UNIVERSAL BIASES IN PHONOLOGICAL LEARNING

ACTL SUMMER SCHOOL, DAY 5

JAMIE WHITE (UCL)

OVERVIEW OF MAXENT MODELS

PLAN

1. What is a maximum entropy model (and specifically, as applied in phonology)?

2. Demystify how the model works by working through mini-cases by hand!

3. Show you how the MaxEnt Grammar Tool works, in case you want to try it on your own data.

4. Look at implementing a substantive (P-map) bias using the prior.

WHAT IS MAXENT?

Maximum entropy models:

- General class of statistical classification model.
- Mathematically very similar to logistic regression.
- Wide, longstanding use in many fields.

As applied to phonological grammars:

- Constraint-based.
- A variety of Harmonic Grammar.
- Comes with a learning algorithm with an objective function.
- Probabilistic ⟶ readily able to account for variation.
- Priors ⟶ allow us to implement learning biases.

- Early application to phonology: Goldwater & Johnson 2003.
- Some other refs: Wilson 2006, Hayes & Wilson 2008, Hayes et al. 2009, White 2013.

‘REGULAR’ HARMONIC GRAMMAR (NON-PROBABILISTIC)

Roots go back to pre-OT (e.g. Legendre et al. 1990), with a resurgence in recent years (e.g. Pater 2009, Potts et al. 2009).

Main difference from OT: Method of evaluation

- Rather than being strictly ranked, each constraint is associated with a weight.
- To determine the winning candidate:
  - Take the sum of the weighted constraint violations.
  - Candidate with the lowest penalty (i.e. most Harmony) is the winner.

Perhaps the most noteworthy property of any harmonic grammar: the “ganging” property.

- Multiple violations of lower weighted constraints can ‘gang up’ to overtake a stronger constraint.

EXAMPLE FROM JAPANESE

Phonotactic restrictions in native Japanese words:

- No words with two voiced obstruents ("Lyman's Law"): *[baga].
- No voiced obstruent geminates: [tt] ok, but *[dd].

Violations possible in loanwords.

- Two voiced obstruents ok:
  - [bogiː] 'bogey', [dogɯma] 'dogma'
- Voiced geminates ok:
  - [habbɯrɯ] 'Hubble', [webbɯ] 'web'

But, both cannot be violated in the same word:

- [betto] 'bed', [dokkɯ] 'dog'

Kawahara 2006, *Language*

CLASSICAL OT FAILS AT THIS

/bogiː/	IDENT(voice)	*D…D	*VOICEDGEM
☞ bogiː		*	
pogiː	*!		
bokiː	*!		

/webbɯ/	IDENT(voice)	*D…D	*VOICEDGEM
☞ webbɯ			*
weppɯ	*!		

/doggɯ/	IDENT(voice)	*D…D	*VOICEDGEM
💣 doggɯ		*	*
☹ dokkɯ	*!		
toggɯ	*!		*

HARMONIC GRAMMAR ANALYSIS

/bogiː/	TOTAL PENALTY	IDENT(voice) 3	*D…D 2	*VOICEDGEM 2
bogiː			*	
pogiː		*		
bokiː		*		

/webbɯ/	TOTAL PENALTY	IDENT(voice) 3	*D…D 2	*VOICEDGEM 2
webbɯ				*
weppɯ		*		

/doggɯ/	TOTAL PENALTY	IDENT(voice) 3	*D…D 2	*VOICEDGEM 2
doggɯ			*	*
dokkɯ		*		
toggɯ		*		*

HARMONIC GRAMMAR ANALYSIS

/bogiː/	TOTAL PENALTY	IDENT(voice) 3	*D…D 2	*VOICEDGEM 2
☞ bogiː	2		2	
pogiː	3	3		
bokiː	3	3		

/webbɯ/	TOTAL PENALTY	IDENT(voice) 3	*D…D 2	*VOICEDGEM 2
☞ webbɯ	2		2	
weppɯ	3	3		

/doggɯ/	TOTAL PENALTY	IDENT(voice) 3	*D…D 2	*VOICEDGEM 2
doggɯ	4		2	2
☞ dokkɯ	3	3		
toggɯ	5	3		2

MAXENT GRAMMAR: A PROBABILISTIC VERSION OF HARMONIC GRAMMAR

GENERATING PROBABILITIES FROM THE GRAMMAR

Each constraint is associated with a non-negative weight.

1. First, take the sum of the weighted constraint violations for each candidate.

- This is the Penalty for each candidate.
- So far, this is the same as regular HG.

2. Take $e^{(-\text{penalty})}$ for each candidate.

3. For each candidate, divide its $e^{(-\text{penalty})}$ by the sum for all candidates (i.e. take each candidate's proportion of the total).

- This is the output probability of each candidate.

LET'S TRY WITH THE JAPANESE CASE

Assume the following input probabilities for three cases:

Words like /doggɯ/ : 57.4% devoiced to [dokkɯ]

Words like /webbɯ/ : 3.7% devoiced to [weppɯ]

Words like /bogii/ : 0.1% devoiced to [bokii]

Weights learned by the grammar:

- IDENT(voice): 11.36
- *D…D: 3.56
- *VOICEDGEM: 8.10

GENERATING PROBABILITIES FROM A GRAMMAR

/beddo/	Predicted probability	$e^{(-\text{penalty})}$	Total Penalty	IDENT(voice) 11.36	*D…D 3.56	*VOICEDGEM 8.10
beddo	.424	.0000086	11.66		3.56	8.10
betto	.576	.0000117	11.36	11.36		
peddo	~ 0	~ 0	19.46	11.36		8.10
petto	~ 0	~ 0	22.72	22.72		
		.0000203				

/webbu/	Predicted probability	$e^{(-\text{penalty})}$	Total Penalty	IDENT(voice) 11.36	*D…D 3.56	*VOICEDGEM 8.10
webbu	.963	.0003035	8.10			8.10
weppu	.037	.0000117	11.36	11.36		
		.0003152				

/bogii/	Predicted probability	$e^{(-\text{penalty})}$	Total Penalty	IDENT(voice) 11.36	*D…D 3.56	*VOICEDGEM 8.10
bogii	.999	.0284389	3.56		3.56	
bokii	~ 0	.0000117	11.36	11.36		
pogii	~ 0	.0000117	11.36	11.36		
pokii	~ 0	~ 0	22.72	22.72		

↓

Sum: .0284623

OK…WE KNOW HOW TO GET THE PROBABILITIES FROM A LEARNED GRAMMAR.

NOW, HOW IS THE GRAMMAR LEARNED?

OBJECTIVE FUNCTION

The goal of learning is to maximize this objective function:

$$\left[\sum_{j=1}^{n} \log \Pr(y_j \mid x_j) \right] - \left[\sum_{i=1}^{m} \frac{(w_i - \mu_i)^2}{2\sigma_i^2} \right]$$

Maximize the (log) probability of the data

Apply a penalty for constraints gone wild (i.e. according to how much they vary from a preferred weight).

= the prior

WHAT DOES IT MEAN TO MAXIMIZE THE (LOG) LIKELIHOOD

Simple example: Imagine that we are trying to model these data:

/badub/ [badup]: 7
[badub]: 3

And we want to compare these two possible grammars:

Grammar$_1$: $*D]_{word}$ 1.55 IDENT(voice) 1.30
Grammar$_2$: $*D]_{word}$ 2.70 IDENT(voice) 2.00

First, we need to know the predicted probability under each grammar:

/badub/	Pred. prob.	$e^{(-pen)}$	Total Penalty	$*D]_{word}$ 1.55	ID(vce) 1.30	/badub/	Pred. prob.	$e^{(-pen)}$	Total Penalty	$*D]_{word}$ 2.70	ID(vce) 2.00
badup	.56	.2725	1.30		1.30	badup	.67	.1353	2.00		2.00
badub	.44	.2122	1.55	1.55		badub	.33	.0672	2.70	2.70	
		.4847						.2025			

Then, calculate the log likelihood under each grammar:

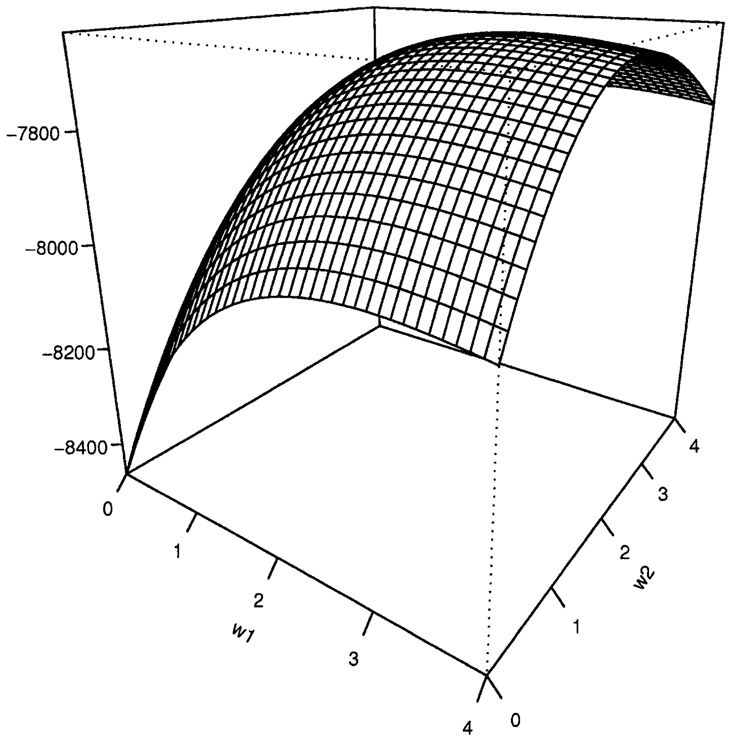
/badub/	Observed frequency	Predicted probability, Grammar$_1$	Grammar$_1$ (ln(p))	ln(p) x obs. freq.	Predicted probability, Grammar$_1$	Grammar$_1$ (ln(p))	ln(p) x obs. freq.
badup	7	.56	–.5798	–4.0586	.67	–.4005	–2.8035
badub	3	.44	–.8210	–2.4630	.33	–1.1087	–3.3261
			Sum:	–6.5216		Sum:	–6.1296

SEARCH SPACE

The real model considers all possible sets of values for the constraints.

- And it is guaranteed to find the best weights. How??

The search space is provably convex.



(from Hayes & Wilson, 2008, p. 387)

OBJECTIVE FUNCTION

The goal of learning is to maximize this objective function:

$$\left[\sum_{j=1}^{n} \log \Pr(y_j \mid x_j)\right] - \left[\sum_{i=1}^{m} \frac{(w_i - \mu_i)^2}{2\sigma_i^2}\right]$$

✔

Apply a penalty for constraints gone wild (i.e. according to how much they vary from a preferred weight).

= the prior

WHAT IS THE PRIOR?

The prior (short for prior distribution) is a way of biasing the model towards certain learning outcomes.

Often used as a "smoothing" component to prevent overfitting.

- In this case, it is a Gaussian (=normal) distribution over each constraint, defined in terms of:
  - μ = *a priori* preferred weight for the constraint.
  - σ = how tightly the constraint is bound to its preferred weight during learning.

- For smoothing purposes, it is common to set them as follows:
  - $\mu = 0$ (i.e. constraints want to be close to 0)
  - σ = some appropriate value, e.g. 1.

LET'S SEE WHY THE PRIOR WORKS THIS WAY

$$\left[\sum_{j=1}^{n} \log \Pr(y_j \mid x_j)\right] - \left[\sum_{i=1}^{m} \frac{(w_i - \mu_i)^2}{2\sigma_i^2}\right]$$

Earlier example:

/badub/	[badup]: 7	Grammar$_1$: *D]$_{word}$ 1.55	IDENT(voice) 1.30
	[badub]: 3	Grammar$_2$: *D]$_{word}$ 2.70	IDENT(voice) 2.00

Let's see how a smoothing prior (μ = 0, σ = 1) affects the outcome:

Grammar$_1$						
Summed log likelihood (from above)		Penalty for C_1 (weight: 1.55)		Penalty for C_2 (weight: 1.30)		Total
–6.5216	–	(1.20125	+	.845)	=	–8.56785
		2.04625				

Grammar$_2$						
Summed log likelihood (from above)		Penalty for C_1 (weight: 2.70)		Penalty for C_2 (weight: 2.00)		Total
–6.1296	–	(7.29	+	4)	=	–17.4196
		11.29				

MAXENT GRAMMAR TOOL

Software developed by Colin Wilson, Ben George, and Bruce Hayes.

- Available at Bruce Hayes's webpage:
- http://www.linguistics.ucla.edu/people/hayes/MaxentGrammarTool/

Takes an input file, a prior file (optional), and an output file.

- Includes a Gaussian prior.

Does the learning and outputs the weights and predicted probabilities for each candidate.

Very user-friendly.

Works on any platform.

MAXENT GRAMMAR TOOL

A little demonstration: MaxEnt does categorical cases.

Sample case: Nasal fusion in Indonesian (examples from Pater 1999)

/məŋ + pilih/ ⟶ [məmilih] ‘choose’

/məŋ + tulis/ ⟶ [mənulis] ‘write’

/məŋ + kasih/ ⟶ [məŋasih] ‘give’

/məŋ + bəli/ ⟶ [məmbəli] ‘buy’

/məŋ + dapat/ ⟶ [məndapat] ‘get’

/məŋ + ganti/ ⟶ [məŋganti] ‘change’

OT ACCOUNT

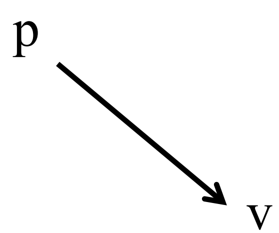
/məŋ$_1$p$_2$ilih/	*NC̥	IDENT(voice)	MAX-NASAL	UNIFORMITY
☞ məm$_{1,2}$ilih				*
məm$_1$b$_2$ilih		*!		
məm$_1$p$_2$ilih	*!			
məp$_2$ilih			*!	

IMPLEMENTING A SUBSTANTIVE BIAS VIA THE PRIOR

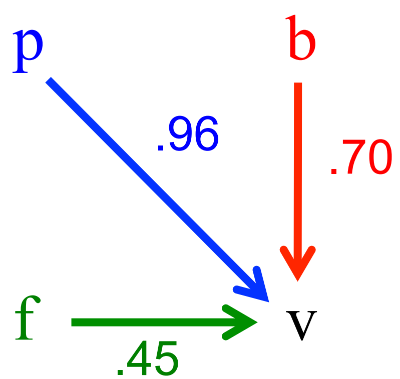
RESULTS (GENERALIZATION PHASE)

Potentially Saltatory condition

Input:

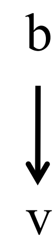


Results:

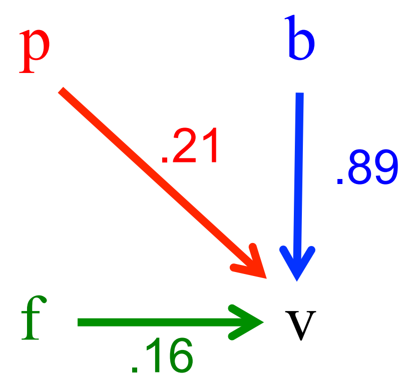


Control condition

Input:



Results:



White (2014), *Cognition*

RECALL THE P-MAP (STERIADE 2001)

1. Mental representation of perceptual similarity between pairs of sounds.

2. Minimal modification bias (large perceptual changes dispreferred by the learner).

How can we implement this in a learning model?

SOLUTION

Expand traditional faithfulness constraints to *MAP constraints (proposed by Zuraw 2007, 2013).

***MAP(x, y):**

- violated if sound x is in correspondence with sound y.
- E.g.: *MAP(p, v) is violated by p ~ v.

These constraints are constrained by a P-map bias.

- *Map constraints penalizing two sounds that are less similar will have a larger preferred weight (μ).

White, 2013, under review

CONFUSION DATA AND THE RESULTING PRIORS

Stimulus	Responses p	b	f	v	Stimulus	Responses t	d	θ	ð
p	1844	54	159	26	t	1765	107	92	26
b	206	1331	241	408	d	91	1640	75	193
f	601	161	1202	93	θ	267	118	712	135
v	51	386	127	1428	ð	44	371	125	680

(confusion data from Wang & Bilger 1973)

Labial sounds		Coronal sounds	
Constraint	Prior weight (μ)	Constraint	Prior weight (μ)
*MAP(p, v)	3.65	*MAP(t, ð)	3.56
*MAP(f, v)	2.56	*MAP(θ, ð)	1.91
*MAP(p, b)	2.44	*MAP(t, d)	2.73
*MAP(f, b)	1.96	*MAP(θ, d)	2.49
*MAP(p, f)	1.34	*MAP(t, θ)	1.94
*MAP(b, v)	1.30	*MAP(d, ð)	1.40

White, 2013, under review

GIVE THE MODEL THE SAME INPUT AS THE EXPERIMENTAL PARTICIPANTS

Experiment 1		Experiment 2	
Potentially Saltatory condition	Control condition	Saltatory condition	Control condition
18 p ⟶ v	18 b ⟶ v	18 p ⟶ v	18 b ⟶ v
18 t ⟶ ð	18 d ⟶ ð	18 t ⟶ ð	18 d ⟶ ð
		9 b ⟶ b	9 p ⟶ p
		9 d ⟶ d	9 t ⟶ t

LET'S TAKE A LOOK IN THE MAXENT GRAMMAR TOOL

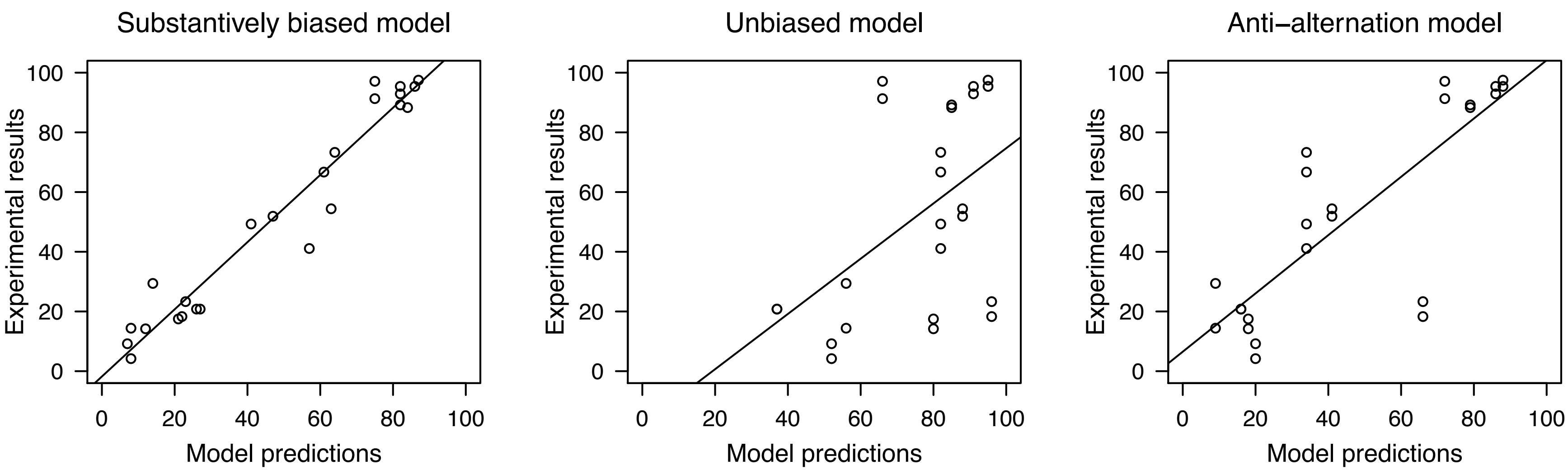
MODEL PREDICTIONS

Experiment 1: Potentially Saltatory condition				
	Experimental result	Substantively biased model	Unbiased model	Anti-alternation model
p ⟶ v	98	87	95	88
t ⟶ ð	95	86	95	88
b ⟶ v	73	64	82	34
d ⟶ ð	67	61	82	34
f ⟶ v	49	41	82	34
θ ⟶ ð	41	57	82	34

Experiment 1: Control condition				
	Experimental results	Substantively biased model	Unbiased model	Anti-alternation model
b ⟶ v	88	84	85	79
d ⟶ ð	89	82	85	79
p ⟶ v	18	22	96	66
t ⟶ ð	23	23	96	66
f ⟶ v	14	12	80	18
θ ⟶ ð	18	21	80	18

White, 2013, under review

OVERALL RESULTS



	All data		Middle two-thirds of data	
Model	r^2	Log likelihood	r^2	Log likelihood
Substantively biased	.94	–1722	.91	–1253
Unbiased	.25	–2827	.15	–2247
Anti-alternation	.67	–1926	.36	–1446

White, 2013, under review

REFERENCES

- Becker, Michael, Nihan Ketrez, & Andrew Nevins. (2011). The surfeit of the stimulus: Analytic biases filter lexical statistics in Turkish laryngeal alternations. *Language, 87,* 84–125.
- Becker, Michael, Andrew Nevins, & Jonathan Levine. (2012). Asymmetries in generalizing alternation to and from initial syllables. *Language, 88,* 231–268.
- Ernestus, Mirjam & R. Harald Baayen. (2003). Predicting the unpredictable: Interpreting neutralized segments in Dutch. *Language*, *79*, 5–38.
- Goldwater, Sharon, & Mark Johnson. (2003). Learning OT constraint rankings using a maximum entropy model. In *Proceedings of the Workshop on Variation within Optimality Theory.*
- Hayes, Bruce, & Colin Wilson. (2008). A maximum entropy model of phonotactics and phonotactic learning. *Linguistic Inquiry, 39,* 379–440.
- Hayes, Bruce, Kie Zuraw, Péter Sipár, Zsuzsa Londe. (2009). Natural and unnatural constraints in Hungarian vowel harmony. *Language, 85*, 822–863.
- Kawahara, Shigeto. (2006). A faithfulness ranking projectd from a perceptibility scale: the case of [+voice] in Japanese. *Language, 82,* 536–574.
- Legendre, G., Miyata, Y., & Smolensky, P. (1990). Harmonic Grammar — A formal multi level connectionist theory of linguistic well formedness: Theoretical foundations. *Proceedings of the Twelfth Annual Conference of the Cognitive Science Society* (385–395). Cambridge, MA.
- Pater, Joe. (2009). Weighted constraints in generative linguistics. *Cognitive Science, 33,* 999–1035.
- Potts, C., Pater, J., Jesney, K., Bhatt, R., & Becker M. (2009). Harmonic grammar with linear programming: from linguistic systems to linguistic typology. Ms., University of Massachusetts, Amherst.
- White, James. (2013). *Bias in phonological learning: Evidence from saltation*. Ph.D. dissertation, UCLA.
- White, James. (2014). Evidence for a learning bias against saltatory phonological alternations. *Cognition, 130,* 96–115.
- Wilson, Colin. (2006). Learning phonology with substantive bias: An experimental and computational study of velar palatalization. *Cognitive Science, 30,* 945–982.