MF 850: Advanced Computational Methods

Professor Gustavo Schwenkler Fall 2019

Final project

Due: Monday, December 17, by end of the day (midnight) EST
via email to Professor Schwenkler (gas@bu.edu)

Assignment

The file "mf850-finalproject-data.csv" contains monthly stock return data for publicly traded companies in the United States during the time span between December 2014 and October 2018. The file also contains fundamentals data characterizing each company in each month, the monthly returns and realized volatilities of the S&P 500 index, and some monthly sentiment measures. Your first task consists of constructing forecasts of the monthly stock return of a company based on the characteristics of the company and the market. Your second task consists of constructing predictors of whether a stock will grow or fall over the course of a month based on the characteristics of the company and the market.

To achieve these tasks, you may use any of the machine learning tools we studied in class. Your goal is to construct and train models that can generate accurate forecasts of the one-month stock return of a company, and of the grow-fall indicator. You may construct any models that work well in your opinion, as long as your model choices are motivated by an empirical analysis of the data which you will summarize in a final paper to be handed to Professor Schwenkler.

Data

Each row of the file "mf850-finalproject-data.csv" contains a realization of the monthly stock return of a company, the monthly return and realized volatility of the S&P 500 during the same month, sentiment measures, and several firm fundamentals. Overall, the file has 88 columns:

- The first column "Date" gives the last day of the month in which the stock return is observed.
- "retmonth_spx" gives the monthly return of the S&P 500 index during the concurrent month, adjusted for dividends, splits, mergers, acquisitions, and other corporate events.
- "realized_vol_spx" gives the realized volatility of the S&P 500 index during the concurrent month, calculated as the standard deviation of the daily returns during the last 20 trading days of a month.
- "compid" is an anonymized company identifier.
- "Close" is the closing price of the company's stock on the first day of the month.

- “Adj_Close” is the closing price of the company’s stock on the first day of the month, adjusted for dividends, splits, mergers, acquisitions, and other corporate events.

- “Volume” gives the average trading volume of the stock over the course of the previous month.

- “Adj_Volume” gives the average trading volume of the stock over the course of the previous month, adjusted for dividends, splits, mergers, acquisitions, and other corporate events.

- “Industry” gives the industry of the firm.

- “RETMONTH” gives the monthly stock return of the firm from the end of the previous month to the end of the current month, adjusted for dividends, splits, mergers, acquisitions, and other corporate events.

- “sentiment_bullish”, “sentiment_neutral”, and “sentiment_bearish” measure the percentage of individual investors who are bullish, neutral, and bearish on the stock market, respectively.

- All other columns are described in the Excel sheet “indicators.csv”.

Deliverables

You should deliver to Professor Schwenkler via email at `gas@bu.edu` the latest by Monday, December 17, end of the day EST, the following items:

(i) A written summary paper of the empirical analysis of the data that motivated your modeling choices. This paper should be at most 15 pages long, and should include all necessary figures, tables, and estimates you computed during your analysis. Any codes you use do not need to be printed out and added to the summary paper. Instead, you should email your codes as a Matlab or R script to Professor Schwenkler.

(ii) A Matlab or R script that includes one function which takes as input all variables of the file “mf850-finalproject-data.csv” except column “RETMONTH”, and returns a forecast of the monthly stock return “RETMONTH” based on the inputs. Note that not all the variables may be relevant (you may wish to run some variable selection method). However, your function should still take as input all variables and assign a zero coefficient to the variables that are irrelevant. The file should be self-contained. That is, any libraries that need to be loaded to run the function should be loaded at the beginning of the script. Any special functions that are used in the script should be defined at the beginning of the script. And any variables that are used throughout the script should be properly declared and defined. You should add comments to your script to make it easily interpretable.

(iii) Another Matlab or R script that includes one function which takes as input all columns of the file “mf850-finalproject-data.csv” except column “RETMONTH”, and returns a prediction whether a stock associated with the inputs will grow or fall over the course of the month. Note that not all the variables may be relevant (you may wish to run some variable selection method). However, your function should still take as input all variables and assign a zero coefficient to the variables that are irrelevant. The file should be self-contained. That is, any libraries that need to be called to run the function should be called at the beginning of the script. Any special functions that are used in the script should be defined at the beginning of the script. And any variables that are used throughout the script should be properly declared and defined. You should add comments to your script to make it easily interpretable.

Your functions from parts (ii) and (iii) do not need to be consistent with each other. That is, it is OK if your function from part (ii) delivers a forecast of the monthly stock return of a firm that is positive, while your function of part (iii) predicts that the same stock will fall over the course of the a month.

Grading

Your final project will be graded on a scale from 0 (worst possible grade) to 40 (best possible grade) as follows:

- *Accuracy of your stock return forecasts (10 points).* Your stock return forecast function will be tested on a special test data set selected by Professor Schwenkler. The test data will include the same variables as the file “mf850-finalproject-data.csv”, but will be out of sample. That is, the test data is not included in the file “mf850-finalproject-data.csv”. The test data will include at least 3000 stock return observations of U.S. companies over the course of a month after October 2018 and before November 2019. The month that will be used for testing purposes will be randomly selected by Professor Schwenkler. Accuracy will be measured by the out-of-sample R^2. Professor Schwenkler will rank all forecasts from most to least accurate and will assign a grade for accuracy according to your achieved rank. If your stock return forecast script does not run on any of Professor Schwenkler’s computers, you will receive a grade of 0 points for this category. Make sure to test your script multiple times on many different computers to ensure that your script will run properly!

- *Accuracy of your grow-or-fall forecasts (10 points).* Similarly, your grow-or-fall forecast function will be tested on the same test data used to test your return forecast function. Here, however, accuracy will be measured through the ratio of correct forecasts (That is, the fraction of all forecasts for which a prediction of “grow” was made when the stock actually grew, or “fall” when the stock actually fell). Professor Schwenkler will rank all forecasts from most to least accurate, and will assign a grade for accuracy according to a your achieved ranking. If your grow-or-fall forecast script does not run on any of Professor Schwenkler's computers, you will receive a grade of 0 points for this category. Make sure to test your script multiple times on many different computers to ensure that your script will run properly!

- *Model justification (15 points).* Professor Schwenkler will judge whether your model choices are justified based on the analysis that you describe in your written summary. Models that appear unjustified (or randomly selected) will receive a low model justification score.
- *Model uniqueness (5 points).* Your model will also be evaluated along the uniqueness dimension. If your predictions are based on model that multiple submissions use, then you will receive a low model uniqueness score.

Rules

- You may work on the final project in a team consisting of **4 to 6 people.** Groups of less than 4 people or more than 6 people will not be accepted.
- Each team should email Professor Schwenkler by **end of the day (midnight) EST on Sunday, December 1**, and let him know who the team members are. Once a team is announced to Professor Schwenkler, no more changes to the team composition can be made.
- Each team should hand in one summary paper (and used codes if necessary), one function for part (ii), and one function for part (iii).
- Your team can collaborate with other teams. However, teams that hand in very similar projects will be penalized with a low *model uniqueness* score.
- You may use additional data than the one you have been provided with to construct your model. Any additional data that you use must either be saved into a file that you will share with me and hard-coded into your model, or should be downloaded in the process of running your codes. However, you may not download, hardwire, or use in any form data that realized after October 2018. Any data that is not included in the dataset that Professor Schwenkler provided to you needs to be explained in the summary paper, the source needs to be indicated, and Professor Schwenkler needs to be able to verify the sources. Failure to credibly demonstrate that the additional data you use does not cover the testing period will be considered cheating and will be penalized as such.
- You may only use data that covers the same time period as the data Professor Schwenkler provided to you for the purpose of training your model. In other words, if your algorithm is such that when I run it on my computer it needs to first fit a model to some training

data, and then it generates the prediction for the test data, then the training data set may only include observations spanning the time period December 2014 through October 2018. Failure to demonstrate that the training data that you use does not span the period after October 2018 will be considered cheating and will be penalized as such.

- You may not use data post October 2018 for the training of your model, but you may use lagged data for building your predictions. Professor Schwenkler will have all of the data between November 2018 and October 2019 available to test your predictions. However, your codes may not download data spanning this out-of-sample period for any reason, and they may not have hardwired out-of-sample data for any reason.

- You may call Python code from R or Matlab if you know how to do that. Professor Schwenkler does not code Python or run Python codes, so this may be a risky strategy in case Professor Schwenkler cannot run your algorithm on his computer. But if you desire to follow this path, make sure to indicate in your summary paper what additional R or Matlab libraries will need to be loaded to run your codes. If in the end Professor Schwenkler cannot run your algorithms on his computer, then that will be viewed as a failing project.

- The test data will have the same columns and column names as the data you were given.

- The test data may not contain all the companies in the data that you have right now. Some companies may have failed, and may no longer be included in the test data. However, the company id will remain the same in the test data as in the training data.

- In an ideal world, the codes that you submit should not run the training of your model. In other words, the codes should already have the trained model hard-coded in the script (all the parameters are given in the script), or your trained model should be saved into a file in the workspace which you can share with Professor Schwenkler so that he can upload it when he runs your codes. If you do not know how to save estimated models into files, these links may be useful:

  `http://machinelearningmastery.com/finalize-machine-learning-models-in-r/`
  `http://www.mathworks.com/matlabcentral/answers/`
  `264160-how-to-save-and-reuse-a-trained-neural-network`.

- There is no a clear expectation about what constitutes a good prediction. What recent research shows is that an out-of-sample R^2 of about 1% for the return prediction and a correctness rate of about 60% for the direction prediction should be extremely good.