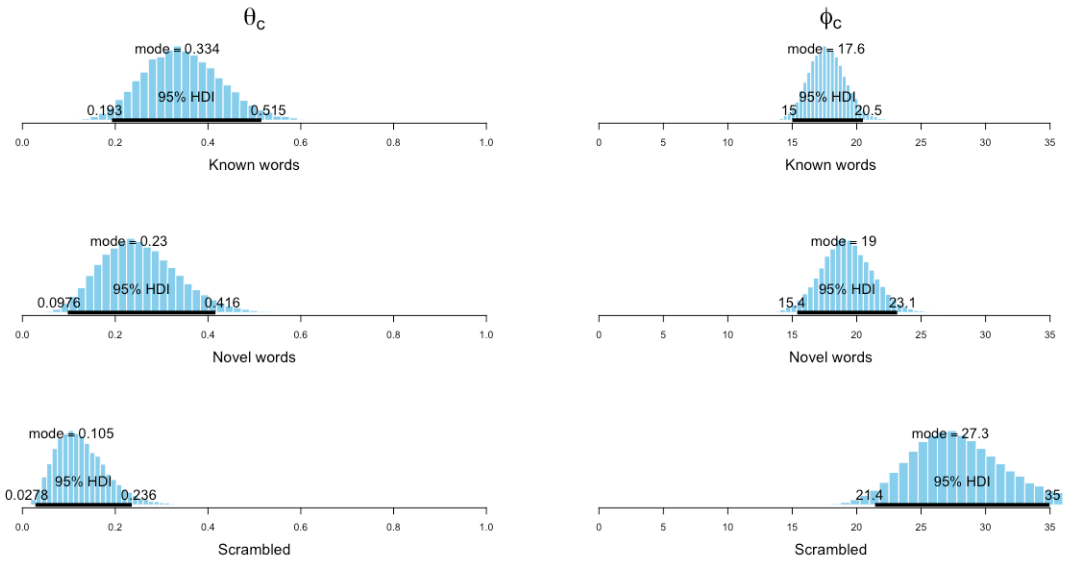

Figure 14. Posterior distribution for context-level estimates of learning probability and learning onset. The left column shows the estimates for $\boldsymbol{\theta_c}$, the probability that a participant viewing context *c* will show evidence of learning. The right column shows the estimates for $\boldsymbol{\varphi_c}$, the mode of the context-level distribution of learning onsets. HDI calculations and plotting methods are from the Doing Bayesian Data Analysis utilities library (Version 21; Kruschke, 2015).

3.2 Experiment 3.2

Learning was too sparse to make robust comparisons across learning contexts in Experiment 3.1, so this experiment was designed to increase the proportion of participants who learned. The experiment included twice as many trials and the interface was modified to show the recent history of the sequence to make sequential dependencies more noticeable.

3.2.1 Method

3.2.1.1 Participants

258 Indiana University students enrolled in an introductory psychology course participated in the experiment in partial fulfilment of a course requirement. None of the participants in this experiment participated in Experiment 3.1.

3.2.1.2 Procedure

The basic task of typing letters as quickly as possible and types of sequences were identical to Experiment 3.1. A few changes were made to try and increase the number of participants who successfully learned the target word by the end of the experiment.

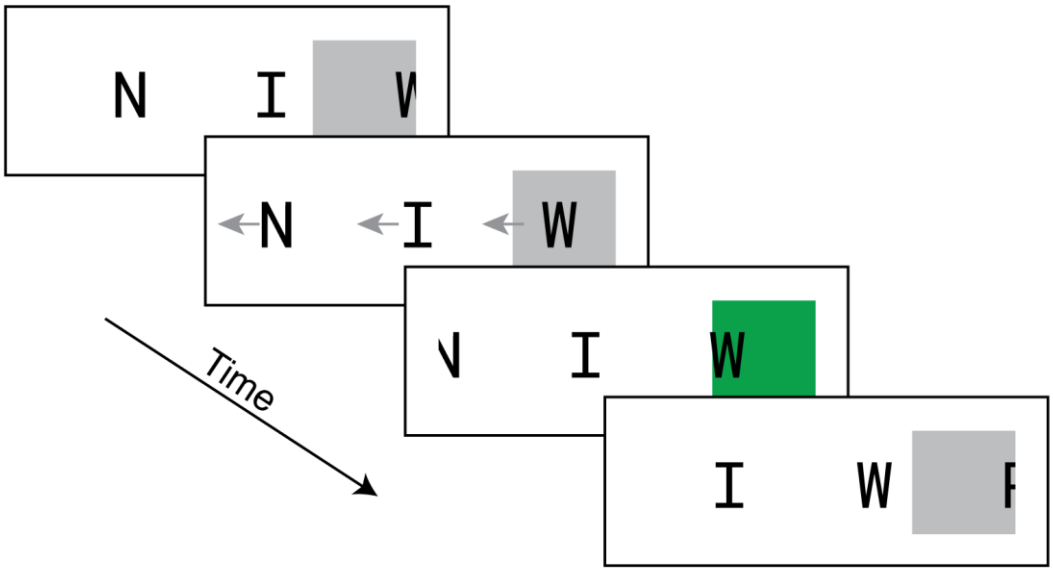

Figure 15. Interface for Experiment 3.2. Participants saw three letters on the screen at the same time, scrolling in unison from right to left. The goal was to press the corresponding key of the rightmost letter before it completely left the gray box. If the correct key was pressed in time, the box turned green. If an incorrect key was pressed, the box turned red. The rate of scrolling was not affected by the participant's response.

The interface was changed to make it easier to notice sequential dependencies. Instead of displaying a single letter at a time, three letters were visible on the screen at any given moment. Letters moved from right to left across the screen. Letters passed over a light grey box on the right third of the screen when they first appeared. The participant's goal was to press the key corresponding to the letter before it was completely outside of the box. Like Experiment 3.1, this version of the task required the participant to respond as quickly as possible to the initial presentation of the letter. The major difference for this version of the task is that the previous two letters were also visible on the screen, providing access to a limited history of the sequence with the goal of making it easier to notice the word-like structure of the sequences (Figure 15).

In Experiment 3.1 participants completed 3 blocks, viewing each letter 36 times. In this experiment, the length was doubled to 6 blocks, with 72 presentations of each individual letter.

3.2.2 Results

3.2.2.1 Group Level

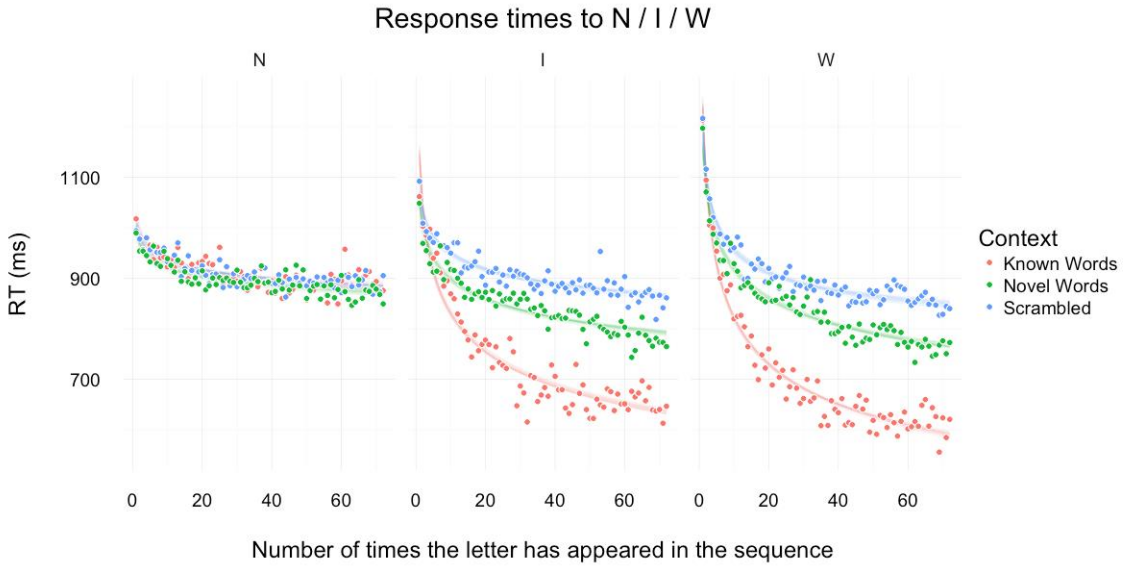

Figure 16. Group-level data and model fits for Experiment 3.2. Each data point is the mean response time for all participants who viewed the target triple in the corresponding context. The lines are posterior predictions of the group-level model, using 25 random samples from the posterior distribution.

The same group-level model used for Experiment 3.1 was used to fit the group-level data for Experiment 3.2. The only change to the model was to support values of *t* from 1 to 72 instead of 1 to 36. Details of the fitting procedure are described in Appendix B.

Figure 16 shows the group-level data and model fits. Unlike Experiment 3.1, here there is a clear separation of response times based on learning context for the predictable elements I and W. The curves for the three contexts are very similar for the unpredictable element N. While the results from the individual model in Experiment 3.2 should make

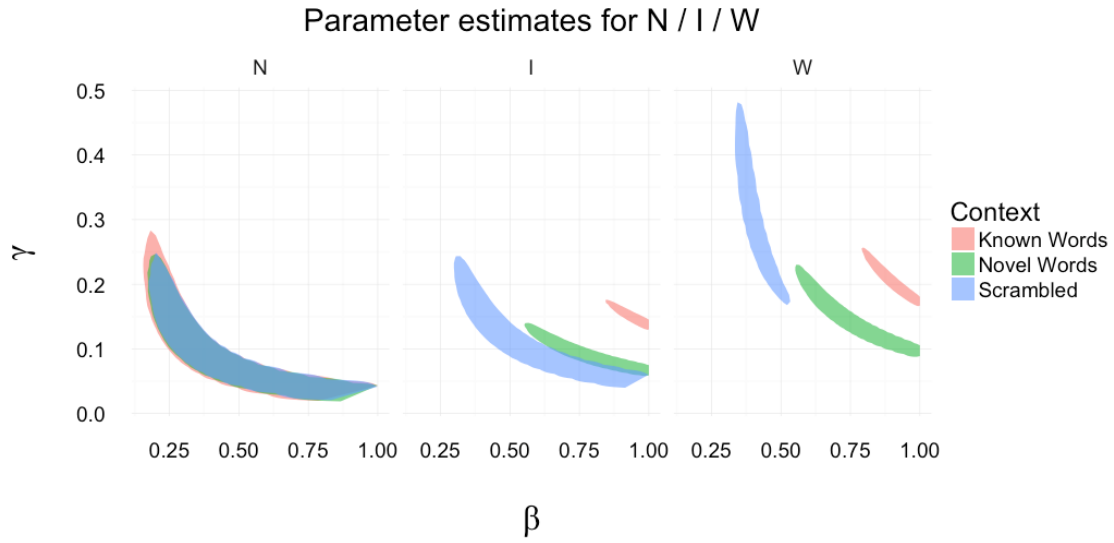

Figure 17. Group-level learning curve parameter estimates for Experiment 3.2. Each colored region is the 95% highest density region for the joint probability of $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$. HDR calculation was done using the same method as in Experiment 3.1.

us cautious about interpreting the parameters of the group-level model, there is the predicted ordering of learning, with the fastest response times in the known-words context, followed by the novel-words context, and finally the scrambled context. The parameter estimates for the rate of learning and amount of learning are again correlated (Figure 17). However, in this experiment there is a clearer separation of the parameter values in the three contexts, and also a clear equivalence of the parameter values for the unpredictable element.

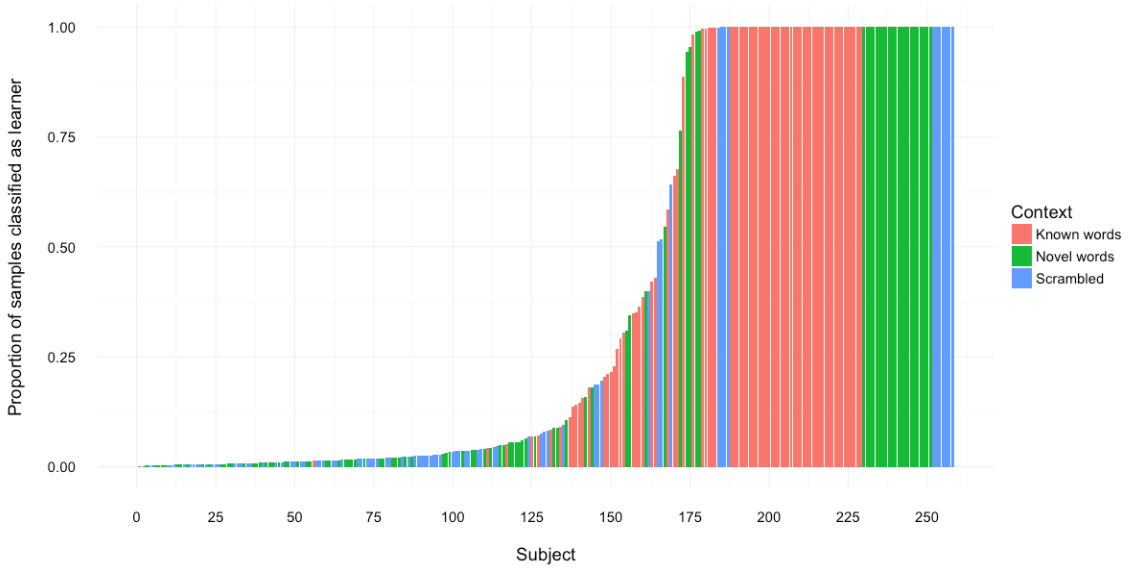

Figure 18. Proportion of times each participant is classified as a learner in Experiment 3.2. Participants are arranged in increasing probability of being classified as a learner.

3.2.2.2 Individual Level

The individual-level data were fit using the same model developed for Experiment 3.1, with two minor modifications. The first was to allow for values of *t* from 1 to 72 instead of 1 to 36. The second was to include a context-level estimate of the steepness and amount of learning, instead of just an overall group-level estimate. This modification makes it possible to determine if the amount or rate of learning varies by learning context.

Even though the experiment was altered to improve the number of participants who learned, learning remained relatively uncommon. As shown in Figure 18, only 94 of the 258 participants (36.4%) were classified as a learner in the model more than 50% of the time. The percentage drops slightly to 33.7% when restricting to participants who were classified as a learner at least 75% of the time. Figure 19 shows a sample of participants who were reliably classified as non-learners (top row), reliably classified as

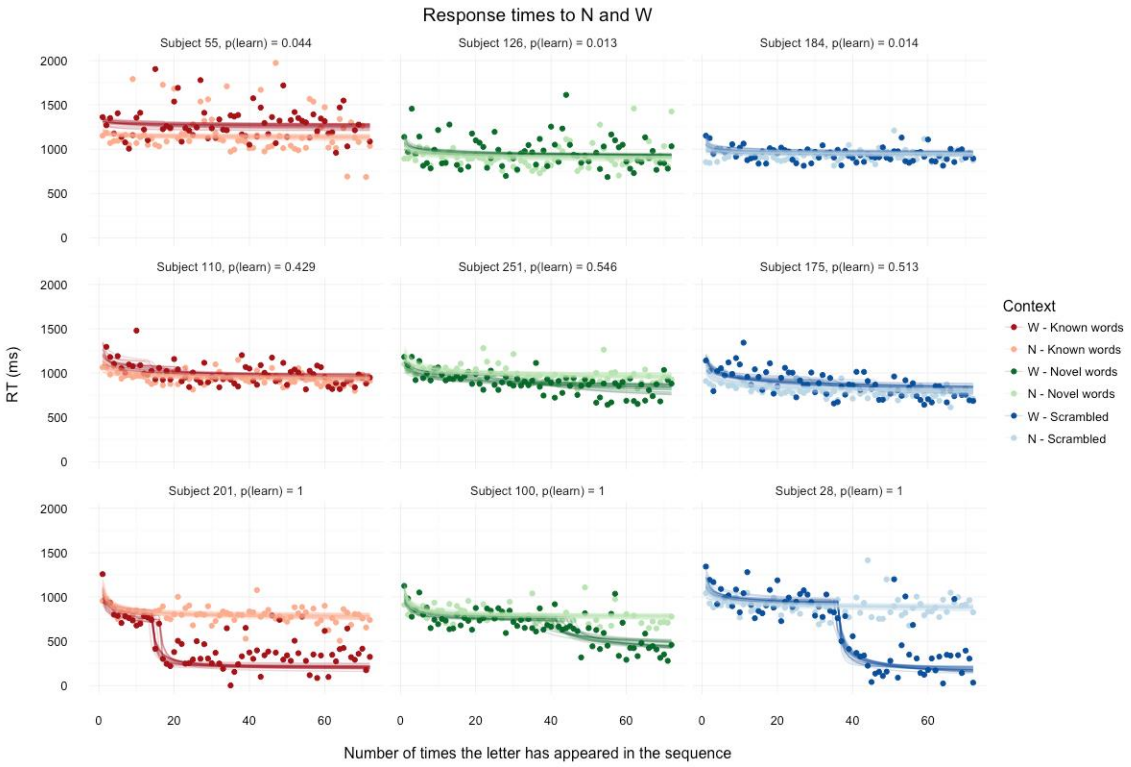

Figure 19. Sample data and posterior predictions from Experiment 3.2. Each panel shows the response times and posterior predictions of the model for a single participant. Data and model fits for the unpredictable element N are shown in a lighter shade. The darker shade is used for the predictable element W. Model fits are generated by taking 25 random samples from the posterior distribution and using the parameters to reconstruct the curve. The variation in the fits shows the uncertainty in the posterior distribution. Participants in the top row were reliably classified as non-learners; those in the middle row were occasionally classified as learners; participants in the bottom row were always classified as learners.

learners (bottom row), or roughly equally classified as learners and non-learners (middle row). The pattern of responses by participants in the middle row is much more consistent with the non-learners, and so the remainder of the analysis looks only at participants who were classified as learners at least 75% of the time. While a majority of participants still failed to learn, the sheer increase in the number of participants combined with an improvement in the proportion of participants who did learn means that there is a large enough sample of participants who learned to make comparisons across different contexts.

Learning, when it occurred, was still consistently delayed from the start of the experiment. As shown in the rightmost panel of Figure 20, the onset of learning was after 0 for all but one of the participants. (This participant had a particularly strange pattern of responses that are possibly the result of data corruption). As in Experiment 3.1, the individual-level estimates for the steepness and amount of learning parameters appear consistent across different learning contexts (left and center panels of Figure 20). The 95% HDIs for the steepness and amount of learning context-level estimates all had substantial overlap, suggesting that there was no robust difference in these parameters (Table 1).

The model estimates that participants in the known-words context are more than twice as likely to learn than participants in the novel-words or scrambled contexts. The modal estimate for the probability of learning for participants in the known-words context is 0.717 (95% HDI: 0.579 to 0.845). For participants in the novel-words context, the mode is 0.304 (95% HDI: 0.207 to 0.431); in the scrambled context the mode is 0.179 (95% HDI: 0.098 to 0.285). Furthermore, the model finds that the learning onsets are

Table 1. HDIs for context-level estimates in Experiment 3.2.

		95% HDI	
Parameter	Context	Low	High
β_{learn_g}	Known words	0.209	0.954
	Novel words	0.352	0.983
	Scrambled	0.034	0.716
γ_{learn_g}	Known words	0.0000088	0.535
	Novel words	0.0000007	0.364
	Scrambled	0.00016	1.34

robustly different between contexts, with learning starting earliest in the known-words context, followed by the novel-words and finally scrambled contexts (Figure 21).

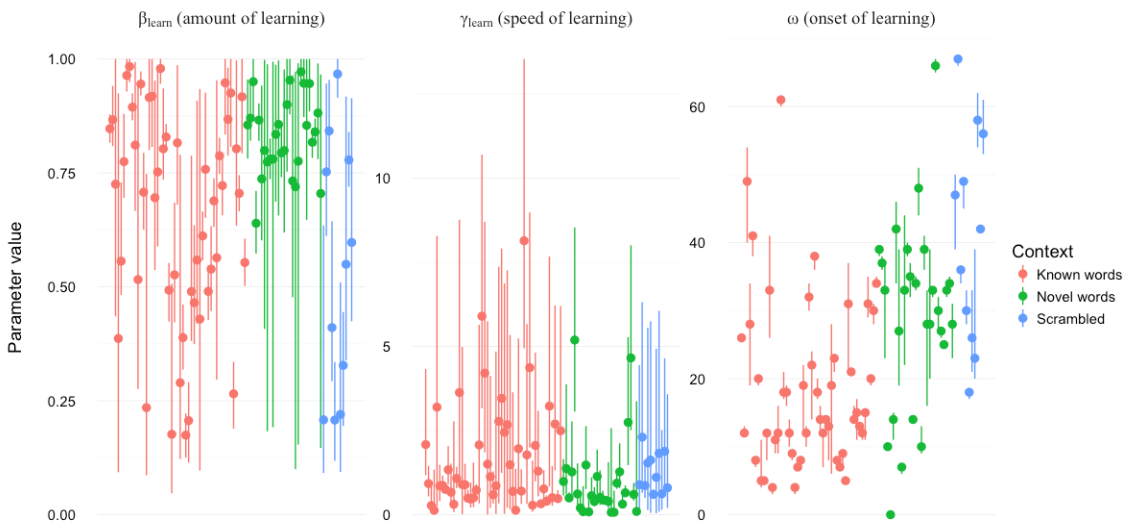

Figure 20. Participant-level estimates of learning curve parameters for Experiment 3.2. The 95% HDIs are plotted individually for each participant who was classified as a learner in at least 75% of the posterior samples. The dot shows the median of the HDI.

3.2.2.3 Explicit Knowledge of the Target Word

I used the same procedure as in Experiment 3.1 for analyzing the explicit report data. 71 out of 258 participants correctly reported the target word. A logistic regression found that the model's classification of learning behavior predicted the correct report of explicit knowledge, $z = 8.58$, $p < 0.0001$. Like in Experiment 3.1, there were a handful of participants who reported the target word but showed no behavioral evidence of learning, and there were many participants who showed strong evidence of learning without reporting the target word. However, the relationship between behavioral and explicit outcomes was stronger in this experiment. The logistic regression predicted that a participant who was classified as a learner 100% of the time had a 73.8% chance of

reporting explicit knowledge, while a participant who was never classified as a learner had only a 2.07% chance of reporting explicit knowledge.

3.3 Discussion

In this chapter, I described two experiments designed to test the hypothesis that knowledge of chunks in a sequence makes it more likely that other chunks in the sequence will be learned. I found evidence that knowing or learning part of a sequence aided in learning other parts of the sequence, consistent with the results found in Chapter 2. This provides further support for models of statistical learning that posit a dynamic interaction between compressed memory and learning. Furthermore, the results indicate that statistical learning, at least in the tasks used for these experiments, results in an abrupt transition point in behavior that occur after a period in which there is no behavioral evidence of learning. This constrains the set of models that can account for statistical learning to models that can predict this transition in behavior.

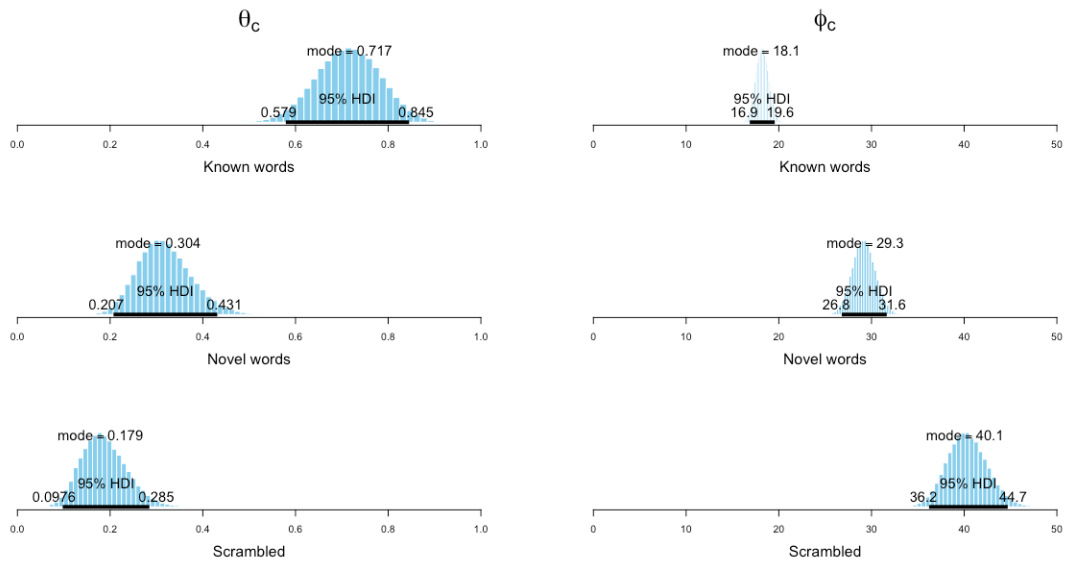

Figure 21. Group-level estimates of learning probability and learning onset for Experiment 3.2. The left column shows the estimates for $\boldsymbol{\theta_c}$, the probability that a participant viewing context *c* will show evidence of learning. The right column shows the estimates for $\boldsymbol{\varphi_c}$, the mode of the context-level distribution of learning onsets. HDI calculations and plotting methods are from the Doing Bayesian Data Analysis utilities library (Version 21; Kruschke, 2015).

3.3.1 Learning Curve Shape

Modeling the raw response time data at the individual level, as opposed to the often-used group level analysis, led to insights about the nature of the learning process. Two features of the learning curves stood out.

First, learning itself was relatively rare. Only about 15% of participants showed consistent evidence of learning in Experiment 3.1. That number rose to only about 34% after modifications to the task to emphasize the sequential relationships in the data. There are two broad possible explanations for the low learning rate. First is that participants in the experiment were not motivated to engage or attend to the task. Without attention – broadly construed – there is no (or at least greatly diminished) learning (Shanks et al., 2005; Toro et al., 2005; Turk-Browne et al., 2005). This is certainly possible, though two pieces of evidence argue against it. Participants who did not learn still produced

appropriate response times, showed task adaptation effects, and missed few trials, suggesting that they were actively watching and interacting with the sequence. The probability of successful learning also differed by condition. By the model's estimate, participants in the known words condition had approximately a 71% chance of learning the target word before the end of Experiment 3.2, while participants in the scrambled condition had only a 18% chance. Together, these pieces of evidence suggest that a second explanation for low learning rates is correct: statistical learning of even modest complexity is difficult, in the sense that it is not guaranteed to happen despite a relatively deterministic statistical structure and repeated exposure.

The second prominent feature of the learning curves was that the behavioral expression of learning was delayed and sudden. Response times for learned items often decreased from about 1,000 milliseconds to under 500 milliseconds for the target letter over the course of one or two presentations, but only after a lengthy period of exposure during which response times were steady. Different sequential contexts had robust effects on the onset of learning. Sudden learning events happened earliest in the experiment when participants already had a promising suggestion as to how to segment the rest of the sequence because it consisted of previously known words. However, there was no evidence that sequence context affected the shape of the learning curve beyond the onset. Participants' response times underwent a rapid shift and then plateaued almost immediately in all sequence contexts. This pattern is similar to the findings of Gallistel et al. (2004), who reported that variability in several associative learning tasks was best explained by changes in learning onset and not in the slope or amount of learning.

3.3.2 Explicit Knowledge

In the original test of the serial response time task, Nissen and Bullemer (1987) found that nearly all of the participants reported noticing the sequential structure when asked at the end of the experiment. Since then there has been substantial effort to understand the contributions of explicit and implicit knowledge to performance on the SRT and other forms of statistical learning (Destrebecqz & Cleeremans, 2001; Dienes, Broadbent, & Berry, 1991; Perruchet & Pacteau, 1990; Rüsseler & Rösler, 2000). Understanding the role of explicit and implicit knowledge in statistical learning tasks is important for model development. Models are divided over whether learning is distributed in a connectionist fashion, consistent with the idea that learning is implicit, based on building up explicit knowledge of the task, or the result of interactions between explicit and implicit knowledge (Cleeremans & Dienes, 2008; Perruchet & Vinter, 1998; Sun et al., 2005).

Establishing that learning *is* implicit requires showing that learning is *not* explicit, which is challenging because it necessitates evidence of the absence of an effect. In a null-hypothesis testing framework, this means accepting the null – which is, of course, verboten. Vadillo, Konstantinidis, and Shanks (2015) conducted a systematic review of the contextual cuing literature, which is a common task for studying implicit visuospatial learning (Goujon, Didierjean, & Thorpe, 2015). They found that researchers frequently interpreted non-significant null hypothesis tests for critical tests of explicit awareness as evidence for implicit learning. If the statistical power of those tests were very high, then these tests would at least rule out the possibility of large effects of explicit knowledge. However, high-powered tests of explicit knowledge can be difficult to conduct because

knowledge may be limited to a few items in the set used for testing – enough to demonstrate a behavioral signature via response time measurements when averaging across items, but too few to show robust awareness in a less sensitive forced choice test. When Vadillo et al. ran a meta-analysis on the studies that reported sufficient information to be included in the analysis, they found strong evidence of explicit awareness.

The individual level model in this chapter could allow for more robust and statistically powerful tests of explicit knowledge. The crucial test for implicit learning is whether someone demonstrates explicit knowledge *given that* they demonstrate learning. Typical measures of explicit knowledge test participants on all the patterns they may have learned. Testing participants on patterns that they did not show any evidence of learning just adds noise to the measure and reduces statistical power. By measuring learning for an individual item at the participant level, as the model developed in this chapter does, items that participants did not learn can be excluded from an analysis of explicit learning.

These experiments showed a significant relationship between behavioral expressions of learning and explicit report of learning. Many participants had explicit knowledge of the sequential structure. However, many also showed strong evidence of learning without reporting explicit knowledge. These participants may have learned without awareness of the pattern, or they may have acquired explicit knowledge during the task, but forgotten the exact letter mappings by the time that their explicit knowledge was tested (see Dienes & Berry, 1997 for a discussion of this issue). Indeed, a handful of participants reported sequences like `NEW` instead of the correct `NIW`. Others reported recognizing some of the other words in the sequence but not the target word, which

would indicate that there was explicit learning happening but perhaps not explicit learning (or recall) of the target item. While it is possible that some participants showed strong learning effects without any explicit knowledge, that does not describe the general pattern observed in the experiment.

3.3.3 Implications for Models and Mechanisms

Are there models that predict the shape of learning observed in this task? Clearly, the gradual associative models (e.g., SRN, LASR, and TRACX) do not. Learning was too sudden to be accounted for by incremental changes in associative weights without some kind of threshold mechanism. MDLChunker also cannot account for the pattern of learning. While MDLChunker does predict sudden learning, it has a difficult time with these results because it does not have a built-in way to predict the low success rate of learning across participants nor the effect of pre-existing knowledge (see results from section 2.4).

PARSER may be able to account for the results with the right set of assumptions about the link between the internal lexicon and behavior. PARSER does predict delayed learning. I ran several instances of a simulation with the default parameters for PARSER using a sequence similar to the novel words condition and tracked learning over time for each of the target triples. Figure 22 shows the results from a representative simulation. During the first phase of learning, a word seen sporadically experiences small jumps in weight, but remains below the shaping threshold. Eventually the random chunking process may select the true word frequently enough to boost it above the shaping threshold. After the word is above the shaping threshold, learning proceeds steadily as the model becomes far more likely to perceive the word when it encounters it. With the

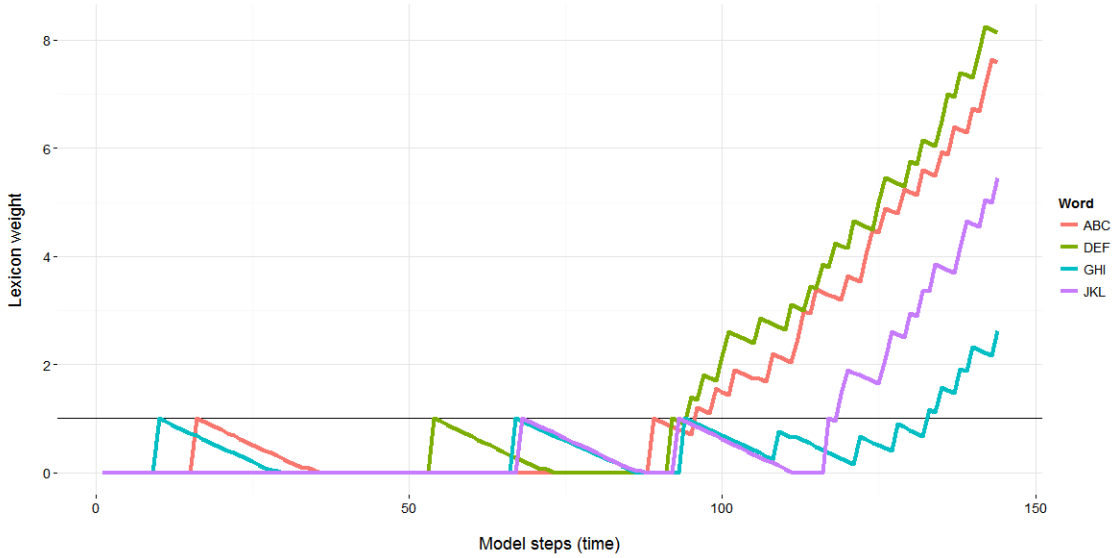

Figure 22. Time course of learning in PARSER. The model was run on a sequence containing 36 instances of four unfamiliar triples. The plot shows the lexicon weight for each of the four triples over time. The solid black line shows the shaping threshold, set to 1 in this simulation. The pattern of learning here (delayed onset and gradual accumulation of weight after crossing the threshold) was consistently observed across many runs of the model.

assumption that weights below the shaping threshold do not affect response time behavior and a weight-to-response-time transformation that mapped small initial changes above the shaping threshold to large changes in response time, PARSER could generate the kinds of learning curves observed in these experiments. PARSER is representative of a potentially larger class of models that probabilistically sample from the sequence and accumulate evidence towards a threshold. Chapter 4 explores this idea further.

The delayed onset of learning raises questions about what process is occurring during the first phase of the experiment before there is evidence of learning. Even though there is no behavioral indication of learning during this phase, something related to learning must be occurring because learning onsets vary in different sequential contexts. PARSER proposes that people are engaged in a form of stochastic encoding, sampling different combinations of adjacent units until some groupings are encountered frequently

enough to stick (Perruchet & Vinter, 1998). Other accounts could include forms of hypothesis testing, with learners explicitly (Sun, Zhang, Slusarz, & Mathews, 2007) or implicitly (Roser, Fiser, Aslin, & Gazzaniga, 2011) generating and (dis)confirming patterns, covert changes in attentional patterns to different elements (Umemoto et al., 2010), or sub-threshold evidence accumulation (Schuck, Gaschler, & Frensch, 2012). The latter is a particularly interesting proposal because it offers a way to integrate sudden changes in behavior with the gradual associative mechanisms that most accounts of statistical learning require (Yurovsky, Fricker, Yu, & Smith, 2014).

The lack of behavioral evidence during this initial period of learning makes distinguishing between these accounts challenging. Gaining further insight into the mechanism(s) will require either clever experimental designs that can distinguish different accounts based on when and/or what people start to learn, or observing non-behavioral markers of learning through neuroimaging techniques (see Reber, 2013; Schapiro & Turk-Browne, 2015 for reviews of neuroimaging research on implicit/statistical learning). The latter approach was taken by Turk-Browne et al. (2009), who measured BOLD responses with fMRI while participants viewed structured and random sequences. The participants completed a cover task during exposure as well as a surprise familiarity-judgment task at the end to test for explicit knowledge about the statistical structure of the structured sequence. Despite the group performing at chance level on the familiarity task, there were reliable differences in the BOLD signal response to structured and random sequences over time. There were no regions that showed reliable differences in BOLD response during the first block of exposure. However, by the second block – at which point the participant only had four exposures to each shape –

some regions showed a greater response to structured sequences. Regions linked to associative learning generally and regions that were selective for the particular stimuli used in the sequence showed greater activation to the structured sequences than the random sequences. This result indicates that there are robust neurological changes occurring early in statistical learning before there is reliable behavioral evidence of the learning. What remains unclear is what computational processes best describe these changes and the process by which these changes eventually cause changes in behavior.

4 A Model of Dependency in Statistical Learning

The results from Chapters 2 and 3 indicate that (1) the expression of learning through behavior seems to be sudden, (2) different contexts (e.g., whether surrounding items are words or non-words) change when items are learned, but not necessarily the shape of the learning curve other than the onset of learning, and (3) some models capture these contextual effects on learning as an emergent phenomenon of chunking. However, chunking and its influence on memory is just one possible mechanism for creating dependency between different items being learned. In this chapter, I argue that it would be premature to suggest that interactions of chunking and memory are *the* cause. Instead, I propose a more abstract model that can account for the results from all of the reported experiments through a generic dependency process, i.e., learning one thing helps you learn another.

PARSER's chunking mechanism creates a particular kind of dependency in the learning process. Once part of the sequence crosses the shaping threshold, memory resources are (indirectly) freed and the learning of other items is accelerated. This is one possible account of the empirical data, but many other plausible accounts exist as well. One possibility is that learning a chunk of a particular size causes a bias towards other chunks of the same size (Lew-Williams & Saffran, 2012). Another is that attention may be bolstered by noticing that there is reliable structure (Kidd et al., 2012; Kidd, Piantadosi, & Aslin, 2014; Zhao et al., 2013), or may be attracted to relevant features as units are learned (Zhao et al., 2011). For this particular set of experiments, the relevant feature would be sequential transitions, or alternatively just greater overall attention to the task, but in more complex scenarios attention could be guided towards particular dimensions, or spatial locations (Zhao et al., 2013). A fourth possibility is that learning

to segment part of the sequence creates anchors that can be used to guide further segmentation (Cunillera, Càmara, Laine, & Rodríguez-Fornells, 2010; Dahan & Brent, 1999; Kurumada, Meylan, & Frank, 2013). For example, Perruchet and Tillmann (2010) found when some of the words in a novel language were naturally more word-like – a property that they measured in an independent study – the other words in that language were more likely to be segmented correctly.

Further support for this last idea comes from models that posit people are trying to construct a parsimonious generative model of the data (Orbán et al., 2008; Pearl, Goldwater, & Steyvers, 2010). In this account, people explore different combinations of chunks (hypotheses) that potentially generate patterns they are viewing. The set of possible hypotheses is greatly reduced once a single chunk is discovered, improving the efficiency of the search in the remainder of the space. This is a plausible explanation given the evidence that people prefer to stick with an initial interpretation of the structure over searching for other (possibly better) structures (Gebhart et al., 2009).

Given these many possible routes to dependence, a productive way to model their collective effect on the learning process is to give a general descriptive account of how learning a chunk is dependent on learning other chunks. In the model developed in this chapter, I start with the assumption that a generic evidence accumulation process towards a threshold can explain the pre-behavior-transition phase of learning. I focus on three basic aspects of the learning process: (1) How quickly is evidence acquired? (2) How noisy is the evidence accumulation process? And (3) Does learning one item speed up the evidence accumulation process of other items? By parameterizing the model in a way that is readily interpretable across many different kinds of statistical learning, the parameter

fits can be used to infer basic features of the learning process. For example, are there cases where dependence is stronger or weaker? Is learning on each exposure consistent, or noisy? The model I introduce below provides insight on these questions.

4.1 The PANDA model

In this section, I lay out the novel Parallel and Dependent Accumulators (PANDA) model. The goal of the model is to describe the time of a set of transition points between two behavioral modes (e.g., non-learning and learning). While the model can be applied to situations in which the participant only learns a single item, the strength of the model is the ability to describe the transition points for many items that the participant is learning in parallel, as well as the relationship between those points.

PANDA is based on the framework of sequential sampling models. The theoretical underpinning of all of these models is that evidence – loosely construed – is sampled from a stimulus until a threshold is reached and a behavior occurs. The time it takes to reach the threshold depends on how much evidence is needed and how quickly the evidence can be sampled. Perhaps the most well-known class of sequential sampling models is the diffusion model of decision making (see Forstmann, Ratcliff, & Wagenmakers, 2016 for a recent review), but many variations of the general idea have been proposed over the last 60 years (e.g., Brown & Heathcote, 2008; Stone, 1960; Van Zandt, Colonius, & Proctor, 2000; Vickers & Lee, 1998; Zandbelt, Purcell, Palmeri, Logan, & Schall, 2014).

The PANDA model is a variant of an accumulator model (e.g., Smith & Vickers, 1988). Historically, accumulator models were developed to account for response time behavior in discrimination tasks that involved a choice between two different options.

The earliest versions described a process of sampling from a stimulus at regular intervals, with a random amount of evidence acquired for each accumulator until one reached a threshold and the decision was made (Vickers, 1970). Variants of the core model have extended this framework to account for trial-to-trial learning effects (e.g., Ludwig, Farrell, Ellis, Hardwicke, & Gilchrist, 2012) and competitive interaction of accumulators before a boundary is reached (Usher & McClelland, 2001). In the standard application of accumulator models, each accumulator represents a mutually exclusive choice and the first accumulator to finish indicates which choice was made and when the choice was made. In the PANDA model, each item to be learned is represented by an accumulator and all the accumulators will eventually finish. The model predicts when learning will occur for each item. Learning effect are incorporated into the model by allowing the accumulation rate to change based on the number of accumulators that have finished.

PANDA has *n* accumulators; the experimental design determines *n*. The model operates in discrete time. On every time step, each accumulator adds a random amount of evidence by sampling from a binomial distribution. When the amount of evidence for a particular accumulator crosses a threshold, that accumulator is finished and the time of the finish is the output of the model. Two parameters control the rate and noise of the

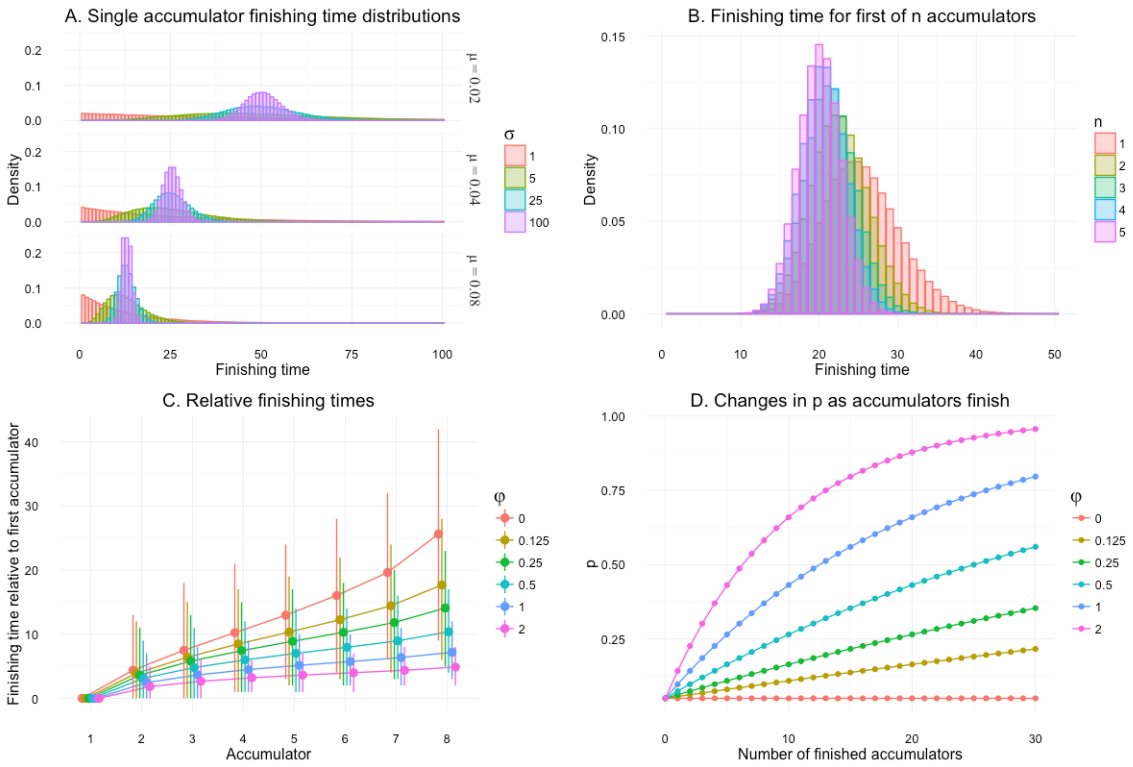

Figure 23. Effect of PANDA model parameters. Panel A: The distribution of finishing times for a single accumulator with different values of accumulation rate $\boldsymbol{\mu}$ and accumulation noise $\boldsymbol{\sigma}$. Panel B: The distribution of finishing times for the first of *n* accumulators. The value of $\boldsymbol{\mu}$ was fixed at 0.04, and $\boldsymbol{\sigma}$ was 25. Panel C: The distribution of finishing times of 8 accumulators relative to the finishing time of the first accumulator as the dependency parameter $\boldsymbol{\varphi}$ changes. The point is the mean finishing time and the error bars show the range of the 95% most common outcomes. The accumulators are ordered on the x-axis by finishing time. Panel D: The value of *p* changes as a function of the number of accumulators that have finished and the dependency parameter $\boldsymbol{\varphi}$. The base accumulation rate $\boldsymbol{\mu}$ for this plot is 0.05.

accumulation process. The μ parameter controls the rate of accumulation, and the σ parameter controls the noise of the accumulation process. On each model step, the gain in evidence for an individual accumulator is a random draw from a binomial distribution with probability of success p and number of attempts σ. (p is closely related to μ in a way that will be described shortly). Once an accumulator's total evidence reaches the threshold σ, the accumulator is finished. Using the same parameter for the threshold and number of attempts in the binomial draw guarantees that the model has a non-zero probability of finishing at any particular time step. Higher values of μ lead to faster

finishing times. Higher values of σ result in less variability in finishing times, because as σ becomes very large the probability that the proportion of evidence accumulated (successes in the binomial draw) on each step is closer to p increases. For example, with only a single attempt, $\sigma = 1$, the proportion of successes is either 0 or 1. If p is 0.5, the proportion of successes on any given trial is quite far away from p. However, if there are many attempts, e.g., $\sigma = 100$, then the proportion of successes will converge towards p. Figure 23 shows the effect of changes in the model parameters.

The number of parallel accumulators, *n*, also affects the finishing time of the model because for a fixed value of μ the probability of at least one accumulator finishing in the next time step increases with the number of accumulators (Figure 23, Panel B). This seems like inappropriate model behavior as it predicts that someone trying to learn millions of items simultaneously would learn one of those items faster than someone just studying a single item would. However, *n* is not intended to be a free parameter of the model and is determined by the particular experimental context. In practice, it is reasonable to expect that μ would be lower in a context where there are millions of accumulators than in a context with a single accumulator.

The key parameter of the model that makes it a useful tool for investigating the types of contextual dependencies I am describing in this dissertation is the dependency parameter φ. While μ controls the baseline probability of success for evidence accumulation, the actual value of p used for each accumulator is:

$$p = 1 - (1 - \mu)^{1+\varphi f}$$

In the above equation, *f* is the number of accumulators that have already reached threshold. When φ is greater than 0, p increases with every accumulator that finishes.

Larger values of φ cause smaller gaps between finishing times with less variability (Figure 23, Panel C). φ must be a non-negative number, or else p can drop below 0.

```
function PANDA(n, mu, sigma, phi)
       accumulators ← new array of size n with values = 0
       finishTimes ← new array of size n with values = 0
       finished ← 0
       time ← 0
       while(finished < n)
              time++
              p ← 1 – (1 – mu) ^ (1 + phi x finished)
              for(i ← 0; i < n; i++)
                     accumulators[i] += random sample from binomial(p, sigma)
                     if(finishTimes[i] == 0 && accumulators[i] ≥ sigma)
                            finishTimes[i] ← time
                            finished++
       return sort(finishTimes)
```

Figure 24. PANDA model pseudocode. Implementations of the model in R and C++ via the Rcpp R package (Eddelbuettel & Fran, 2011) are available in the OSF repository.

4.2 Fitting of Experiment Results

The PANDA model can be used to test the hypothesis that dependence in the learning rate between different items is sufficient to account for the results of Experiments 2.1, 2.2, and 3.2. (I will not fit the results from Experiment 3.1 given the conceptual similarity to Experiment 3.2 and the better overall learning rates in 3.2). In this section, I show that the model can produce very close matches to the empirical data in all of these experiments, but only when the dependency parameter φ is well above 0.

4.2.1 Experiment 2.1

Experiment 2.1 used a learn-then-test procedure. Participants were exposed to 25 instances of each triple in a continuous sequence, and then completed a series of two-

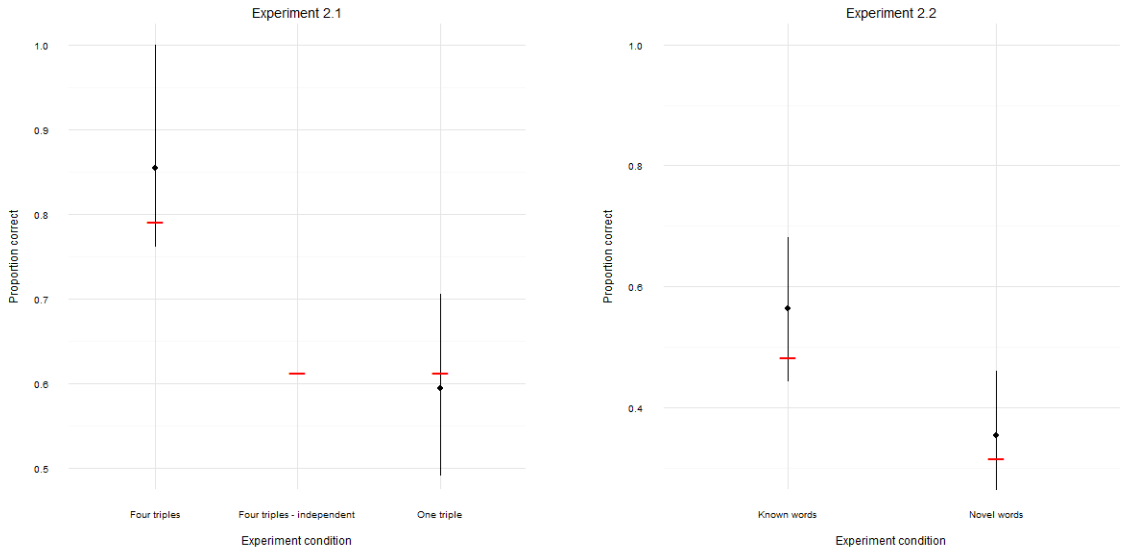

Figure 25. Model simulation results for Experiments 2.1 and 2.2. The black lines show the 95% HDI for each condition, with the circle showing the modal value. The red line shows the model's prediction for each condition. In the results for Experiment 2.1 (left), the middle red line shows the model's prediction when all four accumulators are independent.

alternative forced choice tests to assess learning of the statistical structure. The PANDA model generates learning onsets, but can be used for a learn-then-test procedure by assuming that a triple was learned if the corresponding accumulator finished before the end of the simulated experiment. Various assumptions can then be made about performance on the 2-AFC task as a function of the triples that are learned, but the simplest is that performance on a test for a triple and its foil is perfect if the triple was learned during the experiment, and at chance otherwise.

To model the four-triples and one-triple conditions, I used four accumulators in the four-triples condition and one accumulator in the one-triple condition. The number of accumulators that finished within 25 steps was recorded and converted to a proportion correct using the minimal assumption described in the previous paragraph. I fit the model using the DEoptim R package (Mullen, Ardia, Gil, & Cline, 2011), which implements a

type of genetic search algorithm called differential evolution (Storn & Price, 1997). This procedure does not guarantee that an optimal solution is found, but it is highly effective at searching multi-dimensional stochastic parameter spaces in a relatively short period. The objective function to minimize was the root mean squared error between the PANDA model's predictions and the modal estimate of the underlying proportion correct in each condition. Model predictions were generated by simulating the model 1,000 times with each candidate set of parameters and calculating the summary statistics. I fit the data from Experiment 2.1 and 2.2 simultaneously to find a set of shared parameters that accounts for both sets of data. The results for Experiment 2.2 are reported in the next section.

The best fitting parameters (RMSE = 0.0376) were $\mu = 0.032$, $\sigma = 15$, and $\varphi = 178$. Once the best fitting parameters were found, I simulated the model 10,000 times in each condition. Figure 25 shows the data and model predictions. Performance in all conditions was very close to the observed data and well inside the 95% HDIs.

The parameter of greatest theoretical interest is the dependency parameter, φ. In the one-triple condition, this parameter has no effect because it does not have any impact on the model until at least one accumulator finishes, but in the four-triple condition values of φ that are meaningfully above 0 suggest evidence of dependence between the accumulators. To make the effect of this parameter as transparent as possible, I simulated the model with the same values of $\mu = 0.032$ and $\sigma = 15$, but with $\varphi = 0$. This version of the model makes similar predictions to the one-accumulator model, and fails to capture the difference between conditions. In general, when $\varphi = 0$ the four-accumulator and one-accumulator models make the same predictions. To verify this, I ran both models on

1,020 different parameter combinations of μ and σ. μ values were 0.01 to 0.2 in steps of 0.01, and σ values were 1 to 101 in steps of 2. The correlation coefficient between the predicted accuracy of both models was greater than 0.999. Thus, the only way for the PANDA model to generate the empirical result of four-triples being easier to learn than one-triple is to include dependency in the model ($\varphi > 0$).

4.2.2 Experiment 2.2

In Experiment 2.2, participants saw four triples 20 times each. In one condition, three of the triples were known words and one target triple was a novel word; in the other condition, all the triples were novel words. Participants were more likely to recall the order of the novel word(s) when surrounded by known words.

Modeling the effect of prior knowledge requires a small modification to the model: the accumulators for familiar words start with more evidence (or, equivalently, that the threshold for familiar words is lower). This reflects an assumption that some evidence is still needed to recognize the presence of the word in the sequence, but not as much evidence as would be needed if the word was novel. The choice of how much evidence the accumulators start with unavoidably adds a parameter to the model. For the known-words condition, the model had three accumulators that started with a proportion ε of pre-accumulated evidence. The data were fit simultaneously with the data from Experiment 2.1. The best fitting value of ε was $\frac{2}{15}$.

The model's predictions are inside the 95% HDI in both conditions, and correctly predict the advantage of known words. However, this difference depends on having dependency among the accumulators, because the *target* accumulator in the known-words condition is equivalent to the accumulators in the novel-words condition. Only the

context accumulators, which do not contribute to the final score, are pre-seeded with evidence. The target accumulator is more likely to finish in the known-words condition because the other accumulators finish earlier and boost the sampling rate of the target accumulator.

4.2.3 Experiment 3.2

In Experiment 3.2, participants completed a serial response time task by typing letters that appeared on the screen. All participants repeatedly saw a target novel word embedded throughout the sequence. The content of the sequence varied between participants. For some the sequence consisted of three known words. Others saw three novel words, and the final group saw scrambled letters that did not form predictable units. The key results are (1) learning of the target happened earliest in the known words condition, followed by the novel words condition, and finally the scrambled condition, and (2) participants were most likely to learn in the known words condition, followed by the novel words condition, and then the scrambled condition. Because the PANDA model generates the finishing time for each accumulator, the model can make predictions about both the onset of learning and the probability of learning before the experiment ends. Thus, the data for fitting the model are the mean onsets of learning in each condition and the proportion of participants who learned in each condition. These data are themselves estimates, generated by the individual-level analysis model.

In each condition, a single accumulator represents the target word. In the scrambled condition, this is the only accumulator in the model. In the other two conditions, there are four accumulators. As in the model of Experiment 2.2, the

difference between the unknown and known words conditions is modeled by seeding the three non-target accumulators with evidence in the known-words condition.

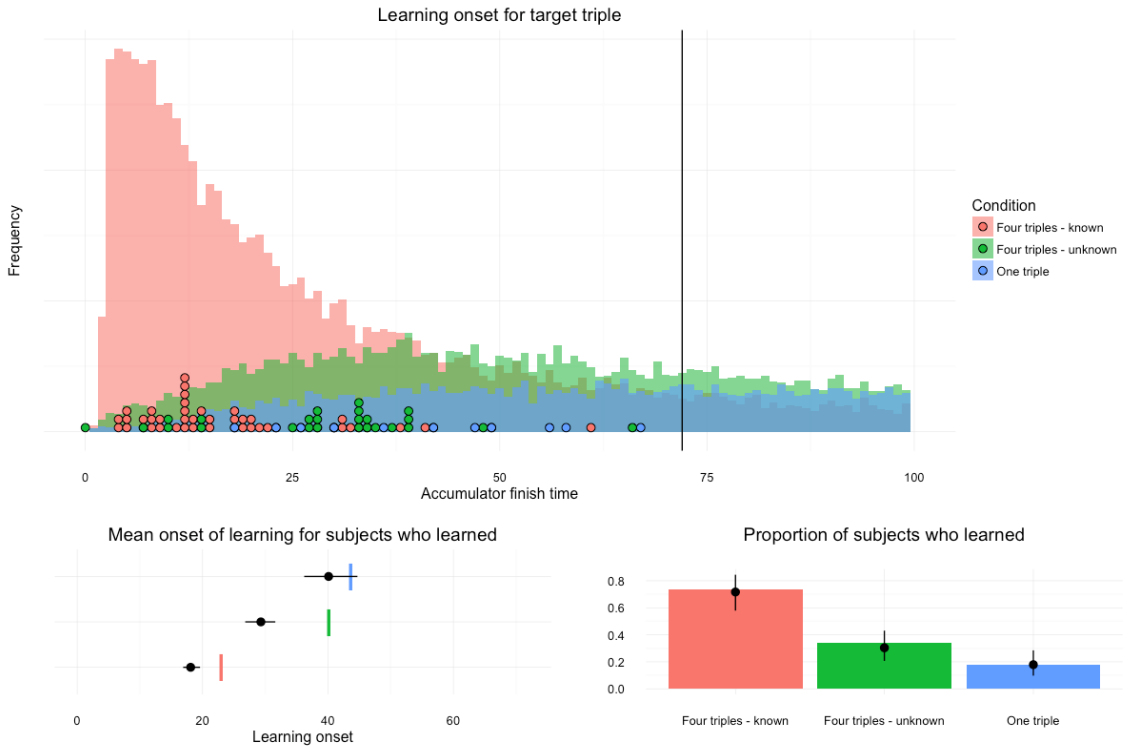

Figure 26. Model fit for Experiment 3.2. Top: The colored histograms show the distribution of accumulator finishing times in the model. The colored circles show the estimated onset of learning predicted by the individual-level model. The solid black line shows the end of the experiment. Many accumulators finished after $t = 100$, but they are not shown here for visual clarity. Bottom left: The black lines show the 95% HDI for the onset of learning for participants who learned in each condition. The black circle is the median of the HDI. The colored lines show the mean finishing time for the target accumulator in cases where the accumulator finished before the end of the experiment. Bottom right: The 95% HDIs for the probability of learning in each condition are shown with the black lines. The colored bars show the model predictions.

I first attempted to fit the standard model identical parameters across all individuals. I again used DEoptim. The objective function was the RMSE of the model's predictions for the learning onsets and proportion of learners in each of the three conditions (six total values). The error for each value was scaled to the range of the dependent measure, to ensure equal weight was given to errors of proportion and onsets.

Model predictions were generated by simulating the model 1,000 times with the candidate parameter values and calculating the summary statistics for fitting. The best fitting parameter set was $\mu = 0.004,\ \sigma = 2, \varphi = 325$, and $\varepsilon = 1/2$ (RMSE=0.548). These parameters produce the correct qualitative order of the effects, but do not capture the observed difference in learning times particularly well. In all three conditions, the model significantly overestimates the average onset of learning. Furthermore, the predicted difference in onset between the scrambled and unknown-words conditions is very small (44.8 and 46.7) compared to the actual difference in onsets (29.3 and 40.1).

The model fit improves if the values of μ and σ are assumed to vary across participants. I created a hierarchical model where individual μ values are randomly generated from an exponential distribution with rate $1/\tau$ and individual σ values are drawn from a Poisson distribution with rate $\lambda - 1$, and then incremented by 1. This was to ensure that the minimum value of σ is 1. Using single-parameter distributions to describe the variability in μ and σ avoids the problem of adding additional parameters to the model, allowing for simple model comparisons using the RMSE[3].

The best fitting parameters for this updated model were $\tau = 0.006$, $\lambda = 4$, $\varphi = 77$, and $\varepsilon = \frac{3}{4}$ (RMSE = 0.259). Figure 26 shows the data and model predictions. The model closely matched the proportion of learners in all three conditions, though it

[3] The fit improves substantially by assuming additional complexity in the distribution of μ values. If two beta distributions with different modes and shared concentration parameter are used to generate μ and a parameter is added to control the probability of coming from either beta distribution, then the model makes nearly perfect predictions about the proportion and onset of learning. The best fitting parameters end up generating a distribution of μ that is unimodal with strong rightward skew, despite the mixture of two distinct distributions. The shape is similar to the exponential, but with a larger concentration of values in low values of μ and a longer tail.

overestimated the onset of learning in the unknown- and known-words conditions. However, the empirical distributions of learning onsets and the distributions predicted by the model are reasonably close in all three conditions, despite the model not being fit to the individual level data. Once again, the model fit required strong dependency between accumulators to describe the data accurately.

4.2.4 Fitting Individual Learning Onsets

The unknown-words condition in Experiment 3.2 provides another opportunity to test for dependence in learning. In this condition, all four words were unfamiliar triples. Each participant generated a set of learning onsets. The model makes predictions about each element in this set, and can be used to look for evidence of dependence within an individual participant's data instead of at the group level.

Fitting this data first required extracting the onsets of learning. I created a modified version of the individual-level model used to find the single onset of learning in Experiments 3.1 and 3.2. The modified version fit the onset for each of the predictable letters in the sequence and estimated the probability that an individual participant showed evidence of learning. Even though the model estimated onsets for all eight predictable letters, in the analysis below I focus only on the third letter from each triple to remain consistent with the analyses of the full data set. The full model and fitting procedure are described in Appendix D.

Of the 97 participants in the unknown-words condition in Experiment 3.2, 30 of them (30.9%) were classified as a learner in at least 95% of the samples from the posterior distribution. This closely matches the estimate from the individual-level model in Experiment 3.2 that predicted a .304 probability of learning for participants in the

novel-words condition. When the model classified participants as learners, it fit onsets and learning curves for all the predictable elements. This means that there may be some letters that participants show little or no evidence of learning, but still have onset values. To account for this possibility, I filtered the onsets to include only items where the participant demonstrated reasonably strong learning ($\beta_L > 0.2$). There were 9 onsets out of 120 that fell below this criterion. 24 of the 30 participants showed robust learning for all four predictable letters. I used the onsets from these 24 participants as the basis for fitting the model. Filtering the data in this way introduces some bias into the parameter estimates, but it is not obvious how to handle cases where learning did not happen when fitting against individual-level onset data. This approach avoids assumptions at the cost of a selection bias. Most, though not all, of the participants who were removed had onset times near the end of the experiment for the letters that they did learn, suggesting that they would have learned the other letters if given enough time.

For fitting the data, I recoded each participant's set of four onsets. The first onset remained the same. Each of the three subsequent onsets was recoded to be the difference from the first onset. The model predictions were coded in the same way. This was done to preserve the within-participant structure of the data for fitting. I used DEoptim to fit the data, and assumed that all participants had identical parameter values. I simulated 1,000 repetitions of the model for each candidate parameter set and took the average initial onset and average differences from the initial onset to generate four data points as the model's prediction. Then I found the RMSE between the model's prediction and the set of empirical onsets. This served as the objective function for optimization.

The best fitting parameter values were $\mu = 0.02$, $\sigma = 4$, and $\varphi = 1.2$. Compared to the group-level model fits, μ is substantially larger and φ is much smaller. This may have to do with the fact that only participants who learned all four words were included in the data. In the group level model, many participants failed to learn, hence the lower value of μ in those fits. A very high value of φ coupled with a low value of μ ensures that many participants will fail to learn, but when participants get more opportunities to learn because there are more words, they will occasionally learn a single item which then exerts a huge influence on the likelihood of learning a second. Removing all of the participants who fail to learn artificially inflates the estimate of μ, which means smaller values of φ are necessary. However, even though φ was not as large as in the group model fits, it was still above zero. To make the effect of these parameters more concrete, when $\mu = 0.02$ and $\varphi = 1.2$, the probability of accumulating evidence when no accumulators have finished is 0.02. When one accumulator finishes, the probability more than doubles to 0.043, after two have finished the probability is 0.066, and after three have finished it is 0.089. Thus, there is still evidence of non-trivial dependence in learning when looking at a highly restricted set of the data.

4.3 Discussion

In this chapter, I developed an accumulator model to predict when learning occurs as multiple items are learned simultaneously. The model was fit to group-level data for three of the experiments reported in Chapters 2 and 3, as well as individual-level data for a subset of the participants in Experiment 3.2. Two trends were evident from the model fits. First, without dependency between the parallel accumulators, the model could not predict the correct qualitative results in any of the experiments, but the model generally

did very well when there was dependency. Second, the best fitting parameter values suggest that the accumulation process is noisy, with highly variable amounts of evidence accumulated on any particular trial. In the following sections, I discuss how viewing statistical learning as an evidence accumulation process can refine models of the underlying mechanisms.

4.3.1 Multiple Paths to Dependency

The PANDA model shows that a basic evidence accumulation model of statistical learning can account for differences in recognition tests, group-level and individual-level learning onsets, and the overall proportion of participants who learn only when the accumulation rate increases after each accumulator finishes. This generalizes the modeling results from Chapter 2, showing that a compressible memory mechanism is not strictly necessary. Any mechanism that produces dependency between items could explain the findings.

A logical path forward is to refine the notion of dependency in models that implement less abstract learning processes, as PARSER does (Perruchet & Vinter, 1998), and then compare competing accounts. However, given the variety of ways that statistical learning can be altered by previous experience (Turk-Browne, 2012), there are multiple possible mechanisms that could create the kind of dependency that the PANDA model relies on. Perhaps we should not expect there to be a single account of the dependency phenomenon. In this case, a descriptive model like PANDA be used to compare general common features of the learning process across different situations even if the underlying mechanisms are different.

4.3.2 What is Learned in a Single Exposure?

A defining feature of statistical learning is that a single exposure is insufficient to extract the underlying structure. Distributional information across exposures is what supports learning. Yet a mechanistic account of statistical learning must describe what is happening on a single trial to enable the discovery of patterns across trials. Proposals include an assortment of mechanisms, such as error-driven associative learning (Elman, 1990; French et al., 2011; Gureckis & Love, 2010), hypothesis formation and refinement (Brent & Cartwright, 1996; Pearl et al., 2010), and encoding of memory traces of individual items and/or chunks (Jamieson & Mewhort, 2009; Perruchet & Vinter, 1998; Robinet et al., 2011). All of these approaches have successfully accounted for empirical data to some degree, suggesting that there are multiple approaches that can be flexibly deployed depending on the situation (e.g., Hunt & Aslin, 2001; Sun et al., 2005) and/or that the models are instantiating similar function through different implementations (C. Yu & Smith, 2012).

An abstract evidence accumulation model lets us infer some high-level properties of what gets learned on a single trial. In particular, the variability parameter σ tells us how consistent the effect of a single exposure is. If the fit value of σ were very large, then that would point to a reliable effect of a single trial where approximately the same amount of evidence is accumulated on each presentation of the event. When σ is small, the effect of a single trial is uncertain. At the extreme, if σ is 1 then learning in a single trial is all-or-nothing.

The fits for σ in these experiments were in the range of 4 to 15. This means that learning is quite noisy, but not all-or-nothing. This finding is congruent with recent

evidence in cross-situational word learning showing that the models supposing unequal, but not all-or-nothing, distribution of attention across potential word-object pairs outperform models that posit equally divided or all-or-nothing attention (Yurovsky & Frank, 2015). In other words, on some trials a particular item might accumulate very little evidence if attention is primarily directed elsewhere, but on other trials the evidence accumulation might be very large if attention is focused on that item. Characterizing the noise of learning at the individual trial level helps constrain the space of possible process models in a way that would be challenging to do at the algorithmic level due to model mimicry (C. Yu & Smith, 2012). With the right modifications, many different kinds of models – associative, connectionist, chunking, hypothesis testing, and so on – can likely implement a noisy learning process at the trial level. Examining the computational-level processes that these models converge towards is a fruitful way to understand the mechanisms underlying statistical learning.

4.3.3 A Cautionary Note

The best-fitting parameter values – and the resulting interpretation of the learning process – depend on assumptions about how parameter values across individuals do or do not vary. In general, I tried to fit the data using as few parameters as possible, which required the assumption that participants were either identical or varied in particular ways. Different assumptions produce drastically different parameter estimates. For example, if we assume that the accumulation rate varies with no constraints across participants, then the best fitting model is one where each participant's accumulation rate is roughly $\frac{1}{t}$ with very high values of σ to ensure that learning always happens at that time.

This situation warrants model selection with appropriate penalties for extra degrees of freedom. However, using these techniques with PANDA is challenging because the likelihood function is unknown and difficult to compute. One solution is to use an empirical likelihood function, created by simulating the model many times and observing the probability of an event (Turner & Sederberg, 2014). What makes this approach computationally expensive for PANDA is that the outcome space grows exponentially with the number of accumulators. The number of possible outcomes is t^n, where t is the maximum time allotted for an accumulator to finish and n is the number of accumulators. Therefore, the number of simulations required to get a sufficiently smooth distribution for empirical likelihoods grows rapidly with the number of accumulators. In tests of this procedure with PANDA, at least 10 million simulations were needed to begin to approximate smooth distributions for a single set of parameters when there were 4 accumulators and 72 possible finishing times. Further refinement of this process, perhaps by using dynamic programming techniques to enumerate the likelihood function, would enable more nuanced model comparisons.

5 General Discussion

5.1 Summary of Findings

In this dissertation, I investigated how constraints on learning change during learning. In Chapter 2, I investigated how memory constraints on sequential statistical learning are altered by learning to chunk portions of the sequence and knowing portions of the sequence in advance. I tested several existing process models of chunking on these tasks, but only PARSER (Perruchet & Vinter, 1998) was able to account for the results of both experiments. Through modifications of the PARSER model, I showed that the reason PARSER is able to account for the results is because it implements a form of compressible memory. Learning in PARSER requires encountering the same chunks repeatedly before they are forgotten. When portions of the sequence are chunked, they require fewer memory resources and other portions of the sequence are more likely to be learned. PARSER is one example of how learning and constraints based on generic cognitive processes interact on short timescales in statistical learning.

In Chapter 3, I extended these results by measuring learning during exposure to the sequence. I was able to characterize the shape of the learning curve for individual items within the sequence for individual participants. Response times for predictable items in the sequence showed gradual adaptation to the task and then a sudden shift, due to learning the statistical structure. Using a set of manipulations that were analogous to the experiments in Chapter 2, I showed that the difference in learning outcomes was due to participants learning earlier in the sequence, but not at a faster rate, in situations where they could chunk contextual information. These results suggest that constraints like those described by PARSER affect *when* something is learned, but not the *shape* of (the

behavioral result of) learning, echoing similar findings in the associative learning literature (Gallistel et al., 2004).

In Chapter 4, I developed a model to predict the transition between adaptation and learning (the onset of behavior evidence of learning) when multiple items are learned simultaneously. The model describes learning at the abstract level of evidence accumulation. Fitting the model to the data from the experiments in Chapters 2 and 3, including both group-level and individual item-level data, showed that the learning rate on any individual trial was (a) influenced by the number of items that had already been learned, and (b) moderately noisy. The importance of dependency between the items for successful model fitting is further evidence for the idea of constraints that vary because of learning, and generalizes the idea beyond the memory-based constraints of PARSER. The level of noise at each opportunity for learning suggests the presence of mechanisms that are somewhere between all-or-nothing and consistent. PARSER happens to describe a learning process that has this characteristic, but there is no in principle reason that other learning mechanisms could not do the same.

5.2 Open Questions

There are several implications of these results for the development and refinement of statistical learning models. First is the importance of dependency. The PANDA model found general support for this idea, but of the other models tested, the only one that implemented a candidate mechanism was PARSER. What other mechanisms might create the kinds of dependencies that the PANDA model predicts? There are empirical hints that selective attention mechanisms might function in the same way (R. Q. Yu &

Zhao, 2015; Zhao et al., 2013, 2011), but these have not yet been implemented in models of statistical segmentation learning.

The distinct behavioral transition point in the response time measurements raises several questions. First: What is happening during the period of exposure when there is no behavioral evidence of learning? It seems necessary that some kind of learning happens during this period, otherwise there is no causal explanation for the onset of learning. There is some neuroimaging evidence indicating that the early stage of learning may occur without a strong effect on behavior. Turk-Browne, Scholl, Chun, and Johnson (2009) were able to detect differential responses in BOLD activity for predictable and unpredictable elements in a standard triple-based statistical learning task even though participants did not show above-chance classification in a forced choice task for familiar versus unfamiliar triples. However, as Vadillo, Konstantinidis, and Shanks (2015) warn, forced choice tasks may be underpowered when participants only learn a subset of the items and attempt each distinction only a few times. Further work is needed to clarify the relationship between neuroimaging markers of learning and behavioral evidence of learning.

Key to understanding the phase of learning prior to the transition point is determining the cause of the sudden onset of rapid changes in response time. One possibility is that there are two kinds of learning processes (c.f. Thiessen, Kronstein, & Hufnagle, 2013), but only one causes detectable changes in response times. A proposal along this line from cross-situational word learning is that learners generate hypotheses about word-referent mappings from knowledge accumulated via associative learning mechanisms (Romberg & Yu, 2014, 2015). If sequence segmentation uses a similar set of

processes, the sudden shift in response times might coincide with the formation of explicit predictions about the statistical structure. This seems plausible given the high degree of explicit knowledge reported by participants who showed behavioral evidence of learning.

Another possibility is that there is a single process that generates the transition point in behavior. PARSER is potentially an example of this because of the shaping threshold in the model, but there is no explicit mapping from PARSER to behavior. This is a general issue with models of statistical learning, which makes the application of these models to response times – or even to accuracy measures – challenging. While fairly standard approaches based on the Luce choice rule have emerged for choice tasks (e.g., Cleeremans & McClelland, 1991; Frank, Goldwater, et al., 2010; Gureckis & Love, 2010), even these mappings are inherently ambiguous. For example, suppose there is a two-alternative choice between the word `ABC` and the part-word `EF/G`. PARSER's internal lexicon might contain chunks like ABC and EF, as well as individual tokens like A, F, and G. Should the representational strength used for the choice rule be just ABC and EFG, or should partial matches be considered as well? Choosing how to handle these situations creates variations of the model. Most models of statistical learning are like PARSER – describing functional mechanisms and representational changes at a level that is not directly connected to behavior. If the model of learning permits too much flexibility in the kinds of behaviors that can be generated, then it is difficult to falsify or refine the model.

This issue is especially important when considering the kinds of behaviors that are often explained by these models: group-level averages or individual-level averages

(averaging over multiple items). Experiments 3.1 and 3.2 were examples of the behavior of the group being distinctly different from the behavior of the individuals. If models describe learning at the level of the individual item, then it is critical to measure behavior at this level or as close to it as possible. The serial response time paradigm is relatively powerful for this kind of analysis, yet is limited in the kinds of questions and statistical structures that it can test. A more general, but less sensitive, approach is to asses learning via accuracy measures. Gallistel et al. (2004) describe a technique for measuring individual learning curves when the response on each trial is a binary outcome. The approach recursively searches for change points where the frequency of correct responses before and after the change point are significantly different. Adopting this analytic strategy in a broader range of statistical learning tasks would produce more informative data about the nature of the learning processes at the level of individual trials.

Finally, perhaps most importantly, is the generalizability of these findings. To what extent should we expect other statistical learning tasks to show similar traits to those observed here? The answer largely depends on how we view the process of statistical learning. If the "family of processes" (e.g., Bays et al., 2016) framework is right, then other statistical learning tasks might have gradual learning curves or independent learning processes among items. On the other hand, a domain-general or singular process account would predict similar kinds of learning curves – though even in this case the curves might vary depending on the input that the statistical learning mechanisms operate on (Frost et al., 2015). Characterizing these features of learning across tasks, stimuli, and statistical structures may be one of the most effective ways to investigate the question of multiple or common processes.

References

Abrahamse, E. L., Jiménez, L., Verwey, W. B., & Clegg, B. a. (2010). Representing serial action and perception. *Psychonomic Bulletin & Review*, *17*(5), 603–623. doi:10.3758/PBR.17.5.603

Ackerman, P. L. (1987). Individual Differences in Skill Learning: An Integration of Psychometric and Information Processing Perspectives. *Psychological Bulletin*, *102*(1), 3–27. doi:10.1037/0033-2909.102.1.3

Addyman, C. (2015). TRACX-Web: Zenodo DOI linking. *Zenodo*. doi:10.5281/zenodo.16436

Altmann, G. T. M., Dienes, Z., & Goode, A. (1995). Modality independence of implicitly learned grammatical knowledge. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *21*(4), 899–912. doi:10.1037/0278-7393.21.4.899

Anderson, R. B. (2001). The power law as an emergent property. *Memory & Cognition*, *29*(7), 1061–8. doi:10.3758/BF03195767

Anderson, R. B., & Tweney, R. D. (1997). Artifactual power curves in forgetting. *Memory & Cognition*, *25*(5), 724–30. doi:10.3758/BF03211315

Aslin, R. N. (2014). Infant learning: Historical, conceptual, and methodological challenges. *Infancy*, *19*(1), 2–27. doi:10.1111/infa.12036

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. doi:10.1177/0963721412436806

Aslin, R. N., Saffran, J. R., & Newport, E. L. (1998). Computation of Conditional Probability Statistics by 8-Month-Old Infants. *Psychological Science*, *9*(4), 321–

324. doi:10.1111/1467-9280.00063

Baker, C. I., Olson, C. R., & Behrmann, M. (2004). Role of attention and perceptual grouping in visual statistical learning. *Psychological Science*, *15*(7), 460–466.

Barakat, B. K., Seitz, A. R., & Shams, L. (2013). The effect of statistical learning on internal stimulus representations: Predictable items are enhanced even when not predicted. *Cognition*, *129*(2), 205–211. doi:10.1016/j.cognition.2013.07.003

Bays, B. C., Turk-Browne, N. B., & Seitz, A. R. (2016). Dissociable behavioural outcomes of visual statistical learning. *Visual Cognition*, 1–26. doi:10.1080/13506285.2016.1139647

Bolker, B. (2016). emdbook: Ecological Models and Data in R.

Boucher, L., & Dienes, Z. (2003). Two ways of learning associations. *Cognitive Science*, *27*(6), 807–842. doi:10.1016/j.cogsci.2003.03.001

Bower, G. H. (1969). Chunks as interference units in free recall. *Journal of Verbal Learning and Verbal Behavior*, *8*, 610–613. doi:10.1016/S0022-5371(69)80112-X

Brady, T. F., Konkle, T., & Alvarez, G. a. (2009). Compression in visual working memory: using statistical regularities to form more efficient memory representations. *Journal of Experimental Psychology: General*, *138*(4), 487–502. doi:10.1167/8.6.199

Brent, M. R., & Cartwright, T. a. (1996). Distributional regularity and phonotactic constraints are useful for segmentation. *Cognition*, *61*(1-2), 93–125. doi:10.1016/S0010-0277(96)00719-6

Brown, S. D., & Heathcote, A. (2008). The simplest complete model of choice response time: Linear ballistic accumulation. *Cognitive Psychology*, *57*(3), 153–178.

doi:10.1016/j.cogpsych.2007.12.002

Buchner, A., Steffens, M. C., & Rothkegel, R. (1998). On the role of fragmentary knowledge in a sequence learning task. *The Quarterly Journal of Experimental Psychology*, *51A*(2), 251–281.

Chase, W. G., & Simon, H. A. (1973). Perception in chess. *Cognitive Psychology*, *61*, 55–61.

Christiansen, M. H., & Chater, N. (2016). The Now-or-Never Bottleneck: A Fundamental Constraint on Language. *Behavioral and Brain Sciences*, *e62*, 1–72. doi:10.1017/S0140525X1500031X

Christiansen, M. H., & Conway, C. M. (2006). Statistical learning within and between modalities. *Psychological Science*, *17*(10), 905–912. doi:10.1111/j.1467-9280.2006.01801.x

Cleeremans, A., & Dienes, Z. (2008). Computational models of implicit learning. In R. Sun (Ed.), *Cambridge Handbook of Computational Psychology* (pp. 396–421). New York: Cambridge University Press.

Cleeremans, A., & McClelland, J. L. (1991). Learning the structure of event sequence. *Journal of Experimental Psychology: General*, *120*(3), 235–253.

Conway, C. M., & Christiansen, M. H. (2005). Modality-constrained statistical learning of tactile, visual, and auditory sequences. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *31*(1), 24–39. doi:10.1037/0278-7393.31.1.24

Cowan, N. (2010). The Magical Mystery Four: How Is Working Memory Capacity Limited, and Why? *Current Directions in Psychological Science*, *19*(1), 51–57. doi:10.1177/0963721409359277

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), 1119–1130. doi:10.1037/0278-7393.30.5.1119

Cunillera, T., Càmara, E., Laine, M., & Rodríguez-Fornells, A. (2010). Words as anchors: Known words facilitate statistical learning. *Experimental Psychology*, *57*(2), 134–141. doi:10.1027/1618-3169/a000017

Dahan, D., & Brent, M. R. (1999). On the discovery of novel wordlike units from utterances: an artificial-language study with implications for native-language acquisition. *Journal of Experimental Psychology: General*, *128*(2), 165–185. doi:10.1037/0096-3445.128.2.165

de Leeuw, J. R. (2015). jsPsych: A JavaScript library for creating behavioral experiments in a Web browser. *Behavior Research Methods*, *47*(1), 1–12. doi:10.3758/s13428-014-0458-y

de Leeuw, J. R., & Goldstone, R. L. (2015). Memory constraints affect statistical learning; statistical learning affects memory constraints . In D. C. Noelle, R. Dale, A. S. Warlaumont, J. Yoshimi, T. Matlock, C. D. Jennings, & P. P. Maglio (Eds.), *Proceedings of the 37th Annual Meeting of the Cognitive Science Society* (pp. 530–535). Austin, TX: Cognitive Science Society.

de Leeuw, J. R., & Motz, B. A. (2016). Psychophysics in a web browser? Comparing response times collected with JavaScript and Psychophysics Toolbox in a visual search task. *Behavior Research Methods*, *48*(1), 1–12.

Denwood, M. J. (2013). runjags: An R package providing interface utilities, parallel

computing methods, and additional distributions for MCMC models in JAGS.

Denwood, M. J. (2014). runjags: Interface utilities, parallel computing methods and additional distributions for MCMC models in JAGS.

Destrebecqz, A., & Cleeremans, A. (2001). Can sequence learning be implicit? New evidence with the process dissociation procedure. *Psychonomic Bulletin & Review*, *8*(2), 343–350. doi:10.3758/BF03196171

Dienes, Z., & Berry, D. (1997). Implicit learning: below the subject threshold. *Psychonomic Bulletin & Review*, *4*(1), 3–23. Retrieved from http://sro.sussex.ac.uk/14463/

Dienes, Z., Broadbent, D. E., & Berry, D. C. (1991). Implicit and explicit knowledge bases in artificial grammar learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *17*(5), 875–87. doi:doi:10.1037/0278-7393.17.5.875

Donner, Y., & Hardy, J. L. (2015). Piecewise power laws in individual learning curves. *Psychonomic Bulletin & Review*, *22*(5), 1308–1319. doi:10.3758/s13423-015-0811-x

Eddelbuettel, D., & Fran, R. (2011). Rcpp : Seamless R and C ++ Integration. *Journal Of Statistical Software*, *40*, 1–18. doi:10.1007/978-1-4614-6868-4

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211. doi:10.1016/0364-0213(90)90002-E

Elman, J. L. (1993). Learning and development in neural networks: the importance of starting small. *Cognition*, *48*(1), 71–99. doi:10.1016/0010-0277(93)90058-4

Elman, J. L., Bates, E. A., Johnson, M. H., Karmiloff-Smith, A., Parisi, D., & Plunkett, K. (1996). *Rethinking Innateness: Connectionist Perspective on Development*.

Rethinking Innateness: {A} Connectionist Perspective on Development. Retrieved from http://crl.ucsd.edu:80/~elman/Papers/book/index.shtml

Emberson, L. L., Liu, R., & Zevin, J. D. (2013). Is statistical learning constrained by lower level perceptual organization? *Cognition*, *128*(1), 82–102. doi:10.1016/j.cognition.2012.12.006

Endress, A. D., Nespor, M., & Mehler, J. (2009). Perceptual and memory constraints on language acquisition. *Trends in Cognitive Sciences*, *13*, 348–353. doi:10.1016/j.tics.2009.05.005

Ericsson, K. A., & Kintsch, W. (1995). Long-term working memory. *Psychological Review*, *102*(2), 211–245. doi:10.1037/0033-295X.102.2.211

Estes, W. K. (1956). The problem of inference from curves based on group data. *Psychological Bulletin*, *53*(2), 134–140. doi:10.1037/h0045156

Estes, W. K. (1960). Learning theory and the new "mental chemistry." *Psychological Review*, *67*(4), 207–223. doi:10.1126/science.3.71.712-a

Estes, W. K. (2002). Traps in the route to models of memory and decision. *Psychonomic Bulletin & Review*, *9*(1), 3–25. doi:10.3758/BF03196254

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

Fiser, J., & Aslin, R. N. (2002). Statistical learning of higher-order temporal structure from visual shape sequences. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *28*(3), 458–467. doi:10.1037/0278-7393.28.3.458

Fiser, J., & Aslin, R. N. (2005). Encoding Multielement Scenes: Statistical Learning of Visual Feature Hierarchies. *Journal of Experimental Psychology: General*, *134*(4),

521–537. doi:10.1037/0096-3445.134.4.521

Fiser, J., Scholl, B. J., & Aslin, R. N. (2007). Perceived object trajectories during occlusion constrain visual statistical learning. *Psychonomic Bulletin & Review*, *14*(1), 173–178. doi:10.3758/BF03194046

Forstmann, B. U., Ratcliff, R., & Wagenmakers, E.-J. (2016). Sequential Sampling Models in Cognitive Neuroscience: Advantages, Applications, and Extensions. *Annual Review of Psychology*, *67*, 641–66. doi:10.1146/annurev-psych-122414-033645

Frank, M. C., & Gibson, E. (2011). Overcoming Memory Limitations in Rule Learning. *Language Learning and Development*, *7*, 130–148. doi:10.1080/15475441.2010.512522

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

Frank, M. C., Tenenbaum, J. B., & Gibson, E. (2013). Learning and Long-Term Retention of Large-Scale Artificial Languages. *PLoS ONE*, *8*(1), 1–6. doi:10.1371/journal.pone.0052500

Frank, M. C., Tily, H., Arnon, I., & Goldwater, S. (2010). Beyond transitional probabilities: Human learners impose a parsimony bias in statistical word segmentation. In S. Ohlsson & R. Catrambone (Eds.), *Proceedings of the 32nd Annual Meeting of the Cognitive Science Society* (pp. 760–765). Austin, TX: Cognitive Science Society.

French, R. M., Addyman, C., & Mareschal, D. (2011). TRACX: a recognition-based

connectionist framework for sequence segmentation and chunk extraction. *Psychological Review, 118*(4), 614–636. doi:10.1037/a0025255

French, R. M., & Cottrell, G. W. (2014). TRACX 2.0 : A memory-based, biologically-plausible model of sequence segmentation and chunk extraction. In P. Bello, M. Guarini, M. McShane, & B. Scassellati (Eds.), *Proceedings of the 36th Annual Conference of the Cognitive Science Society* (pp. 2216–2221). Austin, TX: Cognitive Science Society.

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

Gallistel, C. R., Fairhurst, S., & Balsam, P. (2004). The learning curve: implications of a quantitative analysis. *Proceedings of the National Academy of Sciences, 101*(36), 13124–31. doi:10.1073/pnas.0404965101

Gebhart, A. L., Aslin, R. N., & Newport, E. L. (2009). Changing structures in midstream: Learning along the statistical garden path. *Cognitive Science, 33*, 1087–1116. doi:10.1111/j.1551-6709.2009.01041.x

Gelman, A., & Rubin, D. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science, 7*(4), 457–472.

Gerganov, A., Grinberg, M., Quinn, P. C., & Goldstone, R. L. (2007). Simulating Conceptually-Guided Perceptual Learning. In D. S. McNamara & J. G. Trafton (Eds.), *Proceedings of the Twenty-ninth Annual Conference of the Cognitive Science Society* (pp. 287–292). Austin, TX: Cognitive Science Society. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.219.895

Giroux, I., & Rey, A. (2009). Lexical and sublexical units in speech perception. *Cognitive Science, 33*, 260–272. doi:10.1111/j.1551-6709.2009.01012.x

Gobel, E. W., Sanchez, D. J., & Reber, P. J. (2011). Integration of temporal and ordinal information during serial interception sequence learning. *Journal of Experiment Psychology: Learning, Memory, & Cognition*, *37*(4), 994–1000. doi:10.1037/a0022959

Gobet, F., Lane, P. C., Croker, S., Cheng, P. C. H., Jones, G., Oliver, I., & Pine, J. M. (2001a). Chunking mechanisms in human learning. *Trends in Cognitive Sciences*, *5*(6), 236–243.

Gobet, F., Lane, P. C. R., & Lloyd-Kelly, M. (2015). Chunks, schemata, and retrieval structures: Past and current computational models. *Frontiers in Psychology*, *6*(Nov), 1–4. doi:10.3389/fpsyg.2015.01785

Gobet, F., Lane, P., Croker, S., Cheng, P., Jones, G., Oliver, I., & Pine, J. (2001b). Chunking mechanisms in human learning. *Trends in Cognitive Sciences*, *5*(6), 236–243. doi:10.1016/S1364-6613(00)01662-4

Gobet, F., & Simon, H. A. (1998). Expert chess memory: revisiting the chunking hypothesis. *Memory*, *6*(3), 225–255. doi:10.1080/741942359

Gold, E. M. (1967). Language identification in the limit. *Information and Control, 10*(5), 447–474. doi:10.1016/S0019-9958(67)91165-5

Goldstone, R. L., & Landy, D. (2010). Domain-Creating Constraints. *Cognitive Science*, *34*(7), 1357–1377. doi:10.1111/j.1551-6709.2010.01131.x

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

doi:10.1016/j.cognition.2009.03.008

Goujon, A., Didierjean, A., & Thorpe, S. (2015). Investigating implicit statistical learning mechanisms through contextual cueing. *Trends in Cognitive Sciences*, *19*(9), 524–533. doi:10.1016/j.tics.2015.07.009

Gureckis, T. M., & Love, B. C. (2010). Direct associations or internal transformations? Exploring the mechanisms underlying sequential learning behavior. *Cognitive Science*, *34*(1), 10–50. doi:10.1111/j.1551-6709.2009.01076.x

Haider, H., & Frensch, P. a. (2002). Why aggregated learning follows the power law of practice when individual learning does not: comment on Rickard (1997, 1999), Delaney et al. (1998), and Palmeri (1999). *Journal of Experiment Psychology: Learning, Memory, & Cognition*, *28*(2), 392–406. doi:10.1037/0278-7393.28.2.392

Heathcote, A., Brown, S., & Mewhort, D. J. (2000). The power law repealed: the case for an exponential law of practice. *Psychonomic Bulletin & Review*, *7*(2), 185–207. doi:10.3758/BF03212979

Hilbig, B. E. (2016). Reaction time effects in lab- versus Web-based research: Experimental evidence. *Behavior Research Methods*, Online epub ahead of print.

Hoekstra, R., Morey, R. D., Rouder, J. N., & Wagenmakers, E. J. (2014). Robust misinterpretation of confidence intervals. *Psychonomic Bulletin & Review*, *21*(5), 1157–1164. doi:10.3758/s13423-013-0572-3

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), 658–680.

Jamieson, R. K., & Mewhort, D. J. K. (2009). Applying an exemplar model to the serial

reaction-time task: anticipating from experience. *Quarterly Journal of Expertimental Psychology*, *62*(9), 1757–1783. doi:10.1080/17470210802557637

Jiménez, L. (2008). Taking patterns for chunks: Is there any evidence of chunk learning in continuous serial reaction-time tasks? *Psychological Research*, *72*(4), 387–396. doi:10.1007/s00426-007-0121-7

Johnson, E., & Tyler, M. (2010). Testing the limits of statistical learning for word segmentation. *Developmental Science*, *13*(2), 339–345. doi:10.1111/j.1467-7687.2009.00886.x.Testing

Karpicke, J. D., & Pisoni, D. B. (2004). Using immediate memory span to measure implicit learning. *Memory & Cognition*, *32*(6), 956–964.

Keil, F. C. (1990). Constraints on constraints: Surveying the epigenetic landscape. *Cognitive Science*, *14*(1), 135–168. doi:10.1016/0364-0213(90)90029-V

Kidd, C., Piantadosi, S. T., & Aslin, R. N. (2012). The Goldilocks effect: Human infants allocate attention to visual sequences that are neither too simple nor too complex. *PLoS ONE*, *7*(5), 1–8. doi:10.1371/journal.pone.0036399

Kidd, C., Piantadosi, S. T., & Aslin, R. N. (2014). The Goldilocks Effect in Infant Auditory Attention. *Child Development*, *85*(5), 1795–1804. doi:10.1111/cdev.12263

Klahr, D., Chase, W. G., & Lovelace, E. A. (1983). Structure and process in alphabetic retrieval. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *9*(3), 462–477.

Köhler, W. (1925). Intelligence in Apes. *Pedagogical Seminary and Journal of Genetic Psychology*, *32*, 674–690. doi:10.1037/11020-007

Köhler, W. (1959). Gestalt Psychology Today. *American Psychologist*, *14*, 727–734.

Kruschke, J. K. (2011). *Doing Bayesian Data Analysis: A Tutorial with R and BUGS* (1st ed.). Academic Press.

Kruschke, J. K. (2014). *Doing Bayesian Data Analysis: A Tutorial with R, JAGS, and STAN* (2nd ed.). New York: Academic Press.

Kruschke, J. K. (2015). Doing Bayesian Data Analysis 2nd Edition Programs.

Kruschke, J. K., & Liddell, T. M. (2015). The Bayesian new statistics: Two historical trends converge. Retrieved from http://ssrn.com/abstract=2606016

Kurumada, C., Meylan, S. C., & Frank, M. C. (2013). Zipfian frequency distributions facilitate word segmentation in context. *Cognition*, *127*, 439–453.

Lany, J., & Gómez, R. L. (2008). Twelve-month-old infants benefit from prior experience in statistical learning. *Psychological Science*, *19*(12), 1247–1252.

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

Ludwig, C. J. H., Farrell, S., Ellis, L. A., Hardwicke, T. E., & Gilchrist, I. D. (2012). Context-gated statistical learning and its role in visual-saccadic decisions. *Journal of Experimental Psychology: General*, *141*(1), 150–169. doi:10.1037/a0024916

Mathews, R. C., Buss, R. R., Stanley, W. B., Blanchard-Fields, F., Cho, J. R., & Druhan, B. (1989). Role of implicit and explicit processes in learning from examples: A synergistic effect. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *15*(6), 1083–1100. doi:10.1037/0278-7393.15.6.1083

Mathy, F., & Feldman, J. (2012). What's magic about magic numbers? Chunking and data compression in short-term memory. *Cognition*, *122*(3), 346–362.

doi:10.1016/j.cognition.2011.11.003

Mazur, J. E., & Hastie, R. (1978). Learning as accumulation: A reexamination of the learning curve. *Psychological Bulletin*, *85*(6), 1256–1274. doi:10.1037//0033-2909.85.6.1256

Merriman, W. E., & Bowman, L. L. (1989). The mutual exclusivity bias in children's word learning. *Monographs for the Society for Research in Child Development*, *54*(3-4), 1–129.

Miller, G. A. (1956). The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review*, *101*(2), 343–352. doi:10.1037/h0043158

Misyak, J. B., Christiansen, M. H., & Tomblin, J. B. (2010). Sequential expectations: The role of prediction-based learning in language. *Topics in Cognitive Science*, *2*(1), 138–153. doi:10.1111/j.1756-8765.2009.01072.x

Morey, R. D., Hoekstra, R., Rouder, J. N., Lee, M. D., & Wagenmakers, E.-J. (2015). The fallacy of placing confidence in confidence intervals. *Psychonomic Bulletin & Review*, *23*(1), 103–23. doi:10.3758/s13423-015-0947-8

Mullen, K. M., Ardia, D., Gil, D. L., & Cline, J. (2011). DEoptim : An R Package for Global Optimization by Differential Evolution. *Journal Of Statistical Software*, *40*(6), 1–17. doi:10.18637/jss.v040.i06

Murre, J. M. J., & Chessa, A. G. (2011). Power laws from individual differences in learning and forgetting: mathematical analyses. *Psychonomic Bulletin & Review*, *18*(3), 592–597. doi:10.3758/s13423-011-0076-y

Myung, I. J., Kim, C., & Pitt, M. a. (2000). Toward an explanation of the power law

artifact: insights from response surface analysis. *Memory & Cognition*, *28*(5), 832–840. doi:10.3758/BF03198418

Neath, I., Earle, A., Hallett, D., & Surprenant, A. M. (2011). Response time accuracy in Apple Macintosh computers. *Behavior Research Methods*, *43*(2), 353–62. doi:10.3758/s13428-011-0069-9

Newport, E. L. (1990). Maturational constraints on language learning. *Cognitive Science*, *14*(1), 11–28. doi:10.1016/0364-0213(90)90024-Q

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

Orbán, G., Fiser, J., Aslin, R. N., & Lengyel, M. (2008). Bayesian learning of visual chunks by human observers. *Proceedings of the National Academy of Sciences of the United States of America*, *105*(7), 2745–2750. doi:10.1073/pnas.0708424105

Pearl, L., Goldwater, S., & Steyvers, M. (2010). Online Learning Mechanisms for Bayesian Models of Word Segmentation. *Research on Language and Computation*, *8*(2-3), 107–132. doi:10.1007/s11168-011-9074-5

Perlman, A., Pothos, E. M., Edwards, D. J., & Tzelgov, J. (2010). Task-relevant chunking in sequence learning. *Journal of Experimental Psychology: Human Perception and Performance*, *36*(3), 649–661. doi:10.1037/a0017178

Perruchet, P., & Pacteau, C. (1990). Synthetic grammar learning: Implicit rule abstraction or explicit fragmentary knowledge? *Journal of Experimental Psychology: General*, *119*(3), 264–275. doi:10.1037/0096-3445.119.3.264

Perruchet, P., & Pacton, S. (2006). Implicit learning and statistical learning: one

phenomenon, two approaches. *Trends in Cognitive Sciences*, *10*(5), 233–238. doi:10.1016/j.tics.2006.03.006

Perruchet, P., & Peereman, R. (2004). The exploitation of distributional information in syllable processing. *Journal of Neurolinguistics*, *17*, 97–119. doi:10.1016/S0911-6044(03)00059-9

Perruchet, P., & Tillmann, B. (2010). Exploiting multiple sources of information in learning an artificial language: Human data and modeling. *Cognitive Science*, *34*, 255–285. doi:10.1111/j.1551-6709.2009.01074.x

Perruchet, P., & Vinter, A. (1998). PARSER: A Model for Word Segmentation. *Journal of Memory and Language*, *39*, 246–263. doi:10.1006/jmla.1998.2576

Perruchet, P., Vinter, A., Pacteau, C., & Gallego, J. (2002). The formation of structurally relevant units in artificial grammar learning. *The Quarterly Journal of Experimental Psychology*, *55A*(2), 485–503. doi:10.1080/02724980143000451

Plummer, M. (2003). JAGS : A Program for Analysis of Bayesian Graphical Models Using Gibbs Sampling. In K. Hornik, F. Leisch, & A. Zeileis (Eds.), *Proceedings of the 3rd International Workshop on Distributed Statistical Computing* (pp. 1–10).

Plummer, M. (2014). rjags: Bayesian graphical models using MCMC. *R Package Version 3-13*. doi:http://cran.r-project.org/package=rjags

R Core Team. (2012). *R: A language for data analysis and graphics*. Vienna, Austria: R Foundation for Statistical Computing. Retrieved from http://www.r-project.org

Reber, P. J. (2013). The neural basis of implicit learning and memory: A review of neuropsychological and neuroimaging research. *Neuropsychologia*, *51*(10), 2026–2042. doi:10.1016/j.neuropsychologia.2013.06.019

Reimers, S., & Stewart, N. (2014). Presentation and response timing accuracy in Adobe Flash and HTML5/JavaScript Web experiments. *Behavior Research Methods*. doi:10.3758/s13428-014-0471-1

Rescorla, R. A., & Wagner, A. R. (1972). A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. *Classical Conditioning II: Current Research and Theory, 2*, 64–99.

Roberts, G. O. (1996). Markov chain concepts related to sampling algorithms. In W. R. Gilks, S. Richardson, & D. J. Spiegelhalter (Eds.), *Markov Chain Monte Carlo in Practice* (pp. 45–58). Chapman and Hall.

Robinet, V., Lemaire, B., & Gordon, M. B. (2011). *MDLChunker: A MDL-based cognitive model of inductive learning*. *Cognitive Science* (Vol. 35). doi:10.1111/j.1551-6709.2011.01188.x

Rock, I. (1957). The role of repetition in associative learning. *The American Journal of Psychology*, *70*(2), 186–193. doi:10.2307/1419320

Rock, I., & Heimer, W. (1959). Further evidence of one-trial associative learning. *American Journal of Psychology*, *72*(1), 1–16. Retrieved from papers2://publication/uuid/FB4ACE4A-B2BF-4CB3-B4BE-9A4829CBC09D

Romberg, A. R., & Yu, C. (2014). Interactions between statistical aggregation and hypothesis testing mechanisms during word learning. In P. Bello, M. Guarini, M. McShane, & B. Scassellati (Eds.), *Proceedings of the 36th Annual Conference of the Cognitive Science Society* (pp. 1311–1316). Austin, TX: Cognitive Science Society.

Romberg, A. R., & Yu, C. (2015). Interactions between statistical aggregation and hypothesis testing mechanisms during word learning. In E. Grillo & K. Jepson

(Eds.), *Proceedings of the 39th Annual Boston University Conference on Language Development* (pp. 1311–1316). Somerville, MA: Cascadilla Press.

Roser, M. E., Fiser, J., Aslin, R. N., & Gazzaniga, M. S. (2011). Right hemisphere dominance in visual statistical learning. *Journal of Cognitive Neuroscience*, *23*(5), 1088–1099. doi:10.1162/jocn.2010.21508

Rüsseler, J., & Rösler, F. (2000). Implicit and explicit learning of event sequences: evidence for distinct coding of perceptual and motor representations. *Acta Psychologica, 104*(1), 45–67. doi:10.1016/S0001-6918(99)00053-0

Saffran, J. R. (2002). Constraints on Statistical Language Learning. *Journal of Memory and Language*, *47*, 172–196. doi:10.1006/jmla.2001.2839

Saffran, J. R., Aslin, R. N., & Newport, E. L. (1996). Statistical learning by 8-month-old infants. *Science, 274*(5294), 1926–1928. doi:10.1126/science.274.5294.1926

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

Sanchez, D. J., Gobel, E. W., & Reber, P. J. (2010). Performing the unexplainable: implicit task performance reveals individually reliable sequence learning without explicit knowledge. *Psychonomic Bulletin & Review*, *17*(6), 790–796. doi:10.3758/PBR.17.6.790

Schapiro, A., & Turk-Browne, N. (2015). Statistical Learning. In A. W. Toga (Ed.), *Brain Mapping: An Encyclopedic Reference* (Vol. 3, pp. 501–506). Academic Press: Elsevier. doi:10.1016/B978-0-12-397025-1.00276-1

Schuck, N. W., Gaschler, R., & Frensch, P. A. (2012). Implicit learning of what comes

when and where within a sequence: The time-course of acquiring serial position-item and item-item associations to represent serial order. *Advances in Cognitive Psychology*, *8*(2), 83–97. doi:10.2478/v10053-008-0106-0

Schwarb, H., & Schumacher, E. H. (2012). Generalized lessons about sequence learning from the study of the serial reaction time task. *Advances in Cognitive Psychology*, *8*(2), 165–78. doi:10.2478/v10053-008-0113-1

Shanks, D. R., Rowland, L. A., & Ranger, M. S. (2005). Attentional load and implicit sequence learning. *Psychological Research*, *69*(5-6), 369–382. doi:10.1007/s00426-004-0211-8

Shepard, R. N. (1984). Ecological Constraints on Internal Representation: Resonant Kinematics of Perceiving, Imagining, Thinking, and Dreaming. *Psychological Review*, *91*(4), 417–447. doi:10.1037/0033-295X.91.4.417

Sidman, M. (1952). A note on functional relations obtained from group data. *Psychological Bulletin*, *49*(3), 263–269. doi:10.1037/h0063643

Siegelman, N., & Frost, R. (2015). Statistical learning as an individual ability: Theoretical perspectives and empirical evidence. *Journal of Memory and Language*, *81*, 105–120.

Simon, H. A. (1974). How big is a chunk? *Science*, *183*, 482–488. doi:10.1126/science.183.4124.482

Smith, P. L., & Vickers, D. (1988). The accumulator model of two-choice discrimination. *Journal of Mathematical Psychology*, *32*(2), 135–168. doi:10.1016/0022-2496(88)90043-0

Spelke, E. S., & Kinzler, K. D. (2007). Core knowledge. *Developmental Science*.

doi:10.1111/j.1467-7687.2007.00569.x

Stone, M. (1960). Models for choice-reaction time. *Psychometrika*, *25*(3), 251–260. doi:10.1007/BF02289729

Storn, R., & Price, K. (1997). Differential Evolution -- A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *Journal of Global Optimization*, *11*(4), 341–359. doi:10.1023/A:1008202821328

Sun, R., Slusarz, P., & Terry, C. (2005). The Interaction of the Explicit and the Implicit in Skill Learning: A Dual-Process Approach. *Psychological Review*, *112*(1), 159–192. doi:10.1037/0033-295X.112.1.159

Sun, R., Zhang, X., Slusarz, P., & Mathews, R. (2007). The interaction of implicit learning, explicit hypothesis testing learning and implicit-to-explicit knowledge extraction. *Neural Networks*, *20*, 34–47. doi:10.1016/j.neunet.2006.07.002

Tenenbaum, J. B., Kemp, C., Griffiths, T. L., & Goodman, N. D. (2011). How to grow a mind: statistics, structure, and abstraction. *Science*, *331*(6022), 1279–1285. doi:10.1126/science.1192788

Thiessen, E. D., Kronstein, A. T., & Hufnagle, D. G. (2013). The Extraction and Integration Framework: A Two-Process Account of Statistical Learning. *Psychological Bulletin*, *139*(4), 792–814. doi:10.1037/a0030801

Toro, J. M., Sinnett, S., & Soto-Faraco, S. (2005). Speech segmentation by statistical learning depends on attention. *Cognition*, *97*(2), 25–34. doi:10.1016/j.cognition.2005.01.006

Trueswell, J. C., Medina, T. N., Hafri, A., & Gleitman, L. R. (2013). Propose but verify: Fast mapping meets cross-situational word learning. *Cognitive Psychology*, *66*(1),

126–156. doi:10.1016/j.cogpsych.2012.10.001

Turk-Browne, N. B. (2012). Statistical Learning and Its Consequences. In M. D. Dodd & J. H. Flowers (Eds.), *The Influence of Attention, Learning, and Motivation on Visual Search* (pp. 117–146). Springer. doi:10.1007/978-1-4614-4794-8

Turk-Browne, N. B., Jungé, J., & Scholl, B. J. (2005). The automaticity of visual statistical learning. *Journal of Experimental Psychology: General*, *134*(4), 552–564. doi:10.1167/5.8.1067

Turk-Browne, N. B., & Scholl, B. J. (2009). Flexible visual statistical learning: transfer across space and time. *Journal of Experimental Psychology: Human Perception and Performance*, *35*(1), 195–202. doi:10.1037/0096-1523.35.1.195

Turk-Browne, N. B., 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), 1934–45. doi:10.1162/jocn.2009.21131

Turner, B. M., & Sederberg, P. B. (2014). A generalized, likelihood-free method for posterior estimation. *Psychonomic Bulletin & Review*, *21*(2), 227–250. doi:10.3758/s13423-013-0530-0

Umemoto, A., Scolari, M., Vogel, E. K., & Awh, E. (2010). Statistical learning induces discrete shifts in the allocation of working memory resources. *Journal of Experimental Psychology: Human Perception and Performance*, *36*(6), 1419–1429. doi:10.1037/a0019324

Usher, M., & McClelland, J. L. (2001). The time course of perceptual choice: The leaky, competing accumulator model. *Psychological Review*, *108*(3), 550–592.

doi:10.1037/0033-295X.108.3.550

Vadillo, M. A., Konstantinidis, E., & Shanks, D. R. (2015). Underpowered samples, false negatives, and unconscious learning. *Psychonomic Bulletin & Review*, *23*, 87–102. doi:10.3758/s13423-015-0892-6

Van Zandt, T., Colonius, H., & Proctor, R. W. (2000). A comparison of two response time models applied to perceptual matching. *Psychonomic Bulletin & Review*, *7*(2), 208–256. Retrieved from papers2://publication/uuid/C7B49B0E-BB75-49CA-B00F-B4AB36FAE669

Vickers, D. (1970). Evidence for an accumulator model of psychophysical discrimination. *Ergonomics*, *13*(1), 37–58. doi:10.1080/00140137008931117

Vickers, D., & Lee, M. D. (1998). Dynamic models of simple judgements: I. Properties of a self-regulating accumulator. *Nonlinear Dynamics, Psychology, and Life Sciences*, *2*(3), 169–194. Retrieved from http://www.springerlink.com/openurl.asp?id=doi:10.1023/A:1022371901259\npapers2://publication/doi/10.1023/A:1022371901259

Wixted, J. T., & Ebbesen, E. B. (1997). Genuine power curves in forgetting: a quantitative analysis of individual subject forgetting functions. *Memory & Cognition*, *25*(5), 731–739. doi:10.3758/BF03211316

Wolpert, D. H. (1996). The Lack of A Priori Distinctions Between Learning Algorithms. *Neural Computation*, *8*(7), 1341–1390. doi:10.1162/neco.1996.8.7.1341

Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, *1*(1), 67–82. doi:10.1109/4235.585893

Yu, C., & Smith, L. B. (2012). Modeling cross-situational word–referent learning: Prior questions. *Psychological Review*, *119*(1), 21–39. doi:10.1037/a0026182

Yu, R. Q., & Zhao, J. (2015). The persistence of the attentional bias to regularities in a changing environment. *Attention, Perception, & Psychophysics*. doi:10.3758/s13414-015-0930-5

Yurovsky, D., & Frank, M. C. (2015). An Integrative Account of Constraints on Cross-Situational Learning. *Cognition*, *145*, 53–62. doi:10.1016/j.cognition.2015.07.013

Yurovsky, D., Fricker, D. C., Yu, C., & Smith, L. B. (2014). The role of partial knowledge in statistical word learning. *Psychonomic Bulletin & Review*, *21*(1), 1–22. doi:10.3758/s13423-013-0443-y

Zandbelt, B., Purcell, B. A., Palmeri, T. J., Logan, G. D., & Schall, J. D. (2014). Response times from ensembles of accumulators. *Proceedings of the National Academy of Sciences of the United States of America*, *111*(7), 2848–53. doi:10.1073/pnas.1310577111

Zhao, J., Al-Aidroos, N., & Turk-Browne, N. B. (2013). Attention Is Spontaneously Biased Toward Regularities. *Psychological Science*, *24*(5), 667–677. doi:10.1177/0956797612460407

Zhao, J., Ngo, N., McKendrick, R., & Turk-Browne, N. B. (2011). Mutual interference between statistical summary perception and statistical learning. *Psychological Science*, *22*(9), 1212–9. doi:10.1177/0956797611419304

Appendix A: Analysis Models for Experiments 2.1 and 2.2

A.1. Model for Experiment 2.1

The model treated each participant's number of correct responses, y_s, as a random draw from a binomial distribution with probability of success p_s and number of samples n_s. The number of samples was 32 for participants in the four-triples condition, and 8 for participants in the one-triple condition.

$$y_s \sim \text{binomial}(n_s, p_s)$$

The probability of success, p_s, was estimated for each participant. Each p_s was drawn from a group-level beta distribution. There were two group-level distributions, one for each condition. The group-level distribution was parameterized by the mode and concentration of the beta distribution. The primary parameters of interest are the modes for each condition and the difference in the modes.

$$p_s \sim \text{beta}(a_c, b_c)$$

$$a_c = \omega_c(\kappa_c - 2) + 1$$

$$b_c = (1 - \omega_c)(\kappa_c - 2) + 1$$

ω_c and κ_c both had vague priors appropriate to the scale of the data.

$$\omega_c \sim \text{beta}(1,1)$$

$$\kappa_c \sim \text{gamma}(\text{mode} = 1, \text{sd} = 100) + 2$$

A.2. Fitting Procedure for Experiment 2.1 Model

I estimated the posterior distributions of the model parameters with Bayesian parameter estimation methods (Kruschke, 2011), using R (R Core Team, 2012), JAGS (Plummer, 2003) and the runjags R package for MCMC sampling (Denwood, 2013, 2014). The sampling used three independent chains, with 1,000 steps of adaptation, 10,000 steps of burn-in, and 200,000 steps of sampling. The Gelman-Rubin $\hat{R}$ statistic (Gelman & Rubin, 1992) was below 1.001 for both parameters of interest, indicating acceptable convergence of the independent chains. The effective sample size (ESS; Roberts, 1996), which approximates the number of independent samples drawn from the posterior distribution by taking into account the autocorrelation of the sampler, was above 10,000 for ω_c and the difference in ω_c across conditions. 10,000 is suggested as a reasonable goal for estimating the bounds of the 95% HDI by Kruschke (2014).

A.3. Model for Estimation of Softmax Gamma Parameter.

As in the model for the experiment data, I treated each participant's number of correct responses as a random draw from a binomial distribution.

$$y_i \sim \text{binomial}(n_i, p_i)$$

The probability of success, p_i, is determined by the softmax function:

$$p_i = \frac{e^{a_i \gamma_m}}{e^{a_i \gamma_m} + e^{b_i \gamma_m}}$$

The parameter a_i represents the strength of encoding value for the target triple in cognitive model run i, and the parameter b_i represents the strength of encoding value for the foil triple in cognitive model run i. The parameter γ_m is a free parameter that controls the degree to which the difference in a_i and b_i affects the probability p_i. Higher values of γ_m cause the model to more reliably pick the option with a stronger encoding. The model fit γ_m separately for each cognitive model m. The prior on each γ_m was a uniform distribution from 0 to 100.

The data set for fitting the model consisted of each run of each cognitive model paired with each participant's data from the appropriate condition. There were 1,000 runs of each of 3 cognitive models in each condition, which were combined with the data from 41 participants, giving a total of 123,000 (1,000 x 3 x 41) participant-model combinations to fit.

A.4. Fitting the Softmax Function for Experiment 2.1

I estimated the posterior distribution of the model using the same MCMC sampling software as in section A.2. I used 3 independent chains, with 1,000 steps of adaptation, 1,000 steps of burn-in, and 4,000 steps of sampling. The $\hat{R}$ statistic was below 1.01 for all parameters.

A.5. Model for Experiment 2.2

The model for Experiment 2.2 was the same as the model for Experiment 2.1, except that the number of trials for the binomial distribution was 4 in the novel-words condition and 1 in the known-words condition. All other model details were identical.

A.6. Fitting Procedure for Experiment 2.2 Model

I used the same software as in section A.2 to fit the model. Three parallel chains sampled the posterior distribution for 200,000 iterations with 10 steps of thinning between each sample, after a burn-in period of 5,000 iterations. For all parameters of interest, $\hat{R}$ was below 1.001 and the effective sample sizes was well above 10,000.

A.7. Fitting the Softmax Function for Experiment 2.2

The same softmax rule from section A.3 was used for Experiment 2.2, with the same strategy of generating 1,000 cognitive model runs in each condition and creating the factorial combination of all participant-model combinations. The MCMC sample contained four chains with 10,000 samples each, collected after 1,000 steps of adaptation and 1,000 steps of burn-in. The ESS was well above 10,000 and $\hat{R}$ was below 1.001 for all parameters.

Appendix B: Group-level Model for Experiments 3.1 and 3.2

B.1. Model Description

The group level model describes the mean response time, *y*, in milliseconds across participants to a particular symbol within the sequence as a power law function of the number of times that symbol has appeared in the sequence. The number of appearances is analogous to time, so it will be denoted as *t*.

There are two main predictors in the model. One is the type of sequence that the symbol appeared in, here called the context, *c*, and the other is the position, *p*, that the symbol appeared in within the triplet structure. Both *c* and *p* are nominal predictors with three levels.

Each mean response time from participants in context *c* at position *p* and time *t*, $y_{c,p,t}$, is a random draw from a normal distribution with mean $\mu_{c,p,t}$ and standard deviation σ. The standard deviation is assumed to be invariant to changes in context, position, and time. σ has a vaguely informed prior:

$$\sigma \sim gamma(\text{mode} = 100, \text{standard deviation} = 250)$$

The mean of the normal distribution, $\mu_{c,p,t}$, is a power law function of three parameters:

$$\mu_{c,p,t} = \alpha_{c,p}\left(1 + \beta_{c,p}[t^{-\gamma_{c,p}} - 1]\right)$$

$\alpha_{c,p}$ describes the starting point of the curve at $t = 1$ for context c and position p. $\beta_{c,p}$ describes the overall proportion of learning for context c and position p. The curve will asymptote at $\alpha_{c,p}(1 - \beta_{c,p})$. When $\beta_{c,p}$ is 1 the power curve asymptotes at 0; when $\beta_{c,p}$ is 0 there is no change in μ as a function of t; when $\beta_{c,p}$ is 0.2 the power curve asymptotes at $\alpha_{c,p} \times 0.8$. $\gamma_{c,p}$ describes the steepness of learning, with larger values indicating more rapid change.

The three power law parameters for each combination of context and position come from a higher-level distribution for each parameter that characterizes the overall central tendency across context and position. The hierarchical structure creates shrinkage of the condition-level estimates by allowing parameter estimates to mutually inform each other across contexts and positions.

$$\alpha_{c,p} \sim gamma(\text{mode} = \alpha_{mode}, \text{standard deviation} = \alpha_{sd})$$

$$\beta_{c,p} \sim beta(\beta_{mode}[\beta_{concentration} - 2] + 1, [1 - \beta_{mode}] \times [\beta_{concentration} - 2] + 1)$$

$$\gamma_{c,p} \sim gamma(\text{mode} = \gamma_{mode}, \text{standard deviation} = \gamma_{sd})$$

The priors on the group level estimates are vaguely informed by the scale of the data.

$$\alpha_{mode} \sim normal(\text{mean} = 1000, \text{standard deviation} = 250)$$

$$\alpha_{sd} \sim gamma(\text{mode} = 100, \text{standard deviation} = 300)$$

$$\beta_{mode} \sim uniform(0,1)$$

$$\beta_{concentration} = \beta_k + 2$$

$$\beta_k \sim \exp\left(\frac{1}{10}\right)$$

$$\gamma_{mode} \sim gamma(\text{mode} = 0.1, \text{standard deviation} = 1)$$

$$\gamma_{sd} \sim gamma(\text{mode} = 0.1, \text{standard deviation} = 1)$$

B.2. Fitting Procedure for Experiment 3.1

I used JAGS, the runjags R package, and R to generate the posterior sample using MCMC methods. There were four parallel chains, each with 5,000 steps of adaptation, 10,000 steps of burn-in, and 10,000 samples, with 200 steps of thinning between each sample.

The effective sample size for all parameters was above 10,000 (smallest ESS was 12,087). $\hat{R}$ was ~ 1.0 for all parameters, indicating excellent chain mixing.

B.3. Fitting Procedure for Experiment 3.2

The additional data for Experiment 3.2 substantially increased the computational time needed to sample a sufficient number of times, so the sampling was parallelized on many nodes of a parallel computing environment using JAGS, R, and rjags (Plummer, 2014). There were 252 parallel chains in the sample, divided over 63 computing cores. Each chain had 2,000 steps of adaptation, 10,000 steps of burn-in, and 5,000 samples, with 4 steps of thinning between each sample. After sampling, the chains were further thinned to every 100 steps to reduce the disk space and memory required to store the MCMC samples. The ESS was above 10,000 for all parameters (minimum ESS = 14,478) in the final merged data set. The $\hat{R}$ diagnostic was no larger than 1.02 for all parameters.

Appendix C: Individual-level Models for Experiments 3.1 and 3.2

C.1. Model for Experiment 3.1

The individual-level model describes the response time in milliseconds separately for each participant to the letters N and W over the duration of the experiment. It also estimates some group-level parameters to describe differences in individual-level parameters across contexts. For all participants, regardless of condition, the letter N occurs at the start of a triple and W occurs at the end of the same triple. N is a relatively unpredictable symbol, while W is perfectly predictable if the participant has learned the statistical contingencies of the sequence.

The response times are modeled as a combination of two power law functions of the number of times a symbol has appeared in the sequence. As in the group level model, the number of appearances is analogous to time, and is denoted as *t*.

The first power law describes overall adaptation to the task, reflecting the general speed up in response times as participants become acclimated to the task. Changes in response time to the unpredictable element, N, over time reflect only adaptation to the task because there is no predictable structure that can be learned. The adaptation curve for N is described with three parameters: α_N, the starting point of the curve at time $t = 1$, β_{adapt} the proportion of speed-up due to adaptation, and γ_{adapt}, the steepness of adaptation**.** The mean response time predicted for N at time *t* for participant *s* is:

$$\mu_{N_s}(t) = \alpha_{N_s}\left(1 + \beta_{adapt_s}[t^{-\gamma_{adapt_s}} - 1]\right)$$

For simplicity of explanation, assume for the moment that all changes in response time during the task are due to adaptation. (This is not the case, and in a moment the model will be expanded to account for additional sources of learning.) The model assumes that the shape of the adaptation curve is consistent across both symbols. However, if adaptation alone were responsible for changes in response time, the curves for N and W may still not overlap perfectly. N and W differ in important ways: notably visual shape and the location of the appropriate key on the keyboard. Therefore, it is reasonable to expect that the curve for W may be shifted from the curve for N even if the same underlying adaptation process is governing the response time curve. To account for this possibility, the model uses a different starting point parameter, α, for each curve. For W, the adaptation curve is:

$$\mu_{W_s}(t) = \alpha_{W_s}\left(1 + \beta_{adapt_s}\left[t^{-\gamma_{adapt_s}} - 1\right]\right)$$

The actual raw response times, y, are assumed to be log-normally distributed around the means for W and N. The log-normal distribution is used to account for the skew of raw response time data distributions. (In practice, using a normal distribution makes very little difference in the parameter estimates of the model).

$$y_{N_s}(t) \sim lognormal\left(a_{N_s}(t), b_{N_s}(t)\right)$$

$$y_{W_s}(t) \sim lognormal\left(a_{W_s}(t), b_{W_s}(t)\right)$$

To make the location, a, and scale, b, parameters of the log-normal distribution interpretable on the scale of the original data, the parameters of the log-normal distribution are transformed to the mean and standard deviation, σ, of the non-log data.

$$a_{N_s}(t) = \log\left(\frac{\mu_{N_s}(t)^2}{\sqrt{\sigma^2 + \mu_{N_s}(t)^2}}\right) \qquad a_{W_s}(t) = \log\left(\frac{\mu_{W_s}(t)^2}{\sqrt{\sigma^2 + \mu_{W_s}(t)^2}}\right)$$

$$b_{N_s}(t) = \frac{1}{\log\left(1 + \frac{\sigma^2}{\mu_{N_s}(t)^2}\right)} \qquad b_{W_s}(t) = \frac{1}{\log\left(1 + \frac{\sigma^2}{\mu_{W_s}(t)^2}\right)}$$

Changes in response time to the letter W reflect only task adaptation if the participant has not learned the statistical structure. However, learning the statistical structure of the sequence may also cause changes in response time to W that are in addition to the overall adaptation. The model allows for this additional learning by fitting a second power law. This power law, referred to as the learning curve, is characterized by three parameters. The first is an offset parameter, ω, which describes the time at which statistical learning, above and beyond adaptation, begins to cause changes in the response time. Response times that occur at times $t < \omega$ are fit using only the adaptation curve. Response times that occur at times $t \geq \omega$, are fit using a combination of the adaptation curve and learning curve. The second is a proportion of learning parameter, β_{learn}, which reflects the asymptote of learning relative to the adaptation curve. For example, if this

parameter is 0.25 and the adaptation curve asymptotes at 800ms, then the learning curve will asymptote at 600ms. The second parameter, γ_{learn}, is the steepness of learning.

A Boolean parameter π_s determines whether the individual exhibited any learning beyond adaptation.

$$\pi_s \sim Bernoulli(\theta_c)$$

This parameter is drawn from a Bernoulli distribution with an overall probability θ_c, which is estimated separately for participants in each learning context, *c*.

$$\theta_c \sim uniform(0, 1)$$

If π_s is 0, then no learning is included beyond adaptation. If π_s is 1, then the mean of the response time distribution at time *t* for the symbol W is expressed as a combination of the adaptation and learning curves. Putting all of that together, the mean response time for W at time *t* is:

$$A = \alpha_{W_s}\left(1 + \beta_{adapt_s}[t^{-\gamma_{adapt_s}} - 1]\right)$$

$$\mu_{W_s}(t) = \begin{cases} A, & t < \omega_s \parallel \pi_s = 0 \\ A \times (1 + \beta_{learn_s}[(t - \omega_s + 1)^{-\gamma_{learn_s}} - 1]), & t \geq \omega_s \end{cases}$$

The offset parameter, ω_s, is drawn from a Poisson distribution. The shape parameter of the Poisson is drawn from a group-level distribution, with a separate distribution for each learning context, *c*. The prior on the shape parameter, which is also

the mode of the Poisson distribution, is a uniform distribution covering all possible values of *t* in the experiment.

$$\omega_s \sim Poisson(\varphi_c)$$

$$\varphi_c \sim uniform(0,36)$$

The parameters for the adaptation and learning curves are all estimated using a hierarchical framework with participant-level estimation of the individual curves and a group-level distribution describing overall adaptation and learning (when it exists). The group-level distributions are estimated without taking learning context into account, under the assumption that adaptation and learning are likely to be similar across contexts due to the fact that the triple being learned is identical in each context. The group-level distributions are included in the model to provide shrinkage on the individual-level estimates.

$$\alpha_{adapt_s} \sim normal\left(\alpha_{adapt_{g,mean}}, \alpha_{adapt_{g,sd}}\right), \qquad \alpha_{adapt_s} \in (0.01,2000)$$

$$\beta_{adapt_s} \sim beta\left(\beta_{adapt_A}, \beta_{adapt_B}\right)$$

$$\gamma_{adapt_s} \sim gamma\left(\text{mode} = \gamma_{adapt_{g,mode}}, \text{sd} = \gamma_{adapt_{g,sd}}\right)$$

$$\beta_{learn_s} \sim beta\left(\beta_{learn_A}, \beta_{learn_B}\right)$$

$$\gamma_{learn_s} \sim gamma\left(\text{mode} = \gamma_{learn_{g,mode}}, \text{sd} = \gamma_{learn_{g,sd}}\right)$$

In all cases, the priors on the group-level distributions are vaguely informed by the scale of the data.

$$\alpha_{adapt_{g,mean}} \sim normal(1000,250), \qquad \alpha_{adapt_{g,mean}} \in (0,2000)$$

$$\alpha_{adapt_{g,sd}} \sim gamma(\text{mode} = 20, \text{sd} = 200)$$

$$\beta_{adapt_A} = \beta_{adapt_{g,mode}} \times \left(\beta_{adapt_{g,concentration}} - 2\right) + 1$$

$$\beta_{adapt_B} = \left(1 - \beta_{adapt_{g,mode}}\right) \times \left(\beta_{adapt_{g,concentration}} - 2\right) + 1$$

$$\beta_{adapt_{g,mode}} \sim beta(1.75,3.25)$$

$$\beta_{adapt_{g,concentration}} = 2 + \beta_{adapt_{g,concentration_k}}$$

$$\beta_{adapt_{g,concentration_k}} \sim gamma(\text{mode} = 5, \text{sd} = 20)$$

$$\gamma_{adapt_{g,mode}} \sim gamma(\text{mode} = 0.1, \text{sd} = 10)$$

$$\gamma_{adapt_{g,sd}} \sim gamma(\text{mode} = 0.1, \text{sd} = 10)$$

$$\beta_{learn_A} = \beta_{learn_{g,mode}} \times \left(\beta_{learn_{g,concentration}} - 2\right) + 1$$

$$\beta_{learn_B} = \left(1 - \beta_{learn_{g,mode}}\right) \times \left(\beta_{learn_{g,concentration}} - 2\right) + 1$$

$$\beta_{learn_{g,mode}} \sim beta(1.75,3.25)$$

$$\beta_{learn_{g,concentration}} = 2 + \beta_{learn_{g,concentration_k}}$$

$$\beta_{learn_{g,concentration_k}} \sim gamma(\text{mode} = 5, \text{sd} = 20)$$

$$\gamma_{learn_{g,mode}} \sim gamma(\text{mode} = 0.1, \text{sd} = 10)$$

$$\gamma_{learn_{g,sd}} \sim gamma(\text{mode} = 0.1, \text{sd} = 10)$$

Finally, the standard deviation of the distribution of raw response times around the means estimated by the power curves is assumed to be consistent across all participants.

$$\sigma \sim gamma(\text{mode} = 200, \text{sd} = 500)$$

C.2. Modifications for Experiment 3.2

The model for Experiment 3.2 was very similar to the model for Experiment 3.1, with a few small modifications.

The prior on learning onsets was changed to reflect the length of the task:

$$\varphi_c \sim uniform(0,72)$$

Instead of using a single group-level distribution for the parameters that describe the shape of the learning curve, these parameters had context-level distributions. The participant-level estimate was drawn from a distribution that was specific to the particular context *c* of that participant:

$$\beta_{learn_{s,c}} \sim beta\left(\beta_{learn_A,c}, \beta_{learn_B,c}\right)$$

$$\gamma_{learn_{s,c}} \sim gamma\left(\text{mode} = \gamma_{learn_{c,mode}}, \text{sd} = \gamma_{learn_{c,sd}}\right)$$

The priors on the context-level distributions were identical to the priors on the group-level distributions in the model for Experiment 3.1.

C.3. Fitting Procedure for Experiments 3.1 and 3.2

I fit the models with MCMC methods using the same parallelized approach as in section C.2. 127 parallel chains were used. Each chain had 2,000 steps of adaptation, 10,000 steps of burn-in, and 5,000 sampling steps with 4 steps of thinning. After sampling, each chain was further thinned to reduce the disk size of the posterior sample. For Experiment 3.1, the sample was thinned to every 100 steps. For Experiment 3.2, the sample was thinned to every 40 steps. The difference in thinning was required to keep the effective sample size (ESS) above 10,000 for the majority of parameters.

For Experiment 3.1, the ESS of all but 15 of the 1,145 model parameters was above 10,000. The 15 that were below 10,000 were all π_s individual-level parameters for participants that were robustly classified as learners or non-learners, making these chains essentially deterministic.

For Experiment 3.2, of the 2,089 model parameters, the ESS was above 10,000 for 2,012 of them. Of the remaining 77, all but 2 were either ω_s or π_s individual-level parameters for participants that were robustly classified as learners or non-learners, creating no variability in their samples which makes the ESS impossible to estimate. The ESS for γ_{group} was 9,106, and the ESS for participant 250's β_{learn_s} parameter was 9,807.

Appendix D: Multiple-onset Model for Experiment 3.2.

D.1. Model Description

The multiple-onset model is based on the individual-level model from section C.2, with a few modifications to fit response time curves to all of the different letters instead of just two particular letters. As in the individual-level model, the response times for letter *L* were generated by a lognormal distribution with the mean of the distribution described by one or two power curves, depending on whether the individual was classified as a learner or not.

$$y_L(t) \sim lognormal\big(a_L(t), b_L(t)\big)$$

$$a_L(t) = \log\left(\frac{\mu_L(t)^2}{\sqrt{\sigma^2 + \mu_L(t)^2}}\right)$$

$$b_L(t) = \frac{1}{\log\left(1 + \frac{\sigma^2}{\mu_L(t)^2}\right)}$$

$$\sigma \sim gamma(\text{mode} = 200, \text{sd} = 500)$$

$$A_L = \alpha_L\big(1 + \beta_{adapt}[t^{-\gamma_{adapt}} - 1]\big)$$

$$\mu_L(t) = \begin{cases} A_L, & t < \omega_L \parallel \pi = 0 \parallel L \in U \\ A_L \times (1 + \beta_{learn_L}[(t - \omega_L + 1)^{-\gamma_{learn_L}} - 1]), & t \geq \omega_L \end{cases}$$

$$U = \{'N', 'T', 'R', 'Y'\}$$

In this model, all 12 letters are described by their own curves. The start point parameter for the adaptation curve is fit separately for each letter *L*.

$$\alpha_{adapt_L} \sim normal(1000, 250), \qquad \alpha_{adapt_L} \in (0, 2000)$$

The other adaptation parameters are shared across all letters, as in the individual-level model.

$$\beta_{adapt} \sim beta(1.75, 3.25)$$

$$\gamma_{adapt} \sim gamma(\text{mode} = 0.1, \text{sd} = 2)$$

Parameters for the second power curve and the onset of learning are fit for letters that are predictable – occupying either the 2nd or 3rd spot in a triple. These parameters are fit individually for each letter with no group-level distributions.

$$\beta_{learn_L} \sim beta(1.75, 3.25)$$

$$\gamma_{learn_L} \sim gamma(\text{mode} = 0.1, \text{sd} = 2)$$

$$\omega_L \sim categorical\left(\frac{1}{72}, \frac{1}{72}, \ldots, \frac{1}{72}\right)$$

Finally, an overall estimate of the probability that the participant exhibited any learning beyond adaptation is drawn from a Bernoulli distribution.

$$\pi \sim Bernoulli(0.5)$$

D.2. Model Fitting

The model was run separately for each participant, in part to avoid the enormous computational complexity of fitting all participants simultaneously. However, because the goal of the model is to find the onsets of learning for individual participants,

including group-level constraints would create unwanted shrinkage on the estimates in this case.

The model was fit using JAGS, R, and rjags. Each individual run used 4 parallel chains, with 2,000 steps of adaptation and burn-in, and 10,000 steps of sampling with 20 steps of thinning between each sample. The ESS was *not* above 10,000 due to high autocorrelation, but fewer samples are needed to get a reliable estimate of the posterior when the goal of the model is to estimate the mean onset of learning and not the bounds of the HDI (Kruschke, 2014).

Joshua R. de Leeuw
Curriculum Vitae

Professional Appointment

2016-	Assistant Professor, Vassar College, Poughkeepsie, NY

Education

2016	Ph.D., Indiana University, Cognitive Science and Psychological & Brain Sciences
2008	B.A., Vassar College, Cognitive Science General Honors & Departmental Honors in Cognitive Science

Honors & Awards

2016	The 2016 Cognitive Science Teaching Award, Cognitive Science Program, Indiana University
2013	Commendation on PhD qualification exam, Department of Psychological & Brain Sciences, Indiana University
	IGERT program representative for national poster contest
2012	Outstanding Instructor Award, Department of Psychological & Brain Sciences, Indiana University
2010	NSF Graduate Research Fellowship
	IGERT Training Program Fellowship in Brain-Body-Environment Systems.
2009	2nd place in Microsoft RoboChamps Challenge, robotics competition ($10,000 prize)
2008	Jean Slator Edson Prize for best original music composition, Vassar College
	Phi Beta Kappa
	Sigma Xi
	Psi Chi

Publications & Presentations

Journals

2016 de Leeuw, J. R., Andrews. J. K., Livingston, K. R., & Chin., B. M. (2016). The effects of categorization on perceptual judgment are robust across different assessment tasks. *Collabra*.

Carvalho, P., Braithwaite, D., de Leeuw, J. R., Motz, B., & Goldstone, R. L. (2016). An in-vivo study of self-regulated study sequencing in introductory psychology courses. *PLoS ONE, 11*(3): e0152115. doi:10.1371/journal.pone.0152115

de Leeuw, J. R., & Motz, B. A. (2016). Psychophysics in a web browser? Comparing response times collected with JavaScript and Psychophysics Toolbox in a visual search task. *Behavior Research Methods, 48*(1), 1-12.

2015 Goldstone, R. L., de Leeuw, J. R., & Landy, D. H. Carving nature at its joints or cutting its effective loops?: On the dangers of trying to disentangle intertwined mental processes. Commentary on "Cognition does not affect perception: Evaluating the evidence for 'top-down' effects." *Behavioral and Brain Sciences.* In press.

Goldstone, R. L., de Leeuw, J. R., & Landy, D. H. (2015). Fitting perception in and to cognition. *Cognition, 135*, 24-29. doi:10.1016/j.cognition.2014.11.027

de Leeuw, J. R. (2015). jsPsych: A JavaScript library for creating behavioral experiments in a Web browser. *Behavior Research Methods, 47*(1), 1-12. doi:10.3758/s13428-014-0458-y

2011 Long, J.H. Jr., Krenitsky, N. M., Roberts, S. F., Hirokawa, J., de Leeuw, J. R., and Porter, M. E. (2011). Testing biomimetic structures in bioinspired robots: how vertebrae control the stiffness of the body and the behavior of fish-like swimmers. *Integrative and Comparative Biology, 51*(1), 158-175.

Refereed Conference Proceedings

2015 de Leeuw, J. R., & Goldstone, R. L. (2015). Memory constraints affect statistical learning; Statistical learning affects memory constraints. In Noelle, D. C., Dale, R., Warlaumont, A. S., Yoshimi, J., Matlock, T., Jennings, C. D., & Maglio, P. P. (Eds.), *Proceedings of the 37th Annual Meeting of the Cognitive Science Society* (pp. 530-535). Austin, TX: Cognitive Science Society.

de Leeuw, J. R., & Andrews, J. (2015). Using a task-filled delay during discrimination trials to examine different components of learned visual categorical perception. In Noelle, D. C., Dale, R., Warlaumont, A. S., Yoshimi, J., Matlock, T., Jennings, C. D., & Maglio, P. P. (Eds.), *Proceedings of the 37th Annual Meeting of the Cognitive Science Society* (pp. 536-541). Austin, TX: Cognitive Science Society.

Carvalho, P., Braithwaite, D., de Leeuw, J. R., Motz, B., & Goldstone, R. L. (2015). Effectiveness of learner-regulated study sequence: An in-vivo study in Introductory Psychology courses. In Noelle, D. C., Dale, R., Warlaumont, A. S., Yoshimi, J., Matlock, T., Jennings, C. D., & Maglio, P. P. (Eds.), *Proceedings of the 37th Annual Meeting of the Cognitive Science Society* (pp. 309-314). Austin, TX: Cognitive Science Society.

de Leeuw, J. R., Motz, B. A., Eastwood, J. L., Maltese, A. V., Goldstone, R. L., Danish, J. A. (2015). Needle in the neural haystack: Electroencephalograph signatures of concept learning while viewing naturalistic educational materials. *Proceedings of the 2015 American Educational Research Association Annual Meeting*.

2014 de Leeuw, J. R., Andrews, J., & Livingston, K. (2014). Learned visual categorical perception effects depend on method of assessment and stimulus discriminability. In P. Bello, M. Guarini, M. McShane, & B. Scassellati (Eds.), *Proceedings of the 36th Annual Conference of the Cognitive Science Society* (pp. 391-396). Austin, TX: Cognitive Science Society.

2010 Krishnamurthy, P., Khorrami, F., de Leeuw, J. R., Porter, M. E., Livingston, K., & Long, J. H. (2010). An electric ray inspired biomimetic autonomous underwater vehicle. In *Proceedings of the American Control Conference* (pp. 5224-5229).

2009 de Leeuw, J. R., & Livingston, K. (2009). A self-organizing autonomous prediction system for controlling mobile robots. In Chen, K., Moustafa, K. A. F., and Karras, D. A., editors, *Proceedings of the International Conference on Automation, Robotics and Control Systems (ARCS-09)*, pages 123-129. ISRST.

Krishnamurthy, P., Khorrami, F., de Leeuw, J. R., Porter, M. E., Livingston, K., & Long, J. H. (2009). A multi-body approach for 6dof modeling of biomimetic autonomous underwater vehicles with simulation and experimental results. In *Control Applications, (CCA) & Intelligent Control, (ISIC), 2009 IEEE*, pages 1282-1287.

2007 de Leeuw, J. R., & Livingston, K. (2007). When less is more: Sensor resolution and learning. In Berthouze, L., Prince, C. G., Littman, M.,

Kozima, H., and Balkenius, C., editors, *Proceedings of the Seventh International Conference on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems.*

Workshops & Tutorials

2015 de Leeuw, J. R. (2015). Using jsPsych to conduct behavioral research online. *Workshop in Methods Series at Indiana University.*

de Leeuw, J. R. (2015). Programming online experiments with jsPsych. In Noelle, D. C., Dale, R., Warlaumont, A. S., Yoshimi, J., Matlock, T., Jennings, C. D., & Maglio, P. P. (Eds.), *Proceedings of the 37th Annual Meeting of the Cognitive Science Society* (pp. 13-14). Austin, TX: Cognitive Science Society.

2014 de Leeuw, J. R., Coenen, A., Markant, D., Martin, J. B., McDonnell, J., Rich, A., & Gureckis, T. (2014). Online experiments using jsPsych, psiTurk, and Amazon Mechanical Turk. In P. Bello, M. Guarini, M. McShane, & B. Scassellati (Eds.), *Proceedings of the 36th Annual Conference of the Cognitive Science Society* (pp. 42-43). Austin, TX: Cognitive Science Society.

2007 Livingston, K., & de Leeuw, J. R. (2007). Virtual bottom-up robotics lab with physical robot component. Workshop on Interactive Computer-Based Activities for Undergraduate Cog Sci Instruction at the 29th Annual Meeting of the Cognitive Science Society.

Invited Talks

2016 Using jsPsych to conduct behavioral research online. University of Wisconsin, Madison. February 24, 2016.

2015 Doing Social Science Online: A Research Ethics Discussion with the Creator of jsPsych. Poynter Center for the Study of Ethics and American Institutions, Indiana University. October 14, 2015.

Conference Presentations

2016 Slone, L. K., & de Leeuw, J. R. (2016). Modeling the factors underlying adults' confidence judgments in a visual statistical learning task. Poster presented at the Fifth Implicit Learning Seminar, Lancaster University, UK.

2015 de Leeuw, J. R. (2015). A collaborative, open-source collection of browser-based experiments for teaching demonstrations using jsPsych. Poster presented at the Annual Meeting of the Society for Computers in Psychology, Chicago, IL.

de Leeuw, J. R., & Goldstone, R. L. (2015). Statistical learning of regularities reduces memory constraints on statistical learning. Poster presented at the Annual Meeting of the Psychonomic Society, Chicago, IL.

Eastwood, J., Maltese, A., de Leeuw, J. R., Danish, J., Goldstone, R. L., & Motz, B. (2015). Exploring the inner-working of anatomy learning: an interdisciplinary approach. Poster presented at the Annual Meeting of the American Association of Anatomists, Boston, MA.

2014 de Leeuw, J. R., & Goldstone, R. L. (2014). Context effects in visual statistical learning. Poster presented at the Annual IGERT in Brain-body-environment Systems Symposium, Bloomington, IN.

Braithwaite, D. W., Carvalho, P. F., de Leeuw, J. R., & Goldstone, R. L. (2014). Effectiveness of learner-regulated study sequence. Poster presented at the 55th Annual Meeting of the Psychonomic Society, Long Beach, CA.

Carvalho, P. F., Braithwaite, D. W., de Leeuw, J. R., Motz, B. A., & Goldstone, R. L. (2014). Sequencing examples during concept learning. Poster presented at the 2014 CIRCLE Conference - Integrating cognitive science with innovative teaching in STEM disciplines, St. Louis, MO.

de Leeuw, J. R., & Goldstone, R. L. (2014). Predictable sequences promote the learning of visual statistical regularities. Presented at the 4th Annual Midwest Cognitive Science Conference, Dayton, OH.

2013 de Leeuw, J. R., Andrews, J., & Livingston, K. (2013). Variables influencing the nature of learned categorical perception effects. Poster presented at the 35th Annual Meeting of the Cognitive Science Society, Berlin, Germany.

de Leeuw, J. R., & Todd, P. M. (2013). The influence of hunger on categorical perception of food and non-food items. Presented at the 3rd Annual Midwest Cognitive Science Conference, Columbus, OH.

de Leeuw, J. R. (2013). Common goals coordinate groups of asocial embodied robots. Video and poster presented as part of the 2013 IGERT National Poster & Video Competition, online at http://posterhall.org/igert2013/

de Leeuw, J. R., & Todd, P. M. (2013). Meat-O-Vision: Testing a literary trope in the lab. Poster presented at the Annual IGERT in Brain-body-environment Systems Symposium, Bloomington, IN.

	de Leeuw, J. R., Livingston, K. R., Porter, M. E., & Long, J. H. Jr. (2013). When swarm intelligence isn't: Common goals alone explain emergence of group coordination in asocial embodied robots. Presented at the Society for Integrative and Comparative Biology, San Francisco, CA.
2012	de Leeuw, J. R., Livingston, K., Porter, M. E., & Long, J.H. Jr. (2012). Common goals coordinate groups of asocial embodied robots. Poster presented at the Annual IGERT in Brain-body-environment Systems Symposium, Bloomington, IN.
2011	de Leeuw, J. R. (2011). Testing selection pressures for small-world neural networks in virtual robots. Poster presented at the Annual IGERT in Brain-body-environment Systems Symposium, Bloomington, IN.
2010	de Leeuw, J. R., Porter, M., Livingston, K., & Long, J.H. Jr. (2010). Evolving intelligence in autonomous, fish-like biorobots: does competition for resources matter? Poster presented at the Society for Integrative and Comparative Biology, Seattle, WA.
	Hirokawa, J., Roberts, S., Frias, C., Krenitsky, N., de Leeuw, J. R., Long, J.H., Jr., & Porter, M. E.. (2010). A self-propelled robotic swimmer as a biomechanical testbed: how swimming performance is modulated by the axial length of the intervertebral joints in a biomimetic vertebral column. Poster presented at the Society for Integrative and Comparative Biology, Seattle, WA.

Teaching Experience

Instructor of Record

2015-2016	Statistical Techniques, Indiana University
2012	Methods of Experimental Psychology Lab, Indiana University

Lab Instructor

2014	Prediction, Probability, and Pigskin, Indiana University

Substitute Instructor

2015	Cognitive Psychology, DePauw University (4 class sessions)
	Computational Neuroscience, DePauw University (4 class sessions)

Professional Service

Consulting Editor	Behavior Research Methods
Ad-hoc reviewer	Cognitive Psychology, Cognition, PLoS ONE, Psychonomic Bulletin and Review, Quarterly Journal of Experimental Psychology, Cognitive Science Society