Homework Assignment 3

Due May 2 Midnight

Instructions:

This is an individual homework. Please do not work in groups. I want to make sure everyone has an understanding of how to do this.

In this homework, you'll you've continue to build the Machine Learning pipeline by combining what you have been doing in labs and your previous homework(s). The goal is to improve the pipeline based on the feedback from previous assignments, and add a few components based on what we've covered in the past few lectures and labs. More specifically, you need to:

Coding Assignment:

1. Fix and improve the pipeline code you submitted for the last assignment based on the feedback from the TA. if something critical was pointed out in the feedback, you need to fix it. You'll get the last homework back by ends of thursday so you'll still have time before this one is due to address those comments.

2. Add more classifiers to the pipeline. It should at least have Logistic Regression, K-Nearest Neighbor, Decision Trees, SVM, Random Forests, Boosting, and Bagging. The code should have a parameter for running one or more of these classifiers and your analysis should run all of them.

3. Experiment with different parameters for these classifiers (different values of k for example, as well as parameters that other classifiers have). You should look at the sklearn documentation to see what parameter each classifier can take and what the default values sklearn selects. The labs should be helpful here.

4. Add additional evaluation metrics that we've covered in class to the pipeline (accuracy, precision at different levels, recall at different levels, F1, area under curve, and precision-recall curves).

5. Create temporal validation function in your pipeline that can create training and test sets over time. You can choose the length of these splits based on analyzing the data. For example, the test sets could be six months long and the training sets could be all the data before each test set.

Analysis:

Data: projects_2012_2013.csv

6. Once you've set up the improved pipeline, you should apply it to solve the following problem:

The problem is to predict if a project on donorschoose will not get fully funded within 60 days of posting. This prediction is being done at the time of posting so you can only use data available to

you at that time.. You can read about the problem here but the data and problem are slightly modified. The data is a file that has one row for each project posted with a column for "date_posted" (the date the project was posted) and a column for "datefullyfunded" (the date the project was fully funded - assumption for this assignment is that all projects were fully funded eventually). The task is to predict if a project on donorschoose will not get fully funded within 60 days of posting.

The data spans Jan 1, 2012 to Dec 31, 2013. You should have your validation/test set be a rolling window of 6 months (which should give you three test sets). The training sets should be everything from 1/1/12 to the beginning of the test set.

The code should produce a table with results across train test splits over time and performance metrics (baseline, precision and recall at different thresholds 1%, 2%, 5%, 10%, 20%, 30%, 50% and AUC_ROC)

Report:

You should also write a short report (~2 pages) that compares the performance of the different classifiers across all the metrics for the data set used in the last assignment. Which classifier does better on which metrics? How do the results change over time? What would be your recommendation to someone who's working on this model to identify 5% of posted projects to intervene with, which model should they decide to go forward with and deploy?

The report should not be a list of graphs and numbers. It needs to explain to a policy audience the implications of your analysis and your recommendations as a memo you would send to a policy audience.