Evolving constraints and rules in Harmonic Grammar*

Elliott Moreton
University of North Carolina, Chapel Hill
moreton@unc.edu

Abstract

An evolutionary model of pattern learning in the MaxEnt OT/HG framework is described in which constraint induction and constraint weighting are consequences of reproduction with variation and differential fitness. The model is shown to fit human data from published experiments on both unsupervised phonotactic (Moreton et al., 2017) and supervised visual (Nosofsky et al., 1994) pattern learning, and to account for the observed reversal in difficulty order of exclusive-or vs. gang-effect patterns between the two experiments. Different parameter settings are shown to yield gradual, parallel, connectionist- and abrupt, serial, symbolic-like performance.

1 Introduction

Some constraints in natural-language grammars must be induced from phonological data, such as constraints which refer to specific lexemes, (e.g., McCarthy and Prince 1993; Fukazawa 1999; Pater 2000; Ota 2004; Pater 2007; Coetzee and Pater 2008; Pater 2009; Becker 2009), to specific lexical strata, inflectional paradigms, or other language-particular lexical classes, (e.g., Benua 1997; Alderete 1999; Ito and Mester 2001; Flack 2007a; Inkelas 2008), or to phonetically arbitrary sound classes that do not recur across languages (e.g., Bach and Harms 1972; Anderson 1981; Buckley 2000), as well as those which enforce idiosyncratic requirements (e.g., Prince and Smolensky 1993, 101). [1]

How and when are phonological markedness constraints induced? Proposals in the Optimality Theory/Harmonic Grammar literature fall into two main categories: *exhaustive search*, in which the learner considers all of a set of possible constraints, keeping those that best satisfy criteria (Hayes and Wilson, 2008; Wilson and Gallagher, 2018), and *error-patching*, in which the learner identifies a particular error type and makes a constraint against it (Adriaans and Kager, 2010; Pizzo, 2013; Pater, 2014).[2]

Here we discuss an alternative, *evolution*. Evolution-based algorithms are attractive because they are both an established technology for efficiently searching large, inconveniently-shaped hypothesis spaces (Bäck, 1996; Eiben and Smith, 2003; De Jong, 2006), and the basis of a leading account of human creativity in art, engineering, science, and other domains (Campbell, 1960; Simonton, 1999; Dietrich and Haider, 2015). In the specific model considered here, Winnow-MaxEnt-Subtree Breeder, constraints interact via Max Ent Harmonic Grammar (Goldwater and Johnson, 2003), but weights are population sizes, weight update is population growth or shrinkage in response to fitness-based selection, and constraints are innovated via mutation and recombination.

The paper is structured as follows. §2 describes the model (the "Winnow-MaxEnt-Subtree Breeder"). §3 illustrates some of its properties using a simplified "toy" example (Simulation 1). §4 quantifies a necessary condition for learnability in terms of the learning rate, the mutation rate, and the number of critical constraints. §§5 and 6 illustrate how the model accounts for human data from two published experiments which tested formally analogous patterns but found very different

*The author is indebted for comments and suggestions to Katya Pertsova, Jennifer Smith, participants in the UNC-Chapel Hill P-Side caucus, and three anonymous SCiL reviewers. The research was supported in part by NSF BCS 1651105, "Inside phonological learning", to E. Moreton and K. Pertsova.

[1] Constraint induction from phonetics is a separate issue, and is not addressed here; see, e.g., Hayes 1999; Smith 2002; Flack 2007b.

[2] A learner using positive rather than negative constraints can identify correct forms and make constraints that reward them (Boersma and Pater, 2007).

results, the unsupervised phonotactic learning of Moreton et al. (2017) and the supervised visual pattern learning of Nosofsky et al. (1994). Appropriate parameter settings cause the model to act in the first case more like a connectionist net (e.g., Gluck and Bower 1988b,a) and in the second case more like a serial, rule-based hypothesis-tester (e.g., Nosofsky et al. 1994; Ashby et al. 2011; Goodwin and Johnson-Laird 2013). §7 suggests further empirical tests of the model.

2 Winnow-MaxEnt-Subtree Breeder

The anatomy of Winnow-MaxEnt-Subtree Breeder will be briefly described here. It is based on a model described in Moreton (2010b,a,c) and analyzed in Moreton (2019), which it modifies and extends.[3]. Source code and a replication kit can be found at `https://users.castle.unc.edu/~moreton/Software/SCiL2020ReplicationKit/`.

2.1 Constraints and candidates

Consubstantiality of candidates and constraints. Candidates are represented using prosodic and Feature-Geometric trees familiar from existing phonological theory (Goldsmith, 1976; McCarthy, 1981; Sagey, 1990; Clements and Hume, 1995) — in this paper, a slightly simplified version of the feature system in Gussenhoven and Jacobs (2005, Ch. 5). A box marks the head; L and R mark left and right constituent boundaries. A constraint is a representational subtree, rooted at a PrWd, which describes a locus of violation (or satisfaction depending on the polarity of the constraint). Figure 1 depicts a micro-constraint that implements ONSET, à la Smith (2006). Any notational variant of this micro-constraint would belong to the same macro-constraint.

Meta-constraints. Since constraints are consubstantial with representations, they can evaluate each other. Winnow-MaxEnt-Subtree Breeder therefore allows the user to define metaconstraints, constraints which award a fitness bonus or penalty to other constraints. These can be used to prevent ill-formed constraints (e.g., *[+high][+low]), or to gently encourage or discourage constraints of particular types (e.g., those that mention "salient" features, or express particular phonetic principles).

[3]Erratum for that SCiL paper: p. 5, below Eqn. 35, "$\geq \log x$" should be "$\approx \log x$".

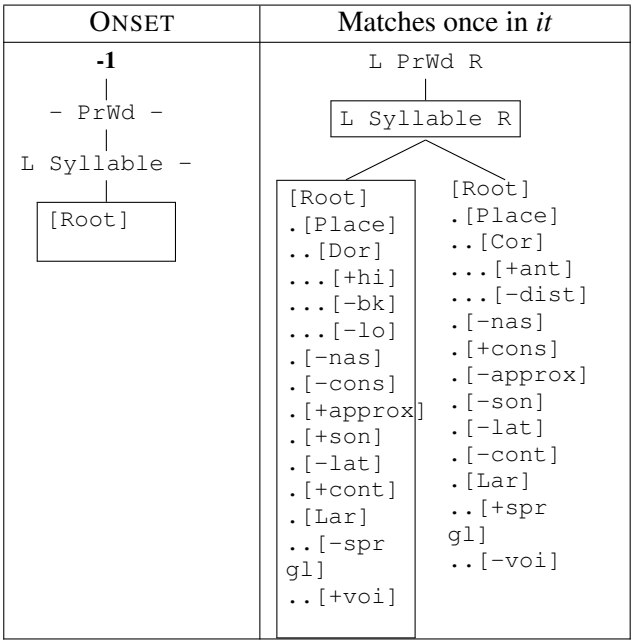

Figure 1: A constraint uses a subtree to describe a locus of violation.

2.2 Learning constraint "weights"

Weights are population sizes. In a Harmonic Grammar framework (Legendre et al., 1990), we can, without changing candidate harmonies, replace any constraint C_i of weight w_i with w_i/ζ constraints that each contribute ζ to harmony. For example, if $\zeta = 0.01$, a MAX constraint of weight 3.5 can be replaced by 350 micro-MAX's, each of which has weight 1 and whose marks are multiplied by 0.01. In Winnow-MaxEnt-Subtree Breeder, all constraints are micro-constraints of fixed weight 1. The harmony of a candidate x is

$$h(x) = \sum_c \zeta c(x) \tag{1}$$

Luce/MaxEnt choice rule. Given the experimenter's intended winner x^+ and intended loser x^-, the learner chooses x^+ with a probability that depends on the harmonies of the candidates.

$$\Pr(x^+ \mid x^+, x^-) = \frac{e^{h(x^+)}}{e^{h(x^+)} + e^{h(x^-)}} \tag{2}$$

This is the two-alternative Luce choice rule (Luce, 1959, 23) applied to the exponentiated harmonies, i.e., a conditional Maximum Entropy model (Goldwater and Johnson, 2003; Jäger, 2007; Hayes and Wilson, 2008). The generalization to k alternatives is straightforward. The total harmony available in the system is thus $N\zeta$, limiting performance.

Macro-constraints. The algorithm itself is cognizant only of micro-constraints. For analytic

d_i	Favors	Expected offspring o_i	Effect on population of $[c_i]$
-1	loser	$1/(1+\eta) < 1$	shrinks
0	neither	1	stays same
$+1$	winner	$1+\eta > 1$	grows

Table 1: Effect of error on offspring of micro-constraint and population of macro-constraint.

convenience, we, looking in from outside, can classify two micro-constraints c_i, c_j as belonging to the same *macro-constraint* if they assign the same scores to all candidates in the representational space. In the example above, the 350 micro-MAX's belong to a macro-constraint with a population size of 350 and an effective weight of 3.5. Macro-constraint membership is an equivalence relation, so we can write $[c_i]$ for the macro-constraint containing the micro-constraint c_i.

Weight update is reproduction. When an error occurs, each micro-constraint c_i produces an expected number of offspring given by

$$o_i = (1+\eta)^{d_i} \tag{3}$$

where η is a learning-rate parameter and $d_i = c_i(x^+) - c_i(x^-)$ is the difference between the winner's and loser's score on c_i. (The quantity o_i is the *fitness* of c_i.) In particular, c_i produces $\lfloor o_i \rfloor$ offspring with certainty, and one more with probability $o_i - \lfloor o_i \rfloor$. E.g., if c_i is binary (awards 0 or 1 marks), then Table 1 shows the expected number of offspring of the micro-constraint and the effect on the population size of the macro-constraint. This update rule induces a variant of the Winnow-2 algorithm (Littlestone, 1988; Moreton, 2019), first mentioned as a possible HG learning algorithm by Magri (2013).

If "soft" meta-constraints (those that assign finite marks) are specified, they add an offset to o_i equal to ζ times the total score they assign to c_i.

2.3 Evolving constraints

The initial constraint population is set by the user. Thereafter, on each error, the population is completely replaced via the following procedure.

Breeding. For each micro-constraint c_i in the pre-error population P, $o_i \cdot s$ identical clone offspring are made and deposited in the reproductive population R. Here s is the "clutch size" parameter, 1 by default, which allows the absolute number of offspring to be varied while maintaining the relative proportions belonging to differently-fit parents.

Recombination. Of the constraints in R, $\lfloor \sigma \cdot |R| + 0.5 \rfloor$ are randomly selected to be *recombinant* breeders, partitioning R into B (recombinant breeders) and $R - B$ (parthenogenetic breeders). The offspring population O is initialized to equal $R - B$. For each breeder $c_i \in B$, another breeder $c_j \in B$ of equal or greater fitness ($o_i \leq o_j$) is selected, and the two constraints are combined as described in Moreton (2019) to make a new constraint $c_{i,j}$, which is then added to O. (Recombination is not used in the simulations described in this paper.)

Mutation. Of the constraints in O, $\lfloor \mu |O| + 0.5 \rfloor$ are randomly selected to undergo mutation. Mutation is undirected, i.e., the probability of a particular mutation is independent of its effect on fitness (just as in the Minimum Description Length learner of Rasin and Katzir 2016). Mutation proceeds recursively, starting with the highest node in the constraint. Mutation operations differ between node genera (Table 2). At each node, every operation that can apply to that node first has a chance to apply. Then the algorithm visits each actual *or potential* dependent of the node, and applies recursively to it. A potential dependent of a unary feature is any currently unrealized dependent feature; e.g., an unfilled [ant] slot under [+Cor]. A potential dependent of a prosodic category is an interval between two of its actual constituents, counting the category's own boundaries as constituents. For example, the PrWd in $[\sigma\sigma]_{\text{PrWd}}$ has two actual dependents (the two σs) and three potential ones: $[\underset{\smile}{\ }\sigma\underset{\smile}{\ }\sigma\underset{\smile}{\ }]_{\text{PrWd}}$. Mutation could add another σ node at any or all of the three potential dependents.

After mutation has applied to a constraint, the mutant and the original are compared, and if they are identical, or if the mutant receives marks from a "hard" meta-constraint (one that assigns marks of $-\infty$), mutation is re-attempted until an actual mutant is achieved. The number of mutants produced on each error is thus $Ns\mu$.

The probability of each operation can be set individually. In the present simulations, all are set to the same probability π, except those for *Gain head*, *Lose head*, and *Duplicate constituent*, which are set to 0. The larger π is, the more the mutant will differ from the parent.

A micro-constraint which is lost from the population and later re-innovated returns with its old fitness value, rather than the default fitness of 1 given to novel micro-constraints. (This design choice is

Invert polarity: Change the sign of the mark given by a constraint.

Add constituent: Applied to a potential dependent in a PrWd (syllable), adds a syllable node (segment node) there. (E.g., $[\sigma\sigma]_{\text{PrWd}}$ has three potential dependents, marked here with $\smile$s: $[\smile\sigma\smile\sigma\smile]_{\text{PrWd}}$. Each $\smile$ could mutate into another syllable.)

Delete constituent: Applied to a syllable node (segment node), deletes it.

Duplicate constituent: Applied to a syllable or segment, makes an adjacent duplicate copy of the syllable or segment, including all of its dependents.

Gain head: Applied to a PrWd (syllable), designates one of its syllables (segments) as the head, or moves the head if there already is one.

Lose head: Applied to a PrWd (syllable), makes it headless by undesignating the existing head (if any)

Flip anchor: Applied to a prosodic boundary marker, toggles it (between – and `L`, or between – and `R`).

Gain unary: Applied to a *potential* unary feature (e.g., the empty position under a [+Place] node where [+Cor] could go), adds that unary feature.

Lose unary: Applied to an *actual* unary feature, deletes it along with all of its dependents.

Gain binary: Applied to a *potential* binary feature (e.g., the empty position under a [+Cor] node where [±ant] could go), adds that feature (with + and − values equally likely).

Lose binary: Applied to an *actual* binary feature, deletes it.

Invert binary coefficient: Applied to an *actual* binary feature, changes + to − and vice versa.

Table 2: List of mutation operations.

crucial to the success of Simulation 3 in §6.)

Memorization. With probability p_{mem}, the learner creates a new micro-constraint that gives $+1$ mark to the candidate that should have won, or -1 mark to the candidate that should not have (the experimenter can set a switch, `mem_polarity`). This constraint is cloned n_{mem} times, and the clones are added to O. (In all simulations in this paper, $p_{\text{mem}} = 0$.)

Population adjustment. The resulting offspring population is adjusted in size to meet the target size of N. The default method (*random adjustment*) is to randomly delete or clone micro-constraints, with equal probability. An alternative (*fitness-based adjustment*) is to choose the fittest N offspring, with ties broken randomly. The adjusted population then completely replaces the previous generation.

The parameters are listed in Table 3. In all the simulations reported here, the parameters were fixed across trials within a simulation, although in fact they can be varied from trial to trial.

N	Number of micro-constraints in population.
ζ	Weight quantum.
η	Learning rate.
μ	Mutation rate.
s	Clutch size.
p_{mem}	Probability to memorize winner/loser as constraint.
n_{mem}	Number of copies of winner/loser memorized.
`mem_polarity`	Memorize winner or loser?
`meta`	Meta-constraint set
`mut`	Mutation probabilities (see Table 2)
`rec`	Recombination parameters (not discussed here)

Table 3: List of simulation parameters.

3 Simulation 1: 2AFC phonological learning (toy example)

Since new macro-constraints arise by mutation out of old ones, existing macro-constraints should prime discovery of new ones that are similar to them. Since high-weighted (populous) macro-constraints initiate more mutations, new macro-constraints should tend to be mutants of (hence, similar to) high-weighted old ones. And because approximate solutions can prosper when the learner has not yet discovered the precise constraints, an approximately-right constraint can focus the learner's mutational searching on its own neighborhood.

We illustrate these general principles of the model's behavior using a stripped-down "toy" example. The stimulus space is the set of all $(C)V(C)$ where C is one of /p, t, k/ and V = /u/ . Pattern A has two place restrictions on the coda; Pattern B has one on the coda and one on the onset (Table 4).

To make analysis easier, σ is set to 0 to make all reproduction asexual (this is true througout this paper). The mutation distance between the critical constraints in Condition A is then 2 (from *[–syll, +Lab]]$_\sigma$ to *[–syll, +Dor]]$_\sigma$: delete [+Lab],

	Pattern A
Unviolated constraints	*[–syll, +Dor]]$_\sigma$ (=NoDorCoda) *[–syll, +Lab]]$_\sigma$ (=NoLabCoda)
Positive	u, ut, pu, put, tu, tut, ku, kut
Negative	up, uk, pup, puk, tup, tuk, kup, kuk
	Pattern B
Unviolated constraints	*[–syll, +Dor]]$_\sigma$ (=NoDorCoda) *$_\sigma$[[–syll, +Lab] (=NoLabOns)
Positive	u, up, ut, tu, tup, tut, ku, kup, kut
Negative	uk, pu, pup, put, tuk, kuk

Table 4: Phonotactic patterns for Simulation 1.

	Discovery of		
	*[–syll, +Dor]]$_\sigma$	*$_\sigma$[[–syll, +Lab] *or* *[–syll, +Lab]]$_\sigma$	Abs. diff.
A	237	243	114
B	313	316	322

Table 5: Median trials to and between discovery of critical constraints in Simulation 1, Conditions A vs. B.

insert [+Dor]), while that between those in Condition B is 4 (from *[–syll, +Lab]]$_\sigma$ to *$_\sigma$[[–syll, +Dor]: delete [+Lab], insert [+Dor], unset right boundary, set left boundary). The same holds for other micro-constraints that instantiate these macro-constraints, because they likewise occur in pairs (e.g., with a useless [+nas] feature added to both). Discovering either critical constraint should therefore prime discovery of the other better in the A condition than in the B condition. Concretely, we expect that in Condition A, as compared to Condition B, (1) time between discovery of the two constraints will be smaller, and (2) the weights of the two constraints will be more strongly correlated (because they co-exist for longer).

The simulation parameters were set as follows: learning rate $\eta = 0.25$, mutation rate $\mu = 0.05$, a population of $N = 200$ constraints initialized to `*(L PrWd R)`, weight quantum $\zeta = 0.05$. The individual probabilities of the mutation operations *Add constituent, Delete constituent, Flip anchor, Gain unary, Lose unary, Gain Binary, Lose binary, Invert binary coefficient* were set to $\pi = 0.005$, and all the others to 0. The time limit was 1024 trials, and 100 replications of each condition were run. Non-discovery was coded as ∞, so aggregate results are reported as medians, not means.

Prediction (1): Time between discovery smaller in A than B: The median number of trials that elapsed between discovery of the two critical constraints was 2.8 times greater in Condition B than in Condition A, as shown in Table 5. The difference was significant by a Wilcoxon-Mann-Whitney rank-sum test ($U = 2657.5, p = 0.003082$, using `wilcox.test` in R's `stats` library, R Core Team 2018).

Prediction (2): Weights of the two constraints more strongly correlated in A than B: Because discovery is more simultaneous in Condition A, the critical macro-constraints' weights develop more asymmetrically in Condition B. The mean correlation between the weights of *[–syll, +Dor] and the other critical macro-constraint was 0.72 in Condition A, 0.56 in Condition B (significantly different by a Wilcoxon-Mann-Whitney rank-sum test, $U = 4761, p = 0.0003484$. Non-discovery meant no correlation could be computed for 5 of the A and 24 of B simulations.).

Attention-like effects: Clues in the data can cause the learner to search some regions of constraint space more intensively. Here, the constraint *[–syll, +Dor]]$_\sigma$ (i.e., NoDorsalCoda, critical in A and B conditions) is one mutation away from *[–syll]]$_\sigma$ (i.e., NoCoda). The latter constraint is discovered early and simultaneously in both A and B (see Table 6). It is better supported by the training data in A (4 out of 8 positive vs. 0 out of 8 negative stimuli) than in B (3 out of 8 positive vs. 1 out of 8 negative). Once discovered, its population grows for longer in A than in B, peaking at 59 micro-constraints on Trial 305 vs. 23 micro-constraints on Trial 260. Between discovery and peak, the *[–syll]]$_\sigma$ population grew at a rate of $59/(305 - 24) = 0.21$ micro-constraints per trial in Condition A, but only $23/(259 - 24) = 0.10$ in Condition B, i.e., half as fast. More population in *[–syll]]$_\sigma$ means more opportunities to spawn *[–syll, +Dor]]$_\sigma$, and indeed that constraint is found sooner in Condition A (estimate is 72 trials by Wilcoxon-Mann-Whitney test, $U = 2657.5, p = 0.003082$). Across all 99 replications in Condition A in which both constraints were discovered, a mean of 47% of all instances of *[–syll, +Dor]]$_\sigma$ were immediate offspring of *[–syll]]$_\sigma$. The analogous figures for Condition B are 91 and 8.7%.

Speaking anthropomorphically, we might say that the A learner "notices" that codas matter, i.e. up-weights *[–syll]]$_\sigma$. That "directs its attention" to the coda position (by allowing the approximate solution *[–syll]]$_\sigma$ to elbow out other constraints). This "focused attention" results in a more-intensive search among neighbors of *[–

syll]]$_\sigma$, which soon finds *both* critical constraints. Thus, R&D work that helps find one constraint also helps find the other. The critical constraints then outcompete the approximate constraint and drive its weight down. In the B condition, it takes longer to discover the critical constraint because the mutant population is divided between constraints targeting the onset and coda positions, i.e., the data does not "call attention" to one position more than the other.

	Median trial number	
Event	A	B
*[–syll]]$_\sigma$ discovered	24	24
*[–syll]]$_\sigma$ population peaks	305	260
(peak pop. size)	(59)	(23)
*[–syll, +Dor]]$_\sigma$ discovered	237	312

Table 6: Discovery of *[–syll]]$_\sigma$ (NoCoda) primes discovery of *[–syll, +Dor]]$_\sigma$ (NoDorsalCoda) in Simulation 1.

4 Mutation, learning, and complexity in a monostratal grammar

The pattern in Simulation 1 can be captured by a monostratal grammar: The two macro-constraints handle disjoint, exhaustive subsets of the pattern, and are not critically ranked (weighted) relative to each other. In the general monostratal case, there are n critical macro-constraints in the minimal solution, with $[c_k]$ having exclusive responsibility for Trial Type k. Suppose that the learner has already found them all, and that ζ and N are big enough that growth in the population of any critical macro-constraint comes mainly at the expense of non-critical constraints (assumed to be neutral). We will see that η and μ impose an upper bound on n.

Let r_k be the probability that when the next error occurs, it will occur on Trial Type k. Then the expected proportional change in the population size w_k of $[c_k]$ is the expected product of its rates of growth through reproduction and of shrinkage through mutation. If we assume what is typically the case, that mutation turns a critical constraint into another critical constraint negligibly often, then on the next error, $[c_k]$ reproduces with probability r_k and then shrinks by mutation with probability 1:

$$\begin{aligned} E\left[w'_k/w_k\right] &= r_k(1+\eta)(1-\mu) + (1-r_k)(1-\mu) \\ &= (1+r_k\eta)(1-\mu) \end{aligned} \tag{4}$$

where w'_k is the updated w_k.

When the learning algorithm converges, $E\left[w'_k/w_k\right] \geq 1$ for all k, i.e., all of the macro-constraint weights are either constant, or else increasing at the expense of the neutral constraints. Setting $E\left[w'_k/w_k\right] = 1$ and solving for r_k yields the critical value

$$r^* = \frac{1}{\eta}\frac{\mu}{1-\mu} \tag{5}$$

If $r_k < r^*$, then $w'_k < w_k$. Hence, a necessary condition for convergence is $\forall k : r_k \geq r^*$. But since the r_k's add to 1, there must be at least one k such that $r_k \leq 1/n$. Hence a stable final grammar exists only if

$$n \leq n_{\text{crit}} = \eta\frac{1-\mu}{\mu} \tag{6}$$

In Simulation 1, η and μ were chosen so that $n_{\text{crit}} = 0.25 \cdot (1-0.05)/0.05 = 4.74 > 2 = n$, and indeed, the average proportion correct for the last 16 trials was above 0.95 in both the A and B conditions. To illustrate the effect of varying n_{crit}, the simulation was re-run with all combinations of $\eta \in \{0.1, 0.15, 0.25, 0.3\}$ and $\mu \in \{0.025, 0.05, 0.1, 0.15\}$. Figure 2 shows the results in terms of proportion correct on the last 16 trials (of 2048). For $n_{\text{crit}} > 2$, the median — indeed, the lower quartile — is never below 0.9. For n_{crit} even slightly below 2, performance drops off rapidly.

The reproduction and mutation rates thus fix an upper bound on the number of critical macro-constraints in a learnable monostratal grammar. A pattern which minimally requires more than

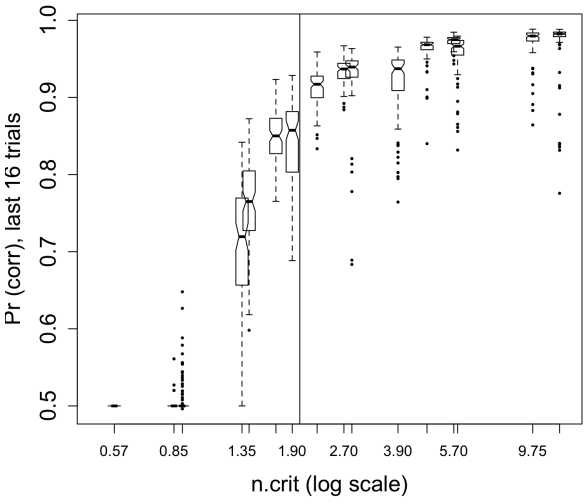

Figure 2: Proportion correct on the last 16 trials as a function of n_{crit}, Simulation 1, Condition A. Vertical line marks $n_{crit} = 2$.

n_{crit} macro-constraints cannot be learned at all. A pattern which can be expressed with n_{crit} or fewer macro-constraints cannot be learned using an equivalent monostratal grammar that has more, e.g., one relying on parochial constraints or stimulus memorization.

5 Simulation 2: Unsupervised phonological learning (Moreton et al., 2017)

When the population size N is large, and the weight quantum ζ is small, the learner approximates a constraint-based model in which the constraint set contains all possible constraints up to a certain size, whose weights vary continuously. The reason is that the mutants created on any error will sample the space of possible constraints densely. Simulation 2 illustrates this point.

In many lab experiments, phonotactic learning is *unsupervised*: Participants are trained by exposure to pattern-conforming stimuli only. Since Winnow-MaxEnt learns from winner-loser pairs, the learner must somehow generate its own loser on each trial.

A straightforward way to do that is for the learner to sample from the probability distribution specified by its current grammar. If the sample differs from the presented stimulus (virtually certain, regardless of how well the pattern has been learned), the stimulus and sample are used as x^+ and x^- in Equation 3. Since x^+ is always pattern-conforming, but x^- is sometimes not, macro-constraints enforcing the pattern prosper (i.e., gain population relative to other constraints).

The hypothesis is tested by simulating three different conditions from a published experiment (Moreton et al., 2017, Exp. 1). The stimulus space consisted of the 256 possible $C_1V_1C_2V_2$ stimuli for which the consonants were one of [t d k g] and the vowels one of [i æ u ɔ] . Human participants were familiarized by hearing and repeating aloud 32 pattern-conforming stimuli in pseudo-random order such that each stimulus occurred 4 times. They then did 32 test trials in which they heard two novel stimuli, one pattern-conforming and one not, and were asked to choose the conforming stimulus.

Three specific patterns were chosen for the simulation, each instantiating a different pattern type in the classification of Shepard et al. (1961, see Figure 3). The pattern "C_1 is voiceless" belongs to Type I, a simple, one-feature affirmation. The pattern "C_1 and C_2 disagree in voicing" is of Type II, an if-and-only-if (equivalently, an exclusive-or) combination of two features. Finally, the pattern "at least two of: (1) C_2 is velar, (2) C_1 is voiceless, (3) V_2 is back" is of Type IV, a three-feature "gang effect".

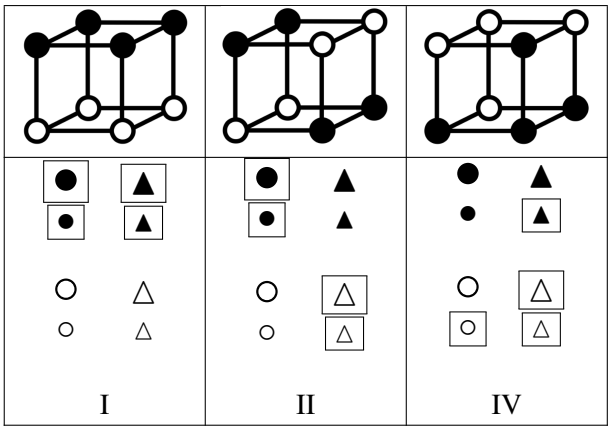

Figure 3: Pattern Types I, II, and IV of Shepard et al. (1961), illustrated using visual stimuli. Type I is defined by a single feature ("the figure is black"); Type II is an iff/xor relation between two features ("black iff round"); and Type IV is a three-feature gang effect ("at least two of white, triangular, small").

For each pattern, 32 conforming training stimuli, 32 conforming test stimuli, and 32 non-conforming test stimuli were randomly chosen. Each of the three patterns can be learned to perfection with $n = 8$ or fewer macro-constraints. The simulation parameters were set at $\eta = 0.33, \mu = 0.025$ (satisfying Equation 6 for $n = 8$), $\zeta = 0.05$, and $N = 2000$ constraints. The values were chosen by trial and error to approximate human performance. The test task for human participants was to decide which of each test pair was "a word in the language you were studying". In the simulation, this was implemented by attaching to each training and test stimulus a `[+real]` feature. The constraint set was initialized to equal proportions of `*[+real]` and `*[-real]`. The learner got as many training and test trials as did the humans. 100 replications of each simulation were run.

Simulation results are shown in Table 7 alongside human performance. The numbers are similar, and the proportion of pattern-conforming test-phase responses decreases in the order $I > IV > II$.

	Pattern type		
	I	II	IV
Sim.	0.83 ± 0.13	0.48 ± 0.02	0.60 ± 0.05
Human	0.73 ± 0.12	0.57 ± 0.11	0.70 ± 0.09

Table 7: Proportion pattern-conforming responses in the test phase ($\pm$ 1 s.d., not s.e.m.) for Simulation 2 and human data (Moreton et al., 2017, Table 5), showing $I > IV > II$ order.

6 Simulation 3: Supervised visual learning (Nosofsky et al., 1994)

When the population size N is small and the weight quantum ζ is large, the Winnow-MaxEnt-Subtree Breeder approximates a serial hypothesis-tester that keeps trying one categorical rule after another until it finds one that works. This is illustrated in Simulation 3.

The human experiment to be replicated is that of Nosofsky et al. (1994). The stimulus space consisted of eight geometric figures varying on three binary dimensions (shape, shading, and size, as in Figure 3). A pattern was an assignment of four stimuli to Category A, and four to B. On each trial, the participant saw a figure, classified it as A or B, and received right/wrong feedback. Training continued until the participant had responded correctly on 32 consecutive trials, or reached a limit of 400 trials. The difficulty order, in terms of trials to criterion or errors to criterion, was $I < II < IV$.

Many hypothesis-testing models in the concept-learning literature account for this difficulty order by positing a bias towards syntactically-simple hypotheses (Shepard et al., 1961; Nosofsky et al., 1994; Feldman, 2006; Ashby et al., 2011; Goodwin and Johnson-Laird, 2013). The bias in Winnow-MaxEnt-Subtree Breeder has a different origin.

It can be seen from Figure 3 that a correct grammar of the Type I problem can be made with just two macro-constraints: `*[-wug][+black]` and `*[+wug][-black]`. These constraints designate the top face of the cube as a wug (i.e., pattern-conforming) and the bottom face as a non-wug. The smallest correct Type II grammar needs four constraints, one for each of the back-to-front edges of the cube (e.g., `*[-wug][+black][+circle]`). The smallest correct Type IV grammar needs six constraints, one for each of the edges radiating from the central wug or non-wug stimulus.[4] A small N should therefore favor Type I over Type II, and Type II over Type IV. For the grammar to give human-like near-categorical responses with so few constraints, the weight quantum ζ must be large, so that each constraint has the effect of a categorical rule.

The parameters for Simulation 3 were adjusted by trial and error to the values $N = 7, \zeta = 12, \eta = 1, \mu = 1, \pi = 1/2$. Clutch size was set to 12. Fitness-based selection was turned on so that the fittest N of the offspring were chosen. The high mutation rate and large clutch size should have the effect of making the offspring population be a diverse random sample of the 54 possible constraints. Any micro-constraint in the sample which has previously been seen to favor a loser will be assigned its previous (negative) fitness, and hence be eliminated from the offspring set by fitness-based selection. (Here is where the learner's memory for the fitness of extinct micro-constraints, mentioned above in §2.3, is crucial.) The result should be that, as the simulation progresses, invalid micro-constraints are gradually discovered and permanently eliminated from consideration, so that the offspring population becomes more and more a random sample of size 7 from the valid constraints.

In the Type I condition, there are 2 valid faces, 8 valid edges, and 8 valid corners, and a correct grammar can be made in many ways: from the 2 faces, from 1 face plus 4 edges, from 1 face plus 3 edges plus 2 corners, etc. In the Type II condition, there are 4 valid edges and 8 valid corners, and a correct grammar can be made from the 4 edges, or 3 edges plus 2 corners, or 2 edges plus 4 corners. In the Type IV condition, there are 6 valid edges and 8 valid corners, and a correct grammar can only be made from the 6 edges, or from 5 edges plus 2 corners, or from 2 faces plus 2 copies of each of 2 corners. Hence a random sample of size $N = 7$ is more likely to solve Type I than Type II, and Type II than Type IV.

The results of the simulation (100 replications) are shown in Table 8. The order of difficulty is the same for the learner as for the humans (who are about 40% faster in all conditions). Changing the model parameters has caused Types II and IV to change places with respect to Simulation 2. Smaller values of N amplify the advantage of

[4] Alternatively, Type IV can be expressed with two face constraints, plus two copies of each of two corner constraints to override the face constraints, which is still six constraints.

	% participants reaching criterion			Mean trials to criterion		
	I	II	IV	I	II	IV
Sim.	100	98	74	68	161	210
Human	100	100	100	44	85	127

Table 8: Attainment of criterion performance (32 consecutive correct responses in 400 trials) for Simulation 3 and human participants (Nosofsky et al., 1994, 356). Mean trials to criterion excludes cases where criterion was not reached. There were 100 replications.

Type II over Type IV. For $N \leq 5$, no Type IV simulations reach criterion.

7 Discussion

The Winnow-MaxEnt-Subtree Breeder links phonological learning theoretically with other kinds of pattern learning and with creativity in other domains, thus spawning future research questions (e.g., whether mutation is undirected, or sensitive to the demands of the problem; Simonton 1999; Dietrich and Haider 2015; whether recombination — sexual reproduction — is empirically motivated, etc.). A more immediate task is to test its empirical adequacy for phonological learning. This section suggests some places to start.

Abruptness. The learning curve in the large-N/small-ζ case is predicted to be more abrupt when the pattern depends on induced constraints rather than preexisting ones from UG or L1 (Moreton, 2019). Complex patterns require a high learning rate η and/or low mutation rate μ (see §4). Lower μ means longer intervals between constraint discoveries, while higher η means faster population growth following discoveries; hence, complex patterns are predicted to be learned as a series of sudden acquisitions of individual subpatterns. I know of no experimental evidence bearing directly on either prediction, but abruptness is a familiar aspect of first-language acquisition ("across-the-board" changes, e.g., Smith 1973; Macken and Barton 1978; Vihman and Velleman 1989; Barlow and Dinnsen 1998; Levelt and van Oostendorp 2007; Gerlach 2010; Becker and Tessier 2011; Guy 2014), and been observed in lab-learned phonology (Moreton and Pertsova, 2016). Individual learning curves for many complex non-linguistic skills show discontinuities alternating with gradual power-law improvements (Haider and Frensch, 2002; Bourne, Jr. et al., 2010; Gray and Lindstedt, 2017; Donner and Hardy, 2015).

Priming. As seen in Simulation 1, a target grammar in the large-N/small-ζ case is found sooner when the relevant macro-constraints are separated by fewer mutations, because finding one constraint generates mutants that are helpful in finding the next. The acquisition of a constraint thus primes acquisition of similar constraints. It may be relevant that, in a sample from P-Base (Mielke, 2008), Carter (2017) found that languages tend to re-use phonological features: The probability that a language which uses Feature F in N phonologically-active classes uses it in $N+1$ classes increases with N (a preferential-attachment process).

Nepotism. A weighty macro-constraint in the large-N/small-ζ case generates many mutant offspring, thereby maintaining related macro-constraints at higher weights than justified by their usefulness. Hence learners should show emergent effects of constraints that are mutationally close to high-weighted ones. In adult segment-class learning, generalization to untrained segments is stronger when they are more similar to trained segments (Cristiá et al., 2013). Prickett (2018) showed that GMECCS (a gradient-ascent Maximum Entropy learner, Pater and Moreton 2012; Moreton et al. 2017) underpredicts that difference, but that the fit can be improved by making weight updates "leak" between featurally-similar constraints. Nepotism may furnish a mechanism to cause such leakage.

Cognitive realism. Human participants report different approaches, including intuition, rote memorization, and explicit reasoning. Differences in self-reported approach correlate with differences in objective measures such as pattern-type difficulty order, learning-curve shape, and ability to verbalize the pattern (Moreton and Pertsova 2016, Moreton and Pertsova, in prep.). Simulations 2 and 3 illustrated parameter settings corresponding to intuition (large N, small ζ, random selection) and to a rudimentary sort of reasoning (small N, large ζ, fitness-based selection), and the p_{mem} parameter enables stimulus memorization. It would be desirable to know if intermediate combinations of parameter values correspond to types of human performance, how parameter values are linked to experimental conditions, and whether the number of parameters can safely be reduced.

References

Adriaans, F. and R. Kager (2010). Adding generalization to statistical learning: the induction of phonotactics from continuous speech. *Journal of Memory and Language 62*(3), 311–331.

Alderete, J. (1999). *Morphologically governed accent in Optimality Theory*. Ph. D. thesis, University of Massachusetts, Amherst.

Anderson, S. R. (1981). Why phonology isn't "natural". *Linguistic Inquiry 12*, 493–539.

Ashby, F. G., E. J. Paul, and W. T. Maddox (2011). COVIS. In E. M. Pothos and A. J. Willis (Eds.), *Formal approaches in categorization*, Chapter 4, pp. 65–87. Cambridge, England: Cambridge University Press.

Bach, E. and R. T. Harms (1972). How do languages get crazy rules? In R. P. Stockwell and R. K. S. Macaulay (Eds.), *Linguistic change and generative theory*, Chapter 1, pp. 1–21. Bloomington: Indiana University Press.

Bäck, T. (Ed.) (1996). *Evolutionary algorithms in theory and practice : evolution strategies, evolutionary programming, genetic algorithms*. New York: Oxford University Press.

Barlow, J. A. and D. A. Dinnsen (1998). Asymmetrical cluster development in a disordered system. *Language Acquisition 7*(1), 1–49.

Becker, M. (2009). *Phonological trends in the lexicon: the role of constraints*. Ph. D. thesis, University of Massachusetts, Amherst.

Becker, M. and A. Tessier (2011). Trajectories of faithfulness in child-specific phonology. *Phonology 28*, 163–196.

Benua, L. (1997). *Transderivational identity: phonological relations between words*. Ph. D. thesis, University of Massachusetts, Amherst, Mass.

Boersma, P. and J. Pater (2007, October). Constructing constraints from language data: the case of Canadian English diphthongs. Handout, NELS 38, University of Ottawa.

Bourne, Jr., L. E., W. D. Raymond, and A. F. Healy (2010). Strategy selection and use during classification skill acquisition. *Journal of Experimental Psychology: Learning, Memory, and Cognition 36*(2), 500–514.

Buckley, E. (2000). On the naturalness of unnatural rules. In *Proceedings from the Second Workshop on American Indigenous Languages*, Volume 9 of *UCSB Working Papers in Linguistics*.

Campbell, D. T. (1960). Blind variation and selective retention in creative thought as in other knowledge processes. *Psychological Review 67*(6), 380–400.

Carter, W. T. (2017). Phonological activeness effects in language acquisition and language structuring. Senior Honors thesis, Department of Linguistics, University of North Carolina, Chapel Hill.

Clements, G. N. and E. V. Hume (1995). The internal organization of speech sounds. In J. A. Goldsmith (Ed.), *The handbook of phonological theory*, Chapter 7, pp. 245–306. Boston: Blackwell.

Coetzee, A. W. and J. Pater (2008). Weighted constraints and gradient restrictions on place co-occurrence in muna and arabic. *Natural Language & Linguistic Theory 26*(2), 289–337.

Cristiá, A., J. Mielke, R. Daland, and S. Peperkamp (2013). Similarity in the generalization of implicitly learned sound patterns. *Laboratory Phonology 4*, 259–285.

De Jong, K. A. (2006). *Evolutionary computation: a unified approach*. Cambridge, Massachusetts: MIT Press.

Dietrich, A. and H. Haider (2015). Human creativity, evolutionary algorithms, and predictive representations: the mechanics of thought trials. *Psychonomic Bulletin and Review 22*, 897–915.

Donner, Y. and J. L. Hardy (2015). Piecewise power laws in individual learning curves. *Psychonomic Bulletin and Review 22*(5), 1308–1319.

Eiben, A. E. and J. E. Smith (2003). *Introduction to evolutionary computing*. Berlin: Springer.

Feldman, J. (2006). An algebra of human concept learning. *Journal of mathematical psychology 50*, 339–368.

Flack, K. (2007a). Templatic morphology and indexed markedness constraints. *Linguistic Inquiry 38*(4), 749–758.

Flack, K. G. (2007b). *The sources of phonological markedness*. Ph. D. thesis, University of Massachusetts, Amherst.

Fukazawa, H. (1999). *Theoretical implications of OCP effects on features in Optimality Theory*. Ph. D. thesis, University of Maryland, College Park.

Gerlach, S. R. (2010). *The acquisition of consonant feature sequences: harmony, metathesis, and deletion patterns in phonological development*. Ph. D. thesis, University of Minnesota.

Gluck, M. A. and G. H. Bower (1988a). Evaluating an adaptive network model of human learning. *Journal of Memory and Language 27*, 166–195.

Gluck, M. A. and G. H. Bower (1988b). From conditioning to category learning: an adaptive network model. *Journal of Experimental Psychology: General 117*(3), 227–247.

Goldsmith, J. A. (1976). *Autosegmental phonology*. Ph. D. thesis, Massachusetts Institute of Technology.

Goldwater, S. J. and M. Johnson (2003). Learning OT constraint rankings using a maximum entropy model. In J. Spenader, A. Erkisson, and O. Dahl (Eds.), *Proceedings of the Stockholm Workshop on Variation within Optimality Theory*, pp. 111–120.

Goodwin, G. P. and P. N. Johnson-Laird (2013). The acquisition of Boolean concepts. *Trends in Cognitive Sciences 17*(3), 128–133.

Gray, W. D. and J. K. Lindstedt (2017). Plateaus, dips, and leaps: where to look for inventions and discoveries during skilled performance. *Cognitive Science 41*, 1838–1870.

Gussenhoven, C. and H. Jacobs (2005). *Understanding phonology* (2nd ed.). Understanding Language Series. London: Hodder Arnold.

Guy, G. R. (2014). Linking usage and grammar: generative phonology, exemplar theory, and variable rules. *Lingua 142*, 57–65.

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

Hayes, B. (1999). Phonetically driven phonology: the role of optimality in inductive grounding. In M. Darnell, E. Moravcsik, M. Noonan, F. Newmeyer, and K. Wheatly (Eds.), *Functionalism and Formalism in Linguistics*, Volume 1: General Papers, pp. 243–285. Amsterdam: John Benjamins.

Hayes, B. and C. Wilson (2008). A Maximum Entropy model of phonotactics and phonotactic learning. *Linguistic Inquiry 39*(3), 379–440.

Inkelas, S. (2008). The morphology-phonology connection. In *Proceedings of the Berkeley Linguistics Society*, Volume 34, Berkeley, California, pp. 145–162. Berkeley Linguistics Society and Linguistic Society of America.

Ito, J. and A. Mester (2001). Covert generalizations in Optimality Theory: the role of stratal faithfulness constraints. *Studies in Phonetics, Phonology, and Morphology 7*, 3–33.

Jäger, G. (2007). Maximum Entropy models and Stochastic Optimality Theory. In J. Grimshaw, J. Maling, C. Manning, J. Simpson, and A. Zaenen (Eds.), *Architectures, rules, and preferences: a festschrift for Joan Bresnan*, pp. 467–479. Stanford, California: CSLI Publications.

Legendre, G., Y. Miyata, and P. Smolensky (1990). Can connectionism contribute to syntax? Harmonic Grammar, with an application. In M. Ziolkowski, M. Noske, and K. Deaton (Eds.), *Proceedings of the 26th Regional Meeting of the Chicago Linguistic Society*, Chicago, pp. 237–252. Chicago Linguistic Society.

Levelt, C. and M. van Oostendorp (2007). Feature co-occurrence constraints in L1 acquisition. *Linguistics in the Netherlands 24*(1), 162–172.

Littlestone, N. (1988). Learning quickly when irrelevant attributes abound: a new linear-threshold algorithm. *Machine Learning 2*, 285–318.

Luce, R. D. (2005 [1959]). *Individual choice behavior: a theoretical analysis*. New York: Dover.

Macken, M. A. and D. Barton (1978, March). The acquisition of the voicing contrast in English: a study of voice-onset time in word-initial stop consonants. Report from the Stanford Child Phonology Project.

Magri, G. (2013). HG has no computational advantages over OT: toward a new toolkit for computational OT. *Linguistic Inquiry 44*(4), 569–609.

McCarthy, J. J. (1981). A prosodic theory of nonconcatenative morphology. *Linguistic Inquiry 12*, 373–418.

McCarthy, J. J. and A. M. Prince (1993). Generalized alignment. In G. Booij and J. van Marle (Eds.), *Yearbook of morphology 1993*, pp. 79–153. Kluwer.

Mielke, J. (2008). *The emergence of distinctive features*. Oxford, England: Oxford University Press.

Moreton, E. (2010a, April). Connecting paradigmatic and syntagmatic simplicity bias in phonotactic learning. Department colloquium, Department of Linguistics, MIT.

Moreton, E. (2010b, February). Constraint induction and simplicity bias. Talk given at the Workshop on Computational Modelling of Sound Pattern Acquisition, University of Alberta.

Moreton, E. (2010c, May). Constraint induction and simplicity bias in phonotactic learning. Handout from a talk at the Workshop on Grammar Induction, Cornell University.

Moreton, E. (2019). Constraint breeding during online incremental learning. In *Proceedings of the Society for Computation in Linguistics*, Volume 2, pp. Article 9.

Moreton, E., J. Pater, and K. Pertsova (2017). Phonological concept learning. *Cognitive Science 41*(1), 4–69.

Moreton, E. and K. Pertsova (2016). Implicit and explicit processes in phonotactic learning. In TBA (Ed.), *Proceedings of the 40th Boston University Conference on Language Development*, Somerville, Mass., pp. TBA. Cascadilla.

Nosofsky, R. M., M. A. Gluck, T. J. Palmeri, S. C. McKinley, and P. Gauthier (1994). Comparing models of rule-based classification learning: a replication and extension of Shepard, Hovland, and Jenkins (1961). *Memory and Cognition 22*(3), 352–369.

Nosofsky, R. M., T. J. Palmeri, and S. C. McKinley (1994). Rule-plus-exception model of classification learning. *Psychological Review 101*(1), 53–79.

Ota, M. (2004). The learnability of the stratified phonological lexicon. *Journal of Japanese Linguistics 20*, 19–40.

Pater, J. (2000). Non-uniformity in English secondary stress: the role of ranked and lexically specific constraints. *Phonology 17*, 237–274.

Pater, J. (2007). The locus of exceptionality: morpheme-specific phonology as constraint indexation. In L. Bateman, M. O'Keefe, E. Reilly, and A. Werle (Eds.), *Papers in Optimality Theory III*, pp. 259–296. Amherst: Graduate Linguistics Students Association, University of Massachusetts.

Pater, J. (2009). Morpheme-specific phonology: constraint indexation and inconsistency resolution. In S. Parker (Ed.), *Phonological argumentation: essays on evidence and motivation*, pp. 1–33. London: Equinox.

Pater, J. (2014). Canadian Raising with language-specific weighted constraints. *Language 90*(1), 230–240.

Pater, J. and E. Moreton (2012). Structurally biased phonology: complexity in learning and typology. *Journal of the English and Foreign Languages University, Hyderabad 3*(2), 1–44.

Pizzo, P. (2013, January 19). Learning phonological alternations with online constraint induction. Slides from a presentation at the 10th Old World Conference on Phonology (OCP 10).

Prickett, B. (2018). Similarity-based phonological generalization. In G. Jarosz and J. Pater (Eds.), *Proceedings of the Society for Computation in Linguistics*, Volume 1, pp. Article 24.

Prince, A. and P. Smolensky (1993). *Optimality Theory: constraint interaction in generative grammar*. Department of Linguistics, Rutgers University.

R Core Team (2018). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.

Rasin, E. and R. Katzir (2016). On evaluation metrics in optimality theory. *Linguistic Inquiry 47*(2), 235–282.

Sagey, E. (1990). *The representation of features in non-linear phonology: the Articulator Node Hierarchy*. New York: Garland.

Shepard, R. N., C. L. Hovland, and H. M. Jenkins (1961). Learning and memorization of classifications. *Psychological Monographs 75*(13, Whole No. 517).

Simonton, D. K. (1999). Creativity as blind variation and selective retention: is the creative process Darwinian? *Psychological Inquiry 10*(4), 309–328.

Smith, J. L. (2002). *Phonological augmentation in prominent positions*. Ph. D. thesis, University of Massachusetts, Amherst.

Smith, J. L. (2006). Representational complexity in syllable structure and its consequences for Gen and Con. MS, Department of Linguistics, University of North Carolina, Chapel Hill. ROA-800.

Smith, N. V. (1973). *The acquisition of phonology: a case study*. Cambridge, England: Cambridge University Press.

Vihman, M. M. and S. Velleman (1989). Phonological reorganization: a case study. *Language and Speech 32*, 149–170.

Wilson, C. and G. Gallagher (2018). Accidental gaps and surface-based phonotactic learning: A case study of South Bolivian Quechua. *Linguistic Inquiry 49*(3), 610–623.