Evaluating an ensemble model of linguistic categorization on three variable morphological patterns in Hungarian

Péter Rácz (racz.peter.marton@ttk.bme.hu)
Cognitive Science Department, Faculty of Natural Sciences
Budapest University of Technology, Hungary

Péter Rebrus
HUN-REN Hungarian Research Centre for Linguistics
Eötvös Loránd University

Szilárd Tóth
Department of Theoretical Linguistics
Eötvös Loránd University, Budapest, Hungary

Abstract

We implemented two instance-based learners, the K-Nearest Neighbors model and the Generalized Context Model, and a rule-based learner, the Minimal Generalization Learner, adapted for linguistic data. We fit these on three distinct, variable patterns of word variation in Hungarian: paradigmatic leveling and vowel deletion in verbs and vowel harmony in nouns. We tested their predictions using a Wug task. The best learners were combined into an ensemble model for each pattern. All three learners explain variation in the test data. The best ensemble models of inflectional variation in the data combine instance-based and rule-based learners. This result suggests that the best psychologically plausible learning model of morphological variation combines instance-based and rule-based approaches and might vary from case to case.

Keywords: morphology; natural language processing; corpus studies; computational modelling

Background

Language users rely on their existing lexical representations when they pick the plural for a previously unseen noun or the past tense for a previously unseen verb (Berko, 1958). Algorithmic learning models, trained on existing natural language corpora, can predict participant preferences in a language task. The current consensus is that lexical representations play into all word formation processes (Lindsay-Smith, Baerman, Beniamine, Sims-Williams, & Round, 2024). The question is which models of word formation best reflect the role of the lexicon in language processing and production (Pierrehumbert, 2006).

There are two distinct views of the core mechanics of word formation. One is that word formation follows the principles of human categorization, comparing input forms to existing, detailed lexical representations. This is the main assumption behind approaches based on nearest-neighbor categorization, like the Tilburg Memory-Based Learner (Daelemans, Zavrel, Van Der Sloot, & Van den Bosch, 2004) or the linguistic implementation of the Generalized Context Model (Dawdy-Hesterberg & Pierrehumbert, 2014). The other one is that word formation relies on higher-level, abstract rules or generalizations. These generalizations are based on lexical representations, but the word formation process discards the representations themselves and uses only the generalizations. This is the central idea of the rule-based Minimal Generalization Learner, which has been highly influential in modeling variable word formation (Albright & Hayes, 2003).

Instance-based and rule-based learning models of word formation are not mutually exclusive. For instance, Rácz, Beckner, Hay, and Pierrehumbert (2020) put forward the idea that instance-based generalizations play a major role in short-term linguistic accommodation and rules gain importance in long-term lexical integration in English past tense formation. Tagliamonte and Baayen (2012) combine different learning models to analyse the distribution of was/were in York English.

In this paper, we look at three different, variable word formation patterns in Hungarian, a morphologically complex language. We implement two instance-based and one rule-based learner, train them on data drawn from a large language corpus and use them to predict participant responses in a Wug task. We have two research questions: (1) Do instance-based or rule-based learners best predict variation in the three word formation patterns? (2) Do instance-based and rule-based learners contribute together to explaining variation in the three patterns?

Stimuli

Rácz and Lukács (2023) collected corpus data and participant responses for three variable morphological patterns in Hungarian (Siptár & Törkenczy, 2000).

Variable patterns

Leveling. Hungarian verbs mark definiteness on the verb ([mond-o**k**] say-1SG.INDEF ‘I say something’ / [mond-o**m**] say-1SG.DEF ‘I say it’). Many Hungarian verbs vary between the two suffixes in an indefinite context: that is, *-k* and *-m* vary in the 1SG.INDEF paradigm slot (e.g. [jaːtsː-o**k**] / [jaːtsː-o**m**] play-.1SG.INDEF, ‘I play’). This variation is not phonetically motivated, it is socially salient, highly productive, and restricted to this paradigm slot.

Vowel deletion. If a Hungarian verb stem ends in two consonants and is followed by a suffix that begins with a consonant, the resulting C_1C_2-C_3 cluster might not be phonotactically well-formed. Some verbs always break it up by adding a linking vowel (C_1C_2-VC_3: [jaː**tsː**-ɒ] play-3SG.DEF ‘(s)he plays it’ / [jaː**tsː**-ɒ-nɒk] play-3PL.INDEF ‘they play’), some insert a vowel in the C_1C_2 cluster (C_1VC_2-C_3: [ʃø**pr**-i] sweep-3SG.DEF ‘(s)he sweeps it’ / [ʃø**p**ø**r**-nɛk] sweep-3PL.INDEF ‘they sweep’). For some verbs, stem-final

In L. K. Samuelson, S. L. Frank, M. Toneva, A. Mackey, & E. Hazeltine (Eds.), *Proceedings of the 46th Annual Conference of the Cognitive Science Society.*

vowel insertion/deletion is variable (C_1C_2-VC_3 / C_1VC_2-C_3: [aːrɒ**ml**-ɒ-nɒk] / [aːrɒ**mol**-nɒk] flow-3PL.INDEF 'they flow'. This variation is phonetically motivated, it is not socially salient, shows limited lexical scope, and occurs with most consonant-initial suffixes.

Vowel harmony. The suffix vowel typically matches the phonological characteristics (like front/backness) of the last stem vowel in Hungarian ([bɛrlin-b**ɛ**n] Berlin-INE 'in Berlin' / [london-b**ɒ**n] London-INE 'in London'). Nouns show variable vowel harmony on case markings (postpositions) if the noun matches the following template: AE, where A is a back vowel and E is a semi-neutral front vowel. These nouns can occur with both a back vowel or a front vowel suffix. (e.g. '[hotɛl-b**ɛ**n]/[hotɛl-b**ɒ**n]' hotel-INE, 'in the hotel').

All three patterns are widely explored in Hungarian morphophonology (Siptár & Törkenczy, 2000). In addition, Rácz and Lukács (2023) discuss the distributional, perceptual, and social differences between the two verb patterns, which provide for an interesting comparison set.

Training and test data

Rácz and Lukács extracted variable forms from the Hungarian Webcorpus, used these to build an ngram model and generated nonce forms for each variable pattern, discarding nonce forms that were too close to existing forms in the language. They then ran a visual forced-choice task in which participants had to pick variant A or variant B (e.g. '[lɒk-o**k**]/[lɒk-o**m**]') for a nonce word in a simple carrier sentence. They collected one response each from 25-35 participants for 162 nonce forms per pattern, 486 in total.

For our learning models, we (i) transcribed test and training forms in a simple phonetic alphabet to have a segment-to-character correspondence in the word forms and (ii) set up discrete categories[1]. For (i) *leveling*, variable verbs in the corpus above the median of the distribution of the log odds of *-k/-m* (the two variants) were assigned the 'high' category label, the others the 'low' label. For (ii) *vowel deletion*, stable CC-V verbs in the corpus were assigned the 'high' label, stable CVC- verbs the 'low' label. For (iii) *vowel harmony*, variable nouns of the template AE above the median of the distribution of the log odds of *back/front* suffix preference were assigned the 'high' label, others the 'low' label. The choice of the median in both cases is abritrary and should be explored further. At the same time, it reflects the preferences of participants in the Wug task.

Since the *vowel harmony* variation applies across a range of suffixes, we fit a generalized linear mixed model of the log odds to estimate random intercepts for stems and suffixes, and used the stem random intercepts instead of raw aggregates. (This was not applicable to the other two patterns: *leveling* is restricted to one paradigm slot and non-varying verb stems served as training forms for *vowel deletion*.) Training sets were further pruned to exclude rare and overlong forms and to balance categories where possible. Data come from a webcorpus (Nemeskey, 2020). Table 1 shows the number of training and test forms for each pattern.

Table 1: Number of training and test forms for each pattern

variation	training forms, cat 1	training forms, cat 2	test forms
leveling	303	302	162
vowel deletion	108	1199	162
vowel harmony	117	117	162

Learner implementation

We implemented three learning models in the R language (R Core Team, 2024).

K-Nearest Neighbors (KNN)

The KNN makes pairwise comparisons of test forms and training forms using a pre-specified word distance value d. We discuss d below. The KNN selects the k training forms most similar to the test form. In our implementation, it then calculates the category average for these forms and assigns this as a category score for the test form. (If $k = 3$ and the nearest neighbors are category 1, category 1, and category 2, the score will be $2/3 = .66$.) Our implementation is much more stripped down than other KNN-based models of linguistic categorisation, like TiMBL (Daelemans et al., 2004).

Generalized Context Model (GCM)

The GCM makes pairwise comparisons of test form and training forms using the formula $exp(-d/s)^p$ where d is the distance measure, s controls the trade-off between the number of comparisons and test-training similarity, whereas p controls whether we see an exponential decay (for $p = 1$) or Gaussian decay ($p = 2$) in the similarity distance. Instead of restricting the comparison set to k forms, it considers all training forms in both categories to calculate a similarity score for the test form (Dawdy-Hesterberg & Pierrehumbert, 2014). It then calculates an overall similarity score for a test form and a category, which is the total similarity to all training forms in the category divided by the total similarity to all training forms.

For the KNN and the GCM, we used three pairwise similarity measures (d): **Levenshtein** distance, **Jaccard** distance, and **phonological** distance. Levenshtein distance is the number of edits between the test form and the training form. Jaccard distance is the set size of the intersect of segments in the test and training form divided by the set size of the union of segments in the two forms. It ranges between 0-1. For identical pairs, the Jaccard distance will be 0 (no test forms are identical to training forms in our data). For pairs with zero segments in common it will be 1. Phonological distance is a version of Levenshtein distance that also takes segmental similarity into account. For example: [f] and [p] are both voiceless, labial obstruents, while [f] and [b] are both labial

[1] For modeling details, see `https://doi.org/10.5281/zenodo.11121262`

obstruents, but [f] is voiceless and [b] is voiced. The [f]-[p] pair share more natural classes (e.g. voiceless, labial, obstruent) than the [f]-[b] pair (e.g. labial, obstruent). Segmental similarity expresses the notion that a distance between two segments can be larger or smaller depending on the number of overlapping natural classes versus total natural classes. In turn, phonological distance is the sum of segmental similarities between two aligned forms, one test form and one training form. Two forms are *aligned* if pairwise comparisons between segments minimize total distance between the two words. Phonological distance has been used to capture the notion that language users rate word similarity using segmental similarity, so that [fa] is closer to [pa] than to [ba]. Our implementation follows Albright and Hayes (2002); Dawdy-Hesterberg and Pierrehumbert (2014); Rácz et al. (2020).

Minimal Generalization Learner

The Minimal Generalization Learner (MGL) looks for input-output correspondences or rules of the form $A \rightarrow B/C\_D$. It then generalizes these rules to as many input-output pairs with overlapping contexts and calculates each rule's reliability (how many of its potential inputs actually undergo the rule), its confidence (the confidence interval over how well the rule would apply to all possible forms – here, a rule that only has three potential inputs will have lower overall confidence than a rule that has thirty inputs, even if it applies to proportionally more of them). The MGL has a strong locality restriction: it ignores contextual overlap further from the location of change if there is mismatch closer in.

The MGL considers whether a smaller rule applies to a subset of a larger rule's contexts and accounts for the reliability of the larger rule. The confidence interval for rule reliability and the subsequent penalty for rules that apply to fewer forms is controlled by the parameter α_{lower}. The confidence interval for comparing the reliability of a superset rule and subset rule and the subsequent penalty on superset rules that have most of their work done by a subset rule is controlled by α^{upper} (Albright & Hayes, 2002, 2003).

Our implementation of the Minimal Generalization Learner has been reverse-engineered from Albright and Hayes (2002) and Mikheev (1997) and uses segment-to-segment correspondence (but no subsegmental detail) and rule impugnment. It checks for segmental overlaps in creating rule contexts but does not generalize across natural classes. Its training and test data have been set up to exclude irrelevant phonological variation (this is typically done in the MGL by a separate module)[2].

Model fitting

We performed a grid search for the three learners. The the parameter space for the KNN, the GCM, and the MGL are in Table 2.

Table 2: Parameter space for the KNN, GCM, and the MGL

model	parameter	value
KNN	k	1, 2, 3, 5, 7, 15
	d	Levenshtein, Jaccard, phonological
GCM	p	1, 2
	s	.1, .3, .5, .7, .9
	d	Levenshtein, Jaccard, phonological
MGL	α^{upper} α_{lower}	.25, .5, .75, .9

Each learner was fit on each variable pattern separately. For the MGL, we further split training data into the three suffixes present for the patterns in the training and test data: *vowel deletion* (1PL, 2PL, and 3PL) and *vowel harmony* (DAT, INE, and ADE).

Model evaluation

The test data come from a forced-choice task with variant A and B for each test prompt. The KNN and the GCM provide a score between 0-1 for each test word based on similarity to training words in category A versus category B for each of the three patterns. The MGL generates rules based on the training data. We followed Albright and Hayes (2003) and used the rules' adjusted confidence (after rule impungment) to find the best rule that generates output A and the best rule that generates output B for each test input and then divided the adjusted confidence of rule A with the summed adjusted confidence of rules A and B per word to get a category score that was comparable to the KNN and the GCM output.

Finding the best model parameters We fit each learner type (KNN, GCM, MGL) on each variation pattern (*leveling*, *vowel deletion*, and *vowel harmony*) exploring the model-specific parameter space (see Table 2). We took the learner's predictions for each word and fit a binomial generalized linear model predicting the proportion of variant A / variant B responses in the test data for each test word using the learner prediction or score for that test word as a predictor ($cbind(a,b) \sim 1 + score$). We then used the Z value of the term estimate for the predictor to find the best learner parameters. This gives us a better evaluation metric than binning test words in categories and calculating an F-score or accuracy.

Building the ensemble model We scaled learner predictions and combined them as predictors in a single generalized linear model for each variation pattern ($cbind(a,b) \sim 1+$ KNN score $+$ GCM score $+$ MGL score). We used a χ^2 test of likelihood ratio to compare models in order to determine whether each learner contributed to explaining variation in the test data.

[2] For details, see the SI. Wilson and Li (2021) offer a CLI implementation of the original MGL. There exists another R-based implementation, by João Verissimo, but it is not currently available.

Results

Best individual learners

All three learners account for variation in the test data across all three variable patterns.

KNN The best KNN learners can be seen in Table 3. All three models use a relatively large number of nearest neighbors (k) as well as different distance metrics: Among the verbal inflection patterns, *leveling* uses simple Levenshtein distance, while *vowel deletion* uses the more detailed phonological distance measure. The noun pattern, *vowel harmony*, uses Jaccard distance.

Table 3: Best KNN learner parameters for each pattern

variation	k	d	est	ste	z
leveling	7	lev	1.23	0.13	9.20
V deletion	15	phon	0.61	0.14	4.29
V harmony	15	jaccard	2.81	0.19	14.91

GCM The best GCM learners can be seen in Table 4. The *leveling* pattern favors phonological distance and a high s, meaning that the model weighs larger, less similar gangs of training forms over smaller, more similar ones. The *vowel deletion* pattern favors Levenshtein distance and a small s, promoting smaller gangs of more similar training forms. Both models use Gaussian decay ($p = 2$). The *vowel harmony* pattern still uses Jaccard distance. Since this measure works differently from the other two, s and p are not interpretable when using Jaccard distance[3]. Note that Jaccard distance is bound between 0-1, resulting in a larger apparent estimate in Table 4 – the z value is more helpful to compare models.

Table 4: Best GCM learner parameters for each pattern

variation	p	s	d	est	ste	z
leveling	2	0.9	phon	27.97	2.40	11.66
V deletion	2	0.3	lev	36.23	6.23	5.81
V harmony	NA		jaccard	499.57	26.32	18.98

Minimal Generalization Learner The best MGL learners can be seen in Table 5. The rule-based approach is particularly successful for the *leveling* pattern.

Ensemble model

We combine the standardized scores of the best KNN, GCM, and MGL learners in a single generalized linear model for each pattern and then use a likelihood ratio test to see whether leaving them out results in worse model fit. The results can be seen in Table 6.

Table 5: Best MGL parameters for each pattern

variation	α^{upper}	α_{lower}	est	ste	z
leveling	0.25	0.25	4.18	0.27	15.31
V deletion	0.25	0.90	6.51	1.32	4.92
V harmony	0.25	0.25	2.49	0.47	5.28

Table 6: Ensemble model tests across the three patterns

variation	name	χ^2	p
vowel deletion	KNN	0.28	0.60
	GCM	10.39	0.00
	MGL	6.77	0.01
vowel harmony	KNN	0.20	0.65
	GCM	149.49	0.00
	MGL	0.32	0.57
leveling	KNN	1.41	0.24
	GCM	23.56	0.00
	MGL	123.60	0.00

The KNN does not contribute to the ensemble model for any of the patterns. The GCM and the MGL together contribute to explaining the two verb patterns. The GCM does all the work for the noun pattern, *vowel harmony*. We take each ensemble model and calculate the 95% Wald confidence intervals for the term estimate for each learner score (each model predictor) as well as McFadden's pseudo-R (1 - deviance / null deviance) for each ensemble model. The results are in Table 7. The confidence intervals tell the same story as the likelihood ratio tests: the KNN does not contribute to any model, and the GCM is the only relevant learner for *vowel harmony*. The best model was fit on *leveling*, the worst one on *vowel deletion*. The *vowel harmony* model, even though two of its three predictors are not helpful, is remarkably close to the best model in accuracy. All three models show low collinearity (with the highest variation inflation factors at 95%CI [2.02, 3.24], [2.20, 3.57], and [1.47, 2.25]).

Table 7: Ensemble model fits for each variation with 95% CI for each term and total McFadden's R for the ensemble model

variation	KNN	GCM	MGL	R
leveling	[-.57, .14]	[.58, 1.36]	[.99, 1.42]	0.44
v deletion	[-.52, .30]	[.27, 1.12]	[.09, .64]	0.15
v harmony	[-.64, .4]	[2.45, 3.42]	[-.38, .21]	0.35

Figure 1 shows how test responses correlate with learner weights. It plots the KNN, GCM, and MGL weights (columns) against test responses in each pattern (rows). The vertical axis is the log odds of response A / B for each test

[3] For more details, see our online supplement

form, the horizontal axis is the test form weight from the best model. The fourth column shows the predictions of the ensemble model built from the best KNN, GCM, and MGL models for each pattern, with the model predictions on the horizontal axis.

Broadly speaking, the KNN learners look very similar to the GCM learners, except that they have noisier predictions. This is why they contribute nothing in the ensemble models. For the two verb patterns, *leveling* and *vowel deletion*, the MGL's rules visibly underfit the data. Combining these with the GCM predictions, in turn, helps reducing the noise in the latter, giving a better combined fit. For the noun pattern, *vowel deletion*, we see that the GCM, using Jaccard distance, is very good at separating the similarity space into two distributions, and this is remarkably close to what the participants are doing. Adding rules here would make the fit worse – this is why rules do not contribute to the ensemble model for *vowel harmony*.

Discussion

We tested two instance-based and one rule-based learner on three variable word formation patterns in Hungarian. We trained the models on corpus data and tested them on a forced-choice word formation task using nonce words. We found that all three models explain variation in all three patterns. For our two verb patterns, both the instance-based GCM and the rule-based MGL independently contributed to explaining variation, echoing results by Albright and Hayes (2003) and Rácz et al. (2020). For the noun pattern, only the GCM was relevant.

The GCM and the MGL have been very successful in encapsulating low-level and high-level linguistic generalizations, respectively, across different types of language data, and have seen widespread deployment in the trenches of morphological variation (Lindsay-Smith et al., 2024). We included the KNN to test an assumption that is often implicit in nonce word stimulus generation, viz. that respondents might completely model their responses to a nonce word solely based on its nearest lexical neighbor. If that were the case, the complex lexical relationships captured by the GCM or the MGL would in fact reflect not on the test word but its nearest neighbor. This has proven false, at least for this set of nonce words which were created to exclude test words too close to real lexical neighbors (Rácz & Lukács, 2023). The best KNNs used many neighbors (meaning that, even if participants did model responses on one or two nearest neighbors, they all used different ones) and were clearly outmatched by the GCM and the MGL in the ensemble model.

In order to better understand how the GCM and the MGL differ from each other in performance, we need to talk more about the Hungarian morphological patterns. Hungarian verbs typically end in one of several derivational suffixes and so verbs as a whole have relatively restricted templatic morphology (Siptár & Törkenczy, 2000): see e.g. [gugli-zik] google-V-3SG.INDEF '(s)he Googles', [feːsbuk-ol], Facebook-V-3SG.INDEF 's(he) uses Facebook'. Different derivational suffixes prefer different morphological variants in both *leveling* and *vowel deletion*. A rule-based learner excels at identifying trends that are tied to word templates, giving it a comparative edge over a learner that uses word-based analogy. At the same time, broad, word-to-word similarity also guides participant responses, and this is best captured by a low-level analogical model. In Hungarian, at least, participants seem to use a relatively coarse distance measure to compare target words with their lexical representations, meaning that learners using phonological distance do not seem to enjoy a clear advantage over edit distance. *Vowel deletion*, which is more limited in scope, applies across a range of paradigm slots, and is lexically specified for at least certain stems, poses a harder learning problem than *leveling*, which is more consistent, more productive, and a social marker to boot. At the same time, a rule-based learner which uses phonological generalizations to find rule contexts could outperform our learner, based on segmental overlap, on both verb patterns.

Learner success looks very different in the noun set. This is, to an extent, idiosyncratic to this pattern. The test forms in *vowel harmony* either had [ɛ] or [eː] as their second vowel, and the latter is much more transparent for vowel harmony (Siptár & Törkenczy, 2000). A rule-based learner that operates on a separate vowel tier would have easily captured this. As is, the shapes of variation were best captured by instance-based analogy. At least, and especially for [ɛ]-forms, overall similarity to real words also plays into participant responses to some degree (see Figure 1).

Taken together, the results suggest that rules and analogy explain different aspects of morphological variation. The best model needs to reflect variable template effects, best captured by a rule-based learner, as well as effects of overall similarity. That is, rule-based and instance-based (or other low-level, see Milin, Divjak, Dimitrijević, and Baayen (2016)) learners might be the best cognitively plausible accounts of not only different linguistic processes, but also different aspects of the same phenomena.

Acknowledgments

We would like to thank Colin Wilson, Adam Albright, Bruce Hayes, and Janet Pierrehumbert. This research was supported by grants OTKA-139271, OTKA-138188, and TKP2021-EGA-02 by the Hungarian National Research, Development and Innovation Fund.

References

Albright, A., & Hayes, B. (2002). Modeling English past tense intuitions with minimal generalization. In *Proceedings of the ACL-02 workshop on morphological and phonological learning* (pp. 58–69).

Albright, A., & Hayes, B. (2003). Rules vs. analogy in English past tenses: A computational/experimental study. *Cognition*, *90*(2), 119–161.

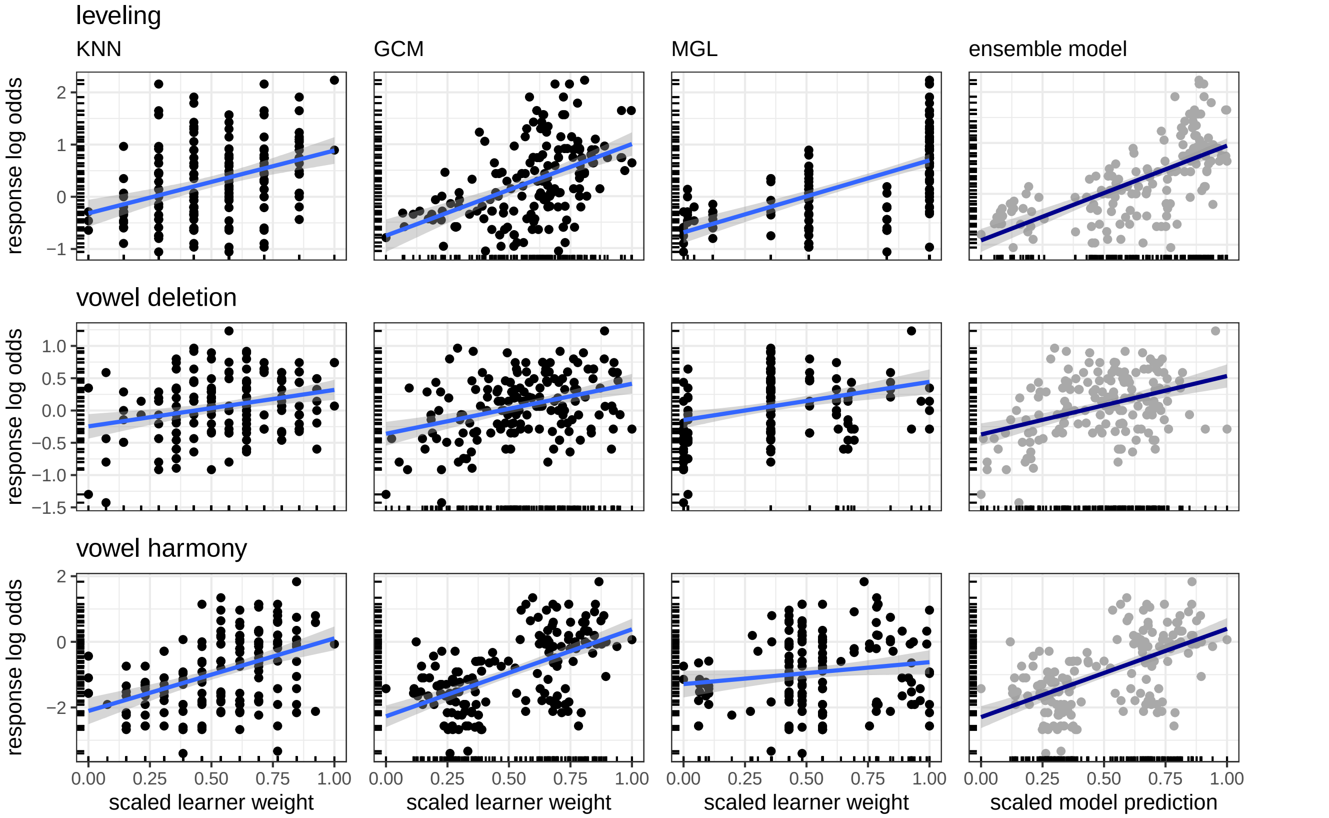

Figure 1: Learner predictions. From top to bottom: the three variation patterns. From left to right: the KNN, the GCM, and the MGL learner weights, as well as predictions from the ensemble model ‘cbind(resp A, resp B) ∼ KNN weight + GCM weight + MGL weight’

Berko, J. (1958). The child’s learning of English morphology. *Word*, *14*(2-3), 150–177.

Daelemans, W., Zavrel, J., Van Der Sloot, K., & Van den Bosch, A. (2004). Timbl: Tilburg memory-based learner. *Tilburg University*.

Dawdy-Hesterberg, L. G., & Pierrehumbert, J. B. (2014). Learnability and generalisation of Arabic broken plural nouns. *Language, cognition and neuroscience*, *29*(10), 1268–1282.

Lindsay-Smith, E., Baerman, M., Beniamine, S., Sims-Williams, H., & Round, E. R. (2024). Analogy in inflection. *Annual review of linguistics*, *10*.

Mikheev, A. (1997). Automatic rule induction for unknown-word guessing. *Computational Linguistics*, *23*(3), 405–423.

Milin, P., Divjak, D., Dimitrijević, S., & Baayen, R. H. (2016). Towards cognitively plausible data science in language research. *Cognitive Linguistics*, *27*(4), 507–526.

Nemeskey, D. M. (2020). Natural language processing methods for language modeling.

Pierrehumbert, J. B. (2006). The next toolkit. *Journal of phonetics*, *34*(4), 516–530.

Rácz, P., Beckner, C., Hay, J. B., & Pierrehumbert, J. B. (2020). Morphological convergence as on-line lexical analogy. *Language*, *96*(4), 735–770.

Rácz, P., & Lukács, Á. (2023). Morphological convergence and sociolinguistic salience: an experimental study. *PsyArXiv preprint PsYArXiv:2023.31234*. doi: https://doi.org/10.31234/osf.io/zqwxv

Siptár, P., & Törkenczy, M. (2000). *The phonology of Hungarian*. OUP Oxford.

Tagliamonte, S. A., & Baayen, R. H. (2012). Models, forests, and trees of York English: Was/were variation as a case study for statistical practice. *Language variation and change*, *24*(2), 135–178.

Wilson, C., & Li, J. S. (2021). Were we there already? applying minimal generalization to the sigmorphon-unimorph shared task on cognitively plausible morphological inflection. In *Proceedings of the 18th SIGMORPHON workshop on computational research in phonetics, phonology, and morphology* (pp. 283–291).