Ease of Learning Explains Semantic Universals

Shane Steinert-Threlkeld
Department of Linguistics
University of Washington
shanest@uw.edu

Jakub Szymanik
Institute for Logic, Language and Computation
Universiteit van Amsterdam
J.K.Szymanik@uva.nl

Forthcoming in Cognition.

Abstract

Semantic universals are properties of meaning shared by the languages of the world. We offer an explanation of the presence of such universals by measuring simplicity in terms of ease of learning, showing that expressions satisfying universals are simpler than those that do not according to this criterion. We measure ease of learning using tools from machine learning and analyze universals in a domain of function words (quantifiers) and content words (color terms). Our results provide strong evidence that semantic universals across both function and content words reflect simplicity as measured by ease of learning.

1 Introduction

In spite of extraordinary differences between languages, linguists have identified properties shared by all (or nearly all) languages at many levels of linguistic analysis, e.g., phonology (Hyman, 2008), syntax (Chomsky, 1965; Croft, 1990; Dixon, 1977), and semantics (Barwise and Cooper, 1981; Fintel and Matthewson, 2008; Goddard and Wierzbicka, 1994).1 Linguistic universals are crucial for understanding human cognition. Because they are attested across different languages and communicative niches, they likely reflect general features of human cognitive makeup. Explaining why a universal holds requires establishing a connection between language and a feature of the mind.

While the literature offers some explanatory theories of linguistic universals (E. Gibson et al., 2019; Hawkins, 2014), relatively few apply to semantic domains. Even fewer existing explanations are expressed in a precise mathematical modeling framework. A notable

---

1See Evans and Levinson, 2009 for an argument that most cross-linguistic regularities are not fully universal.
exception is a recent explanation in terms of optimal communication: meanings expressed by languages vary but the scope of possible variation is constrained by two complementary forces of simplicity and informativeness (Kemp and Regier, 2012; Kemp, Xu, and Regier, 2018). However, this type of explanation has a drawback: within this framework, in order to explain a given universal, one needs to come up with a complexity measure for the conceptual domain in question. Such measures will usually depend on choices (e.g. of primitives) made by the theorist about the relevant domain (Feldman, 2000; Piantadosi, Tenenbaum, and Goodman, 2016).

In this paper, we propose a unified cognitive explanation of linguistic universals by measuring complexity in terms of learnability. In particular, we suggest that linguistic universals arise because expressions satisfying them are easier to learn than those that do not. To operationalize the difficulty of learning, we have not tailor-made a computational model for our specific learning tasks. Instead, we use off-the-shelf computational tools: neural network architectures currently in widespread use in deep learning (Goodfellow, Bengio, and Courville, 2016; Lecun, Bengio, and Hinton, 2015). The ease of learning by such models provides a definition of simplicity that applies easily to many different domains.2

We focus on semantic universals in two domains: quantifiers and color terms. After presenting existing results on the former (Section 2), we present new computational results on the latter (Section 3). Our results show that universal semantic properties in both domains make meanings easier to acquire for neural networks. We have chosen these two extremely rich linguistic domains for a number of reasons. Historically, they have become a testing ground for research into the relationship between language and thought (Evans and Levinson, 2009; Sapir, 1929; Whorf, 1940). Both domains involve the interaction of various aspects of language, cognition, and perception (Berlin and Kay, 1969; Dehaene, 1997; Nieder and Dehaene, 2009). Furthermore, color terms are content words while quantifiers are function words. While recent work has drawn parallels between semantic properties in both content and function words (Chemla, Buccola, and Dautriche, 2019), our explanation of semantic universals is the first to be applied to expressions from both of these fundamental categories. A single computational explanation of the source of the semantic universals in these two domains constitutes a strong argument for the main claim of the paper that linguistic universals reflect simplicity as measured by ease of learning.

2 Quantifiers

The study of number terms (called here and in the linguistics literature quantifiers) has been one of the cornerstones of natural language semantics since its inception (Barwise and Cooper, 1981; Peters and Westerståhl, 2006). Quantifiers are the semantic objects expressed by determiners—every, some, most, the, a, three, et cetera—as they occur in syntactic configurations like [S [NP [Det N]] [VP . . .]]. Determiners express binary relations between sets of objects. For instance, the sentence Every student is happy is true if and only if each element in the set of students is in the set of happy things, i.e. if the former is a subset of the latter. Thus, Every can be taken to express the subset relation.

2We return to the relation between learnability and complexity in general in the discussion.
Only a very limited subset of all logically possible quantifiers is expressed by simple determiners in any natural language. Simple combinatorics tells us that there are 65,536 possible quantifiers to describe any situation consisting of two objects and two properties (Keenan and Stavi, 1986). Very few of those logically possible quantifiers are expressed in natural language. It turns out that there are strong regularities across languages in terms of which quantifiers are lexicalized. Thus, the domain of quantification provides a rich area of semantic universals. We now highlight two of them, showing how they are exhibited by valid inference patterns, before presenting earlier results on them.

2.1 Quantity

The first universal captures the fact that quantifiers genuinely talk about quantities. In particular, whether or not a sentence of the form Det CN VP is true or not should only depend on the sizes of the sets denoted by the common noun and verb phrase (as well as their intersection and differences). To exhibit this fact, we observe the validity of the following inference pattern.

At least three houses on Cambridge Ave are blue.

There are exactly as many blue and non-blue houses on El Camino Real as on Cambridge Ave.

Therefore, at least three houses on El Camino Real are blue.

The second premise of this inference says that the number of blue houses and of non-blue houses is equal in the two locations; the reader can verify that this pattern is valid for all choices of predicates instead of ‘blue’ and ‘house on X’. This validity shows us that at least three is what we will call quantitative. This property has been proposed as a semantic universal:

**Quantity Universal:** All simple determiners are quantitative (Benthem, 1984; Keenan and Stavi, 1986; Peters and Westerståhl, 2006).

To see what sort of expressions are ruled out, consider ‘the first three’. If we replace ‘at least three’ with ‘the first three’ uniformly in the inference pattern above, it no longer turns out valid. Whether or not the first three houses are blue depends on more than just the number of blue and non-blue houses: it depends on the order in which they are arranged on the street. This dependence on additional structure beyond set sizes is what the universal rules out. The claim, then, is that no language has a single lexical item free such that ‘free houses are blue’ means the same thing as ‘the first three houses are blue’.

2.2 Monotonicity

For the second universal, observe that the following inference pattern is valid.

Many scientists know the R programming language.

Many scientists know a programming language.
Figure 1: Example of encoding of an input situation for the model that learns quantifiers. For example, \( A \) and \( B \) represent the houses and the blue things in a sentence like ‘At least three of the houses are blue’. Each object in the situation is assigned a label based on which zone in the Venn diagram it belongs to. Not pictured: appended to each input is a label corresponding to the quantifier. The output of the network is a guess at the truth-value of the quantified sentence. See Steinert-Threlkeld and Szymanik (2018) for more details.

All that we have done is replaced a more specific term—‘know the R programming language’—with a strictly more general term—‘know a programming language’. The reader can verify that the inference pattern would be valid for any choice of expression instead of ‘scientist’ and pair of expressions that stand in the same specific/general relation. This validity shows that \textit{many} is upward monotone. The pattern reverses for \textit{few}.

Few scientists know a programming language.

Few scientists know the R programming language.

This shows that \textit{few} is downward monotone. We say that a determiner is monotone if it is either upward or downward monotone. Of course, not every quantifier is monotone: those expressed by the complex determiners \textit{an even number of} and \textit{at least 6 or at most 2} are not monotone.

\textbf{Monotonicity Universal:} All simple determiners are monotone (Barwise and Cooper, 1981; Peters and Westerstål, 2006).

\section{2.3 Methods}

In Steinert-Threlkeld and Szymanik (2018), we trained a recurrent neural network (specifically, a long short-term memory network (Hochreiter and Schmidhuber, 1997)) to learn the meanings of quantifiers. The input was a sequence of objects, coded to which of four regions of a Venn diagram they belong to, with the two sets corresponding to the denotation of the common noun and verb phrase of a ‘Det CN VP’ sentence. In other words, the network verifies a sentence like ‘At least three houses are blue’ against a world-model, which is represented as a sequence of objects, each labelled according to its combination of being a house and a blue thing. An example encoding can be seen in Figure 1.
For each universal, we compared one quantifier satisfying the universal with one quantifier not satisfying the universal. We ran $n = 30$ trials for each universal and measured for each quantifier and each trial the convergence point: the first point above 95% accuracy on a held-out test set of examples it has not seen and for which the future mean is above that same threshold. For complete details on model architecture, input encoding, and experimental paradigm, we refer the reader to Steinert-Threlkeld and Szymanik (2018) and the corresponding code-base: https://github.com/shanest/quantifier-rnn-learning.

### 2.4 Results

Accuracy on test set vs. training time for four different pairs of quantifiers

![Figure 2: Results of four learnability experiments for quantifiers. In all figures, a quantifier satisfying a universal is in blue, compared to a minimally different one not satisfying a universal in red. The $x$ axis is the number of training steps of a recurrent neural network, and the $y$ axis is accuracy on a held-out test set. Top row: upward monotone at least 4 (left) and downward monotone at most 3 (right) versus the non-monotone at least 6 or at most 2. Bottom row: quantitative at least 3 versus non-quantitative first 3 (left) and last 3 (right). In all cases, the quantifier satisfying the universal is learned faster than the one that does not. Figures lightly adapted from data reported in Steinert-Threlkeld and Szymanik (2018).](image-url)
monotone quantifier (top-left; at least 4) and a downward monotone quantifier (top-right; at most 3) to a non-monotone quantifier (at least 6 or at most 2). The bottom row looks at Quantity, comparing two non-quantitative quantifiers (left: the first 3; right: the last 3) to a quantitative quantifier (at least 3). Visual inspection makes it appear that the blue lines hit their convergence points earlier than the red lines, as symbolized by when they cross the green dashed line. This was confirmed by paired t-tests (all $p < 10^{-8}$), as reported in Steinert-Threlkeld and Szymanik (2018).

2.5 Discussion

For each of these experiments, we found a statistically significant difference between the convergence points of the quantifier satisfying the universal and the one that did not, allowing us to conclude that the quantifier satisfying the universal converged earlier than the one that did not. This constitutes preliminary evidence that the cross-linguistic patterns observed for quantifiers reflect the relative learnability thereof: languages tend to express in simple ways easy-to-learn meanings, and rely on complex syntax and compositional interpretation to express hard-to-learn meanings. To fully argue that constraints on the lexicon reflect learnability, one should extend these results by (i) incorporating a plausible model of the evolution of language, (ii) ruling out alternative explanations, and (iii) analyzing semantic universals in other domains. We return to (i) and (ii) in the discussion later in this paper. For (iii): our argument that the presence of quantifier universals stems from learnability should not be particular to quantifiers, but should in principle apply to other linguistic domains where constraints on the lexicon have been observed. In the next section, we present new results, extending our approach to one such domain: color terms.

3 Color Terms

We argue that the cognitive explanation in terms of learnability that we offered for function words may be the right explanation for (possibly) all semantic universals by showing that it extends naturally to content words. Specifically, we chose color terms as our case study. It has been argued that all natural language color terms denote geometrically well-behaved—specifically, convex—regions of color space (Gärdenfors, 2000, 2014; Jäger, 2010). We show that convexity can indeed be explained by accounting for its role in the process of learning color terms.

3.1 The Convexity Universal

The study of color naming has been on the forefront of the relativism vs. nativism debate that has driven much of linguistics, philosophy, and cognitive science in the twentieth century. This is only natural: although colors have a relatively simple and well-understood

---

3 See Carcassi, Steinert-Threlkeld, and Szymanik (2019) for a model of the emergence of monotone quantifiers.

4 See Steinert-Threlkeld (2019) for a similar approach to responsive predicates, those attitude verbs like ‘know’ which can embed both interrogative and declarative complements.
structure, there is at the same time fairly complex cross-linguistic variation when it comes to color terms. At first, the debate was dominated by the simple question: which colors get names (are lexicalized) across different languages? Building on pioneering early work (Berlin and Kay, 1969) and using their larger-scale World Color Survey, Kay and Berlin have proposed the existence of universal constraints on cross-language color naming in the form of a partially fixed evolutionary progression according to which languages gain color terms over time (Berlin and Kay, 1969; Kay et al., 2010). These are strong nonlinguistic/non-cognitive constraints that can be explained by postulating that color categories are organized around universal focal colors (Loreto, Mukherjee, and Tria, 2012). The relativistic approach, however, seems less concerned with explaining some of the most striking universal properties of color categorization across languages (Regier and Kay, 2009). Apart from which colors get lexicalized, all color terms denote very geometrically well-behaved regions of color space.

**Convexity Universal:** All color terms denote convex regions of color space (Gärdenfors, 2000, 2014; Jäger, 2007, 2010).

A convex region of a geometrical space is defined by the following property: for any two points \( x \) and \( y \) in the region, the straight line connecting \( x \) and \( y \) remains entirely inside the region. (More formally: \( X \) is convex if and only if for all \( x, y \in X \), for all \( t \in [0,1] \), \( t \cdot x + (1-t) \cdot y \in X \).) This has the effect that the border of the region must be very well-behaved: for instance, the regular polygons in the plane are convex, while many Tetris pieces are not. More examples will be provided in the next section.

Then the **Convexity Universal** says that the denotation of every color term is a convex region in the CIELab color space that approximates human color vision. Most importantly, this space is perceptually uniform with respect to human color vision, meaning that the distance in the space corresponds well with the visually perceived color change (Commission Internationale de L’Eclairage, 2004; Gärdenfors, 2014).

### 3.2 Methods

To test the hypothesis that convex color systems are easier to learn than non-convex ones, we generated a large number of artificial color systems, measured how convex they are, and then trained artificial neural networks to learn those color systems. Each of these steps will now be explained in turn.

We generated 300 artificial color-naming systems, by partitioning the CIELab color space into distinct categories, using Algorithm 1 below. The algorithm works intuitively as follows: first, each of the \( N \) categories are initialized randomly with a ball of \( b \) points in the space. Next, points are chosen at random, and are probabilistically assigned to the category that they are closest to. A parameter \( t \) controls how close the distribution looks to argmax (i.e. is a softmax parameter). In a final step, the borders are smoothed, by taking some fraction \( s \) of points that are mislabelled (in the sense of being inside the convex hull of a category other than the one they are assigned) are re-assigned.

To see more intuitively how the algorithm functions, Figure 3 contains samples of partitioning a 2D grid (not CIELab space), according to various settings of the two parameters \( s \)
Algorithm 1: Generate an artificial color system

**Parameters:** smooth ($s$), conn ($c$), initial ball size ($b$)

**Inputs:** a set $X$, distance measure $d$, number of categories $N$

```
UNLABELED ← $X$; LABELED$_i$ ← $\emptyset$ ($\forall i \in \{1, \ldots, N\}$)

Choose $x_1, \ldots, x_N$ uniformly at random from $X$

for $i = 1, \ldots, N$ do
    LABELED$_i$ += $x_i$; pop($x_i$, UNLABELED)
    for all $x \in $ NearestNeighbors($x_i$, $b$) do
        LABELED$_i$ += $x$; pop($x$, UNLABELED)
    end for
end for

while UNLABELED $\neq \emptyset$ do
    $x$ ← uniformly randomly sampled point from UNLABELED
    $d_i$ ← $1/(\min_{x' \in $ LABELED$_i$} d(x, x'))^{1/4}$
    $p_i$ ← $e^{d_i/c} \sum_j e^{d_j/c}$
    Choose label $i$ with probability $p_i$
    LABELED$_i$ += $x$; pop($x$, UNLABELED)
end while

for $i = 1, \ldots, N$, ordered by increasing size of LABELED$_i$ do
    $M_i$ ← ConvexHull(LABELD$_i$) \ LABELED$_i$
    $R_i$ ← ClosestPoints($M_i$, LABELED$_i$, $s \cdot |M_i|$)
    for all $x \in R_i$ do
        LABELED$_i$ += $x$; pop($x$, cell($x$))
    end for
end for
```

and $c$. As can be seen, the top-left corner looks like complete noise. As $c$ gets smaller, the regions appear to be more connected, but with quite jagged borders. Then, as $s$ increases, the borders get smoother and smoother. The two bottom samples in the right-most column are in fact perfectly convex.

For the results reported in the paper, we set $b = 100$, $N = 7$, and used Euclidean distance. Our set of points $X$ was 9261 points in CIELab space generated by taking the unit cube, with each axis partitioned at steps of interval 0.05, and converting these aRGB points to CIELab. The parameter $c$ came from $\{5, 1, 0.1, 0.01, 0.001, 0.0001\}$ and $s$ from $\{0, 0.25, 0.5, 0.75, 1\}$. We ran 10 trials for each pair of the two parameters, for a total of 300 trials.

For each color system, we measured its degree of convexity. Our measure is intended to capture the idea that some regions are more convex than others, even though they are not fully convex. This can already be seen in Figure 3, where the top-left samples are much ‘less convex’ than the rest, on an intuitive notion. Our measure captures the following idea: a color system’s degree of convexity corresponds to how close it is to the closest convex color system. This can be made precise as follows. Every region $X$ can be extended to the smallest convex region that contains it; this is known as its convex hull. (Note that $X$ is convex just in case $X = \text{ConvexHull}(X)$.) For a region $X$, we can take $|X|/|\text{ConvexHull}(X)|$—how much of the convex hull does $X$ already cover—as a measure for how convex it is. For example,
consider the three regions in Figure 4.

$C$ is perfectly convex. Both $nC$ and $nC^+$ are non-convex and in particular have the exact same convex hull: the regular pentagon. Intuitively, however, $nC^+$ is less convex than $nC$. Our measure captures this, since $nC^+$ covers a smaller fraction of its own convex hull than does $nC$. One would have to do more work to convert $nC^+$ into its nearest convex region than for $nC$.

Finally, to measure degree of convexity for an entire color system, we take the weighted average of the above measure for each color term in the system. More precisely, letting $P$ be a partition of a space and writing $P^i$ for the $i$th cell of the partition, and $|\cdot|$ for the size of a set:

$$\text{degree of convexity}(P) := \frac{\sum_i |P^i| \cdot \frac{|P^i|}{\text{ConvexHull}(P^i)}}{\sum_i |P^i|}$$

To measure how learnable each color system is, we trained an artificial neural network to learn each system (Goodfellow, Bengio, and Courville, 2016). In particular, we used a multi-layer feed-forward neural network, with three input neurons (corresponding to dimensions in CIELab color space), $N = 7$ output neurons, and two hidden layers of 32 units each with
Figure 4: Three regions in a space. \( C \) is perfectly convex. \( nC \) and \( nC^+ \) are both not convex, but the latter is intuitively less convex than the former. Our measure captures this intuition.

the Exponential Linear activation function (Clevert, Unterthiner, and Hochreiter, 2016). We trained for six epochs, using the Adam optimizer (Kingma and Ba, 2015) with default parameters. Everything was implemented in TensorFlow (Abadi et al., 2016).

More details, as well as all of the code and data, may be found at https://github.com/shanest/color-learning.

3.3 Results

Figure 5 shows the accuracy of the neural networks on a held-out test set (examples from the color systems that it did not see during training) plotted against the degree of convexity on the \( x \) axis. There’s a clear positive correlation, as confirmed by a linear regression (\( R^2 = 0.505; F = 304.4; p = 1.85e^{-47} \)).

To show that degree of convexity is the best explanation of the network’s ability to learn a color system, we did two things. First, we did a linear regression of accuracy against many other variables of interest: the parameters \( s \) and \( c \) of Algorithm 1, as well as their interaction, \( s \cdot c \), and a few geometric properties of the color systems: the size of the largest (max) color, of the smallest (min) color, their difference (max-min) and ratio (min/max), as well as the median color size. The first column of Table 1 shows that the degree of convexity has by far the highest \( R^2 \) of any of the variables, with almost all of them having negligible values.

Secondly, we used a technique called commonality analysis to measure the unique effect of each variable (Nimon and Reio, 2011). This works as follows: first, a multiple regression was run with all of the aforementioned variables as predictors for accuracy. Then, for each variable, we ran a multiple regression using all variables except for it. The unique effect of each variable is the difference in \( R^2 \) (\( \Delta R^2 \)) between those two regressions. In other words, how much variance in the accuracy data can be explained by each variable, over and above the variance that can be explained by all of the other variables.

The second column of Table 1 shows that degree of convexity is the only variable with
Figure 5: Network accuracy on test set versus degree of convexity for 300 color systems as generated by partitioning CIELab space according to Algorithm 1.

a significant unique effect. In particular, neither of the algorithm’s parameters, nor their interaction, have a non-negligible unique effect. These analyses show that degree of convexity does the main explanatory work in our main result.

### 3.3.1 Controlling for Linear Separability

We further analyzed the results of our experiment to address the following worry. Convex regions have linear boundaries,\(^5\) and linearly separable categories are known to be easy to learn for neural networks.\(^6\) Because of this, it is possible that the role of degree of convexity in explaining network accuracy stems entirely from its correlation with linear separability. In order to control for such an effect, we sought to measure a degree of linear separability of the color systems, and then check whether the degree of convexity still explained more of the data than can be explained by linear separability on its own.\(^7\) If so, this would show that the effect we have observed is not due entirely to how linearly separable the color systems are.

---

5The circle is the infinite limit of regular polygons.

6A neural network with zero hidden layers—a perceptron—just is a linear classifier.

7Thanks to two anonymous reviewers and to Emmanuel Chemla for pressing this line of thinking, and to an anonymous reviewer for suggesting the control that we implemented.
Table 1: For each variable of interest: $R^2$ for a regression of accuracy on that variable, and the unique contribution of that variable, measured as $\Delta R^2$, the difference in $R^2$ between the complete multiple regression and a regression with all variables except the indicated one.

<table>
<thead>
<tr>
<th>Variable</th>
<th>$R^2$</th>
<th>$\Delta R^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>conn</td>
<td>0.180</td>
<td>0.0003</td>
</tr>
<tr>
<td>smooth</td>
<td>0.008</td>
<td>0.0365</td>
</tr>
<tr>
<td>degree of convexity</td>
<td>0.505</td>
<td>0.3726</td>
</tr>
<tr>
<td>conn · smooth</td>
<td>0.054</td>
<td>0.0019</td>
</tr>
<tr>
<td>min size</td>
<td>0.014</td>
<td>0.0000</td>
</tr>
<tr>
<td>max size</td>
<td>0.001</td>
<td>0.0000</td>
</tr>
<tr>
<td>median size</td>
<td>0.000</td>
<td>0.0007</td>
</tr>
<tr>
<td>min / max</td>
<td>0.043</td>
<td>0.0014</td>
</tr>
<tr>
<td>max − min</td>
<td>0.000</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

To measure the linear separability of the color systems, we did the following: we trained a linear classifier to learn the color system and then measured its accuracy. Note that we did not use a train/test split in the data for these purposes: we want to measure not how well a linear model can generalize in the data, but rather how well it can memorize the data, since that will correspond to how linearly separable it is.

We then re-did the unique effects analysis described above, with this measure of linear separability included as a predictor variable. Table 2 shows the analysis for degree of convexity and linear separability. While we incorporated all of the variables as in the analysis reported above, all other variables were negligibly different from what was reported in Table 1. For completeness, we include the entire table in the Supplementary Materials section.

The left-hand column—which reports the $R^2$ of a regression with each variable on its own—shows that both linear separability and degree of convexity are good predictors of accuracy, though degree of convexity explains a larger portion of the variance in the accuracy results.

Most important, however, are the unique effects of each variable, reported in the right-hand column. The low value for linear separability shows that it does not explain any of the variance in accuracy that cannot be explained by all of the other variables (which includes degree of convexity). On the other hand, however, even in the presence of a predictor for linear separability, we still see that the degree of convexity has a large unique effect ($\Delta R^2$). That is to say: degree of convexity does explain a significant amount of the variance in the accuracy data, over and above what can be explained by the linear separability of the color systems (and all of the other variables that we measured).

We also include the results of the large regression, with all variables, in Table 3. Degree of convexity and $s$ are the only significant variables in the regression, though the former has a much larger coefficient.
Table 2: Commonality analysis, exactly as in Table 1, but with linear separability added as a predictor. While the $\Delta R^2$ value for degree of convexity does go down in the presence of linear separability, it remains the only significant value. (Other variables are omitted for space, since their values were nearly identical.)

<table>
<thead>
<tr>
<th>Variable</th>
<th>$R^2$</th>
<th>$\Delta R^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>degree of convexity</td>
<td>0.505</td>
<td>0.1288</td>
</tr>
<tr>
<td>linear separability</td>
<td>0.418</td>
<td>0.0005</td>
</tr>
</tbody>
</table>

Table 3: Multiple regression of network accuracy against all the variables we measured.

<table>
<thead>
<tr>
<th>Variable</th>
<th>coef</th>
<th>std err</th>
<th>$t$</th>
<th>$p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>intercept</td>
<td>0.1112</td>
<td>0.090</td>
<td>1.231</td>
<td>0.219</td>
</tr>
<tr>
<td>degree of convexity</td>
<td>0.7571</td>
<td>0.074</td>
<td>10.173</td>
<td>5.487e-21</td>
</tr>
<tr>
<td>$c$</td>
<td>-0.0040</td>
<td>0.009</td>
<td>-0.440</td>
<td>0.660</td>
</tr>
<tr>
<td>$s$</td>
<td>-0.1806</td>
<td>0.033</td>
<td>-5.448</td>
<td>1.088e-7</td>
</tr>
<tr>
<td>$s \cdot c$</td>
<td>0.0153</td>
<td>0.014</td>
<td>1.102</td>
<td>0.271</td>
</tr>
<tr>
<td>linear accuracy</td>
<td>-0.0475</td>
<td>0.075</td>
<td>-0.634</td>
<td>0.526</td>
</tr>
<tr>
<td>max</td>
<td>1.651e-6</td>
<td>3.68e-5</td>
<td>0.045</td>
<td>0.964</td>
</tr>
<tr>
<td>min</td>
<td>3.027e-5</td>
<td>6.93e-5</td>
<td>0.436</td>
<td>0.663</td>
</tr>
<tr>
<td>max - min</td>
<td>-2.86e-5</td>
<td>3.37e-5</td>
<td>-0.850</td>
<td>0.396</td>
</tr>
<tr>
<td>min / max</td>
<td>-0.1508</td>
<td>0.138</td>
<td>-1.091</td>
<td>0.276</td>
</tr>
<tr>
<td>median</td>
<td>3.16e-5</td>
<td>4.14e-5</td>
<td>0.764</td>
<td>0.445</td>
</tr>
</tbody>
</table>

3.3.2 Cluster Analysis

For a final analysis, we sought to determine whether something could explain why there appears to be a cluster of color naming systems that are fairly high in degree of convexity while being very hard to learn (i.e. those in the bottom-right of 5).8

In particular, we performed model-based clustering using finite Gaussian mixture modelling.9 In particular, several candidate clustering models were tested, using varying numbers of clusters. We chose the best model according to its Bayes Information Criterion. This model had five clusters,10 one of which very nicely captured the high-convexity but low-accuracy data points.

To understand one factor driving this clustering, Figure 6 shows our primary data with the shape of each point indicating its cluster, and the color of the points indicating the median size of a color term in the color system. A very clear pattern can be seen: the points in the convex-but-hard-to-learn cluster (#3) all have very low median size. This suggests that these color systems are very unbalanced: one or two color terms span most of the space.

---

8Thanks to an anonymous referee and Leendert van Maanen for posing this question, and to the latter for assisting with the analysis.
9All of this was done using the mclust package for R (Scrucca et al., 2016).
10In particular, it was a VVI model of clustering, which has variable volume and shape ellipsoidal clusters and a diagonal distribution. See Scrucca et al., 2016 for more details on the clustering models compared.
Figure 6: Accuracy versus degree of convexity, optimally clustered. The different clusters are
given by the shape of the points. The color corresponds to the median size of a color term in
the system. As can be seen, the cluster that has high degree of convexity but low accuracy
also has very low median size. This suggests that those systems are very unbalanced.

It makes sense that these are likely to be convex: larger regions are more likely to have a high
degree of convexity, and they also receive more weight in the weighted average. It makes
sense that these systems will be harder to learn: to make the training data balanced, we
upsampled all categories to have as many data points as the largest; but this could prompt
the network to memorize the training data from very small colors, hurting its ability to
generalize. Future work will further explore the role of and empirical motivation for more
balanced color systems.

We also note that, as expected, removing the relevant cluster increases the strength of
our main result: a regression of accuracy as predicted by degree of convexity for the subset
of the data without that cluster has a much higher $R^2 = 0.7$ ($F = 620.7, p = 2.2e-16$).

3.4 Discussion

We generated 300 artificial color systems and measured how well a multi-layer perceptron
can learn each of them, and predicted this accuracy from many different variables. We found
the degree of convexity to be the strongest individual predictor of the accuracy results. It is
also the only variable to have a significant unique effect, i.e. to make a unique contribution to explaining variance over and above all of the other variables. This result holds even when controlling for the linear separability of the color systems. These results provide strong evidence that more convex color systems are indeed easier to learn.

There are minimally two ways of interpreting the measurement of simplicity by ease of learning for a neural network. On the one hand, neural networks are universal function approximators that have exploded in popularity because of their ease of implementation on modern hardware. Because they are designed to work on any function, the learning biases that they have are biases that are likely to be shared by any (universal) function approximator. Therefore, the fact that certain quantifiers and color systems are easier to learn for such networks provides evidence that they are easier to learn in general. On the other hand, this simplicity measure could be taken to be similar to that defined by human cognitive systems. While there is some evidence for such similarity (Kirov and Cotterell, 2018; Peterson, Abbott, and Griffiths, 2018), the issue is very far from settled. We are officially agnostic on which of these interpretations to take, and will return to the general issue of complexity versus learnability after discussing two other topics: our measure of the degree of convexity and possible alternative models of learning.

3.4.1 Other Measures of Convexity

We note that our result links neural network learnability to one particular measure of the degree of convexity: the (weighted) average of the convex hull covered by each cell of a partition. It is natural to wonder what properties of this measure are responsible for the learnability results and whether other measures of a degree of convexity would behave similarly. To the latter end, we note that our measure is intended to capture how much ‘work’ would need to be done to turn a color system into the nearest convex one. It does not directly encode another intuition about degree of convexity, namely that such a degree should correlate with smooth boundaries. On our measure, figures like a solid capital ‘T’ will have relatively low degree of convexity despite having borders that are smooth except for a few right angles. By contrast, figures with very rough borders, but with deviations from straight lines that are small in absolute terms, can have very high degree of convexity on our measure. Precisely formulating alternative measures of the degree of convexity and studying whether neural networks and other models of learning are more sensitive to the roughness of the border or to larger departures from the convex hull (or combinations thereof) constitute interesting avenues for future work. Along these lines, a fuller understanding of the decision regions of neural networks may help uncover factors beyond convexity that explain the accuracy results presented here (G. J. Gibson and Cowan, 1990; Montúfar et al., 2014; Nguyen, Mukkamala, and Hein, 2018).

3.4.2 Other Models of Learning

The other side of the link in our main result is the model of learning. It is in principle possible that the correlation between degree of convexity and learnability will vanish for other models of learning. For example, $k$ nearest neighbors classification may not be biased.

\footnote{Thanks to an anonymous reviewer for this articulation of the issue.}
towards convex systems in the same way. And given that the exemplar theory of concepts works similarly, it’s possible that human concept learning will behave differently (Medin and Schaffer, 1978; Nosofsky, 1988; Smith and Medin, 1999). Nevertheless, recent work has provided a neurobiological grounding for exemplar theory (Ashby and Rosedahl, 2017), so the difference between the two approaches may not be as large as it initially appears. More generally, as discussed above, we find it plausible that the features of convex systems that make them easier to learn for neural networks are likely to make them easier to learn for many other models of learning, as well as for humans. And while one cannot infer from our results that human learning of colors resembles our model more than others, we do note that as more evidence for the learnability hypothesis accumulates, consistency with cross-linguistic typology can provide another source of constraints in modeling learning.

3.4.3 Complexity and Learnability

We have proposed ease of learning by an artificial neural network as a measure of complexity. But there is no a priori connection between complexity/simplicity and learnability. On top of that, the discovery of an empirical connection between simplicity and human concept learning has been a significant advance in modern cognitive science (Feldman, 2000; Goodman et al., 2008; Piantadosi, Tenenbaum, and Goodman, 2016). This work, among others, has caused some authors to posit simplicity as a unifying principle in human cognition more generally, beyond just concept learning (Chater and Vitányi, 2003; Feldman, 2003, 2016). The conceptual gap between learnability and complexity raises important questions for the present work: should neural network learnability be taken as an alternative measure of complexity, or might some of the existing measures just cited explain both the learnability facts and the typological facts?

We present some preliminary evidence to be skeptical of the fact that existing approaches to complexity will also explain the learnability results presented here. In particular, van de Pol, Steinert-Threlkeld, and Szymanik (2019) apply a common measure of simplicity as compressibility—(approximate) Kolmogorov complexity (Chater and Vitányi, 2003; Dingle, Camargo, and Louis, 2018; Feldman, 2016; Lempel and Ziv, 1976; Li and Vitányi, 2008)—to generalized quantifiers and compare the results to the learnability results presented in Section 2. They find that the monotone quantifiers studied here are less complex than the non-monotone one, but that there is no consistent difference in complexity between the quantitative and non-quantitative quantifiers. While much work remains to be done here, these results tentatively suggest that our measure of learnability differs from other standard measures of complexity and that learnability better explains the presence of semantic universals.

Another common and related measure of complexity in the aforementioned literature consists in the minimum description length in a Language of Thought. On this approach, concepts are represented as composed out of conceptual primitives, and those that can be expressed more simply tend to be easier to learn (Feldman, 2000; Goodman et al., 2008; Piantadosi, Tenenbaum, and Goodman, 2016). It remains an open and interesting problem to apply these methods to the semantic domains discussed in the present paper. We note that Steinert-Threlkeld and Szymanik (2018) provide some preliminary reason for skepticism about such measures of complexity correlating with the universals for quantifiers. And while
it is natural to think that convex color systems will be easier to describe than non-convex ones—they can be described by specifying the separating hyper-planes corresponding to their borders—providing a general geometric language for describing such systems on which this pans out will be a very non-trivial task.\footnote{See Amalric et al. (2017) for an initial step in that direction.}

The upshot of this discussion is that while neural network learnability provides a unified explanation of semantic universals in many different domains, whether or not other general notions of complexity from the literature also can remains open (and there is initial reason for skepticism about Kolmogorov complexity). It is, for instance, possible that minimal description length in an appropriate LoT can explain both the presence of the universals and their ease of learning. We welcome future work exploring the relations between all of these notions.

4 Conclusion

It’s natural to think that the languages of the world are shaped both by the cognitive makeup of language users and their local communicative needs, hence the nativism–relativism debate. Universals are more likely to be explained by cognitive makeup, since this does not vary in the way that communicative needs vary. We have argued that one aspect of our cognitive makeup—simplicity as measured by ease of learning—can explain the presence of multiple semantic universals in disparate domains. This explanation represents the first explanation of semantic universals that applies both to function words and to content words.

Acknowledgments

The authors have received funding from the European Research Council under the European Union’s Seventh Framework Programme (FP/2007-2013)/ERC Grant Agreement n. STG 716230 CoSaQ.

Supplementary Material

The code for running the quantifier experiments, as well as the data and the analysis scripts, are publicly available at \url{https://github.com/shanest/quantifier-rnn-learning}. The code for running the color experiment, as well as the data and the analysis scripts, are publicly available at \url{https://github.com/shanest/color-learning}.

Table 4 reports the full commonality analysis with linear separability; the most significant variables were reported in the main body as Table 2.
<table>
<thead>
<tr>
<th>Variable</th>
<th>$R^2$</th>
<th>$\Delta R^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>conn</td>
<td>0.180</td>
<td>0.0003</td>
</tr>
<tr>
<td>smooth</td>
<td>0.008</td>
<td>0.0369</td>
</tr>
<tr>
<td>degree of convexity</td>
<td>0.505</td>
<td>0.1288</td>
</tr>
<tr>
<td>linear separability</td>
<td>0.418</td>
<td>0.0005</td>
</tr>
<tr>
<td>conn $\cdot$ smooth</td>
<td>0.054</td>
<td>0.0000</td>
</tr>
<tr>
<td>min size</td>
<td>0.014</td>
<td>0.0000</td>
</tr>
<tr>
<td>max size</td>
<td>0.001</td>
<td>0.0000</td>
</tr>
<tr>
<td>median size</td>
<td>0.000</td>
<td>0.0007</td>
</tr>
<tr>
<td>min / max</td>
<td>0.043</td>
<td>0.0015</td>
</tr>
<tr>
<td>max $-$ min</td>
<td>0.000</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

Table 4: Commonality analysis, exactly as in Table 1, but with linear separability added as a predictor. While the $\Delta R^2$ value for degree of convexity does go down in the presence of linear separability, it remains the only significant value. This expands the partial table Figure 2 which only reported the degree of convexity and linear separability rows.

References


