STAT 6240 - HW 3 - Due 2/21

Due Date: February 21st, Thursday, 6:10 PM

Your homework submission should contain: (i) Your R code, (ii) outputs from R (including figures), (iii) answers to the questions.

You should use *R Markdown* or *R Notebooks*. **Do not print out your homework.** Save it as a pdf (or an html) file and upload it to Blackboard.

Q1. Clickthrough Rate Analysis (60 points)

In the first two questions of this week's homework assignment, you will analyze HW 0's dataset (on Wikipedia searches) using various classification methods.

1. Get the dataset and create a new dataset with all of the search events (`searchResultPage`) in the original dataset. The new dataset should also include ALL of the variables for these `searchResultPage` items.
2. To your new dataset, add a new variable called `clickthrough` which should be `TRUE` if the user clicked on a link immediately after this search.
3. Use either the `strptime` or the `substr` functions (or one of `str_sub`, `ymd_hms` if you are Hadley Wickham fan) to extract the hour and the minute of the search, and add these variables to your dataset.
4. Randomly sample 10% of your dataset and store it as a new dataset. This will be your training data. Store the other 90% of the data in a separate data frame, and this will be the testing data. *Please note that, you should use a large portion of the data as the training set in any other problem you face in the future. However, your computer might not be ideal for fitting a classifier with very large samples sizes, and we will work with a small sample of the full data in this exercise.*
5. Treating this as a classification problem, where `clickthrough` is the label, fit a classifier using:
   - Naive Bayes
   - LDA
   - QDA
   - Logistic Regression
   - Decision Trees (without bagging)
6. Evaluate each of your classifiers on the testing set. Plot the resulting ROC curves and calculate the AUC score. In addition, calculate the ROC curves and the AUC score for the training data.
7. Summarize your findings in two paragraphs. Detail which procedures had better (or comparable) performances. Relate the performances of each classifier to its assumptions. Which variables were found to be important?

Q2. 30 Second Check-in (35 points)

In this part, you will repeat the same analysis with a different variable on the same dataset.

This step should not take your more than half an hour if your code from the first question is reusable.

1. As before, create a new dataset with all of the search events (`searchResultPage`) in the original dataset. The new dataset should also include the variables for these searches.
2. To your new dataset, add a new variable called `check30` which should be `TRUE` if the user clicked on a link after this search, and then checked in at this page for at least 30 seconds or more.
3. Repeat your analysis from the previous part. You will:
   - Add the hour and minute of the search to the dataset.
   - Obtain a random sample for training and testing.
   - Fit different classifiers.
   - Evaluate these classifiers on the training and the testing data.

You do not need to provide a summary for this question.

Q3. Intro to Avito Duplicate Ads Detection (5 points)

In this question, you will work (or more likely, attempt to work) on the "Avito Duplicate Ads Detection" competition, which ran between May-July 2016. The competition is available at https://www.kaggle.com/c/avito-duplicate-ads-detection. Go to the website and read "Description" and "Evaluation" under the "Overview" tab, and the data explanations under the "Data" tab.

1. Sign up on kaggle.com.
2. Join the competition as a "late submission", and download all files from the "Data" tab.
3. Your task is to come up with a classifier with the highest possible AUC. Try different classifiers and variable transformations to improve your model. At this stage, one of two things will happen: Either you will have some issues with the dataset, or, being the data mining master you are, you will build numerous successful models.

If you face any issues even when you attempt this problem (possibly due to complexity of the dataset), write a short report (1 or 2 paragraphs) that details your experiences. The report should include a list of issues that make this dataset particularly hard. *I expect at least 90% of the class to take this option.*

If you can successfully fit a decent model, then evaluate the quality of your model on Kaggle by performing out-of-sample predictions on the test data (use `ItemInfo_test.csv` and `ItemPairs_test.csv`) and uploading your predictions to Kaggle. You can see how well your method worked on the test dataset under the "Leaderboard" tab. You will get extra points depending on your AUC score on the leaderboard. See the following table for the specifics:

AUC Score	Extra Points
0.70-0.75	5
0.75-0.80	25
0.80-0.85	100
0.85-0.90	200
0.90-0.95	600
>0.95	10000

If you earn 100 extra points, then you could skip out on one homework assignment. By extension, if you earn 600 points, you will not have to submit any homework assignments for the rest of the

class. If you earn 10000 extra points, then you will be crowned as *the Champion of STAT 6240*, will immediately receive an A for the class, and everyone in the class (including myself) will call you "Professor".

We will discuss your solutions and/or issues with the dataset in class.

Some hints:

- If you have no idea where to start from, Kaggle has numerous "Kernels" which are R Notebooks prepared by some of the competitors. These "Kernels" contain code and (usually) a decent approach to the problem. You can find them under the "Kernels" tab for the competition. At this link you can find an R Kernel that prepares the data, creates very basic features and uses logistic regression for prediction.
- If you are loooking for inspiration, checkout the interviews with the 1st, 2nd and 3rd place teams of this competition.