Estimating Sectors – Readme

Library Name	Version used	Required
pandas	0.16.2	Yes
numpy	1.11	Yes
scikit-learn	0.16	Yes
wordcloud[1]	1.2	Yes
Selenium[2]	3.0	No (only used to fetch data)
yahoo-finance[3]	1.4	No (only used to fetch data)

The project is divided into 10 modules:

1) caps_master.py: This is the main module and calls all other required modules. **To run the model and generate requisite data and results, it is sufficient to run this file (takes around 20 minutes on a high end desktop without cross validation).**
2) config.py: This module contains configuration parameters. Specifically, whether the raw data needs to be fetched (changing this parameter to True is **STRONGLY DISCOURAGED.** It takes around 12 hours for the raw data to download) and if one fold cross validation needs to be performed (changing this parameter to True will add around 15 minutes to run time).
3) caps_common.py: This module contains functions which are used by other modules. Function get_ticker_list generates a universe of tickers from the file `companylist.csv` available at http://www.nasdaq.com/screening/companies-by-industry.aspx?exchange=NYSE eliminating tickers without a Sector assignment and tickers with a caret or dot in them.
4) F1_get_fundamental_data.py: This module will fetch various fundamental data from Morningstar website. You will need to use Firefox browser for this to work. In addition, you will need to install latest version of geckodriver[4]. This module will not be run by default. To run this, change parameter 'fetch_raw_data' to True in config.py. This is **STRONGLY DISCOURAGED** as it takes around 12 hours to finish. If this is changed to True, you will also need to provide the path of the firefox exe file. The data is stored in file `fundamental_data.csv.`
5) F2_get_market_data.py: This module fetches the adjusted closing price data from yahoo finance using yahoo-finance library. It then converts the data prices into returns. The converted data is stored in file `returns_data.csv.`
6) F3_clean_data.py: This (and all higher number modules) is run by default. This module fixes basic issues with data (such as use of '0.2Mil' instead of 200000) and converts all data in appropriate formats.
7) F4_create_features.py: Create the features from fundamental data (various financial ratios and 'Summary' feature which will be engineered later).

[1] https://github.com/amueller/word_cloud
[2] http://selenium-python.readthedocs.io/
[3] https://pypi.python.org/pypi/yahoo-finance
[4] https://github.com/mozilla/geckodriver/releases

8) F5_create_training_data.py: Splits full data set returned by previous module into two parts: train and test data. Also creates 30 word counts features (1 for each of the top 30 most common words in training data 'Summary'). Also creates 11 features, 1 for each of the sector correlations (to returns of the firm). Sector correlations are calculated using training data only.
9) F6_preprocess_features.py: Replaces missing values with median values of that feature and scales the data to have 0 mean and unit variance.
10) F7_train_models.py: Fits a linear SVC classifier to the data.