Learning a frequency-matching grammar together with lexical idiosyncrasy: MaxEnt versus mixed-effects logistic regression

SPEAKERS KNOW AGGREGATE GENERALIZATIONS AND IDIOSYNCRASIES

1. Language learners frequency match to statistical generalizations across the lexicon

- E.g., Hungarian vowel harmony (Hayes & Londe 2006): dative forms takes -nek or -nok, depending on backness of preceding stem vowel. Stems ending in...
  - front V tend to take -nek: [kɛrt-nek] ‘garden’-DAT, [yʃ]-nek] ‘cauldron’-DAT
  - back V tend to take -nok: [ɔblɔk-nok] ‘window’-DAT, [bi:ro:-nok], ‘judge’-DAT
- Corpus study of monosyl. stems ending in front, unrounded V: 92% take -nek; 8% -nok.
- In wug tests, speakers presented with fake monosyllabic stems with a front unrounded vowel, in aggregate, closely frequency-matched to the 8% -nok rate.

(1) [-nek -nok] (Hayes & Londe 2006)

<table>
<thead>
<tr>
<th>Corpus rate:</th>
<th>92%</th>
<th>8%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wug test rate:</td>
<td>93%</td>
<td>7%</td>
</tr>
</tbody>
</table>

2. But language learners also know lexical idiosyncrasies

- Speakers know which attested words harmonize, versus not (Hayes & Londe 2006).
- French speakers even track morpheme-specific rates of liaison (Zymet 2018).

3. Language learners thus internalize nested hierarchy of generalizations:

(2)

<table>
<thead>
<tr>
<th></th>
<th>Overall harmony rate</th>
<th>Aggregate generalization</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>92%</td>
<td>HARMONIZE</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Harmony rate after /kɛrt/</th>
<th>Harmony rate after /tsi:m/</th>
<th>Harmony rate after /hi:d/</th>
<th>Granular generalizations</th>
</tr>
</thead>
<tbody>
<tr>
<td>100%</td>
<td>100%</td>
<td>0%</td>
<td></td>
</tr>
</tbody>
</table>

- Recent MaxEnt model (Moore-Cantwell & Pater 2016, Zuraw & Hayes 2017, Tanaka 2017):
  - General constraint to frequency match general trend across the lexicon (HARMONIZE)
  - Lexical constraints for specific attested words (HARMN(kɛrt), DON’T-HARMN(hi:d))

---

4. Today: Modeling learning of frequency-matching grammar with lexical idiosyncrasy

- Learning simulations reveal that \textbf{lexical constraints are too powerful in MaxEnt}:
  - A priori, general constraint and set of lexical constraints considered \textit{equally viable} hypotheses about the data in MaxEnt;
  - at high levels of learning, lexical constraints come to explain every form in dataset, rendering the general constraint superfluous and ineffective.
  - General constraint weight plummets to zero, failing to predict learners’ frequency-matching abilities in wug tests.

- \textbf{Solution}: Switch from MaxEnt—essentially single-level logistic regression model—to hierarchical \textit{MIXED-EFFECTS LOGISTIC REGRESSION MODEL}.
  - General/lexical constraints no longer equal: general constraints preserved as fixed effects; lexical constraints form random effect.
  - Hierarchical model captures \textit{hierarchy of generalizations}: aggregate trend + idiosyncrasies of individual words.
  - We apply mixed model to variable Slovenian palatalization—with promising results.

MAXENT: THE GRAMMAR-LEXICON BALANCING PROBLEM

- Constraints have numerical weights instead of rankings;
- surface forms assigned probabilities as function of weights.
- Learning rooted in \textit{accuracy} and \textit{simplicity}: model takes constraints, finds best weights it can to fit overall rates in dataset; useless constraints discarded—weight set to zero.

- \textbullet{} But MaxEnt fits to \textit{overall rates}; investigators hadn’t tried to get MaxEnt to also learn which words are un/exceptional until recently. The new approach:
  - General constraints for overall trend, lexical constraints for specific-word behavior

6. Does the MaxEnt approach to learning frequency matching & idiosyncrasy work?
- Suppose we have 46 regulars, 4 irregulars—irregularity rate of \textbf{8\%}.
- 3 constraints: BEREG, BELEX(regulars), BELEX(irregulars) initiated at 0 weight
- If we want to learn the dataset better? Multiply frequencies by 10. Worse? By 0.1.
- (Caveat: introduced \textit{a little} variability: 0.001\% /regs/ surface [irreg]; 0.001\% /irregs/ as [reg])

<table>
<thead>
<tr>
<th>UR</th>
<th>SR</th>
<th>Freq.</th>
<th>BEREG 0</th>
<th>BELEX(reg) 0</th>
<th>BELEX(irreg) 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>/Regular/</td>
<td>Regular:</td>
<td>≈ 46</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Irregular:</td>
<td>≈ 0</td>
<td>-1</td>
<td>-1</td>
<td></td>
</tr>
<tr>
<td>/Irregular/</td>
<td>Regular:</td>
<td>≈ 0</td>
<td></td>
<td></td>
<td>-1</td>
</tr>
<tr>
<td></td>
<td>Irregular:</td>
<td>≈ 4</td>
<td></td>
<td>-1</td>
<td></td>
</tr>
</tbody>
</table>

\textit{Table 1: MaxEnt input}
We want MaxEnt to learn weights such that:
- in wug test, irregular form picked \( \approx 8\% \) of time;
- \textit{attested} words (=words in learner input) are pronounced correctly \( \approx 100\% \) of time.
- \( w(\text{BEREG}) = 4, w(\text{BELEX-reg}) = 3, w(\text{BELEX-irreg}) = 11 \) gives great results.

But does MaxEnt \textit{learn} good weights from the input? Let’s run learning simulation using Excel Solver, which can fit parameters of nonlinear models (Fylstra et al. 1998, Harris 1998):
- Trial run by using data in Table 1,
- and multiplying frequencies of the dataset by a small factor (0, 0.001, etc.)—we call this “childhood”. We learn poor weights \( w(\text{BELEX-irreg})=0.5 \) that don’t fit the data.
- After each trial, increase frequency factor slightly, get new weights—“adolescence”.
- When frequencies get large and we think we have final weights—“adulthood”.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure1.png}
\caption{MaxEnt fails to learn generalization together with idiosyncrasy \( (\sigma = 100) \)}
\end{figure}

- With frequency factor 0, baby learns 0-valued weights, prefers 50/50 regular/irregular.
- As child grows (freq. factor 0.0001), rapidly starts to learn regulars, slowly tackling irregulars.
- Early in learning, \textit{BEREG} is used to explain much of the variation—we see that with low nonce irregularity rate.
- But eventually \textit{BELEX} constraints grow very high, coming to explain entire set of attested data. \textit{BEREG} comes to explain increasingly \textit{less} of data, eventually \textit{perishing}.
- By adulthood (freq. mult. 1000), \textit{BEREG sinks to 0, rendered superfluous/ineffective.}
- At that point, the learner selects regulars/irregulars at 50/50 rate in wug tests—forgetting the grammar entirely. See Appendix for simulation output numbers.
7. **Hence, the GRAMMAR-LEXICON BALANCING PROBLEM. In MaxEnt...**

- A priori, general constraint/set of lexical constraints *equally viable hypotheses* about data,
- **consequently lexical constraints too powerful**: lexical constraints learn each word’s behavior before general constraint matches overall 8% trend, at which point frequency matching ceases and the general constraint becomes ineffective.
- No phonological learning; just lexical learning. Implausible that speakers fail wug tests once they learn lexicon (see Shademan 2007 for learning in elderly).
- We need a theory that, while accounting for idiosyncrasies, *preserves grammar*.

- We search for model possessing **GENERALITY BIAS**: general, grammatical constraints must be privileged to lexical constraints in the learning process.
- Adjusting MaxEnt penalty term does not work: dividing $\sigma$’s by 10 = dividing freq. multiplier by 100—merely *delays* overfitting (see Appendix).
- High $\sigma$(BEREG)/low $\sigma$(BELEX) so far does not work; overfits at higher multiplier.

**LEARNING LEXICAL VARIATION WITH MIXED-EFFECTS LOGISTIC REGRESSION**

8. **What about the hierarchical MIXED-EFFECTS LOGISTIC REGRESSION model?**

- Similar to binomial logistic regression, except constraints hierarchically arranged as follows:
  - **Fixed effects**: those constraints that we are actually interested in—e.g., phonological constraints, yielding the statistical generalizations in the dataset
  - **Random effects**: constraints that capture the idiosyncrasies in the data—deviations from generalizations captured by fixed effects.
- We might call this **Mixed Effects Maximum Entropy Harmonic Grammar**.
- Used widely in science to capture trends & idiosyncrasies in variable datasets;
- Linguists employ random intercepts to measure by-word/lexical class idiosyncrasy (Fruehwald 2012, Zuraw & Hayes 2017, Smith & Moore-Cantwell 2017, *inter alia*);
- Shih & Inkelas (2016)/Shih (2018) even adopt multilevel model as theory of learner.

9. **Mixed models hierarchical: random effects “depreciated” relative to fixed effects**

We have a fixed effect—a general constraint—BEREGULAR, whose weight is estimated based on average harmony rate across the entire dataset—92%.

$$w(BE): \mu_{\text{all words}}$$

- We want this weight to accurately estimate the average rate across all words, as that would be a **frequency-matching grammar**, mimicking human behavior in wug tests.

We have a random effect (random intercept) consisting of weights for lexical constraints:
- $w$BELEX–irreg1, for example, estimated by rate irregular1 (0.001) ...
- **and by overall rate across dataset**:
(3b) \( w(\text{BELEX–irreg1}): \lambda_{\text{irregular1}} \ast \mu_{\text{irregular1}} + (1 - \lambda_{\text{irregular1}}) \ast \mu_{\text{all words}} \)

Raudenbush & Bryk (2012), Snijders & Bosker (2012)

- \( \lambda \): value between 0 and 1, depends on size of the group: \( w(\text{BELEX–irreg1}) \) will be determined more by \( \mu_{\text{irregular1}} \) if data have lots of \text{irreg1} tokens rather than few.
  - Predicts more idiosyncrasy with frequent forms, but more grammatical behavior with infrequent forms (Morgan & Levy 2016, Moore-Cantwell & Smith 2016).
- **Think of mixed models as follows**: fixed effect weights predicts overall rate, and random effect weights predict **word-specific offsets** from overall rate.
- Source of the generality bias: lexical constraint weights depend on overall average rate.

10. Mixed model performs well on strict exceptionality dataset

We want the learning model to predict:
- With BEREG, the average rate across all Words in the dataset—hence a frequency-matching grammar
- With BELEX–reg/BELEX–irreg, the specific rates for every word.

We run a model of the dataset using the `glmer` function of the `lme4` package R.
- weight of BEREG is the general intercept
- weight of BELEX constraints are the coefficients of the levels of the random intercept.

To extract predicted nonce rate from model, you cannot simply plug \( w\text{BEREG} \) into inverse logit—rather, you must “average” over the levels of the random intercept (Pavlou et al. 2015).
- This involves a complex integral that cannot be calculated analytically;
- Zeger et al. (1998) provide a good approximation:
  - \( c \) is constant equal to \( \frac{16\sqrt{3}}{15\pi} \)
  - \( \tau^2 \) is variance of random intercept (14.77)

\[
\frac{\exp\left(\frac{w\text{BEREG}}{\sqrt{c^2\tau^2 + 1}}\right)}{1 + \exp\left(\frac{w\text{BEREG}}{\sqrt{c^2\tau^2 + 1}}\right)}
\]
### Results

<table>
<thead>
<tr>
<th>Word</th>
<th>wBELEX</th>
<th>Actual rate</th>
<th>Predicted rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>reg1</td>
<td>0.69</td>
<td>0.999</td>
<td>0.999</td>
</tr>
<tr>
<td>reg2</td>
<td>0.69</td>
<td>0.999</td>
<td>0.999</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>reg46</td>
<td>0.69</td>
<td>0.999</td>
<td>0.999</td>
</tr>
<tr>
<td>irreg47</td>
<td>-12.46</td>
<td>0.001</td>
<td>0.002</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>irreg50</td>
<td>-12.46</td>
<td>0.001</td>
<td>0.002</td>
</tr>
</tbody>
</table>

\[wBE\text{REG} = 6.167\]

**OVERALL IRREGULARITY RATE: 8%**

**PREDICTED NONCE IRREGULARITY RATE: 7.4%**

This model:
- Predicts word-specific rates—learns lexical effects.
- Frequency-matches overall rate—mimicking subjects in wug tests—without lexical constraints *starving* general constraint. *Grammar sustained after lexical learning.*

### 11. Mixed model performs well on dataset with different lexical rates

- Consider the following: twelve words, each with 1000 tokens, with the different tokens undergoing, say, harmony, at different rates.

<table>
<thead>
<tr>
<th>Word</th>
<th>Rate</th>
<th>Word</th>
<th>Rate</th>
<th>Word</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.00</td>
<td>5</td>
<td>0.30</td>
<td>9</td>
<td>1.00</td>
</tr>
<tr>
<td>2</td>
<td>0.00</td>
<td>6</td>
<td>0.80</td>
<td>10</td>
<td>1.00</td>
</tr>
<tr>
<td>3</td>
<td>0.10</td>
<td>7</td>
<td>0.90</td>
<td>11</td>
<td>1.00</td>
</tr>
<tr>
<td>4</td>
<td>0.20</td>
<td>8</td>
<td>1.00</td>
<td>12</td>
<td>1.00</td>
</tr>
</tbody>
</table>

Average over all rates: 0.61

**Table 2: propensity dataset**

- Two kinds of constraints:
  - APPLY (HARMONIZE), whose weight should frequency match to 61% overall rate
  - APPLY-Word1, ..., APPLY-Word12, assists with specific rates
(6) Results:

<table>
<thead>
<tr>
<th>Word</th>
<th>wLex constr.</th>
<th>Actual rate</th>
<th>Predicted rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Word1</td>
<td>-16.56</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>Word2</td>
<td>-16.56</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>Word3</td>
<td>-7.32</td>
<td>0.100</td>
<td>0.100</td>
</tr>
<tr>
<td>Word4</td>
<td>-6.51</td>
<td>0.200</td>
<td>0.200</td>
</tr>
<tr>
<td>Word5</td>
<td>-5.97</td>
<td>0.300</td>
<td>0.300</td>
</tr>
<tr>
<td>Word6</td>
<td>-3.74</td>
<td>0.800</td>
<td>0.800</td>
</tr>
<tr>
<td>Word7</td>
<td>-2.93</td>
<td>0.900</td>
<td>0.900</td>
</tr>
<tr>
<td>Word8</td>
<td>7.14</td>
<td>1.000</td>
<td>0.999</td>
</tr>
<tr>
<td>Word9</td>
<td>7.14</td>
<td>1.000</td>
<td>0.999</td>
</tr>
<tr>
<td>Word10</td>
<td>7.14</td>
<td>1.000</td>
<td>0.999</td>
</tr>
<tr>
<td>Word11</td>
<td>7.14</td>
<td>1.000</td>
<td>0.999</td>
</tr>
<tr>
<td>Word12</td>
<td>7.14</td>
<td>1.000</td>
<td>0.999</td>
</tr>
</tbody>
</table>

\[ w_{HARMONIZE} = 5.130 \]

OVERALL AVERAGE APPLICATION RATE: 0.61
PREDICTED APPLICATION RATE TO NONCE WORDS: 0.66

- I tried MaxEnt on this dataset:
  - Outcomes similar to other dataset, except \( w_{APPLY} \) vacillates/plummets to 0 at high levels of lexical learning—see Appendix.
  - See Zymet (2018) for further details.

APPLYING THE MIXED MODEL TO VARIABLE SLOVENIAN PALATALIZATION

- For example, only some suffixes trigger it.

(7a) Stem | Triggering suffix /-itsa/ | Non-triggering suffix /-inja/
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>luk-a</td>
<td>port-GEN</td>
<td>lutf-itsa, port-DIM</td>
</tr>
<tr>
<td>bog-a</td>
<td>god-GEN</td>
<td>bož-itsa, god-DIM</td>
</tr>
</tbody>
</table>


- Different palatalizing suffixes trigger at different rates, suggesting suffix identity plays role:

(7b) /luk-itʃ/, port-DIM | /luk-inʃ/, port-ABS | /luk-itsa/, port-DIM
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>lutf-itʃ, 18% (558/3147)</td>
<td>lutf-inʃ, 50% (50/100)</td>
<td>lutf-itsa, 98% (39/40)</td>
</tr>
<tr>
<td>luk-itʃ, 82% (2589/3147)</td>
<td>luk-inʃ, 50% (50/100)</td>
<td>luk-itsa, 2% (1/40)</td>
</tr>
</tbody>
</table>
Stems undergo at different rates before same suffix, suggesting stem identity plays role.

<table>
<thead>
<tr>
<th>Stem</th>
<th>Stem before diminutive -itsa</th>
<th>Undergoer</th>
<th>Vacillator</th>
<th>Non-undergoer</th>
</tr>
</thead>
<tbody>
<tr>
<td>oblak-a</td>
<td>‘cloud’-GEN</td>
<td>oblatʃ-itsa</td>
<td>‘cloud’-DIM</td>
<td>*</td>
</tr>
<tr>
<td>nɔg-a</td>
<td>‘leg’-GEN</td>
<td>nɔg-itsa ~ nɔʒ-itsa</td>
<td>‘leg’-DIM</td>
<td>*</td>
</tr>
<tr>
<td>jak-a</td>
<td>‘yak’-GEN</td>
<td>jak-itsa</td>
<td>‘yak’-GEN</td>
<td>*</td>
</tr>
</tbody>
</table>

10. Jurgec (2016) on Slovenian palatalization

Jurgec extracted words with velar-final stem + palatalizing suffix from two dictionaries:
- Dictionary of Standard Slovenian (Bajec 2000; 110,000 word types)
- Slovenian Orthographic Dictionary (Toporišč 2001; 130,000 word types).

To obtain token rates for each word, he fed them into Gigafida (Logar-Berginc et al. 2012):
- Text corpus w/ ~1.2 billion tokens from written sources ca. 1990–2011.
- His resulting data set included ~5.7 million tokens.

Jurgec suggests phonological factors condition variation in his data:
- Suffixes with front vocoids trigger more regularly
- Velars k, g undergo more regularly than x.
- Suffixes with š trigger less regularly.
- Palatalization regularly applies to avoid geminate in /...{k, g}+k/ (-k = -DIM)
- Palatalization blocked by distant postalveolars earlier in the stem.

Jurgec gives MaxEnt account of phonological conditioning; suffix idiosyncrasy encoded with [+/- Pal’n]—only picks out suffixes with any degree of palatalization.
- But he does observe suffix-specific rates in his study—lexical propensities left to further research.


I show that:
- Morphemes have LEXICAL PROPENSITIES: suffs trigger at different rates, and stems undergo at different rates, patterning across an entire spectrum ([0.7 Pal’n]).
- Mixed model encodes propensities while frequency matching to trends.

Extraction method similar to Jurgec:
- Words consisting of velar-final stems + palatalizing suffix extracted from Dictionary of Standard Slovenian.
- Each extracted stem concatenated with each of nine suffixes, creating hypot. words
- Fed each word into Gigafida, extracting frequencies/token rates
- Yielded ~3 million tokens of words either undergoing/not undergoing palatalization

I calculated palatalization rates for each suffix. /ag/ undergoes 22% of time before -/je/, /kak/ 99% of time; average rate before -/je/ is 88%.
What about stems? A histogram of rates across 246 stems occurring before at least four suffixes:

- Results suggest morphemes have **LEXICAL PROPENSITIES**: suffixes trigger at different rates, and stems undergo at different rates, patterning across an entire spectrum.

- We use mixed-effects logistic regression to encode morphemes on a spectrum ([0.7 Pal’n])—significantly improves model fit relative to binary scale ([+/- Pal’n]).
  - Models run using `glmer` functions of `lme4` package (Bates & Maechler 2011) in R.
In this handout, we focus on compare performance of following logistic models:

- **Baseline Model**, containing fixed effects for:
  - Stem-final velar identity (k, g, x)
  - Whether suffix begins with a front vocoid
  - Whether stem contains an earlier post-alveolar
  - Whether the suffix contains a post-alveolar affricate
  - (Contains random effect for whole word; thus we’re regressing over frequency-weighted types.)

- **Stem+Suffix Model**, containing:
  - all factors in Baseline Model
  - plus stem identity and suffix identity, **encoded as random intercepts**.

Models compared using Akaike Information Criteria (AIC; Akaike 1973), which scores models based on number of parameters and fit to the data: **lower score = better.**

- See Bolker et al. (2009) for justification on using this to compare mixed models.

**13. Results of Baseline Model**
- Stem-final velar identity significant: k > g (seemingly a faith effect: k → tj but g → ʒ)
- Geminate avoidance significant
- f...tf+ avoidance significant
- Suffix with ʦ significantly associated with lower rates
- frontvocoid not significant
- AIC: **8767.8**

<table>
<thead>
<tr>
<th></th>
<th>Estimate</th>
<th>Std. Err.</th>
<th>z value</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept ref: consg</td>
<td>3.95</td>
<td>0.47</td>
<td>8.29</td>
<td>&lt;0.001 ***</td>
</tr>
<tr>
<td>consx</td>
<td>1.59</td>
<td>0.66</td>
<td>2.41</td>
<td>0.015 *</td>
</tr>
<tr>
<td>consk</td>
<td>1.96</td>
<td>0.47</td>
<td>4.11</td>
<td>&lt;0.001 ***</td>
</tr>
<tr>
<td>kk</td>
<td>4.94</td>
<td>0.80</td>
<td>6.12</td>
<td>&lt;0.001 ***</td>
</tr>
<tr>
<td>frontvocoid</td>
<td>-0.52</td>
<td>0.39</td>
<td>-1.32</td>
<td>0.183 (n.s.)</td>
</tr>
<tr>
<td>S...S</td>
<td>-1.67</td>
<td>0.78</td>
<td>-2.12</td>
<td>0.033 *</td>
</tr>
<tr>
<td>suff.with.ts</td>
<td>-3.53</td>
<td>0.46</td>
<td>-7.55</td>
<td>&lt;0.001 ***</td>
</tr>
</tbody>
</table>

**Output 1a: Baseline Model results for Slovenian palatalization**

**13. Results of Stem+Suffix Model**
- Significant k > g effect and geminate effect, but no f...tf+ effect or suffix-with-ʦ effect
- **Stem and suffix variances highly positive**—suggest stem and suffix condition variation
- Stem variance bigger than suffix variance: maybe undergoers louder than triggers; or linearly-first-element bias; or just relative morpheme counts. Feel free to ask in Q&A.
- AIC value: **7801.5** — substantial reduction from Baseline Model’s **8767.8**;
Jesse Zymet
AMP ’18 Talk

Random effects:

- **Groups** Name        Variance Std.Dev.
  - stem   (Intercept) **68.06**   8.25
  - suffix (Intercept) **19.54**   4.42

Number of obs: 2940918; words: 4822; stems: 2720; suffixes: 9

Fixed effects:

<table>
<thead>
<tr>
<th></th>
<th>Estimate</th>
<th>Std. err.</th>
<th>z value</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>1.15</td>
<td>2.24</td>
<td>0.51</td>
<td>0.60</td>
</tr>
<tr>
<td>ref: consg</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| consx            | 2.36     | 1.00      | 2.35    | 0.02    | *
| consk            | 2.59     | 0.69      | 3.75    | <0.001  | ***
| k+k              | 7.94     | 1.32      | 6.01    | <0.001  | ***
| frontvocoid      | 2.72     | 3.01      | 0.90    | 0.366   |
| S...S            | -1.20    | 1.12      | -1.06   | 0.284   |
| suff.with.ts     | -1.88    | 3.58      | -0.52   | 0.598   |

**Output 1b:** Stem+Suffix Model results for Slovenian palatalization

14. **AICs suggest suffix and stem identities matter** ($p < 0.001$ by likelihood ratio test)
   - Baseline Model AIC: **8767.8**
   - Suffix-Only Model AIC: **8283.7**
   - Stem-Only Model AIC: **8128.9**
   - Stem+Suffix Model AIC: **7801.5**
   - See Zymet (2018) for further elaboration on all these models.

15. **Mixed model learns the phonology**, frequency matching to statistical trends
   - Matching to overall palatalization rate for $k$-final stems, and $g$-final stems
   - Predicts $k > g$ effect
   - Predicts geminate-avoiding palatalization

![Figure 3a](image-url): *model succeeds in predicting phonological trends*
16. The mixed model learns lexical propensities

- Fares well in predicting suffix-specific rates:

**Figure 3b:** model-predicted suffix rates generally match corpus rates

- I submit **mixed-effects logistic regression** as viable approach to modeling lexical variation—to learning of frequency-matching grammar with lexical propensities.

CONCLUSION

- Language learners internalize nested hierarchy of generalizations:
  - they can frequency match to aggregate statistical generalizations across the lexicon,
  - but also know which words are idiosyncratically exceptional, and which are not.

- MaxEnt/single-level regression doesn’t recognize **hierarchicality of generalities**. I suspect problem is broader than just lexical variation:
  - If learner knows two groups of data have different rates,
  - and averages over rates when encountering novel data lying outside both groups,
  - then how could we model this averaging if we have accurate model of group rates?
  - MaxEnt: specific constraints enough to explain data, general constraint superfluous.
Mixed-effects logit/mixed-effects MaxEnt surmounts balancing problem:
  - *Hierarchical* theory for a hierarchy of generalizations
  - Idiosyncratic effects of vocabulary subordinated to broad effects of grammar
  - Prior studies suggest it as potential model; today I give reason *why* this should be our theory of language competence.

Future questions I hope to work on:
  - How should hierarchical theory look—e.g., how to plug random intercept into theory?
  - Exactly what constraints/kinds of constraints should be considered fixed vs. random?
  - How to expand mixed-effects MaxEnt to cover more than just the binomial case?
  - How to get >2 levels of generalization?

**Appendix**

FAILED LEARNING SIMULATION IN MAXENT (strict exceptionality, 8% irregularity rate)

<table>
<thead>
<tr>
<th>Freq. multiplier</th>
<th>Be Reg</th>
<th>BeLex (regs)</th>
<th>BeLex (irregs)</th>
<th>Regular correct</th>
<th>Irreg. correct</th>
<th>Nonce irreg. rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.5000</td>
</tr>
<tr>
<td>0.0001</td>
<td>1.50</td>
<td>1.62</td>
<td>1.62</td>
<td>0.5304</td>
<td>0.9582</td>
<td>0.1815</td>
</tr>
<tr>
<td>0.001</td>
<td>1.27</td>
<td>2.88</td>
<td>2.88</td>
<td>0.8331</td>
<td>0.9845</td>
<td>0.2185</td>
</tr>
<tr>
<td>0.01</td>
<td>1.07</td>
<td>4.63</td>
<td>4.63</td>
<td>0.9723</td>
<td>0.9966</td>
<td>0.2548</td>
</tr>
<tr>
<td>0.1</td>
<td>0.41</td>
<td>6.22</td>
<td>5.82</td>
<td>0.9955</td>
<td>0.9986</td>
<td>0.3986</td>
</tr>
<tr>
<td>1</td>
<td>0.17</td>
<td>6.69</td>
<td>6.78</td>
<td>0.9986</td>
<td>0.9989</td>
<td>0.4551</td>
</tr>
<tr>
<td>10</td>
<td>0.02</td>
<td>6.87</td>
<td>6.89</td>
<td>0.9989</td>
<td>0.9989</td>
<td>0.4925</td>
</tr>
<tr>
<td>100</td>
<td>0</td>
<td>6.90</td>
<td>6.90</td>
<td>0.9989</td>
<td>0.9989</td>
<td>0.5000</td>
</tr>
<tr>
<td>1000</td>
<td>0</td>
<td>6.90</td>
<td>6.90</td>
<td>0.9989</td>
<td>0.9990</td>
<td>0.5000</td>
</tr>
</tbody>
</table>

*Table A1: MaxEnt learning simulation output numbers for strict exceptionality data*
Failed Learning Simulation in MaxEnt (propensities dataset, 61% applic’n rate)

Table A2: MaxEnt learning simulation output numbers for propensity data
OVERFITTING OUTCOME GENERAL ACROSS MAXENT PENALTY SETTINGS

- E.g., multiplying $\sigma$’s by 10 yields same result as multiplying frequency multiplier by 100.
- Evident in the table below, which presents results of a series of learning simulations of the strict exceptionality dataset from above (but only fitting the weight of BeREG to it).
- Hence decreasing $\sigma$ merely has the effect of delaying learner overfitting

<table>
<thead>
<tr>
<th></th>
<th>$\sigma = 1$</th>
<th></th>
<th>$\sigma = 10$</th>
<th></th>
<th>$\sigma = 100$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>irreg. rate</td>
<td>weight</td>
<td>irreg. rate</td>
<td>weight</td>
<td>irreg. rate</td>
</tr>
<tr>
<td>$m = 0.01$</td>
<td>0.4748</td>
<td>0.1008</td>
<td>0.1127</td>
<td>2.0629</td>
<td>0.0213</td>
</tr>
<tr>
<td>$m = 1$</td>
<td>0.1127</td>
<td>2.0629</td>
<td>0.0213</td>
<td>3.8258</td>
<td></td>
</tr>
<tr>
<td>$m = 100$</td>
<td>0.0213</td>
<td>3.8258</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table: identical learning outcomes across different values of $m$ and $\sigma$

- Manipulating $\mu$ also has no effect—yields same learning outcome as if we set $\mu = 0$.
- What about high $\sigma$(BeREG) and low $\sigma$(BeLEX)?
- I tried it on a few strict exceptionality datasets (but not including the one given in this handout...), and so far the results are negative:
- Setting $\sigma = 1,000$ for BeREGULAR and $\sigma = 10$ for the lexical constraints, for example, still yielded overfitting, albeit at a high frequency multiplier.

COEFFICIENTS FOR STEMS AND SUFFIXES IN SLOVENIAN

<table>
<thead>
<tr>
<th>Suffix</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>-ovje</td>
<td>-4.05</td>
</tr>
<tr>
<td>-ina</td>
<td>-1.27</td>
</tr>
<tr>
<td>-nat</td>
<td>-0.40</td>
</tr>
<tr>
<td>-itʃ</td>
<td>-0.38</td>
</tr>
<tr>
<td>-ts</td>
<td>-0.16</td>
</tr>
<tr>
<td>-itsa</td>
<td>0.16</td>
</tr>
<tr>
<td>-k</td>
<td>0.58</td>
</tr>
<tr>
<td>-je</td>
<td>1.48</td>
</tr>
<tr>
<td>-n</td>
<td>4.03</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Stems (sample)</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>trak-</td>
<td>-5.34</td>
</tr>
<tr>
<td>tramik-</td>
<td>0.00</td>
</tr>
<tr>
<td>tradicionalistik-</td>
<td>0.55</td>
</tr>
<tr>
<td>tragikomik-</td>
<td>1.14</td>
</tr>
<tr>
<td>travmatik-</td>
<td>1.30</td>
</tr>
<tr>
<td>tragik-</td>
<td>2.31</td>
</tr>
</tbody>
</table>

- Coefficients run the gamut, suggesting gradience.
- Suffix coefficients generally track suffix rates we saw toward the beginning.

References


