Computational Phonology, class 7: Modeling typology

Adam Albright

CreteLing 2022 — July 2022

SCAN ME

creteling2022.computational.phonology.party

Universal Con?

- Assumption so far: the set of constraints is known ahead of time
  - Prince and Smolensky (2004): fixed and universal (UG)
- Assuming universal Con has been productive in reasoning about phonological typology
  - We'll consider the question of how one might learn constraints in the next class
- Factorial typology
  - Space of possible phonological grammars $=$ permutations of ranked constraints
- How to assess factorial typology?
  - Empirical typology contains all and only the predicted patterns?
  - Overgeneration, undergeneration

The role of learning

- Every attested adult grammar must be representable in the theory
- Not every representable grammar is necessarily expected to be attested
- Attested grammars must also be *acquirable*
  - Learners can arrive at it (priors+data)
- Predicted typology = representable languages, filtered through learning

Assessing fit

- Undergeneration: fatal, if true (empirical adequacy)
  - But apparent exceptions merit careful scrutiny
- Overgeneration
  - Accidental gaps? (low expected probability, or historical ‘accident’)
  - Additional pressures, such as learnability

Assigning stress without feet

- Constraints on position of stresses and intervals between them

ALIGN-L, ALIGN-R	Assign one * for each σ separating stress from the L/R edge of the word
NONFINALITY	Assign one * for stress on the final σ
*LAPSE	Assign one * for each sequence of two stressless σ's
*LAPSE-L/R	Assign one * if neither of the initial/final two σ's is stressed
*EXTENDLAPSE	Assign one * for each sequence of three stressless σ's
*EXTLAPSE-L/R	Assign one * if none of the initial/final three σ's is stressed

Assigning stress without feet (*cont.*)

- E.g., antepenultimate stress

/σσσσσσ/	*ExtLapse(R)	*ExtLapse(L)	Align(L)	Align(R)
a. σ́σσσσσσ	*! W		L	****** W
b. σσ́σσσσσ	*! W		* L	***** W
c. σσσ́σσσσ	*! W		** L	**** W
d. σσσσ́σσσ	*! W	*	*** L	*** W
☞ e. σσσσσ́σσ		*	****	**
f. σσσσσσ́σ		*	*****! W	* L
g. σσσσσσσ́		*	*****!* W	L

- The insight behind the analysis
    - Stress wants to be as far left as possible (Align(L) ≫ Align(R))
    - But it can't go further than the 3rd σ from the end (ExtLapse(R) ≫ Align(L))

Staubs (2014a) on Stress windows

Stress windows

- Languages may have stress at a fixed distance from an edge, but not all distances are observed

Attested:	final	σ́, σσ́, σσσ́, σσσσ́, σσσσσ́, σσσσσσ́
Attested:	penultimate	σ́, σ́σ, σσ́σ, σσσ́σ, σσσσ́σ, σσσσσ́σ
Attested:	antepenultimate	σ́, σ́σ, σ́σσ, σσ́σσ, σσσ́σσ, σσσσ́σσ
Unattested:	preantepenultimate (and beyond)	σ́, σ́σ, σ́σσ, σ́σσσ, σσ́σσσ, σσσ́σσσ

Stress windows (*cont.*)

- Fixed Con solution: limit on constraints
  - *ExtendedLapse(R) penalizes όσσσ, prefers antepenultimate όσσ
  - Hypothesis: no equivalent *ExtendedExtendedLapse(R) (*όσσσσ)

/σσσσσσ/	*ExtLapse(R)	*ExtLapse(L)	Align(L)	Align(R)
a. όσσσσσσ	*! W		L	****** W
b. σόσσσσσ	*! W		* L	***** W
c. σσόσσσσ	*! W		** L	**** W
d. σσσόσσσ	*! W	*	*** L	*** W
☞ e. σσσσόσσ		*	****	**
f. σσσσσόσ		*	*****! W	* L
g. σσσσσσό		*	*****!* W	L

Staub's general claim

- Phonological grammar allows for window lengths of any(?) size
- However, longer window lengths are harder to learn, because the data needed to distinguish them from short windows is rare
  - Long windows show up only in long words (following Prince 1993, Pater 2009)
- Harder to learn = lower probability that an individual learner will acquire it successfully
- Iterated across generations: frequency of such patterns is reduced or eliminated

Illustrating the idea

- Staubs uses MaxEnt models with foot-based constraints, which intrinsically predict 4+σ windows
- To keep things consistent, I'll recast the problem into the constraints already presented above, adding an additional constraint to allow for 4σ windows
  - *3-Lapse(R/L): penalizes stressless σσσσ#, #σσσσ
  - With this notation, *ExtLapse = *2-Lapse

Amount of ranking data

/σσ/	*3-Lapse(L)	*3-Lapse(R)	*2-Lapse(L)	*2-Lapse(R)	Align-L	Align-R
a. όσ						1
b. σό					1	

/σσσ/	*3-Lapse(L)	*3-Lapse(R)	*2-Lapse(L)	*2-Lapse(R)	Align-L	Align-R
a. όσσ						2
b. σόσ					1	1
c. σσό					2	

/σσσσ/	*3-Lapse(L)	*3-Lapse(R)	*2-Lapse(L)	*2-Lapse(R)	Align-L	Align-R
a. όσσσ				1		3
b. σόσσ					1	2
c. σσόσ					2	1
d. σσσό			1		3	

/σσσσσ/	*3-Lapse(L)	*3-Lapse(R)	*2-Lapse(L)	*2-Lapse(R)	Align-L	Align-R
a. όσσσσ		1		1		4
b. σόσσσ				1	1	3
c. σσόσσ					2	2
d. σσσόσ			1		3	1
e. σσσσό	1		1		4	

Making the model sensitive to amount of data

- Gradual learning: fewer relevant W/L pairs means fewer updates
- Priors/regularization: less data means the prior gets more of a say

Why would priors favor shorter windows?

- Given ambiguous data, why would learners favor shorter windows?
    - Train on 4σ window, but mislearn as 3σ window?
- 4σ window: *3-Lapse(R) ≫ Align-L ≫ *2-Lapse(R)
- 3σ window: *2-Lapse(R) ≫ Align-L
    - Ranking of *2-Lapse(R) doesn't matter
- Idea: if constraints are promoted, *2-Lapse(R) has more data favoring its promotion
- Even if data has words of all lengths, more inputs promote *2-Lapse(R)
    - 4σ and up for *2-Lapse, vs. 5σ and up for *3-Lapse

Midpoint Pathology

The midpoint pathology (Kager, 2012; Stanton, 2016)

- Short words: can satisfy both *(EXTENDED)LAPSE(L) and *(EXTENDED)LAPSE(R), by keeping stress towards the middle of the word

/σσσσσ/	*EXTLAPSE(L)	*EXTLAPSE(R)
☞ a. σσόσσ		
b. σσσσό	*! W	
c. όσσσσ		*! W

- Longer words: can't satisfy both, so satisfy the higher-ranked one with stress at the relevant edge

/σσσσσσσ/	*EXTLAPSE(L)	*EXTLAPSE(R)
a. σσσόσσσ	*! W	*
☞ b. όσσσσσσ		*
c. σσσσσσό	*! W	L

Example: a 'midpoint-stress' language

*ExtLapse(L) ≫ *ExtLapse(R) ≫ Align(L) ≫ Align(R)

2 syl	σ́σ
3 syl	σ́σσ
4 syl	σσ́σσ
5 syl	σσσ́σσ
6 syl	σ́σσσσσ
7 syl	σ́σσσσσσ
8 syl	σ́σσσσσσσ

- *ExtLapse(L/R) ≫ Align(L/R): stress can move inside word to avoid extended lapse
- *ExtLapse(L) ≫ *ExtLapse(R): when too long to satisfy both, it moves to the left side of the word
- Align(L) ≫ Align(R): when on the left side of the word, it falls on the very first syllable

Stanton’s observation

/σσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
☞ a. όσ				*
b. σό			*! W	L

/σσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
☞ a. όσσ				**
b. σόσ			*! W	* L
c. σσό			*!* W	L

/σσσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
a. όσσσ		*! W	L	*** W
☞ b. σόσσ			*	**
c. σσόσ			**! W	* L
d. σσσό	*! W		*** W	L

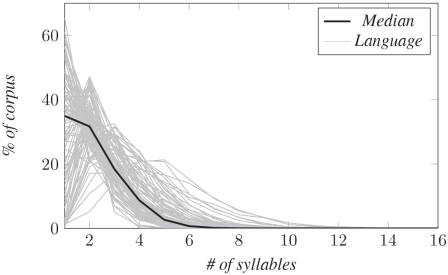
/σσσσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
a. όσσσσ		*! W	L	**** W
b. σόσσσ		*! W	* L	*** W
☞ c. σσόσσ			**	**
d. σσσόσ	*! W		*** W	* L
e. σσσσό	*! W		**** W	L

/σσσσσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
☞ a. όσσσσσ		*		*****
b. σόσσσσ		*	* W	**** L
c. σσόσσσ		*	** W	*** L
d. σσσόσσ	*! W	L	*** W	** L
e. σσσσόσ	*! W	L	**** W	* L
f. σσσσσό	*! W	L	***** W	L

/σσσσσσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
☞ a. όσσσσσσ		*		******
b. σόσσσσσ		*	* W	***** L
c. σσόσσσσ		*	** W	**** L
d. σσσόσσσ	*! W	*	*** W	*** L
e. σσσσόσσ	*! W	L	**** W	** L
f. σσσσσόσ	*! W	L	***** W	* L

- Clear evidence for Align(L) ≫ Align(R) in 2,3,4-syllable words
- Evidence for *ExtLapse(R) ≫ Align(L) from 5-syllable words
- Evidence for *ExtLapse(L) ≫ *ExtLapse(R) only from 6-syllable words and longer

On the relative scarcity of long words



#	Median
1	35%
2	32%
3	18%
4	9%
5	3%
6	1%
7	0%

FIGURE 1. Results of the survey of text corpora from 102 languages (see the appendices for more details).

- Rough estimate of relative proportion of words of different lengths in texts of 102 languages
- With a few notable exceptions, $\geq 6\sigma$ words are a very small proportion of the input
- Also: long words tend to be morphologically complex (may show other patterns)

Learning from short words

/σσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
☞ a. όσ				*
b. σό			* W	L

/σσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
☞ a. όσσ				**
b. σόσ			* W	* L
c. σσό			** W	L

/σσσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
a. όσσσ		* W	L	*** W
☞ b. σόσσ			*	**
c. σσόσ			** W	* L
d. σσσό	* W		*** W	L

/σσσσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
a. όσσσσ		* W	L	**** W
b. σόσσσ		* W	* L	*** W
☞ c. σσόσσ			**	**
d. σσσόσ	* W		*** W	* L
e. σσσσό	* W		**** W	L

Learning from short words

/σσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
☞ a. όσ				*
b. σό			* W	L

/σσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
☞ a. όσσ				**
b. σόσ			* W	* L
c. σσό			** W	L

/σσσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
a. όσσσ		* W	L	*** W
☞ b. σόσσ			*	**
c. σσόσ			** W	* L
d. σσσό	* W		*** W	L

/σσσσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
a. όσσσσ		* W	L	**** W
b. σόσσσ		* W	* L	*** W
☞ c. σσόσσ			**	**
d. σσσόσ	* W		*** W	* L
e. σσσσό	* W		**** W	L

Learning from short words

/σσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
☞ a. όσ				*
b. σό			* W	L

/σσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
☞ a. όσσ				**
b. σόσ			* W	* L
c. σσό			** W	L

/σσσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
a. όσσσ		* W	L	*** W
☞ b. σόσσ			*	**
c. σσόσ			** W	* L
d. σσσό	* W		*** W	L

/σσσσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
a. όσσσσ		* W	L	**** W
b. σόσσσ		* W	* L	*** W
☞ c. σσόσσ			**	**
d. σσσόσ	* W		*** W	* L
e. σσσσό	* W		**** W	L

Two possible refinements

(☹= preferred by generating grammar, losing in acquired grammar)

*ExtLapse(L) ≫ *ExtLapse(R)

/σσσσσσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
☞ a. όσσσσσσ		*		******
b. σόσσσσσ		*	* W	***** L
c. σσόσσσσ		*	** W	**** L
d. σσσόσσσ	*! W	*	*** W	*** L
e. σσσσόσσ	*! W	L	**** W	** L
f. σσσσσόσ	*! W	L	***** W	* L
g. σσσσσσό	*! W	L	****** W	L

/σσσσσσσσ/	*ExtLapse(L)	*ExtLapse(R)	Align(L)	Align(R)
☞ a. όσσσσσσσ		*		*******
b. σόσσσσσσ		*	* W	****** L
c. σσόσσσσσ		*	** W	***** L
d. σσσόσσσσ	*! W	*	*** W	**** L
e. σσσσόσσσ	*! W	*	**** W	*** L
f. σσσσσόσσ	*! W	L	***** W	** L
g. σσσσσσόσ	*! W	L	****** W	* L
h. σσσσσσσό	*! W	L	******* W	L

Midpoint system

2 syl	όσ	5 syl	σσόσσ
3 syl	όσσ	6 syl	όσσσσσ
4 syl	σόσσ	7 syl	όσσσσσσ

*ExtLapse(R) ≫ *ExtLapse(L)

/σσσσσσσ/	*ExtLapse(R)	*ExtLapse(L)	Align(L)	Align(R)
☹ a. όσσσσσσ	*! W		L	****** W
b. σόσσσσσ	*! W		* L	***** W
c. σσόσσσσ	*! W		** L	**** W
d. σσσόσσσ	*! W	*	*** L	*** W
☞ e. σσσσόσσ		*	****	**
f. σσσσσόσ		*	*****! W	* L
g. σσσσσσό		*	*****!* W	L

/σσσσσσσσ/	*ExtLapse(R)	*ExtLapse(L)	Align(L)	Align(R)
☹ a. όσσσσσσσ	*! W			******* W
b. σόσσσσσσ	*! W		* L	****** W
c. σσόσσσσσ	*! W		** L	***** W
d. σσσόσσσσ	*! W	*	*** L	**** W
e. σσσσόσσσ	*! W	*	**** L	*** W
☞ f. σσσσσόσσ		*	*****	**
g. σσσσσσόσ		*	******! W	* L
h. σσσσσσσό		*	******!* W	L

Antepenultimate stress

2 syl	όσ	5 syl	σσόσσ
3 syl	όσσ	6 syl	σσσόσσ
4 syl	σόσσ	7 syl	σσσσόσσ

Ambiguity in short words

- So far: based on $< 6\sigma$ data, learners exposed to a midpoint system *might* infer antepenultimate stress
    - Hoped-for claim: midpoint system is ‘unstable’, and learners may learn antepenultimate stress instead
- Problem: ambiguity cuts both ways! Learners exposed to antepenultimate stress might assume that they are learning a midpoint system
    - Actual prediction: variability or changes in both directions
- Where does the antepenultimate bias come from?

The learning algorithm matters

- RCD doesn't explain antepenultimate bias, because in short words, *ExtLapse(L) and *ExtLapse(R) are 'W-only' constraints $\Rightarrow$ remain highly ranked
- Stanton's conjecture: human learners actually use a ranking algorithm that doesn't just demote L's, but also promotes W's (Boersma, 1997; Magri, 2012a)

The learning algorithm matters (*cont.*)

- Why this will help:
  - Short words give lots of evidence for ALIGN(L) $\gg$ ALIGN(R)
  - If the learner demotes ALIGN(R) and *promotes* ALIGN(L), then ALIGN(L) will end up above other markedness constraints
  - Similarly, 4–5σ provide evidence for *EXTLAPSE(R) $\gg$ ALIGN(L), causing it to be promoted
  - Consequence: *EXTLAPSE(L) is ‘left in the dust’ (not promoted until you get 6+ syllable words)

Learning from short words: promotion and demotion

/σσ/	*ExtLapse(L)	*ExtLapse(R)	←Align(L)	Align(R)→
☞ a. σ́σ				*
b. σσ́			* W	L

/σσσ/	*ExtLapse(L)	*ExtLapse(R)	←Align(L)	Align(R)→
☞ a. σ́σσ				**
b. σσ́σ			* W	* L
c. σσσ́			** W	L

/σσσσ/	*ExtLapse(L)	*ExtLapse(R)	←Align(L)	Align(R)→
a. σ́σσσ		* W	L	*** W
☞ b. σσ́σσ			*	**
c. σσσ́σ			** W	* L
d. σσσσ́	* W		*** W	L

/σσσσσ/	*ExtLapse(L)	*ExtLapse(R)	←Align(L)	Align(R)→
a. σ́σσσσ		* W	L	**** W
b. σσ́σσσ		* W	* L	*** W
☞ c. σσσ́σσ			**	**
d. σσσσ́σ	* W		*** W	* L
e. σσσσσ́	* W		**** W	L

Learning from short words: promotion and demotion

/σσ/	ALIGN(L)←	*EXTLAPSE(L)	*EXTLAPSE(R)	→ALIGN(R)
☞ a. όσ				*
b. σό	* W			L

/σσσ/	ALIGN(L)←	*EXTLAPSE(L)	*EXTLAPSE(R)	→ALIGN(R)
☞ a. όσσ				**
b. σόσ	* W			* L
c. σσό	** W			L

/σσσσ/	ALIGN(L)←	*EXTLAPSE(L)	*EXTLAPSE(R)	→ALIGN(R)
a. όσσσ	L		* W	*** W
☞ b. σόσσ	*			**
c. σσόσ	** W			* L
d. σσσό	*** W	* W		L

/σσσσσ/	ALIGN(L)←	*EXTLAPSE(L)	*EXTLAPSE(R)	→ALIGN(R)
a. όσσσσ	L		* W	**** W
b. σόσσσ	* L		* W	*** W
☞ c. σσόσσ	**			**
d. σσσόσ	*** W	* W		* L
e. σσσσό	**** W	* W		L

Learning from short words: promotion and demotion

/σσ/	←ALIGN(L)→	*EXTLAPSE(L)	←*EXTLAPSE(R)	←ALIGN(R)→
☞ a. όσ				*
b. σό	* W			L

/σσσ/	←ALIGN(L)→	*EXTLAPSE(L)	←*EXTLAPSE(R)	←ALIGN(R)→
☞ a. όσσ				**
b. σόσ	* W			* L
c. σσό	** W			L

/σσσσ/	←ALIGN(L)→	*EXTLAPSE(L)	←*EXTLAPSE(R)	←ALIGN(R)→
a. όσσσ	L		* W	*** W
☞ b. σόσσ	*			**
c. σσόσ	** W			* L
d. σσσό	*** W	* W		L

/σσσσσ/	←ALIGN(L)→	*EXTLAPSE(L)	←*EXTLAPSE(R)	←ALIGN(R)→
a. όσσσσ	L		* W	**** W
b. σόσσσ	* L		* W	*** W
☞ c. σσόσσ	**			**
d. σσσόσ	*** W	* W		* L
e. σσσσό	**** W	* W		L

Learning from short words: promotion and demotion

/σσ/	*ExtLapse(R)←	←Align(L)→	*ExtLapse(L)	←Align(R)→
☞ a. όσ				*
b. σό		* W		L

/σσσ/	*ExtLapse(R)←	←Align(L)→	*ExtLapse(L)	←Align(R)→
☞ a. όσσ				**
b. σόσ		* W		* L
c. σσό		** W		L

/σσσσ/	*ExtLapse(R)←	←Align(L)→	*ExtLapse(L)	←Align(R)→
a. όσσσ	* W	L		*** W
☞ b. σόσσ		*		**
c. σσόσ		** W		* L
d. σσσό		*** W	* W	L

/σσσσσ/	*ExtLapse(R)←	←Align(L)→	*ExtLapse(L)	←Align(R)→
a. όσσσσ	* W	L		**** W
b. σόσσσ	* W	* L		*** W
☞ c. σσόσσ		**		**
d. σσσόσ		*** W	* W	* L
e. σσσσό		**** W	* W	L

*ExtLapse(R) ≫ Align(L) ≫ *ExtLapse(L) ≫ Align(R)

- This ranking works for words of 2–5 syllables
- But it predicts antepenultimate stress for longer words

/σσσσσσ/	*ExtLapse(R)	Align(L)	*ExtLapse(L)	Align(R)
☹ a. όσσσσσ	*! W	L	L	***** W
b. σόσσσσ	*! W	* L	L	**** W
c. σσόσσσ	*! W	** L	L	*** W
☞ d. σσσόσσ		***	*	**
e. σσσσόσ		****! W	*	* L
f. σσσσσό		****!* W	*	L

/σσσσσσσ/	*ExtLapse(R)	Align(L)	*ExtLapse(L)	Align(R)
☹ a. όσσσσσσ	*! W	L	L	****** W
b. σόσσσσσ	*! W	* L	L	***** W
c. σσόσσσσ	*! W	** L	L	**** W
d. σσσόσσσ	*! W	*** L	*	*** W
☞ e. σσσσόσσ		****	*	**
f. σσσσσόσ		*****! W	*	* L
g. σσσσσσό		*****!* W	*	L

- Learner trained on both midpoint and antepenultimate patterns learns an antepenultimate grammar
  - …until long words are encountered, if it's not too late

Stepping back: the approach, more generally

- Some unattested systems may be possible to capture grammatically, but are difficult to learn
- Goal: theory of grammatical learning that predicts that learners, when exposed to typical input from a 'difficult' pattern, systematically misacquire it as a different, more commonly attested pattern
- Potential to explain not only unattested systems, but also rare systems (which we can't exclude as impossible grammars, anyway)
- Converging evidence: acquisition data, learning in the lab?

Stress typology

Modeling typological frequency

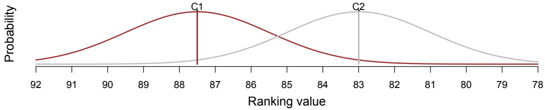
Why are some patterns common cross-linguistically, while others are rare or non-existent?

- So far: focus on why certain patterns are unattested
  - Can’t be expressed in theory
  - Can be expressed, but difficult to learn
- If learnability is a filter, should affect not only what is attested, but also relative frequencies

Common vs. rare patterns

Modeling relative frequency by sampling rankings

- Recall: stochastic ranking (Boersma, 1997; Boersma and Hayes, 2001)
  - Rankings are sampled from distributions at time of evaluation
  - Relative frequency of patterns is determined by ranking distributions



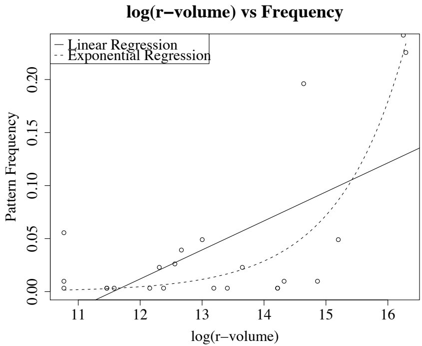
Predicting typological frequency by sampling rankings

An (overly) simple place to start

- Assume no fixed rankings
- Sample rankings from full factorial typology
- Relative freq of patterns $\propto$ probability of sampling rankings that generate them
  - If all rankings are equally probable, this equals the *number* of rankings that generate them
  - Bane and Riggle (2008); Riggle (2010): R-VOLUME

Predicting typological frequency by sampling rankings (*cont.*)

- Positive correlation for freq. of single stress systems: Bane and Riggle (2008, Fig. 6)



The role of learnability

- ‘Big Bang’ model of typology: distribution generated randomly, stable over time
- In actuality, languages change, in non-random ways
- Reanalysis: patterns are reinterpreted as similar (“confusable”; Bane and Riggle, 2008) patterns
- Reanalysis is directional
    - Certain patterns are difficult to acquire because the evidence that distinguishes them from other “preferred” patterns is rare in the input to learners (Staubs, 2014b; Stanton, 2016)
    - These patterns may be mislearned, becoming rare or even unattested over time

- Staubs, Stanton show that loss through reanalysis is useful for specific gaps in the attested typology (long windows, midpoint systems)
- Broader question: does iterated learning help improve the fit to attested typology, over simple r-volume?
    - Work in progress with Vighnesh Subramaniam (MIT)
    - Preview of result: yes, but discrepancies remain

QI single fixed stress

A fruitful testing grounds

Quantity insensitive fixed stress (Gordon, 2002; Heinz, 2007; Bane and Riggle, 2008)

- Can be modeled with modest number of constraints ($\approx$12)
- Candidates for length n: 2^n (ignoring primary placement)
- Large enough to be interesting, constrained enough to be tractable
- Databases: Gordon (2002), StressTyp2 (Goedemans and van der Hulst, 2009)

Focus here: single stress systems

- Granularity: descriptions of secondary stress frequently too coarse to differentiate patterns distinguished only in longer words
- Feasibility: reduced constraint set, much reduced candidate set (linear in length of word)
- Caveat: distinction between single stress vs. secondary stress unmentioned/uncertain may be murky; we return to this issue below

Reminder of constraints

Gordon (2002); Kager (2012); Stanton (2016)

ALIGN-L, ALIGN-R	Assign one * for each σ separating stress from the L/R edge of the word
NONFINALITY	Assign one * for stress on the final σ
*LAPSE	Assign one * for each sequence of two stressless σ's
*LAPSE-L/R	Assign one * if neither of the initial/final two σ's is stressed
*EXTENDLAPSE	Assign one * for each sequence of three stressless σ's
*EXTLAPSE-L/R	Assign one * if none of the initial/final three σ's is stressed

- Also assume unviolated constraints limiting outputs to single stress
  - CULMINATIVITY: every output has a primary stress
  - *STRESS: eliminates additional stresses

Factorial typology

- Factorial typology calculated using OTSoft (Hayes et al., 2013)
- Inputs: words of 2 through 8 syllables
- Output candidates: stress on each syllable
- $8! = 40320$ possible rankings (‘grammars’), and 8! logically possible combinations of output candidates (‘patterns’/‘languages’)
- Of these, only 48 patterns are predicted to occur, while 40272 patterns cannot be derived

Categorical attestation

- Counted fixed primary stress placement in StressTyp2
  - Simple positions: final, penultimate, antepenultimate; initial, peninitial, postpeninitial
  - Elaborations: penititial + non-finality
  - Two possible ‘midpoint’ systems (Içũa Tupi, Bhojpuri)
- None of the 40272 underivable patterns are attested
- Overgeneration: 39 patterns are predicted by the constraint set, but unattested
- All 9 attested patterns can be derived, though with very different frequencies

Challenge: does an iterated learning model explain why 39 patterns are unattested, and help predict the relative frequencies of attested patterns?

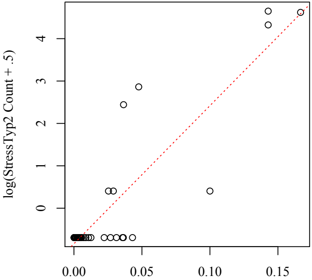
Ranking-Based Frequencies

Baseline: r-volume

- Along with Bane and Riggle (2008), Riggle (2010), Staubs (2014b), Stanton (2016) and others, we assume a baseline in which grammars are sampled randomly
- OT: probability of a language L $=$ its r-volume
$$\frac{\text{Number of rankings that derive L}}{\text{Total number of rankings}}$$
- Calculated r-volume for each of the 48 predicted patterns, using recursive technique proposed by Riggle (2010)

Comparison to empirical data

- As in Bane and Riggle (2008), compared r-volume with count in StressTyp2
- Bane and Riggle suggest that typological frequency may best be modeled by log(r-volume)
- However, it appears to us that r-volume linearly predicts the log of typological frequency, in line with Poisson models of count data



Assessment

- Correlation between (log) r-volume and frequencies of 9 attested patterns: Spearman's $\rho = .612$
- Qualitatively imperfect
  - 39 patterns with non-zero r-volume are empirically unattested
  - Although some of these may be accidental gaps, many are high enough to yield non-zero predicted counts in a database this size (Stanton, 2016)
- Quantitatively imperfect
  - Correlation artificially inflated by large number of unattested patterns
  - Relative frequencies of attested patterns are not well predicted

Hypothesis: iterated learning could improve both aspects of the model

- Difficult to learn patterns are reanalyzed and become rarer or even unattested over time
- Probability mass reassigned, changing relative predicted relative frequency of attested patterns

Learning-Based Frequencies

In order to test this hypothesis, we submitted each of the 48 derivable stress patterns to an iterated learning model, to see how probabilities are reassigned over time

The learning model

- Gradual Learning Algorithm (Boersma, 1997), as modified by Magri (2012b) to guarantee convergence
- Error-driven learning
  - Model receives input/output pairs: /σσσσ/, [σσσ́σ]
  - Model uses current grammar to derive predicted output
  - If incorrect, adjust ranking values to favor trained output
- Parameters
  - All markedness constraints start out equally ranked (100)
  - Plasticity: starts at 1, gradually decreases to .001 (learning slows with age; step=.004)
  - Learning trials: 2000

Training

- Trained the learning model on each of the 48 stress patterns
- Inputs: words of 2–8 syllables
- Shorter words presented more often than long words, according to mean frequencies of word lengths given by Stanton (2016)

2σ	32%	6σ	1%
3σ	18%	7σ	.1%
4σ	9%	8σ	.1%
5σ	3%		

- Each pattern run 1000 times, yielding a probability distribution over learning outcomes

Iterated learning

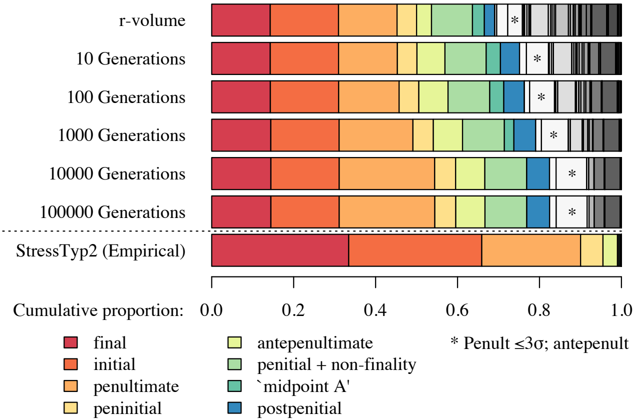
- Starting point: each pattern assigned probability according to baseline distribution (r-volume)
- Imperfect learning changes the probability distribution—e.g., when trained on Pattern 19 (antepenult. $\leq$ 5σ; else peninit.), the model learned:

Pattern 19	(antepenult $\leq$5σ; else peninit)	30.0%
Pattern 20	(antepenult. $\leq$5σ; else postpeninit)	34.1%
Pattern 21	(antepenultimate)	25.4%
Pattern 18	(antepenult. $\leq$5σ; else init)	8.9%
Pattern 23	(antepenult $\leq$4σ; else peninit)	.9%
Patterns 24, 7	(initial, etc.)	$<$1%

- Generations
  - Baseline (r-volume) distribution reassigned according to learned distributions
  - Iterated for 100,000 generations

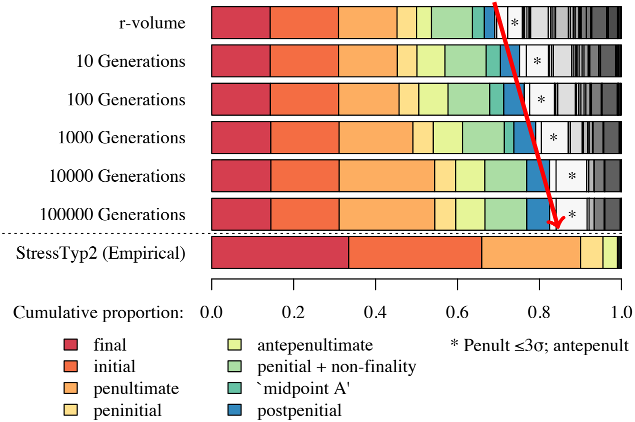
Results

Promising: reduction of unattested patterns



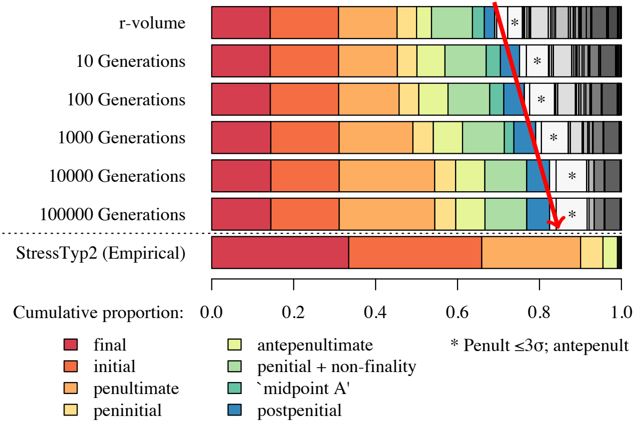
- Patterns attested in StressTyp2 (colored bars) increase in frequency over time

Promising: reduction of unattested patterns



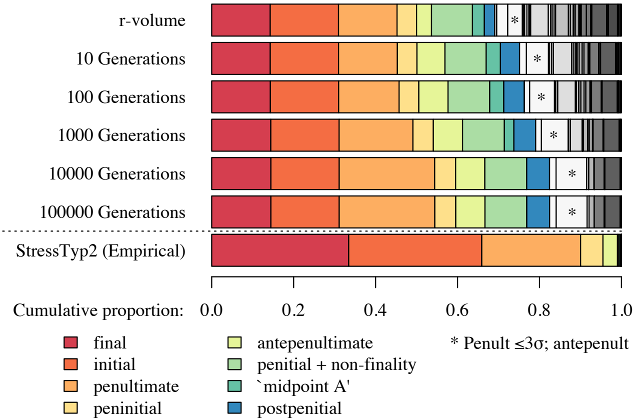
- Patterns attested in StressTyp2 (colored bars) increase in frequency over time

Promising: reduction of unattested patterns



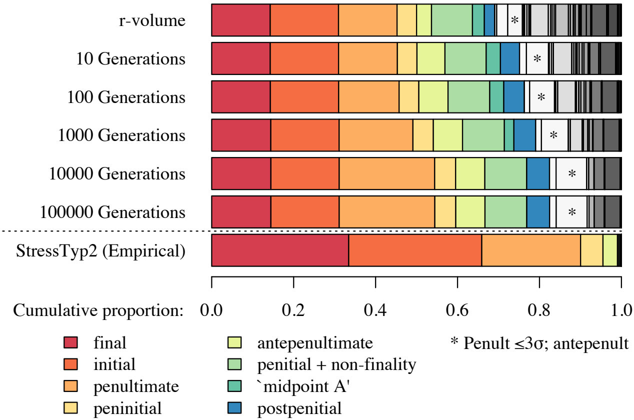
- Unattested patterns (gray bars) decrease in frequency over time

Promising: reduction of unattested patterns



- Some rare or controversially attested patterns also successfully nearly eliminated (‘midpoint A’)

Promising: reduction of unattested patterns



- Consistent with hypothesis that unattested patterns may be eliminated by learning (Blevins, 2004; Staubs, 2014b; Stanton, 2016)

Discrepancy 1: stubborn unattested patterns

- Although unattested patterns gradually decrease in relative frequency, some remain or even increase
- Example: '*' penult ≤3σ, antepenult longer (a 'midpoint system')

*LAPSE, *EXTLAPSE, *EXTLAPSE-R, NONFIN

$\gg$

ALIGN-L, *LAPSE-L, *EXTLAPSE-L

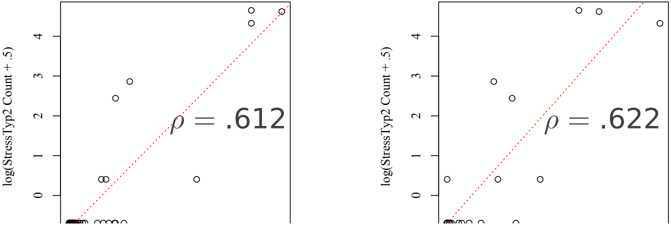
$\gg$

ALIGN-R, *LAPSE-R

- This pattern starts out with non-negligible probability (r-volume), and increases as other patterns are reanalyzed as it (does not rely heavily on long words)

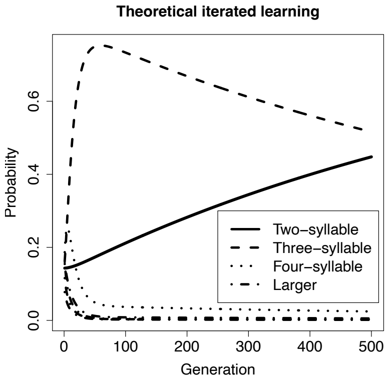
Discrepancy 2: unexpected beneficiaries

- Among attested (colored) patterns, predicted final distribution is *less* like attested (bottom row) than in the baseline (r-volume)
- Example: peninitial stress with non-finality ('σσ, σ'σσ)
  - Scarcely attested (Southern Paiute), but easily learned
- Post-peninitial stress is also a popular reanalysis for midpoint systems
- Result: no clear improvement in predictions for relative frequency of attested patterns



A similar effect from Staubs (2014a)

Fig 2.8 (p. 96)



- As longer windows are reanalyzed, frequency of antepenultimate stress increases dramatically

Local summary

- An attempt to scale up to a (slightly) more system-wide test of an idea that is intuitive and has proven useful in specific cases
  - Iterated grammatical learning winnows some patterns from the attested typology
  - Reanalysis shapes the relative frequency of attested patterns
- Modest support for winnowing, though not the whole story
  - Some unattested patterns remain; explanation for gap must lie elsewhere
- No clear support for learnability-based redistribution
  - In fact, r-volume remains marginally better predictor of typological frequency

What does a modeler do in this situation?

Wrong constraints?

- Hyde (2008, 2015): redefine ALIGN
- Kager (2012): reject *LAPSE because it can generate midpoint systems
  - Stanton (2016) addresses some but not all of these
- Switch to foot-based constraints?
  - Could eradicate 'Penult $\leq$3σ else antepenult' problem
  - May incorrectly rule out potential midpoint-type systems (Içũa Tupi, Bhojpuri)
  - Role of learnability far more difficult to explore: Hidden Structure (Tesar and Smolensky, 2000; Jarosz, 2013; Boersma and Pater, 2016)

R-volume as a prior?

- R-volume is not perfect, but striking that it does as well as it does. What creates this distribution?
- Bayesian inference: $P(H|D) \propto P(D|H) * P(H)$
  - Suppose 'H' is 'grammar generating pattern *n*' (aggregate over equivalent rankings)
  - Learners may retain H, even when likelihood ($P(D|H)$) is not especially high
  - Acoustic ambiguity of stress makes it especially prone to strong priors?
- Yang, Albright and Feldman (2022) Assessing the learnability of process interactions using grammatical spaces. (*Cogsci* paper)

A couple lessons

- The importance of scaling up
  - Prosthetic thought: calculating typology and r-volume, multiple runs of iterated learning
  - Results are not always what one expects
- Baselines
  - Results here aren't a particularly accurate model of typology, but they help point to some areas for future improvement
  - Empirical: accuracy of relative frequencies in STRESSTYP?
  - Attention to mismatches may point to changes in the constraints or learning model

References

Bane, Max and Jason Riggle. 2008. Three correlates of the typological frequency of quantity-insensitive stress systems. In Proceedings of the Tenth Meeting of ACL Special Interest Group on Computational Morphology and Phonology, 29–38. Association for Computational Linguistics.

Blevins, Julette. 2004. Evolutionary Phonology: The Emergence of Sound Patterns. Cambridge University Press.

Boersma, Paul. 1997. How we learn variation, optionality, and probability. Proceedings of the Institute of Phonetic Sciences of the University of Amsterdam 21, 43–58. http://fon.hum.uva.nl/paul/.

References (*cont.*)

Boersma, Paul and Bruce Hayes. 2001. Empirical Tests of the Gradual Learning Algorithm. Linguistic Inquiry 32(1), 45–86.

Boersma, Paul and Joe Pater. 2016. Convergence properties of a gradual learning algorithm for Harmonic Grammar. In Harmonic Serialism and Harmonic Grammar, ed. by John McCarthy and Joe Pater, 389–434. Sheffield: Equinox.

Goedemans, Rob and Harry van der Hulst. 2009. Stresstyp: A database for word accentual patterns in the world's languages. The use of databases in cross-linguistics research 235–282.

Gordon, Matthew. 2002. A factorial typology of quantity-insensitive stress. Natural Language & Linguistic Theory 20(3), 491–552.

References (*cont.*)

Hayes, Bruce; Bruce Tesar; and Kie Zuraw. 2013. OTSoft 2.5 software package. http://www.linguistics.ucla.edu/people/hayes/otsoft/.

Heinz, Jeffrey. 2007. Inductive learning of phonotactic patterns. Ph.D. thesis, UCLA.

Hyde, Brett. 2008. Alignment continued: distance-sensitivity, order-sensitivity, and the midpoint pathology. ROA 998.

—. 2015. The midpoint pathology: What it is and what it isn't. ROA 1231.

Jarosz, Gaja. 2013. Learning with hidden structure in Optimality Theory and Harmonic Grammar: Beyond Robust Interpretive Parsing. Phonology 30, 27–71.

References (*cont.*)

Kager, René. 2012. Stress in windows: Language typology and factorial typology. Lingua 122, 1454–1493.

Magri, Giorgio. 2012a. Convergence of error-driven ranking algorithms. Phonology 29(2), 213–269.

—. 2012b. Convergence of error-driven ranking algorithms. Phonology 29(2), 213–269.

Prince, Alan and Paul Smolensky. 2004. Optimality Theory: Constraint Interaction in Generative Grammar. Blackwell Publishing.

Riggle, Jason. 2010. Sampling rankings. ROA-1075 .

Stanton, Juliet. 2016. Learnability shapes typology: the case of the midpoint pathology. Language 92(4), 753–791.

References (*cont.*)

Staubs, Robert D. 2014a. Computational modeling of learning biases in stress typology. Ph.D. thesis, University of Massachusetts, Amherst.

Staubs, Robert D. 2014b. Computational modeling of learning biases in stress typology .

Tesar, Bruce and Paul Smolensky. 2000. Learnability in Optimality Theory. Cambridge, MA: MIT Press.