36-491 HOMEWORK 2

Due: Wednesday 9/18/13

For this homework, you'll need to download/install the `RecordLinkage` and `tree` libraries. We'll be using the `RLdata500` data set in R which you can load into your workspace by (after typing the command `library(RecordLinkage)`) typing `data(RLdata500)`. Typing `ls( )` then shows two objects: the actual text data `RLdata500` and `identity.RLdata500`. See `help(RLdata500)` for more details.

Our goal is to build supervised models to predict whether or not pairs of records match. To do that, we need labels (which we have, luckily, using `identity.RLdata500`) and similarity metrics for the 124,750 pairs of text records.

1. We want to build a matrix with 124,750 rows and 8 columns. The rows correspond to pairs of records, and the columns correspond to the similarity metric for the different fields for the pair of records. For example, the first row would be for the record pair (1,2); the second row for (1,3), etc. The first column is for `fname_c1`, the second for `fname_c2`, etc. The eighth column is a binary indicator of whether or not that pair of records matches (1 = yes; 0 = no).

   Fill in the first seven columns of your matrix with the Jaro-Winkler scores for each field; the eighth column should indicate match/non-match.
   In this problem, treat all fields as text strings, even the birthdate information.

   The JW score requires character strings; you may need to use `as.character( )` on the field values. Also, if one or both of the strings is NA, JW = NA.
   One coding suggestion would be to first create a 500 x 500 matrix of JW scores for each field and then build the larger matrix by extracting the upper triangular part of the matrices for each field.

2. Use your matrix to fit a logistic regression model predicting whether or not the pairs of records are a match using all fields except `fname_c2, lname_c2` as predictor variables. (The "second" names have very few values.)

   Which fields were significant predictors of being a match? Anything surprise you?

3. In the above, we chose to treat the numeric birthday variables as text (reasonable assumption). What if instead of using a JW score, we just used exact matching on birth year, month, and day? How would that change our results from 2)?

4. We can also use classification trees to try to link our records. (`library(tree)`, `help(tree)`). Use your five JW scores to predict matches/non-matches with a classification tree (again, leaving out the "second" names).

   Plot your tree and describe it. What were the most useful variables?
   Describe the combinations that are likely to be matches (high match probability at the leaf) and those not likely to be matches (low match probability at the leaf).

5. Logistic regression and the classification tree both predict the probability of a match. We can choose our match probability threshold (e.g. 0.80, 0.90, 0.95, etc).

   Recall that we can summarize performance of these models using sensitivity and specificity and the corresponding ROC curve. (See notes from 9/11 for formulas.)

   For each model, calculate the performance values for match thresholds of [0.1, 0.2, ..., 0.8, 0.9] and create a ROC curve, plotting *1 - sensitivity* on the x-axis and *specificity* on the y-axis. (You'll need to get the predicted probabilities from the models; see `predict.tree` and `predict.glm`.)

   (You can draw in the extrapolated curve between the points. There are R packages that calculate the ROC curve, but it's not necessary for this HW.)

   Compare your two models.
   How do you think they did? Which one would you pick and why?

6. Repeat 1-5) using the Levenshtein similarity instead. Compare the results for the two similarity metrics? Anything surprising? Which would you choose?

7. You probably noticed that it can sometimes be difficult to model matches when there are so few in the data set (e.g. your Jaro-Winkler logistic regression). How might your results change if you had 50 matches and 50 non-matches?

   Randomly select 50 non-matches and run a logistic regression using the Jaro-Winkler similarity scores for 50 matches and 50 non–matches.
   Calculate the corresponding ROC curve (as done in 5). How did it do?