Linguistics 251 Hayes/Zuraw
Variation in Phonology Spring 2013

Exercise 2: Comparing Maxent and Stochastic OT/GLA

Due in class 4/18/13

This is a brief series of learning runs, to be carried out *both* using the GLA/Stochastic OT and maxent. For maxent, you can use either OTSoft or the Maxent Grammar Tool.

Make brief replies where appropriate, and paste appropriate bits of program output to demonstrate your point.

On OTSoft, the same learning parameters are used for both GLA and Maxent. Adjust them to get reasonably accurate outputs. This means: make sure there are enough iterations, the opening plasticity should be pretty high (like 2), and the closing plasticity should be pretty low (like 0.01).

1. Replicating strict ranking

The file StrictRanking.txt assesses the ability of a stochastic algorithm to replicate the results of classical OT on the fairly deep chain of strict rankings. It looks like this:

			PreferA	PreferB	PreferC	PreferD	PreferE
			PreferA	PreferB	PreferC	PreferD	PreferE
ABChoice	A	1000		1			
	B		1				
BCChoice	B	1000			1		
	C			1			
CDChoice	C	1000				1	
	D				1		
DEChoice	D	1000					1
	E					1	

Run this in both the GLA and Maxent. If you use OTSoft maxent, request enough decimal places in the output (like 7) to get an accurate assessment.

2. Constraints that tweak

Here, we have a truly stochastic simulation that test whether a weak constraint can "tweak" the result of a competition that comes out all-or-nothing elsewhere.

			PreferA	PreferB	PreferC	PreferPrime
			PreferA	PreferB	PreferC	PreferPrime
ABChoice	A	1000		1		
	B		1			
BCChoice	B	1000			1	
	C			1		
ABChoiceWithPrime	A	900		1		1
	BPrime	100	1			

If learning comes out ok, PreferPrime will "tweak" the counts of A vs. B a bit, making the victory of A over B less complete for the third input.

3. Double-tweak

Can a weak constraint "tweak" the output of two other constraints that are ranked far apart? The following input file combines aspects of the previous two to test this possibility.

			PreferA	PreferB	PreferC	PreferD	PreferE	PreferPrime
			PreferA	PreferB	PreferC	PreferD	PreferE	PreferPrime
ABChoice	A	1000		1				
	B		1					
BCChoice	B	1000			1			
	C			1				
CDChoice	C	1000				1		
	D				1			
DEChoice	D	1000					1	
	E					1		
ABChoicePrime	A	90		1				1
	BPrime	10	1					
DEChoicePrime	D	90					1	1
	E	10				1		

Try this out in both maxent and GLA. If GLA has trouble, try to diagnose. Assume, for purposes of argument, that the real world really does include "double tweak" cases. If so, is this a problem of *learning* (i.e. GLA) or of *framework* (i.e. stochastic OT)? I.e. is there *any* stochastic OT grammar that can generate this pattern?