

Assignment 2: Learning rule orderings

Due: Sept. 23

1. The program `italian.pl` (presented in class) provides a possible (but extremely stupid and inefficient) approach to finding a rule ordering that is consistent with the data. In psuedo-code:

```
Pick one rule (R1) at random;
Pick a second rule (R2) at random;
Swap R1 and R2 in the list of ordered rules;
```

Can you think of a more sensible approach, that might guide the learner to modify the current hypothesis in a more efficient way? Explain your proposal in prose (≈ 1 paragraph) and try to formalize it in pseudo-code

- Optional: try to implement your idea by modifying the *italian.pl* program. If your idea requires getting Perl to do something that we haven't seen before, ask me and I can try to point you to the relevant commands. (This exercise would be very helpful in cementing your new-found Perl skills, but it is not required, since I want to leave you time to do the readings)
2. Read Hutchinson, chapter 1, on basic terminology to characterize learning algorithms. Now consider the following learning agent: a phonology student, whose task is to find the solution to a typical phonology problem set. Characterize the learning task. What is the training set like? (§1.1; open/closed domain, clean/noisy, etc.) How would you characterize the data (§1.2)? What is the solution space (or what determines it)? What type of algorithm(s) do such agents tend to employ (§1.5)? Is it supervised? unsupervised? How do you know that a solution is right? Does order of examples play a role?

Now think about children (infants) learning the phonology of their language. How does the task differ from that of a phonology student? Is there a difference between learning phonotactics (the inventory of the language, possible combinations) and learning alternations?