Finishing up with finding generalizations; course summary

(1) Review of the past week: Excel as a tool for finding generalizations and testing hypotheses

- Great tool for collecting, organizing, and counting data
- Even moderately capable of calculating properties of words (sonority of the final segment, weight of the penultimate syllable, etc.), using functions
- More complex tasks such as converting orthography to phonemic transcription, syllabifying words to determine if VCCV is VC.CV or V.CCV, are not so easy with functions; better to write a program to take care of these things
  - Excel does have built-in macro capability, using the VisualBasic programming language; could write programs to do these things in the macro module
  - Or, could write separate Perl scripts, etc.
  - I highly encourage you to take a programming class or work through an intro textbook in Perl/Java/C++/whatever you prefer, esp. if you intend to go on in linguistics, or are thinking about a career in linguistics-related industry
- But, of course, Excel can't tell you what it all means!
  - There are millions of patterns and significant differences in the world, but not all of them are equally interesting
  - When you find an intriguing pattern, there is still a lot of analysis left to do
  - Impossible to make sense of observations without careful thought, statistics, corroborating experimental results, and most of all, a theory

(2) Some things to keep in mind as you do your final assignment

- You are free to pick whatever hypotheses you want to test, but it's always a good idea to have some reason for believing that your hypothesis stands a good chance of being right
  - If you're trying to explain property X, some a priori reason for expecting that property Y may have an effect on X.
  - Since your assignment is to test at least 2 different properties, you might also find it instructive to test one hypothesis that strikes you as quite likely and one that strikes you as relatively unlikely
  - Your choice of properties will also be restricted a bit by the types of things that can be calculated in Excel (talk to me if you want to test a particular hypothesis, but can't figure out how to calculate the relevant property with Excel)

(3) Another approach to finding generalizations: using the rule-based model from Week 8

- There is a file named SpanishConjugations.in in the directory mingenlearner; task is to predict the infinitive ending (-ar, -er, -ir) based on the root
  - E.g., habl → hablar, habler, or hablir
  - Rules are of the form ∅ → -ar / X
  - If certain phonological environments (such as after [s], after velars, after stop + [r], etc.) are often associated with a particular class, then there will be very reliable rules deriving that class in that environment
  - Strategy: let the model learn rules to attach the infinitive ending to roots of different shapes, and then test it on a wide variety of real or made-up verbs to see which ones are most likely to belong to a particular class, and why.
Course summary

(4) Overall goal: expose you to a variety of concepts, techniques and skills that are useful in computer-assisted analysis of language
  - Using Perl scripts to process text, and do simple tasks like grapheme-to-phoneme conversion
    - For more hands-on experience with using regular expressions and text processing, check out Ling 80G (Intro to Unix)
  - Finding boundaries
    - \( n \)-gram transition probabilities between phonemes to find word/morpheme boundaries
      - \( n \)-gram probabilities between words or parts of speech are also often used to find phrase and sentence boundaries
    - Successor counts (Harris)
    - Minimum Description Length: economical to treat recurring substrings as single units
  - Comparing phonemes and strings
    - Aligning and comparing sequences with String edit (Levenshtein) distance
    - Calculating similarity of phonemes using shared and unshared natural classes
  - Morphological parsing
    - Introduction to the task: what properties should a morphological parser have?
    - Balancing look-up and decomposition; when one or the other is necessary
    - Simple parsing using Microsoft Excel
    - A fancier formalism: Finite State Machines
  - Generating new, morphologically complex forms
    - ...with FSA's, using a hand-crafted grammar
    - ...with rules, using an inductively learned grammar
    - ...by analogy, using no grammar (just a lexicon of known words)
  - Finding generalizations from large databases

(5) A couple topics that would logically come next, if we had time
  - An elementary introduction to statistics, for testing how significant an observed pattern is (are \( C_iV_iC_i \) roots really underattested in English? should we trust the observation about height harmony in Spanish conjugation classes?)
  - Statistical models of “wordlikeness”: what makes \( \text{mip} \) and \( \text{pack} \) perfect words of English, while \( \text{tweep} \) and \( \text{smick} \) are sort of odd and \( \text{bnick} \) and \( \text{skaog} \) are impossible.
  - Relating data to theory: learning rules/constraint rankings on the basis of (possibly messy) input data; in other words, a model of how the learner constructs the hypotheses to be tested
  - Constructing databases to suit your purposes is also another important topic that deserves some attention!

(6) Computers, linguistics, the future, and you:

A few of the things I hope you take away from this class
  - A basic understanding of concepts like \( n \)-grams, minimum description length, finite state machines, etc., so when people mention them, you will know what they are talking about
  - A sense of some of the ways in which computational techniques can be useful in carrying out linguistic analysis (in both practical and theoretical ways)
  - An eagerness to turn to computers to organize data and solve problems, both in linguistics and in other areas

Thanks for your attention and patience!