Theory-Neutral System Regularity Measurements

Patrick Juola (PATRICK.JUOLA@PSY.OX.AC.UK)[1]
Todd M. Bailey (TODD.BAILEY@PSY.OX.AC.UK)
Emmanuel M. Pothos (POTHOS@PSY.OX.AC.UK)
Department of Experimental Psychology
University of Oxford
Oxford, OX1 3UD UNITED KINGDOM

Abstract

Traditionally, regularity in a data set is assessed by fitting a model to the data and examining the extent to which the variance accounted for by the model is large compared to the overall variance in the system. Such approaches, however, do not address the complementary question of how much regularity is present in the data, in the first place, and how much work is expected to be required to capture a particular amount of regularity. In this work we use the notion of Kolmogorov complexity to derive a measure of system regularity independent of any particular model. Thus, in our framework, the explanatory adequacy of a model can be readily quantified, so that one can examine the extent to which the model is satisfactory, or whether additional mechanisms need be postulated.

Introduction

Reading [English] aloud — the process of converting printed letters into sounds — is a classic and well-studied example of a quasi-regular system (e.g. Plaut, McClelland, Seidenberg, & Patterson, 1996). Most of the data can be captured very simply, but there are still a large number of irregular "exceptions" of varying degrees of subregularity themselves, and a full account of the reading problem must take these into account. Different models of reading will address irregularity in different ways, and a reasonable question is how much regularity must be taken into account, or just how irregular is the particular system of reading aloud? Similar quasi-regularity can be found in many other systems, such as the production of morphologically inflected forms from their stems, the acquisition and description of syntax, similarity judgements, or almost any non-trivial psychological process.

Traditional statistical techniques that address such issues provide a measure of variance, not regularity, in the absence of a specific model. Regularity itself can usually only be measured after the fact. One develops a model, confirms that it accounts (or does not account) for enough of the data to be interesting, and then declares a system or relation to be regular if the data is well described and the model is simple enough. Neither of these determinations are particularly objective or quantifiable; nor are they easy to compare between different models. Furthermore, there is little insight on the total regularity present and, crucially, of the proportion of regularity accounted for by any given model.

In this work, we use the notion of "Kolmogorov complexity" (Li & Vitányi, 1997; Chaitin, 1995/1996) to suggest a framework under which regularity can be directly and quantitatively measured. This paper develops a numerical theory that can be used to measure the complexity of a cognitive or computational function (such as reading aloud or inflecting word stems to other forms) in terms of the expected performance of an estimated best possible model, and thus independent of any specific model that the researcher may choose to employ. This will allow researchers to address previously unanswerable questions such as the degree of irregularity, or the degree to which *any* model could predict a set of data, or the amount of additional information that would need to be provided to allow accurate predictions. It further provides a theory-neutral common ground for the discussion of the performance of various models on the same task, or the same model on different tasks.

Background

System Regularity

What is meant, in this context, by "system regularity"? Focusing for the moment on the problem of reading aloud (in English), a simple letter-to-sound conversion table will produce good, but not superb, results. Performance can be improved by adding more complex rules, for example that 'c' is pronounced /s/ in front of an 'e' 'i' or 'y', or by enumerating exceptions with irregular pronunciation. The more complex the rules need to be, or the more exceptions that need be tabulated, the greater is the complexity of the system described.

This notion should not be confused with simple variance; a system may produce widely varying outputs based on subtle but very simple changes in the inputs. For example, dialing telephone numbers at random is likely to produce extreme variation in where the call goes. However, inspection of the first few digits (the area code or the exchange) will allow someone to predict the call's destination to within a small area, capturing much of the geographic variance by a relatively simple system. Without the insight to look for area codes and exchanges, however, predicting of destinations (perhaps as a function of the factors of the telephone numbers?) would be very difficult.

[1]Current address: Department of Mathematics and Computer Science, Duquesne University, Pittsburgh, PA 15282 (JUOLA@MATHCS.DUQ.EDU)

In many cases, a determination of the exact rules underlying a particular task is not easy or possible — for example, in the telephone system above or in artificial grammar learning [see later, as well (Reber, 1967; Pothos & Bailey, 1997)]. However, a system can still be regarded as more regular if a few, common, easy rules can capture more of its variance than rules of comparable complexity could capture of another. In this framework, then, system regularity can be regarded as a measure of the ease of expressing a set of dependent data in terms of an associated set of independent observations. If this "difficulty" can be suitably formalized, this yields a theory-neutral measure of system regularity.

Kolmogorov Complexity

Even given a highly predictable system, the prediction task may be non-trivial, as the system/data may not reduce to an obvious model — for example, a high-degree polynomial is very predictable, but attempts to find a low degree (or linear) fit to that polynomial will result in abject failures. It is useful to have an idea of how redundant — or predictable — a system is, in order to figure out whether a particular model seems to be able to capture a meaningful generalization of the data, or to determine when one is doing pointless curve-fitting.

To this end, we use the information-theoretic concept of Kolmogorov complexity (Li & Vitányi, 1997; Chaitin, 1995/1996). For any sequence of symbols, the Kolmogorov complexity is the length of the shortest algorithm (computer program) that will exactly generate that sequence (and then stop). A sequence of a million question marks, for example, would have a very low Kolmogorov complexity, as it could be generated from a simple loop and a counter. On the other hand, a university telephone book, although probably containing fewer than a million characters, would have a much higher Kolmogorov complexity, in part because the assignment of names to telephone numbers is difficult to predict (and would have to be hand-coded in to any program). At the extreme, a sequence of random digits has the highest Kolmogorov complexity of all, as there is *no* algorithm (by definition) that can exactly reconstruct a sequence of random digits shorter than hand-copying them from memory.

Conditional Kolmogorov complexity measures the minimal amount of information (the shortest program) required to generate a string of dependent data from a string of independent data (conditional complexity is closely related to "information distance", but the former is asymmetric while the latter is symmetric; cf. Li and Vitányi (1997), ch. 8.3). Chater and Hahn (1997) have argued, more generally, that conditional Kolmogorov complexity captures the psychological aspects of distortion and similarity. Specifically, the amount of work necessary to transform one mental representation into another can be linked both to the computational complexity, broadly defined, of the transformation as well as the perceptual similarity between the two representations.

Juola has similarly argued that Kolmogorov complexity can be used as a metric for the amount of information contained in a given text, and that this complexity can be used to directly measure cultural effects and explicitness in translation (Juola, 1997) and indirectly to assess morphological complexity (Juola, in press). By systematically distorting morphological regularities, one can determine the degree of importance of morphological regularities by the amount that the total complexity changes. A similar technique is used here to systematically distort functional relationships as an assessment of system regularity.

Specifically, we assume that the data describing a system can be encoded into an (ordered) string of $< x_i >$, $F(< x_i >)$ pairs, where $< x_i >$ and $F(< x_i >)$ might be, for instance, a string of characters to be read aloud and the correct pronunciation of that string. Appending many spelling-pronunciation pairs together (in some order) will yield a very long string defining the global data set. This string describes every data point that needs to be fitted in a model of the data — and thus the most regular model for that string will be very close to a most regular model for the data itself.

A regular system is just one in which each $F(< x_i >)$ is predictable from $< x_i >$. We can further identify subcategories of regularity, including a regularity imposed by lack of variance (if every $F(< x >)$ were constant), or mapping regularity, where $F(< x_i >)$ is predictable from $< x_i >$ but not, in general, from any (random) $< x_j >$). In other words, in a regular system, each $< x_i >$ contains a lot of information about the corresponding $F(< x_i >)$, information which is not necessarily trivially available but which depends on the relationship defined in the pairing.

Measuring Kolmogorov Complexity

Working directly with the Kolmogorov complexity of objects is difficult. First, Kolmogorov complexity itself is, strictly speaking, uncomputable — although the question "does this string have a complexity of less than N" can be definitively answered affirmatively (by showing such a program), there is no way to prove a "no" answer without proving that no program shorter than N prints the string and then stops. Proving that a program stops is the well-known and unsolvable "halting problem." Thus, finding an exact measure of Kolmogorov complexity is also unsolvable. Secondly, Kolmogorov complexity is only well-defined for strings, and not, for example, for functions. The related notion (conditional Kolmogorov complexity, or information distance) that can be used to describe the complexity of functions, is no more computable. Nevertheless, Kolmogorov complexity can be estimated.

In fact, any successful file compression, as done by any number of standard tools such as *gzip*, can be regarded as an estimate of the Kolmogorov complexity of a given file. The proof (found in Li & Vitányi, 1997), can be seen intuitively by observing that a decompression program *plus the compressed file* serve as a computer program to (re)generate the original file. Because the decompression program itself is held constant, complexity differences in the original strings (to be compressed) can only appear

as differences in the length of the compressed data. Most modern compression algorithms, such as Ziv and Lempel (1977, 1978), can be proven to be "asymptotically optimal," or in other words, to converge to within any desired degree of accuracy given enough data to work with.

The Compressivity Test

We apply compression to get estimates of the system regularities present in a set of data. Specifically, we assume that the data can be encoded into an (ordered) string of $< x >$, $F(< x >)$ pairs, as described above. Compressing this string yields an estimate of the collective complexities of the data, the ordering of the data, and the functional relationship F. Accidental effects produced by the ordering of the data are eliminated by repeatedly permuting the set of pairs prior to compression (in other words, appending the pairs in different orders) and taking the mean value of the complexity estimates. This mean value, which we denote by C, is an estimate of the order-independent complexity of the data set. Specifically, C can be interpreted as an estimate of the total complexity of the distribution of independent variables $< x >$, the dependent variables $F(< x >)$, and the mapping between them.

To estimate the degree to which mapping regularity, in particular, plays a part in the overall complexity measurement, we break the (fixed) mapping by reassigning the set of dependent variables randomly. The degree of this sort of regularity can be quantified in terms of the difference in complexity between the string of $< x_i >$, $F(< x_i >)$ pairs and the string of scrambled $< x_i >$, $F(< x_j >)$ pairs (mapping regularity is related to, but distinct from "algorithmic information" as defined by Li and Vitányi (1997), ch. 2.8). In other words, we produce a new string of $< x_i >$, $F(< x_j >)$ pairs, where $< x_j >$ is some element chosen randomly (without replacement) from the set of all stimuli. Compressing this string, of course, will yield a slightly different complexity estimate of a different function over the same domain and range, but with the causal and statistical connections between domain and range broken (and replaced by random "noise"). Again, permuting the pairs and taking the mean complexity yields an overall estimate of the complexity of the scrambled data; averaging across different scrambled pairings (this mean value will be denoted R) yields an estimate of the complexity (and predictability) of the scrambled data itself irrespective of any system regularities that can be used to predict the original $F < x >$ from $< x >$. Thus, R is estimates the total complexity of the data under the assumption that the mapping is random (and hence maximally complex). The difference between these two measures $(R - C)$ estimates the the degree to which the mapping from $< x >$ to $F < x >$ differs from a completely random assignment, i.e. the degree of system regularity described by F. It may seem odd to take the *mean* of a series of overestimates, instead of the minimum, but for simple comparisons between two sets of semi-reliable estimates (as done here), the mean difference is a more consistent distance measure than the minimum. The following section describes two pilot computer simulations to demonstrate the application of this technique.

Simulations

Simulation 1

The first simulation is a demonstration of the correspondence between intuitive notions of "regularity" and the results of the compression technique described above. The data set can be regarded as a very simple version of reading aloud, with a simple (regular) task, and a gradually increasing number of "exceptions" which require something slightly different (although still self-consistent) to be done.

For independent variables, we used a collection of ten-place binary vectors and included all vectors with exactly one or two bits set to 1 (and the rest, of course, set to zero). This yielded an experimental set of fifty-five vectors. With each vector, we associated as the dependent variable either the vector itself (identity output), or the bitwise inverse of the vector (so the vector 0100100000 would be associated either with the vector 0100100000 or with 1011011111). Intuitively, one can think of ten-bit images and producing either a positive or a negative image. The choice between the vector or its inverse was made randomly (per datum), with some probability p between zero and one. (Alternately, we assigned some number N between 0 and 55, inclusive, of the vectors to their inverses).

We further analyze the total complexity (C) of the of the resulting file as composed of three factors — the complexity of the domain (which is constant), the complexity of the range as distinct from the domain (which will gradually and monotonically increase as p increases, reflecting the gradual addition of more and more novel range elements), and the complexity of the mapping between the domain and range (which intuitively is the difficulty of deciding whether or not to invert any particular domain element).

Intuitively, setting p (N) equal to 0 describes a very simple task of copying inputs to outputs, requiring no processing at all for F. This produces an extremely simple, regular file with relatively low Kolmogorov complexity. Setting p equal to 1 (N = 55) will, intuitively, result in a slightly less compressable file — although the function itself is of approximately equal complexity, the complexity of the domain and the range, taken as a set union, is greater. Thus, the overall complexity of the file would be increased. For intermediate values of p, however, we expect to see greatly increased complexity resulting from the indeterminacy of whether or not any individual datum takes a positive or negative image. At the two extremes, the task of deciding whether or not to invert the datum is very simple as it is either always, or never, done. In the middle, however, the problem of deciding whether a particular pattern is inverted is simply one of deciding between two random subsets. For $p = 0.5$, this decision is maximally difficult (and maximally complex), dropping symmetrically to zero at the extremes. Combining these two predictions suggests that C will be at a

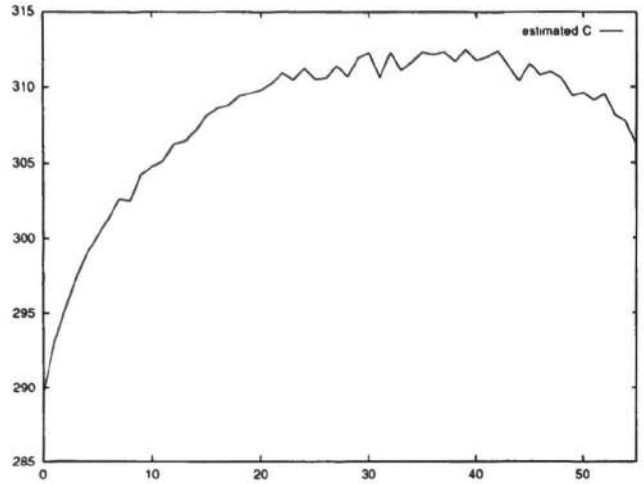

Figure 1: Mean values of C for increasing number of "inverse" productions

global minimum at $p = 0$, a relative minimum at $p = 1$, and achieve a maximum somewhere just above $p = 0.5$ (where the near-maximum decision complexity combines with the smoothly increasing domain/range complexity to achieve a maximum).

Figure 1 shows the mean value of C for 100 trials at each value of N. Graphing N against C shows the expected concave-downward pattern that achieves a stable maximum just past the halfway point. The increasing complexity observed in the left half of the graph reflects the increasing complexity of the range as additional exception items are added, combined with the increasing complexity of the mapping from domain to range as the mapping becomes less predictable. The *decrease* in C as N increases past about 30 shows that, even though the complexity of the range continues increasing as more and more exception items are added, this is more than offset by the increased regularity with which the decision to invert is predicted. In other words, the overall complexity of this quasi-regular system is *measurably* decreasing (at $N > 30$).

Simulation 2

The second simulation was a replication of the analysis of the artificial grammar learning (henceforth AGL) data from (Pothos & Bailey, 1997). As is typical in AGL experiments, subjects in this experiment were presented with a set of training images generated by a standard finite-state language. (Strictly speaking, subjects were exposed to a set of nested geometric shapes converted via a simple transformation from a standard finite-state language). Subjects were then asked, first, whether a set of novel stimuli complied with the "rules" of the finite-state language (whether the novel stimuli were "grammatical"), and secondly, to make pairwise judgements about the similarity between an individual subset of the test and training items.

The question asked, and answered, by Pothos and Bailey, was whether the collective set of pairwise similarities was a significant predictor of the grammaticality judgements. They measured the extent to which a Nosofsky classifier (Nosofsky, 1992) would correctly predic t the subjects' grammaticality judgements on the basis of their (independent) similarity judgements. Their results, in brief, are that similarity — or more accurately, the sort of similarity to which a Nosofsky classifier is sensitive — can account for a significant amount of the variance in grammaticality judgements.

Of course, a failure to find any similarity effects would not have demonstrated that similarity was not predictive of grammaticality judgements, only that the model that they applied (the particular version of the Nosofsky classifier) was not sensitive to the types of similarity that *were* present in the experiment and predictive. For instance, to find this effect required a certain amount of analysis and tuning of the parameter space, such as determining the optimum number of dimensions in which to project the similarity judgements, and the optimal metric to use. A set of suboptimal parameters would have resulted in a worse fit, possibly even a non-significant fit, with corresponding implications about the effects of similarity judgements on classification. Similarly, some classification problems might be structured such that no Nosofsky model will capture the variance, while still being (in theory) predictive according to a more appropriate model. An example of such a problem is given in the following section.

We applied the compressivity test to determine whether or not Kolmogorov complexity would confirm that similarity predicts grammaticality, without the necessity of deriving or developing a model. By testing whether C (the overall complexity of empirical similarity-grammaticality pairs) differs significant ly from R (the complexity of a random mapping between similarity and grammaticality judgements), one can determine if, in principle, there is any structure or predictivity to be discovered. Because of the limited amount of data available [only 472 data points, each describing the behavior of one subject (of 16) on one particular novel pattern (of 31)], the similarity matrix and grammaticality endorsements were thermometer coded. Each number was replaced by a string of asterisks ('*') of length proportional to its magnitude. This coding scheme makes explicit the similarity structure assume d to exist between numerically similarity ratings. We expected this would facilitate compression in general, making the resulting estimates of complexity more reliable and sensitive to system regularity if it was there.

The set of similarity judgements was treated as the independent variable vector, the grammaticality judgements were treated as the dependent variable vector, and the pairings permuted (or rearranged and permuted) to yield estimates, as above, of C (R, respectively).

Simple statistical methods were applied to determine whether R was significantly larger than C. We estimated the value of C by taking the mean of 40,000 permutations of the data points, and compared this with an estimate of R calculated by taking 200 separate random mappings, and estimating their complexity 200 times each. This

yielded a sample of 200 "random" complexity estimates, from which we calculated means and deviations. Consistent with the findings of Pothos and Bailey, a one-tailed t-test yielded significant differences ($t(198) = 2.175$, $p < .025$). We conclude that there is a significant amount of information in the similarity judgements that can be applied to the grammaticality judgement task, and thus — without developing a model of how similarity determines grammaticality — we know that there is at least a partially regular mapping between the two.

Discussion

So why go to all this trouble of taking complexity estimates when a simple Nosofsky classifier will already show that similarity and grammaticality judgements are related? As hinted above, the Nosofsky classifier will capture different amounts of variance, depending upon the dimensions and metrics used. Furthermore, the generalized Nosofsky model is itself sensitive to particular aspects of similarity, derived from specific assumptions about the way similarity is determined. This is very useful from a theoretical perspective as it may tell us something about the way the human mind determines similarity. However, it will obviously not capture aspects of systematic regularity that do not fit those assumptions.

Part of the reason for applying the compressivity test might simply be economy of effort — rather than carefully developing and tuning a set of parameters to find the best model, complexity estimates can provide a simple, model-free description of how much regularity is in a particular data set to find. This can be used as an upper bound to compare with the regularities found under any given model, or more simply as a smoke test to determine whether regularities are there to find.

Another problem with the model-based analyses, of course, is the risk of picking the wrong model. Consider, for a moment, a hypothetical problem-space where the high-scoring test items are the ones of intermediate similarity from the training items. A physical example of this might be a model of judges' opinions of artwork or essays — an essay too close to any particular training datum could be regarded as derivative, or even plagiaristic, while an essay too far away from all training data is irrelevant, misguided, or simply unacceptable. However, a Nosofsky-style classifier developed as a model for this sort of judgement distribution would not describe it particularly well. Offhand, it's not clear what sort of well-understood and accepted model would capture this distribution. The regularity estimation technique developed above, by generalizing over all models, would provide a way of demonstrating the existence and degree, if not the type, of regularity.

More generally, the units in which Kolmogorov complexity are measured are the same "bits" corresponding to real-world yes/no questions, and can be regarded as a description not of how much variance there is, but how how predictable the variance is, and thus of how much work might be expected to be necessary to capture the variance. In this sense, one can in principle use this as a guide to determine what sort of models are overcomplex and overdetermined, and also as a well-unders tood unit for describing how overdetermined they are.

Another major advantage of this framework is that it provides a common currency for comparing different sorts of irregularity and complexity. The artistic example above, for instance, can be argued to be the superposition or of two (nested) judgements, one of originality, and a much broader one of relevance. Similarly, the data used in the first simulation above could be a very simple model of some sort of morphology. From a very structured vocabulary of fifty-five words, there are N "irregular" forms that are inverted upon inflection, while the rest are ("regular") identity mappings. Intuitively, one would consider a language like this, with two inflectional paradigms, to be less complex than a language with three or four separate paradigms. However, one would also consider a language with four very rare paradigms and one very common one to be more regular than a language with two equally common paradigms. This formalism, then, can provide a unified structure to examine questions such as the complexity of English past tense morphology, with its dominant "regular" (add "ed") paradigm and many semi-regular paradigms such as identity mappings, vowel changes, and so forth.

Future Work

One obvious future development for this work would be the development and adaptation of better compression or complexity estimation techniques; although *gzip* is asymptotically optimal, there are algorithms with faster convergence rates that might be applied to achieve more accurate estimates, especially on small data sets. Also, the *gzip* program is more sensitive to short-range (or adjacent) dependencies than to longer-distance relationships. Some more general framework [such as mutual information or Kc-complexity (Li & Vitányi, 1997)] may ultimately provide better results.

The thermometer coding introduced in the second simulation was intended to cue the compression algorithm to numerical structure which is not explicit in digital codings. As yet, it is unclear to what extent the resulting complexity estimates are dependent on assumptions built into a particular coding scheme.

Perhaps most importantly for practical applications, some normalization technique should be developed to address the issue of the data size; as developed here, a larger set of data points will almost certainly have a larger complexity than a smaller set on the same function. This is intuitive in that the data points themselves should add complexity, and also that adding data points constrains the function further (and thus makes it more complex).

Conclusions

Kolmogorov complexity can be regarded as the ultimate measure of the degree of (lack of) structure of a given symbol-string, even when one particular model or approach can find no structure. Despite the computational intractability, it can be reliably and quickly estimated by any standard compression technique. We present here a

method for applying Kolmogorov complexity to capture the notion of system regularity as a unified measure for the computational — and, as has been suggested (Chater & Hahn, 1997), cognitive — difficulty of a particular system.

In particular, we have applied this technique in two small-scale simulations. The results easily and clearly capture standard intuitions and psychological results. The strong differences in the areas measured (on the one hand, complexity and regularity of a functional transformation, and on the other, the degree of predictability and regularity captured by a particular set of data) indicate that this technique can be widely and easily applied as a model-free method of determining what sort of performance can be expected from any (or the optimal) model before actually developing the model. It further provides a fundamental, theory-neutral framework for describing the difficulty and complexity of a task and can be used as common ground for comparing widely differing tasks in a controlled environment — for example, comparing the degree of irregularity present in past tense acquisition vs. grapheme to phoneme acquisition, both of which have been presented as quintessential complex, irregular systems. Finally, the units of measurements, bits, are themselves meaningful and can be directly related to real-world behavioral and information-theoretic data.

Further work is required to sharpen the tools used to measure and to determine where the limits of application are. Our results, however, demonstrate a novel and powerful technique for the measurement and comparison of system regularity in a general, model-free, and theory-neutral environment.

Acknowledgements

Patrick Juola was supported by grant number 70 from the ESRC. Todd Bailey was funded by a grant from the McDonnell-Pew Centre for Cognitive Neuroscience. Emmanuel Pothos was supported by the UK MRC (reference number: G78/4804), the Bodossaki Foundation, and the A. S. Onassis Foundation (reference: Group S-076/1996-1997).

References

Chaitin, G. J. (1995/1996). A new verson of algorithmic information theory. *Complexity, 1*(4), 55–9.

Chater, N., & Hahn, U. (1997). Representational distortion, similarity, and the Universal Law of Generalization. In *Proceedings of the interdisciplinary workshop on similarity and categorization (SimCat 97)* (pp. 31–36). University of Edinburgh.

Juola, P. (1997). A numerical analysis of cultural context in translation. In *Proceedings of the second European conference on cognitive science* (pp. 207–210). Manchester, UK.

Juola, P. (in press). Measuring lingusitic complexity : The morphological tier. *J. Quantitative Linguistics.*

Li, M., & Vitányi, P. (1997). *An introduction to Kolmogorov complexity and its applications* (2nd ed.). New York: Springer.

Nosofsky, R. M. (1992). Similarity scaling and cognitive process models. *Annual Review of Psychology, 43*, 25–53.

Plaut, D., McClelland, J., Seidenberg, M., & Patterson, K. (1996). Understanding normal and impaired word reading computational principles in quasi-regular domains. *Psychological Review, 103*(1), 56–115.

Pothos, E. M., & Bailey, T. M. (1997). Rules vs. similarity in artificial grammar learning. In *Proceedings of the interdisciplinary workshop on similarity and categorization (SimCat 97)* (pp. 197–203). University of Edinburgh.

Reber, A. S. (1967). Implict learning of artificial grammars. *Journal of Verbal Learning and Verbal Behavior, 6*, 855–63.

Wolpert, D. H., & Macready, W. G. (1995). *No free lunch theorems for search* (Tech. Rep. No. TR 95-02-010). Santa Fe Institute.

Ziv, J., & Lempel, A. (1977). A universal algorithm for sequential data compression. *IEEE Transactions on Information Theory, IT-23*(3), 373–343.

Ziv, J., & Lempel, A. (1978). Compression of individual sequences via variable rate coding. *IEEE Transactions on Information Theory, IT-24*(5), 530–536.