UNIVERSIDAD DE BUENOS AIRES

FACULTAD DE CIENCIAS EXACTAS Y NATURALES

Departamento de Computación

Aprendizaje automático a partir de cuerpos de datos ralos: un enfoque basado en la inferencia bayesiana

Tesis presentada para optar por el título de Doctor de la Universidad de Buenos Aires en el área Ciencias de la Computación

SERGIO ROMANO

Director de Tesis: Santiago Figueira

Co-Director de Tesis: Mariano Sigman

Consejero de Estudios: Diego Fernández Slezak

Lugar de trabajo: Instituto de Ciencias de la Computación, UBA / CONICET.

Buenos Aires, 2021

Esta tesis está dedicada a...

Agradecimientos

En este trabajo agradezco a...

Resumen

Por acá

Palabras Clave:

Abstract

Índice

Capítulo 1

Introducción

En las últimas dos décadas distintas técnicas de ingeniería reversa del aprendizaje en humanos han inspirado con éxito distintos algoritmos de inteligencia artificial [Russell and Norvig, 2002]. Los avances recientes en las técnicas de aprendizaje profundo han logrado resultados notables en numerosos dominios como reconocimiento visual de objetos, el reconocimiento automático del habla, la búsqueda de respuestas y las traducciones automáticas [LeCun et al., 2015]. En la mayoría de estos enfoques, el resultado y el objeto del proceso de aprendizaje es una función estadística de reconocimiento de patrones específicos en los datos. Sin embargo, en muchas situaciones, el aprendizaje humano implica la construcción de modelos estructurados de conocimiento abstracto a partir de pocos datos, y este tipo de sistemas no han sido capaces de imitar esa habilidad [Lake et al., 2017].

¿Cómo pueden las personas adquirir un vasto universo de conceptos con muy poca exposición aparente? Una posible solución a este enigma, conocida como el problema de Platón [Chomsky, 1986, Chomsky et al., 2006], surge del aprendizaje automático probabi-

lístico. Este enfoque está arrojando algo de luz sobre cómo los humanos pueden construir modelos y abstracciones bajo incertidumbre y a partir de datos escasos [Tenenbaum et al., 2011, Ghahramani, 2015], y está renovando la hipótesis de Jerry Fodor que afirma que el pensamiento toma forma en una especie de lenguaje mental del pensamiento (LoT, por sus siglas en inglés) compuesto por un conjunto limitado de símbolos atómicos que se pueden combinar para formar estructuras más complejas siguiendo reglas combinatorias [Fodor, 1975].

Nuestra investigación se suscribe a uno de las líneas actuales del aprendizaje automático probabilístico conocido como programación probabilística, un esquema general para expresar modelos probabilísticos y métodos de inferencia como programas informáticos [Ghahramani, 2015]. Esto significa que en nuestros modelos asumimos que el LoT es un lenguaje de programación capaz de generar programas para modelar conceptos en el mundo. Con nuestro trabajo pretendemos mejorar nuestro entendimiento del proceso de aprendizaje a partir de cuerpos ralos de datos y desarrollar nuevos métodos y algoritmos de programación probabilística para replicar esta notable capacidad humana.

En la sección..... (acá se explicaría capítulo por capítulo)

1.1. Teoría computacional de la mente

Explicar critica conexionismo y surgimiento teoría computacional de la mente. Breve mención a críticas del conectivismo a partir del 80 (Fodor y Steven Pinker). Reversión al asociacionismo.

Explicar Simbólico

"The infite use of finite means"(Humboldt's sobre el lenguaje)

1) How does abstract knowledge guide learning and inference from sparse data? Bayesian inference in probabilistic generative models

2) What form does that knowledge take, across different domains and tasks? Probabilities defined over richly structured symbolic representations: spaces, graphs, grammars, logical predicates

3) How is that knowledge itself constructed / updated / validated? Hierarchical models, transfer learning, herramientas papers

Los investigadores han modelado estas categorías mentales o clases conceptuales con dos enfoques clásicos: en términos de similitud con un ejemplo genérico o prototipo [Rosch, 1999, Nosofsky, 1986, Rosch et al., 1976, Rosch and Mervis, 1975] o basándose en una representación simbólica de reglas [Boole, 1854, Fodor, 1975, Gentner, 1983].

Enfoques simbólicos como la hipótesis del *lenguaje del pensamiento* (LoT, por sus siglas en ingés) [Fodor, 1975], afirman que el pensamiento toma forma en una especie de lenguaje mental, compuesto por un conjunto limitado de símbolos atómicos que pueden combinarse para formar estructuras más complejas. siguiendo reglas combinatorias.

Explicar críticas a simbólico [Blackburn, 1984, Loewer and Rey, 1991, Knowles, 1998, Aydede, 1997]

1.2. Lenguaje del pensamiento

Profundizar Fodor y Cognición. Aparición de Probabilistic Language of Thought

A pesar de las críticas y objeciones, los enfoques simbólicos en general — y la hipótesis de LoT en particular — han ganado una atención renovada con resultados recientes que podrían explicar el aprendizaje a través de diferentes dominios como inferencia estadística sobre un espacio de hipótesis estructurado composicionalmente [Tenenbaum et al., 2011, Piantadosi and Jacobs, 2016].

El LoT no es necesariamente único. De hecho, la forma que adopta se ha modelado de muchas formas diferentes según el dominio del problema: aprendizaje de conceptos numéricos [Piantadosi et al., 2012], aprendizaje de secuencias [Amalric et al., 2017a, Yildirim and Jacobs, 2015, Romano et al., 2013], aprendizaje de conceptos visuales [Ellis et al., 2015], aprendizaje de teorías [Ullman et al., 2012], etc.

Si bien los marcos pueden diferir en cómo se puede implementar un LoT computacionalmente, todos comparten la propiedad de estar construidos a partir de un conjunto de símbolos y reglas atómicos mediante los cuales se pueden combinar para formar expresiones nuevas y más complejas.

La mayoría de los estudios de LoT se han centrado en el aspecto compositivo del lenguaje, que se ha modelado dentro de un [Tenenbaum et al., 2011] bayesiano o un marco [Amalric et al., 2017a, Goldsmith, 2002, Romano et al., 2013, Goldsmith, 2001] de longitud mínima de descripción (MDL).

El método común es definir una gramática con un conjunto de producciones basadas en

operaciones que son intuitivas para los investigadores y luego estudiar cómo diferentes procesos de inferencia coinciden con patrones regulares en el aprendizaje humano. Un estudio reciente [Piantadosi et al., 2016] pone el foco en el proceso de cómo elegir empíricamente el conjunto de producciones y cómo diferentes definiciones de LoT pueden crear diferentes patrones de aprendizaje.

1.2.1. Gramáticas

Explicar gramáticas

Explicar diferencia entre sintaxis y semántica

El proyecto de análisis bayesiano del aprendizaje de conceptos de modelos LoT utilizando inferencia bayesiana en un espacio de hipótesis estructurado gramaticalmente [Goodman et al., 2008]. Cada propuesta de LoT suele formalizarse mediante una gramática libre de contexto $\mathcal{G}$ que define las funciones o programas válidos que se pueden generar, como en cualquier otro lenguaje de programación. Un programa es un árbol de derivación de $\mathcal{G}$ que debe interpretarse o ejecutarse de acuerdo con una semántica determinada para obtener una descripción real del concepto en la tarea cognitiva en cuestión. Por lo tanto, cada concepto es luego representado por cualquiera de los programas que lo describen y se define un proceso de inferencia bayesiano para inferir de los datos observados la distribución de programas válidos en $\mathcal{G}$ que describen los conceptos.

1.2.2. Composición

Los lenguajes combinatorios pueden describir un vasto conjunto de conceptos a partir de un pequeño conjunto de primitivas. Esto se puede entender en un ejemplo relativamente simple en el dominio de las formas. Un lenguaje combinatorio y simbólico similar a Logo [Abelson et al., 1974] puede combinar operaciones como "mover", "pluma arriba", "pluma abajo.º rotar"para generar un conjunto infinito de expresiones (o programas) que, cuando se evalúa, puede transmitir todo tipo de formas.

Un lenguaje que describe conceptos (como formas) también proporciona una noción natural de su complejidad [Kolmogorov, 1968]. Un concepto es simple, relativo a ese lenguaje, cuando puede describirse mediante un programa corto. Por el contrario, es complejo cuando todas sus descripciones requieren una larga secuencia de instrucciones. Por ejemplo, en el caso del lenguaje Logo, un cuadrado puede simplemente instruirse como un bucle de cuatro desplazamientos seguidos de rotaciones de 90 grados. En este lenguaje, el icono de un rostro se implementará mediante un programa mucho más largo y, por lo tanto, será más complejo. Sin embargo, este concepto sería más sencillo cuando se describiera en un lenguaje en el que el icono de un rostro (o los símbolos de nariz, boca, etc.) estén disponibles como primitivos en el lenguaje.

En el dominio de los conceptos booleanos, se estudió una amplia gama de variedades lógicas de conceptos en [Feldman, 2003], revelando una ley sorprendentemente simple: la dificultad subjetiva de un concepto booleano para un aprendiz humano es directamente proporcional a la longitud del programa compatible más corto en el lenguaje de la lógica proposicional (es decir, variables booleanas combinadas con los operadores *and*, *or* y *not*). Este resultado puede sugerir que el LoT humano está equipado con reglas y símbolos

similares a los que se encuentran en la lógica proposicional. De hecho, la correlación entre la dificultad subjetiva de los conceptos y su complejidad se ha utilizado como vehículo general para estudiar el LoT humano en varios dominios [Piantadosi et al., 2016, Leeuwenberg, 1971, Amalric et al., 2017b, Romano et al., 2018, Lupyan et al., 2007]. Aunque a menudo está implícito, la estrategia general es *(1)* asumir un idioma; *(2)* encontrar el programa compatible más corto para algunos conceptos en ese idioma; *(3)* comparar la duración de estos programas con la dificultad subjetiva de los conceptos; y finalmente *(4)* repetir este proceso para varios idiomas dentro de un universo de posibles candidatos y elegir el idioma que mejor se ajuste en *(3)*. Como se mencionó anteriormente, la longitud del programa depende de las primitivas del lenguaje en el que está escrito este programa, por lo que diferentes lenguajes hacen diferentes predicciones.

Una pregunta natural, sin embargo, es si las primitivas de una LoT son universales –tanto a través de diferentes individuos como a lo largo del desarrollo– o si, en cambio, el repertorio semántico de un lenguaje es dinámico y está moldeado por la experiencia. De hecho, es probable que nuestra capacidad para representar automáticamente conceptos booleanos de manera sucinta no se deba a un lenguaje proposicional eficiente innato en nuestra mente. En cambio, proponemos que esta capacidad surge como un subproducto de nuestro cerebro que aprende rápidamente representaciones eficientes de los conceptos que generalmente encontramos en la vida cotidiana.

Nuestra pregunta de investigación es: ¿con qué rapidez podemos adaptar nuestros mecanismos de aprendizaje cuando nos encontramos con un nuevo dominio en el que nuestras representaciones a priori ya no son eficientes? Examinamos la hipótesis de que los humanos tienen la capacidad de recombinar rápidamente proposiciones en su LoT, agregando nuevas primitivas a su lenguaje. En otras palabras, ese aprendizaje conduce a un

proceso de compilación de rutinas en funciones dentro de el LoT.

En el ejemplo del lenguaje Logo se puede imaginar que si las producciones que dibujan cuadrados son muy frecuentes, sería eficaz dedicar un nuevo símbolo a esta producción. El nuevo símbolo çuadrado.[es] una construcción jerárquica de "segundo orden"de las primitivas de "primer orden"del lenguaje. Tiene un costo (de incrementar el léxico del lenguaje) pero en el nuevo lenguaje, dibujar un cuadrado puede ser instanciado con un programa muy corto (es decir, çuadrado") y por lo tanto usa menos memoria. De hecho, un lenguaje de nivel superior nos permite alcanzar un nivel superior de abstracción al liberar la memoria y el poder de procesamiento, haciendo así pensables pensamientos más complejos [Minsky, 1967, Murphy, 1988].

La mayor parte del trabajo en la literatura sobre LoT, aunque incluye naturalmente un mecanismo de aprendizaje, tiende a acercarse al LoT como un sistema estable que deben descubrir los experimentadores, que prueban diferentes plantillas candidatas y seleccionan la que mejor se ajusta a los datos después del entrenamiento [Goodman et al., 2008, Kemp, 2012, Piantadosi et al., 2016]. Aún así, queda por descubrir cómo las diferentes trayectorias de la experiencia pueden dar forma a la adquisición de manera diferente y pueden cambiar constantemente el repertorio de un LoT después de cada exposición.

1.2.2.1. Longitud Mínima de Descripción

MDL

Complejidad de kolmogorov

1.2.2.2. Ciencia Cognitiva Bayesiana

Rational analysis y plot

Capítulo 2

Lenguaje del pensamiento en secuencias binarias

2.1. Trabajos Previos

2.2. Modelo

2.3. Experimento

2.4. Resultados

2.5. Discusión

Capítulo 3

Validación bayesiana de gramáticas para el lenguaje del pensamiento

3.1. Método

3.2. Aplicación al lenguaje de geometría

3.3. Resultados

3.4. Discusión

3.5. Anexo: Probando el teorema de codificación

Capítulo 4

Actualización bayesiana de gramáticas para el lenguaje del pensamiento

4.1. Método

4.1.1. Lenguaje lógico

4.1.2. Modelo libre

4.1.3. Modelo estático

4.1.4. Modelo dinámico

4.2. Experimento

4.3. Resultados

4.4. Discusión

Capítulo 5

Un nuevo marco para estudiar los sesgos de aprendizaje de conceptos en el lenguaje del pensamiento

5.1. Método

5.1.1. Experimento

5.1.2. Representación

5.1.3. Hipótesis

5.2. Resultados

5.3. Discusión

Capítulo 6

BORRAR: Bayesian validation of grammar productions for the language of thought

6.1. Introduction

> It was not only difficult for him to understand that the generic term dog embraced so many unlike specimens of differing sizes and different forms; he was disturbed by the fact that a dog at three-fourteen (seen in profile) should have the same name as the dog at three-fifteen (seen from the front). (...)With no effort he had learned English, French, Portuguese and Latin. I suspect, however, that he was not very capable of thought. To think is to forget differences, generalize, make abstractions. In the teeming world of Funes, there were only details, almost immediate in their presence. [Borges, 1944]

In his fantasy story, the writer Jorge Luis Borges described a fictional character, Funes, capable of remembering every detail of his life but not being able to generalize any of that data into mental categories and hence –Borges stressed– not capable of thinking.

Researchers have modeled these mental categories or conceptual classes with two classical approaches: in terms of similarity to a generic example or prototype [Rosch, 1999, Nosofsky, 1986, Rosch et al., 1976, Rosch and Mervis, 1975] or based on a symbolic/rule-like representation [Boole, 1854, Fodor, 1975, Gentner, 1983].

Symbolic approaches like the *language of thought* (LoT) hypothesis [Fodor, 1975], claim that thinking takes form in a sort of mental language, composed of a limited set of atomic symbols that can be combined to form more complex structures following combinatorial rules.

Despite criticisms and objections [Blackburn, 1984, Loewer and Rey, 1991, Knowles, 1998, Aydede, 1997], symbolic approaches —in general— and the LoT hypothesis —in particular— have gained some renewed attention with recent results that might explain learning across different domains as statistical inference over a compositionally structured hypothesis space [Tenenbaum et al., 2011, Piantadosi and Jacobs, 2016].

The LoT is not necessarily unique. In fact, the form that it takes has been modeled in many different ways depending on the problem domain: numerical concept learning [Piantadosi et al., 2012], sequence learning [Amalric et al., 2017a, Yildirim and Jacobs, 2015, Romano et al., 2013], visual concept learning [Ellis et al., 2015], theory learning [Ullman et al., 2012], etc.

While frameworks may differ on how a LoT may be implemented computationally, they all share the property of being built from a set of atomic symbols and rules by which

they can be combined to form new and more complex expressions.

Most studies of LoTs have focused on the compositional aspect of the language, which has either been modeled within a Bayesian [Tenenbaum et al., 2011] or a Minimum Description Length (MDL) framework [Amalric et al., 2017a, Goldsmith, 2002, Romano et al., 2013, Goldsmith, 2001].

The common method is to define a grammar with a set of productions based on operations that are intuitive to researchers and then study how different inference processes match regular patterns in human learning. A recent study [Piantadosi et al., 2016] puts the focus on the process of how to empirically choose the set of productions and how different LoT definitions can create different patterns of learning. Here, we move along that direction but use Bayesian inference to individuate the LoT instead of comparing several of them by hand.

Broadly, our aim is to propose a method to select the set of atomic symbols in an inferential process by pruning and trimming from a broad repertoire. More precisely, we test whether Bayesian inference can be used to decide the proper set of productions in a LoT defined by a context free grammar. These productions are derived from the subjects' experimental data. In order to do this, a researcher builds a broader language with two sets of productions: 1) those for which she has a strong prior conviction that they should be used in the cognitive task, and 2) other productions that could be used to structure the data and extract regularities even if she believes are not part of the human reasoning repertoire for the task. With the new broader language, she should then turn the context free grammar that defines it into a probabilistic context free grammar (PCFG) and use Bayesian analysis to infer the probability of each production in order to choose the set that best explains the data.

In the next section we formalize this procedure and then apply it on the *language of geometry* presented by Amalric et al. in a recent study about geometrical sequence learning [Amalric et al., 2017a]. This LoT defines a language with some basic geometric instructions as the grammar productions and then models their composition within the MDL framework. Our method, however, can be applied to any LoT model that defines a grammar, independently of whether its compositional aspect is modeled using a Bayesian framework or a MDL approach.

Finally, even with the recent surge of popularity of Bayesian inference and MDL in cognitive science, there are –to the best of our knowledge– no practical attempts to close the gap between probabilistic and complexity approaches to LoT models.

The theory of computation, through Levin's Coding Theorem [Levin, 1974], exposes a remarkable relationship between the *Kolmogorov complexity* of a sequence and its *universal probability*, largely used in algorithmic information theory. Although both metrics are actually non-computable and defined over a universal prefix Turing Machine, we can apply both ideas to other non-universal Turing Machines in the same way that the concept of complexity used in MDL can be computed for specific, non-universal languages.

In this work, we examine the extent to which this theoretical prediction for infinite sequences holds empirically for a specific LoT, the *language of geometry*. Although the inverse logarithmic relationship between both metrics is proved for universal languages in the Coding Theorem, testing this same property for a particular non-universal language shows that the language shares some interesting properties of general languages. This constitutes a first step towards a formal link between probability and complexity modeling frameworks for LoTs.

6.2. Bayesian inference for LoT's productions

The project of Bayesian analysis of the LoT models concept learning using Bayesian inference in a grammatically structured hypothesis space [Goodman et al., 2008]. Each LoT proposal is usually formalized by a context free grammar $\mathcal{G}$ that defines the valid functions or programs that can be generated, like in any other programming language. A program is a derivation tree of $\mathcal{G}$ that needs to be interpreted or executed according to a given semantics in order to get an actual description of the concept in the cognitive task at hand. Therefore, each concept is then represented by any of the programs that describe it and a Bayesian inference process is defined in order to infer from the observed data the distribution of valid programs in $\mathcal{G}$ that describes the concepts.

As explained above, our aim is to derive the productions of $\mathcal{G}$ from the data, instead of just conjecturing them using a priori knowledge about the task. Prior work on LoTs has fit probabilities of productions in a context free grammar using Bayesian inference, however, the focus has been put in integrating out the production probabilities to better predict the data without changing the grammar definition [Piantadosi et al., 2016]. Here, we want to study if the inference process could let us decide which productions can be safely pruned from the grammar. We introduce a generic method that can be used on any grammar to select and test the proper set of productions. Instead of using a fixed grammar and adjusting the probabilities of the productions to predict the data, we use Bayesian inference to rule out productions with probability lower than a certain threshold. This allows the researcher to validate the adequacy of the productions she has chosen for the grammar or even define one that is broad enough to express different regularities and let the method select the best set for the observed data.

To infer the probability for each production based on the observed data, we need to add a vector of probabilities θ associated with each production in order to convert the context free grammar $\mathcal{G}$ into a probabilistic context free grammar (PCFG) [Manning and Schütze, 1999].

Let $D = (d_1, d_2, \ldots, d_n)$ denote the list of concepts produced by the subjects in an experiment. This means that each d_i is a concept produced by a subject in each trial. Then, $P(\theta \mid D)$, the posterior probability of the weights of each production after the observed data, can be calculated by marginalizing over the possible programs that compute D:

$$P(\theta \mid D) = \sum_{\text{Prog}} P(\text{Prog}, \theta \mid D), \tag{6.1}$$

where each $\text{Prog} = (p_1, p_2, \cdots, p_n)$ is a possible set of programs such that each p_i computes the corresponding concept d_i.

We can use Bayesian inference to learn the corresponding programs Prog and the vector θ for each production in the grammar, applying Bayes rule in the following way:

$$P(\text{Prog}, \theta \mid D) \propto P(D \mid \text{Prog})\ P(\text{Prog} \mid \theta)\ P(\theta), \tag{6.2}$$

Sampling the set of programs from $P(\text{Prog} \mid \theta)$ forces an inductive bias which is needed to handle uncertainty under sparse data. Here we use a standard prior for programs that is common in the LoT literature to introduce a syntactic complexity bias that favors shorter programs [Goodman et al., 2008, Overlan et al., 2017]. Intuitively, the probability of sampling a certain program is proportional to the product of the production rules that were used to generate such program, and therefore inversely proportional to the size of the

derivation tree. Formally, it is defined as:

$$P(\text{Prog} \mid \theta) = \prod_{i=1}^{n} P(p_i \mid \theta), \tag{6.3}$$

where $P(p_i \mid \theta) = \prod_{r \in G} \theta_r^{f_r(p_i)}$ is the probability of the program p_i in the grammar, and $f_r(p_i)$ is the number of occurrences of the production r in program p_i.

In (6.2), $P(\theta)$ is a Dirichlet prior over the productions of the grammar. By using the term $P(\theta)$ we are abusing notation for simplicity. The proper term would be $P(\theta \mid \alpha)$ to express a Dirichlet prior with $\alpha \in \mathbb{R}^{\ell}$ its associated concentration vector hyper-parameter where ℓ is the number of productions in the grammar. This hierarchical Dirichlet prior has sometimes been replaced with a uniform prior on productions as it shows no significant differences in prediction results [Piantadosi et al., 2012, Yildirim and Jacobs, 2015]. However, here we will use the Dirichlet prior to be able to infer the production probabilities from this more flexible model.

The likelihood function is straightforward. It does not use any free parameter to account for perception errors in the observation. This forces that only programs that compute the exact concept are taken into account, and it can be easily calculated as follows:

$$P(D \mid \text{Prog}) = \prod_{i=1}^{n} P(d_i \mid p_i), \tag{6.4}$$

where $P(d_i \mid p_i) = 1$ if the program p_i computes d_i, and 0 otherwise.

Calculating $P(\theta \mid D)$ directly is, however, not tractable since it requires to sum over all possible combinations of programs Prog for each of the possible values of θ. To this aim, then, we used a Gibbs Sampling [Geman and Geman, 1984] algorithm for PCFGs via

Markov Chain Monte Carlo (MCMC) similar to the one proposed at [Johnson et al., 2007], which alternates in each step of the chain between the two conditional distributions:

$$P(\text{Prog} \mid \theta, D) = \prod_{i=1}^{n} P(p_i \mid d_i, \theta). \tag{6.5}$$

$$P(\theta \mid \text{Prog}, D) = P_D(\theta \mid f(\text{Prog}) + \alpha). \tag{6.6}$$

Here, P_D is the Dirichlet distribution where the positions of the vector α were updated by counting the occurrences of the corresponding productions for all programs $p_i \in \text{Prog}$.

In the next section, we apply this method to a specific LoT. We add a new set of ad-hoc productions to the grammar that can explain regularities but are not related to the cognitive task. Intuitively, these ad-hoc productions should not be part of the human LoT repertory, still all of them can be used in many possible programs to express each concept.

So far, Probabilistic LoT approaches have been successful to model concept learning from few examples [Tenenbaum et al., 2011, Piantadosi and Jacobs, 2016]. However, this does not mean that Bayesian models would be able to infer the syntax of the model's grammar from sparse data. Here we test such hypothesis. If the method is effective, it should assign a low probability to the ad-hoc productions and instead favor the original set of productions selected by the researchers for the cognitive task. This would not only provide additional empirical evidence about the adequacy of the choice of the original productions for the selected LoT but, more importantly, about the usefulness of Bayesian inference for validating the set of productions involved in different LoTs.

6.3. The Language of Geometry: $\mathcal{G}eo$

The *language of geometry*, $\mathcal{G}eo$ [Amalric et al., 2017a], is a probabilistic generator of sequences of movements on a regular octagon like the one in Fig 6.1. It has been used to model human sequence predictions in adults, preschoolers, and adult members of an indigene group in the Amazon. As in other LoT domains, different models have been proposed for similar spatial sequence domains like the one in [Yildirim and Jacobs, 2015]. Although both successfully model the sequences in their experiments, they propose different grammars for their models (in particular, [Amalric et al., 2017a] contains productions for expressing symmetry reflections). This difference can be explained by the particularities of each experiment. On the one hand, [Amalric et al., 2017a] categorized the sequences in 12 groups based on their complexity, displayed them in an octagon and evaluate the performance of a diverse population to extrapolate them. On the other hand, [Yildirim and Jacobs, 2015] categorized the sequences in 4 groups, displayed them in an heptagon and evaluate the performance of adults not just to predict how the sequence continues, but to transfer the knowledge from the learned sequence across auditory and visual domains. Despite the domains not being equal, the differences in the grammars strengths the need for automatic methods to test and validate multiple grammars for the same domain in the LoT community.

The production rules of grammar $\mathcal{G}eo$ were selected based on previous claims of the universality of certain human geometrical knowledge [Izard et al., 2011, Dehaene et al., 2006, Dillon et al., 2013] such as spatial notions [Landau et al., 1981, Lee et al., 2012] and detection of symmetries [Westphal-Fitch et al., 2012, Machilsen et al., 2009].

With these production rules, sequences are described by concatenating or repeating

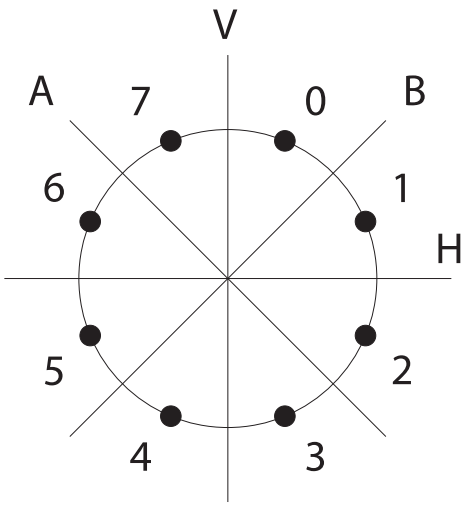

Figura 6.1: **Possible sequence positions and reflection axes.** Σ points around a circle to map current position in the octagon, and the reflection axes.

sequence of movements in the octagon. The original set of productions is shown in Table 6.1 and –besides the concatenation and repetition operators– it includes the following family of atomic geometrical transition productions: anticlockwise movements, staying at the same location, clockwise movements and symmetry movements.

The language actually supports not just a simple n times repetition of a block of productions, but it also supports two more complex productions in the repetition family: repeating with a change in the starting point after each cycle and repeating with a change to the resulting sequence after each cycle. More details about the formal syntax and semantics can be found in [Amalric et al., 2017a], though they are not needed here.

Each program p generated by the grammar describes a mapping $\Sigma \rightarrow \Sigma^+$, for $\Sigma = \{0, \ldots, 7\}$. Here, Σ^+ represents the set of all (non empty) finite sequences over the alphabet Σ, which can be understood as a finite sequence of points in the octagon. These programs must then be executed or interpreted from a starting point in order to get the resulting sequence of points. Let $p =$ [+1,+1] be a program, then $p(0)$ is the result of executing p

Cuadro 6.1: **Original grammar**

Start production			
START	$\rightarrow$	[INST]	start symbol
Basic productions			
INST	$\rightarrow$	ATOMIC	atomic production
INST	$\rightarrow$	INST,INST	concatenation
INST	$\rightarrow$	REP[INST]^n	repeat family with $n \in [2, 8]$
REP	$\rightarrow$	REP0	simple repeat
REP	$\rightarrow$	REP1<ATOMIC>	repeat with starting point variation using ATOMIC
REP	$\rightarrow$	REP2<ATOMIC>	repeat with resulting sequence variation using ATOMIC
Atomic productions			
ATOMIC	$\rightarrow$	-1	next element anticlockwise (ACW)
ATOMIC	$\rightarrow$	-2	second element ACW
ATOMIC	$\rightarrow$	-3	third element ACW
ATOMIC	$\rightarrow$	+0	stays at same location
ATOMIC	$\rightarrow$	+1	next element clockwise (CW)
ATOMIC	$\rightarrow$	+2	second element CW
ATOMIC	$\rightarrow$	+3	third element CW
ATOMIC	$\rightarrow$	A	symmetry around one diagonal axis
ATOMIC	$\rightarrow$	B	symmetry around the other diagonal axis
ATOMIC	$\rightarrow$	H	horizontal symmetry
ATOMIC	$\rightarrow$	V	vertical symmetry
ATOMIC	$\rightarrow$	P	rotational symmetry

starting from point 0 (that is, sequence $1, 2$) and $p(4)$ is the result of executing the same program starting from point 4 in the octagon (sequence $5, 6$).

Each sequence can be described with many different programs: from a simple concatenation of atomic productions to more compressed forms using repetitions. For example, to move through all the octagon clockwise one point at a time starting from point 0, one can use [+1,+1,+1,+1,+1,+1,+1,+1](0) or [REP[+1]^8](0) or [REP[+1]^7, +1](0), etc. To alternate 8 times between points 6 and 7, one can use a reflection production like [REP[A]^8](6), or [REP[+1,-1]^4](6).

6.3.1. $\mathcal{G}eo$'s original experiment

To infer the productions from the observed data, we used the original data from the experiment in [Amalric et al., 2017a]. In the experiment, volunteers were exposed to a series of spatial sequences defined on an octagon and were asked to predict future locations. The sequences were selected according to their MDL in the *language of geometry* so that each sequence could be easily described with few productions.

Participants The data used in this work comes, except otherwise stated, from Experiment 1 in which participants were 23 French adults (12 female, mean age $= 26{,}6$, age range $= 20 - 46$) with college-level education. Data from Experiment 2 is later used when comparing adults and children results. In the later, participants where 24 preschoolers (minimal age $= 5{,}33$, max $= 6{,}29$, mean $= 5{,}83 \pm 0{,}05$).

Procedure On each trial, the first two points from the sequence were flashed sequentially in the octagon and the user had to click on the next location. If the subject selected the correct location, she was asked to continue with the next point until the eight points of the sequences were completed. If there was an error at any point, the mistake was corrected, the sequence flashed again from the first point to the corrected point and the user asked to predict the next location. Each $d_i \in \Sigma^8$ from our dataset D is thus the sequence of eight positions clicked in each subject's trial. The detailed procedure can be found in the cited work.

6.3.2. Extending $\mathcal{G}eo$'s grammar

We will now expand the original set of productions in $\mathcal{G}eo$ with a new set of productions that can also express regularities but are not related to any geometrical intuitions to test our Bayesian inference model.

In Table 6.2 we show the new set of productions which includes instructions like moving to the point whose label is the square of the current location's label, or using the current point location i to select the i^{th}digit of a well-known number like π or Chaitin's number (calculated for a particular universal Turing Machine and programs up to 84 bits long [Calude et al., 2002]). All digits are returned in arithmetic module 8 to get a valid point for the next position. For example, PI(0) returns the first digit of π, that is PI$(0) = 3$ $\text{mód } (8) = 3$; and PI$(1) = 1$.

Cuadro 6.2: **Ad-hoc productions**

ATOMIC	$\rightarrow$	DOUBLE	$(\text{location} * 2) \ \text{ mód } 8$
ATOMIC	$\rightarrow$	-DOUBLE	$(\text{location} * -2) \ \text{ mód } 8$
ATOMIC	$\rightarrow$	SQUARE	$(\text{location}^2) \ \text{ mód } 8$
ATOMIC	$\rightarrow$	GAMMA	$\Gamma(\text{location}+1) \ \text{ mód } 8$
ATOMIC	$\rightarrow$	PI	location-th digit of π
ATOMIC	$\rightarrow$	EULER	location-th digit of e
ATOMIC	$\rightarrow$	GOLD	location-th digit of ϕ
ATOMIC	$\rightarrow$	PYTH	location-th digit of $\sqrt{2}$
ATOMIC	$\rightarrow$	KHINCHIN	location-th digit of Khinchin's constant
ATOMIC	$\rightarrow$	GLAISHER	location-th digit of Glaisher's constant
ATOMIC	$\rightarrow$	CHAITIN	location-th digit of Chaitin Omega's constant

6.3.3. Inference results for $\mathcal{G}eo$

To let the MCMC converge faster (and to later compare the concept's probability with their corresponding MDL), we generated all the programs that explain each of the observed sequences from the experiment. In this way, we are able to sample from the exact distribution $P(p_i \mid d_i, \theta)$ by sampling from a multinomial distribution of all the possible programs p_i that compute d_i, where each p_i has probability of occurrence equal to $P(p_i \mid \theta)$.

To get an idea of the expressiveness of the grammar to generate different programs for a sequence and the cost of computing them, it is worth mentioning that there are more than 159 million programs that compute the 292 unique sequences generated by the subjects in the experiment, and that for each sequence there is an average of 546,713 programs (min = $10,749$, max = $5,500,026$, σ = $693,618$).

Fig 6.2 shows the inferred θ for the observed sequences from subjects, with a unit concentration parameter for the Dirichlet prior, $\alpha = (1, \ldots, 1)$. Each bar shows the mean probability and the standard error of each of the atomic productions after 50 steps of the MCMC, leaving the first 10 steps out as burn-in.

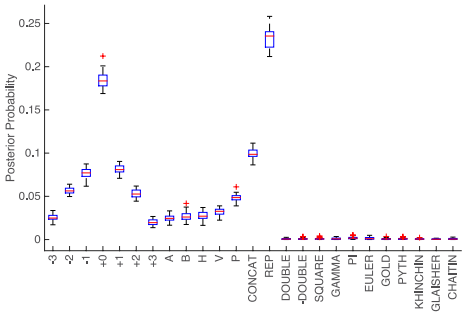

Figura 6.2: **Inferred θ_i. Inferred probability for each production in the grammar**

Although 50 steps might seem low for a MCMC algorithm to converge, our method calculated $P(p_i \mid d_i, \theta)$ exactly in order to speed up convergence and to be able to later

compare the probability with the complexity from the original MDL model. In Fig 6.3, we show an example trace for four MCMC runs for θ_{+0}, which corresponds to the atomic production +0, but is representative of the behavior of all θ_i. (see **MCMC steps for $\mathcal{G}eo$'s productions. MCMC steps for the rest of $\mathcal{G}eo$'s grammar productions.** for the full set of productions).

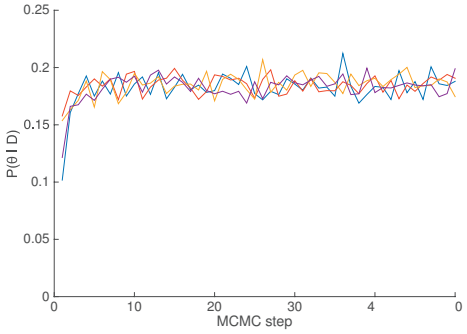

Figura 6.3: **Inferred θ_{+0}.** Inferred probability for +0 production at each step in four MCMC chains.

Fig 6.2 shows a remarkable difference between the probability of the productions that were originally used based on geometrical intuitions and the ad-hoc productions. The plot also shows that each clockwise production has almost the same probability as its corresponding anticlockwise production, and a similar relation appears between horizontal and vertical symmetry (H and V) and symmetries around diagonal axes (A and B). This is important because the original experiment was designed to balance such behavior; the inferred grammar reflects this.

Fig 6.4 shows the same inferred θ but grouped according to production family. Grouping stresses the low probability of all the ad-hoc productions, but also shows an important difference between REP and the rest of the productions, particularly the simple concatenation of productions (CONCAT). This indicates that the *language of geometry* is capable of reusing simpler structures that capture geometrical meaning to explain the observed data, a key

aspect of a successful model of LoT.

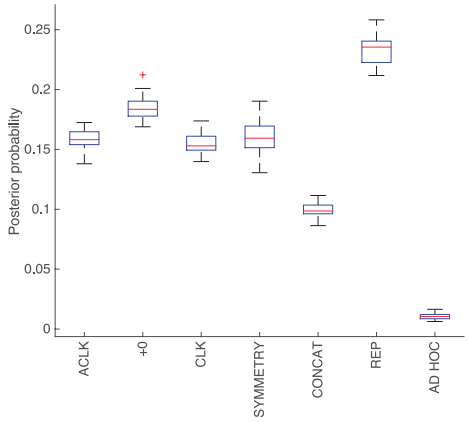

Figura 6.4: **Inferred θ_i grouped by family.** Inferred probability for each production in the grammar grouped by family.

We then ran the same inference method using observed sequences from other experiments but only with the original grammar productions (i.e. setting aside the ad-hoc productions). We compared the result of inferring over our previously analyzed sequences generated by adults with sequences generated by children (experiment 2 from [Amalric et al., 2017a]) and the actual expected sequences for an ideal player.

Fig 6.5 shows the probabilities for each atomic production that is inferred after each population. The figure denotes that different populations can converge to different probabilities and thus different LoTs. Specifically, it is worth mentioning that the ideal learner indeed uses more repetition productions than simple concatenations when compared to adults. In the same way, adults use more repetitions than children. This could mean that the ideal learner is capable of reproducing the sequences by recursively embedding other smaller programs, whereas adults and children more so have problems understanding or learning the smaller concept that can explain all the sequences from the experiments, which is consistent with the results from the MDL model in [Amalric et al., 2017a].

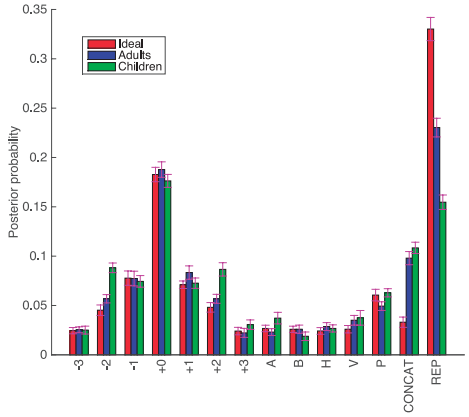

Figura 6.5: **Inferred θ_i for ideal learner, adults and children.** Inferred probability for each production in the grammar for different population data.

It is worth mentioning that in [Amalric et al., 2017a] the complete grammar for the *language of geometry* could explain adults' behavior but had problems to reproduce the children's patterns for some sequences. However, they also showed that penalizing the rotational symmetry (P) could adequately explain children's behavior. In Fig 6.5, we see that the mean value of (P) for children is 0.06 whereas in adults it's 0.05 (a two-sample t-test reveals t = -12.6, p = 10-19). This might not necessarily be contradictory, as the model for children in [Amalric et al., 2017a] was used to predict the next symbol of a sequence after seeing its prefix by adding a penalization for extensions that use the rotational symmetry in the *minimal* program of each sequence. On the other hand, the Bayesian model in this work tries to explain the observed sequences produced by children considering the probability of a sequence summing over *all* the possible programs that can generate it and not just the ones with minimal size. Thus, a production like (P) that might not be part of the minimal program for a sequence might not necessarily be less probable when considering the entire distribution of programs for that same sequence.

6.4. Coding Theorem

For each phenomenon there can always be an extremely large, possibly infinite, number of explanations. In a LoT model, this space is constrained by the grammar $\mathcal{G}$ that defines the valid hypotheses in the language. Still, one has to define how a hypothesis is chosen among all possibilities. Following Occam's razor, one should choose the simplest hypothesis amongst all the possible ones that explain a phenomenon. In cognitive science, the MDL framework has been widely used to model such bias in human cognition, and in *the language of geometry* in particular [Amalric et al., 2017a]. The MDL framework is based on the ideas of information theory [Shannon, 1948], Kolmogorov complexity [Kolmogorov, 1968] and Solomonoff induction [Solomonoff, 1964].

Occam's razor was formalized by Solomonoff [Solomonoff, 1964] in his theory of universal inductive inference, which proposes a universal prediction method that successfully approximates any distribution μ based on previous observations, with the only assumption of μ being computable. In short, Solomonoff's theory uses all programs (in the form of prefix Turing machines) that can describe previous observations of a sequence to calculate the probability of the next symbols in an optimal fashion, giving more weight to shorter programs. Intuitively, simpler theories with low complexity have higher probability than theories with higher complexity. Formally, this relationship is described by the Coding Theorem [Levin, 1974], which closes the gap between the concepts of Kolmogorov complexity and probability theory. However, LoT models that define a probabilistic distribution for their hypotheses do not attempt to compare it with a complexity measure of the hypotheses like the ones used in MDL, nor the other way around.

In what follows we formalize the Coding Theorem (for more information, see [Li and

Vitányi, 2013]) and test it experimentally. To the best our knowledge, this is the first attempt to validate these ideas for a particular (non universal) language. The reader should note that we are not validating the theorem itself as it has already been proved for universal Turing Machines. Here, we are testing whether the inverse logarithmic relationship between the probability and complexity holds true when defined for a specific non universal language.

6.4.1. The formal statement

Let M be a prefix Turing machine –by *prefix* we mean that if $M(x)$ is defined, then M is undefined for every proper extension of x. Let $P_M(x)$ be the probability that the machine M computes output x when the input is filled-up with the results of fair coin tosses, and let $K_M(x)$ be the *Kolmogorov complexity of* x *relative to* M, which is defined as the length of the shortest program which outputs x, when executed on M. The Coding Theorem states that for every string x we have

$$\log \frac{1}{P_U(x)} = K_U(x) \tag{6.7}$$

up to an additive constant, whenever U is a *universal* prefix Turing machine –by *universal* we mean a machine which is capable of simulating every other Turing machine; it can be understood as the underlying (Turing-complete) chosen programming language. It is important to remark that neither P_U, nor K_U are computable, which means that such mappings cannot be obtained through effective means. However, for specific (non-universal) machines M, one can, indeed, compute both P_M and K_M.

6.4.2. Testing the Coding Theorem for $\mathcal{G}eo$

Despite the fact that P_M and K_M are defined over a Turing Machine M, the reader should note that a LoT is not usually formalized with a Turing Machine, but instead as a programming language with its own syntax of valid programs and semantics of execution, which stipulates how to compute a concept from a program. However, one can understand programming languages as defining an equivalent (not necessarily universal) Turing Machine model, and a LoT as defining its equivalent (not necessarily universal) Turing Machine $\mathcal{G}$. In short, machines and languages are interchangeable in this context: they both specify the programs/terms, which are symbolic objects that, in turn, describe semantic objects, namely, strings.

The Kolmogorov complexity relative to $\mathcal{G}eo$ In [Amalric et al., 2017a], the Minimal Description Length was used to model the combination of productions from the *language of geometry* into concepts by defining a Kolmogorov complexity relative to the *language of geometry*, which we denote $K_{\mathcal{G}eo}$. $K_{\mathcal{G}eo}(x)$ is the minimal size of an expression in the grammar of $\mathcal{G}eo$ which describes x. The formal definition of 'size' can be found in the cited work but in short: each of the atomic productions adds a fixed cost of 2 units; using any of the repetition productions to iterate n times a list of other productions adds the cost of the list, plus $\lfloor \log(n) \rfloor$; and joining two lists with a concatenation costs the same as the sum of the costs of both lists.

The probability relative to $\mathcal{G}eo$ On the other hand, with the Bayesian model specified in this work, we can define $P(x \mid \mathcal{G}eo, \theta)$ which is the probability of a string x relative to $\mathcal{G}eo$ and its vector of probabilities for each of the productions.

For the sake of simplicity, we will use $P_{\mathcal{G}eo}(x)$ to denote $P(x \mid \mathcal{G}eo, \theta)$ when θ is the inferred probability from the observed adult sequences from the experiment.

$$
\begin{aligned}
P_{\mathcal{G}eo}(x) &= P(x \mid \mathcal{G}eo, \theta) && (6.8)\\
&= \sum_{\text{prog}} P(x \mid \text{prog}, \theta) && (6.9)\\
&\propto \sum_{prog} P(x \mid \text{prog}) P(\text{prog} \mid \theta). && (6.10)
\end{aligned}
$$

Here, we calculate both $P_{\mathcal{G}eo}(x)$ and $K_{\mathcal{G}eo(x)}$ in an exact way (note that $\mathcal{G}eo$, seen as a programming language, is not Turing-complete). In this section, we show an experimental equivalence between such measures which is consistent with the Coding Theorem. We should stress, once more, that the theorem does not predict that this relationship should hold for a specific non-universal Turing Machine.

To calculate $P_{\mathcal{G}eo}(x)$ we are not interested in the normalization factor of $P(x \mid \text{prog}) P(\text{prog} \mid \theta)$ because we are just trying to measure the relationship between $P_{\mathcal{G}eo}$ and $K_{\mathcal{G}eo}$ in terms of the Coding Theorem. Note, however, that calculating $P_{\mathcal{G}eo}(x)$ involves calculating all programs that compute each of the sequences as in our previous experiment. To make this tractable we calculated $P_{\mathcal{G}eo}(x)$ for 10,000 unique random sequences for each of the possible sequence lengths from the experiment (i.e., up to eight). When the length of the sequence did not allow 10,000 unique combinations, we used all the possible sequences of that length.

6.4.3. Coding Theorem Results

Fig 6.6 shows the mean probability $P_{\mathcal{G}eo}(x)$ for all sequences x with the same value of $K_{\mathcal{G}eo(x)}$ and length between 4 and 8 ($|x| \in [4, 8]$) for all generated sequences x. The data is plotted with a logarithmic scale for the x-axis, illustrating the inverse logarithmic relationship between $K_{\mathcal{G}eo}(x)$ and $P_{\mathcal{G}eo}(x)$. The fit is very good, with $R^2 = ,99$, $R^2 = ,94$, $R^2 = ,97$, $R^2 = ,99$ and $R^2 = ,98$ for Fig 6.6A, Fig 6.6B, Fig 6.6C, Fig 6.6D and Fig 6.6E, respectively.

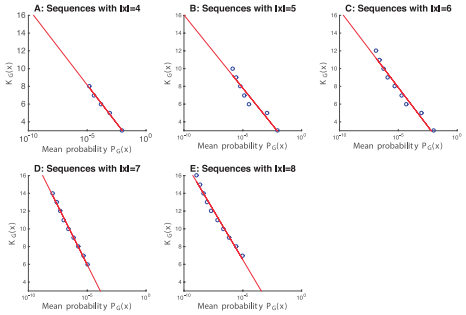

Figura 6.6: **Mean probability** $P_{\mathcal{G}eo}(x)$**.** Mean probability $P_{\mathcal{G}eo}(x)$ for all sequences x with the same complexity. Subfigure A: Sequences with $|x| = 4$. Subfigure B: Sequences with $|x| = 5$. Subfigure C: Sequences with $|x| = 6$. Subfigure D: Sequences with $|x| = 7$. Subfigure E: Sequences with $|x| = 8$.

This relationship between the complexity $K_{\mathcal{G}eo}$ and the probability $P_{\mathcal{G}eo}$ defined for finite sequences in the *language of geometry*, matches the theoretical prediction for infinite sequences in universal languages described in the Coding Theorem. At the same time, it captures the Occam's razor intuition that the simpler sequences one can produce or explain with this language are also the more probable.

Fig 6.7 and Fig 6.8 show the histogram of $P_{\mathcal{G}eo}(x)$ and $K_{\mathcal{G}eo}(x)$, respectively, for sequences with length = 8 to get a better insight about both measures. The histogram of the rest of the sequence's lengths are included in **Histograms of complexity** $K_{\mathcal{G}eo}(x)$**.**

Histograms of complexity for sequences with length between 4 and 8. and **Histograms of probability $P_{\mathcal{G}eo}(x)$. Histograms of probability for sequences with length between 4 and 8.** for completeness, and they all show the same behavior.

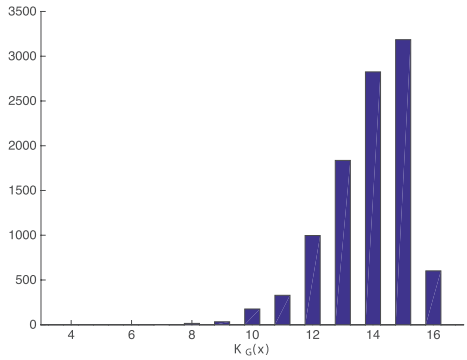

Figura 6.7: **Histogram of complexity $K_{\mathcal{G}eo}(x)$.** Histogram of complexity for sequences x with $|x| = 8$.

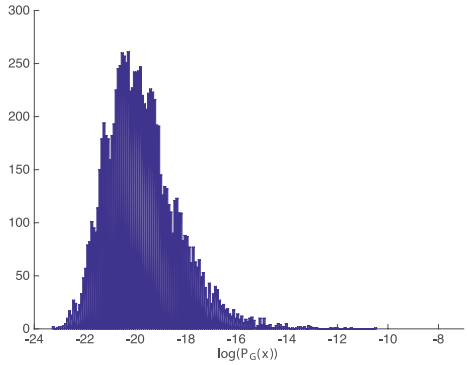

Figura 6.8: **Histogram of probability $P_{\mathcal{G}eo}(x)$.** Histogram of probability for sequences x with $|x| = 8$.

6.5. Discussion

We have presented a Bayesian inference method to select the set of productions for a LoT and test its effectiveness in the domain of a geometrical cognition task. We have

shown that this method is useful to distinguish between arbitrary ad-hoc productions and productions that were intuitively selected to mimic human abilities in such domain.

The proposal to use Bayesian models tied to PCFG grammars in a LoT is not new. However, previous work has not used the inferred probabilities to gain more insight about the grammar definition in order to modify it. Instead, it had usually integrated out the production probabilities to better predict the data, and even found that hierarchical priors for grammar productions show no significant differences in prediction results over uniform priors [Piantadosi et al., 2012, Yildirim and Jacobs, 2015].

We believe that inferring production probabilities can help prove the adequacy of the grammar, and can further lead to a formal mechanism for selecting the correct set of productions when it is not clear what a proper set should be. Researchers could use a much broader set of productions than what might seem intuitive or relevant for the domain and let the hierarchical Bayesian inference framework select the best subset.

Selecting a broader set of productions still leaves some arbitrary decisions to be made. However, it can help to build a more robust methodology that –combined with other ideas like testing grammars with different productions for the same task [Piantadosi et al., 2016]– could provide more evidence of the adequacy of the proposed LoT.

Having a principled method for defining grammars in LoTs is a crucial aspect for their success because slightly different grammars can lead to different results, as has been shown in [Piantadosi et al., 2016].

The experimental data used in this work was designed at [Amalric et al., 2017a] to understand how humans encode visuo-spatial sequences as structured expressions. As future research, we plan to perform a specific experiment to test these ideas in a broader

range of domains. Additionally, data from more domains is needed to demonstrate if this method could also be used to effectively prove whether different people use different LoT productions as outlined in Fig 6.5.

Finally, we showed an empirical equivalence between the complexity of a sequence in a minimal description length (MDL) model and the probability of the same sequence in a Bayesian inference model which is consistent with the theoretical relationship described in the Coding Theorem. This opens an opportunity to bridge the gap between these two approaches that had been described ad complementary by some authors [MacKay, 2003].

6.6. Supporting information

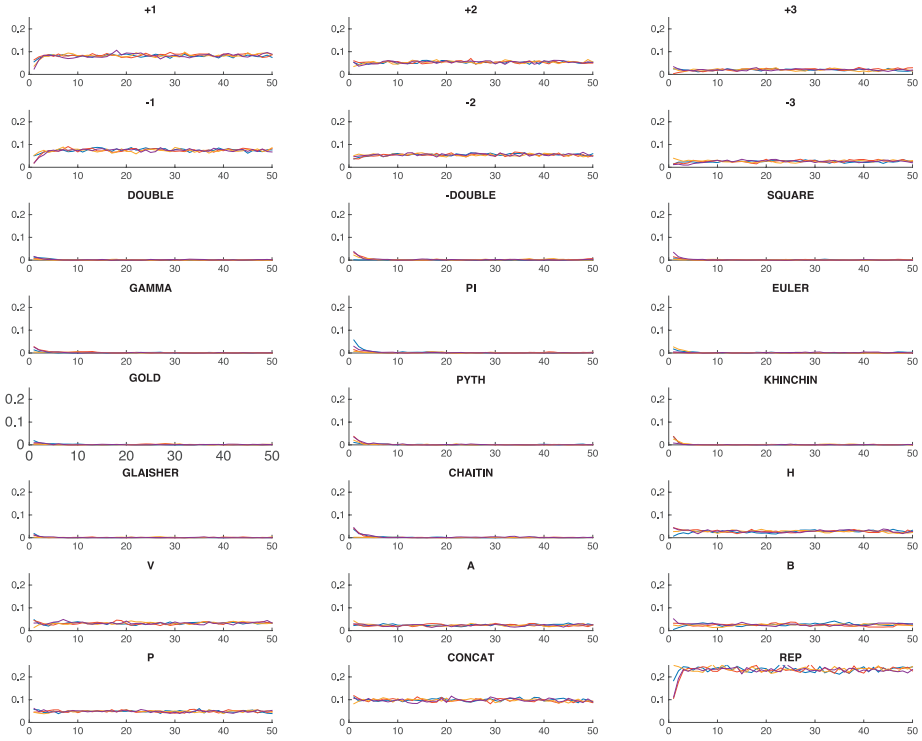

Figura 6.9: **MCMC steps for $\mathcal{G}eo$'s productions. MCMC steps for the rest of $\mathcal{G}eo$'s grammar productions.**

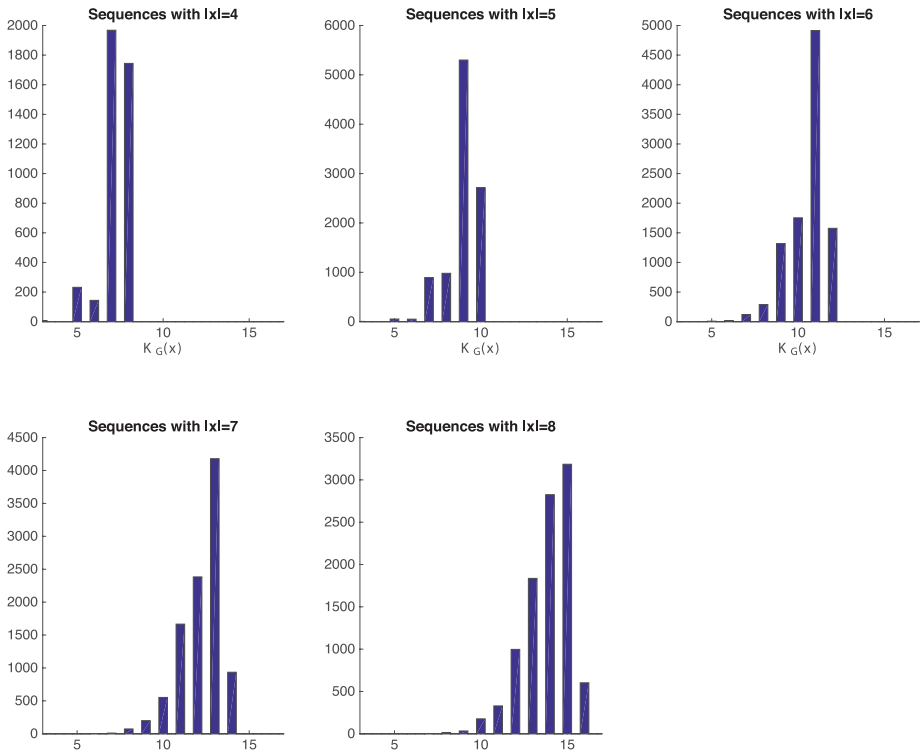

Figura 6.10: **Histograms of complexity $K_{Geo}(x)$. Histograms of complexity for sequences with length between 4 and 8.**

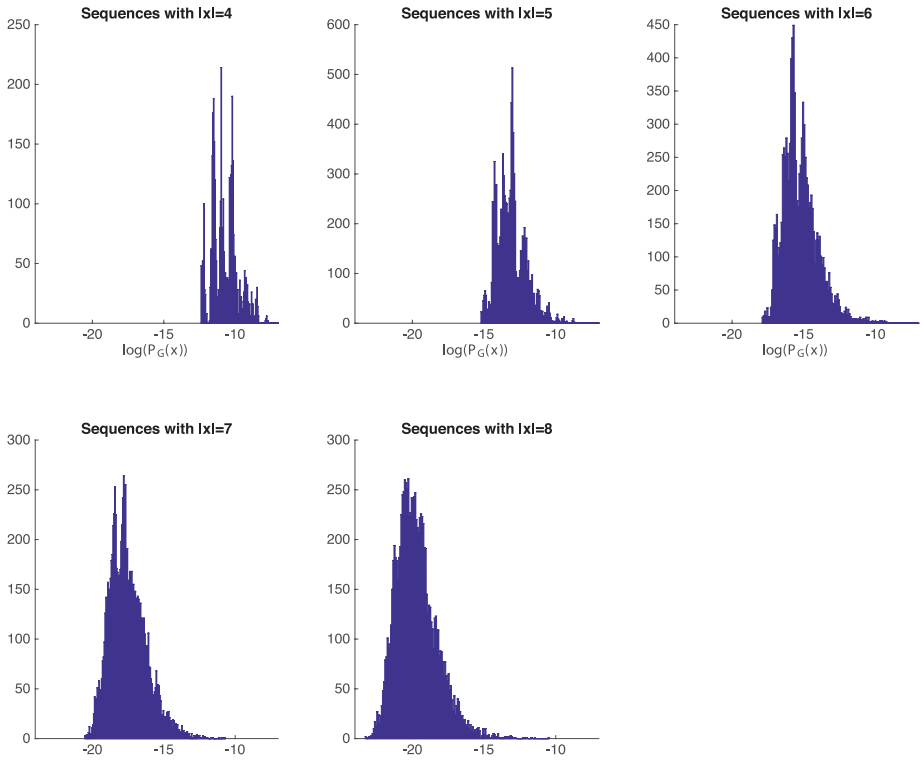

Figura 6.11: **Histograms of probability $P_{\mathcal{G}eo}(x)$. Histograms of probability for sequences with length between 4 and 8.**

Capítulo 7

BORRAR: Towards a more flexible Language of Thought: Bayesian grammar updates after each concept exposure

7.1. Introduction

How can children acquire a vast universe of concepts with seemingly very little exposure? One possible solution to this conundrum, known as the Plato Problem [Chomsky, 1986, Chomsky et al., 2006], builds on the human capacity to describe concepts –and more generally of all elements of thought– through the use of a symbolic and combinatorial mental language [Newell, 1980], referred as *language of thought* (LoT) [Fodor, 1975].

Combinatorial languages can describe a vast set of concepts from a small set of primitives. This can be understood in a relatively simple example in the domain of shapes. A combinatorial and symbolic language similar to Logo [Abelson et al., 1974] can combine operations such as “move", “pen up", “pen down.or “rotate"to generate an infinite set of expressions (or programs) which, when evaluated, can convey all sort of shapes.

A language describing concepts (like shapes) also provides a natural notion of their complexity [Kolmogorov, 1968]. A concept is simple, relative to that language, when it can be described by a short program. On the contrary, it is complex when all its descriptions require a long sequence of instructions. For example, in the case of the Logo language, a square can simply be instructed as a loop of four displacements followed by rotations of 90 degrees. In this language, the icon of a face will be implemented by a significant lengthier program and hence will be more complex. However, this concept would be simpler when described in a language in which the icon of a face (or the symbols for nose, mouth, etc.) are available as primitives in the language.

In the domain of Boolean concepts, a wide range of logical varieties of concepts was studied in [Feldman, 2003], revealing a surprisingly simple ‘law’: the subjective difficulty of a Boolean concept for a human learner is directly proportional to the length of the shortest compatible program in the language of propositional logic (i.e. Boolean variables combined with the operators *and*, *or* and *not*). This result may suggest that human LoT is equipped with rules and symbols similar to those found in propositional logic. Indeed, the correlation between the subjective difficulty of concepts and their complexity has been used as a general vehicle to study human LoT in various domains [Piantadosi et al., 2016, Leeuwenberg, 1971, Amalric et al., 2017b, Romano et al., 2018, Lupyan et al., 2007]. Although often implicit, the general strategy is to *(1)* assume a language; *(2)* find the

shortest compatible program for some concepts in that language; *(3)* compare the length of these programs with the subjective difficulty of the concepts; and finally *(4)* repeat this process for various languages within a universe of possible candidates and choose the language that gives the best match in *(3)*. As mentioned before, the length of the program depends on the primitives of the language in which this program is written, so different languages make different predictions.

A natural question, however, is whether the primitives of a LoT are universal –both across different individuals and also throughout development– or if instead the semantic repertoire of a language is dynamic and shaped by experience. Indeed, it is likely that our ability to automatically represent Boolean concepts in a succinct manner is not due to an innate efficient propositional language in our mind. Instead, we propose that this ability arises as a byproduct of our brain rapidly learning efficient representations for the concepts we usually encounter in everyday life. Our research question is: how rapidly can we adapt our learning mechanisms when we encounter a new domain in which our a priori representations are no longer efficient? We examine the hypothesis that humans have the ability to rapidly recombine propositions in their LoT, adding new primitives to their language. In other words, that learning leads to a process of compiling routines into functions within the LoT.

In the example of the Logo language one can imagine that if productions which draw squares are very frequent, it would be efficient to devote a new symbol to this production. The new symbol ‘square’ is a hierarchical ‘second order’ construction of the ‘first order’ primitives of the language. It has a cost (of increasing the lexicon of the language) but in the new language, drawing a square can be instantiated with a very short program (namely, ‘square’) and hence uses less memory. Indeed, a higher level language allows us to reach a

higher level of abstraction by freeing memory and processing power, thus making more complex thoughts thinkable [Minsky, 1967, Murphy, 1988].

Most work in the LoT literature, while naturally including a learning mechanism, tends to approach the LoT as a stable system to be unearthed by experimenters, who try different candidate templates and select the one which best fits the data after training [Goodman et al., 2008, Kemp, 2012, Piantadosi et al., 2016]. Still, how different tracks of experience can shape acquisition differently and can constantly change the repertoire of a LoT after each exposure remains to be discovered.

Here, we perform a Boolean concept learning experiment to show that humans can change very rapidly –in the course of an experiment– the repertoire of symbols they use to identify concepts. We also provide a dynamic model that is flexible enough to update its underlying language after each concept exposure.

In our experiment, participants are divided in two groups, in such a way that each group is presented with a different sequence of concepts. One of the two groups is presented with concepts that are succinctly described only if the logical operator 'exclusive or' (xor, notated $\oplus$) is used, which we presume does not form part of the natural repertoire of LoT in this specific domain [Piantadosi et al., 2016]. However, these concepts can also be described with a sensibly lengthy combination of primitives excluding $\oplus$. We show how the exposure to this set of concepts 'compiles' the $\oplus$ operator in a way that, after exposure, subjective difficulty is described by an extended language in which $\oplus$ has been incorporated to the set of primitives. Furthermore, we show that the subjective difficulty of concepts throughout the task is consistent with that of a Bayesian agent that rationally updates the probability of compiling $\oplus$ according to how useful it has been to compress concepts so far.

7.2. The logical setting

We consider two propositional logics, both containing only four propositional variables $\mathsf{Vars} = \{x_1, x_2, x_3, x_4\}$. P is defined over the signature $\wedge$, $\vee$ and $\neg$, and $\mathsf{P}^{\oplus}$ is defined over the signature $\wedge$, $\vee$, $\neg$ and $\oplus$. As one can see from the grammars defined in Fig. 7.1, the only difference between P and $\mathsf{P}^{\oplus}$ is that the latter has an additional operator $\oplus$.

$$
\begin{array}{lcl}
\text{START} & \rightarrow & \text{BOOL} \\
\text{BOOL} & \rightarrow & (\text{BOOL} \wedge \text{BOOL}) \\
\text{BOOL} & \rightarrow & (\text{BOOL} \vee \text{BOOL}) \\
\text{BOOL} & \rightarrow & \text{ATOM}
\end{array}
\qquad \text{For } i = 1, 2, 3, 4
\qquad
\begin{array}{lcl}
\text{ATOM} & \rightarrow & x_i \\
\text{ATOM} & \rightarrow & \neg x_i
\end{array}
$$

Figura 7.1: The context free grammar for language P. Language $\mathsf{P}^{\oplus}$ has an extra rule: $\text{BOOL} \rightarrow (\text{BOOL} \oplus \text{BOOL})$

The semantics of $\wedge$, $\vee$ and $\neg$ are standard: conjunction, disjunction and negation, respectively. We let $\oplus$ denote the exclusive disjunction. As usual, $v \models \varphi$, represents that the formula φ is true for the valuation $v : \mathsf{Vars} \rightarrow \{0, 1\}$ and we denote the *semantics* of φ by $[\![\varphi]\!] = \{v \colon v \models \varphi\}$. A *concept* **Con** is a set of valuations $\mathsf{Vars} \rightarrow \{0, 1\}$. The complement of **Con** is denoted $\overline{\textbf{Con}}$ and is defined as $\overline{\textbf{Con}} = \{0, 1\}^{\mathsf{Vars}} \setminus \textbf{Con}$. Observe that $\#\textbf{Con} + \#\overline{\textbf{Con}} = 16$. We say that a formula φ is *compatible* with concept **Con** if $[\![\varphi]\!] = \textbf{Con}$. We regard logics as languages for describing concepts. Any concept **Con** has infinitely many descriptions, namely, all formulas φ such that $[\![\varphi]\!] = \textbf{Con}$.

Example. In Fig. 7.2 we depict a concept **Con** (variables are represented by colors) such that $\#\textbf{Con} = 4$. One can see that the formula x_3 is not compatible with **Con** but $x_1 \wedge x_2$, or $x_1 \wedge x_2 \wedge (x_3 \vee \neg x_3)$, are compatible with **Con**. $\overline{\textbf{Con}}$ may be described by $\neg x_1 \vee \neg x_2$.

We will often identify concepts with any formula compatible with it, so we will talk

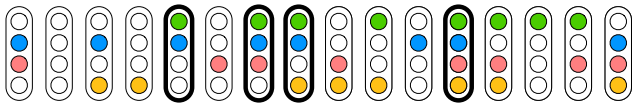

Figura 7.2: Example of a concept **Con**, as shown in the experiment. Variables in $\mathsf{Vars} = \{x_1, x_2, x_3, x_4\}$ correspond to the presence of a color light in the object ($x_1 =$ green, $x_2 =$ blue, $x_3 =$ red, $x_4 =$ orange). Items (valuations) belonging to **Con** are highlighted with bold border. **Con** may be described by $x_1 \wedge x_2$. As in a traffic light, each color is fixed to each position.

of "concept φ" to refer to "concept $[\![\varphi]\!]$". However, it should be noted that a concept is a semantic object that has many descriptions in the logical language.

A lower bound for the complexity of a concept in a given logic corresponds to the shortest description of that concept, that is, its minimum description length (MDL).

The *size* of a formula φ is denoted $|\varphi|$ and it is defined as the number of operators plus the number of atoms (i.e. possibly negated propositional symbols), that is: $|x_i| = |\neg x_i| = 1$ for $i = 1 \ldots 4$ and $|\varphi_1 * \varphi_2| = |\varphi_1| + |\varphi_2| + 1$ for $* \in \{\wedge, \vee, \oplus\}$. For $\mathsf{L} \in \{\mathsf{P}, \mathsf{P}^{\oplus}\}$ and a concept **Con** we define the *minimum description length of* ***Con*** *with respect to* L as

$$MDL\mathsf{L}(\mathbf{Con}) = \min\{|\varphi| \colon \varphi \in \mathcal{L}, [\![\varphi]\!] = \mathbf{Con}\}.$$

Since P is a sublanguage of $\mathsf{P}^{\oplus}$, we have $MDL\mathsf{P}^{\oplus}(\mathbf{Con}) \leq MDL\mathsf{P}(\mathbf{Con})$ for any concept **Con**.

Example. The concept $\mathbf{Con} = \{v \colon v(x_1) + v(x_2) = 1\}$, which expresses that x_1 is true or x_2 is true but not both can be described in $\mathsf{P}^{\oplus}$ as $\varphi = x_1 \oplus x_2$, of length 3. In fact, one can check that this is the shortest formula compatible with **Con**, and so $MDL\mathsf{P}^{\oplus}(\mathbf{Con}) = 3$. If we now switch to P, we can no longer describe **Con** as $x_1 \oplus x_2$, since $\oplus$ is not part of its signature. However, in P, the concept **Con** may be described by

formula $\psi = (x_1 \wedge \neg x_2) \vee (x_2 \wedge \neg x_1)$, of size 7. Since this formula has minimal size, we have that $MDL\mathsf{P}(\mathbf{Con}) = 7$.

7.3. Experiment

	Target group	$MDL\mathsf{P}^{\oplus}(\mathbf{Con})$	$MDL\mathsf{P}(\mathbf{Con})$	Control group	$MDL\mathsf{P}^{\oplus}(\mathbf{Con})$
Training	$\mathbf{Con}^1 \quad x_i$	1	1		⟵Idem
	$\mathbf{Con}_t^2 \quad x_i \oplus x_j$	3	7	$\mathbf{Con}_c^2 \quad x_i \vee x_j$	3
	$\mathbf{Con}_t^3 \quad x_i \oplus x_j \oplus x_k$	5	19	$\mathbf{Con}_c^3 \quad x_i \vee (x_j \wedge x_k)$	5
	$\mathbf{Con}_t^4 \quad x_k \oplus x_l$	3	7	$\mathbf{Con}_c^4 \quad x_k \vee x_l$	3
Test	$\mathbf{Con}^5 \quad x_i \wedge (x_j \oplus x_k)$	5	9		⟵Idem
	$\mathbf{Con}^6 \quad x_i \wedge (x_j \vee x_k)$	5	5		⟵Idem

Cuadro 7.1: Sequence of concepts presented in the experiment: $\mathbf{Con}^1$, $\mathbf{Con}_t^2$, $\mathbf{Con}_t^3$, $\mathbf{Con}_t^4$, $\mathbf{Con}^5$, $\mathbf{Con}^6$ for target group and $\mathbf{Con}^1$, $\mathbf{Con}_c^2$, $\mathbf{Con}_c^3$, $\mathbf{Con}_c^4$, $\mathbf{Con}^5$, $\mathbf{Con}^6$ for control group. Each concept **Con** is represented by a minimal formula φ such that $[\![\varphi]\!] = \mathbf{Con}$.

55 participants participated in the experiment over the world wide web using the Amazon Mechanical Turk crowd sourcing platform. All were US residents over the age of 18 and had more than 95 % of past tasks successfully approved by other requesters. 44 participants completed the experiment through all the stages and declared not cheating (using pen, screenshots or a similar method to copy the answers) at the end of the experiment. Only data from these participants were used in the analyses reported below.[1]

Participants were divided randomly into a control group ($N = 21$) and a target group ($N = 23$). Both groups were presented with different sequences of six concepts. For each concept, there was a learning phase, a testing phase and a feedback phase. The average time spent in each concept was 167±20 s.e.m. seconds, and the average duration of the task was 21±4 s.e.m. minutes. After moving through the learning, testing and feedback

[1]The learning times of all participants can be found in https://figshare.com/s/04d338adbbc4b1e83bf0.

phase of each of the six concepts, participants were asked if they used a pen or recorded the screen information in any way. They were also told that the answer to this question will not affect their payment, but that it was crucial for the experimenters to know.

During the learning phase, all 16 items were presented in the screen (in random order), and items belonging to the concept were identified with bold boundaries, as shown in Fig. 7.2. Participants were told that only the items with bold boundaries were 'blickets' (or 'tufas', etc.: we used different words for each concept in the sequence), and asked them to try to identify what a blicket was. During the testing phase, the 16 items were shuffled in the screen, and participants were asked to click on items that were blickets. If they made mistakes after submitting their answer, they were directed to the feedback phase, in which items that were incorrectly classified were indicated with a red cross. After having studied the feedback, participants were redirected to the testing screen, where items were reshuffled. When every item was correctly classified, participants were asked to give a verbal description of the concept and then continued on to the following concept after a resting period. We characterize the subjective difficulty of each concept as the time the participant spent in learning, testing and feedback phases for that concept (excluding the time spent in the verbal description).

Both groups (target and control), were exposed to 6 concepts. The second, third and fourth concepts are *training* concepts, and were different between both groups. The last two concepts are the *test* concepts, and were the same for both groups. The first concept was the trivial concept x_i for both groups, which was aimed to get participants started in the task. Importantly, variables (i.e. color lights inside objects in Fig. 7.2) were randomized for every concept, so paying selective attention to a specific variable across subsequent concepts was not beneficial for learning the concept sequence.

As shown in Table 7.1, we presented the target group with training concepts which are succinctly described when $\oplus$ is part of the language, but necessarily described with lengthier formulas if $\oplus$ is absent; more technically, concepts for which $MDL\mathsf{P}^{\oplus}$ is much smaller than $MDL\mathsf{P}$. We also corroborated that for $\mathbf{Con}_t^2$, $\mathbf{Con}_t^3$, $\mathbf{Con}_t^4$ and $\mathbf{Con}^5$ the number of formulas in $\mathsf{P}^{\oplus}$ with length strictly smaller than $MDL\mathsf{P}$ was at least 10 times greater than the number of formulas in P with length equal to $MDL\mathsf{P}$.

Participants in the control group, on the other hand, experienced a sequence of concepts that could be easily described using the language given by P. After these training concepts, both groups were presented with the same pair of test concepts: one which could be only succinctly described in $\mathsf{P}^{\oplus}$, and one for which the MDL did not depend on the underlying language $\mathsf{P}^{\oplus}$ or P. We compared learning times between the two groups for these last two concepts.

As shown in Table 7.1, training concepts for the target (xor) group were: x_i, $x_i \oplus x_j$, $x_i \oplus x_j \oplus x_k$, and $x_k \oplus x_l$, called $\mathbf{Con}^1$, $\mathbf{Con}_t^2$, $\mathbf{Con}_t^3$ and $\mathbf{Con}_t^4$ respectively. Training concepts for the control group were: x_i, $x_i \vee x_j$, $x_i \vee (x_j \wedge x_k)$, and $x_k \vee x_l$ called $\mathbf{Con}^1$, $\mathbf{Con}_c^2$, $\mathbf{Con}_c^3$ and $\mathbf{Con}_c^4$ respectively. We use the indexes *i, j, k, l* instead of numbers because variables were randomized in each trial. x_i could stand for x_1, x_2, x_3 or x_4, that is, for any of the four colors. After these four concepts, both groups were presented with the same test concepts: $x_i \wedge (x_j \oplus x_k)$, and $x_i \wedge (x_j \vee x_k)$, called $\mathbf{Con}^5$ and $\mathbf{Con}^6$ respectively.

Choosing which concepts to show the target group in order for them to 'learn' the $\oplus$ operator is critical in our experiment. Crucially, the learner must have an option between two alternatives that describe the concept: one that is succinct but uses $\oplus$, or necessarily a much longer one in the absence of $\oplus$. In other words, these concepts must be compatible with short logical formulas if and only if we take $\mathsf{P}^{\oplus}$ as the language of description. To

ensure that this was the case, we enumerated, for each concept, all formulas compatible with it and produced by the P and $\mathsf{P}^{\oplus}$ grammars up to length 19. For all training concepts of the target group, the shortest compatible formula without $\oplus$ is much longer than the shortest compatible formula with $\oplus$. This is shown in Table7.1.

7.4. Model-Free Results

We measure the subjective difficulty of a given concept as the total time needed by the participant to successfully encode the concept, which indicates that they can reliably express which exemplars belong to the concept and which do not.

Participants from the target group spent almost half the time than participants from the control group in $\mathbf{Con}^5$, which could be succinctly described only in $\mathsf{P}^{\oplus}$ (111 ± 16 s.e.m. seconds versus 214 ± 37 s.e.m. seconds, a two-sample t-test reveals $t_{42} = 2{,}6$, $P < 0{,}01$), as shown in Fig. 7.3 (a). We also found that the control group learned much faster $\mathbf{Con}^6$ (143 ± 14 s.e.m. seconds for the target group versus 76 ± 10 s.e.m. seconds for the control group, $t_{42} = 3{,}5$, $P < 0{,}01$). A mixed ANOVA with $\mathbf{Con}^5$-$\mathbf{Con}^6$ as within subject factor and target-control groups as between subject factor reveals a strong interaction between group and $\mathbf{Con}^5$-$\mathbf{Con}^6$ ($F = 15{,}3$, $P < 0{,}001$), indicating that the differences in learning times for $\mathbf{Con}^5$ and $\mathbf{Con}^6$ were very different between the two groups.

The target group encoded $\mathbf{Con}^5$ more efficiently than the control group. We propose that the control group expected to find in $\mathbf{Con}^5$ and $\mathbf{Con}^6$ structures that could be easily built in P. The target group, on the other hand, became biased towards the $\oplus$ structure, and they expected to find it in $\mathbf{Con}^5$ and $\mathbf{Con}^6$. This caused $\mathbf{Con}^5$ to be encoded more rapidly by the target group and $\mathbf{Con}^6$ more rapidly by the control group. Assuming that the

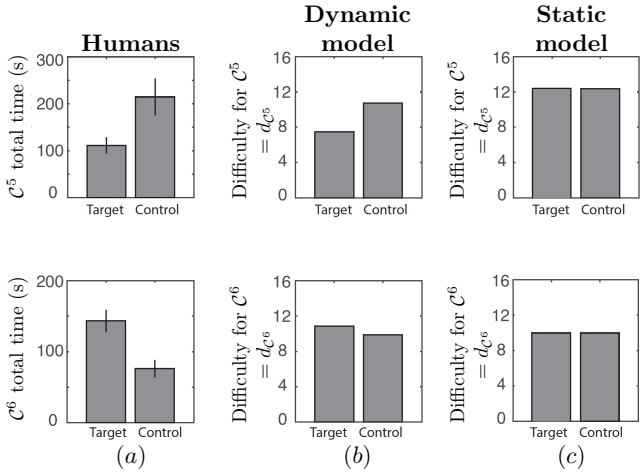

Figura 7.3: Concept learning time (a) and difficulty predicted (b), (c) for the two test concepts (**Con**^{5} and **Con**^{6}). Error bars are s.e.m. across subjects.

subjective difficulty of learning a concept is proportional to the complexity of its internal representation, we conclude that after exposure to the training concepts, participants in the target group represented the $\oplus$ more efficiently than the control group, and expected to find this structure in **Con**^{5} and **Con**^{6}.

7.5. Model

When presented with a concept (e.g. Fig. 7.2), our model generates logical formulas and evaluate them to true or false for that concept, keeping the formula only if it is true. To generate formulas, the model uses a symbolic language in which each rule (symbols and operators) is associated with a probability of being used. The probability of generating a formula is proportional to the product of the probabilities of the rules required for building it, and therefore it decreases exponentially with its length. Furthermore, if one of the rules has a very low probability of being used, formulas that require it will also have very low

probability.

The *Static model* maintains the rules' probabilities fixed throughout the concept sequence (the 6 concepts in Table 7.1). The *Dynamic model* updates the probabilities after each concept, in order to minimize the expected description length of future concepts, assuming they have similar structure to the concepts learnt so far. We include in this model the $\oplus$ rule a priori in the language, but with vanishing probability of being used. Changes in this probability can be analogously interpreted as the probability that a rational agent without the compiled symbol a priori decides to add the compiled expression as a new primitive into her language.

7.5.1. Static Model

Under the LoT assumption, given a concept **Con** (e.g. Fig. 7.2), the probability that an agent uses formula φ to explain this concept is defined by Bayes theorem:

$$P(\varphi \mid \mathbf{Con}) \propto P(\mathbf{Con} \mid \varphi)P(\varphi).$$

The likelihood $P(\mathbf{Con} \mid \varphi)$ of a logical statement φ can be simply defined as 1 if $[\![\varphi]\!] = \mathbf{Con}$ and 0 otherwise. In other words, for any given concept, only explanations that describe this concept are considered as possible explanations. The likelihood term has been defined more flexibly in the literature [Goodman et al., 2008, Piantadosi et al., 2016], allowing for mislabeled elements. We keep this simpler definition in order to reduce the number of free parameters of the model, as we do not intend to account for mislabeling errors in our experiment.

The prior $P(\varphi)$ is defined by augmenting the context-free grammars shown in Fig. 7.1 into a probabilistic context-free grammars (PCFG). In the PCFG, each rule has associated a parameter indicating the probability of using that rule. A PCFG can be used to produce logical statements similar to a CFG. Each non-terminal remaining in the statement is expanded using a rule of the PCFG with probability proportional to that rule's associated parameter, until no non-terminals remain in the statement.

We assume that the probability that a subject uses formula φ to explain concept **Con** is proportional to the posterior $P(\varphi \mid \textbf{Con})$, and the subjective difficulty $d_{\textbf{Con}}$ of a concept **Con** to a participant is proportional to the length of the formula that the participant is using to explain that concept. However, there is no way to know directly which internal formula φ the participant is using (and therefore we do not know $|\varphi|$). Hence, the most parsimonious approach is to consider the entire posterior distribution $\mathbf{P}(\varphi \mid \textbf{Con})$ over possible formulas.[2]

Given a concept **Con**, the expected length $E_{\textbf{Con}}$ of the formulas used by the participant is simply

$$E_{\textbf{Con}} = \sum_{[\![\varphi]\!]=\textbf{Con}} |\varphi| \; P(\varphi \mid \textbf{Con}), \tag{7.1}$$

where the sum is over all formulas φ compatible with **Con**. We define the difficulty $d_{\textbf{Con}}$ of a concept experienced by the participant as

$$d_{\textbf{Con}} \propto E_{\textbf{Con}} + \alpha N_{\textbf{Con}},$$

where we added a term that accounts for the cardinality of the concept: $N_{\textbf{Con}}$ is the cardinality of the concept or its complement, the one being smaller, i.e. $N_{\textbf{Con}} = \min\{\#\textbf{Con}, \#\overline{\textbf{Con}}\}$

[2]This is equivalent to the Sampling Hypothesis described in [Denison et al., 2013], by which participants represent distributions through samples. Similar results are obtained if each participant carries entire probability distributions.

(e.g. $N_{\textbf{Con}} = 4$ for the concept **Con** of Fig. 7.2), and α is a free parameter fitted globally for all concepts and participants to its maximum likelihood value of 0.9. In this way, we remove the asymmetry between positive and negative examples, while accounting for the toil taken by considering a larger number of items simultaneously.

In practice, to approximate $E_{\textbf{Con}}$ for each concept **Con**, we calculated the posterior probability $P(\varphi \mid \textbf{Con})$ of all compatible formulas φs up to size 19 with $P(\varphi \mid \textbf{Con})$ and then use (7.1). Since the space of all possible φs grows exponentially with $|\varphi|$, normative procedures for estimating $P(\varphi \mid \textbf{Con})$ in this space involve stochastic search algorithms. However, in our case, we were able to exhaustively enumerate and calculate the posterior probability of *all* formulas generated by the PCFG up to a sufficiently high size M such that all formulas with $|\varphi| > M$ have vanishing probabilities when compared to shorter compatible formulas for the current concept (because the prior $P(\varphi)$ decreases exponentially with the size of the formula).

7.5.2. Dynamic Model

Up to this point, we assumed that, given a concept **Con**, the posterior distribution over formulas $P(\varphi \mid \textbf{Con})$ was independent of the other concepts presented in the sequence. However, if the LoT (i.e. the PCFG) updates with experience, the prior $P(\varphi)$ in $P(\varphi \mid \textbf{Con})$ will change, and so will $E_{\textbf{Con}}$ in (7.1) and finally the subjective difficulty $d_{\textbf{Con}}$. Therefore, $d_{\textbf{Con}}$ will depend on the sequence of concepts that were previously presented to the participant.

In other words, since now $P(\varphi)$ depends on the sequence of concepts experienced by

the participant, instead of $P(\varphi \mid \mathbf{Con})$, we have

$$P(\varphi \mid \mathbf{Con}^t, \ldots, \mathbf{Con}^1) \propto P(\mathbf{Con}^t \mid \varphi) P(\varphi \mid \mathbf{Con}^1, \ldots, \mathbf{Con}^{t-1})$$

, where $\mathbf{Con}^t$ is the concept presented at trial t, and $P(\varphi \mid \mathbf{Con}^1, \ldots, \mathbf{Con}^{t-1})$ depends on the state of the PCFG at trial t, which in turn depends on how the PCFG gets updated from trial to trial.

Intuitively, the update process increases the probability of using a certain rule in the PCFG accordingly to how useful this rule was to compress compatible formulas for the concepts previously learned in the same domain. Specifically, we model the update process in a normative manner: the probability of using a rule of the PCFG at trial t is equal to the Bayesian posterior probability that this rule will enable the learner to find compressed explanations at trial t, according to how useful it was to compress explanations in trials $1, \ldots, t-1$.

To formalize the update of the PCFG, we define $P(\varphi)$ similarly to [Goodman et al., 2008]. Specifically, the prior probability of a logical statement at trial t in the concept sequence uses a single Dirichlet-multinomial for the set of rule expansions. The Dirichlet is parameterized by a set of positive real numbers D_i^t, one for each rule i in the PCFG, which in turn determine the probability of using rule i at trial t: a higher D_i indicates a higher probability of using rule i.

The prior is specified by the set Dirichlet parameters $\mathbf{D}^0$ with which we start the experiment ($\mathbf{D}^0$ represents a vector containing the prior parameters of all rules in the grammar at trial 0). In our experiment, we set the prior Dirichlet parameters of all rules equal to 1, and the parameter of the rule that expands the target operator to a value several

orders of magnitude smaller ($\approx 10^{-4}$). This means that the target operator was practically absent at the beginning of the experiment, but it was technically possible to 'learn it' by increasing its probability as the experiment developed.

Under the Dirichlet model, the prior $P(\varphi \mid \mathbf{Con}^1, \ldots, \mathbf{Con}^{t-1})$ can be rewritten using the Dirichlet parameters as $P(\varphi \mid \mathbf{D}^t)$. Therefore, to know how $P(\varphi \mid \mathbf{Con})$ updates from trial to trial, we only need to know how $\mathbf{D}$ updates from trial to trial.

The Dirichlet parameter of rule i at trial $t+1$ is equal to its parameter at trial t plus the amount of times the production i was used in generating all formulas compatible with the concept at trial t (we note $M_i(\varphi)$ as the number of times that rule i is used in generating formula φ), weighted by each formula's posterior probability at trial t:

$$D_i^{t+1} = D_i^t + \sum_{[\![\varphi]\!]=\mathbf{Con}^t} P(\varphi \mid \mathbf{D}^t)\, M_i(\varphi). \tag{7.2}$$

This Bayesian learning mechanism increases the probability of using rules that allow concepts to be succinctly described. This happens because these formulas have higher probability $P(\varphi \mid \mathbf{D})$ than longer formulas, so the Dirichlet parameters of the rules that build these formulas increase more strongly than those of the rules that build longer formulas.

7.6. Results

The Bayesian agent that minimizes the expected complexity of future concepts by optimally adapting its LoT to the inferred structure of the task accurately captures the dynamics of human learning across concepts. If we did not allow the model to update the probability of the operators after each concept, and particularly the compiled operator $\oplus$,

the control group and the target group would be indistinguishable to the model as it would predict equal average formula length for both groups (see Fig. 7.3, *Static Model*). Instead, as shown in Fig. 7.4, by adjusting the prior probabilities based on concept exposure the dynamic model is able to capture learning time patterns in the target groups ($R^2 = 0{,}96$ compared to $R^2 = 0{,}73$ for the static model). Expectedly, both models perform similarly in the control groups as they were designed to not encourage the use of any particular operator ($R^2 = 0{,}72$; $R^2 = 0{,}71$ for the static model). The impact of the learning capability of the model is most evident in the target group concept sequence, which was designed to this effect. If the structure of the concepts does not bias the LoT primitives one way or the other, it is expected that a static model will provide a reasonable fit. However, it is difficult to tell a priori how unbiased a set of concepts really is, so experiments relying on repeated concept exposure should always take between-concept learning into account.

Allowing the model to constantly update its beliefs from concept to concept is a requisite to capture human learning times. We now explain how the pattern of subjective difficulties in Fig. 7.4 emerged in the *Dynamic model*. In this scenario, learning for the model is formalized by the update of rule parameters from concept t to concept $t+1$ according to (7.2). In Fig. 7.5 we show how this learning takes place in the concept sequence for the target group. There are mainly two competing formulas when $\mathbf{Con}_t^2$ is presented: $x_i \oplus x_j$ and $(x_i \wedge \neg x_j) \vee (\neg x_i \wedge x_j)$. Given the low a priori value of the parameter of the $\oplus$ rule, the posterior of the formulas of type $(x_i \wedge \neg x_j) \vee (\neg x_i \wedge x_j)$, which do not use the $\oplus$ operator, is much higher than the posterior of $x_i \oplus x_j$. Therefore, in Fig. 7.4 we see a large predicted difficulty by the dynamic model for this concept (since the posterior lies mainly over these longer formulas without $\oplus$, see (7.1)).

However, the little increment in the $\oplus$ rule after $\mathbf{Con}_t^2$ (see Fig. 7.5) is sufficient for

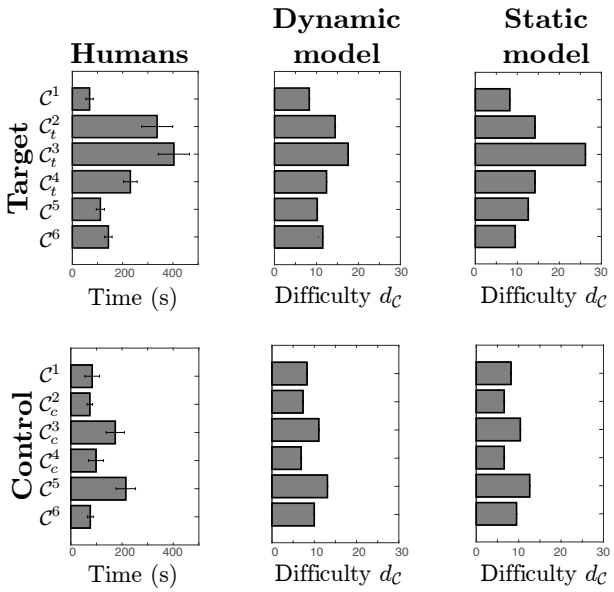

Figura 7.4: Learning times and model predictions for target and control groups (see Table 7.1 for concept details). The predicted difficulties of each model were calculated using $d_{\textbf{Con}}$. Error bars are s.e.m.

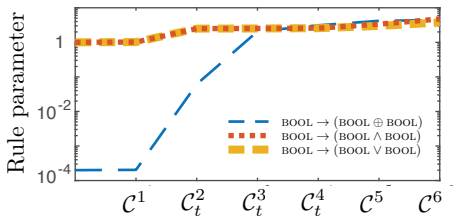

Figura 7.5: Evolution of Dirichlet parameters of different rules after each concept experienced by the target group.

making the formula $x_k \oplus x_l$ to have higher relative posterior in the next concepts, making the increment in the parameter of the $\oplus$ rule much greater than before. Additionally, the difficulty inferred by the model is much smaller the second time the concept is presented (compare $\mathbf{Con}_t^4$ and $\mathbf{Con}_t^2$ concepts in Fig. 7.4), since now the posterior is more evenly distributed between long (without $\oplus$) and short (with $\oplus$) formulas (see Eq. (7.1)). Finally, when the concept $\mathbf{Con}^5$ is presented, the learner has completely compiled the $\oplus$ rule into her language, ascribing the formulas that use the $\oplus$ operator a much higher posterior probability relative to the long formulas that do not use the $\oplus$ operator. Therefore, the inferred difficulty for $\mathbf{Con}^5$ is much smaller than those describing previous concepts, almost as simple as concept $\mathbf{Con}^1$ (see Fig. 7.4).

Finally, the strong $\oplus$ acquired by the target group increases the difficulty of $\mathbf{Con}^6$ relative to the control group (see Fig. 3). This occurs because there are several formulas of length 9 that use the $\oplus$ operator (around 6000), significantly increasing the expected difficulty of the concept (see Eq. (7.1)). For the control group, the posterior probability of these formulas is very low, causing a smaller increase in the expected difficulty.

The previous results point to a competition between different rules in the grammar. In our model, competition between $\oplus$ and the other operators is modulated by the initial relative value of the Dirichlet prior of the $\oplus$ rule, and the overall magnitude of the priors of all rules. The initial $\oplus$ prior measures how useful $\oplus$ should be (relative to the other rules) in order to increase the likelihood of using it in the future. If the $\oplus$ prior is too low relative to the priors of other rules, then formulas with $\oplus$ must be much shorter than formulas without $\oplus$ in order for them to have appreciable posterior and increase the $\oplus$ parameter in Eq. (7.2). In our experiment, if the prior is smaller 10^{-12} (and 1 for all other rules), then the predictions of the dynamic and static model for the target group are approximately equal:

the advantage of using $\oplus$ in the target concepts is not enough to increase the likelihood of using $\oplus$. On the other hand, if the $\oplus$ prior is too high, we cannot model the high difficulty of $\mathbf{Con}_t^2$ for the target group and the high difficulty of $\mathbf{Con}^5$ for the control group. For example, if the $\oplus$ prior is higher than 0.05 (and 1 for all other rules), the difficulty of $\mathbf{Con}_t^2$ and $\mathbf{Con}_t^4$ are approximately equal (corresponding to the short formula with $\oplus$) and also the difficulties of $\mathbf{Con}^5$ for control and target groups.

The other free parameter that modulates competition is the overall magnitude of the Dirichlet priors, which determines how many times an efficient rule should be encountered before incorporating it. If the magnitude is too high, then observing a useful rule does not significantly change its Dirichlet parameter relative to the others, eliminating from the model the rapid rule acquisition clearly showed by participants. This happens because in Eq. (7.2) the magnitude of the updates from t to $t+1$ are at most of order M, the number of times that operators appear in formulas with high posterior. In our experiment, if all rules have prior equal to 1 and $\oplus$ has 1/1000 we get similar results to the ones in Fig. 7.4, but if all rules have prior equal to 10000 and $\oplus$ has 10 the additions to the $\oplus$ parameter are insignificant, so the dynamic and static models make the same predictions for the target group.

In our model a large enough exposure to a concepts will increase the Dirichlet parameters without bounds, progressively decreasing learning flexibility. Although our experiment is not long enough to test it, such inflexibility is very unlikely to be true. For example, in the LoT fitting experiment from [Piantadosi et al., 2016] they found that human Dirichlet priors for most propositional operators are between 0.3 and 3, instead of orders of magnitude higher (as expected by Eq. (7.2) after exposure to a large number of concepts). Therefore, a more complete model of lifelong language acquisition should include an extra

normalization or forgetting parameter that decreases the overall magnitude of the Dirichlet parameters, preserving the high learning flexibility that we observed in our experiment.

7.7. Discussion

We measured the subjective difficulty that participants experience when learning a sequence of concepts. To explain this subjective difficulty, we resource to propositional logic as a base description language. In the target group we experimented with concepts which can be succinctly described in the base language *that also contains an extra operator* $\oplus$ for exclusive disjunction but that needed necessarily longer descriptions over the base language (where this operator is absent). On the contrary, the control group is exposed to concepts where $\oplus$ does not help to achieve succinctness.

Learning times are consistent with the hypothesis that participants in the target group smoothly adopt the $\oplus$ as a new primitive of their LoT in order to absorb the concepts they have been exposed to, with no more incentive than decreasing the expected complexity of future concepts. We do not claim that participants have learned the $\oplus$ operator defined by any specific formula using the previous operators, however, their LoT seems to have constructed an operation that matches the semantics of the exclusive or in order to compress such patterns of data and identify them more efficiently.

Here, we focus on transfer learning effects when learning sequential concepts that share the same hierarchical structure. We acknowledge, however, that several other transfer learning effects are present in human sequential logical concept learning, such as when subsequent concepts differ in the relevant variables (e.g. color lights in our experiment) [Blair et al., 2009], when changing the relevant variables in subsequent exclusive disjun-

ctions [Kruschke, 1996], or when two categories are learned in an interleaved or a focused manner [Carvalho and Goldstone, 2014]. However, unlike superficial knowledge about the task (like the frequency of appearance of different symbols and logical operators in the concept sequence), identifying the latent hierarchical structure of concepts have extremely important computational consequences: it allows for exponentially less complex representations [Bengio et al., 2013, Lake et al., 2015], maximizing the expected value of future computations within resource-bounded constraints [Gershman et al., 2015]. In our task, in order to focus primarily on the learning process of the $\oplus$ structure, we randomize variables in each trial, such that other kinds of transitions are averaged out across participants.

Most LoT studies provide a language that is fixed once trained or inferred over a specific data. We claim that when a specific language beats a second one at fitting some experimental data, what we may be seeing is an effect of prior experience (including from the experiment itself), more than an intrinsic feature of the LoT. This leads to a fundamental difficulty in trying to experimentally uncover what the actual human symbolic substrate of thought is. Experimental results have shown for instance that a grammar with *and, or*, and *not* better explains Boolean concept learning than one with *nand*, despite both being expressively equivalent [Piantadosi et al., 2016]. In our view, this cannot be taken to mean anything more than that in the current state of affairs of the world, the *nand* operator is not very useful for compressing information. We have shown that participants can rapidly compile new expressions in their LoT if they begin to be useful, which emphasizes that one cannot simply ignore the order in which concepts are presented to the participant when studying aspects of the LoT.

When Fodor proposed the Language of Thought hypothesis [Fodor, 1975], what he had in mind was a symbolic system we all came equipped with from birth. Stating that this

language is in fact always flexible might seem in outright contradiction with Fodor's original idea. In fact, what studies in the LoT literature (including this one) are probably probing is one among many languages in a hierarchy of increasing abstraction. As we progress in life, we find some conceptual summaries useful, and compiled them in a more abstract token. It is even likely that there is no proper hierarchy with sharply defined boundaries between levels, but instead a less organized progression of concepts of increasing abstraction, with thought progressing seamlessly using constructs at different levels.

7.8. Conclusion

We defined a model to measure the subjective difficulty of learning a sequence of concepts. The model updates the grammar production probabilities between concepts and predicts difficulty as the size of compatible formulas weighted by their posterior probability. This learning mechanism allows to simulate the emergence of a new primitive in the language, as it becomes useful to encode the concepts presented so far. The predicted difficulties strongly resembles the pattern of human learning times in a sequence of concepts that required the $\oplus$ operator in order to be efficiently represented.

Capítulo 8

Un marco lógico para estudiar aprendizaje de conceptos en presencia de explicaciones múltiples

Resumen

Cuando las personas buscan comprender conceptos a partir de un conjunto incompleto de ejemplos y contraejemplos, suele haber una cantidad exponencial de reglas de clasificación que pueden clasificar correctamente los datos observados, según las características de los ejemplos que se utilicen para construir estas reglas. Una aproximación mecanicista del aprendizaje de conceptos humanos debería ayudar a explicar cómo los humanos prefieren algunas reglas por sobre otras cuando hay muchas que pueden usarse para clasificar correctamente los datos observados. Aquí, explotamos las herramientas de la lógica proposicional para desarrollar un marco experimental que controle las reglas mínimas que son *simultáneamente* consistentes con los ejemplos presentados. Por ejemplo, nuestro marco nos permite presentar a los participantes conceptos consistentes con una disyunción *y también* con una conjunción, dependiendo de qué características se usen para construir la regla. Del mismo modo, nos permite presentar conceptos que son simultáneamente consistentes con dos o más reglas de diferente complejidad y que utilizan diferentes características. Es importante destacar que nuestro marco controla completamente qué reglas mínimas compiten para explicar los ejemplos y es capaz de recuperar las características utilizadas por el participante para construir la regla de clasificación, sin depender de mecanismos complementarios de seguimiento de la atención (por ejemplo, *eye-tracking*). Explotamos nuestro marco en un experimento con una secuencia pruebas competitivas como las mencionadas, e ilustramos la aparición de varios efectos de transferencia que sesgan la atención previa de los participantes a conjuntos específicos de características durante el aprendizaje.

La adquisición de conceptos es un aspecto clave y ampliamente estudiado de la cognición diaria humana [Cohen and Lefebvre, 2005, Ashby and Maddox, 2011]. Muchos investigadores han afirmado que un sistema de codificación y un conjunto de reglas subyacen a algunas de nuestras habilidades para adquirir conceptos [Nosofsky et al., 1994b, Tenenbaum et al., 2011, Maddox and Ashby, 1993], y se ha observado que parece que aprendemos conceptos de objetos con más facilidad cuando hay reglas 'más simples' que pueden explicar esas agrupaciones [Shepard et al., 1961, Nosofsky et al., 1994a, Rehder and Hoffman, 2005, Lewandowsky, 2011, Feldman, 2000, Blair and Homa, 2003, Minda and Smith, 2001].

En el mundo real, los humanos aprenden descripciones de conceptos mientras deciden simultáneamente a qué características atender [Schyns et al., 1998]; y el conjunto de características seleccionado generalmente determina la estructura y complejidad de las reglas mínimas que pueden describir el concepto. Por ejemplo, el concepto *perro* se puede explicar como *una mascota de cuatro patas que no es un gato* o como *un animal para caza, pastoreo, tira de trineos o compañía*. Ambas descripciones son totalmente compatibles con el concepto *perro*, pero nuestra experiencia nos induce a elegir diferentes características relevantes para definir el concepto. Mientras que la primera descripción de *perro* podría muy bien haber sido dada por un niño que tiene un perro en casa, la segunda podría haber sido presentada por un pastor o quizás un etólogo. Es probable que las características utilizadas para describir *perro* por cada agente les permitan describir de manera compacta el concepto, al mismo tiempo que lo separan de otros conceptos que se encuentran con frecuencia en su entorno. Aquí, preguntamos qué características usan los participantes para describir conceptos, dependiendo de la estructura lógica de la descripción que usa esas características y también de su exposición a conceptos anteriores. ¿Por qué alguien usaría

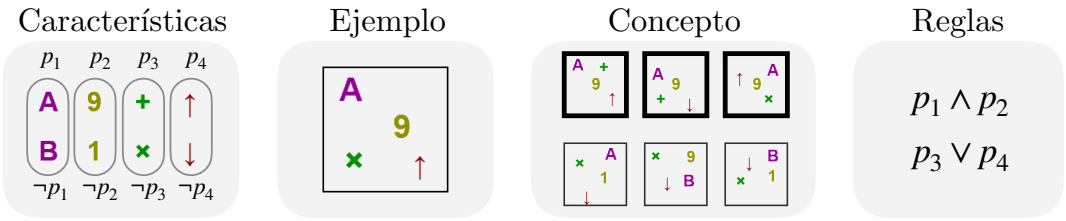

Figura 8.1: Ilustración de las características $\{p_1, p_2, p_3, p_4\}$, el ejemplo $(1, 1, 0, 1)$, y un concepto (los ejemplos positivos están marcados con marcos gruesos y los ejemplos negativos con marcos delgados). El concepto se puede explicar con las dos reglas mínimas $p_1 \wedge p_2$ o $p_3 \vee p_4$, dependiendo de las características que se usen para construir la regla (las dos primeras características o las dos últimas características, respectivamente).

gato o *caza* para definir *perro*?

En los experimentos de aprendizaje de conceptos proposicionales, a los participantes se les presenta un conjunto de *ejemplos*, cada uno conformado por N *features* proposicionales, que pueden tomar valores positivos o negativos. Por ejemplo, para $N = 4$ un ejemplo se puede representar lógicamente como el elemento $(1, 1, 0, 1)$, que toma valores positivos para la primera, segunda y cuarta características y negativos para la segunda, como ilustramos en la Figura 8.1. Un *concepto* puede entenderse intuitivamente como un conjunto de ejemplos, algunos de ellos marcados como pertenecientes al concepto y el resto marcados como no pertenecientes, es decir, ejemplos positivos y negativos. En la Figura 8.1 mostramos un ejemplo de un concepto *subdeterminado*, en el sentido de que, dado que no se muestra universo completo de ejemplos (es decir, las 2^4 posibilidades), diferentes conceptos determinados pueden ser coherentes con este conjunto más pequeño al extender el conjunto de ejemplos al universo completo.

Una *regla* consistente con el concepto es una fórmula lógica construida con las características y los operadores de conjunción ($\wedge$), disyunción ($\vee$) y negación ($\neg$), que se

evalúa como verdadera para objetos que pertenecen al concepto y falsa en caso contrario (por ejemplo, $p_1 \wedge p_2$, donde p_i es la i-ésima característica, ver Figura 8.1). La *longitud mínima de descripción* (*MDL* por sus siglas en inglés) de un concepto es la longitud de la regla más corta consistente con el concepto [Grünwald and Grunwald, 2007] (aquí, la *longitud* de una fórmula se define como número de apariciones positivos o negativos de símbolos proposicionales, más el número de apariciones de los operadores $\wedge$ o $\vee$ contenidos en él; por ejemplo, la longitud de $p_1 \wedge \neg p_3$ es 3, y la longitud de $(p_1 \wedge \neg p_3) \vee p_2$ es 5). Es importante destacar que la mayoría de los estudios sobre la dificultad subjetiva en el aprendizaje de conceptos están diseñados de manera que se pueda usar una *única* regla mínima para describir el concepto (por ejemplo, $p_1 \wedge p_2$) [Ashby and Maddox, 2005, Feldman, 2000], incluso cuando la dificultad de encontrar las características que componen esa regla (p_1 y p_2) se mide con mecanismos de seguimiento de atención (por ejemplo, [Blair et al., 2009, Hoffman and Rehder, 2010]). Esta limitación se debe posiblemente a la cantidad prohibitivamente grande de reglas que se pueden construir con un conjunto de características dado, lo que dificulta el control de las reglas que el participante podría usar al observar un conjunto de ejemplos. Por caso, para determinar la dificultad que tienen los participantes en aprender la regla lógica $p_1 \vee p_2$, es crucial controlar que ninguna otra regla de complejidad razonable pueda explicar el concepto (por ejemplo, $p_1 \wedge p_3$). En este trabajo, utilizamos las herramientas de la lógica proposicional para construir un marco experimental que nos permita presentar ejemplos consistentes con dos (o más) reglas elegidas, dependiendo de qué características se observen. Por ejemplo, el concepto mostrado en la Figura 8.1 es consistente con la explicación $p_1 \wedge p_2$ *y también* con la explicación $p_3 \vee p_4$, dependiendo de qué características se observen. En general, el experimentador puede elegir cualquier par de reglas que usen cualquier número de características (no superpuestas), y

nuestro marco garantiza que los ejemplos presentados solo son consistentes con las dos reglas mínimas elegidas por el experimentador. Luego, al presentar ejemplos novedosos que sean consistentes con solo una de las reglas anteriores, el experimentador puede determinar qué regla usaron los participantes internamente para aprender el concepto y, por lo tanto, a qué características atendieron.

Presentar las reglas A y B (por ejemplo, $p_1 \wedge p_2$ y $p_3 \vee p_4$) utilizando el mismo conjunto de ejemplos tiene varias ventajas experimentales sobre la presentación por separado de un conjunto de ejemplos coherentes con la regla A y luego un conjunto de ejemplos consistentes con la regla B. Algunas de las ventajas son:

(1) Cuando comparamos la dificultad relativa de aprender A y B en el mismo participante, si presentamos los ejemplos por separado, se complica superar los efectos de transferencia que hacen que la dificultad subjetiva dependa de la historia de conceptos aprendidos previamente en la tarea, y provoquen diferentes dificultades relativas si A se aprende antes de B en comparación a si B se aprende antes de A (ver por ejemplo [Tano et al., 2020]). El experimentador podría comparar los tiempos de aprendizaje para A y B entre los participantes, pero para reglas razonablemente estrictas, existen diferencias idiosincrásicas muy grandes en las dificultades de aprendizaje que aumentan enormemente la variación de los tiempos de aprendizaje (ver, por ejemplo, [Feldman, 2000]). Además, el experimentador no puede normalizar la historia pasada de cada participante antes del experimento. Por otro lado, presentar A y B simultáneamente a través del mismo conjunto de ejemplos nos permite medir directamente cuál de las dos reglas encuentra más fácilmente el participante, cuando las dos se presentan exactamente bajo las mismas condiciones experimentales.

(2) El hecho de que la regla A se aprenda más fácilmente que B cuando se presentan por separado no significa necesariamente que suceda lo mismo cuando se presenta en conjunto. Esto no podría ser válido si existiera una interacción entre los operadores lógicos que se están aprendiendo (que componen las reglas A y B) y el mecanismo de búsqueda utilizado para encontrar las reglas correspondientes. Por ejemplo, el mecanismo de búsqueda que permite a los humanos encontrar una regla de disyunción consistente con los ejemplos podría interactuar con el mecanismo que permite encontrar conjunciones, interacción que solo podría caracterizarse cuando la conjunción y la disyunción se presentan al mismo tiempo.

(3) Nuestro marco nos permite probar efectos de dificultad subjetiva de segundo orden (por ejemplo, la regla A se aprende más rápido si se presenta junto con la regla B que si se presenta junto con la regla C), así como efectos de aprendizaje de transferencia de segundo orden (por ejemplo, los participantes aprenden más rápidamente la regla C si primero han observado la regla A presentada conjuntamente con una regla arbitraria B_1, en comparación con junto con una regla diferente B_2).

(4) Si uno está interesado en qué características observa preferentemente el participante en una prueba determinada (por ejemplo, las características $\{p_1, p_2\}$ o $\{p_3, p_4\}$), simplemente se podría elegir la misma estructura lógica por A y B (por ejemplo, haciendo que A y B sean iguales a $p_1 \wedge p_2$ y $p_3 \wedge p_4$) y comprobar si el participante aprende A o B . Entonces, cualquier preferencia por aprender A sobre B solo podría deberse a una preferencia sobre las características en sí mismas ($\{p_1, p_2\}$), y no por la descripción lógica del concepto que usa esas características (esto es, $\cdot \wedge \cdot$).

Ilustramos estas ventajas en un experimento en el que a los participantes se les presenta

una secuencia de 6 pruebas, observando en cada prueba un conjunto de ejemplos consistentes con dos reglas alternativas. Ilustramos la ventaja (1) y (2) discutida anteriormente presentando una conjunción junto con una disyunción; y una regla simple junto con una regla compleja. Luego, mostramos que después de observar en varias pruebas que un subconjunto de características es útil para encontrar reglas concisas, inducimos en los participantes un sesgo para describir conceptos usando preferentemente esas características; este sesgo se probó aprovechando la ventaja (4).

8.1. Experimento

8.1.1. Participantes

El experimento se llevó a cabo como una tarea de Human Intelligence Task (HIT) en Mechanical Turk [Crump et al., 2013, Buhrmester et al., 2011, Stewart et al., 2015] de Amazon. Hubo 100 participantes, trabajadores autoseleccionados que vieron, aceptaron y terminaron el HIT publicado. Requerimos que los trabajadores tuvieran una tasa de aprobación HIT de $95\,\%$ o más. Se informó a los trabajadores que el pago por completar el experimento sería de 1,5 dólares estadounidenses, y que a 1 de cada 20 participantes se le asignaría aleatoriamente una bonificación de 10 dólares, independientemente de su desempeño en las tareas del experimento, siempre que terminaran el experimento (pero tener en cuenta que las pruebas no terminaron hasta que aprendieron correctamente cada concepto).

Para conocer los criterios de exclusión, consultar el apéndice §.1.

8.1.2. Configuración del experimento

La idea principal de nuestro marco experimental se esquematiza en la Figura 8.2. Los participantes observan un concepto *indeterminado*. Este concepto se presenta a los participantes como un conjunto de elementos que le pertenecen (ejemplos positivos), y un conjunto de elementos que no (ejemplos negativos). En la Figura 8.2, los elementos marcados como ejemplos positivos son los que están en la intersección de los dos conceptos y los ejemplos negativos son los que están fuera de ambos conceptos. Es importante destacar que la lista es incompleta, en el sentido de que no se muestran todos los elementos del universo. La idea fundamental es que, al extender el conjunto de ejemplos al universo completo, hay más de un concepto posible que es consistente con los ejemplos observados. Por ejemplo, en la Figura 8.2, los ejemplos presentados son consistentes con la regla mínima de C_1 (es decir, $\varphi_1 = p_1 \vee p_2$) *y también* con la regla mínima de C_2 (es decir, $\varphi_2 = p_3 \wedge p_4$). Como explicamos en el resto de esta sección, la elección adecuada de C_1 y C_2 puede aprovecharse para controlar las reglas mínimas que son consistentes con los ejemplos que observan los participantes.

El experimento real que implementamos consiste en una secuencia de 6 pruebas, cada una de las cuales está construida de esta manera. Ahora expandimos las 3 etapas que componen la i-ésima prueba del experimento. Para una mejor comprensión, consultar la Figura 8.3, que consiste en una vista esquemática de una prueba. Tenga en cuenta que esta figura es meramente ilustrativa y no pretende describir los detalles de una prueba, sino más bien la secuencia de fases y el flujo lógico dentro de una prueba. En particular, tener en cuenta que el número de elementos A, B, C y D en la figura no son significativos, ya que varían de prueba en prueba a lo largo del experimento. Los conceptos reales utilizados en cada ensayo, así como el número de ejemplos positivos y negativos se enumeran en

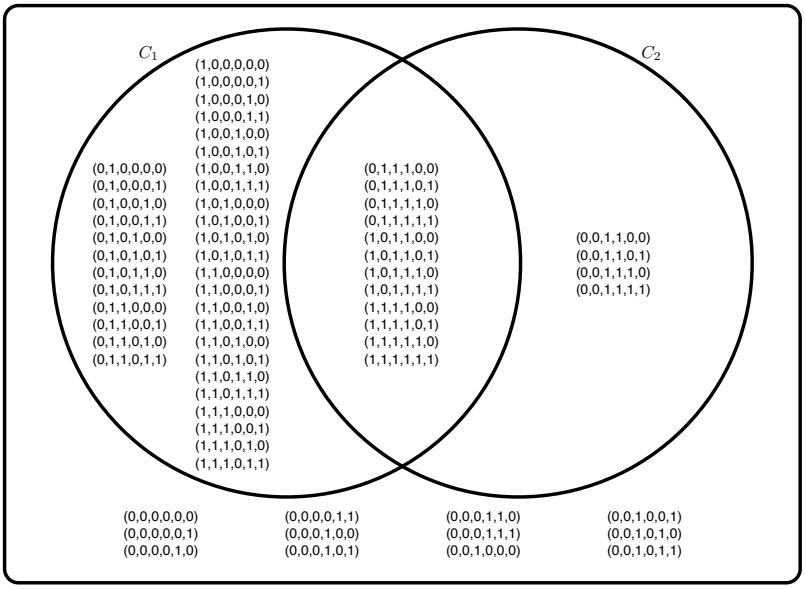

Figura 8.2: Un ejemplo de un par de conceptos C_1 y C_2 con 6 características. El concepto C_1 puede ser descrito por $\varphi_1 = p_1 \vee p_2$, y C_2 por $\varphi_2 = p_3 \wedge p_4$. Esta es solo una ilustración esquemática de dónde se coloca cada elemento (tupla) con respecto a los conceptos. Estos conceptos corresponden a los utilizados en la Prueba 1 del experimento real. Sin embargo, los elementos del experimento real no se representan de esta manera (es decir, como tuplas de ceros y unos).

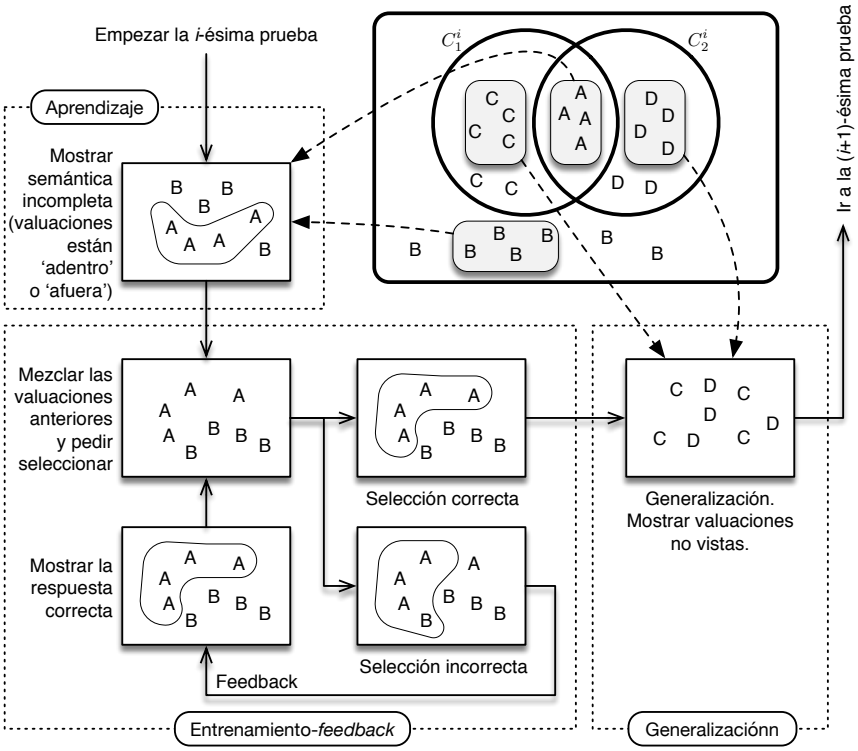

Figura 8.3: El esquema de nuestro marco experimental para estudiar el aprendizaje de conceptos en presencia de múltiples explicaciones. Ilustramos las tres fases que constituyen cada ensayo: fase de aprendizaje, fase de formación-*feedback* y fase de generalización. Los elementos se representan con las letras A, B, C y D (por ejemplo, las cuatro letras A en la intersección representan cuatro elementos diferentes en la intersección). El número representado de tales letras A, B, C o D es irrelevante (por ejemplo, habría 12 As y 4 Ds para los conceptos de la Figura 8.2).

la Tabla 8.1 (los grupos X, Y solo son relevantes para la Hipótesis III, por lo que pueden ignorarse por ahora), y se pueden encontrar más detalles de la implementación real en §8.2.2 y §8.2.3.

1. **Etapa de aprendizaje.** El participante se expone a un conjunto de elementos 'adentro' correspondientes a $C_1^i \cap C_2^i$ (marcados como ' sf A' en la Figura 8.3), y un conjunto de Elementos 'afuera' correspondientes al *complemento* de $C_1^i \cup C_2^i$ (mar-

cados como‘ B’ en la Figura reffig:trials).

A estos elementos mostrados los llamamos ‘ejemplos positivos’ y ‘ejemplos negativos’, respectivamente. Hay que tener en cuenta que esta información es incompleta, en el sentido de que no todos los ejemplos posibles se muestran al participante (ya que los únicos ejemplos que se muestran de $C_1^i \cup C_2^i$ son los de $C_1^i \cap C_2^i$). En el ejemplo ilustrativo de la Figura 8.2 (correspondiente a los conceptos de la Prueba 1 del experimento real), se mostrarían 24 elementos: los 12 ejemplos positivos en la intersección de C_1 y C_2, y los 12 ejemplos negativos fuera de C_1 y fuera de C_2. Se pide al participante que aprenda el concepto representado por ejemplos positivos.

Como demostramos formalmente en el Apéndice .3, el diseño experimental garantiza que solo hay dos reglas proposicionales (φ_1 y φ_2 en la Figura 8.2), mínimas sobre sus respectivos conjuntos de características, tales que: *(1)* son explicaciones *consistentes* con los ejemplos mostrados (esto es, satisfacen los ejemplos positivos pero no satisfacen los ejemplos negativos), *(2)* usan características diferentes entre sí (por ejemplo, $\{p_1, p_2\}$ en φ_1 y $\{p_3, p_4\}$ en φ_2) y, lo que es más importante, *(3) cualquier* regla consistente con los ejemplos debe usar un superconjunto del conjunto de características de al menos una de estas reglas mínimas. Por ejemplo, en la Figura 8.2 cualquier regla que solo use $\{p_2, p_3\}$ no puede explicar los ejemplos, ya que $(1, \mathbf{0}, \mathbf{1}, 1, 1, 1)$ es un ejemplo positivo, pero $(0, \mathbf{0}, \mathbf{1}, 0, 1, 1)$ es un ejemplo negativo. Cualquier regla que pueda explicar consistentemente los ejemplos debe mencionar un superconjunto de $\{p_1, p_2\}$ (por ejemplo, $\{p_1, p_2, p_3\}$) o un superconjunto de $\{p_3, p_4\}$. La prueba de esta condición se muestra en el Teorema 3, pero también lo esbozamos aquí. Observar que en la Figura 8.2 el ejemplo negativo $(0, \mathbf{0}, \mathbf{1}, 0, 1, 1)$ se construyó a partir del ejemplo positivo $(1, \mathbf{0}, \mathbf{1}, 1, 1, 1)$ invirtiendo los valores de p_1 y p_4, y hacerlo da como

resultado un elemento que es inconsistente tanto con φ_1 como con φ_2. Cuando una explicación alternativa deja sin usar algunas características p, q que aparecen en φ_1 y φ_2 respectivamente, debe haber algún elemento que satisfaga ambas reglas φ_1, φ_2, pero ninguna de ellas es satisfecha cuando se invierten los valores de p y q. Dado que el valor de verdad de la regla alternativa se mantiene cuando cambian características que no aparecen en ella, y dado que estamos mostrando como ejemplos positivos todos los elementos que satisfacen ambas reglas φ_1, φ_2 y como ejemplos negativos todos aquellos que no satisfacen ninguno de ellos, dicha explicación alternativa debe ser inconsistente con los datos mostrados.

Estas tres condiciones garantizan que el procedimiento experimental ilustrado en la Figura 8.2 es un método lógicamente sólido para presentar un concepto consistente con dos reglas mínimas elegidas por el experimentador (φ_1 y φ_2), dependiendo sobre qué características se basa el participante para construir la regla.

2. **Etapa de entrenamiento-*feedback***. Los *mismos* ejemplos de la etapa de aprendizaje se muestran al participante, pero esta vez sin indicar si son negativos o positivos y en orden aleatorio. Se le pide al participante que etiquete cada elemento como 'adentro' o 'auera', de la misma manera que se etiquetaron en el paso anterior. Si todos los elementos están clasificados correctamente, el participante pasa a la siguiente etapa. De lo contrario, se informa al participante sobre los errores en su etiquetado, y después de eso, la etapa de capacitación-*feedback* comienza nuevamente.

3. **Etapa de generalización.** *Los elementos no vistos anteriormente* se muestran al participante [1]. Estos elementos se toman de $C_1^i \setminus C_2^i$ y de $C_2^i \setminus C_1^i$ (aquí, '$\setminus$ ' denota la

[1] Con la excepción de la Prueba 6, donde un elemento se vuelve a mostrar para testear mejor la Hipótesis II. Ver §8.1.3.

diferencia de conjuntos). Estos elementos están marcados respectivamente como ‘C’ y ‘D’ en el esquema de la Figura 8.3. Se pide al participante que identifique aquellos elementos que corresponden al concepto aprendido en la etapa de aprendizaje. Después de hacerlo, comienza la siguiente prueba. Si el participante selecciona los de $C_1^i \setminus C_2^i$, el concepto aprendido en la etapa de Aprendizaje fue C_1^i, y si el participante selecciona los de $Ci_2 \setminus C_1^i$, el concepto que aprendieron fue C_2^i. Continuando con el ejemplo de la Figura 8.2, este proceso nos permitiría determinar si el participante estaba pensando en una regla con las características $\{p_1, p_2\}$ (es decir, $varphi_1$) o $\{p_3, p_4\}$ (es decir, φ_2) para explicar el concepto. Por supuesto, en la práctica, el participante puede seleccionar otros elementos, sin una justificación clara.

Una vez que el participante elige los elementos, se le pide que escriba una explicación de lo que constituye el concepto; esta respuesta no es parte del análisis de datos, excepto que nos permite excluir a los participantes que están usando métodos fuera del alcance del experimento (como tomar fotografías). Además, las respuestas escritas sirven como una *sanity check* adicional de si los participantes realmente están pensando de una manera consistente con el marco de la lógica proposicional (ver §.1 para las observaciones sobre las explicaciones escritas obtenidas en el experimento) .

Se pueden encontrar más detalles del experimento y su estructura en la Sección 8.2, particularmente en §8.2.2 y §8.2.3.

8.1.3. Ensayos experimentales

El conjunto de pruebas elegidas en el experimento (Tabla 8.1) tiene como objetivo revelar los sesgos que hacen que los participantes elijan un conjunto de características

sobre otro en este marco donde ambos conjuntos de características tienen sus propias reglas mínimas consistentes con los ejemplos observados positivos y negativos. Por ejemplo, en la Figura 8.2, ¿qué hace que los participantes elijan $\{p_1, p_2\}$ versus $\{p_3, p_4\}$ para explicar el concepto? Nuestra hipótesis es que un sesgo inductivo clave es simplemente la frecuencia con la que se utilizó previamente un subconjunto de características para explicar conceptos pasados. Denominamos este sesgo como *característica adherente*.

Prueba	**Grupo**	$\varphi_1^{\mathbf{i}}$	$\varphi_2^{\mathbf{i}}$	**Caract. mostradas**	**Hipótesis testeadas**				**Ejemplos mostrados #Positivos (#Negativos)**
					I	II	III	IV	
$i = 1$	X, Y	$p_1 \vee p_2$	$p_3 \wedge p_4$	p_1 to p_6	•			•	12 (12)
$i = 2$	X, Y	$\neg p_1 \wedge p_2$	$p_3 \vee \neg p_4$					•	12 (12)
$i = 3$	X	$p_1 \wedge p_2$	MDL15				•		10 (18)
	Y	$p_5 \wedge p_6$	MDL15						
$i = 4$	X, Y	$\neg p_5 \wedge p_6$	MDL15				•		10 (18)
$i = 5$	X, Y	$p_7 \wedge p_8$	MDL15	p_3 to p_8		•			10 (18)
$i = 6$	X, Y	$\neg p_7 \wedge \neg p_8$	$p_3 \wedge p_4$			•			4 (36)

Cuadro 8.1: Las pruebas del experimento. Aquí φ_1^i y φ_2^i representan los dos conceptos en competencia C_1^i y C_2^i en la i-ésima prueba (denotamos cada concepto por la regla proposicional más corta cuya semántica describe el concepto). Por "MDL15" denotamos un concepto cuya regla más corta es de longitud 15 (y está compuesta por tres símbolos proposicionales distintos de la regla en competencia en el ensayo correspondiente, ver §8.3.5 para más detalles). En todas las pruebas, el tamaño total del universo es de $2^6 = 64$, correspondiente a todos los elementos posibles sobre 6 características proposicionales. Indicamos cómo se dividió a los participantes en los grupos X e Y, que se usó solo para la Hipótesis III. También indicamos qué características se muestran en los ejemplos, qué hipótesis se testearon y el número de ejemplos positivos y negativos que se muestran en las fases de aprendizaje y entrenamiento para cada ensayo.

A continuación presentamos las principales hipótesis de este trabajo y su relación con las distintas pruebas experimentales.

Hipótesis I. En la Prueba 1, exploramos si los mismos factores que determinan la dificultad

en el aprendizaje de las reglas cuando se aprenden de forma aislada también determinan qué características usan los participantes al explicar un conjunto de ejemplos consistentes con dos reglas mínimas. En particular, es bien sabido que los conceptos que involucran conjunciones lógicas se aprenden más rápido que los conceptos que involucran disyunciones lógicas [Bourne, 1970].

En la Prueba 1, la regla mínima consistente es una disyunción si las características observadas son $\{p_1, p_2\}$, y una conjunción si las características observadas son $\{p_3, p_4\}$. Es importante destacar que, a diferencia de otros experimentos de aprendizaje de conceptos, tanto la disyunción como la conjunción de dos características son consistentes con el conjunto de ejemplos observado. Presumimos que el sesgo de aprendizaje que hace que la conjunción se aprenda más fácilmente que la disyunción también se trasladará a este marco si ambas explicaciones son posibles (utilizando características diferentes). Como se explicó antes, usamos la etapa de generalización de la Prueba 1 para determinar si los participantes entendieron el concepto usando $\{p_1, p_2\}$ (correspondiente a una disyunción) o usando $\{p_3, p_4\}$ (correspondiente a una conjunción).

Esta hipótesis fue prerregistrada como:

> En un escenario de dos posibles explicaciones para un concepto, una de las cuales puede ser modelada por el $\wedge$ lógico entre dos características y otra que puede ser modelada por el $\vee$ lógico entre otras dos características, la mayoría de la gente encontrará la explicación de $\wedge$ sobre la explicación de $\vee$.

Hipótesis II. El sesgo de *característica adherente* se testea en las Pruebas 5 y 6 del experimento. Una vez que los participantes han adquirido suficiente experiencia con la tarea, en la Prueba 5, los participantes encuentran un conjunto de ejemplos consistentes con

dos explicaciones mínimas, una muy simple que usa las características $\{p_7, p_8\}$ y otra muy compleja que usa $\{p_4, p_5, p_6\}$. Esto lleva a los participantes a explicar el concepto usando $\{p_7, p_8\}$, o de lo contrario tendrían que descubrir una explicación excesivamente compleja. Por lo tanto, planteamos la hipótesis de que en este caso la mayoría de los participantes seleccionarían las características $\{p_7, p_8\}$ [2].

En el siguiente concepto (Prueba 6), los participantes deben elegir entre explicaciones que utilizan las funciones previamente útiles $\{p_7, p_8\}$ u otro conjunto nuevo de funciones $\{p_3, p_4\}$. Suponemos que es más probable que los participantes expliquen el concepto usando $\{p_7, p_8\}$, solo porque estas características fueron útiles en el concepto anterior. Además, recordemos que las explicaciones que utilizan un conjunto de características que contienen $\{p_7, p_8\}$ o $\{p_3, p_4\}$ también son compatibles. Por ejemplo, en la Prueba 6, la explicación $p_3 \wedge p_4 \wedge \neg p_7$ es compatible con los ejemplos observados. También estamos interesados en estas reglas (por ejemplo, creemos que es más probable que los participantes usen $\{p_7, p_8, p_3\}$ que $\{p_3, p_4, p_7\}$). Los siete elementos elegidos para la etapa de generalización de la Prueba 6 nos permiten hacer precisamente esto: aparecen 7 elementos en la pantalla, con p_3, p_4, p_7, p_8 respectivamente iguales a $(1, 1, 1, 1)$, $(1, 1, 0, 1)$, $(1, 1, 1, 0)$, $(1, 1, 0, 0)$, $(1, 0, 0, 0)$, $(0, 1, 0, 0)$, $(0, 0, 0, 0)$. Estos elementos son respectivamente consistentes con las reglas mínimas $p_3 \wedge p_4$, $p_3 \wedge p_4 \wedge \neg p_7$, $p_3 \wedge p_4 \wedge \neg p_7 \wedge \neg p_8$, $p_3 \wedge \neg p_7 \wedge \neg p_8$, $p_4 \wedge \neg p_7 \wedge \neg p_8$ y $\neg p_7 \wedge \neg p_8$. Es importante destacar que ninguno de los elementos es coherente con más de una de las dos reglas mínimas.

Esta hipótesis fue prerregistrada como:

[2] Tener en cuenta que las características $\{p_5, p_6\}$ que se utilizaron en la Prueba 4 también aparecen la formula MDL15 de la Prueba 5. Sin embargo, planteamos la hipótesis de que la extrema complejidad de la explicación MDL15 sobrepasa el posible efecto de adherencia de características de la Prueba 4 a la 5. De hecho, encontramos que ninguno de los participantes utilizó la fórmula MDL15 en la Prueba 5.

> Si una persona ha utilizado un conjunto de características en la construcción de una explicación para un concepto, es más probable que también encuentre una explicación que contenga esas características en la siguiente prueba.

Hipótesis III. Abordamos la cuestión de si el sesgo de adherencia de características representa una ventaja computacional en sí mismo. Más concretamente, preguntamos si los participantes encuentran una regla coherente *más rápido* cuando están reutilizando las mismas funciones que en la prueba anterior. Tenga en cuenta que este es un fenómeno distinto al de la Hipótesis II, que se ocupa de la selección preferencial y no de los tiempos. Testeamos esta pregunta, independientemente del efecto del sesgo de adherencia de la característica, en las Pruebas 3 y 4 del experimento. En la Prueba 3, separamos a los participantes en los grupos X e Y. De la misma manera que en la Prueba 5, en la Prueba 3 el grupo X está predispuesto a aprender la regla usando $\{p_1, p_2\}$, y el grupo Y usando $\{p_5, p_6\}$. En la siguiente prueba (Prueba 4), los participantes están predispuestos a aprender la regla usando $\{p_5, p_6\}$. Suponemos que los participantes del grupo Y aprenderán el concepto C_1^4 más rápido que los participantes del grupo X, dado que están reutilizando las mismas características que usaron en la prueba anterior.

Esta hipótesis fue prerregistrada como:

> Cuando un concepto solo puede describirse razonablemente mediante un conjunto de características dado, una persona encontrará esta descripción más rápido si ese mismo conjunto de características le fue útil en la prueba inmediatamente anterior.

Hipótesis IV. Otra pregunta, testeada con las Pruebas 1 y 2, examina la fuerza relativa del sesgo de característica versus el sesgo del operador. Es decir, queremos determinar si hay

algún efecto fuerte que claramente desvíe la atención hacia las características (o, más bien, hacia los operadores) que previamente se han encontrado útiles para describir conceptos. Probamos esto cambiando el operador ($\vee$ / $\wedge$) que cada par de características puede usar para formar una regla útil en cada prueba, y luego comparando el número de participantes que explican los ejemplos mostrados de la Prueba 2 reutilizando las mismas funciones de la Prueba 1 frente a los que reutilizaron el operador pero utilizaron funciones diferentes.

Esta hipótesis fue prerregistrada como:

> En un escenario en el que tanto las características como los operadores se repiten de una prueba a la siguiente, habrá un efecto de adherencia que favorecerá a uno de ellos sobre el otro.

8.2. Metodología

8.2.1. Preregistración y datos

La metodología de este estudio, los procedimientos de recopilación de datos, el tamaño de la muestra, los criterios de exclusión y las hipótesis se registraron previamente en el Open Science Framework (OSF) antes de la recopilación y el análisis de los datos. Se puede acceder a la preregistración en https://osf.io/mgex3, mientras que los datos obtenidos y el experimento realizado por los participantes están disponibles en https://osf.io/gtuwp/.

En este trabajo también realizamos algunos análisis exploratorios (no prerregistrados): corregimos las explicaciones verbales que no eran consistentes con una interpretación

positiva del concepto para la Hipótesis I, excluimos los valores atípicos del análisis en la Hipótesis III, y consideramos el efecto del historial de aprendizaje del participante más allá de la prueba inmediatamente anterior en la Hipótesis II. También analizamos explícitamente, en este marco de múltiples explicaciones consistentes, la diferencia en la dificultad revelada entre reglas de longitud mínima muy diferente.

8.2.2. Detalles de representación

La estructura matemática subyacente de las pruebas utiliza variables proposicionales, valuaciones y conjuntos de valuaciones. Sin embargo, estos no se muestran de forma abstracta, sino que se representan mediante correspondencias con características (símbolos), elementos (cajas) y conceptos (colecciones de elementos).

A continuación, describimos los detalles de las representaciones utilizadas para el experimento y sus conceptos en competencia.

Características—variables proposicionales El experimento abarca ocho variables proposicionales: $p_1, \ldots, p_8$. Cada variable puede tomar uno de dos valores posibles, y estos valores están representados gráficamente por iconos. Por ejemplo, a p_1 se le puede asignar el icono 'A' o el icono 'B', que representan los valores 1 (positivo) y 0 (negativo) respectivamente, a p_3 se le puede asignar un icono '$+$' o el ícono '$\times$' que representa 1 y 0 respectivamente, y así sucesivamente.

La Figura 8.4 muestra los pares de valores para cada una de las ocho variables proposicionales. La asignación de pares de iconos a las variables proposicionales es aleatoria al comienzo del experimento y no varía dentro del experimento. La razón para elegir iconos

en lugar de valores (de color) 0, 1 es para evitar la posibilidad de aprender mentalmente un concepto usando 'conteo' u otros operadores que no están presentes en la lógica proposicional. Por ejemplo, mostrando valores $\{0, 1\}$ explícitos, una posible explicación para un concepto podría ser *más de 3 unos*, pero tal descripción sería mucho más difícil en la representación basada en iconos, ya que diferentes variables proposicionales carecen de símbolos en común. En §8.2.4 discutimos más detalles sobre estas consideraciones.

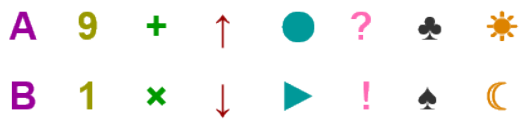

Figura 8.4: En la imagen de arriba se muestran las características, la representación visual de los valores positivos y negativos de las variables proposicionales. La fila superior representa valores positivos de las variables proposicionales, mientras que la fila inferior representa su negación.

Elementos (cajas)—valuaciones. Una valuación sobre las variables proposicionales se representa visualmente como un cuadrado/caja con los valores (iconos) de todas las variables proposicionales colocadas en posiciones aleatorias dentro de la caja. Llamamos a esta representación un 'elemento' (ver Figura 8.5 para ver un ejemplo de tal elemento). La razón para elegir esta representación es evitar sesgos direccionales que podrían influir en el aprendizaje y excluir el orden y otros operadores del lenguaje del pensamiento (consultar §8.2.4 para obtener más detalles). Cada vez que se muestra un elemento (en particular, dentro del ciclo en la etapa de entrenamiento-*feedback*) se elige una nueva posición aleatoria para las características proposicionales dentro de él.

Conceptos indeterminados—conjuntos de valuaciones positivas/negativas. El concepto que se muestra en la etapa de aprendizaje de una prueba corresponde a dos conjuntos

Figura 8.5: Un elemento. Esta caja que contiene características es la representación visual de una valuación sobre seis variables proposicionales. Aquí la caja aparece con un borde neutro, pero las cajas del experimento siempre aparecen con un borde que denota si son ejemplos positivos o negativos. La posición de los símbolos es irrelevante para los conceptos y se asigna al azar.

de valuaciones que no se superponen, y estos dos conjuntos no cubren todas las valuaciones posibles. Esto se representa como una secuencia de elementos 'adentro' y 'afuera', sin información sobre los elementos que no se muestran. En la etapa de aprendizaje, los elementos 'adentro' (ejemplos positivos) se representan como una caja con borde verde y los elementos 'afuera (ejemplos negativos) como una caja con borde rojo. Consultar la Figura 8.6 para ver un ejemplo de una secuencia etiquetada de elementos utilizados en la etapa de aprendizaje. Cada vez que se presenta el concepto, barajamos el orden en el que se muestran sus ejemplos positivos y negativos, pero siempre presentando todos los ejemplos positivos primero (además, a cada valuación se le asignan nuevas posiciones aleatorias para las características dentro de la caja correspondiente).

Conceptos (ocultos)—fórmulas. Sobre el conjunto completo de valuaciones, un concepto es simplemente el conjunto de valuaciones que lo describen positivamente. Los dos conceptos ocultos para cada prueba corresponden a las generalizaciones válidas y mínimas que se pueden hacer a partir de los conceptos incompletos. Pueden describirse como la semántica de las dos fórmulas proposicionales (reglas) que pueden usarse para

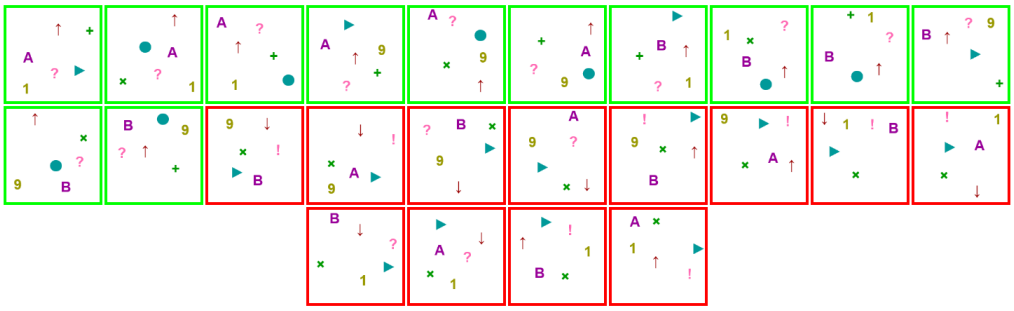

Figura 8.6: Una secuencia de ejemplos positivos y negativos en una etapa de aprendizaje, correspondiente a la Prueba 1. Un borde verde informa al participante que el elemento pertenece al concepto, mientras que un borde rojo informa que no pertenece al concepto. En este caso, los ejemplos podrían explicarse como 'cajas que contienen una flecha que apunta hacia arriba y un signo de interrogación' o como 'cajas que contienen un círculo o un signo más', pero hay que tener en cuenta que estas dos reglas determinan conceptos diferentes sobre el conjunto completo de posibles elementos.

explicar el concepto incompleto (ver Tabla 8.1); si bien estas reglas coinciden en el universo incompleto que se muestra en la etapa de aprendizaje, difieren en el conjunto de todas las valuaciones. Para obtener más detalles, recuerde el comienzo de §8.1.2 y su Item 1. Para obtener detalles técnicos, consultar §.3.

En la Tabla 8.2 resumimos la principal terminología lógica usada para definir la semántica formal, y su contraparte representacional adoptada en nuestra configuración experimental.

Santi: Parte (no toda) de esta tabla puede ser mandada al primer capítulo, dado que comparte mucho con el paper de PRE. habría que unificar nomenclatura entre PRE y BRM.

8.2.3. Details of the experiment's structure

As we explain in Section 8.1, each instance of the experiment consists of 6 trials where the participants must learn a concept from an incomplete universe. The presented

Terminología matemática	Terminología representacional
Valuación: una tupla $\overline{v} = (v_1, \ldots, v_n)$ donde cada v_i es 0 o 1.	**Elemento**: una caja con n símbolos dentro (ver Figura 8.5). Hay un código implícito en la Figura 8.4 (por ejemplo, $v_1 = 1$ se representa por una 'A' y $v_1 = 0$ se representa por una 'B', $v_3 = 1$ se representa por un '$+$' y $v_3 = 0$ se representa por un '$\times$', y así sucesivamente).
Variable proposicional: p_i toma el valor v_i bajo la valuación $\overline{v} = (v_1, \ldots, v_n)$.	**Característica**: p_i se representa, vía la codificación implícita, por uno de los pares de la Figura 8.4 dentro de un elemento que representa $\overline{v}$.
Concepto: un conjunto U de valuaciones que representa aquellas que son 'positivas' (por ejemplo, C_1 en la Figura 8.2). Notar que las valuaciones negativas son simplemente todas las valuaciones que no están en U. Observar que cualquier concepto U tiene una correspondiente **formula/regla** minimal φ_U que la caracteriza (es decir, φ_U es verdadera para las valuaciones en U, y es falsa sobre el complemento de U).	**Concepto**: cualquier categorización que divida el espacio de todos los elementos posibles en positivos (todos aquellos elementos que pertenecen a U) o negativos (elementos que no pertenecen a U).
Concepto indeterminado: un par $\langle U, V \rangle$ de conjuntos de valuaciones que representan los valores 'positivos' y 'negativos' respectivamente, de modo que $U \cap V = \emptyset$ y $U \cup V$ no es el conjunto de todas las valuaciones (por ejemplo, el par $\langle C_1 \cap C_2, \overline{C_1 \cup C_2} \rangle$ en la Figura 8.2). Observar que un concepto indeterminado $\langle U, V \rangle$ puede generalizarse de más de una forma mediante fórmulas (mínimas) φ_1 y φ_2 tales que *a)* φ_i $(i = 1, 2)$ es verdadera en todas las valuaciones en U, y falsa en todas las valuaciones en V, y *b)* el conjunto de *todas* las valuaciones positivas donde φ_1 es verdadera es diferente del conjunto de *todas* las valuaciones donde φ_2 es verdadera. Por ejemplo, el concepto indeterminado que se muestra en la i-ésima prueba del experimento se puede generalizar mediante las dos fórmulas mínimas correspondientes φ_1^i y φ_2^i	**Concepto indeterminado**: una secuencia de elementos positivos (borde verde) que representan U y elementos negativos (borde rojo) que representan V (consultar la Figura 8.6 para un ejemplo). Es importante destacar que U y V no cubren todo el universo de posibilidades que abarcan las funciones.

positive and negative examples are such that there are exactly two minimal rules (up to logical equivalence) in propositional logic that *1)* are consistent explanations for the shown examples; *2)* use disjoint sets of variables from each another; and *3)* any rule consistent with the examples must use a superset of the set of features of at least one of these minimal rules. This experimental setup will allow us to distinguish which of these rules best represents the way that the participant learned the concept. See §.3 for technical details.

Observe that merely asking the participant to select already seen elements does not give us any obvious insight into the internal process that derived into the learning of the concept; even if they internalized the concept using one of the two rules, it would remain uncertain which one they used, as both rules have the same semantics over the shown universe. In order to distinguish between these two cases, we use a generalization stage where previously unseen elements of the universe are shown, and the participant must select those that they believe belong to the concept. Of these new elements, some are consistent with only one of the rules, and other are consistent only with the other rule[3]. Furthermore, immediately afterwards we ask for a written explanation of what characteristics the participant thinks describe the concept.

Structurally, the experiment begins with the (hidden) assignment of the participant to one of two groups X or Y (see Table 8.1) and the exposition to a page with instructions. Afterwards, there are 6 trials with the following structure: they begin with a learning stage; they continue to a training stage where they get feedback if they fail to correctly select the elements that belong to the concept; a generalization stage where they must choose between elements of the universe that were not shown previously; and, in all but the last trial, a stage where the participants can rest between trials.

[3]The Trial 6 is an exception, and has an element that is consistent with both rules.

In what follows, we describe each stage of the experiment plus the introductory page, with a greater detail than that of §8.1.2.

8.2.3.1. Introduction and explanation

This is the page that subjects are shown at the beginning of the experiment. It describes the main task they will be asked to perform: that of learning from examples to distinguish what kind of 'boxes' belong to a certain concept. These elements are represented as a collection of 6 symbols, no more than one from a same pair. It is also informed that the position of the symbols does not matter. See Figure 8.5 for an example element.

When the subject indicates they have finished reading the instructions, they are sent to a fullscreen page with three multiple-choice questions whose purpose is to verify that the participant has understood the instructions; if they miss some answer, they are returned to the previous page and the cycle is repeated until they succeed.

If the participant answers correctly, they are now ready to begin, and the phases §8.2.3.2, §8.2.3.3, and §8.2.3.4 are then entered sequentially for each of the 6 trials.

8.2.3.2. The learning phase

In this phase of a Trial i, the participant is shown a set $S^i \subsetneq U^i$, a proper subset of elements from the current universe. Each universe syntactically corresponds to all the combinations of truth values for 6 propositional variables taken from the set $\{p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8\}$, thus spawning a set U^i of 64 elements. On the semantic side we call 'features' the visual representations of the propositional variables, and these representations remain fixed through the experiment (recall Figure 8.4).

The elements of S^i are shown as boxes, some of which have green border (denoting a positive example, that the element belongs to the concept), while the rest have red borders (denoting a negative example, that they do not belong).The green-bordered boxes are shown first, with the red-bordered ones appearing after the last box with green border. See Figure 8.6 for an example learning set.

If the graphical representations are abstracted away to the underlying basic structure, there are two propositional rules φ_1^i and φ_2^i (of minimum length in their class of logically equivalent rules, see Table 8.1) whose semantics correctly classify the positive and negative examples shown. If we call C_1^i, C_2^i the sets of valuations that satisfy φ_1^i, φ_2^i, respectively, we have that $S^i = (C_1^i \cap C_2^i) \cup \overline{(C_1^i \cup C_2^i)}$. The rules φ_1^i, φ_2^i use at most[4] 3 of the 6 propositional variables available in U^i, and the two rules do not have propositional variables in common.

When the participant believes they have learned which elements belong to the concept, they can click a button to proceed to the next stage.

8.2.3.3. The training–feedback phase

In this phase, the participant is shown a random rearrangement of S^i, with all the elements now surrounded by a red-bordered square. The subject must click exactly those elements (if any) they believe belong to the concept —changing them to a dotted green border (see Figure 8.7)— and then has to click a button to submit their choice.

If their selection is incorrect, the participant is shown which elements they misclassified (either by clicking them incorrectly or by failing to click them, see Figure 8.8). When they click a button to continue, they restart this stage (with a fresh randomization).

[4]The rules that are actually 'learnable' use exactly 2 propositional variables.

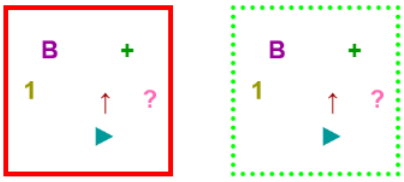

Figura 8.7: An unselected element, to the left, is represented by solid red borders. The same element in a selected state, to the right, is indicated by dotted green borders.

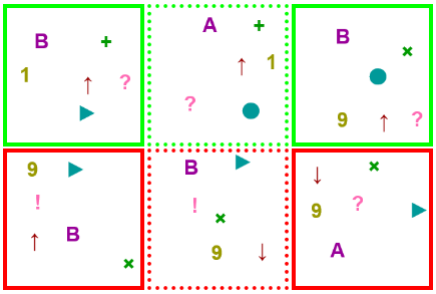

Figura 8.8: A partial section of the feedback resulting from a wrong selection. A solid green border means that the box was correctly selected as belonging to the concept. A solid red border means that it was correctly left unselected, meaning that it did not belong to the concept. A dotted green border means the box belongs to the concept but was not selected, and a dotted red border means that the box does not belong to the concept but was selected.

When the participant finally makes the correct selection, they continue to the next phase.

8.2.3.4. The generalization phase

In this phase, the participant is shown a subset of $U^i \backslash S^i$ (namely, in $(C_1^i \cup C_2^i) \backslash (C_1^i \cap C_2^i)$), that is, a selection of elements that were *not* present in the learning phase (hence nor in the training phase). The participant must classify which of these elements they think belong to the concept. The participant does not receive feedback on the choices they make here. Except for the sixth trial, part of these elements satisfy the rule $\varphi_1^i \wedge \neg\varphi_2^i$, while the rest satisfy $\varphi_2^i \wedge \neg\varphi_1^i$. Thus —assuming the participant learned the concept via a process akin to

a representation of one of the two rules— this phase crucially serves to distinguish which rule they have learned, if any.

After this selection, the participant is asked to submit a written explanation of what characteristics they think constitute the concept. This written explanation serves as an additional validation of whether they are thinking in a way describable by propositional logic according to our assumptions, or if rather they are using other methods (memorization, pen and paper, screenshots, other logics or formalisms, etc.).

8.2.4. Notes on the experiment design

The elements, universes, and rules that constitute our experiment are devised in terms of propositional logic. However, it is important to be careful with the semantics, i.e. the way elements are actually shown to the participants. We have to avoid giving more salience to the semantics of a propositional variable over the others, and it is imperative to select the semantics of variables in a way such that they do not share characteristics that might escape our propositional grammar: for example, if the propositional variables were represented as circles that can be distinctly colored or not, it would be quite natural to assume that counting colored or uncolored circles could provide information, but this option is not considered in a theoretical design that assumes only propositional operators to describe rules. A related consideration is that we must also avoid introducing other regularities extraneous to the propositional formulation: if the images corresponding to all propositional variables are always shown in a straight line in the same order, salience effects might appear *even if* we avoid semantics that become more expressive thanks to the ordered nature of the represented variables (such as with descriptions of the form *the first and last elements*

are of the same size).

Building adequate semantic representations for our logic. Taking these precautions into account, we choose to match each propositional variable with a particular image or figure, whose position in a square would be randomized (but avoiding superpositions). It is harder to decide exactly what would be the matching, but our final decision consists in matching each propositional variable with a set of two related Unicode characters (such as a triangle when the variable is 0, and a circle otherwise). See Figure 8.4 for the exact representations. We take care to choose different types of characters for different variables: having A, B for p_1 and Y, Z for p_5 is out as a possibility, since it naturally introduces counting of the type 'there is no more than 1 letter' and the like. Of course, this process is not fail-safe, as there are countless possible semantics associations that could introduce extra-propositional grammar into the experiment. But we try to minimize the chance that this happens easily or naturally, and we use the written explanation stage as a way to catch these exceptions if they occur[5].

Finally, to minimize possible salience effects from showing symbols that could have (despite our intentions to the contrary) different levels of conspicuousness, we randomize on a per-participant basis the assignment between pairs of symbols and propositional variables (but we do not randomize the assignment to the positive or negative value of a variable; the same Unicode characters are always positive in all randomizations, or always negative).

Ordering of positive and negative examples. As mentioned before, in the learning stage we shuffle the order in which their positive and negative examples are shown, but

[5] In the end, they did not occur. See §.1.

always presenting all positive examples first. Also, the number of positive examples is smaller or equal to the number of negative examples for all concepts (see Table 8.1).

The purpose of placing the positive examples first and having less positive examples than negative ones is to bias the participant into thinking of the concept by its positive formulation, instead of possibly thinking of a rule that would describe the negative examples, and then negating that rule to obtain the positive one. This becomes important when we want to reason about the ease of learning of different operators: the default assumption is that participants that correctly select positive examples of the concept are thinking the positive rule, which differs in its operator from the negative rule (by the De Morgan laws).

8.3. Results

8.3.1. Hypothesis I

We asked whether the conjunction-disjunction bias (which is known to affect learning times in the case of a single explanation [Bourne, 1970]) also determines which features are used to describe a concept when two alternative explanations are consistent with the observed universe. In the first trial, the observed examples were consistent with $p_1 \vee p_2$ and with $p_3 \wedge p_4$. As explained in §8.1.2, in the generalization stage we can determine if participants explained the concept using $\{p_1, p_2\}$ or $\{p_3, p_4\}$. We found that 77 of the 100 participants attended to $\{p_3, p_4\}$, which corresponds to an explanation that uses a conjunction. 11 participants attended to $\{p_1, p_2\}$ (corresponding to the use of a disjunction for the explanation), and 12 participants selected examples in the generalization stage inconsistent with both $p_3 \wedge p_4$ and $p_1 \vee p_2$. To test the significance of this result, we

performed a permutation test. Under the null hypothesis that participants randomly choose between explaining the concept using features $\{p_1, p_2\}$ and explaining it using $\{p_3, p_4\}$, the probability that 77 of the 100 participants attend to $\{p_3, p_4\}$ is $P < 10^{-12}$. Thus we conclude that the observed difference is significant.

Note that it is in principle possible that the participant learned the concept with a focus on negative examples (B's in Figure 8.3) instead of on positive examples (A's in Figure 8.3) (i.e. finding a correct explanation for the negative examples and then negating that rule to obtain an explanation for the positive examples).

As we mention in §8.2.4, we induced a bias to understand the concept in the appropriate way by first presenting the positive examples in the learning phase and by asking them to click on the positive ones in the training phase. We note, however, that 9 participants gave verbal explanations consistent with focusing on the negative examples. In this particular trial, a reverse interpretation is problematic since the negation of a conjunction corresponds to a disjunction, and the negation of the disjunction to a conjunction (i.e. $p \wedge q$ is logically equivalent to $\neg(\neg p \vee \neg q)$). Thus, a more comprehensive analysis should take into account participants' verbal explanations in this trial. However, even considering the worst-case scenario in which these 9 participants were originally regarded as part of the 'conjunction' group and they are now considered part of the 'disjunction' group, the conjunction-disjunction bias is still significant ($P < 10^{-7}$). We therefore conclude that, in this framework where multiple explanations are possible depending on the attended features, there is a bias favoring conjunctive explanations over disjunctive explanations.

8.3.2. Hypothesis II

Most participants understood the concept in Trial 6 using the same features $\{p_7, p_8\}$ used to describe the concept in Trial 5, even when the logical structure of the rule was exactly the same independently of attending to $\{p_7, p_8\}$ or to $\{p_3, p_4\}$[6]. To show this, we study participants' choices in the generalization stage of Trial 6 (see Figure 8.9).

Suppose that a participant is thinking of the rule $\neg p_7 \wedge \neg p_8$, thus they are only attending to features $\{p_7, p_8\}$ while ignoring the features $\{p_3, p_4\}$. Since $\{p_3, p_4\}$ are being ignored, the participant should mark those elements in which $\{p_7, p_8\}$ agrees with the rule $\neg p_7 \wedge \neg p_8$, irrespective of the values of $\{p_3, p_4\}$. That is, the participant should mark the elements with $\{p_3, p_4, p_7, p_8\}$ equal to $(0, 0, \mathbf{0}, \mathbf{0})$, $(1, 0, \mathbf{0}, \mathbf{0})$, $(0, 1, \mathbf{0}, \mathbf{0})$ and $(1, 1, \mathbf{0}, \mathbf{0})$. These elements have $\{p_7, p_8\}$ equal to $(0, 0)$ and 'anything' for $\{p_3, p_4\}$. On the other hand, if the participant is thinking of the rule $p_3 \wedge \neg p_7 \wedge \neg p_8$, then she is attending to $\{p_3, p_7, p_8\}$, and she should mark $(\mathbf{1}, 0, \mathbf{0}, \mathbf{0})$ and $(\mathbf{1}, 1, \mathbf{0}, \mathbf{0})$.

In general, by studying which of the 7 examples shown in Figure 8.9 (left) the participant selects in the generalization phase, we can deduce which features they were attending to (Figure 8.9, right). For example, all participants should mark the example with $\{p_3, p_4, p_7, p_8\}$ equal to $(1, 1, 0, 0)$, since it is consistent with all the logical rules irrespective of which features are used.

Indeed, as shown in Figure 8.9 (left), all participants selected this example. Although in practice the participant can select any of the 7 examples in the generalization stage, we found that all but five participants respected the rules of coherence illustrated in the

[6] As expected by our experiment design, 94 of the 100 participants understood the concept in Trial 5 using features $\{p_7, p_8\}$ (6 selected features with no clear rationale). Using features $\{p_7, p_8\}$ is indeed the only plausible way to learn the concept, given the high complexity of the alternative MDL15 formula.

previous paragraph. These 5 participants were 'one example away' of respecting the rule, however, we leave them out of the feature stickiness analysis, but including them does not change our conclusions. We also excluded 6 participants that selected elements with no clear rationale in the previous trial, since they may not have used features $\{p_7, p_8\}$. However, including these participants (and assuming they did use $\{p_7, p_8\}$ in the previous trial) does not significantly change the results. In total, these two exclusions leaves 89 participants for this analysis. The grey lines in Figure 8.9 (left) show simulations of agents that randomly select one of the seven possible subsets of features, and then proceed to select the examples consistent with the logical rule using that features. Participants responses (black line) were biased towards explanations using $\{p_7, p_8\}$, as predicted by the feature-stickiness bias. This can also be seen in Figure 8.9 (right), after inferring which features participants used to build the rule for the concept. In addition to being biased towards $\{p_7, p_8\}$, several participants explained the concept using all available features $\{p_3, p_4, p_7, p_8\}$. This shows that, in addition to the feature stickiness bias, when the number of features is relatively small, participants were also biased to describe the concept using all available features.

To quantify the feature stickiness bias, we assign a score to each participant according to the attended features in Trial 6 (deduced from the marked examples). The scores for the subsets $\{p_7, p_8\}$, $\{p_3, p_7, p_8\}$, $\{p_4, p_7, p_8\}$, $\{p_3, p_4, p_7, p_8\}$, $\{p_3, p_4, p_7\}$, $\{p_3, p_4, p_8\}$ and $\{p_3, p_4\}$ are 1, 2/3, 2/3, 1/2, 1/3, 1/3 and 0 respectively[7]. The average score for the 89 participants was 0.68 ($P < 10^{-6}$ in a permutation test with the null hypothesis of randomly attending to one of the seven subsets of features, which correspond to the grey lines in Figure 8.9), indicating a significant effect of the feature stickness bias. Although the

[7]Part (d) of the Analysis Plan section in our preregistration had a mistake in the use of features names: the learnable concept corresponding to the fifth trial uses p_7 and p_8, not p_3 and p_4 as erroneously written in that part; compare with the section on Study design, which matches Table 8.1.

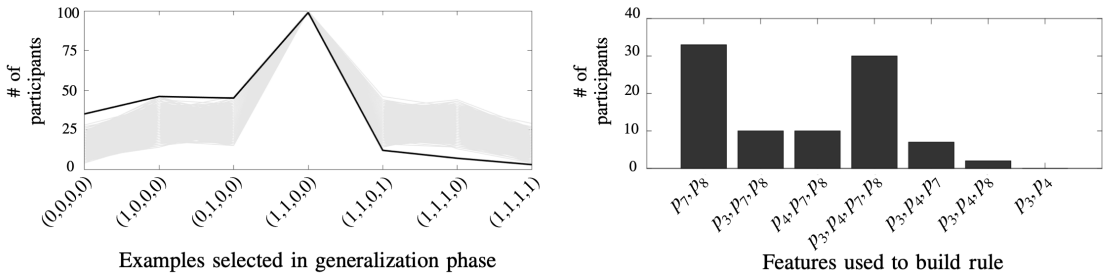

Figura 8.9: **(Left)** Number of participants (100 participants total) that, in the generalization stage of Trial 6, selected an element (possibly among others; the numbers add up to more than 100) with the elements written on the x-axis, indicating the values of the features $\{p_3, p_4, p_7, p_8\}$ respectively. As multiple choices were possible, the sum for all choices adds up to a value greater than 100. In grey we show 100,000 simulations in which 100 agents randomly attend to one of the seven subset of features (see text). **(Right)** From the selected objects in the generalization phase we can infer which features participants used to build the rule for the concept (89 valid participants, see main text).

feature stickiness bias was significant for both groups independently (Group X: average score 0.62, $P < 10^{-5}$; Group Y: average score 0.74, $P < 10^{-6}$), we found that feature stickiness was higher in Group Y (two-sample t-test comparing the scores of the two groups shows $t = 2{,}35$, $P < 0{,}05$). The only difference between the groups is that Group Y had already (artificially) experienced feature stickiness between the previous Trials 3 and 4, so they have already identified it as an useful bias for the task. This suggests that the entire concept-learning sequence can be important when studying learning biases.

8.3.3. Hypothesis III

This hypothesis regarded the behavioral advantage of the feature stickiness effect, which we tested by comparing learning *times* in Trial 4 for participants of Groups X versus Y (see Figure 8.10). If the feature stickiness bias represents a behavioral advantage, Group Y should learn concept C_1^4 *faster* than Group X. To avoid confounds due to inter-individual

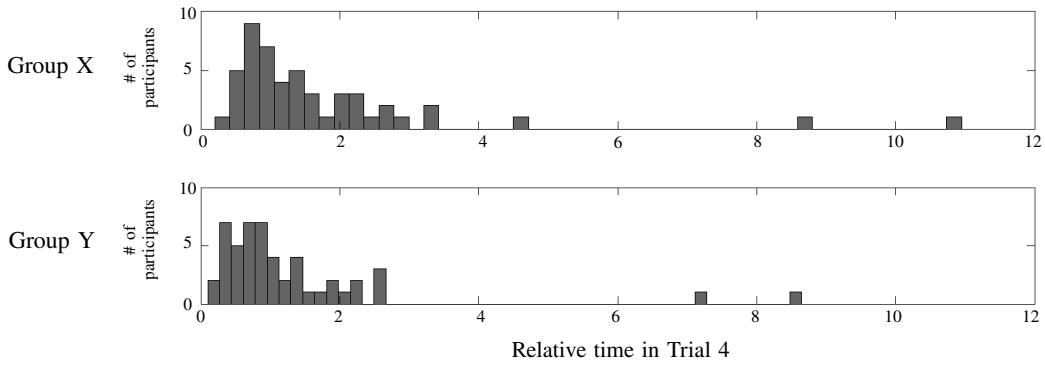

Figura 8.10: Relative time spent in Trial 4 by participants from the two groups, normalized by the time spent in Trial 5.

differences in absolute learning time, for this analysis we normalize individual learning times with the time spent in Trial 5, which uses different features than the previous concepts and should not be affected by any obvious inter-trial relation with previous concepts[8]. Thus we compare between the two groups (X and Y) the time spent in Trial 4 divided the time expended in Trial 5. This gives one number for each participant, and we compare the lists of numbers of the two groups using a two-sample t-test. The differences in the learning times between the groups are not significant if we analyze the data of all participants as shown in Figure 8.10 (two-sample t-test shows $t_{98} = 1{,}26$, $P = 0{,}2$; Cohen's $d = 0{,}25$), but they are significant if we rule out from this analysis 5 outliers that spent more than 5 times in concept 4 than 5, or in concept 5 than 4 ($t_{98} = 2{,}18$, $P < 0{,}05$, Cohen's $d = 0{,}42$)[9].

[8]Indeed, Trial 5 was pre-registered as a 'normalizer' trial.

[9]The ANOVA proposed in the pre-registration also did not reveal significant differences in learning times. For simplicity in the analysis of the outliers, we replaced here the ANOVA for a simple t-test between the normalized learning times of the two groups.

8.3.4. Hypothesis IV

The idea of this hypothesis is to test if participants prefer sticking to operators or sticking to features form one trial to the next. In this work we did not find conclusive evidence regarding this hypothesis. We suspect that the cause was an experimental setup that underestimated the strength of the bias favoring the $\wedge$ operator over the $\vee$ operator. We found that 77 of the 100 participants explained Trial 1 using $\wedge$, 11 explained it using $\vee$ and 12 selected elements in the generalization phase with no clear rationale. Of the 77 that used $\wedge$, 64 also used $\wedge$ in Trial 2, thus changing features but maintaining operator; and 7 of them used $\vee$, changing operator but maintaining features (the other 6 selected elements with no clear rationale). Of the 11 that used $\vee$, 10 used $\wedge$ in Trial 2, changing operator but maintaining features; and 1 of them used $\vee$ in the second trial. We realize, however, that a change from using $\vee$ in the first concept to $\wedge$ in the second one could not only be due to the effect of feature stickiness, but also simply to the stronger preference for $\wedge$. Thus without a precise quantitative knowledge of the prior preference of $\wedge$ over $\vee$, we cannot conclude about the effect of operator stickiness vs. feature stickiness. A future experiment could probe the existence of operator stickiness by having longer consecutive periods where feature reuse is not a useful bias and where only one logical operator remains useful for explaining a concept, before finally presenting a concept that can be explained via two different rules, each using different operators. Thus we leave for future work the task of studying the interaction between the feature stickiness bias and the precise structure of the logical rules being learnt.

8.3.5. MDL bias

The MDL-bias hypothesis posits that concept-learning difficulty increases with its MDL [Feldman, 2000]. In addition to their other roles, Trials 3 (group X and Y), 4, and 5 served to test this hypothesis in the new framework of multiple consistent explanations. In these trials, there were two possible explanations that were consistent with the shown data, one of much higher MDL than the other (15 vs. 3). For example, in the Group X of Trial 3, the short explanation was $p_1 \wedge p_2$, while the longer one was $((p_3 \vee (p_4 \vee p_5)) \wedge (\neg p_3 \vee ((p_4 \vee \neg p_5) \wedge (p_5 \vee \neg p_4))))$; the longer rule in other trials was always a substitution of features applied to this one (in order to keep the features disjoint between the two explanations). For these 3 trials, the responses of the 100 participants add to a total of 300 responses. From this total, 18 responses in the generalization phase did not choose objects consistent with any of the two explanations; 2 responses were consistent with the MDL 15 rule; and 280 responses were consistent with the MDL 3 rule. While this was expected by the experimental design (since we included a MDL 15 rule in those trials where we wanted to bias the participants into finding the other rule), we conclude that the MDL-bias hypothesis holds in this framework of multiple consistent explanations. Future work could explore in greater detail the relative difficulty of rules with slightly different MDL in this framework.

8.4. Discussion

In this work, we design an experimental framework in which participants observe an incomplete set of examples, which are consistent with two alternative minimal descriptions depending on which features are observed. We illustrate several advantages of our method compared to separately presenting sets of examples consistent with only one minimal

description at a time. First, we show that when a set of examples is consistent with a disjunction *and also* with a conjunction, participants are more likely to find the conjunction, in accordance with well-known previous results that show that the conjunction is learnt faster than the disjunction when presented separately [Bourne, 1970]. Then, we show that when rules of significantly different MDL are consistent with the observations, almost all participants discover the simpler rules, consistent with previous result showing that, when rules of different MDL are tested separately, learning times are proportional to MDLs [Feldman, 2000]. Finally, we show that when the logical structure of the minimal rules is independent of the selected features, participants are more likely to reuse the same features used to describe previous concepts, and preliminary results suggest that reusing features allows them to learn concepts faster than a control group that is not reusing features. To our knowledge this effect has not been previously characterized in the concept-learning literature, adding to the library of effects illustrating how human attention is biased towards features that are useful to describe the concepts (see [Blair et al., 2009, Kruschke and Blair, 2000, Kruschke et al., 2005, Hoffman and Rehder, 2010], among others).

Eye-tracking studies in categorization tasks have revealed that feature attention rapidly changes between trials depending on which features are relevant for classification in each trial [Blair et al., 2009], as well as depending on prior knowledge about feature relevance [Kim and Rehder, 2011]. In [Kruschke et al., 2005] it is found that eye movements confirmed that attention was learned in the basic learned inhibition paradigm, and in [Hoffman and Rehder, 2010] it is also found that eye movements revealed how an attention profile learned during a first phase of learning affected a second phase. Our experimental setup allows us to test an arguably simpler complementary hypothesis: everything else being equal, participants are biased to use the same features used in the past. Importantly, we were only

able to test this hypothesis thanks to our framework, which allows us to present a set of examples consistent with two rules of exactly the same logical structure, but using different sets of features. Then, without using eye-tracking, we can recover which rule the participants learned, and thus which set of features they attended to. Since the two sets of features explain the examples using exactly the same logical structure, preferentially explaining the concept using one set of features over the other can only be due to a preference over the features themselves, and not a preference over alternative logical structures.

Although some of the hypothesis that we test are aligned with the well-known Einstellung effect which states that adopted solutions may hinder simpler ones when aiming at tackling novel problems, our experimental setting is different to the classical water jar test (the most commonly cited example of an Einstellung effect, where participants need to discover how to measure a certain amount of water using three jars with different and fixed capacity) [Luchins, 1942] in two senses. First, we do not drive the experiment to control and supervise the aspects that participants have to pay attention to. On the contrary, our focus is on the *choice* of the features that show to be useful for learning a concept with more than one rational explanation. Second, our experimental framework is consistent with the Language of Thought (LoT) hypothesis [Fodor, 1975], which states that the human capacity to describe concepts —and, more generally, of all elements of thought— builds on the use of a symbolic and combinatorial mental language and it is specifically conceived to handle expressions in propositional Logic (but expansible to other formal languages), which is the ground where the rational explanations can be formalized. Such approach enables us to treat the notion of *feature* in a very precise way.

We note that other frameworks besides LoT can be used for our experiment. For example, consider similarity-based classification rules [**?**, **?**], where each feature is multiplied by a

weight and the classification rule is a function of the sum of the weighted features, usually a linear function with a soft decision boundary [**?**]. In this framework, the generalization phase would determine which of two possible decision boundaries was used by the participants (both consistent with the elements observed in the learning phase); and the feature-stickiness effect would be explained by the inertia of the weights' values from one concept to the next. However, two obstacles in this framework makes us prefer the LoT framework for Boolean concept-learning tasks. First, although a linear classification rule can readily learn the conjunctions and disjunctions in our experiment, more complex classification rules would require nonlinear functions of the features (e.g. the exclusive-or (XOR)). For nonlinear boundaries, the values of the weights that accompany the features could be hard to interpret, since it might no longer be true that a higher weight means higher feature importance. In contrast, in the LoT framework complex classification rules are compositionally built to accommodate concepts of any complexity, and feature importance can always be modeled as the probability of including a feature in a formula, independently of its complexity. Second, unlike similarity-based rules, the LoT framework naturally explains how humans can built verbal explanations for the learned concepts. Indeed, almost all participants gave informal explanations of conjunctions and disjunctions in propositional logic after learning each concept (see the shared data online for the list of verbal explanations).

Another well-studied phenomenon related to our work is Kamin's cue *blocking*, where the learning of a given stimulus B is *blocked* by the mere fact that it was preceded by a set of stimuli A that already pairs with the outcome. This shows that the subject learned that stimulus B was not useful, and hence disregards their attention to it in the upcoming events [Wagner, 1970, Mackintosh, 1975, Rescorla and Wagner, 1972]. Studied in humans in [Chapman and Robbins, 1990, Arcediano et al., 1997, Kruschke and Blair, 2000] among

others, our work differs from these approaches in that we never introduce a stage were a feature A is intentionally exposed in absence to B, in order to guide the attention of the participant.

We conjecture that most first-order determinants of subjective concept difficulty will also hold in a relative manner in our dual-concept setup, such as the MDL bias (for less extreme cases than evaluated in this work) [Feldman, 2003] and the transfer learning hierarchical structure bias [Tano et al., 2020]. Importantly, our experimental setup also allows to directly test second-order subjective difficulty effects (e.g. concept A is learnt faster if presented jointly with concept B than with concept C), as well as second-order transfer learning effects (e.g. participants learn more rapidly concept C if they have first observed concept A coupled with B_1, compared to A coupled with B_2). We believe that a systematic study of concept-learning difficulty with two (or more) concepts presented at the same time in each trial may open a new window into the dynamics of human concept-learning mechanisms. For example, consider the study in [Piantadosi et al., 2016], where participants gradually learn one concept while simultaneously selecting elements currently believed to belong to that concept. Here, the authors fit a Bayesian language model to participants' choices in order to illustrate how the posterior probability of the different rules in the grammar varied across time, to approximate the order in which different rules are learned. In contrast, using our experimental setting we can directly estimate, in a model-free manner, the probability that each rule is learnt faster than another. One simply needs to jointly present (in an incomplete and mutually compatible way) a set of examples consistent with those two minimal rules, and then measure the fraction of participants that discover each rule.

Usually, concept-learning biases have been studied in an isolated manner: the participant

observes examples indicated as inside or outside a *single* concept, and the experimenter evaluates its subjective difficulty for the participant. Although different methods have been used to present the concept to the participant (e.g. all elements at the same time [Tano et al., 2020, Kemp, 2012] or small sets of elements presented in series [Piantadosi et al., 2016]), to the best of our knowledge all previous category-learning studies have attempted to evaluate a single concept at a time. Here, we present a controlled logical setting to evaluate the relative difficulty of two concepts presented at the same time and under the same experimental conditions, and the framework could be generalized to more concepts straightforwardly.

Open Practices Statement. This study's methodology, data collection procedures, sample size, exclusion criteria, and hypotheses were preregistered on the Open Science Framework (OSF) in advance of the data collection and analysis, in order to ensure transparency, reproducibility, and rigour. The preregistration of this study can be found at `https://osf.io/mgex3`. The actual experiment as presented to the participants, together with all the experimental data analyzed, is available online at `https://osf.io/gtuwp/`.

Appendices

.1. Exclusion criteria and data processing

We decided to collect data for up to 3 weeks or until we reached a total of 100 participants. Via restrictions on the platform where the experiment was conducted, participants that took more than 4 hours or who did not complete all the trials were automatically excluded from the analysis. We were also prepared to exclude afterward the results from those participants whose verbal explanations denoted the use of external aids or methods outside the scope of the paper, such as using external help or taking screenshots of the concept, but there were no clear-cut cases of that behaviour ($N = 0$).

Additionally, while our preregistered exclusion criteria did not encompass the potential cases of written explanations that were legitimate but indicative of use of rules extraneous to propositional logic or to our semantic framework, in the end we did not detect any of these cases. This encouraging result is weakly indicative of the usefulness of our careful considerations for building adequate semantic representations, as mentioned in Section 8.2.4. For the comprehensive written explanations of the participants, we refer the reader to the uploaded raw data at https://osf.io/gtuwp/.

Balanced division into the two groups was handled via the psiTurk library, which decides the group a new worker will be assigned to, based on the current number of completed experiments in each group.

We ignored individual trails from participants that in the generalization stage chose a generalization inconsistent with any valid explanation (but this did not provoke the exclusion of other independent trials by the same participant). See Section 8.3 for details.

.2. Pilot

This experiment is informed by a previous pilot with 22 participants, which we executed in order to have some validation for our expected effects before making the preregistration. This pilot used more complex pairs of concepts, with a longer minimum description length for the two corresponding rules, and where using both $\wedge$ and $\vee$ in the same rule was often necessary. Originally, we expected a naturally arising separation into different groups, depending on the features of explanation found for the first trial. However, we encountered a very strong preference for explanations using solely $\wedge$, and this prompted various changes in the final design of the experiment that was preregistered in the OSF version.

More precisely, in our first trial in that pilot, 81 % ($N = 18$) of the workers explained the (incomplete) concept as a conjunction of three variables, while only 9 % ($N = 2$) explained it as a disjunction of two. This happened even though we had made the $\wedge$ explanation longer with the intention to compensate for the relative ease of $\wedge$ with respect to $\vee$ (so as to avoid getting a statistically inadequate number of participants self-selecting to the $\vee$ case). This result goes in line with known work about the relative hardness of learning concepts with the $\vee$ operator [Bourne, 1970]. In our framework of more than one plausible rule, a possible explanation to this population disparity could be that, when looking for common characteristics, it is natural to search first for individual features that always appear. Another explanation could be that, in a universe with low number of features, repetition of many of them becomes very salient, and thus the relation between hardness and number of conjunctions is not necessarily monotonic. In any case, this result was not part of the preregistration, so it is presented here only as an indication of an interesting effect to study.

.3. Technical results

Let us fix a non-empty set of propositional variables PROP. A valuation is formally defined as a function $v : \text{PROP} \to \{0, 1\}$ that determines the truth value of the propositional variables. A valuation can be extended in the standard way to preserve the usual semantics of Boolean operators and thus to determine the truth value of propositional formulas (which we call 'rules' in the context of describing concepts). We say that a valuation v satisfies a formula φ if $v(\varphi) = 1$. We say that a formula φ is a contingency if there exist a valuation v_t that satisfies it and a valuation v_f that does not.

Given a propositional formula φ, we define $\text{VAR}(\varphi)$ as the set of variables that appear in it. For example, if $\varphi_e = p_1 \vee (p_2 \wedge \neg p_2)$, then $\text{VAR}(\varphi_e) = \{p_1, p_2\}$.

We say that a formula φ is variable-minimal if there is no other formula ψ such that the truth values of φ and ψ coincide over all valuations and $\text{VAR}(\psi) \subsetneq \text{VAR}(\varphi)$. For example, the previous φ_e is not variable-minimal, since it is equivalent to $\psi = p_1$, which uses one less propositional variable.

We begin by proving a very basic lemma for illustrative purposes.

Lemma 1. *Let φ_1 and φ_2 be two contingencies such that $\text{VAR}(\varphi_1) \cap \text{VAR}(\varphi_2) = \emptyset$.*

Then there exists a valuation v_{in} such that v_{in} satisfies both φ_1 and φ_2, and a valuation v_{out} that satisfies neither φ_1 nor φ_2.

In other words, the lemma says that when we have two non-trivial concepts concerning non-overlapping sets of features, then there is at least one (positive) example that satisfies both concepts simultaneously and at least one (negative) example that satisfies none of them.

Demostración. Whether a valuation satisfies or not a formula φ depends only on how it evaluates propositional variables on $\text{VAR}(\varphi)$. Since $\text{VAR}(\varphi_1) \cap \text{VAR}(\varphi_2) = \emptyset$ and both formula are satisfiable via some v_1 and v_2 respectively, we can construct a valuation v_{in} by joining the values of v_1, v_2 on the (disjoint) sets of variables of each formula: $v_{in}(p) = v_1(p)$ if $p \in \text{VAR}(\varphi_1)$, $v_{in}(p) = v_2(p)$ if $p \in \text{VAR}(\varphi_2)$, and $v_{in}(p) = 0$ otherwise.

Similarly, since φ_1, φ_2 are not contingencies, there exist valuations $\bar{v}_1$ and $\bar{v}_2$ that do not satisfy φ_1 and φ_2 respectively. We use these valuations as before to construct a valuation v_{out} that does not satisfy φ_1 nor φ_2, as we wanted. □

Lemma 2. *If φ is a variable-minimal contingency, and $p \in \text{VAR}(\varphi)$, then there exists a valuation v such that v satisfies φ but $\tilde{v}$ does not, where $\tilde{v}$ is the single valuation that coincides with v except on p.*

Demostración. By way of contradiction, assume the conclusion does not hold: that for any valuation, its satisfaction of φ is independent of its value on p. In this case, necessarily $\{p\} \neq \text{VAR}(\varphi)$, or otherwise φ would not be a contingency (as it would always be true or always false).

Now consider V_φ the (non-empty) set of valuations that satisfy φ, and consider V_φ^{-p} its restriction to $\text{VAR}(\varphi)\backslash\{p\}$. From V_φ^{-p} we can construct, in a standard way via truth tables, a formula $\tilde{\varphi}$ with $\text{VAR}(\tilde{\varphi}) = \text{VAR}(\varphi)\backslash\{p\}$ such that a valuation v satisfies $\tilde{\varphi}$ if and only if $v|_{\text{VAR}(\tilde{\varphi})} \in V_\varphi^{-p}$. Since by assumption the value of p does not matter for φ, we have by construction that φ is equivalent to $\tilde{\varphi}$, but $\text{VAR}(\tilde{\varphi}) \subsetneq \text{VAR}(\varphi)$, which contradicts the variable-minimality of φ. □

The following theorem shows the general theoretical correctness of our experimental setup. It says that if we show as positive examples the full intersection of two non-trivial

concepts whose minimal descriptions contain no features in common, and show as negative examples the complement of the union of both concepts, any rule used to explain the seen (incomplete) concept must use a superset of the variables used to minimally describe one of these concepts. Otherwise, the chosen rule would be incompatible with the known data.

Theorem 3. *Let φ_1 and φ_2 be two variable-minimal contingencies such that* $\text{VAR}(\varphi_1) \cap \text{VAR}(\varphi_2) = \emptyset$. *Let ψ be a formula such that* $\text{VAR}(\psi) \cap \text{VAR}(\varphi_1) \neq \text{VAR}(\varphi_1)$ *and such that* $\text{VAR}(\psi) \cap \text{VAR}(\varphi_2) \neq \text{VAR}(\varphi_2)$. *Furthermore, assume that for all valuations v that satisfy $\varphi_1 \wedge \varphi_2$, v also satisfies ψ. Then there exist two valuations v_{in}, v_{out} such that:*

1. *v_{in} satisfies $\varphi_1 \wedge \varphi_2$*
2. *v_{out} does not satisfy $\varphi_1 \vee \varphi_2$*
3. *v_{in} and v_{out} both satisfy ψ.*

Demostración. From the hypotheses we know that there is a variable $p_1 \in \text{VAR}(\varphi_1) \backslash \text{VAR}(\psi)$ and a variable $p_2 \in \text{VAR}(\varphi_2) \backslash \text{VAR}(\psi)$. Since φ_1, φ_2 are variable-minimal contingencies, from Lemma 2 we have that there exist valuations v_1 and v_2 such that they satisfy φ_1 and φ_2 respectively, but where $\tilde{v}_1$ and $\tilde{v}_2$ do not, with $\tilde{v}_1$ and $\tilde{v}_2$ being the valuations that coincide with v_1 and v_2 save on p_1 and p_2 respectively. Using that $\text{VAR}(\varphi_1) \cap \text{VAR}(\varphi_2) = \emptyset$, we can construct from v_1 and v_2 (as we did in the proof of Lemma 1) a valuation v_{in} such that v_{in} satifies both φ_1 and φ_2, and also such that v_{out} does not satisfy neither of them, where we take v_{out} to coincide with v_{in} save on p_1 and on p_2. From the hypothesis, necessarily v_{in} satisfies ψ. However, since $\{p_1, p_2\} \cap \text{VAR}(\psi) = \emptyset$, the value over p_1 or p_2 does not matter for the satisfaction of ψ, and thus v_{out} also satisfies ψ, as we wanted to see. □

Note that the statement of Theorem 3 can be generalized to any number of non-trivial rules $\varphi_1, \ldots, \varphi_n$ such that $\text{VAR}(\varphi_i) \cap \text{VAR}(\varphi_j) = \emptyset$ for all $i \neq j$, and with ψ such that $\text{VAR}(\psi) \cap \text{VAR}(\varphi_i) \neq \text{VAR}(\varphi_i)$ for all i. This means that we can test concept learning under any multiplicity of possible explanations, as long as the underlying propositional universe is large enough and the rules are chosen adequately.

Capítulo 9

BORRAR: A theory of memory for binary sequences: Evidence for a mental compression algorithm in humans

A theory of memory for binary sequences:

Evidence for a mental compression algorithm in humans

Samuel Planton[1], Timo van Kerkoerle[1], Leïla Abbih[1], Maxime Maheu[1,2], Florent Meyniel[1], Mariano Sigman[3,4,5], Liping Wang[6], Santiago Figueira[4,7], Sergio Romano[4,7], Stanislas Dehaene[1,8]

[1] Cognitive Neuroimaging Unit, CEA, INSERM, Université Paris-Sud, Université Paris-Saclay, NeuroSpin center, 91191 Gif/Yvette, France

[2] Université de Paris, 75006 Paris, France

[3] Laboratorio de Neurociencia, Universidad Torcuato Di Tella, Buenos Aires, Argentina

[4] CONICET (Consejo Nacional de Investigaciones Científicas y Tecnicas), Argentina

[5] Facultad de Lenguas y Educacion, Universidad Nebrija, Madrid, Spain

[6] Institute of Neuroscience, Key Laboratory of Primate Neurobiology, CAS Center for Excellence in Brain Science and Intelligence Technology, Chinese Academy of Sciences, Shanghai, 200031, China

[7] Universidad de Buenos Aires. Facultad de Ciencias Exactas y Naturales. Departamento de Computacion, Buenos Aires, Argentina

[8] Collège de France, 11 Place Marcelin Berthelot, 75005 Paris, France

Correspondence concerning this article should be addressed to Samuel Planton, Cognitive Neuroimaging Unit, CEA, INSERM, Université Paris-Sud, Université Paris-Saclay, NeuroSpin center, 91191 Gif/Yvette, France

Contact: samuel.planton@cea.fr

Abstract

The capacity to store information in working memory strongly depends upon the ability to recode the information in condensed form. Here, we tested the theory that human adults encode binary sequences of stimuli in memory using a recursive compression algorithm. The theory predicts that the psychological complexity of a given sequence should be proportional to the length of its shortest description in the proposed language, which can capture any nested pattern of repetitions and alternations. Five experiments examine the capacity of the theory to predict human adults' memory for a great variety of auditory and visual sequences. We used a sequence violation paradigm in which participants detected occasional violations in an otherwise fixed sequence. Both subjective complexity ratings and objective violation detection rates were well predicted by our theoretical measure of complexity. While a simpler transition-probability model accounted for significant variance in the data, the language model dominated over the transition probability model for long sequences whose number of elements far exceeded the limits of working memory. Model comparison also showed that shortest description length in a recursive language provides a better fit than a variety of previous encoding models for sequences. The data support the hypothesis that, beyond the extraction of statistical knowledge, human sequence coding relies on an internal compression using language-like nested structures.

Keywords

Sequence processing; language of thought; complexity; novelty detection; statistical learning

Sequence processing, the ability to encode and represent in memory a temporally ordered series of discrete elements, plays a central role in numerous human activities, including language. In the 1950's, Karl Lashley (1951) and Noam Chomsky (Chomsky, 1957) famously argued that the sequential structures that humans can produce and remember cannot be viewed as mere associative links between consecutive item, but must be mentally represented as nested structures – the syntax of language, for instance, involves potentially unlimited embeddings of phrases within phrases. The goal of the present paper is to introduce a precise theory of sequence representation for the much simpler case of binary sequences, i.e. sequences informed by two elements A and B (e.g. high and low pitch, or red and green dots). We present experimental evidence that, even in this case, a similar postulation of nested structures is required in order to account for human memory performance.

Understanding how humans and other animals encode and represent temporal sequences has recently emerged as a crucial issue in the study of comparative cognition, as it allows a direct comparison between species (Dehaene et al., 2015; Wilson et al., 2017). Recursive phrase structures have been proposed to lie at the core of the human language faculty (Hauser et al., 2002), and a competence for nested trees has been postulated to underlie several other human cognitive abilities such as mathematics or music (Conway & Christiansen, 2001; Dehaene et al., 2015; Fitch, 2014; Hauser & Watumull, 2017). According to a recent review (Dehaene et al., 2015), non-human animals may encode sequences using a variety of encoding schemes, including transition probabilities, ordinal regularities (what comes first, second, etc.), recurring chunks, and algebraic patterns (Fujii, 2003; Jiang et al., 2018; Marcus et al., 1999; Wang et al., 2015; Wilson et al., 2013). However, several authors hypothesize that only humans have access to a language-like representation

of nested trees (Dehaene et al., 2015; Fitch, 2014), also being described as a "universal generative faculty" (Hauser & Watumull, 2017) or "language of thought" (Amalric et al., 2017; Fodor, 1975) capable of encoding arbitrarily nested rules.

Here we propose a precise language capable of encoding any arbitrary nesting of repetition and alternation structures, and we test the hypothesis that humans spontaneously encode sequences using the nested tree structures of this language. We do so using the simplest form of temporal sequences, namely binary sequences. Indeed, as opposed to more complex sequences, such as the ones of the natural language, which involve numerous factors that are difficult to properly control (prior knowledge, semantic content, word frequency, etc.), binary sequences allow to easily control the information content of the input. Furthermore, they are potentially accessible to a wide variety of populations beyond human adults, including infants and non-human primates. Finally, binary sequences are also widely used to study the cognitive processes and brain mechanisms involved in the perception of randomness and in statistical learning (Falk & Konold, 1997; Griffiths & Tenenbaum, 2003; Huettel et al., 2002; Maheu et al., 2019; Meyniel et al., 2016; Oskarsson et al., 2009). While minimal, they nevertheless preserve the possibility of forming structures at different hierarchical levels, from simple chunking to language-like rules, and thus of arbitrating between different models of sequence encoding.

A short review of theories and experiments on sequence complexity

The concept of compression in working memory has a long history. Much research shows that human memory is not simply determined by the number of words, digits or locations that must be remembered, but also by their capacity to be "compressed" into a smaller number of known phrases, groups, or chunks (Brady et al., 2009; Chase & Ericsson, 1982; Cowan, 2001; Ericsson et al., 1980; Feldman, 2000; Gilchrist et al., 2008; Miller, 1956). The apparent discrepancies between the different limits of working memory capacity proposed in the past, e.g. 7 ± 2 items (Miller, 1956) versus 4 items (Baddeley & Hitch, 1974; Cowan, 2001) can indeed be reconciled if one takes into account the possibility of constituting chunks rather than encoding a complete series of individual items (Mathy & Feldman, 2012). The formation of chunks can be seen as a data compression process, and it was proposed that the complexity of a sequence can be defined as the size of its most compressed representation (Chater & Vitányi, 2003; E. L. Leeuwenberg, 1969; Mathy & Feldman, 2012; Simon, 1972).

Experimentally, half a century of behavioral studies have shown that accuracy in sequence encoding and production tasks varies according to the complexity or compressibility of the sequence. Glanzer and Clark (1963) already proposed to use the length of the most compact internal description of a sequence as a measure of its complexity. They found that the number of words that participants used to describe an array of eight symbols, each colored either in black or in white, was correlated with the accuracy in reproducing it. Such *mean verbalization length* (MVL) predicted behavior better than a simple count of the number of runs in the sequence (e.g. "AAABBBAA" has three runs), particularly

for the “ABABABAB” , which could be simply described as “alternating” .

Generalizing upon this early work, one may propose that the complexity of a sequence relates to the length of its compressed form when it is recoded using an internal language. Consistent with such idea, Restle and Brown (1970) showed that participants learned a series of 10 button presses, not as an associative chain of elements, but by encoding it as an abstract pattern, defined as the set of rules that were needed to generate it. The profile of errors suggested that participants represented the sequences as hierarchical trees of embedded rules (i.e., repetition, transposition, mirroring), equivalent to the tree structures found in language (Restle, 1970). The psychological reality of this proposal was strengthened by showing that performance decreased precisely at the boundaries of higher hierarchical level groups of elements (Restle, 1970, 1973; Restle & Brown, 1970). However, this approach was not developed into a full-blown universal language explaining how any sequence or pattern would be encoded.

A more formal approach for estimating the complexity of patterns, usually referred to as algorithmic complexity, program size complexity or *Kolmogorov complexity* (KC), was proposed by Kolmogorov (1965), Chaitin (1969) and Solomonoff (1964), within the framework of Algorithmic Information Theory. These mathematicians defined the complexity of a sequence as the length of the shortest computer program capable of producing it. Strictly speaking, the algorithmic complexity is defined relative to a specific descriptive language (or programming language). When this language is Turing complete –which means one can simulate any other Turing machine on it - we talk about the universal or plain KC. Unfortunately, since it is impossible to determine whether any Universal Turing machine will halt or not, KC is not computable. However, when the encoding language has reduced expressive power, the algorithmic complexity can be calculated and used as

a subjective measure of complexity even if it no longer implies a universal measure of complexity for any two sequences (Romano et al., 2013). Recently, the group of Gauvrit, Delahaye, Zenil and Soler-Toscano proposed an approximation to KC using the *coding theorem*, which relates the algorithmic complexity of a sequence to the probability that a universal machine outputs that sequence (Delahaye & Zenil, 2012; Gauvrit et al., 2014, 2016; Soler-Toscano et al., 2014). They provided algorithmic complexity measures for a large set of short sequences. This proposal was presented as the best approximation of "an ultimate measure of randomness" and appeared to predict the biases observed when individuals are asked to either judge the randomness of patterns or to produce random patterns (Gauvrit et al., 2014, 2016).

As an alternative to algorithmic complexity, Aksentijevic and Gibson (2012) proposed another measure of sequence complexity, based on the notion of "change" (the inverse of invariance), which they called *change complexity*. They argued that humans attend to the structural information conveyed by the transition from one element to the next, rather than the symbols themselves. Change complexity is thus computed by quantifying the average amount of change across all sub-sequences contained in a sequence. Aksentijevic and Gibson (2012) further show that their measure has interesting properties such as a sensitivity to periodicity and symmetries, and that it performs better than previously proposed measures in predicting objective behavioral performance and subjective complexity of sequences.

As stated above, a proposal tightly related to KC is that human subjects compress sequences internally, not necessarily using a set of instructions of a Turing-complete language, but using a variety of computer-like primitives such as for-loops, while-loops, and other routines forming a specific internal "language of thought" (Fodor, 1975), strong

enough to describe any sequence, but weak enough as to permit an explicit computation of KC. Such a language would allow the combination of simple primitives into complex embedded patterns or recursive rules. Language of thought (LoT) models have been proposed very early on (see Simon, 1972). Simon & Kotovsky (1963) used concepts such as "same" , "next" (on the alphabet), and the ability to cycle through a series, to build a formal representation of the human memory for sequences of letters (e.g. "cadaeafa... "). Similarly, Restle (1970) used the operations "repeat" , "transposition" and "mirror image" . Similar languages, based on repetitions with variations, were also used to encode linear geometric figures and more elaborated 2D and 3D shapes (Leeuwenberg, 1969; Leeuwenberg, 1971). More recently, similar proposals have been used with success to study different aspects of human learning, particularly concept learning (Feldman, 2000; Goodman et al., 2008, 2011; Piantadosi et al., 2012; Piantadosi & Jacobs, 2016; Siskind, 1996). Boolean complexity, i.e. the length of the shortest logical expression that captures the concept (a notion closely related to KC) was shown to closely capture human behavior (Feldman, 2000, 2003). Going beyond the pre-specification of a specific language, the LoT approach has also be used to specify which grammar and which set of primitive operations best captures the behavior of human subjects (e.g. Piantadosi et al., 2016; Romano et al., 2018).

The proposed language for binary sequence

The development of a LoT model for sequence representation involves the selection of a set of rules or operations whose combination allows the (lossless) recoding of any given sequence. We introduce here a formal language for sequence processing as a variant of the

language of geometry previously introduced by our team to model human performance in the domain of spatial working memory (Amalric et al., 2017). In this previous study, human participants were presented with a sequence of eight locations on a regular octagon. Using both behavioral and brain-imaging data, we showed the necessity and adequacy of a computer-like language consisting of geometrical primitives of rotation and symmetry plus the ability to repeat them with various variations in starting point or symmetries (Al Roumi et al., 2020; Amalric et al., 2017; Romano et al., 2018; Wang et al., 2019). This language was shown to predict which sequences appear as regular, and how educated adults, uneducated Amazon Indians and young children performed in an explicit sequence completion task (Amalric et al., 2017) or in an implicit eye-tracking task (Wang et al., 2019). Sequence complexity, defined as minimal description length, also predicted human brain activation in a large cortical circuit including dorsolateral prefrontal cortex (Wang et al., 2019).

This language of geometry enables the generation of programs that can encode any sequence of spatial locations on an octagon. It uses primitive instructions or rules regarding the size and the direction of the next step (e.g. +1 = next element clockwise; +2 = second element clockwise), as well as the reflection over some axes (e.g. H = horizontal symmetry, picking the symmetrical location along a horizontal axis). Furthermore, these elements can be repeated, for instance +1^ 8 describes a full clockwise turn around the octagon. Finally, those repetitions can be arbitrarily embedded. For instance, the expression [[+2]^ 4]^ 2<+1> first draws a square, as determined by the subexpression [+2]^ 4), then a second one with an offset of +1 in the starting point (see Amalric et al, 2017, for a full formal description).

In the present study, we test the highly constrained hypothesis that the same language,

when reduced to only two locations, suffices to account for the human encoding of a completely different type of sequence, namely non-spatial binary sequences composed of only two arbitrary states A, B instead of the eight locations of the octagon. For such sequences, the language can be stripped of most of its primitives. We kept only the operations of staying ("+0") versus moving to the other item ("b" , i.e., the alternation instruction – or any specific symmetry in the original language), and the operation of repetition, possibly with a variation in the starting point. The language is thus able to encode any repetition of instructions in a compressed manner. The sequence "AAAA" , for instance, would be denoted [+0]ˆ 4 (i.e., same state four times), the sequence "ABAB" would be denoted [+0]ˆ 4<b> (i.e., alternations from the initial state four times). The language is recursive and can produce nested descriptions, for instance "AABAAB" can be described as "two repetitions of [two repetitions plus one change]" (see example Figure 1A). Because of recursion, even long sequences can be encoded compactly in an easy-to-remember form, for instance "ABABABBBBBBBABABABBBBBBB" is "2 times [5 alternations and 5 repetitions]" . The code is available online at https://github.com/sromano/language-of-geometry.

Given this language of thought, for each sequence, one can find the simplest expression that describes it, and its associated complexity (analogous to KC). Complexity is calculated by adding a fixed cost for each primitive instruction +0 and b. As in our previous work (Amalric et al., 2017), the additional cost for repeating an instruction n times is assumed to correspond to $\log_{10}(n)$ (rounded up), i.e. the number of digits needed to encode the number in decimal notation. The relative value of those two costs is such that even a single repetition compresses an expression: +0ˆ 2 is assumed to be more compressed than the mere concatenation of +0 +0 (see supporting information in Amalric et al., 2017,

for details). As a result, the language favors an abstract description of sequences based on the maximum amount of nested repetitions, thus sharply dissociating sequence length and complexity. Among the multiple expressions that can describe the same sequence, the expression with the lowest complexity is considered to correspond to the human mental representation of the sequence. In a nutshell, the assumption is that, in order to minimize memory load, participants mentally compress the sequence structure using the proposed formal language.

Probing memory for sequences: The sequence violation paradigm

In preparation for future experiments involving infants or non-human animals, it is useful to probe sequence processing using a paradigm that does not require language skills, nor explicit production of responses. A classic approach consists in introducing rare violations in an otherwise regular sequential input. At the most elementary level, in the *oddball* paradigm, the simple repetition of an auditory or visual stimulus with a regular timing suffices for the brain to generate expectations, such that the unexpected violation of this regularity, by suddenly replacing the stimulus by a different one, gives rise to an automatic surprise or novelty response. Such a surprise effect can be detected behaviorally, e.g. using an explicit detection, a pupillary response, or electrophysiological signatures including the mismatch negativity (Garrido et al., 2009; Näätänen, 2003; Squires et al., 1975) and it has been successfully used in non-human primates (e.g. Gil-da-Costa et al., 2013; Uhrig et al., 2014; Wilson et al., 2017).

A more complex brain response to novelty arises in the local/global paradigm (Bekinschtein et al., 2009; Wacongne et al., 2011), which contrasts two levels of violation: a local one, when a B stimulus follows a series of As (as in "AAAAB"); and a global one where, at a higher hierarchical level, the habitual sequence (e.g. "AAAAB" repeated multiple times) is replaced by a difference sequence (e.g. "AAAAA"). The use of this paradigm with neuroimaging made it for instance possible to show that macaques tend to spontaneously encode simple sequential patterns, using a cerebral network similar to the one in humans (Chao et al., 2018; Uhrig et al., 2014; Wang et al., 2015), or that such ability is already present in human infants (Basirat et al., 2014). It was also successfully used to show, with asleep participants or unconscious patients, that the processing of auditory sequential inputs at the global level (i.e., the level of patterns) is mainly restricted to conscious processing (Bekinschtein et al., 2009; Faugeras et al., 2011; Strauss et al., 2015). Behavioral and hemodynamic novelty responses to violations were also used by Huettel et al. (2002) to show that human adults spontaneously encoded simple repeating and alternating patterns, and that their response times and fMRI frontal activity patterns varied when such a local pattern was violated. Interestingly, the strength of the novelty response observed when the pattern was broken, was proportional to the length of the preceding pattern, suggesting that the novelty response may perhaps track sequence complexity.

Here, we test the hypothesis that the violation detection task can be used to probe the encoding of sequences of higher level of complexity, thus revealing their degree of psychological regularity and give an insight into the internal language of thought used to encode them. While we focus on explicit behavioral responses in a violation detection task, we do this with the aim of paving the way to future studies using non-verbal subjects or using

brain measures of implicit violation detection.

Statistical learning in sequence processing

A language of thought is by no means the only way to encode binary sequences. At a lower level of abstraction, the detection of sequential structures in the environment may involve the identification of statistical regularities in the frequencies of events or the transitions between them (Dehaene et al., 2015; Maheu et al., 2019). Even in the language domain, transition probabilities are known to play an important role. Eight-month-old infants have for instance been shown to rely on transition probabilities between syllables in order to segment a continuous stream of syllables into distinct words (Romberg & Saffran, 2010; Saffran et al., 1996). Transition probability learning, revealed by the observation of a novelty response to an improbable event, was also reported in the visual modality (Abla & Okanoya, 2009; Kirkham et al., 2002), as well as in non-human primates (Hauser et al., 2001; Meyer & Olson, 2011). This process appears to be automatic and continues to operate under non-conscious conditions (Bekinschtein et al., 2009; Faugeras et al., 2011; Strauss et al., 2015). When using the novelty effect as an indicator of sequence complexity, it is therefore essential to separate the respective contributions of statistical learning and of a putative language of thought.

Computational models relying on probabilistic inference have been proposed for statistical learning. Mars et al., (2008) for instance showed that the trial-by-trial modulation of the amplitude of the P300 (an event-related potential response associated with unexpected events) could be explained by a model tracking the frequency of occurrence of items (among 4) in a temporal sequence. Similarly, our team proposed a Bayesian model for

the acquisition of transition probabilities (not simply item frequency), and showed that it could explain a great variety of different behavioral and brain observations in binary sequence processing experiments (Maheu et al., 2019; Meyniel et al., 2016). The degree of confidence in a prediction can furthermore be predicted using such computational approach (Meyniel & Dehaene, 2017). In these models, Shannon surprise, a mathematical measure of the improbability of an event considering past events (the negative log probability of events) (Friston, 2010; Shannon, 1948; Strange et al., 2005), is considered a good predictor of behavioral and neural responses.

In summary, prior research indicates that, at a minimum, two distinct systems may underlie sequence learning in the human brain: statistical versus rule-based learning (Bekinschtein et al., 2009; Dehaene et al., 2015; Maheu et al., 2020). What is unknown is whether they operate independently and whether one is privileged at the expense of the other depending on the nature of the information to be encoded. We argue that any attempt to uncover the specific cognitive mechanisms behind rule learning in humans, especially in comparison with other species, must take into account the contribution of the less abstract yet powerful prediction system based on the statistical properties of events.

The current study

Our hypothesis is that, when confronted with a sequential input, individuals tend to spontaneously recode the sequence in an abstract form, using an internal "language of thought" composed of a limited set of simple rules that can be hierarchically embedded. To test this hypothesis, we conducted a series of behavioral experiments in which participants were asked to listen to short auditory binary sequences (alternations of a sound "A" and

a sound "B"), whose statistical properties and predicted complexity varied. Learning was assessed by examining the capacity of participants to detect rare violations of the learned sequence (i.e. when one tone was replaced by the other). Our hypothesis was that, for equal sequence length, error rate and response time in violation detection would increase in parallel with sequence complexity. In some experiments, in addition to those objective indices of complexity, we also asked participants to report subjective rating of complexity. Finally, in one experiment, we also compared auditory and visual sequences to test whether our findings were dependent on the sensory modality.

For analysis, we examined if the results correlated with the shortest description length in the proposed language of thought (hereafter called LoT complexity to distinguish it from other complexity measures). To separate rule-based and statistical learning, we compared LoT complexity and surprise as predictors of performance. We also compare LoT complexity with other computational approaches to sequence complexity. We started with long sequences of 16 items (experiment 1), and then probed the adequacy of the proposed language to shorter sequences (experiments 2-5). A simple prediction is that shorter sequences are more likely to be stored in a verbatim representation in working memory, without any internal compression. Thus, we predicted that the effect of LoT complexity in the proposed language of thought would decrease as the sequence gets shorter. On the other hand, given the automaticity of statistical learning, we did not expect any difference in its contribution to long versus short sequences.

Experiment 1: auditory sequences with 16 items

In experiment 1, we selected 10 auditory sequences of 16 items, a number that vastly exceeds working memory capacity, which is typically between 4 to 9 items (Cowan, 2001, 2010; Miller, 1956). All sequences had equal numbers of sounds A and B (to reduce confounds related to the relative probability of As and Bs), yet they varied widely in LoT complexity (see Figure 2). We obtained a subjective measure of complexity as well as objective measures of complexity based on the response to occasional violations. Two types of violations were introduced: sequence deviants in which an A was replaced by a B or vice-versa; and "super-deviants" , in which an A or B was replaced by a rare novel tone C (see Figure 1B). We predicted that the detection of sequence deviants would be affected by sequence complexity, since responding to them required the detection of a discrepancy between the observed and the predicted stimuli, and such a prediction would be more difficult for more complex sequences. Super-deviants were not expected to yield a complexity effect, however, since they deviated from other stimuli at the most basic stimulus-frequency level. Super-deviants stimuli were introduced in an effort to ensure an invariant task which would equalize level of attention in all blocks, regardless of sequence complexity.

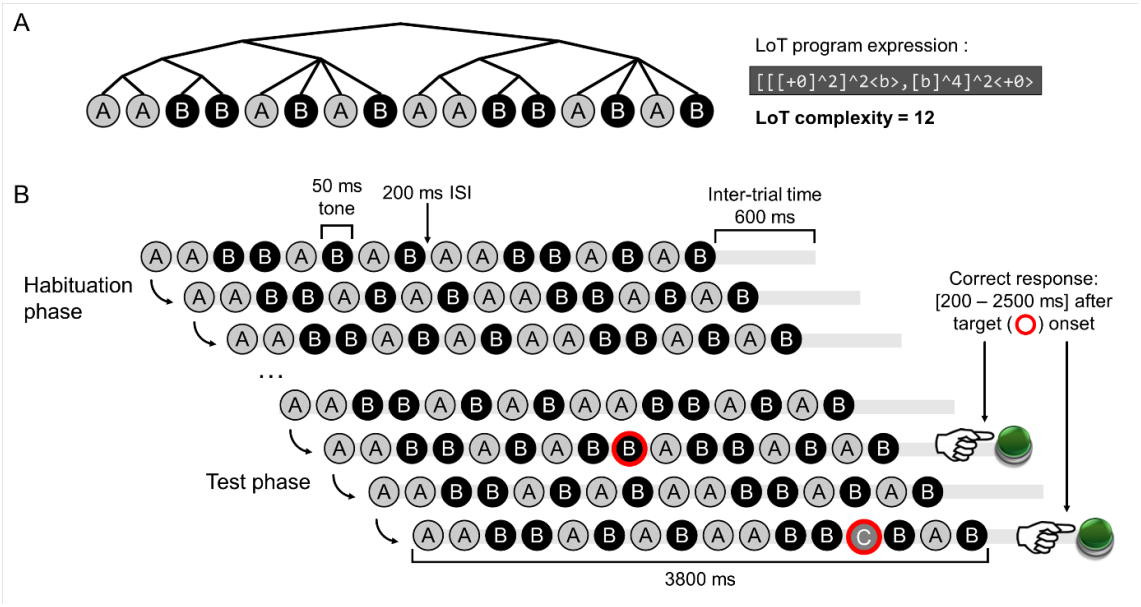

Figure 1: (A) Example of a 16-items long sequential pattern, with its shortest representation in the language of thought (i.e. LoT program expression) and the tree-structure derived from this expression (illustrating the hierarchical representation). The LoT complexity of this sequence is also indicated. (B) Experimental design of the violation detection task: a session with the sequence "AABBABABAABBABAB" is represented, with one example of a target sequence deviant item ("A" replaced by "B" , at position 9) and one example of a target super-deviant item ("C" at position 13) (deviants could occur at positions 9, 11, 13 or 15).

Method

Participants

Twenty-eight healthy volunteers ($M_{\text{age}} = 24.3$, $SD = 3.2$, 16 women) participated in the current experiment. They all gave written consent to participate and were paid for their participation. All participants performed the subjective complexity rating task. Due to time constraints, 7 of them performed only 6 out of the 10 independent short sessions of deviance detection.

Stimuli

Auditory binary sequences were composed of an alternation of two different tones; low pitch and high pitch. Each stimulus was a complex tone synthesized with the superimposition of four sine waves. Sound frequencies were chosen to correspond to musical notes: 494, 740, 988 and 1480Hz (i.e., B, F# , B, F#) for the lower pitch tone, and 622, 932, 1245 and 1865Hz (i.e., D# , Bb, D# , Bb) for the higher pitch tone. The two complex tones were randomly assigned to items A and B (i.e., the two elements composing the binary sequential patterns) for each experimental session. Thus, stimulus attribution changed from one sequence to the next and from one participant to the next but was kept constant for a given sequence in a given participant. In addition, one lower pitch tone (415, 622, 831 and 1245Hz) and one higher pitch tone (740, 1109, 1480 and 2217Hz) were synthesized, to be used as easy-to-detect super-deviant (or C) stimuli. All tones were 50 ms long, with 5 ms initial and final ramp. Inter-stimulus interval (ISI) was 200 ms. Total sequence duration was 3800 ms.

Ten sequences were chosen (see Figure 2), which were all composed of the same number of items (8 As, 8 Bs). The first four sequences, of lowest complexity, followed the simple algebraic pattern $(A^nB^n)^x$: $(AB)^8$, $(A^2B^2)^4$, $(A^4B^4)^2$ and A^8B^8. The period of these sequences differed (2, 4, 8 and 16 tones), but the complexity was identical (LoT complexity = 6). The other 6 sequences had LoT complexity values ranging from 12 to 23. Half of them were periodic (period of 8).

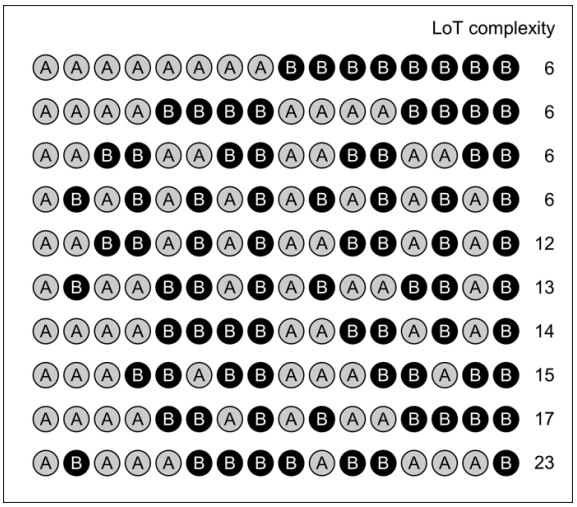

Figure 2: Ten 16-items long sequential patterns used in Experiment 1, with their corresponding LoT complexity value.

Procedure

Participants were seated in front of a computer in a quiet room and were wearing headphones. Stimuli were delivered using the Psychophysics Toolbox 3 (Brainard & Vision, 1997; Kleiner et al., 2007) running on Matlab R2016a (Mathworks Inc., Natick, MA, USA). Before starting the experiment, participants listened to a sample of the stimuli (different sequences from the ones used in the main experiment) and the sound volume was adjusted if necessary.

In the first part of the experiment, participants performed the complexity rating task. They were asked to judge each sequence on a scale going from "1: very simple" to "9: very complex" , by pressing the corresponding key on the keyboard just after sequence presentation. A response was requested at each trial. Each of the ten sequences was presented three times, in a pseudo-random order (30 trials). The low-pitch and high-pitch tone were randomly assigned to either A and B or to B and A at each presentation.

In the second part, the violation detection task, each of the ten sequences was tested

in a different short session of approximately 4 min (Figure 1B). Order of sessions was randomized for each participant. Each session comprised three blocks separated by pauses and in which the sequence (3800 ms long) was repeatedly presented with a 600 ms inter-trial duration. In the first block, the habituation block, the unaltered sequence was presented eight times. Participants were asked to listen to the stimuli and try to remember the sequence. In the two following blocks, the testing blocks, participants were asked to respond whenever they detected that the sequence had been altered (one deviant tone), by pressing the space key of the keyboard as quickly as possible (without necessarily waiting until the end of sequence presentation). Each of the two test blocks comprised 18 sequences, 9 of them containing one deviant tone (among the sixteen tones composing the sequence). Two-thirds of the deviant sequences were produced by replacing a tone A by a tone B, or conversely ("sequence deviant" tones, 12 trials per session). The remaining third were obtained by replacing one tone by a low or high-pitch C sound ("super-deviant" tone, 6 trials per session). Deviant tones could occur at only four, equally probable, positions within the second half of the sequence (positions 9, 11, 13 or 15).

Data analysis

The responses collected in the complexity rating task, ranging from 1 to 9, were normalized for each participant using a z-score transformation of the raw ratings within each participant. An average complexity rating was computed for each sequence and subject and entered into a mixed effect model with participant as random factor and LoT complexity value as a fixed effect predictor. Here and in following mixed effect analyses, similar results were obtained using classical repeated-measures ANOVAs with participants as the random factor.

For the violation detection task, a button press occurring between 200 and 2500 ms after deviant stimulus onset was considered a hit (i.e. correct response). An absence of response during this interval was counted as a miss. False alarms were collected and analyzed separately (using a simple linear regression analysis with the LoT complexity predictor). Note that participants were not aware of the number or frequency of targets, and could respond at any time. Thus, only the number of false alarms, rather than a ratio depending on the number of trials, was relevant. The Linear Integrated Speed-Accuracy Score (LISAS) (Vandierendonck, 2017, 2018), an integrated measure of response times and error rates, was used as the main indicator of performance (results with response times and miss rates were quite convergent and are provided in supplementary materials). This score was computed for each sequence, each deviant type in each subject, according to the following formula: $= RT_c + MR \times \frac{S_{RT}}{S_{MR}}$, where RT_c refers to the average response time (of correct responses), MR to the miss rate, S_{RT} to participant's overall RT standard deviation and S_{MR} to the participant's overall MR standard deviation. These scores were computed after removing extreme response times (2.5 standard deviations (SD) above or below the median in each condition and subject, 2.0% of data). Participants with excessive average miss rate over the entire session (i.e. 2.5 SD above group median), average response time and/or average number of false alarms were excluded (three participants). All data analyzes were performed in R 3.6.0 (R Core Team, 2017).

We performed statistical analyses using a mixed model in which the dependent variable was the LISAS within each participant and each cell of the design, participants were the random factor, and LoT complexity and deviant type (sequence deviants vs. super-deviant) were fixed factors. To clarify the interactions, we also computed the same mixed effect model after restricting the data to each deviant type. All computations were per-

formed using the lme4 (Bates et al., 2015) and lmerTest (Kuznetsova et al., 2017) packages. P-values for each factor were obtained using Kenward-Roger approximation for degrees of freedom (Kenward & Roger, 1997).

Since statistical trends were also expected to play a role in how participants react to deviant stimuli, another predictor, distinct from LoT complexity, was constructed. We used Shannon surprise, defined as the negative log-probability of the likelihood of an observation (Friston, 2010; Meyniel et al., 2016; Meyniel & Dehaene, 2017; Shannon, 1948; Strange et al., 2005), to characterize how unexpected a deviant stimulus would be for an observer that tracks transition probabilities of the original sequence; for binary sequences: $p(\mathrm{A}|\ \mathrm{A}) = 1 - p(\mathrm{B}|\ \mathrm{A})$ and $p(\mathrm{A}|\ \mathrm{B}) = 1 - p(\mathrm{B}|\ \mathrm{B})$. Since the sequence was considered to be learnt after the habituation phase, fixed probabilities were used (rather than evolving on a trial-by-trial basis based on a recent time window, as used for instance by Maheu et al., 2019, 2020; Meyniel et al., 2016). For instance, in the $\mathrm{A}^8\mathrm{B}^8$ sequence, $p(\mathrm{A}|\ \mathrm{A})$ has a probability of 0.875 in the original sequence. Thus, the corresponding surprise of getting an A instead of a B at the 9^{th} position is low ($-\log_2(0.875) \approx 0.18 bit$. In the same sequence, $p(\mathrm{A}|\ \mathrm{B}) = 0$ (since B is always followed by another B), and therefore the surprise of getting an A instead of a B at, say, the 11^{th} position, is maximal. To avoid an infinite when computing surprise, probabilities of 0 were replaced by a small but non-zero probability of $p = 0.01$, capping the maximum surprise value at around 6.64 bits. To test whether this would affect our conclusions, complementary analyses were also conducted while excluding deviants with such zero probability. Note that, contrary to the LoT complexity, which is identical whatever the position of the deviant, surprise varies with deviant location in the sequence (up to four different values in one given block). Analyses comparing the surprise and LoT complexity predictors were performed using the

same mixed model as above, including participants as random effects. To compare pair of models, we used likelihood ratio combined with chi-square statistical tests. When more than 2 models were involved, we computed the Akaike information criterion for each model (Akaike, 1998). Note that both methods penalize for model complexity (i.e. the number of predictors included in the regression), which varies depending on whether, or not, LoT complexity was included in addition to Shannon surprise (see above). Super-deviant trials were not included in these analyses.

In addition to those mixed effect statistics, we also report the results of simple regression analyses, which provide a summary view of the Pearson correlation coefficient r between LoT complexity and either subjective complexity ratings or the LISAS for each sequence, after averaging across participants (this is the r value reported in the figures). Supplementary figures provide this statistic for RTs and miss rates.

Results and discussion

Complexity rating task

We observed a strong positive linear relationship between the average subjective complexity ratings and the LoT complexity ($t(278) = 24.6$, $p < .0001$; Pearson correlation coefficient on the average ratings for each sequence, $r = .94$) (see Figure 3A). These results indicate that participants were readily able to judge whether a pattern is "more complex" than another, and that the formal language we used to compute sequence complexity is close to how individuals form such complexity judgements.

Deviant type and complexity effects in the violation detection task

We observed a linear relationship of LoT complexity and performance in the violation detection task (using LISAS). We observed main effects of LoT complexity ($t(415.0) = 18.1$, $p < .0001$), deviant type (994 ms for sequence deviants vs. 570 ms for super-deviants; $t(414.4) = 18.9$, $p < .0001$) and their interaction ($t(414.5) = 11.7$, $p < .0001$). Indeed, the slope of the complexity effect was significantly stronger, by an order of magnitude, for sequence deviants as opposed to super-deviants (respectively +51 ms vs. +5 ms in simple regression, $t(16) = 11.7$, $p < .0001$; see Figure 3B, and Figure S1 for the corresponding results using response times or miss rate instead of LISAS). Nevertheless, separate analyses revealed that LoT complexity was a strong predictor of performance for sequence deviants ($t(193.0) = 15.5$, $p < .0001$; $r = .98$) and also, surprisingly, for super-deviants ($t(198.5) = 4.08$, $p < .0001$; $r = .72$) (Figure 3B). The latter effect on LISAS was however mainly driven by response times, since the average hit-rate for super-deviants was high (96%) and weakly modulated by LoT complexity ($t(200.7) = 2.32$, $p < .03$).

The number of false alarms per sequence (which was 1.99 on average) also increased with sequence LoT complexity ($t(214.4)$=4.20, $p < .0001$; $r = .74$), suggesting here again that the LoT complexity was a good predictor of the quality of sequence encoding.

The results of this first experiment with long binary auditory sequences (16 items) thus indicate that the formal language used to describe sequences in a compressed form, based on simple (possibly embedded) rules, is highly relevant to predict (1) how "complex" an auditory sequence is judged by adult participants after having listened to it once and (2) how difficult it was to learn these sequences in order to detect alterations.

Sequence complexity was expected to have little or no impact on the detection of super-

deviants, i.e. high-pitch or low-pitch tones different from the two tones composing the binary auditory sequence. Our rationale was that such "C" tones were detectable even without any prior knowledge of sequence structure. While performance in detecting super-deviants was much better than for sequence deviants, even for the simplest sequences, a clear relationship between LoT complexity and performance continued to be observed. We see at least two interpretations of this finding. First, there could be an increased attentional cost of having to detect violations in more complex sequences, thus placing subjects in a dual-task setting of having to simultaneously maintain a complex representation in memory and to respond to deviants. Alternatively, the effect could reflect the influence of a top-down prediction system which would use sequence structure to generate predictions of the incoming stimuli. Complex sequences would be less well predicted, and this would in turn affect the speed with which any deviant is detected. We return to this question in the *General Discussion*.

Surprise effects

Many prior experiments, using either or both behavior and brain-imaging measures, have shown that individuals constantly entertain predictions about future observations using probabilistic knowledge based on past observations (e.g. Maheu et al., 2019; Meyniel et al., 2016). In order to test whether task performance could be explained by transition probabilities (surprise) or also implied an encoding of sequence structure, a mixed model (with participants as a random effect) including fixed effects of both LoT complexity and surprise (averaged across the 4 possible positions of deviants in a given sequence) was compared to a null model including only the latter. The effect of surprise in the null model with surprise alone was significant ($t(193.0) = 5.31$, $p < .0001$). However, a likelihood

ratio test showed that adding LoT complexity significantly improved the goodness of fit: χ (1) = 130.9, p < .0001. Adding a "period" factor (i.e., period values were 2, 4, 8 or 16) did not improve the model fit ($\chi^2(1) = 1.23$, p = .27), confirming the prediction that the four included A^nB^n patterns have the same psychological complexity, and suggesting that this information is already captured by LoT complexity. Adding the interaction between surprise and LoT complexity did not improve goodness of fit either ($\chi^2(1) = 2.50$, p = .11). As reported in Table 1, the LoT complexity fixed effect was significant in the final full model ($t(192.4) = 13.6$, p < .0001), but not the surprise fixed effect ($t(191.8) = 0.60$, p = .55). The absence of a significant effect of surprise once sequence complexity is taken into account reflects the existence of a small correlation between the two measures (r = -.54), biased transition probabilities in less complex sequences tending to make deviants more easily surprising. It also shows that when these two slightly colinear factors are included, LoT is more effective than surprise describing the variance of the data.

As our choice of attributing an arbitrary padding value (0.01) to deviant transitions events with zero probability when computing surprise may have biased the results, we recomputed the LISAS and average surprise while excluding all such trials (i.e. all deviant positions in the $(AB)^8$ pattern, 3 out of 4 deviant positions in the A^8B^8 pattern). Here again, a likelihood ratio test showed that the goodness of fit increased significantly when adding LoT complexity to a null model containing only surprise (χ (1) = 116.3, p < .0001). However, both complexity ($t(165.5) = 12.9$, p < .0001) and surprise ($t(165.8) = 3.82$, p < .0001) were significant with this subset of the data.

In conclusion, the strong complexity effects observed here indicated that participants used some form of compression of information to encode the sequence and perform the task over and above statistical information. Although no instruction was given in that sense,

this strategy may be needed in order to deal with a difficult, memory-demanding task. Indeed, at the maximum level of complexity used, performance in violation detection was very low (the violation detection rate dropped to 41% for sequence deviants). In the subsequent experiments, we asked whether similar complexity effects emerged using the same paradigm with shorter sequences; when the sequence can be more easily encoded and stored "as a whole" , without necessarily requiring a re-encoding in a more abstract, compressed form. In these less demanding conditions, it can be expected that the spontaneous encoding of transitions probabilities between items will play a more important role in the detection of violations.

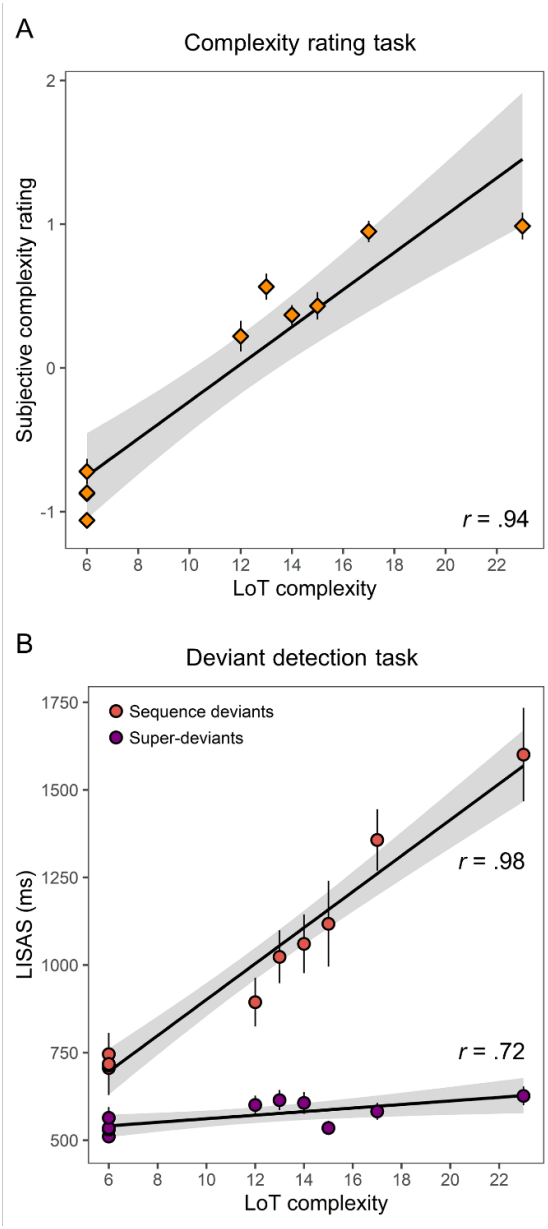

Figure 1: : Linear relationship between LoT complexity and subjective and objective measures obtained in experiment 1 with ten 16-items long auditory sequences (with 95% confidence intervals bands in gray). The Pearson correlation (r) coefficient is indicated. Each marker represents the group-average for a given sequence. Error bars represent SEM across participants. (A) LoT complexity vs. subjective complexity ratings. (B) LoT complexity vs. performance in the violation detection task (Linear Integrated Speed-Accuracy Score), for sequence deviants and super-deviants.

Experiment 1 (16-items sequences, excluding super-deviants)

continued on next page

continued from previous page

Predictors	*Estimates*	*Std. Error*	*T-value*	*95% CI*	*p*
(Intercept)	356.90	80.51	4.43	199.5 – 514.3	< **.0001**
Complexity	52.15	3.84	13.60	44.6 – 59.7	< **.0001**
Surprise	6.77	11.31	0.60	-15.4 – 28.9	.55

Experiment 2 (12-items sequences, excluding super-deviants)

Predictors	*Estimates*	*Std. Error*	*T-value*	*95% CI*	*p*
(Intercept)	852.38	124.91	6.82	608.5 – 1096.2	< **.0001**
Complexity	24.21	6.29	3.85	11.9 – 36.5	< **.0002**
Surprise	-43.13	21.06	-2.05	-84.4 – -1.9	< **.05**

Experiment 3 (8-items sequences)

Predictors	*Estimates*	*Std. Error*	*T-value*	*95% CI*	*p*
(Intercept)	852.40	73.39	11.62	707.8 – 997	< **.0001**
Complexity	10.75	3.49	3.08	3.9 – 17.6	< **.003**
Surprise	-32.37	5.60	-5.78	-43.3 – -21.4	< **.0001**

Experiment 4 (6-items sequences, sequence 'AAAAAA' excluded)

Predictors	*Estimates*	*Std. Error*	*T-value*	*95% CI*	*p*
(Intercept)	751.6	47.5	15.8	658.8 – 844.5	< **.0001**
Complexity	1.4	4.4	0.3	-7.2 – 9.9	.75

continued on next page

continued from previous page

Surprise	-15.3	3.8	-4.1	-22.7 – -7.9	< **.0001**

Experiment 5 (8-items sequences, auditory and visual)

Predictors	*Estimates*	*Std. Error*	*T-value*	*95% CI*	*p*
(Intercept)	645.1	92.2	7.0	464.4 – 825.9	< **.0001**
Complexity	25.2	25.2	4.4	14 – 36.4	< **.0001**
Surprise	-36.7	8.1	-4.5	-52.5 – -20.8	< **.0001**
Modality (Visual)	337.0	337.0	14.2	290.7 – 383.3	< **.0001**

Experiment 2: auditory sequences with 12 items

Methods

Participants

Twenty healthy volunteers ($M_{\text{age}} = 26.5$, $SD = 9.5$, 15 women) participated in experiment 2. They all gave written consent to participate and were paid for their participation.

Stimuli

Auditory binary sequences of twelve sounds were used for this experiment. They were composed of an alternation of the same two complex tones as in the previous experiment, with the same duration and SOA. Total sequence duration was 2800 ms. Twelve different

sequential patterns, each composed of 6 As and 6 Bs were presented to each participant (Figure 4).

Procedure

The same procedure and material as in the previous experiment was used. The complexity rating task was performed first (each of the twelve sequences was presented three times, in a pseudo-random order) followed by the violation detection task. In the latter, each sequence was tested in a different short session of approximately 3 min (habituation block of 8 trials, two test blocks of 18 trials each), followed by a pause. Each sequence lasted 2800 ms and was followed by a 1000 ms intertrial blank. Order of blocks was randomized for each participant. Half of the trials in tests block contained one deviant tone (at positions 7, 8, 9, 10, 11 or 12): 2/3 of "sequence deviants" , 1/3 of "super-deviants" . Participants were asked to press the button, as quickly as possible, as soon as they detected that the sequence has been altered.

Data analysis

The same analysis as in the previous experiment were conducted. Extreme response times were removed (using the same procedure as in experiment 1), and represented 1.2% of all RTs. One participant was excluded (average number of false alarms per sequence more than 2.5 *SD* above the group median).

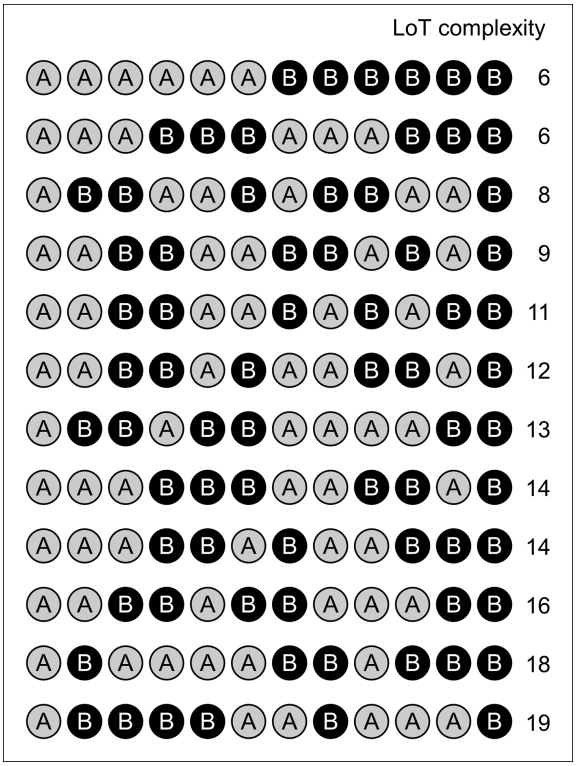

Figure 4: Twelve 12-item sequences used in experiment 2, with their corresponding LoT complexity value (in bits).

Results and discussion

Complexity rating task

A positive linear relationship was found between subjective complexity ratings and LoT complexity ($t(238) = 6.81$ p $< .0001$, $r = .61$). The correlation of the average score per sequence with LoT complexity was however less strong than what was observed in the previous experiment with 16-items long sequences ($r = .61$, see Figure 5A). Subjective complexity was clearly underestimated for one specific sequence ("ABBAABABBAAB" , predicted complexity of 8), which is confirmed by an inspection of the residuals of the regression (residual 1.99 *SD* above average for this sequence).

Deviant type and complexity effects in the violation detection task

Regarding the violation detection task, main effects of LoT complexity ($t(431.1) = 6.43$, $p < .0001$) and deviant type (1078 ms for sequence deviants vs. 545 ms for super-deviants; $t(431.0) = 19.3$, $p < .0001$) were observed, as well as their interaction ($t(431.1) = 3.48$, $p < .0006$). The slope of the complexity effect appeared indeed stronger for sequence deviants as opposed to super-deviants (respectively +30 ms, vs. +7 ms; see Figure 5B). Separated analyses revealed that it was significant in analyses including either sequence deviants only ($t(205.1) = 5.78$, $p < .0001$; $r = .63$), or super-deviants only ($t(208.0) = 2.88$, $p < .005$; $r = .59$). The number of false alarms per sequence (3.88 on average) was also predicted by the LoT complexity of the sequence ($t(208.0) = 3.50$, $p < .0006$; $r = .56$).

As in the complexity rating task, although the overall correlation was high, a noticeable deviation between predicted complexity and observed performance was present for some of the sequences. In fact, the correlation profiles observed in the Figure 5A and 5B suggest that the psychological complexity of the pattern, as indexed by subjective rating or violation detection task performance, might have been, for some sequences, consistently overestimated or underestimated by the LoT across both tasks (the largest residual in the regression with the sequence deviants, 1.50 *SD* above average, corresponded to the same sequence identified by complexity ratings : "ABBAABABBAAB"). To further test this idea, we computed the correlation between the residuals of both linear regressions. The correlation was significant ($t(10) = 4.02$; $p < .003$), indicating that even after regressing out the effect of LoT complexity, the data from both experiments remained correlated with each other, and thus that, although the proposed LoT is a good predictor, it does not fully account for all details of the psychological complexity of patterns. One attempt

to address such potential limitations of the language is reported in the *Further analysis* section.

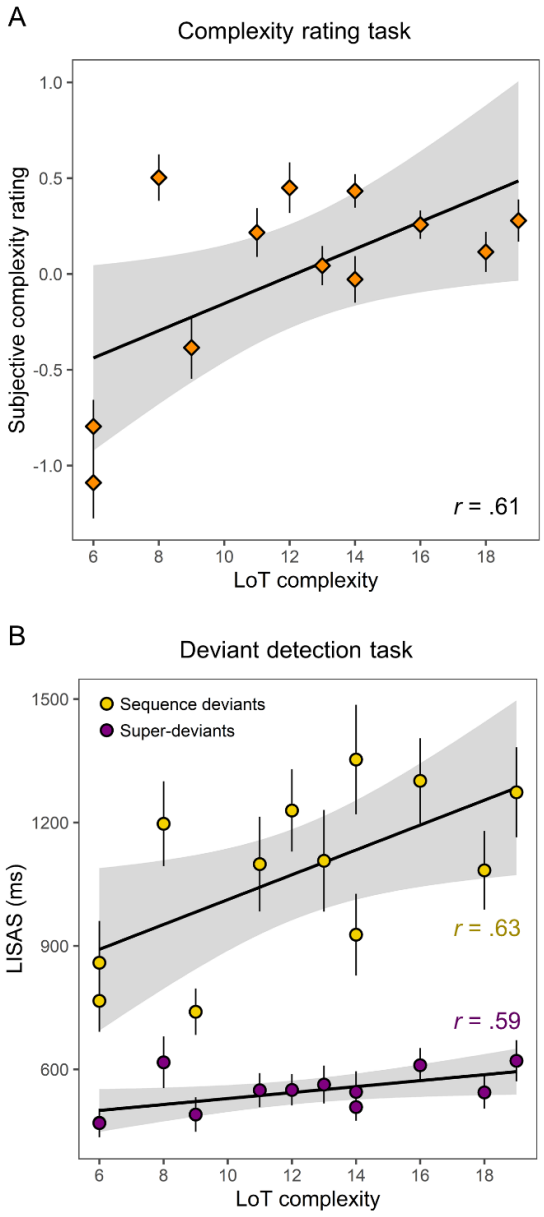

Figure 2: : Linear relationship between LoT complexity and scores obtained in the two tasks of experiment 2 with 12-item auditory sequences (with 95% confidence intervals bands in gray). Same format as Figure 3.

Surprise effects

A comparison of mixed models (with participants as a random effect) showed that, compared to a null model only including the predicting power of surprise (null model; in which the main predictor was significant: $t(205.0) = 4.67$, $p < .0001$), a model also including LoT complexity (full model) fitted the data better (likelihood ratio test : $\chi\ (1) = 14.4$, $p < .0002$). Both fixed effects were significant in the full model: LoT complexity ($t(204.1) = 3.85$, $p < .0001$), as well as surprise ($t(204.0) = 2.05$, $p < .05$) (see Table 1). Although we can conclude that the effect of statistical learning (indexed by the level of surprise of deviant items) is here stronger than in the previous experiment (in which it was not clearly significant), note that the effect of surprise remained low.

Experiment 3 and 4: auditory sequences with 6 or 8 items

The main objective of experiments 3 (with 8-items long sequences) and 4 (with 6-items long sequences) was to test whether the effect of complexity observed in the first two experiments could be generalized to larger sets of shorter sequences, where we could examine more gradual variations in complexity. The same violation detection paradigm was used. No subjective complexity ratings were collected (given the increased number of included sequences compared to the previous experiments).

Methods

Participants

Thirty-two healthy volunteers ($M_{\text{age}} = 27.4$, $SD = 5.3$, 21 women) participated in experiment 3 and twenty-three in experiment 4 ($M_{\text{age}} = 23.4$, $SD = 4.5$, 18 women). They all gave written consent to participate and were paid for their participation.

Stimuli

In experiment 3, auditory binary sequences of eight sounds were used. They were composed of an alternation of the same two complex tones as in the previous experiment, with the same duration and SOA. Total sequence duration was 1800 ms. Thirty-five different sequential patterns were presented to each participant, i.e. all possible 8-element-long binary combinations that contained the same number of As and Bs (as in experiment 1 and 2).

In experiment 4, auditory binary sequences of six sounds were used (1300 ms). Thirty-two different sequential patterns were presented to each participant, representing all 2^5 types of 6-element sequences (given that the labelling of As and Bs is arbitrary, sequences such ABABAB and BABABA were considered identical). Note that, in this case, the proportion of As vs. Bs varied across sequences.

Procedure

The same procedure and material as in previous experiments were used. Each sequence was however here tested in a single block of 35 trials (auditory sequence of 1800 or 1300

ms and inter-trial duration of 1000 ms). Alterations of the sequence occur on 1/3 of the trials, starting from the 9th trial (i.e. the habituation phase comprised 8 repetitions). Deviant tones (sounds A replaced by B or conversely — there were no super-deviants in these experiments) were positioned in the second half of the sequence (four or three equiprobable positions). As before, participants were asked to press the button, as quickly as possible, as soon as they detected that the sequence has been altered.

Data analysis

The same analyzes as in the previous experiments were conducted. Extreme response times that were removed represented 1.6% of RTs in experiment 3 and 1.6% in experiment 4. One participant was excluded in experiment 3 (average number of false alarms per sequence more than 2.5 *SD* above the group median), and one in experiment 4 (average miss rate more than 2.5 *SD* above the group median).

Results and discussion

Here again, we tested (using mixed models) whether surprise suffices to explain the variance in performance or if a significant proportion remained yet to be explained by sequence complexity (all models included participants as a random effect). In experiment 3 (8-items sequences, N = 35), goodness of fit improved when LoT complexity was included in the model (χ (1) = 9.47, p < .003). Both fixed effects were significant in the full model: LoT complexity ($t(1042.0) = 3.08$, p < .003; see Figure 6A), as well as surprise ($t(1042.0) = 5.78$, p < .0001) (see Table 1). Note that the surprise fixed effect was already highly significant in the null model ($t(1043.0) = 8.72$, p < .0001).

Similarly, in experiment 4 (6-items sequences, N = 32), goodness of fit improved when LoT complexity was included to the surprise-only null model (χ (1) = 6.20, p < .02) with both fixed effects significant in the full model (LoT complexity: $t(649.00) = 2.49$, p < .02; see Figure 6B), and surprise ($t(649.0) = 5.48$, p < .0001). The surprise fixed effect was here again already highly significant in the null model ($t(650.0) = 6.78$, p < .0001). However, one sequence appeared as an outlier in this experiment, with an average LISAS 3.9 *SD* below the average of all sequences (i.e. indicating a much better performance): the "AAAAAA" sequence. In this case, performing the task requires no sequence learning, but merely remembering the identity of the A sound, and violation detection is therefore similar to a classic oddball paradigm. When this sequence was removed from the dataset (it was also excluded from further analyses), the inclusion of the complexity fixed factor did no longer improved model goodness of fit (χ (1) = 0.10, p = .75). Indeed, the LoT complexity fixed effect was not significant in the full model ($t(628.0) = 0.32$, p = .75), as opposed to the surprise fixed effect ($t(628.0) = 4.07$, p < .0001) (see Table 1). No improvement in model fit was found when including the interaction between complexity and surprise (χ (1) = 0.08 in experiment 3, χ (1) = 0.34 in experiment 4).

Beside the effect of complexity, the strong effect of surprise in both experiments indicates that participants were quicker and more likely to detect a deviant when it violated statistical regularities characterizing the auditory sequence being repeatedly played. This is consistent with the idea that humans spontaneously encode the probabilities associated with events and react to surprising events depending on their level of predictability (Huettel et al., 2002; Meyniel et al., 2016).

The number of false alarms was low in the present experiments (0.91 per sequence on average in experiment 3, 0.60 in experiment 4). It was slightly related to sequence com-

plexity in experiment 3 $t(1048) = 2.19$, $p < .03$) but not in experiment 4 $t(650.0) = 0.29$, $p = .77$).

Compared to the previous experiment with lengths 12 and 16, it was expected here, with sequences of 8 or 6 items, that the effect of LoT complexity would be mitigated, since those auditory sequences may become short enough to be stored in working memory as a simple chain (note that the range of LoT complexity values was also smaller). The correlation of performance with LoT complexity was in fact still present with 8-items sequences (at a similar level as in experiment 2) but disappeared with 6-items sequences. This is in line with the assumption that complexity is tightly linked with the idea of compressibility in memory, and suggests that such a compression strategy, whether it is simple chunking or involves a hierarchical representation, is more likely to be involved when the number of items to store in working memory exceeds the typical working memory span (MacGregor, 1987; Mathy & Feldman, 2012). However, rather than a clear threshold above which complexity would become predictive of performance, the estimates of the LoT complexity effect across the four experiments (in the mixed models taking into account surprise) reveal a gradient: with stronger effects of complexity for longer sequences (respectively +1.4 ms, +10.8 ms, +24.2 ms, and +52.2 ms, for the experiments with length 6, 8, 12 and 16 respectively; see Table 1). The effect of surprise seemed to follow an inverse trend, with insignificant or marginal effects in long sequences (experiments 1 and 2) and highly significant effects in short sequences (experiments 3 and 4). To test this idea, the data from experiments 1-4 (excluding super-deviants) were combined in a single mixed model including the three fixed factors of LoT complexity, surprise and length (as a continuous predictor), as well as the three two-way interactions (with participants as the random factor). An ANOVA on the mixed model revealed main effects of LoT complexity ($F(1,$

2336.4) = 48.0, p < .0001) and surprise ($F(1, 2334.1) = 4.91$, p < .03). The main effect of sequence length was marginally significant ($F(1, 96.6) = 3.08$, p = .082). As expected, a strong interaction between LoT complexity and length was present ($F(1, 2347.5) = 63.3$, p < .0001), indicating a stronger effect of complexity when sequence length increased. The estimated slopes for the LoT complexity effect indeed increased with each sequence length (+15.5 ms, +46.0 ms, +107.1 ms, and +168.1 ms, for length 6, 8, 12 and 16, respectively). The interaction between length and surprise was not significant ($F(1, 2330.0) = 1.19$, p = .28). However, the estimated slopes for the surprise effect followed our initial observation: they decreased with each sequence length (-15.6 ms, -12.0 ms, -4.9 ms, and +2.2 ms).

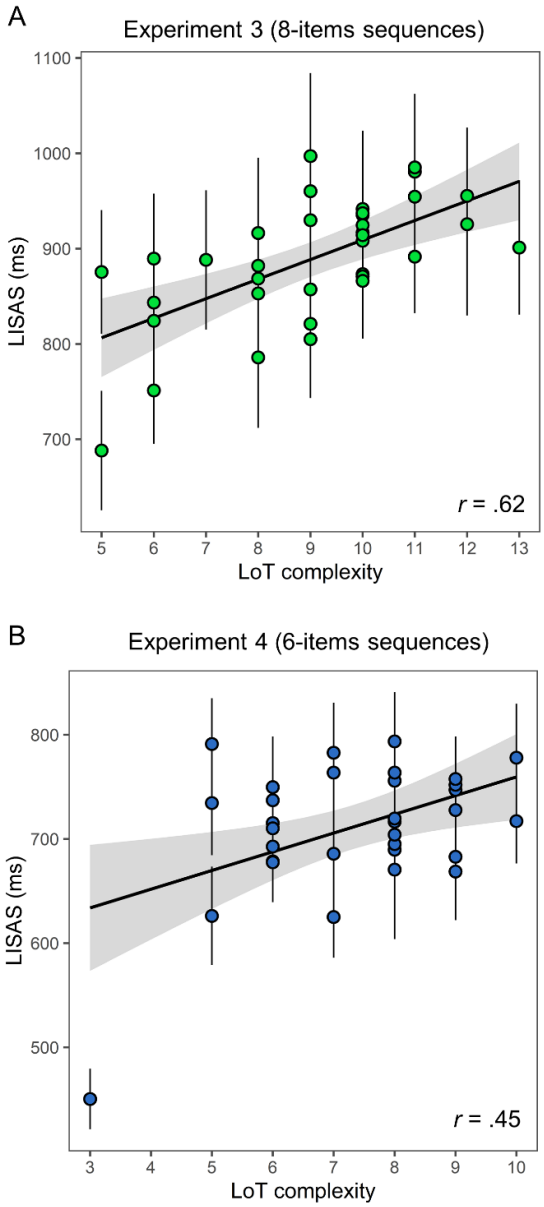

Figure 3: : Linear relationship between LoT complexity and violation detection task performance (LISAS) in: (A) experiment 3 (8-items sequences) and (B) experiment 4 (6-items sequences).

Experiment 5: auditory and visual sequences

The observation of a LoT complexity effect on sequences of length 8 and higher is consistent with our initial claim that individuals spontaneously apply simple rules (mainly based on nested repetitions) in order to recode auditory sequences in a compressed abstract form in memory. It may be argued, however, that rather than being abstract and universal, some of these effects may reflect the great ability of our auditory system to manipulate and find regularities in acoustic stimuli; whether it is in spoken language or in music listening. In experiment 5, we wished to replicate the findings of previous experiment and extend them to the visual modality. Sequences of 8 items (allowing to use a sufficient number of trials while still expecting clear complexity effects) were presented to a group of participants in both a visual and in an auditory form (in different experimental blocks), using the same violation detection paradigm. Due to constraints in the perception of repeated visual stimuli, stimulus onset asynchrony was lengthened to 400 ms in both auditory and visual sessions, resulting in a sequence duration of 3000 ms (compared to 1800 ms in experiment 2).

Methods

Participants

Participants were eighteen healthy volunteers ($M_{\text{age}} = 25.5$, $SD = 5.7$, 15 women). They all gave written consent to participate and were paid for their participation.

Stimuli

Fifteen binary sequential patterns of eight items were used for this experiment (all were composed of 4 items A and 4 items B). All were previously used in experiment 2. They

were selected based on their LoT complexity, in order to preserve a large and homogenous distribution of complexity values. The same sequences were presented to participants in auditory and visual forms (in different blocks). Auditory sequences were composed of the same two complex tones as in the previous experiments. Visual sequences were composed of two colored Gabor patches presented in the center of the screen (a red Gabor patch with 45° orientation, and a green patch with 135° orientation). Stimulus duration was 200 ms with 200 ms inter-stimulus interval in both modalities. Total sequence duration was 3000 ms.

Procedure

The same procedure and material as in previous experiments were used in the auditory blocks. Participants were instructed to fixate the center of the screen in the visual blocks. Each sequence was tested in a short block of approximately 2.5 min., followed by a pause. Since each sequence was presented twice (i.e. in the visual and in the auditory form), the experiment was divided in two sessions of fourteen blocks, separated by a longer pause. Each pattern appeared once in a given session, which comprised equal numbers of auditory and visual blocks. Order of blocks within each session was randomized for each participant. Each block comprised 35 trials (sequence of 3000 ms and inter-trial duration of 1000 ms). The habituation phase contained at least eight trials, alterations of the sequence occur on 1/3 of the remaining trials (i.e., 9 deviant trials). As before, deviant items only appeared within the second half of the sequence (positions 5, 6, 7 or 8). Participants were asked to press the button, as quickly as possible, as soon as they detected a change in the sequence.

Data analysis

LISAS were computed for each sequence per modality per subject using correct response times and miss rate (after removing 2.4% extreme response times). One participant was excluded (miss rate and number of false alarms more than $2.5SD$ above group median). The same analysis procedure described before was adopted with the sole exception that some analyses included modality as a categorical two-levels (auditory vs. visual) predictor.

Results and discussion

Complexity and modality effects

To assess the impact of LoT complexity and modality on performance, we first computed a mixed model including complexity and modality as fixed factors and participants as a random factor. Effects of LoT complexity ($t(486.0) = 3.08$, $p < .003$), modality (average LISAS of 1110 ms in visual blocks vs. 780 ms in auditory blocks; $t(486.0) = 14.1$, $p < .0001$) and their interaction ($t(486.0) = 3.19$, $p < .002$) were significant. The slope of the complexity effect was steeper in the visual than in the auditory modality (+54 ms vs. +22 ms, $t(486) = 3.19$; see Figure 7). Separate analyses indicated that LoT complexity was a strong predictor of performance for visual sequences ($t(233.0) = 6.82$, $p < .0001$; $r = .76$), and also for auditory sequences ($t(237.0) = 3.76$, $p < .0003$; $r = .63$).

Note that, although the effects appeared stronger for the visual modality, the average performance in the visual and the auditory modality were highly correlated ($r = .85$, $p < .0001$). This suggests a common, cross-modal mechanism behind the observed differences in performance between sequences. It can however be acknowledged, here again, that this is not fully explained by complexity. Indeed, the residuals of linear regressions with LoT complexity in the visual and in the auditory modality (using average LISAS per sequence)

were still correlated (r = .73, $t(13)$ = 3.92; p < .002).

The number of false alarms per sequence was related to the task modality (mean number of FA: 0.58 in auditory blocks; 1.16 in visual blocks; difference between modalities: $t(487.0)$ = 5.73, p < .0001) but not to sequence LoT complexity ($t(487.0)$ = 0.08, p = .94).

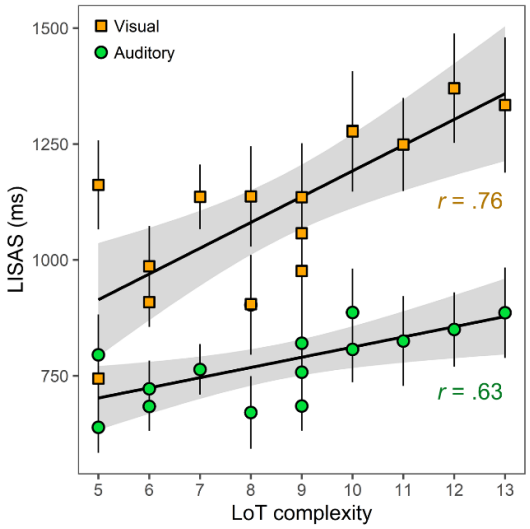

Figure 4: : Linear relationship between LoT complexity and violation detection task performance (LISAS) for each modality in experiment 5 (8-items auditory and visual sequences).

Surprise effects

As in previous experiments, a surprise effect was also observed, in both modalities, when considered independently; deviants inducing rare transitions were more easily and quickly detected than frequent ones (effect of surprise in a mixed model with auditory trials only: $t(237.0)$ = 3.87, p < .0002; r = –.65; with visual trials only: $t(233.0)$ = 6.79, p < .0001; r = –.78). This effect suggests that a common, or at least a similar, mechanism is at play in the encoding of statistical regularities characterizing the sequences in both the visual and the auditory modality.

In order to test whether evidence for sequence compression could still be observed once the surprise effect was taken into account, we performed a comparison of mixed effects models. The null model included the surprise predictor, the modality as a categorical predictor and the subject random factor. It was compared against a full model including the same predictors, with the addition of the LoT complexity. This comparison was highly significant (χ (1) = 19.0, p < .0001), indicating that goodness of fit improved when LoT complexity was added to the model. All three fixed effects were significant in the full model (LoT complexity: $t(486.0) = 4.39$, p < .0001; surprise: $t(486.0) = 4.54$, p < .0001; modality: $t(486.0) = 14.2$, p < .0001, see Table 1).

Overall, the results obtained in the visual modality are very similar to those obtained in the auditory modality in the same and in previous experiments. We however observed here stronger effects of both LoT complexity and surprise. It should be noted that the overall difficulty of the task increased in the visual modality (as indicated by higher average miss rates per sequence; 22% vs. 11% , $t(14) = 7.49$, p < .0001; and longer average response times per sequence; 831 ms vs. 645 ms, $t(14) = 10.5$, p < .0001). 8-items visual sequences may have been more difficult to encode than 8-items auditory sequences, due to the known superiority of the auditory processing system in the processing of temporal sequences and rhythms (Freides, 1974; Patel et al., 2005). This increased encoding difficulty in the visual domain may have in turn lead to an increased need for the "mental sequence compression" mechanism that our language of thought aims to describe.

The present experiment also extends the results of experiment 3 by using a slower presentation rate. Indeed, although the participants in experiment 5 appeared to respond faster (in the auditory blocks) than those from experiment 3, the same relationship with

complexity was found (correlation of performance with LoT complexity of .62 and .63 respectively). It suggests that the effect of complexity is robust across sequence durations (as expected given than LoT complexity is based on abstract sequence patterns). More importantly, the fact that a similar complexity effect was observed irrespective of the modality is consistent with the idea of "language of thought" used to compress sequential information at an abstract, symbolic level. Such an assumption has already been supported by results from Yildirim and Jacobs (2015), who showed cross-modal transfer of sequence knowledge: learning to categorize visual sequences facilitated the categorization of auditory sequences and vice versa. In fact, the language we used here was initially designed to represent visually presented, geometrical patterns (Amalric et al., 2017). The present results thus confirm that this language can account for sequence representations in various modalities and presentation contexts.

Further analysis: comparison with other measures of sequence complexity

The complexity, or the "compressibility" , of a sequence can be assessed in several ways, and various measures have been previously proposed in the psychological literature (e.g. Aksentijevic & Gibson, 2012; Alexander & Carey, 1968; Gauvrit et al., 2014; Glanzer & Clark, 1963; Griffiths & Tenenbaum, 2003; Mathy & Feldman, 2012; Psotka, 1975; Vitz, 1968; Vitz & Todd, 1969). In this last section, we examined how our LoT complexity value compares to six other measures in predicting task performance over different sequence lengths. These measures were the following:

Chunk complexity: following the observation that the number of chunks (or runs) is correlated to performance in sequence encoding tasks (e.g. Glanzer and Clark, 1963), we here define chunk complexity using the formula proposed by Mathy & Feldman (2012), which they showed to correlate with performance in the encoding of series of digits:

$$Chunkcomplexity = \sum_{i=1}^{K} \log_2 (1 + L_i)$$

Where K is the number of chunks and L_i the length of the i-th run. Note that contrary to Mathy & Feldman, (2012), whose sequences where composed of digits and chunks defined based on constant (positive or negative) increments from one digit to the next (e.g. "1234" , "7531"), we here simply define chunks as consecutive repetitions of the same item, e.g. the sequence "AAABAA" has a 3 chunks, and a chunk complexity of *$log_2(4)$ + $log_2(2)$ + $log_2(3)$*.

Entropy of apparent transition probabilities: here we computed the Shannon entropy (H), a measure of information that quantifies the uncertainty of a distribution, of the probability of pairs of items, (AA, AB, BA, BB), in order to capture the effect of order-1 transition probabilities (Maheu et al., 2020). Given that the probability of a given pair is defined as $p(X,Y) = p(X) \cdot p(Y|X)$, H is computed as follow:

$$H = -[p(A) \cdot p(A|A) \cdot (\log_2 p(A) + \log_2 p(A|A)) + p(A) \cdot p(B|A) \cdot (\log_2 p(A) + \log_2 p(B|A)) + p(B)$$

We used the convention that $0 * \log_2(0) = 0$ when null probabilities occurred.

Lempel-Zif complexity (Lempel & Ziv, 1976) is derived from the popular lossless data compression algorithm, the Lempel-Ziv (LZ) algorithm. Briefly, the LZ algorithm works by scanning the sequence from left to right and adding to a vocabulary each new substring it has never encountered before. LZ complexity is the number of substrings in this vocabulary once the scan is complete. Beyond the field of computer data compression, LZ complexity has been used in various domains, for instance, to measure the complexity of rhythmic patterns in music (Thul & Toussaint, 2008), or to assess the complexity of human (Peng et al., 2014) or non-human behaviors (Belkaid et al., 2020).

The number of **subsymmetries** is the number of symmetric sub-sequences of any length within a sequence. For instance, the sequence "AABBAB" has two symmetric subsequences of length 2 ("AA" and "BB"), one of length 3 ("BAB"), and one of length 4 ("ABBA"), for a total of four subsymmetries. This measure was proposed by Alexander and Carey (1968) and shown to be negatively correlated to performance in perception and production tasks with visual and auditory patterns (Alexander & Carey, 1968; Toussaint & Beltran, 2013).

Change complexity is an advanced measure proposed by Aksentijevic and Gibson (2012), based on the notion of "change" (the inverse of invariance), computed across all sub-sequences contained in a sequence, and showing interesting properties such as a sensibility to periodicity and symmetries.

Algorithmic complexity was introduced by Gauvrit et al., (2014, 2016) and Soler-Toscano et al., (2014). It is based on the mathematical definition of Kolmogorov-Chaitin complexity (Chaitin, 1969; Kolmogorov, 1968) and derived from the probability of obtaining a given pattern in the output of a randomly chosen Universal Turing Machine that halts.

LoT chunk complexity. Note that the alternative measures of complexity tested here, which provide a unique metric for each pattern, are quite conceptually different from the one we propose. LoT complexity is based on the proposal that humans possess a language of thought, composed of a small number of atomic rules which they use recursively to recode the abstract structure of the pattern in a compressed form. Such a recursive representation differs radically from, say, the mere counting of the number of chunks. However, it is possible to combine the two ideas. The formal language we proposed produces many legal expressions for each sequence (the number of possible expressions can reach several tens of thousands for a sequence of length 16), which correspond to distinct "parses" of the same sequence. We initially assumed that the shortest expression is always selected, and thus that LoT complexity is equal to the shortest possible description using this language. However, it is unclear whether humans could ever search such a vast space of possibilities. A more plausible hypothesis is that participants begin by chunking the sequence into groups of identical items, and only then compress it by detecting repetitions of those chunks (for a similar proposal, see E. Leeuwenberg, 1969; Leeuwenberg, 1971). According to this idea, the shortest sequence should only be accepted when its proposed parsing coincides with chunk boundaries. Consider the sequence "ABBAAB" , which consists of 4 chunks [A] [BB] [AA] [B]. According to our language, its optimal description is [AB] [BA] [AB] (i.e. 3 repetitions of the stay-change program; LoT complexity = 5), but that representation does not coincide with chunk boundaries. Interestingly, the data suggested that the shortest description may not be optimal in similar cases (see *Experiment 2, Results and discussion*). To test this idea, we recomputed LoT values restricted to chunk-preserving expressions (i.e., excluding expressions producing "A][A" or "B][B"). We called this new LoT complexity the **LoT chunk complexity.** Its

value was higher than the original one for 58% of sequences (and remained the same for the others). For instance, the sequence "ABBAAB" from the previous example, when described as four chunks [A] [BB] [AA] [B], has an LoT-chunk complexity = 9. We tested LoT chunk complexity as another potential predictor of behavioral performance.

Model comparison

To conduct the analyses, data were pooled from all previous experiment with auditory sequences (using LISAS to index task performance), excluding super-deviants trials. Unfortunately, due to the nature of algorithmic complexity (derived from the output frequency for a pattern using small Turing machines, which decreases rapidly with sequence length), no values were available for the ten length-16 patterns that we used in experiment 1, as well as for one length-12 pattern used in experiment 2. Those sequences were therefore excluded from some analyses. The sequence "AAAAAA" from experiment 4 was also excluded. Consequently, a first pooled dataset, for which all 8 different predictors could be compared, included performance with 77 different auditory sequences (and 88 different participants), of length 6 (n = 31 sequences), length 8 (n = 35) and length 12 (n = 11), while a second one, for which 7 different predictors were compared, also included sequences of length 16 (n = 88 sequences, 113 participants).

To assess whether one measure was a better predictor of task performance, we first computed different mixed models, which all included the predictor of interest as the only fixed effect and participants as a random effect (note that this is a way to control for the fact that different participants coming from different experiments, with different sets of stimuli, were pooled together). We then report the Akaike information criterion (AIC)

as an indicator of goodness of fit which penalizes for model complexity (i.e. the number of predictors); the model with the lowest AIC value being considered the best (or with lowest "Δ (AIC)" value, i.e. the relative difference in AIC with the best model: for a model *i*, $\Delta\ (\mathrm{AIC})_i = \mathrm{AIC}_i - \mathrm{AIC}_{min}$). Note that we also report the Bayesian information criterion (BIC) which, in addition, scales the strength of penalization by the (log) number of data points. Second, since, as we reported earlier, surprise derived from the learning of transition probabilities may strongly affect the performance in such violation detection task, all these models were computed again, this time including surprise as a fixed effect covariate.

Dataset with sequences of length 6, 8 and 12.

Sixteen different mixed models were fitted using datasets with sequences of length 6, 8 and 12. As illustrated in Figure 8A, model fit, as indexed by the Δ (AIC) value, always improved when the surprise associated to the deviants was included in the model. This finding confirms that the effect due to transition probabilities needs to be taken into account when assessing responses to deviants in the violation detection paradigm. The improvement in model fit was smallest for the model with entropy. This effect was expected since entropy and surprise are two tightly related information measures (i.e. Shannon entropy is the average of Shannon surprise).

When considering either only single-predictor models (i.e. without the surprise covariate) or two-predictors models (i.e. with surprise), the two best models were the ones with our modified version of LoT complexity (i.e. LoT-chunk, with the "no-splitting" chunks constraint) followed by the one with original LoT complexity (see Table 2). In order to test

whether the differences in the raw AIC values were meaningful, we computed the Akaike weights for this set of 14 models. Akaike weights can be interpreted as the probability that a given model is the best model of the set (Wagenmakers & Farrell, 2004). Akaike weight was .99 for the LoT-chunk complexity (+ surprise) model, .01 for the LoT complexity (+ surprise) model, and below .01 for all other models (see Table 2 and Figure 8A).

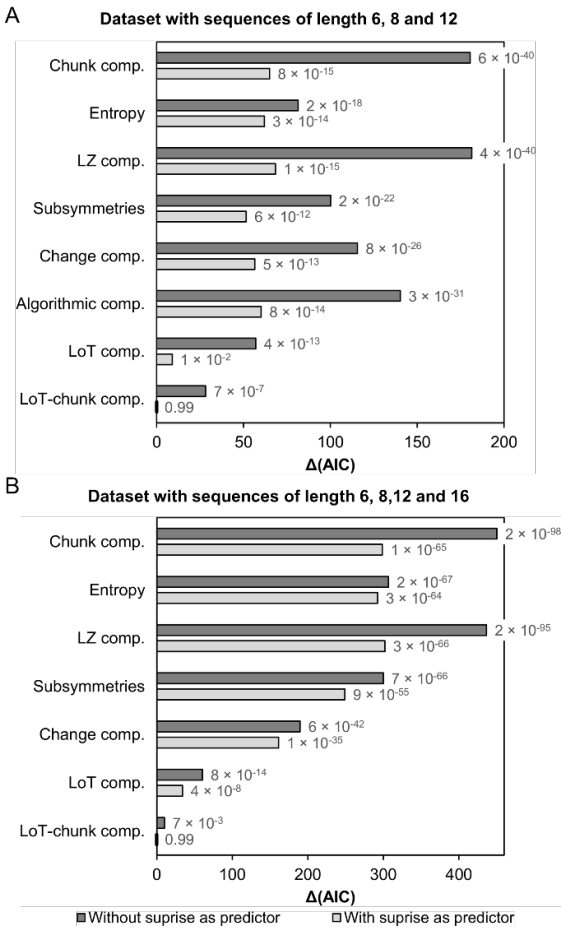

Figure 8: Δ (AIC) for the sixteen mixed models tested using the dataset including the task performance (LISAS) for sequences of length 6, 8 and 12 (A), and for the twelve different mixed models tested using the dataset with sequences of length 6, 8, 12 and 16 (B). The fixed effect of interest is indicated along the vertical axis (all models included participants as a random effect and could include surprise as a covariate — light gray bars). Akaike weight for each model is also reported. The model with lower AIC (Δ (AIC) = 0) is indicated by short dark vertical line on the vertical axis.

Although the correlations of performance with LoT complexity in experiments 2, 3 and 4 (lengths 6, 8 and 12) were small in comparison with experiment 1 (length 16), LoT complexity again appears as the best predictor of performance in the violation detection task with sequences of length ≤ 12. Notably, the constraint of excluding, for each pattern, the expressions that resulted in the splitting of a chunk (before the selection of shortest expression) improved the fit to the behavioral data. This observation suggests that participants did not always find the best way of coding some patterns (best in the sense of the language of thought considered here) because of a propensity to perform an initial chunking solely based on consecutive runs of identical items.

The next best model was the one with the "number of subsymmetries" predictor (and including the surprise covariate), suggesting that it also provides a good measure of the psychological complexity of patterns. However, while this appeared true here using statistical models partially controlling for sequence length (i.e. by including participant index as a random factor, since each participant performed the task with only one given sequence length), this measure appears inappropriate to predict complexity across different lengths. Indeed, when we computed the Pearson correlation of average LISAS per sequence for the pooled dataset (sequences of length 6, 8 and 12), we obtained a *positive* correlation value of .39. Such positive correlation is in conflict with the presupposition that patterns containing more symmetries should be simpler. It is due to the fact that the number of subsymmetries tends to increase with sequence length. These correlations were actually negative when each length was considered independently ($r = -.44$ for length 6; $r = -.54$ for length 8; and $r = -.58$ for length 12). This is illustrated in Figure 9, where the average LISAS for each sequence is presented in relation to each complexity measure (see also Figure S6 and S7 for the equivalent with reaction times and miss rates). To summa-

rize, although this measure is quite good in predicting the complexity of sequences of a given length, it is not efficient in predicting the variations in complexity due to sequence length.

Another similar limitation applies to algorithmic complexity, where the correlation observed across lengths ($r = .79$) is mostly because this value presents excessive discontinuities with length: algorithmic complexity ranges roughly between 14 and 16 for length 6; between 19 and 23 for length 8; and between 31 and 35 for length 12 (see Figure 9). Such large increases in complexity with length are not consistent with behavior. Again, LoT complexity provides a better correlation with the present behavioral data across a large range of sequence lengths, because it correctly predicts that, for instance, some 6-items long sequences can be more complex than some 12-items ones.

	Dataset with sequences of length 6, 8 and 12				***Dataset with sequences of length 6, 8, 12 and 16***			
Model fixed effect(s)	Log-lik.	Δ(AIC)	Δ(BIC)	w(AIC)	Log-lik.	Δ(AIC)	Δ(BIC)	w(AIC)
LoT comp.	-14886	57	51	4.0×10^{-13}	-16653	60	55	7.8×10^{-14}
LoT comp. + Surp.	-14861	9	9	1.1×10^{-2}	-16639	34	34	4.1×10^{-8}
LoT-chunk comp.	-14872	28	23	7.4×10^{-7}	-16628	10	4	6.8×10^{-3}
LoT-chunk comp. + Surp.	-14857	0	0	0.99	-16622	0	0	0.99
Chunk comp.	-14948	180	175	6.4×10^{-40}	-16848	450	445	1.6×10^{-98}
Chunk comp. + Surp.	-14889	65	65	7.5×10^{-15}	-16771	299	299	1.4×10^{-65}
Entropy	-14899	81	76	2.0×10^{-18}	-16776	307	301	2.3×10^{-67}
Entropy + Surp.	-14888	62	62	3.4×10^{-14}	-16768	292	292	3.3×10^{-64}
LZ comp.	-14948	181	176	3.9×10^{-40}	-16841	436	431	1.6×10^{-95}
LZ comp. + Surp.	-14891	68	68	1.4×10^{-15}	-16773	302	302	2.6×10^{-66}
Subsymmetries	-14908	100	94	1.2×10^{-22}	-16773	300	294	8.8×10^{-55}
Subsymmetries + Surp.	-14883	52	52	6.2×10^{-12}	-16746	249	249	1.3×10^{-17}
Change comp.	-14916	116	110	7.6×10^{-26}	-16718	190	184	6.2×10^{-42}
Change comp. + Surp.	-14885	57	57	5.3×10^{-13}	-16703	161	161	1.0×10^{-35}
Algorithmic comp.	-14928	140	135	3.4×10^{-31}		*N.A.*		
Algorithmic comp. + Surp.	-14887	60	60	8.4×10^{-14}		*N.A.*		

Note. All models included participants as a random effect, and either one or two fixed effect(s) (i.e. "+ Surp." : with additional surprise fixed effect). Log-lik. = log of the maximum likelihood for the model. Δ (AIC) = AIC difference with the model with the lowest AIC value (where AIC is the Akaike Information Criterion). Δ (BIC) = BIC difference with the model with the lowest BIC value (where BIC is the Bayesian Information Criterion). w(AIC) = Akaike weight.

Dataset with sequences of length 6, 8, 12 and 16.

Fourteen different mixed models (with participants as a random effect) were here fitted, using the same dataset as before to which was added data from 11 sequences for which algorithmic complexity value was not available (thus now with sequences of length 6, 8, 12 and 16). The same predictors as above were used, with the exception of algorithmic complexity. Here again, as illustrated in Figure 8B, goodness of fit systematically increased when surprise was included. LoT-chunk complexity and LoT complexity (with or without surprise as a covariate) were again the best predictors of performance (see Table 2). As opposed to the previous set of analyses in which the data from experiment 1 (length 16) was not included, the model with change complexity performed clearly better than the one with the number of subsymmetries. The long sequences used in experiment 1 indeed presented important differences in their number of subsymmetries (e.g. 56 for $(AB)^8$ vs. 32 for $(A^4B^4)^2$), which were clearly not predictive of performance. Consequently, and as stated earlier, the number of subsymmetries does not appear as a good predictor of task performance across different sequence lengths. Change complexity also appeared as a much better predictor when performing a simple linear regressions on average LISAS per sequence (see Figure 9), resulting in an $r = .81$, which is close to the one obtained with LoT complexity ($r = .82$). It indicates that change complexity can also be a good measure of the psychological complexity of a sequence regardless of its length. It must however be noted that, contrary to the mixed models, these linear regressions using data averaged over participants did not control for the variance accounted for by surprise, or due to inter-subject variability. Important variations were indeed observed across participants regarding the correlation with complexity (especially for experiments with shorter sequences). When computed at the level of individual participants, the correlation

with LoT complexity appeared on average stronger (mean $r = .31$, $SD = .32$) than the one with change complexity (mean $r = .23$, $SD = .30$) ($t(112) = 3.54$, $p < .0006$).

With both datasets, two measures performed poorly, LZ complexity and chunk complexity. Contrary to our language, the LZ algorithm has the advantage to be able to quickly "parse" any sequence of any number of different characters, by building for each sequence its own vocabulary of substrings. Its adequacy to human behavior, however, appears limited since, when scanning the sequence from one item to the next, it does not necessarily take into consideration runs of repeated items ("AAA" can be described with two substrings, "A" and "AA") and fails to capture repeating patterns. This deficiency is especially striking for a low LoT complexity sequence such as $(A^2B^2)^4$ (i.e., AABBAABB...), where 8 substrings are present in the vocabulary at the end of scanning (the first four substrings encountered by the algorithm are "A" , "AB" , "B" , "AA"). This gives this sequence the lower level of LZ compressibility among those tested, which is clearly not predictive of performance.

Similarly, "chunk complexity" , like other methods solely based on quantifying chunks (number of chunks, chunks length, or a combination of both), is strongly dependent on how chunks are defined. Here, since chunks are defined as runs of identical items, the complexity of sequences containing alternations tends to be overestimated (e.g. "ABABABAB" has 8 chunks). Assessing complexity based on chunks therefore requires first building a model that defines what chunks are for the sequence processing cognitive system, which is not trivial. Another limitation of this measure is an excessive sensitivity to sequence length. In the absence of any recursive compression, complexity increases linearly with the number of chunks. Allowing compression based on consecutive repetitions of chunks (chunks of chunks), as in the LoT model proposed here, appears to be a better strategy for

predicting the subjective complexity of sequences. Note that, notwithstanding the aforementioned concerns, change complexity captures relatively well the complexity variations due to both structure and length (Figure 9). This may be due to the fact that change complexity is computed within substrings of all possible lengths, which is another way to capture regularities at multiple hierarchical levels.

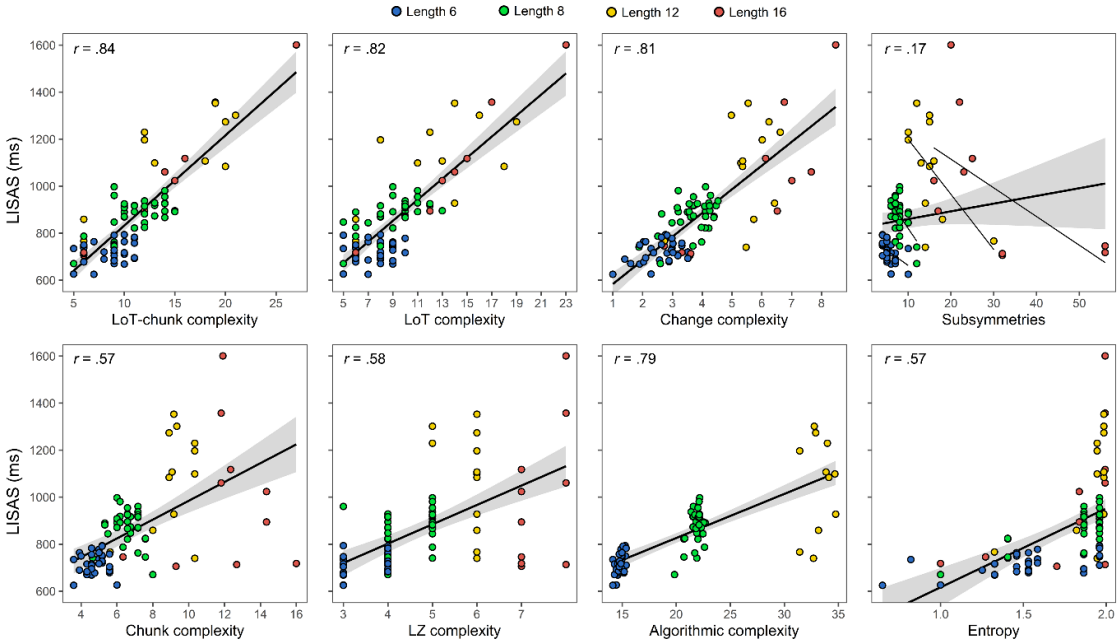

Figure 5: : Linear regressions of average performance per sequence (LISAS, in ms) with eight different predictors of interest when combining data from experiments with auditory sequences of 4 different lengths. Each marker corresponds to one sequence. Sequences of different lengths are indicated by different markers only for illustration purposes (the length factor was not taken into account when computing the correlation, r, coefficient). 16-items long sequences (as well as one 12-items sequence) could not be included in the regression with algorithmic complexity. Regressions lines for each sequence length were added in the subsymmetries plot, in order to illustrate the fact that negative correlations were observed when each length was considered separately. Note that the average performance data presented here does not take into account the effects of surprise, inter-subject, or inter-experiment variability.

General discussion

The main goal of this series of experiments was to evaluate the mental representation of binary sequences and to test the adequacy of a formal language of thought previously proposed to account for geometrical sequences (Amalric et al., 2017). Similar models were proposed in the past (e.g. Leeuwenberg, 1969; Restle, 1970; Simon & Kotovsky, 1963), but they were not submitted to a full experimental validation, particularly in comparison to the most recent approaches to sequence complexity assessment. Moreover, we sought to distinguish the effects related to statistical transition-probability learning, which are unavoidable when dealing with temporal sequences of stimuli, from the putative influence of rule-based encoding. Across five different experiments with sequences of different lengths, in the auditory but also in the visual modality, we found consistent evidence that a significant part of the variations in sequence encoding performance (as indexed by the capacity to detect sequence violations) was explained by the length of the shortest possible description of the sequence in the proposed formal language (i.e. LoT complexity). These results are consistent with the idea that upon hearing or seeing a binary sequence, subjects form an internal representation corresponding to an abstract and compressed form of the sequence content. It is remarkable that a language merely composed of two simple instructions ("same" and "change") and their recursive embeddings suffices to model the formation of such a representation. The complexity measure derived from this language was indeed better predictive of the degree of psychological complexity than other sophisticated approaches designed as alternatives to the non-computable Kolmogorov complexity (Aksentijevic & Gibson, 2012; Soler-Toscano et al., 2014).

The assumption that the length of the shortest description in the formal language corre-

sponds to perceived sequence complexity was further corroborated by subjective complexity rating (experiments 1 and 2). Moreover, we found that sequence structure was not the only information encoded by the participants, since the level of surprise derived from the statistical estimation of transition probabilities also consistently helped explaining the variance in violation detection performance. The effects of surprise and of complexity on responses to violations were found to vary differently depending on sequence length, thus providing new insights on how the human brain makes predictions in temporal sequences.

The predictive power of the LoT approach was most notable for the longest sequences tested, in particular for 16 items long sequences (experiment 1; $r = 98$). Indeed, massive differences in miss rates were observed between the sequences predicted to be the least complex (A^nB^n patterns, with LoT complexity = 6) and those predicted to be the most complex (a set of 10 instructions, LoT complexity = 23), suggesting that subjects simply could not learn the latter efficiently, even after eight or more repetitions. An additional prediction of LoT was verified, namely the fact that the four sequences based on the A^nB^n pattern were associated with a similar performance level, regardless of n (= 1, 2, 4, or 8). In the language, this is because the complexity of a repetition is proportional to the logarithm of the number of repetitions, rounded up to the nearest integer. For a total number of 16 items, it therefore does not matter when the sequence is decomposed in 2 chunks of 8, 4 chunks of 4, 8 chunks of 2, or 16 chunks of 1: the sum of the weights remains unchanged, leading to a LoT complexity of 6 bits in all cases — and indeed, the observed performance remained stable across such a broad variation ranging from huge chunks to pure alternation (see Figure 3).

The correlation of performance with LoT complexity decreased in subsequent experiments using increasingly shorter sequences, until it became almost absent for sequences compris-

ing only six elements. Rather than an indication of an intrinsic limitation of the language for describing very short binary patterns, we believe that a significant part of this effect relates to differences in working memory demands. The number 6 indeed falls within the usual limits for the number of elements that can be stored in working memory, which is around 7± 2 items when there is no compression (Mathy & Feldman, 2012; Miller, 1956). Thus, subjects could have solved the violation detection task without compression, purely by storing each 6-items sequence "as is" in working memory. Similarly, 8-items sequences could have been stored as a mere series of "chunks" , which are thought to be the units of encoding in working memory (Cowan, 2010; Cowan et al., 2004; Luck & Vogel, 1997; Mathy & Feldman, 2012), without any recursive embedding. All in all, an increasingly greater need to rely on compression would explain why the predictive power of LoT complexity increases with sequence length.

Although the definition of working memory chunks as "a collection of elements having strong associations with one another" (Cowan, 2001; Gobet et al., 2001) is too vague to be rigorously tested using the present data, it is easy to imagine that both conceptions can lead to similar predictions (sequences composed of a small number of small chunks also have a short description in our language). Note however that, when considering all tested sequences, LoT complexity outperformed the "chunk complexity" predictor, for which chunks are defined using consecutive repetitions of the same item. In fact, a crucial feature of our theory lies in going beyond a simple concatenation of chunks and forming recursively embedded or nested representations, that is the ability to represent "chunks of chunks" or "repetitions of repetitions" . Indeed, the construction of recursively nested structured has been proposed as a core human ability, which sets us apart from other primates (Conway & Christiansen, 2001; Dehaene et al., 2015; Fitch & Hauser, 2004;

Hauser et al., 2002). Our results support the idea that the inclusion of such feature is essential to explain human behavior when working memory capacity is exceeded and compression is most beneficial.

The fact that we reached such a conclusion using the simplest type of temporal sequences (binary sequences) and a simple deviant detection task (rather than the more demanding recall, completion or production tasks using in the previous literature) is consistent with Fitch's "dendrophilia hypothesis" (Fitch, 2014) which states that "humans have a multi-domain capacity and proclivity to infer tree structures from strings" even in the simplest cases. The present work provides a foundation for future experiments in non-human primates, which would allow us to test the second aspect of this hypothesis, namely that this capacity for building recursive tree structures is only available to humans (Dehaene et al., 2015; Fitch, 2014; Hauser et al., 2002). In non-human primates, we postulate that a simpler language will suffice to account for sequence coding.

Numerous other frameworks for the estimation of pattern complexity have been proposed in the past, such as change complexity (Aksentijevic & Gibson, 2012), algorithmic complexity (Gauvrit et al., 2014, 2016; Soler-Toscano et al., 2014), subsymmetries (Alexander & Carey, 1968) or entropy (see also Glanzer & Clark, 1963; Psotka, 1975; Vitz, 1968, 2019; Vitz & Todd, 1969). These models are often based on quantitative aspects of information, such as the length, the number of transitions or runs, the probability of those transitions, the number of symmetries, or the number of changes. Although they all show some level of success in predicting behavior, they fail to capture recursive nesting, which as noted above seem to be an essential factor in human cognition (Dehaene et al., 2015; Hauser et al., 2002). The same limitation apply to the Lempel-Zif data compression algorithm, which compresses sequences by storing in memory a set of unique substrings that can

occur at different location in a sequence. Although it may seem psychologically relevant, this specific algorithm is unable to consider relationships between substrings mediated by an abstract, higher-level operation of repetition or change, as a LoT model does. In addition, this algorithm does not take advantage of contiguous repetitions. Conversely, the notion of repetition with variations is central to the success of our language. Others have also proposed that humans possess a "repetition detector" , as they are much better to learn repetition-based grammars than other forms of simple grammars (Endress et al., 2007). Repetition detection may already be present at birth, which suggests that it may be an innate neurocognitive function, perhaps essential for language acquisition (Gervain et al., 2008). It may therefore not be surprising that nested repetition with variation suffices to account for the human memory for sequences, and that models that do not incorporate it struggle to replicate human behavior.

Following others in the domain of concept learning (e.g. Piantadosi et al., 2012, 2016), the approach adopted here assumes that binary sequences are encoded using a specific cognitive system that manipulates abstract, symbolic representations — a language of thought with recursive calls to a limited number of primitive operations. Thus, the present proposal does not merely provide a numerical value for complexity, but also a parse tree and a precise internal format of representation, both of which could possibly be tested in future behavioral or brain-imaging experiments.

Although the current study is based on the use of a "fixed" language, with predetermined rules and associated weights, some evidence suggests that a better description of human behavior can be achieved by incorporating a probabilistic component to the modeling. This approach, advocated by Piantadosi & Jacobs (2016) under the term *probabilistic language of thought* (pLOT), consists in using Bayesian probabilistic inference to estimate

the likelihood of the existence of some set of rules (a proposed formal language), given the observed data. It has been shown to be especially efficient in modeling concept learning, for instance by replicating the patterns of errors throughout learning (Goodman et al., 2008; Piantadosi et al., 2012, 2016). This approach was also adopted to investigate how humans assess randomness in their environment. Human biases in subjective randomness judgments (e.g. Kahneman & Tversky, 1972; Lopes & Oden, 1987) could be explained by assuming that the representation of randomness results from a statistical inference about the processes that generated the sequence, i.e. an estimation of the probability that a given regular process produced it (Griffiths et al., 2018). A good fit to human behavior was obtained without using the full power of Turing machines, but only finite-state automata with a stack, which are able to recognize repetition, alternation or symmetry (Griffiths et al., 2018; Griffiths & Tenenbaum, 2003). Thus, despite fundamental differences (notably, deterministic versus probabilistic languages), the pLOT theory shares with our approach the need to consider similar types of primitive operations. Given the strong links between subjective randomness and complexity, we can reasonably expect that our formal language may also predict whether a pattern is perceived as random or not – this possibility remains to be tested in future work.

Beside the learning of conceptual knowledge and work on subjective randomness, a pLOT approach was also used to model the learning of spatial sequences: to study the cross-modal transfer of sequence knowledge (Yildirim & Jacobs, 2015), and to investigate the adequacy of the language of geometry (Romano et al., 2018). Indeed, by using the behavioral data from the octagon task of Amalric et al. (2017), Romano et al. (2018) showed that the primitives included in the language of geometry were all required in order to best account for human behavior. In spite of its successes, a number of questions and potential

limitations of the LoT approach remain. First, the construction of our formal language implied methodological choices that could be considered as arbitrary or at least requiring more experimental validation. The primitive instructions included in our formal language were chosen for their alleged simplicity and because they suffice to represent any binary sequence. Other primitives could be tested (e.g. counting and a system of arithmetic; or temporal inversion or "mirroring" , see Jiang et al., 2018). Furthermore, modifications of the weights associated with each instruction or their number of repetitions may lead to different estimates of complexity. Finding the correct language for a given population is crucial, especially in the context of the debate on the uniqueness of human sequence processing skills, and specific statistical methodologies need to be developed for this purpose. As mentioned earlier, the pLOT approach which, using Bayesian inference, allows to find the most likely concepts and rules from a grammatically structured hypothesis space containing several candidates, appears to be a very promising approach for that purpose (Goodman et al., 2008; Piantadosi & Jacobs, 2016; Romano et al., 2018). Nevertheless, we also found that some of the minimal expressions produced by this language did not fit well with the way participants represent some sequences. The addition of the constraint that the minimal parse tree should respect the chunks or runs of consecutive repetitions, and never split any such chunk, was found to lead to a noticeable improvement in model fit. We speculate that this finding reflects the way participants build their internal representation of sequences: since the space of possible programs is immense, they would restrict the search to only those programs that, at the lowest level, generate the observed consecutive runs in the sequence. The perceptual dominance of the runs could act as a bottleneck, an initial grouping that would then restrict the sequence parsing process (as is sometimes assumed in some complexity estimation models; see e.g. Vitz & Todd,

1969). A better characterization of this parsing process during sequence learning could help address the current limitations of our language.

Another limitation is that, although we argued that the capacity to represent sequences using hierarchically embedded or nested descriptions is an essential feature of human behavior (Dehaene et al., 2015), about half of the minimal expressions for the sequences that we used included only two hierarchical levels (a single level of embedding) (the average hierarchical depth was 2.5). Only a few sequences such as AABBABABAABBABA explicitly required repetitions of repetitions of repetitions. Although our model correctly predicted their subjective and objective complexity (see Figure 3), and although embedding is an effective compression process, more research is needed to probe whether human participants always consider such deep levels of embedding as beneficial in the processing of short auditory sequences. Increasing the hierarchical depth may imply an additional processing cost, making it useful only in specific situations (e.g. for more demanding learning tasks or with long sequences).

Finally, our approach assumes that the mental compression of sequences does not necessarily occur at the level of the sensory events (i.e. grouping contiguous identical elements) but at the more abstract level of the relationships between events. Besides its success in predicting the psychological complexity of sequences of tones, one argument in favor of such an abstract symbolic representation is that it fitted equally well the complexity of visual binary sequences. However, it could be proposed that the mental encoding of temporal sequence does not involve a modal, domain-general processing mechanisms, but rather two similarly organized modality-specific systems, or even a single modality-specific cognitive system dedicated to auditory processing; visual sequences would then be converted into an auditory representation prior to compression. Indeed, we observed a lower performance

and slower responses in the visual compared to the auditory modality, a difference which has been postulated to reflect a dominance of the auditory system for the encoding of temporal information (Conway & Christiansen, 2005; Glenberg et al., 1989; Guttman et al., 2005). One potential strategy for performing the task of experiment 5 with visual stimuli could have been a subvocal naming of the items, and a maintenance in working memory using the phonological loop (Baddeley, 1992; Baddeley & Hitch, 1974). Further investigation is required to resolve these points, perhaps by relying on other sensory modalities, by testing transfer across modalities, or by using brain-imaging to determine the sensory versus higher-level nature of the brain mechanisms at play. We merely note here that activation of supra-modal prefrontal cortices has been reported during sequence processing (e.g. Huettel et al., 2002; Wang et al., 2019); that the existence of an automatic visual-to-auditory conversion in sequence processing has been challenged (McAuley & Henry, 2010); and that the existence of an abstract representation of sequences as proposed here, allowing a transfer of knowledge across modalities, is already supported by some behavioral data (see Yildirim & Jacobs, 2015).

The violation detection task used in the present study implied the learning of a specific and deterministic sequence in each block, which was repeated multiple times with predictable timings. Our results, however, indicate that the statistical properties of the original sequence were also computed in parallel to the compression process and used for prediction, since, for a given sequence, performance varied according to the level of surprise, i.e. the transition probability of the deviant sound in the context of the current sequence. For equal complexity, we observed a higher accuracy and faster response times for deviants that induced less frequent transitions. The observation that transition probability affects behavior even within a deterministic sequence (see also Maheu et al., 2020), as opposed

to the stochastic sequences that were used in previous studies of statistical learning (e.g. Huettel et al., 2002; Mars et al., 2008; Garrido et al., 2013; Meyniel et al., 2016; Meyniel and Dehaene, 2017; Maheu et al., 2019), suggests that the learning of transition probabilities between items may occur automatically and in parallel to compression in working memory. This is compatible with the large amount of evidence showing that the brain encodes statistical regularities in sensory inputs in an implicit and unconscious manner (Barascud et al., 2016; Bendixen et al., 2009; McDermott et al., 2013; Paavilainen, 2013; Saffran et al., 1996). Since the effect of surprise occurred over and above any effect of sequence complexity, it also suggests that this statistical learning system is distinct from the more strategic system based on the learning of the deterministic sequence structure. Again, this is compatible with prior brain imaging results on the local-global paradigm, which indicate that the mismatch negativity (MMN), sensitive to local transition probability, can be dissociated from the P3b response associated with the acquisition of the global sequence (Bekinschtein et al., 2009; Strauss et al., 2015; Wacongne et al., 2011).

When pooling datasets from experiments with different sequence lengths, the linear mixed models with surprise and complexity as predictors fitted the data better than models including one predictor alone, indicating that those two predictors captured distinct variance. However, one may note that the size of the surprise effect varied across experiments. Surprise and complexity showed opposite patterns, with a stronger effect of complexity for longer sequences than for short ones and conversely, a strong effect of surprise only with the shortest sequences. Given the evidence that we just cited, showing that transition probabilities are constantly being computed unconsciously, the most likely interpretation is probably that task difficulty increased with sequence length and resulted in longer response times, thus masking the contribution of statistical learning and rendering it more

difficult to detect. To evaluate this idea, future work should use event-related potentials such as the MMN, which may provide a more sensitive measure of transition-probability learning.–

Finally, we found a complexity effect even when subjects responded to "super-deviants" items, i.e. outlier sounds that could be detected without any knowledge of the sequence because their identity itself was novel. We suggest two putative interpretations of this unexpected effect. First, it could be due to the increased attentional load associated with more complex sequences. Essentially, participants would be placed in a dual-task situation of having to attend to two things are once: the complex sequence and the occasional deviants. In support of this idea, increased attentional load has indeed been found associated to sequence learning impairment in dual-task experiments (see Shanks et al., 2005). A second interpretation, within the predictive coding framework, is that deviance detection, even for extremely salient deviants, is easier for predictable than for unpredictable stimuli. Indeed, Southwell and Chait (2018) found a larger brain response to deviant stimuli within a regular sequence than within a random sequence of tones. The authors propose that it could reflect a difference in the *precision* or predictability associated with the flow of sensory information. Indeed, in addition to the prediction regarding the content of incoming stimuli (manifested by prediction error signals), recent versions of predictive coding theories also formalize the concept of precision, which corresponds to the reliability of the prediction (Auksztulewicz et al., 2017; Feldman & Friston, 2010; Heilbron & Chait, 2018; Rao, 2005). Precision would manifest itself as a gain modulation of the relevant neural units (which is tightly related to attention), with increased precision leading to an increasing sensitivity to the predicted stimuli. This theory can explain the increased and sustained neuronal responses observed in a highly predictable context (Auksztulewicz et

al., 2017; Barascud et al., 2016; Southwell et al., 2017; Southwell & Chait, 2018). The present complexity effect observed for super-deviants may thus indicate that responses to completely unexpected events were modulated by the degree of predictability of the pattern, which itself depends upon the complexity of the pattern. A precision-weighting mechanism would thus explain why greater complexity leads to slower response times to any kind of violations in our violation detection task. Overall, the distinct contributions of surprise and complexity underline the joint contributions of statistical versus rule-based information in temporal sequence processing.

Conclusion

Our study provides a first demonstration that, even after accounting for statistical transition-probability learning, responses to sequence violations can be used to uncover the properties of the abstract mental language used by individuals to encode sequential patterns. The present proposal, which takes the form of a psychologically plausible formal language composed of a restricted set of simple rules, proved to be more effective than alternative approaches in modeling the human memory for simple sequences. The observed relationship between sequence complexity and performance in the detection of violations is consistent with the idea that the brain acts as a compressor of incoming information that captures regularities and uses them to predict the remainder of the sequence. The present non-verbal passive paradigm paves the way to future neurophysiological recording studies that would probe the similarities and differences between humans and other species (Wang et al., 2015) or test the abilities of preverbal infants (Basirat et al., 2014). A fundamental question for future research is whether the same formal language can explain sequence processing in other primate species, or if such a language is unique to humans (Hauser et

al., 2002).

Abla, D., & Okanoya, K. (2009). Visual statistical learning of shape sequences: An ERP study. *Neuroscience Research*, *64*(2), 185–190. https://doi.org/10.1016/j.neures.2009.02.013

Akaike, H. (1998). Information Theory and an Extension of the Maximum Likelihood Principle. In E. Parzen, K. Tanabe, & G. Kitagawa (Eds.), *Selected Papers of Hirotugu Akaike* (pp. 199–213). Springer. https://doi.org/10.1007/978-1-4612-1694-0_15

Aksentijevic, A., & Gibson, K. (2012). Complexity equals change. *Cognitive Systems Research*, *15–16*, 1–16. https://doi.org/10.1016/j.cogsys.2011.01.002

Al Roumi, F., Marti, S., Wang, L., Amalric, M., & Dehaene, S. (2020). An abstract language of thought for spatial sequences in humans. *BioRxiv*, 2020.01.16.908665. https://doi.org/10.1101/2020.01.16.908665

Alexander, C., & Carey, S. (1968). Subsymmetries. *Perception & Psychophysics*, *4*(2), 73–77. https://doi.org/10.3758/BF

Amalric, M., Wang, L., Pica, P., Figueira, S., Sigman, M., & Dehaene, S. (2017). The language of geometry: Fast comprehension of geometrical primitives and rules in human adults and preschoolers. *PLOS Computational Biology*, *13*(1), e1005273. https://doi.org/10.1371/journal.pcbi.1005273

Auksztulewicz, R., Barascud, N., Cooray, G., Nobre, A. C., Chait, M., & Friston, K. (2017). The cumulative effects of predictability on synaptic gain in the auditory processing stream. *Journal of Neuroscience*, *37*(28), 6751–6760.

Baddeley, A. (1992). Working memory. *Science*, *255*(5044), 556–559. https://doi.org/10.1126/science.1736359

Baddeley, A. D., & Hitch, G. (1974). Working Memory. In G. H. Bower (Ed.), *Psychology of Learning and Motivation* (Vol. 8, pp. 47–89). Academic Press. https://doi.org/10.1016/S0079-7421(08)60452-1

Barascud, N., Pearce, M. T., Griffiths, T. D., Friston, K. J., & Chait, M. (2016). Brain responses in humans reveal ideal observer-like sensitivity to complex acoustic patterns. *Proceedings of the National Academy of Sciences*, *113*(5), E616–E625. https://doi.org/10.1073/pnas.1508523113

Basirat, A., Dehaene, S., & Dehaene-Lambertz, G. (2014). A hierarchy of cortical responses to sequence violations in three-month-old infants. *Cognition*, *132*(2), 137–150. https://doi.org/10.1016/j.cognition.2014.03.013

Bates, D., Mächler, M., Bolker, B., & Walker, S. (2015). Fitting Linear Mixed-Effects Models Using lme4. *Journal of Statistical Software*, *67*(1). https://doi.org/10.18637/jss.v067.i01

Bekinschtein, T. A., Dehaene, S., Rohaut, B., Tadel, F., Cohen, L., & Naccache, L. (2009). Neural signature of the conscious processing of auditory regularities. *Proceedings of the National Academy of Sciences*, *106*(5), 1672–1677. https://doi.org/10.1073/pnas.0809667106

Belkaid, M., Bousseyrol, E., Cuttoli, R. D., Dongelmans, M., Duranté, E. K., Yahia, T. A., Didienne, S., Hanesse, B., Come, M., Mourot, A., Naudé, J., Sigaud, O., & Faure, P. (2020). Mice adaptively generate choice variability in a deterministic task. *Communications Biology*, *3*(1), 1–9. https://doi.org/10.1038/s42003-020-0759-x

Bendixen, A., Schröger, E., & Winkler, I. (2009). I heard that coming: Event-related potential evidence for stimulus-driven prediction in the auditory system. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, *29*(26), 8447–8451. https://doi.org/10.1523/JNEUROSCI.1493-09.2009

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. https://doi.org/10.1037/a0016797

Brainard, D. H., & Vision, S. (1997). The psychophysics toolbox. *Spatial Vision*, *10*, 433–436.

Chaitin, G. J. (1969). On the length of programs for computing finite binary sequences: Statistical considerations. *Journal of the ACM (JACM)*, *16*(1), 145–159.

Chao, Z. C., Takaura, K., Wang, L., Fujii, N., & Dehaene, S. (2018). Large-Scale Cortical Networks for Hierarchical Prediction and Prediction Error in the Primate Brain. *Neuron*, *0*(0). https://doi.org/10.1016/j.neuron.2018.10.0

Chase, W. G., & Ericsson, K. A. (1982). Skill and Working Memory. In G. H. Bower (Ed.), *Psychology of Learning and Motivation* (Vol. 16, pp. 1–58). Academic Press. https://doi.org/10.1016/S0079-7421(08)60546-0

Chater, N., & Vitányi, P. (2003). Simplicity: A unifying principle in cognitive science? *Trends in*

Cognitive Sciences, *7*(1), 19–22. https://doi.org/10.1016/S1364-6613(02)00005-0

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

Conway, C. M., & Christiansen, M. H. (2001). Sequential learning in non-human primates. *Trends in Cognitive Sciences*, *5*(12), 539–546. https://doi.org/10.1016/S1364-6613(00)01800-3

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. https://doi.org/10.1037/0278-7393.31.1.24

Cowan, N. (2001). The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and Brain Sciences*, *24*(1), 87–114. https://doi.org/10.1017/S0140525X01003922

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

Cowan, N., Chen, Z., & Rouder, J. N. (2004). Constant capacity in an immediate serial-recall task: A logical sequel to Miller (1956). *Psychological Science*, *15*(9), 634–640. https://doi.org/10.1111/j.0956-7976.2004.00732.x

Dehaene, S., Meyniel, F., Wacongne, C., Wang, L., & Pallier, C. (2015). The Neural Representation of Sequences: From Transition Probabilities to Algebraic Patterns and Linguistic Trees. *Neuron*, *88*(1), 2–19. https://doi.org/10.1016/j.neuron.2015.09.019

Delahaye, J.-P., & Zenil, H. (2012). Numerical evaluation of algorithmic complexity for short strings: A glance into the innermost structure of randomness. *Applied Mathematics and Computation*, *219*(1), 63–77. https://doi.org/10.1016/j.amc.2011.10.006

Endress, A. D., Dehaene-Lambertz, G., & Mehler, J. (2007). Perceptual constraints and the learnability of simple grammars. *Cognition*, *105*(3), 577–614. https://doi.org/10.1016/j.cognition.2006.12.014

Ericsson, K. A., Chase, W. G., & Faloon, S. (1980). Acquisition of a memory skill. *Science (New York, N.Y.)*, *208*(4448), 1181–1182. https://doi.org/10.1126/science.7375930

Falk, R., & Konold, C. (1997). Making sense of randomness: Implicit encoding as a basis for judgment. *Psychological Review*, *104*(2), 301–318. https://doi.org/10.1037/0033-295X.104.2.301

Faugeras, F., Rohaut, B., Weiss, N., Bekinschtein, T. A., Galanaud, D., Puybasset, L., Bolgert, F., Sergent, C., Cohen, L., Dehaene, S., & Naccache, L. (2011). Probing consciousness with event-related potentials in the vegetative state. *Neurology*, *77*(3), 264–268. https://doi.org/10.1212/WNL.0b013e3182217ee8

Feldman, H., & Friston, K. (2010). Attention, Uncertainty, and Free-Energy. *Frontiers in Human Neuroscience*, *4*. https://doi.org/10.3389/fnhum.2010.00215

Feldman, J. (2000). Minimization of Boolean complexity in human concept learning. *Nature*, *407*(6804), 630–633. https://doi.org/10.1038/35036586

Feldman, J. (2003). The Simplicity Principle in Human Concept Learning. *Current Directions in Psychological Science*, *12*(6), 227–232. https://doi.org/10.1046/j.0963-7214.2003.01267.x

Fitch, W. T. (2014). Toward a computational framework for cognitive biology: Unifying approaches from cognitive neuroscience and comparative cognition. *Physics of Life Reviews*, *11*(3), 329–364. https://doi.org/10.1016/j.plre

Fitch, W. T., & Hauser, M. D. (2004). Computational constraints on syntactic processing in a nonhuman primate. *Science (New York, N.Y.)*, *303*(5656), 377–380. https://doi.org/10.1126/science.1089401

Fodor, J. A. (1975). *The language of thought* (Vol. 5). Harvard university press.

Freides, D. (1974). Human information processing and sensory modality: Cross-modal functions, information complexity, memory, and deficit. *Psychological Bulletin*, *81*(5), 284.

Friston, K. (2010). The free-energy principle: A unified brain theory? *Nature Reviews Neuroscience*, *11*(2), 127–138. https://doi.org/10.1038/nrn2787

Fujii, N. (2003). Representation of Action Sequence Boundaries by Macaque Prefrontal Cortical Neurons. *Science*, *301*(5637), 1246–1249. https://doi.org/10.1126/science.1086872

Garrido, M. I., Kilner, J. M., Stephan, K. E., & Friston, K. J. (2009). The mismatch negativity: A review

of underlying mechanisms. *Clinical Neurophysiology*, *120*(3), 453–463. https://doi.org/10.1016/j.clinph.2008.11.029

Garrido, M. I., Sahani, M., & Dolan, R. J. (2013). Outlier responses reflect sensitivity to statistical structure in the human brain. *PLoS Computational Biology*, *9*(3), e1002999. https://doi.org/10.1371/journal.pcbi.1002999

Gauvrit, N., Singmann, H., Soler-Toscano, F., & Zenil, H. (2016). Algorithmic complexity for psychology: A user-friendly implementation of the coding theorem method. *Behavior Research Methods*, *48*(1), 314–329. https://doi.org/10.3758/s13428-015-0574-3

Gauvrit, N., Zenil, H., Delahaye, J.-P., & Soler-Toscano, F. (2014). Algorithmic complexity for short binary strings applied to psychology: A primer. *Behavior Research Methods*, *46*(3), 732–744. https://doi.org/10.3758/s13428
013-0416-0

Gervain, J., Macagno, F., Cogoi, S., Peña, M., & Mehler, J. (2008). The neonate brain detects speech structure. *Proceedings of the National Academy of Sciences*, *105*(37), 14222–14227. https://doi.org/10.1073/pnas.0806530

Gilchrist, A. L., Cowan, N., & Naveh-Benjamin, M. (2008). Working Memory Capacity for Spoken Sentences Decreases with Adult Aging: Recall of Fewer, but not Smaller Chunks in Older Adults. *Memory (Hove, England)*, *16*(7), 773–787. https://doi.org/10.1080/09658210802261124

Gil-da-Costa, R., Stoner, G. R., Fung, R., & Albright, T. D. (2013). Nonhuman primate model of schizophrenia using a noninvasive EEG method. *Proceedings of the National Academy of Sciences*, *110*(38), 15425–15430. https://doi.org/10.1073/pnas.1312264110

Glanzer, M., & Clark, W. H. (1963). Accuracy of perceptual recall: An analysis of organization. *Journal of Verbal Learning and Verbal Behavior*, *1*(4), 289–299. https://doi.org/10.1016/S0022-5371(63)80008-0

Glenberg, A. M., Mann, S., Altman, L., Forman, T., & Procise, S. (1989). Modality effects in the coding reproduction of rhythms. *Memory & Cognition*, *17*(4), 373–383. https://doi.org/10.3758/BF03202611

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

Goodman, N. D., Tenenbaum, J. B., Feldman, J., & Griffiths, T. L. (2008). A Rational Analysis of Rule-Based Concept Learning. *Cognitive Science*, *32*(1), 108–154. https://doi.org/10.1080/03640210701802071

Goodman, N. D., Ullman, T. D., & Tenenbaum, J. B. (2011). Learning a theory of causality. *Psychological Review*, *118*(1), 110–119. https://doi.org/10.1037/a0021336

Griffiths, T. L., Daniels, D., Austerweil, J. L., & Tenenbaum, J. B. (2018). Subjective randomness as statistical inference. *Cognitive Psychology*, *103*, 85–109. https://doi.org/10.1016/j.cogpsych.2018.02.003

Griffiths, T. L., & Tenenbaum, J. B. (2003). Probability, algorithmic complexity, and subjective randomness. *Proceedings of the Annual Meeting of the Cognitive Science Society*, *25*.

Guttman, S. E., Gilroy, L. A., & Blake, R. (2005). Hearing what the eyes see: Auditory encoding of visual temporal sequences. *Psychological Science*, *16*(3), 228–235. https://doi.org/10.1111/j.0956-7976.2005.00808.x

Hauser, M. D., Chomsky, N., & Fitch, W. T. (2002). The Faculty of Language: What Is It, Who Has It, and How Did It Evolve? *Science*, *298*(5598), 1569–1579. https://doi.org/10.1126/science.298.5598.1569

Hauser, M. D., Newport, E. L., & Aslin, R. N. (2001). Segmentation of the speech stream in a non-human primate: Statistical learning in cotton-top tamarins. *Cognition*, *78*(3), B53–B64. https://doi.org/10.1016/S0010-0277(00)00132-3

Hauser, M. D., & Watumull, J. (2017). The Universal Generative Faculty: The source of our expressive power in language, mathematics, morality, and music. *Journal of Neurolinguistics*. https://doi.org/10.1016/j.jneuroling.20

Heilbron, M., & Chait, M. (2018). Great Expectations: Is there Evidence for Predictive Coding in Auditory Cortex? *Neuroscience*, *389*, 54–73. https://doi.org/10.1016/j.neuroscience.2017.07.061

Huettel, S. A., Mack, P. B., & McCarthy, G. (2002). Perceiving patterns in random series: Dynamic processing of sequence in prefrontal cortex. *Nature Neuroscience*, *5*(5), 485–490. https://doi.org/10.1038/nn841

Jiang, X., Long, T., Cao, W., Li, J., Dehaene, S., & Wang, L. (2018). Production of Supra-regular Spatial Sequences by Macaque Monkeys. *Current Biology: CB*, *28*(12), 1851-1859.e4. https://doi.org/10.1016/j.cub.2018.04.047

Kenward, M. G., & Roger, J. H. (1997). Small sample inference for fixed effects from restricted maximum likelihood. *Biometrics*, *53*(3), 983–997.

Kirkham, N. Z., Slemmer, J. A., & Johnson, S. P. (2002). Visual statistical learning in infancy: Evidence for a domain general learning mechanism. *Cognition*, *83*(2), B35–B42. https://doi.org/10.1016/S0010-0277(02)00004-5

Kleiner, M., Brainard, D., Pelli, D., Ingling, A., Murray, R., & Broussard, C. (2007). What's new in Psychtoolbox-3. *Perception*, *36*(14), 1.

Kolmogorov, A. N. (1968). Three approaches to the quantitative definition of information. *International Journal of Computer Mathematics*, *2*(1–4), 157–168.

Kuznetsova, A., Brockhoff, P. B., & Christensen, R. H. (2017). lmerTest package: Tests in linear mixed effects models. *Journal of Statistical Software*, *82*(13), 1–26.

Lashley, K. S. (1951). The problem of serial order in behavior. In L. A. Jeffress & L. A. (Ed) Jeffress (Eds.), *Cerebral mechanisms in behavior; the Hixon Symposium.* (1952-04498-003; pp. 112–146). Wiley.

Leeuwenberg, E. L. (1969). Quantitative specification of information in sequential patterns. *Psychological Review*, *76*(2), 216–220. https://doi.org/10.1037/h0027285

Leeuwenberg, E. L. J. (1971). A Perceptual Coding Language for Visual and Auditory Patterns. *The American Journal of Psychology*, *84*(3), 307–349. JSTOR. https://doi.org/10.2307/1420464

Lempel, A., & Ziv, J. (1976). On the Complexity of Finite Sequences. *IEEE Transactions on Information Theory*, *22*(1), 75–81. https://doi.org/10.1109/TIT.1976.1055501

Luck, S. J., & Vogel, E. K. (1997). The capacity of visual working memory for features and conjunctions. *Nature*, *390*(6657), 279–281. https://doi.org/10.1038/36846

MacGregor, J. (1987). Short-Term Memory Capacity: Limitation or Optimization? *Psychological Review*, *94*(1), 107–108.

Maheu, M., Dehaene, S., & Meyniel, F. (2019). Brain signatures of a multiscale process of sequence learning in humans. *ELife*, *8*. https://doi.org/10.7554/eLife.41541

Maheu, M., Meyniel, F., & Dehaene, S. (2020). Rational arbitration between statistics and rules in human sequence learning. *BioRxiv*, 2020.02.06.937706. https://doi.org/10.1101/2020.02.06.937706

Marcus, G. F., Vijayan, S., Bandi Rao, S., & Vishton, P. M. (1999). Rule learning by seven-month-old infants. *Science*, *283*(5398), 77–80.

Mars, R. B., Debener, S., Gladwin, T. E., Harrison, L. M., Haggard, P., Rothwell, J. C., & Bestmann, S. (2008). Trial-by-Trial Fluctuations in the Event-Related Electroencephalogram Reflect Dynamic Changes in the Degree of Surprise. *Journal of Neuroscience*, *28*(47), 12539–12545. https://doi.org/10.1523/JNEUROSCI.2925-08.2008

Mathy, F., & Feldman, J. (2012). What's magic about magic numbers? Chunking and data compression in short-term memory. *Cognition*, *122*(3), 346–362. https://doi.org/10.1016/j.cognition.2011.11.003

McAuley, J. D., & Henry, M. J. (2010). Modality effects in rhythm processing: Auditory encoding of visual rhythms is neither obligatory nor automatic. *Attention, Perception, & Psychophysics*, *72*(5), 1377–1389. https://doi.org/10.3758/APP.72.5.1377

McDermott, J. H., Schemitsch, M., & Simoncelli, E. P. (2013). Summary statistics in auditory perception. *Nature Neuroscience*, *16*(4), 493–498. https://doi.org/10.1038/nn.3347

Meyer, T., & Olson, C. R. (2011). Statistical learning of visual transitions in monkey inferotemporal cortex. *Proceedings of the National Academy of Sciences*, *108*(48), 19401–19406. https://doi.org/10.1073/pnas.1112895108

Meyniel, F., & Dehaene, S. (2017). Brain networks for confidence weighting and hierarchical inference during probabilistic learning. *Proceedings of the National Academy of Sciences*, *114*(19), E3859–E3868. https://doi.org/10.1073/pnas.1615773114

Meyniel, F., Maheu, M., & Dehaene, S. (2016). Human Inferences about Sequences: A Minimal Transition Probability Model. *PLOS Computational Biology*, *12*(12), e1005260. https://doi.org/10.1371/journal.pcbi.1005260

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for

processing information. *Psychological Review*, *63*(2), 81–97. https://doi.org/10.1037/h0043158

Näätänen, R. (2003). Mismatch negativity: Clinical research and possible applications. *International Journal of Psychophysiology*, *48*(2), 179–188. https://doi.org/10.1016/S0167-8760(03)00053-9

Oskarsson, A. T., Van Boven, L., McClelland, G. H., & Hastie, R. (2009). What's next? Judging sequences of binary events. *Psychological Bulletin*, *135*(2), 262–285. https://doi.org/10.1037/a0014821

Paavilainen, P. (2013). The mismatch-negativity (MMN) component of the auditory event-related potential to violations of abstract regularities: A review. *International Journal of Psychophysiology: Official Journal of the International Organization of Psychophysiology*, *88*(2), 109–123. https://doi.org/10.1016/j.ijpsycho.2013.03

Patel, A. D., Iversen, J. R., Chen, Y., & Repp, B. H. (2005). The influence of metricality and modality on synchronization with a beat. *Experimental Brain Research*, *163*(2), 226–238. https://doi.org/10.1007/s00221-004-2159-8

Peng, Z., Genewein, T., & Braun, D. A. (2014). Assessing randomness and complexity in human motion trajectories through analysis of symbolic sequences. *Frontiers in Human Neuroscience*, *8*. https://doi.org/10.3389/fnhum.2

Piantadosi, S. T., & Jacobs, R. A. (2016). Four Problems Solved by the Probabilistic Language of Thought. *Current Directions in Psychological Science*, *25*(1), 54–59. https://doi.org/10.1177/0963721415609581

Piantadosi, S. T., Tenenbaum, J. B., & Goodman, N. D. (2012). Bootstrapping in a language of thought: A formal model of numerical concept learning. *Cognition*, *123*(2), 199–217. https://doi.org/10.1016/j.cognition.2011.11.00

Piantadosi, S. T., Tenenbaum, J. B., & Goodman, N. D. (2016). The logical primitives of thought: Empirical foundations for compositional cognitive models. *Psychological Review*, *123*(4), 392–424. https://doi.org/10.1037/a0039

Psotka, J. (1975). Simplicity, symmetry, and syntely: Stimulus measures of binary pattern structure. *Memory & Cognition*, *3*(4), 434–444. https://doi.org/10.3758/BF03212938

R Core Team. (2017). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/

Rao, R. P. N. (2005). Bayesian inference and attentional modulation in the visual cortex. *NeuroReport*, *16*(16), 1843. https://doi.org/10.1097/01.wnr.0000183900.92901.fc

Restle, F. (1970). Theory of serial pattern learning: Structural trees. *Psychological Review*, *77*(6), 481–495. https://doi.org/10.1037/h0029964

Restle, F. (1973). Serial pattern learning: Higher order transitions. *Journal of Experimental Psychology*, *99*(1), 61–69. https://doi.org/10.1037/h0034751

Restle, F., & Brown, E. R. (1970). Serial Pattern Learning. *Journal of Experimental Psychology*, *83*(1, Pt.1), 120–125. https://doi.org/10.1037/h0028530

Romano, S., Salles, A., Amalric, M., Dehaene, S., Sigman, M., & Figueira, S. (2018). Bayesian validation of grammar productions for the language of thought. *PloS One*, *13*(7), e0200420. https://doi.org/10.1371/journal.pone.020

Romano, S., Sigman, M., & Figueira, S. (2013). $ LTˆ 2Cˆ 2$: A language of thought with Turing-computable Kolmogorov complexity. *Papers in Physics*, *5*, 050001–050001. https://doi.org/10.4279/pip.050001

Romberg, A. R., & Saffran, J. R. (2010). Statistical learning and language acquisition. *Wiley Interdisciplinary Reviews: Cognitive Science*, *1*(6), 906–914. https://doi.org/10.1002/wcs.78

Saffran, J. R., Aslin, R. N., & Newport, E. L. (1996). Statistical Learning by 8-Month-Old Infants. *Science*, *274*(5294), 1926–1928. https://doi.org/10.1126/science.274.5294.1926

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

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, *27*(3), 379–423.

Simon, H. A. (1972). Complexity and the representation of patterned sequences of symbols. *Psychological Review*, *79*(5), 369–382. https://doi.org/10.1037/h0033118

Simon, H. A., & Kotovsky, K. (1963). Human acquisition of concepts for sequential patterns. *Psychological Review*, *70*(6), 534–546. https://doi.org/10.1037/h0043901

Siskind, J. M. (1996). A computational study of cross-situational techniques for learning word-to-meaning mappings. *Cognition*, *61*(1), 39–91. https://doi.org/10.1016/S0010-0277(96)00728-7

Soler-Toscano, F., Zenil, H., Delahaye, J.-P., & Gauvrit, N. (2014). Calculating Kolmogorov Complexity from the Output Frequency Distributions of Small Turing Machines. *PLoS ONE*, *9*(5), e96223. https://doi.org/10.1371/journal.pone.0096223

Solomonoff, R. J. (1964). A formal theory of inductive inference. Part I and Part II. *Information and Control*, *7*(1), 1–22. https://doi.org/10.1016/S0019-9958(64)90223-2

Southwell, R., Baumann, A., Gal, C., Barascud, N., Friston, K., & Chait, M. (2017). Is predictability salient? A study of attentional capture by auditory patterns. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, *372*(1714). https://doi.org/10.1098/rstb.2016.0105

Southwell, Rosy, & Chait, M. (2018). Enhanced deviant responses in patterned relative to random sound sequences. *Cortex; a Journal Devoted to the Study of the Nervous System and Behavior*, *109*, 92–103. https://doi.org/10.1016/j.cortex.2018.08.032

Squires, N. K., Squires, K. C., & Hillyard, S. A. (1975). Two varieties of long-latency positive waves evoked by unpredictable auditory stimuli in man. *Electroencephalography and Clinical Neurophysiology*, *38*(4), 387–401. https://doi.org/10.1016/0013-4694(75)90263-1

Strange, B. A., Duggins, A., Penny, W., Dolan, R. J., & Friston, K. J. (2005). Information theory, novelty and hippocampal responses: Unpredicted or unpredictable? *Neural Networks*, *18*(3), 225–230. https://doi.org/10.1016/j.neunet.2004.12.004

Strauss, M., Sitt, J. D., King, J.-R., Elbaz, M., Azizi, L., Buiatti, M., Naccache, L., van Wassenhove, V., & Dehaene, S. (2015). Disruption of hierarchical predictive coding during sleep. *Proceedings of the National Academy of Sciences*, *112*(11), E1353–E1362. https://doi.org/10.1073/pnas.1501026112

Thul, E., & Toussaint, G. T. (2008). Rhythm Complexity Measures: A Comparison of Mathematical Models of Human Perception and Performance. *Rhythm and Meter*, 6.

Toussaint, G. T., & Beltran, J. F. (2013). Subsymmetries predict auditory and visual pattern complexity. *Perception*, *42*(10), 1095–1100. https://doi.org/10.1068/p7614

Uhrig, L., Dehaene, S., & Jarraya, B. (2014). A Hierarchy of Responses to Auditory Regularities in the Macaque Brain. *Journal of Neuroscience*, *34*(4), 1127–1132. https://doi.org/10.1523/JNEUROSCI.3165-13.2014

Vandierendonck, A. (2017). A comparison of methods to combine speed and accuracy measures of performance: A rejoinder on the binning procedure. *Behavior Research Methods*, *49*(2), 653–673. https://doi.org/10.3758/s13428-016-0721-5

Vandierendonck, A. (2018). Further Tests of the Utility of Integrated Speed-Accuracy Measures in Task Switching. *Journal of Cognition*, *1*(1). https://doi.org/10.5334/joc.6

Vitz, P. C. (1968). Information, run structure and binary pattern complexity. *Perception & Psychophysics*, *3*(4), 275–280. https://doi.org/10.3758/BF03212743

Vitz, P. C. (2019). A hierarchical model of binary pattern learning. *Learning and Motivation*, *65*, 52–59. https://doi.org/10.1016/j.lmot.2019.01.002

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

Wacongne, C., Labyt, E., Wassenhove, V. van, Bekinschtein, T., Naccache, L., & Dehaene, S. (2011). Evidence for a hierarchy of predictions and prediction errors in human cortex. *Proceedings of the National Academy of Sciences*, *108*(51), 20754–20759. https://doi.org/10.1073/pnas.1117807108

Wagenmakers, E.-J., & Farrell, S. (2004). AIC model selection using Akaike weights. *Psychonomic Bulletin & Review*, *11*(1), 192–196. https://doi.org/10.3758/BF03206482

Wang, L., Amalric, M., Fang, W., Jiang, X., Pallier, C., Figueira, S., Sigman, M., & Dehaene, S. (2019). Representation of spatial sequences using nested rules in human prefrontal cortex. *NeuroImage*, *186*, 245–255. https://doi.org/10.1016/j.neuroimage.2018.10.061

Wang, L., Uhrig, L., Jarraya, B., & Dehaene, S. (2015). Representation of Numerical and Sequential Pat-

terns in Macaque and Human Brains. *Current Biology*, *25*(15), 1966–1974. https://doi.org/10.1016/j.cub.2015.06.035

Wilson, B., Marslen-Wilson, W. D., & Petkov, C. I. (2017). Conserved Sequence Processing in Primate Frontal Cortex. *Trends in Neurosciences*, *40*(2), 72–82. https://doi.org/10.1016/j.tins.2016.11.004

Wilson, B., Slater, H., Kikuchi, Y., Milne, A. E., Marslen-Wilson, W. D., Smith, K., & Petkov, C. I. (2013). Auditory Artificial Grammar Learning in Macaque and Marmoset Monkeys. *Journal of Neuroscience*, *33*(48), 18825–18835. https://doi.org/10.1523/JNEUROSCI.2414-13.2013

Yildirim, I., & Jacobs, R. A. (2015). Learning multisensory representations for auditory-visual transfer of sequence category knowledge: A probabilistic language of thought approach. *Psychonomic Bulletin & Review*, *22*(3), 673–686. https://doi.org/10.3758/s13423-014-0734-y

Bibliografía

[Abelson et al., 1974] Abelson, H., Goodman, N., and Rudolph, L. (1974). *Logo manual*. -.

[Amalric et al., 2017a] Amalric, M., Wang, L., Pica, P., Figueira, S., Sigman, M., and Dehaene, S. (2017a). The language of geometry: Fast comprehension of geometrical primitives and rules in human adults and preschoolers. *PLOS Computational Biology*, 13(1):e1005273.

[Amalric et al., 2017b] Amalric, M., Wang, L., Pica, P., Figueira, S., Sigman, M., and Dehaene, S. (2017b). The language of geometry: Fast comprehension of geometrical primitives and rules in human adults and preschoolers. *PLoS computational biology*, 13(1):e1005273.

[Arcediano et al., 1997] Arcediano, F., Matute, H., and Miller, R. R. (1997). Blocking of pavlovian conditioning in humans. *Learning and Motivation*, 28(2):188–199.

[Ashby and Maddox, 2005] Ashby, F. G. and Maddox, W. T. (2005). Human category learning. *Annu. Rev. Psychol.*, 56:149–178.

[Ashby and Maddox, 2011] Ashby, F. G. and Maddox, W. T. (2011). Human category learning 2.0. *Annals of the New York Academy of Sciences*, 1224:147.

[Aydede, 1997] Aydede, M. (1997). Language of thought: The connectionist contribution. *Minds and Machines*, 7(1):57–101.

[Bengio et al., 2013] Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828.

[Blackburn, 1984] Blackburn, S. (1984). Spreading the word: Grounding in the philosophy of language.

[Blair and Homa, 2003] Blair, M. and Homa, D. (2003). As easy to memorize as they are to classify: The 5–4 categories and the category advantage. *Memory & Cognition*, 31(8):1293–1301.

[Blair et al., 2009] Blair, M. R., Watson, M. R., Walshe, R. C., and Maj, F. (2009). Extremely selective attention: Eye-tracking studies of the dynamic allocation of attention to stimulus features in categorization. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 35(5):1196.

[Boole, 1854] Boole, G. (1854). *An investigation of the laws of thought: on which are founded the mathematical theories of logic and probabilities*. Dover Publications.

[Borges, 1944] Borges, J. L. (1944). *Ficciones, 1935-1944*. Buenos Aires: Sur.

[Bourne, 1970] Bourne, L. E. (1970). Knowing and using concepts. *Psychological Review*, 77(6):546.

[Buhrmester et al., 2011] Buhrmester, M., Kwang, T., and Gosling, S. D. (2011). Amazon's Mechanical Turk: A new source of inexpensive, yet high-quality, data? *Perspectives on psychological science*, 6(1):3–5.

[Calude et al., 2002] Calude, C. S., Dinneen, M. J., Shu, C.-K., et al. (2002). Computing a glimpse of randomness. *Experimental Mathematics*, 11(3):361–370.

[Carvalho and Goldstone, 2014] Carvalho, P. F. and Goldstone, R. L. (2014). Putting category learning in order: Category structure and temporal arrangement affect the benefit of interleaved over blocked study. *Memory & cognition*, 42(3):481–495.

[Chapman and Robbins, 1990] Chapman, G. B. and Robbins, S. J. (1990). Cue interaction in human contingency judgment. *Memory & Cognition*, 18(5):537–545.

[Chomsky, 1986] Chomsky, N. (1986). *Knowledge of language: Its nature, origin, and use*. Greenwood Publishing Group.

[Chomsky et al., 2006] Chomsky, N. et al. (2006). On cognitive structures and their development: A reply to piaget. *Philosophy of mind: Classical problems/contemporary issues*, pages 751–755.

[Cohen and Lefebvre, 2005] Cohen, H. and Lefebvre, C. (2005). *Handbook of categorization in cognitive science*. Elsevier.

[Crump et al., 2013] Crump, M. J., McDonnell, J. V., and Gureckis, T. M. (2013). Evaluating Amazon's Mechanical Turk as a tool for experimental behavioral research. *PloS one*, 8(3):e57410.

[Dehaene et al., 2006] Dehaene, S., Izard, V., Pica, P., and Spelke, E. (2006). Core knowledge of geometry in an amazonian indigene group. *Science*, 311(5759):381–384.

[Denison et al., 2013] Denison, S., Bonawitz, E., Gopnik, A., and Griffiths, T. L. (2013). Rational variability in children's causal inferences: The sampling hypothesis. *Cognition*, 126(2):285–300.

[Dillon et al., 2013] Dillon, M. R., Huang, Y., and Spelke, E. S. (2013). Core foundations of abstract geometry. *Proceedings of the National Academy of Sciences*, 110(35):14191–14195.

[Ellis et al., 2015] Ellis, K., Solar-Lezama, A., and Tenenbaum, J. (2015). Unsupervised learning by program synthesis. In *Advances in Neural Information Processing Systems*, pages 973–981.

[Feldman, 2000] Feldman, J. (2000). Minimization of boolean complexity in human concept learning. *Nature*, 407(6804):630–633.

[Feldman, 2003] Feldman, J. (2003). The simplicity principle in human concept learning. *Current Directions in Psychological Science*, 12(6):227–232.

[Fodor, 1975] Fodor, J. (1975). *The Language of Thought*. Language and thought series. Harvard University Press.

[Geman and Geman, 1984] Geman, S. and Geman, D. (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6):721–741.

[Gentner, 1983] Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive science*, 7(2):155–170.

[Gershman et al., 2015] Gershman, S. J., Horvitz, E. J., and Tenenbaum, J. B. (2015). Computational rationality: A converging paradigm for intelligence in brains, minds, and machines. *Science*, 349(6245):273–278.

[Ghahramani, 2015] Ghahramani, Z. (2015). Probabilistic machine learning and artificial intelligence. *Nature*, 521(7553):452.

[Goldsmith, 2001] Goldsmith, J. (2001). Unsupervised learning of the morphology of a natural language. *Computational linguistics*, 27(2):153–198.

[Goldsmith, 2002] Goldsmith, J. (2002). Probabilistic models of grammar: Phonology as information minimization. *Phonological Studies*, 5:21–46.

[Goodman et al., 2008] Goodman, N. D., Tenenbaum, J. B., Feldman, J., and Griffiths, T. L. (2008). A rational analysis of rule-based concept learning. *Cognitive Science*, 32(1):108–154.

[Grünwald and Grunwald, 2007] Grünwald, P. D. and Grunwald, A. (2007). *The minimum description length principle*. MIT press.

[Hoffman and Rehder, 2010] Hoffman, A. B. and Rehder, B. (2010). The costs of supervised classification: The effect of learning task on conceptual flexibility. *Journal of Experimental Psychology: General*, 139(2):319.

[Izard et al., 2011] Izard, V., Pica, P., Dehaene, S., Hinchey, D., and Spelke, E. (2011). Geometry as a universal mental construction. *Space, Time and Number in the Brain*, 19:319–332.

[Johnson et al., 2007] Johnson, M., Griffiths, T. L., and Goldwater, S. (2007). Bayesian inference for pcfgs via markov chain monte carlo. In *HLT-NAACL*, pages 139–146.

[Kemp, 2012] Kemp, C. (2012). Exploring the conceptual universe. *Psychological review*, 119(4):685.

[Kim and Rehder, 2011] Kim, S. and Rehder, B. (2011). How prior knowledge affects selective attention during category learning: An eyetracking study. *Memory & cognition*, 39(4):649–665.

[Knowles, 1998] Knowles, J. (1998). The language of thought and natural language understanding. *Analysis*, 58(4):264–272.

[Kolmogorov, 1968] Kolmogorov, A. N. (1968). Three approaches to the quantitative definition of information*. *International Journal of Computer Mathematics*, 2(1-4):157–168.

[Kruschke, 1996] Kruschke, J. K. (1996). Dimensional relevance shifts in category learning. *Connection Science*, 8(2):225–248.

[Kruschke and Blair, 2000] Kruschke, J. K. and Blair, N. J. (2000). Blocking and backward blocking involve learned inattention. *Psychonomic Bulletin & Review*, 7(4):636–645.

[Kruschke et al., 2005] Kruschke, J. K., Kappenman, E. S., and Hetrick, W. P. (2005). Eye gaze and individual differences consistent with learned attention in associative blocking and highlighting. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 31(5):830.

[Lake et al., 2015] Lake, B. M., Salakhutdinov, R., and Tenenbaum, J. B. (2015). Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338.

[Lake et al., 2017] Lake, B. M., Ullman, T. D., Tenenbaum, J. B., and Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40.

[Landau et al., 1981] Landau, B., Gleitman, H., and Spelke, E. (1981). Spatial knowledge and geometric representation in a child blind from birth. *Science*, 213(4513):1275–1278.

[LeCun et al., 2015] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436.

[Lee et al., 2012] Lee, S. A., Sovrano, V. A., and Spelke, E. S. (2012). Navigation as a source of geometric knowledge: Young children's use of length, angle, distance, and direction in a reorientation task. *Cognition*, 123(1):144–161.

[Leeuwenberg, 1971] Leeuwenberg, E. L. (1971). A perceptual coding language for visual and auditory patterns. *The American Journal of Psychology*, pages 307–349.

[Levin, 1974] Levin, L. A. (1974). Laws of information conservation (nongrowth) and aspects of the foundation of probability theory. *Problemy Peredachi Informatsii*, 10(3):30–35.

[Lewandowsky, 2011] Lewandowsky, S. (2011). Working memory capacity and categorization: individual differences and modeling. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 37(3):720.

[Li and Vitányi, 2013] Li, M. and Vitányi, P. (2013). *An introduction to Kolmogorov complexity and its applications*. Springer Science & Business Media.

[Loewer and Rey, 1991] Loewer, B. and Rey, G. (1991). Meaning in mind. *Fodor and his Critics*.

[Luchins, 1942] Luchins, A. S. (1942). Mechanization in problem solving: The effect of einstellung. *Psychological monographs*, 54(6):i.

[Lupyan et al., 2007] Lupyan, G., Rakison, D. H., and McClelland, J. L. (2007). Language is not just for talking: Redundant labels facilitate learning of novel categories. *Psychological science*, 18(12):1077–1083.

[Machilsen et al., 2009] Machilsen, B., Pauwels, M., and Wagemans, J. (2009). The role of vertical mirror symmetry in visual shape detection. *Journal of Vision*, 9(12):11–11.

[MacKay, 2003] MacKay, D. J. (2003). *Information theory, inference and learning algorithms*. Cambridge university press.

[Mackintosh, 1975] Mackintosh, N. J. (1975). A theory of attention: Variations in the associability of stimuli with reinforcement. *Psychological review*, 82(4):276.

[Maddox and Ashby, 1993] Maddox, W. T. and Ashby, F. G. (1993). Comparing decision bound and exemplar models of categorization. *Perception & psychophysics*, 53(1):49–70.

[Manning and Schütze, 1999] Manning, C. and Schütze, H. (1999). *Foundations of Statistical Natural Language Processing*. MIT Press.

[Minda and Smith, 2001] Minda, J. P. and Smith, J. D. (2001). Prototypes in category learning: the effects of category size, category structure, and stimulus complexity. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 27(3):775.

[Minsky, 1967] Minsky, M. L. (1967). *Computation: finite and infinite machines*. Prentice-Hall, Inc.

[Murphy, 1988] Murphy, G. L. (1988). Comprehending complex concepts. *Cognitive science*, 12(4):529–562.

[Newell, 1980] Newell, A. (1980). Physical symbol systems. *Cognitive science*, 4(2):135–183.

[Nosofsky, 1986] Nosofsky, R. M. (1986). Attention, similarity, and the identification–categorization relationship. *Journal of experimental psychology: General*, 115(1):39.

[Nosofsky et al., 1994a] Nosofsky, R. M., Gluck, M. A., Palmeri, T. J., McKinley, S. C., and Glauthier, P. (1994a). Comparing modes of rule-based classification learning: A replication and extension of shepard, hovland, and jenkins (1961). *Memory & cognition*, 22(3):352–369.

[Nosofsky et al., 1994b] Nosofsky, R. M., Palmeri, T. J., and McKinley, S. C. (1994b). Rule-plus-exception model of classification learning. *Psychological review*, 101(1):53.

[Overlan et al., 2017] Overlan, M. C., Jacobs, R. A., and Piantadosi, S. T. (2017). Learning abstract visual concepts via probabilistic program induction in a language of thought. *Cognition*, 168:320–334.

[Piantadosi and Jacobs, 2016] Piantadosi, S. T. and Jacobs, R. A. (2016). Four problems solved by the probabilistic language of thought. *Current Directions in Psychological Science*, 25(1):54–59.

[Piantadosi et al., 2012] Piantadosi, S. T., Tenenbaum, J. B., and Goodman, N. D. (2012). Bootstrapping in a language of thought: A formal model of numerical concept learning. *Cognition*, 123(2):199–217.

[Piantadosi et al., 2016] Piantadosi, S. T., Tenenbaum, J. B., and Goodman, N. D. (2016). The logical primitives of thought: Empirical foundations for compositional cognitive models. *Psychological review*, 123(4):392.

[Rehder and Hoffman, 2005] Rehder, B. and Hoffman, A. B. (2005). Eyetracking and selective attention in category learning. *Cognitive psychology*, 51(1):1–41.

[Rescorla and Wagner, 1972] Rescorla, R. A. and Wagner, A. R. (1972). A theory of Pavlovian conditioning: Variations on the effectiveness of reinforcement and non-reinforcement. In Black, A. H. and Prokasy, W. F., editors, *Classical conditioning II: Current research and theory*, pages 64–99. Appleton-Century-Crofts, New York.

[Romano et al., 2018] Romano, S., Salles, A., Amalric, M., Dehaene, S., Sigman, M., and Figueira, S. (2018). Bayesian validation of grammar productions for the language of thought. *PLOS ONE*, 13(7):1–20.

[Romano et al., 2013] Romano, S., Sigman, M., and Figueira, S. (2013). : A language of thought with turing-computable kolmogorov complexity. *Papers in Physics*, 5:050001.

[Rosch, 1999] Rosch, E. (1999). Principles of categorization. *Concepts: core readings*, 189.

[Rosch and Mervis, 1975] Rosch, E. and Mervis, C. B. (1975). Family resemblances: Studies in the internal structure of categories. *Cognitive psychology*, 7(4):573–605.

[Rosch et al., 1976] Rosch, E., Simpson, C., and Miller, R. S. (1976). Structural bases of typicality effects. *Journal of Experimental Psychology: Human perception and performance*, 2(4):491.

[Russell and Norvig, 2002] Russell, S. and Norvig, P. (2002). Artificial intelligence: a modern approach.

[Schyns et al., 1998] Schyns, P. G., Goldstone, R. L., and Thibaut, J.-P. (1998). The development of features in object concepts. *Behavioral and brain Sciences*, 21(1):1–17.

[Shannon, 1948] Shannon, C. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656.

[Shepard et al., 1961] Shepard, R. N., Hovland, C. I., and Jenkins, H. M. (1961). Learning and memorization of classifications. *Psychological monographs: General and applied*, 75(13):1.

[Solomonoff, 1964] Solomonoff, R. J. (1964). A formal theory of inductive inference. part i. *Information and control*, 7(1):1–22.

[Stewart et al., 2015] Stewart, N., Ungemach, C., Harris, A. J., Bartels, D. M., Newell, B. R., Paolacci, G., Chandler, J., et al. (2015). The average laboratory samples a population of 7,300 Amazon Mechanical Turk workers. *Judgment and Decision making*, 10(5):479–491.

[Tano et al., 2020] Tano, P., Romano, S., Sigman, M., Salles, A., and Figueira, S. (2020). Towards a more flexible language of thought: Bayesian grammar updates after each concept exposure. *Phys. Rev. E*, 101:042128.

[Tenenbaum et al., 2011] Tenenbaum, J. B., Kemp, C., Griffiths, T. L., and Goodman, N. D. (2011). How to grow a mind: Statistics, structure, and abstraction. *science*, 331(6022):1279–1285.

[Ullman et al., 2012] Ullman, T. D., Goodman, N. D., and Tenenbaum, J. B. (2012). Theory learning as stochastic search in the language of thought. *Cognitive Development*, 27(4):455–480.

[Wagner, 1970] Wagner, A. R. (1970). Stimulus selection and a "modified continuity theoryrdquo. In *Psychology of learning and motivation*, volume 3, pages 1–41. Elsevier.

[Westphal-Fitch et al., 2012] Westphal-Fitch, G., Huber, L., Gómez, J. C., and Fitch, W. T. (2012). Production and perception rules underlying visual patterns: effects of symmetry and hierarchy. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 367(1598):2007–2022.

[Yildirim and Jacobs, 2015] Yildirim, I. and Jacobs, R. A. (2015). Learning multisensory representations for auditory-visual transfer of sequence category knowledge: a probabilistic language of thought approach. *Psychonomic bulletin & review*, 22(3):673–686.