
Week 6 – Phonological Learning Models

March 24 and 26, 2008

1 The Gradual Learning Algorithm

- (1) The gradual learning algorithm (GLA) is an algorithm designed to learn stochastic OT grammars.
 - a. There are many ways to make OT stochastic, we review some below.
 - b. Stochastic OT is intended to address the problem of free variation in phonology.

1.1 Free variation in phonology

- (2) Sociolinguistic references
 - a. Sociolinguistics, by Peter Trudgill, a short text
 - b. Sociolinguistic Patterns and Language in the Inner City, early classics by William Labov
 - c. Labov's magnum opus, Principles of Linguistic Change, in three volumes, of which two have appeared
- (3) Essentially the claim is that variations in pronunciations vary in lockstep by speaking style.
 - a. “(r-0)” *beard* is r-less: [bɪəd]; “r-1” is r-ful
 - b. “eh-1” *bad* [bɪəd], diphthongized; “eh-4” is [æ]
 - c. “oh-1” coffee [ˈkʊəfi]; “oh-5” is [ɔ]
- (4) If true, this is something that theories need to predict.
- (5) from 81 native speakers of New York City English
 - a. Vertical axis: what percentage of underlying /ɹ/ are retained in the output?
 - b. Hypothesis: /ɹ/ is underlyingly present, learned from nondeleted tokens in the ambient language an independent investigation sorted the speakers into their social classes.
 - c. The “leaping up” of the lower-middle-class speakers in the formal styles is found in other studies, and is claimed to reflect a social insecurity independently diag-

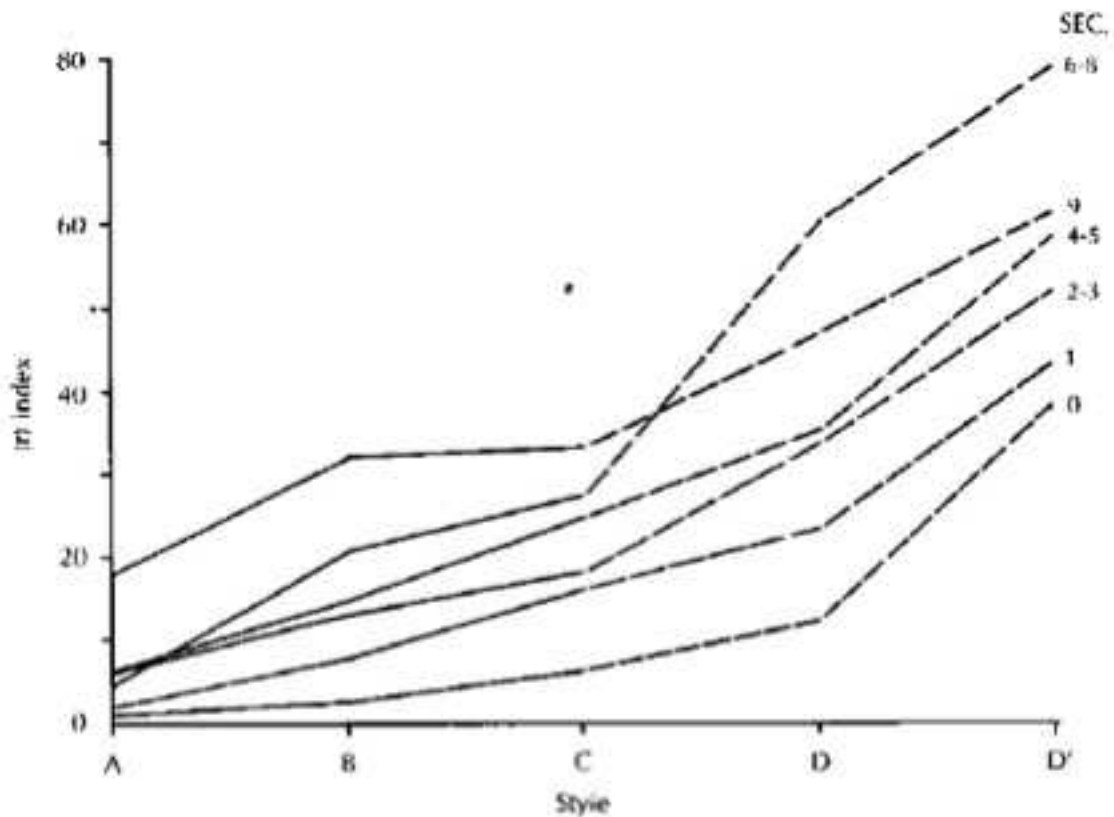


Fig. 4.2. Class stratification of a linguistic variable in process of change: (r) in *grard, our, bear, beard, board*, etc. SEC (Socio-economic class) scale: 0-1, lower class; 2-4, working class; 5-6, 7-8, lower middle class; 9, upper middle class. A, casual speech; B, careful speech; C, reading style; D, word lists; D', minimal pairs.

Figure 1: This figure from William Labov (1972) *Sociolinguistic Patterns*

nosed by other tests¹

1.2 Approaches to free variation in phonology

1.2.1 Variable rules

- (6) This was the primary device used in the classic analytic work of formal sociolinguists; see work of David Sankoff and colleagues, e.g. Cedergren and Sankoff (1974).

Rules as a whole, or parts of their structural descriptions, could be annotated for frequency of application.

¹For example: series of questions: "how do you say this word? ... how should this word be said?".

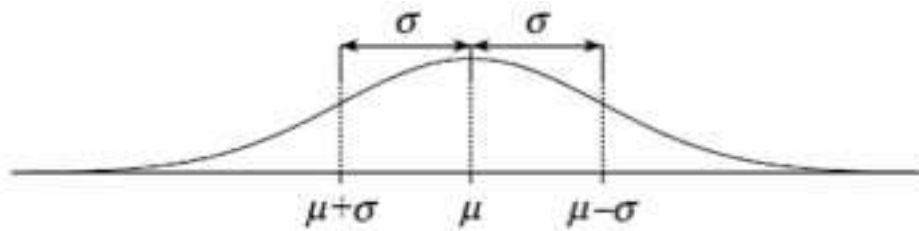
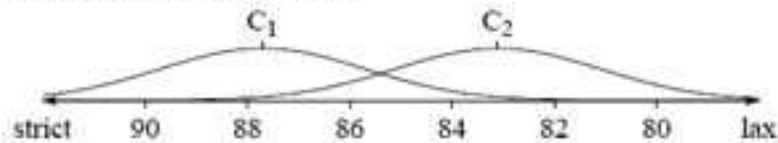


Figure 3: The Gaussian Curve

(6) *Overlapping ranking distributions*

In (6), the ranking values for C_1 and C_2 are at the hypothetical values 87.7 and 83.1. Since the evaluation noise is 2.0, the normal distributions assigned to C_1 and C_2 overlap substantially. While the selection points for C_1 and C_2 will most often occur somewhere in the central “hump” of their distributions, they will on occasion be found quite a bit further away. Thus, C_1 will outrank C_2 at evaluation time in most cases, but the opposite ranking will occasionally hold. Simple calculations show that the percentages for these outcomes will tend towards the values 94.8% ($C_1 \gg C_2$) and 5.2% ($C_2 \gg C_1$).

Figure 4: Boersma's Stochastic OT

1.2.4 Boersmas version uses Gaussian distributions

1.2.5 Relative ranking probability from overlapping distributions

(12)	Difference in ranking value	Probability higher outranks lower
	0	0.5
	0.1	0.51
	0.5	0.57
	1	0.64
	5	0.96
	10	0.9998
	50	1.00000000

- ★ Construct a stochastic OT grammar that drops /r/ in codas 36 of the time and in onsets never.

1.3 Learning Free Variation: How the GLA works

(13) Main refs: Boersma (1997), Boersma and Hayes (2001)

Algorithm 1 The Gradual Learning Algorithm in plain language

Input: a set of constraints all set with the same center value (say 100), a *plasticity* value, say 2

Output: a stratified constraint hierarchy (if there is one)

while the grammar has not reached a *stable* state **do**

1. Hear a datum, i.e. a (UR, SR) pair. Use current grammar to generate SR' given the UR.

2. Compare observed SR to generated SR'.

if the form just generated by the grammar matches the learning datum **then**

do nothing

else

adjust the grammar, so the revised grammar will be more likely to derive the correct form.

Adjustments: Raise the ranking value of all constraints which prefer the winner (SR), by a small amount. Lower the ranking value of all constraints which prefer the loser (SR'), by a small amount

end if

reduce the plasticity a little

end while

(14) Some comments on the GLA

- a. Other possible initial states could be explored, e.g. “start Faithfulness low.” (cf. Biased Constraint Demotion)
- b. Like Recursive Constraint Demotion (RCD), GLA assumes the UR is known.
- c. The **while** loop terminates when the differences between the grammars “pre-loop” and “post-loop” are below some tiny threshold.
- d. Plasticity
 - (i) the small amount of adjustment made
 - (ii) In practice, one sets a “plasticity schedule” for learning—large at first, then smaller
 - (iii) This is supposed to be why the GLA eventually stabilizes (because the plasticity becomes close to zero so the grammars will never change all that much)

(15) The idea behind the GLA:

- a. The altered grammar becomes slightly less likely to generate its wrong-guess output, and slightly more likely to generate the correct output.
- b. But note it is not guaranteed to converge. See <http://www.fon.hum.uva.nl/paul/gla/> for Boersma’s comments on this.

1.4 Demo with r-dropping.eps

- (16) Suppose we are modeling a particular speaker, and a particular style, in which /ɹ/ is dropped 80% of the time.

			*r	Max(r) /	_____	V	Max(r)
			*r	Max(r) /	_____	V	Max(r)
kar	kar	0.2	1				
	ka	0.8					1
rak	rak	1	1				
	ak			1			1

- (17) Learning Regimen
- a. 50,000 forms given at random, matching frequencies of input file.
 - b. Plasticity dropped from 1 to .001, gradually.
- (18) First Few Actions of the Algorithm

Generated	Heard	*r	now	Max(r) /	_____	V	now	Max(r)	now
(Initial)			100				100		100
kar	ka	1	101					-1	99
kar	ka	1	102					-1	98
ak	rak	-1	101	1			101	1	99
ak	rak	-1	100	1			102	1	100
kar	ka	1	101					-1	99
ak	rak	-1	100	1			103	1	100
kar	ka	1	101					-1	99
kar	ka	1	102					-1	98
ak	rak	-1	101	1			104	1	99
ak	rak	-1	100	1			105	1	100
ak	rak	-1	99	1			106	1	101
kar	ka	1	100					-1	100
ka	kar	-1	99					1	101
kar	ka	1	100					-1	100
kar	ka	1	101					-1	99
kar	ka	1	102					-1	98
ka	kar	-1	101					1	99
kar	ka	1	102					-1	98
kar	ka	1	103					-1	97
ak	rak	-1	102	1			107	1	98

- (19) End Result:

The algorithm yielded the following ranking values:

111.000 Max(r) / ___V
 101.159 *r
 98.841 Max(r)

These correspond to the following pairwise ranking probabilities:

.9998 Max(r) / ___V >> *r
 .794 *r >> Max(r)

Matchup to Input Frequencies (stochastic trials)

/kar/	Input Fr.	Gen Fr.	Gen. #
ka	0.800	0.771	1541
kar	0.200	0.230	459

/rak/	Input Fr.	Gen Fr.	Gen. #
rak	1.000	1.000	2000
ak	0.000	0.000	

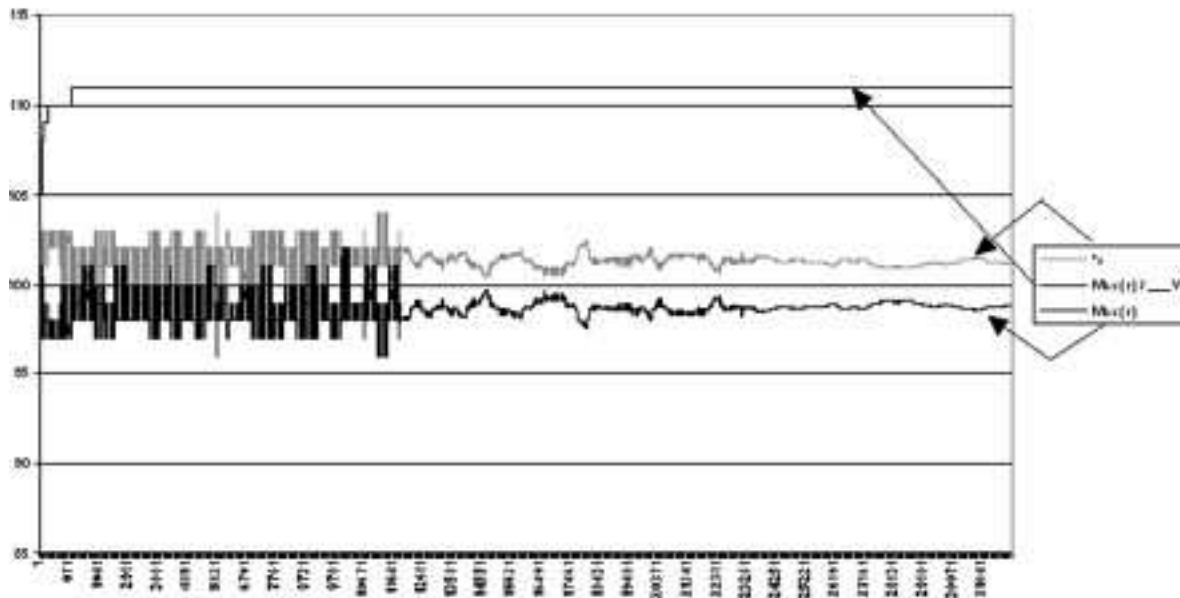


Figure 5: History of Ranking

1.5 Problems with the GLA

- (20) How to learn style?
- Different cases of free variation can occur in “lockstep” with speaking style (Labov).
 - See Appendix to Boersma and Hayes for a suggestion.
 - Is semi-grammaticality related to frequency?
 - If so, can this be used to analyze gradient grammaticality?
- (21) Can intermediate stages of learning be affiliated with observed intermediate stages in children?
- Two attempts:
 - Boersma, Paul and Clara Levelt, “Gradual Constraint-ranking Learning Algorithm predicts acquisition order,” *Proceedings of Child Language Research Forum 30*, Stanford, California, pp. 229-237. ROA 361.
 - Curtin, Suzanne and Kie Zuraw (to appear). Explaining Constraint Demotion in a Developing System. In Anna H.-J. Do, Laura Domnguez, and Aimee Johansen, editors, *BUCLD 26: Proceedings of the 26th Annual Boston University Conference on Language Development*. Cascadilla Press.
- (22) Is there a better algorithm for stochastic OT than the GLA?
- The GLA has been recently shown to fail in plausible cases by Joe Pater.
 - A better algorithm, the “Luce choice ranker”, is a current project of Colin Wilson.

2 Principles and Parameters Learning Models

- (23) Main refs: Dresher and Kaye (1990), Gillis et al. (1995), Dresher (1999)
- Additional refs (for learning P&P grammars in the domain of syntax): Gibson and Wexler (1994), Niyogi and Berwick (1996), Frank and Kapur (1996), Niyogi (2006)

2.1 The basic idea

- (24) Recall that one way to specify a stress pattern of a language is by setting various parameters
- Iambic/Trochaic
 - iterative/non-iterative,
 - Right-to-left/Left-to-Right/
 - Quantity Insensitive/Sensitive
 - Nonfinality
 - ...

- (25) In other words, a grammar is just a finite vector of values:
- < Trochaic, Iterative, Right-to-Left, Quantity-sensitive, ... >
- (26) The learner's job:
- a. go from some initial setting of these parameters to the target setting giving only (positive) examples from the target language.
- (27) The Triggering Learning Algorithm (TLA) ((Gibson and Wexler 1994)):
- a. The learner possesses knowledge about *cues* which *trigger* parameter settings.
 - b. Essentially, the learner searches the linguistic input for cues and then sets parameters as needed.
- (28) Triggers:
- a. Some triggers are *global*—no matter the current state of the grammar, observing this cue in the input causes the relevant parameter to be set appropriately.
 - b. Some triggers are *local*—in order for an observed cue to set the relevant parameter appropriately, the state of the current grammar matters.
- (29) Main results:
- a. The TLA is essentially Markovian (given current state of grammar and cue, the next state of grammar is predictable).
 - (i) There are “sinkholes”— i.e. parameter settings that a learner can get stuck in and can never escape.
 - (ii) Frank and Kapur also have some surprising results regarding global triggers.

2.2 Dresher (1999) criticisms of the TLA

Dresher (1999):

... at the most general level... the learning algorithm is independent of the content of the grammar. ... for example, ... it makes no difference to the [Triggering Learning Algorithm (Gibson and Wexler 1994)] what the content of a parameter is: the same chart serves for syntactic word order parameters as for parameters of metrical theory, or even for nonlinguistic parameters.

- (30) In other words, in the P&P and OT frameworks, the proposed learning mechanisms operate over the structure provided by the framework, and not any inherent structure that may exist in the hypothesis space itself.
- (31) Dresher (1999:28) draws a distinction between the Triggering Learning Algorithm (TLA) and the ordered cue learning model of Dresher and Kaye (1990), explaining that ‘Cues must be appropriate to their parameters in the sense that the cue must reflect a fundamental property of the parameter, rather than being fortuitously related to it.’
- (32) What does Dresher mean exactly?

- (32) Neither Dresher and Kaye (1990) nor Dresher (1999) offer a precise explanation of what a ‘fundamental property’ of a parameter would look like, or what properties of an associated cue make it appropriate.
- (32) Thus it is not exactly clear how different the ordered cue based learner is from the TLA in this respect. This is exactly the problem Gillis et al. (1995) run into with Dresher’s approach.
- (33) Conclusion: Dresher is right, but doesn’t have a viable alternative.

2.3 What about OT

- (34) Tesar and Smolensky (2000:7-8) state quite clearly:
- OT is a theory of UG that provides sufficient structure at the level of the grammatical framework itself to allow general but grammatically informed learning algorithms to be defined... Yet the structure that makes these algorithms possible is not the structure of a theory of stress, nor a theory of phonology: it is the structure defining any OT grammar...
- (35) I.e. OT suffers the same problem. This is not controversial (see above), though hardly ever emphasized, as I do here.

3 Other frameworks

- (36) Finally, for completeness, there have been other approaches to learning stress patterns as well: e.g. dynamic systems and connectionist models Goldsmith (1994), Gupta and Touretzky (1991, 1994).

4 Learning-theoretic Models

- (37) Stay tuned for mini-presentation on Wednesday...
- (38) The goal is convince you that viable alternatives do exist, and that the models that I am proposing, while by no means the end of the story, do in fact meet Dresher’s criteria, which both OT and P&P models do not.

References

- Anttila, Arto. 1997a. Deriving variation from grammar: A study of Finnish genitives. In *Variation, change and phonological theory*, edited by Frans Hinskens, Roeland van Hout, and Leo Wetzels.
- Anttila, Arto. 1997b. Variation in Finnish phonology and morphology. Ph.D. thesis, Stanford University.

- Boersma, Paul. 1997. How we learn variation, optionality, and probability. *Proceedings of the Institute of Phonetic Sciences* 21. University of Amsterdam.
- Boersma, Paul and Bruce Hayes. 2001. Empirical tests of the Gradual Learning Algorithm. *Linguistic Inquiry* 32:45–86.
- Cedergren, Henrietta J. and David Sankoff. 1974. Language. *Performance as a Statistical Reflection of Competence* 50:333.
- Dresher, Elan. 1999. Charting the Learning Path: Cues to Parameter Setting. *Linguistic Inquiry* 30:27–67.
- Dresher, Elan and Jonathan Kaye. 1990. A Computational Learning Model for Metrical Phonology. *Cognition* 34:137–195.
- Frank, Robert and Shyam Kapur. 1996. On the use of triggers in parameter setting. *Linguistic Inquiry* 27(623-660).
- Gibson, Edward and Kenneth Wexler. 1994. Triggers. *Linguistic Inquiry* 25(3):407–454.
- Gillis, Steven, Gert Durieux, and Walter Daelemans. 1995. A computational model of P&P: Dresher & Kaye (1990) revisited. In *Approaches to parameter setting*, edited by Frank Wijnen and Maaïke Verrips. Vakgroep Algemene Taalwetenschap, Universiteit van Amsterdam, pages 135–173.
- Goldsmith, John. 1994. A Dynamic Computational Theory of Accent Systems. In *Perspectives in Phonology*, edited by Jennifer Cole and Charles Kisseberth. Stanford: Center for the Study of Language and Information, pages 1–28.
- Gupta, Prahlad and David Touretzky. 1991. What a Perceptron Reveals about Metrical Phonology. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*. pages 334–339.
- Gupta, Prahlad and David Touretzky. 1994. Connectionist models and linguistic theory: Investigations of stress systems in language. *Cognitive Science* 18(1):1–50.
- Niyogi, Partha. 2006. *The Computational Nature of Language Learning and Evolution*. The MIT Press.
- Niyogi, Partha and Robert Berwick. 1996. A language learning model for finite parameter spaces. *Cognition* 61:161–193.
- Tesar, Bruce and Paul Smolensky. 2000. *Learnability in Optimality Theory*. MIT Press.