Data Analytics and Machine Learning | Prof. Lochstoer

==========

Problem Set 6

==========

Use the DJIA_Headline_News.csv CSV dataset available on CCLE for this exercise. The data is downloaded from www.kaggle.com (a really cool website for data and code, if you haven't seen it). You can and should look at the raw data using Excel before starting this exercise. The first column is date, the second columns is "label". This is 1 if the Dow Jones goes up or stays the same during that day and 0 otherwise. The remaining columns are 25 news headlines that morning.

Creating a sentiment index from text data

Before you start, please download the script "CleanUpScript_PS6.R" from CCLE. Run it on the data to do some pre-pre-processing so we're all starting from the same data set.

1. Use Corpus(VectorSource(data)) to load the corpus. VectorSource makes each line a document, so now each document corresponds to a different date in the dataset.

2. Pre-process the data as in the lecture notes. Feel free to use the code from Code Snippets Topic 5 on CCLE. That is, remove numbers, make all lower case, remove stopwords, stemming, etc.

3. As in the lecture note, create a DocumentTermMatrix, call it dtm. Run the line "inspect(dtm[5:10, 801:810])" Notice that the matrix is quite *sparse* (a lot of zeros).

4. As in the lecture note, create a freq matrix as the column sums of dtm. Show in a bar plot the frequency of words that occur more than 1000 times.

5. Create a wordcloud of the 100 most frequent words. Based on this (and 4.), how would you characterize the typical headline in terms of the news subject (economic, entertainment, domestic affairs, foreign affairs, etc.)? Are there words that, intuitively, can matter for the stock market returns that day?

6. Create the data "y_data <- as.factor(data$Label)" and "x_data <- as.matrix(dtm)". You will try to construct an index based on the words in dtm that predicts the direction of stock returns.

7. Split the data into a training data-set, based on data up to and including 2014-12-31. The remaining data should be used for actual out-of-sample testing.

8. The text data is very noisy. You will create a sentiment index based on words you think are likely to matter for the stock market. So that we all are considering the same words, define a new dtm-variable as follows:

```
dtm_sentiment <-
dtm[,c("invest","growth","grow","high","strong","lead","bankrupt","good","bull","bear","interest","market","hous","rate","oil","loss","weak","low","fear","poor","risk","stock","debt","financi","fiscal","reserv","crash","war","recess")]
```

Note how I express words in their stemmed form. What is the overall frequency of these words in the database? How do these frequency numbers compare to the frequencies of the words you plotted in sub-question 4.?

9. Fit a standard logistic regression to the y_data using the data in dtm_sentiment. Report the results. Are any words signficiant? Which seem to be the most important?

10. Create the ROC curve for this model. Is it better than random? You likely want to use the predict function to get the model prediction.

11. Now, fit an elastic net model with alpha = 0.5 using cross-validation. First, give the objective function of such a model (using the notes). Report the results by plotting the fitted coefficients and the cross-validation curve as in the lecture notes. Also, which words are the most important?

12. Create the ROC curve for this model. Is it better than random? Is it better than that in 10.?

13. Now, using the ***test sample*** and your two fitted models, what is the proportion of days your model would have made the right prediction in this new sample? Report this for both models. Are they better than random (50/50)? Which model is best?

14. Finally, create, in the training sample, 63 day moving average of the prediction (pred_ma) from both models, as well as the y-variable (y_ma). Plot pred_me versus date. Does the sentiment index behave reasonably? Run a standard ols of y_ma on pred_ma. What is the R2 of each model? Are the sentiment scores and stock returns significantly positively related, as the model would predict at this lower frequency?