Class 13: Model comparison

1 Back to overfitting vs. underfitting

- As we've discussed, we want to strike a balance
  - Overfitting: model closely fits observed data, but is likely to make wrong predictions about the next item to come along
  - Underfitting: model fails to capture important aspects of observed data, and therefore is also likely to make wrong predictions about new data
- But how do we find the sweet spot in between?
  - How do we know which aspects of the observed data are important?
  - How precisely should we fit those aspects?

2 Roadmap

- Whether a factor/constraint can justify its presence in a model
  - Wald test (and arguments against them)
  - Likelihood ratio test
- Whole-model comparisons: AIC/BIC
- Machine-learning approaches
  - Empirical evaluation of over/under-fit through cross-validation
- Stephanie Shih presents: tutorial on random forests and related issues

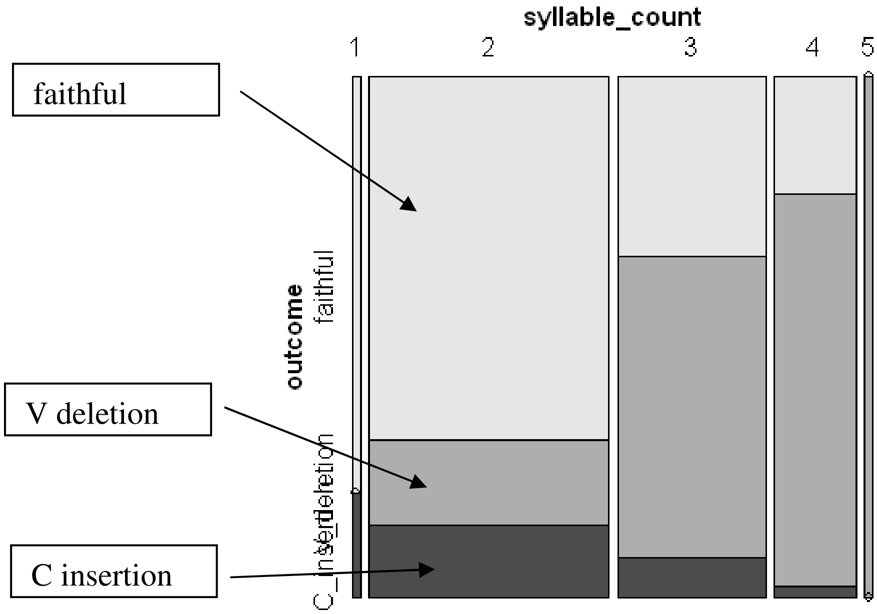
3 Today's data set: French adjectives in *–esque* [ɛsk]

- To learn more: Plénat 1997, Plénat et al. 2002
- Highly productive suffix (similar meaning as in English); can even attach to phrases:
  - ben-et-jerry-esque 'Ben and Jerry[ice cream brand]-esque'
  - Eric-et-Ramzy-esque 'Eric and Ramzy[comedy duo]-esque'
  - bonnes-resolutionesque 'good-resolutions-esque'
  - little-green-footballsesque 'Little Green Footballs [blog]-esque'
- Creates a hiatus problem with V-final stems. 3 solutions
  - faithful: zola-esque 'Zola-esque'
  - delete V: zol-esque
  - insert C: zolat-esque
- As Plénat points out, the choice is sensitive to...
  - stem length: the shorter the stem, the worse deletion is
  - stem-final V quality: higher vowels are less likely to delete—perhaps the hiatus they create isn't as bad
- Some additional phenomena we'll ignore:
  - Final C or VC can also delete, esp. if the C is a sibilant or a velar (OCP): cervant-esque
  - Occasionally the suffix seems to be *–iesque* instead
  - There's also an option *–este* [ɛst] sometimes used if stem contains velar: blog-este

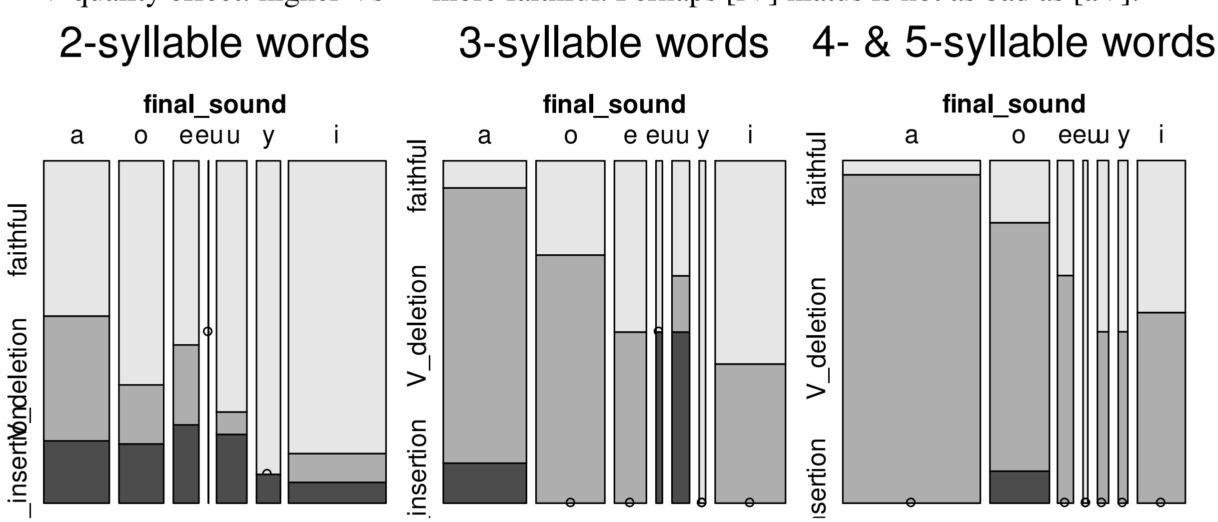
- Data sources:
  - frWAC (1.6 billion word web corpus), Jožef Stefan Institute interface [1]
  - supplemented with items from Wiktionnaire,[2] TLFi[3]
  - 2800 potential word types ending in *esque* or *este*
  - Italian/Spanish loans omitted (*grotesque, churrigueresque*)
  - 294 clear cases of vowel-final stems (and no latent/liaison consonant available)

4 Exploring the data

- Syllable-count effect: longer words have more V-deletion, at expense of other 2 options



- V-quality effect: higher Vs → more faithful. Perhaps [iV] hiatus is not as bad as [aV].



[1] http://nl.ijs.si/noske/wacs.cgi/first_form
[2] http://fr.wiktionary.org/wiki/-esque
[3] http://atilf.atilf.fr/

5 Do we want an interaction between syllable count and vowel quality?

- An interaction term implies that vowel quality works differently within each syllable-count group (and vice versa).
- To start, let's have a binary model (deletion or non-deletion)—later we'll consider the ternary model.
  - I'm treating syllable-count as an integer
  - We ask R to find the best values of *a*, *b*, *c*, *d*, *e*, *f*, *g*:

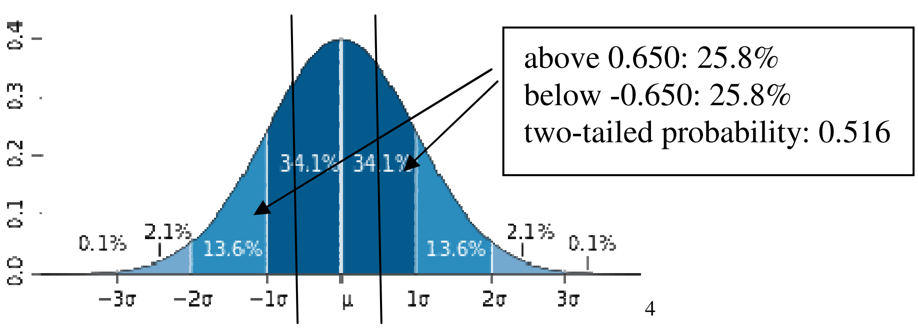
Where *p* is probability of deletion,

```
ln(p/(1-p)) =
      a*(finalV=lo) + b*(finalV=mid) + c*(finalV=hi) + d*syll_count
            + e*(syllcount*finalV=mid) + f*(syllcount*finalV=hi)

   glm(formula = delete_or_not ~ final_V_height * syllable_count,
       family = binomial(logit), data = esque)

                                     Estimate  Std. Error  z value  Pr(>|z|)
   (Intercept)                        -4.5022      1.1246   -4.004  6.24e-05 ***
   final_V_height=mid                  0.9718      1.4951    0.650     0.516
   final_V_height=hi                  -0.8437      1.4967   -0.564     0.573
   syllable_count                      1.9631      0.4457    4.404  1.06e-05 ***
   final_V_height=mid:syllable_count  -0.7678      0.5637   -1.362     0.173
   final_V_height=hi:syllable_count   -0.5300      0.5583   -0.949     0.342
```

R's best guesses for a-f *estimate/standard_error*

6 Wald test

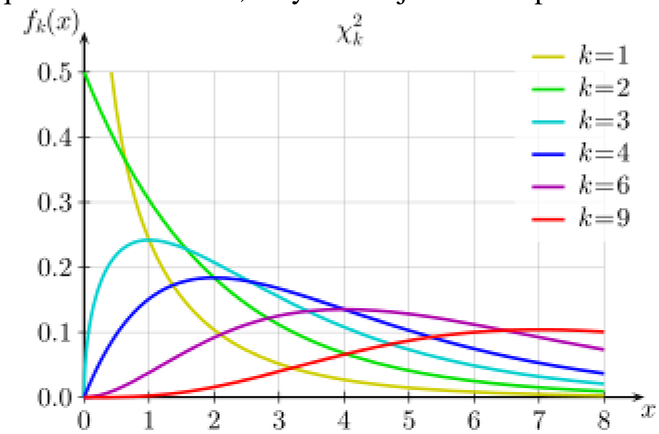
- The rightmost column in the results above asks, for each *z*-value...
  - in a random set of data (e.g., no difference between mid and low Vs), how often would we expect to see a *z*-value (e.g., 0.650) that far from zero or further?
  - E.g., if *p*=0.516, we expect that substantial a *z* value to occur by chance about half the time
- How it does it in this case: a Z-test
  - Make a big assumption: coefficient estimates will be approximately normally distributed
  - So how far out on the tail of a normal distribution is the estimate?



[4]

4 http://commons.wikimedia.org/wiki/File:Standard_deviation_diagram.svg

- Something else you'll often see: a chi-square test
  - Again, assume coefficient estimates are approximately normally distributed
  - Then the estimate squared and divided by its variance (*t*-value) would approximately follow a chi-squared distribution, so you can just look up the value there:



[5]

7 Something more reliable: likelihood ratio test

- Let's compare the model above to the same thing but without the interaction.
- Of course, it will fit better with the interaction
  - log likelihood of full model: –136.2825 (R command: `logLik(myModel)`)
    - That is, the model gives the observed data a probability of $6.5 * 10^{-60}$
  - log likelihood of model with no interaction: –137.2806
    - Model gives observed data a probability of $2.4 * 10^{-60}$

- But is it worth it? Does the interaction improve the model fit *enough*?
- Likelihood ratio—or rather, diff. between log likelihoods: –136.2825 – (–137.2806) = 0.9981
  - Multiply by 2: 1.9962
  - Magically, this number has a chi-squared distribution, with *k* ("degrees of freedom") equal to the number of predictors removed (or, more technically, constrained to be zero)
    - In our case, *k*=2, since we removed the interaction's two subparts
    - As you can see by inspecting the chi-squared distribution above, this will yield an unimpressive *p*-value of about 0.3.
    - We can get R to do all of this for us

Name of model without interaction

What I named the model with the interaction

```
> anova(esque.binary12, esque.binary1times2, test="Chisq")
Analysis of Deviance Table

Model 1: delete_or_not ~ final_V_height + syllable_count
Model 2: delete_or_not ~ final_V_height * syllable_count

  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1       290     274.56
2       288     272.56  2   1.9962   0.3686
```

[5] http://commons.wikimedia.org/wiki/File:Chi-square_pdf.svg

8 Let's take a quick look at the model without the interaction

```
glm(formula = delete_or_not ~ final_V_height + syllable_count,
    family = binomial(logit), data = esque)

                    Estimate  Std. Error  z value  Pr(>|z|)
(Intercept)          -3.3405      0.5901   -5.661  1.51e-08 ***
final_V_height=mid  -1.0195      0.3646   -2.796   0.00517 **
final_V_height=hi   -2.1808      0.3812   -5.721  1.06e-08 ***
syllable_count       1.4950      0.2094    7.138  9.48e-13 ***
```

- Wald tests are very promising
- R can do the likelihood ratio test for each submodel that's missing one constraint:

```
> library(car)
> Anova(esque.binary12, type=2)

               LR Chisq Df Pr(>Chisq)
final_V_height   37.261  2  8.106e-09 ***
syllable_count   68.214  1  < 2.2e-16 ***
```

  - So it looks like we do want both of these predictors.

9 Getting more serious: our dependent variable should really be ternary, not binary

- Let's also include the penultimate sound's type (V, C, glide), since if V-deletion just exposes another V, that doesn't solve hiatus.
- Resulting model is too wide to include on the handout, but let's do likelihood ratio tests on each factor (including interactions):

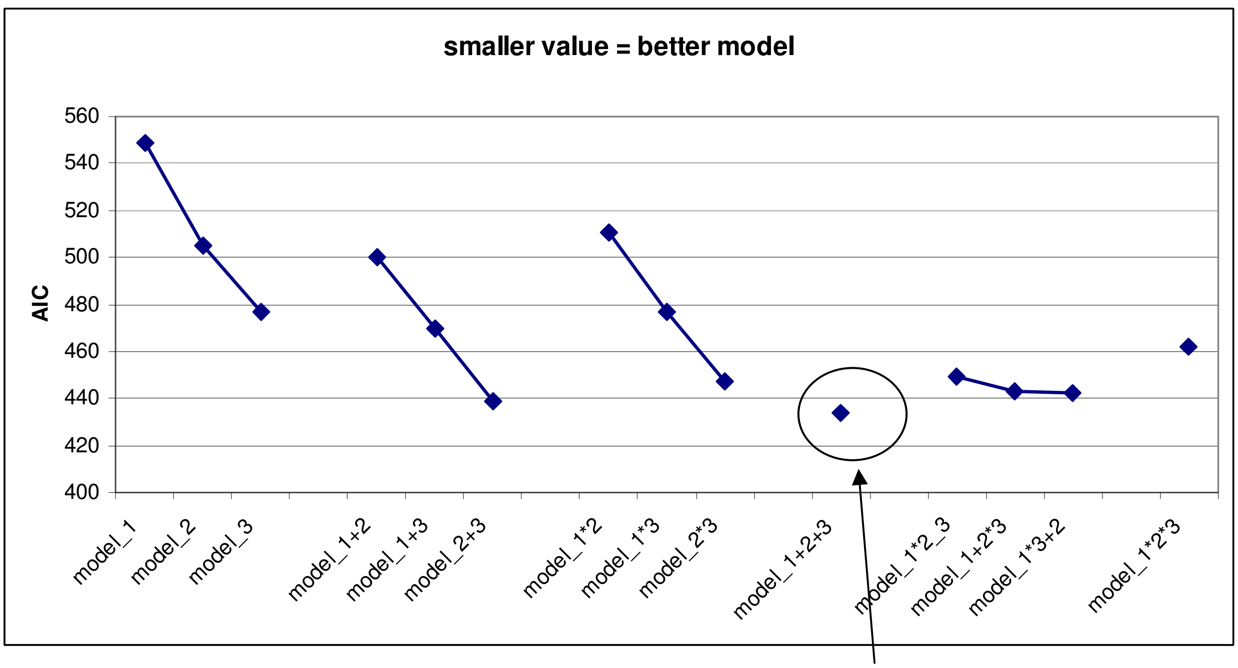
```
> Anova(esque.multinom1times2times3_prime, type=2)
Analysis of Deviance Table (Type II tests)

Response: outcome
                                                  LR Chisq Df Pr(>Chisq)
penultimate_coarse                                  13.631  4   0.008572 **
final_V_height                                      40.866  4  2.865e-08 ***
syllable_count                                      65.421  2  6.224e-15 ***
penultimate_coarse:final_V_height                    7.084  8   0.527649
penultimate_coarse:syllable_count                    5.184  4   0.268964
final_V_height:syllable_count                        5.233  4   0.264187
penultimate_coarse:final_V_height:syllable_count     6.359  8   0.607106
```

- The interactions don't seem to do much good.

10 An overall measure of model goodness: AIC/BIC

- AIC (Akaike Information Criterion): $2k - 2\ln(L)$
  - Where k is number of parameters (e.g., coefficients), L is likelihood
  - Smaller is better
  - Penalty for having more parameters, bonus for fitting data better
  - Does this remind you of anything??
- BIC (Bayesian Information Criterion): $-2\ln(L) + k\ln(n)$
  - Where n is number of data points
  - Again, smaller is better
  - Penalty for having more parameters grows faster if you have more data.
- In R, you should find both of these at the bottom of your model summary— `summary(myModel)`

- AIC results for models with different combinations of our 3 predictors and interactions between them
  - Slightly different because in doing this I kept each V separate instead of grouping the heights together



Best model has all 3 factors (penultimate sound type, final sound, syllable count), but no interactions.

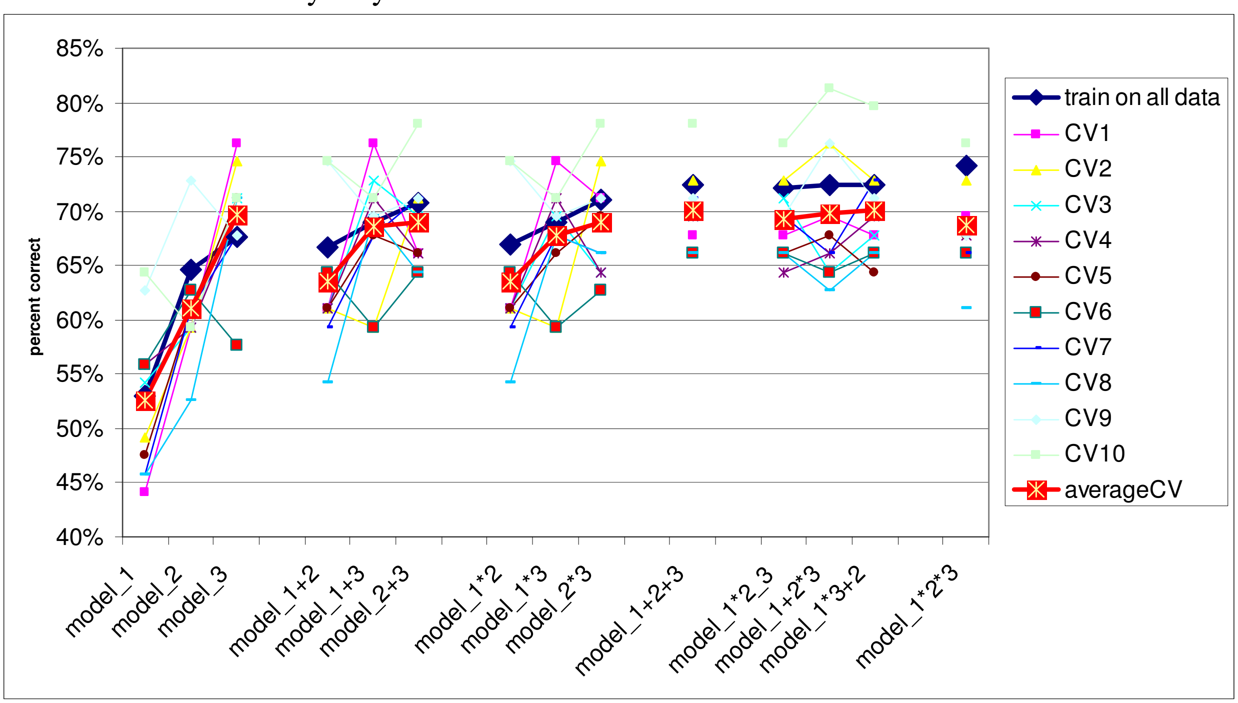
11 Cross-validation

- In the machine learning field, researchers are generally concerned less with finding "the truth" (how much more do spammers use all caps as compared to real e-mailers?) and more concerned with building a system that works well.
  - The cost of under- or over-fitting is practical: the system will do a poor job of classifying *new* messages as spam or not.

- Their solution: if you want to know how your model does on new data, test it on new data!

- Or, simulate this by holding some of your data back for **cross-validation**
  - Designate a randomly-selected 20% of your data as the cross-validation set
  - Train your model on the remaining 80%
  - Then test it on the held-out 20%
  - Probably repeat this a bunch of times
  - The model that does the best on the cross-validation data can be said to be the best (not under-, not over-) fitting model.

- I did this for the 14 models above.
  - The large diamonds represent fit when training and testing on all data
    - I used a crude measure of model fit: % of items assigned to correct outcome (faithful, C-insertion, V-deletion)
    - More-sophisticated measures would ask how far off the model was
      - Assigning 90% probability to the wrong choice is worse than 70%.
      - Assigning 90% probability to the right choice is better than 70%.
  - 10 cross-validation runs—average % correct is the large squares with Xs in between
  - Finer lines represent the 10 individual cross-validation runs, to give you an idea of how much they vary



High when trained on all data, low in cross-validation = overfitting.

Low on both = underfitting

High in cross-validation = just right

12 One more demo: *-esque* in MaxEnt

- Constraints
  - *VV violated by *zola-esque*, etc.
  - *[lo]V violated by z*olaes-que*, but also by *bilba-esque*, from *Bilbao*
  - *[mid]V violated by *cyrano-esque*
  - *[hi]V violated by *paganini-esque*, but also by *sanantoni-esque*, from *San Antonio*
  - DEP-C
  - MAX-V
  - MAX-V/1ST SYLL violated by *sp-esque*, from *spa*
  - MAX-V/1ST-2ND SYLL violated by *sp-esque*, *Monr-esque*, from *Monroe*
  - MAX-V/1ST -3RDSYLL violated by *sp-esque*, *monr-esque*, *figar-esque* from *Figaro*
  - MAX-V/1ST- 4THSYLL viol. by *sp-esque*, *monr-esque*, *figar-esque*, *miyazak-esque* from *Miyazaki*

- 3 candidates per input: *zola-esque, zol-esque, zolat-esque*

- Probability of each candidate is 1 or 0

- Results with huge sigma—weights are free to get as big as they want

*VV (mu=0.0, sigma^2=100000.0)	9.36
*[lo]V (mu=0.0, sigma^2=100000.0)	2.10
*[mid]V (mu=0.0, sigma^2=100000.0)	1.11
*[hi]V (mu=0.0, sigma^2=100000.0)	0.00
DEP-C (mu=0.0, sigma^2=100000.0)	11.95
MAX-V (mu=0.0, sigma^2=100000.0)	0.00
MAX-V/1ST SYLL (mu=0.0, sigma^2=100000.0)	8.42
MAX-V/1ST-2ND SYLL (mu=0.0, sigma^2=100000.0)	2.03
MAX-V/1ST -3RDSYLL (mu=0.0, sigma^2=100000.0)	0.50
MAX-V/1ST- 4THSYLL (mu=0.0, sigma^2=100000.0)	9.17

  - 72% correct (winning candidate more probable than either of the other two)

13 Cross-validation on MaxEnt model

- 20% of data is held out

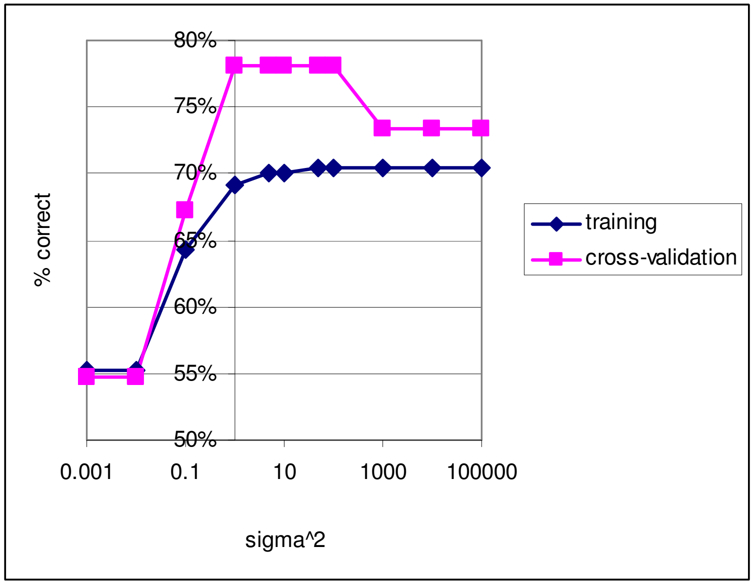
- MaxEnt Grammar tool trains on remaining 80%, tests on held-out 20%
  - I didn't have the programming time to set this up to repeat—so be warned that this isn't very reliable.
  - I seem to have chosen a strange slice, where the CV data are "easier" than the training data!

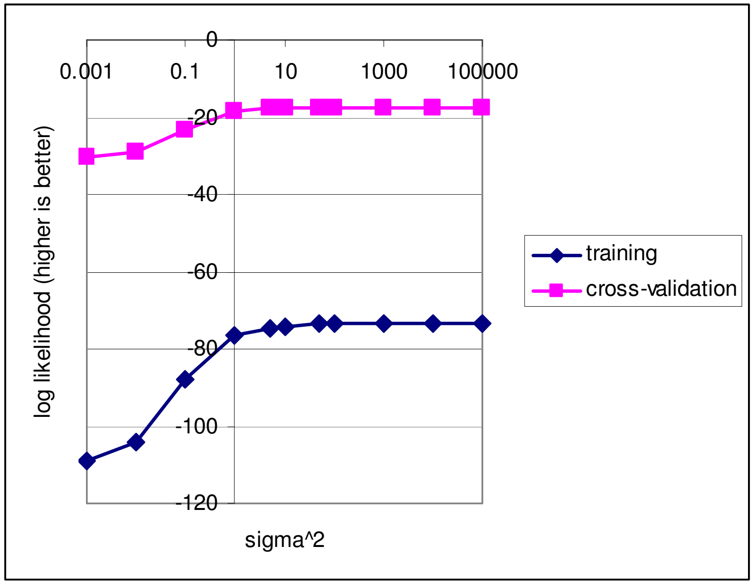
- Model comparison: what's the best sigma^2 (mu always 0)?

  (see over)

- Bigger σ^2 : better fit to training data
- Medium σ^2 : better fit to cross-validation data

	σ^2=100,000	σ^2=10,000	σ^2=1,000	σ^2=100	σ^2=50	σ^2=10	σ^2=5	σ^2=1	σ^2=0.1	σ^2=0.01	σ^2=0.001
*VV	8.88	6.78	4.77	2.86	2.33	1.25	0.88	0.24	0.00	0.00	0.00
*[lo]V	2.57	2.57	2.56	2.50	2.43	2.07	1.79	1.38	0.43	0.04	0.00
*[mid]V	1.53	1.53	1.52	1.46	1.40	1.08	0.84	0.57	0.05	0.00	0.00
*[hi]V	0.63	0.63	0.63	0.58	0.53	0.27	0.07	0.00	0.00	0.00	0.00
DEP-C	11.89	9.79	7.78	5.81	5.21	3.79	3.17	2.18	1.01	0.22	0.03
MAX-V	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
MAX-V/1ST	7.57	5.59	3.71	2.06	1.64	0.88	0.63	0.25	0.05	0.01	0.00
MAX-V/1ST-2ND	1.93	1.93	1.93	1.92	1.91	1.85	1.80	1.52	0.63	0.10	0.01
MAX-V/1ST -3RD	0.62	0.62	0.63	0.64	0.65	0.69	0.70	0.51	0.18	0.04	0.00
MAX-V/1ST- 4TH	9.17	7.07	5.05	3.09	2.50	1.13	0.57	0.00	0.00	0.00	0.00
% correct on trained data	70.4%	70.4%	70.4%	70.4%	70.4%	70.0%	70.0%	69.1%	64.3%	55.2%	55.2%
log likelihood of trained data	-73.3	-73.3	-73.3	-73.4	-73.5	-74.1	-74.6	-76.4	-87.8	-104.1	-109.0
% correct on CV data	73.4%	73.4%	73.4%	78.1%	78.1%	78.1%	78.1%	78.1%	67.2%	54.7%	54.7%
log likelihood of CV data	-17.5	-17.5	-17.5	-17.5	-17.5	-17.4	-17.4	-18.3	-23.3	-28.8	-30.3





14 I'll turn it over to Stephanie for random forests

References

Ahn, Suzy. 2011. Master's thesis. Seoul National University master's thesis.

Coetzee, Andries & Joe Pater. 2005. *Gradient phonotactics in Muna and Optimality Theory*. University of Michigan and University of Massachusetts.

Daland, Robert, Bruce Hayes, James White, Marc Garellek, Andrea Davis & Ingrid Norrmann. 2011. Explaining sonority projection effects. *Phonology* 28(02). 197–234. doi:10.1017/S0952675711000145.

Fleischhacker, Heidi. 2006. Similarity in phonology: evidence from reduplication and loan adaptation. UCLA Ph.D. dissertation.

Frisch, Stefan A, Janet B Pierrehumbert & Michael B Broe. 2004. Similarity Avoidance and the OCP. *Natural Language & Linguistic Theory* 22(1). 179–228.

Hayes, Bruce. 2009. Faithfulness and componentiality in metrics. In Sharon Inkelas & Kristin Hanson (eds.), *The nature of the word*, 113–148. Cambridge, MA: MIT Press.

Kawahara, Shigeto. 2007. Half rhymes in Japanese rap lyrics and knowledge of similarity. *Journal of East Asian Linguistics* 16(2). 113–144. doi:10.1007/s10831-007-9009-1 (14 February, 2012).

Kawahara, Shigeto. 2010. *Papers on Japanese imperfect puns*.

Martin, Andrew. 2007. The evolving lexicon. University of California, Los Angeles Ph.D. Dissertation.

Orgun, Cemil Orhan & Ronald L Sprouse. 1999. From "MParse" to "Control": Deriving Ungrammaticality. *Phonology* 16(2). 191–224.

Plénat, Marc. 1997. Analyse morpho-phonologique d'un corpus d'adjectifs dérivés en -esque. *Journal of French Language Studies* 7. 163–179.

Plénat, Marc, Stéphanie Lignon, Nicole Serna & Ludovic Tanguy. 2002. La conjecture de Pichon. *Corpus et recherches linguistiques* 1. 105–150.

Raffelsiefen, Renate. 1996. Gaps in word formation. In Ursula Kleinhenz (ed.), *Interfaces in phonology*, 194–209. Berlin: Akademie Verlag.

Raffelsiefen, Renate. 1999. Phonological constraints on English word formation. In Geert E Booij & Jaap van Marle (eds.), *Yearbook of Morphology 1998*, 225–287. (Yearbook of Morphology 8). Springer.

Raffelsiefen, Renate. 2004. Absolute ill-formedness and other morphophonological effects. *Phonology* 21(1). 91–142.

Shih, Stephanie. 2012. Linguistic determinants in English personal name choice. Presentation. Paper presented at the LSA annual meeting, Portland, OR.

Steriade, Donca. 2003. Knowledge of perceptual similarity and its uses: evidence from half-rhymes. In M.J. Solé, D Recasens & J Romero (eds.), *Proceedings of the 15th International Congress of Phonetic Sciences*, 363–366. Barcelona: Futurgraphic.

Zwicky, Arnold M & Elizabeth D Zwicky. 1986. Imperfect puns, markedness, and phonological similarity: with fronds like these, who needs anemones? *Folia Linguistica* 20(3-4). 493–544.