

Project 1: Stock price reaction to analyst recommendations

(Due 11:59pm on Friday, November 6)

Overview

In this project, you will evaluate the stock price reaction to changes to analyst recommendations. Each observation will represent an analyst either upgrading or downgrading a stock from their previous recommendation.

The deadline for this project is 11:59pm on Friday, November 6.

This project has three parts, each worth a third of the total mark for the project:

Part 1: Combine stock price data distributed across multiple files.

Part 2: Prepare the analyst recommendation data distributed across multiple files.

Part 3: Perform the event study

This project is to be completed individually by each student. You should not collaborate. You are allowed to post questions about the project in the thread I will create in the Discussion Board. However, please do not answer any questions posted by other students. I will answer any project-related questions myself.

Files

You received a zipped file with the following contents:

```
<Zid>/
|  <Zid>_event_study.py
|  README.txt
|  TICKERS.txt
|__data/
|   |   ff_daily.csv
|   |   <tic>_prc.dat
|   |   <tic>_rec.csv
```

where <Zid> represents your Zid, <tic> represents a ticker (in lower case) and:

- <Zid>_project1.py contains the functions you will use to perform this study. Some of the functions are already written, while others you will write yourself. Please see the instructions below for more information.
- TICKERS.txt contains a list of tickers, one per line. These tickers may be in upper or lower case.
- README.txt contains information about how the stock data is stored in the .dat files.
- data is a folder containing all the data you will need to complete this project. Inside this folder:
 - <tic>_prc.dat contains stock price data for the ticker <tic>.
 - <tic>_rec.csv is a CSV file containing analyst recommendation data for the ticker <tic>.

Instructions

1. Decompress the contents of the zip file onto your computer.
2. Complete the user-written functions in `<Zid>_project1.py`.
3. Submit the completed version of `<Zid>_project.py` of your project.

The following notes are **very important**:

1. Functions produced in one part of the project will be used in subsequent parts. That means you have to complete Parts 1, 2, and 3 below in sequence.
2. When completing your code, please keep the following in mind. We will run your functions through test cases designed to check specific parts of your code. The data we will use is different than what you received. This means that your code should be general (follow the instructions and it will be). For instance, you should not create a variable called `tickers` and then copy the specific tickers you received in your `TICKERS.txt` file. Instead, your code should read the `TICKERS.txt` file, produce a list of tickers, and store that in a variable.
3. When writing functions in the file `<Zid>_project1.py`:
 - Do not modify the function names or the parameters.
 - Only modify the parts indicated by the “” tag.
 - You do not need to import any other module. Please do not modify the import statements.

Part 1: Combine stock price data distributed across multiple files.

It is often the case that one needs to put together a data set from different sources. The source data may not be stored in a format that is easy to import into a spreadsheet or directly into Pandas. To simulate this real-world problem, your data contains stock price information distributed across many `<tic>_prc.dat` files. We are only interested in the files containing stock price data (`<tic>_prc.dat`) for this part of the project.

To complete this part of the project modify the following sections in `<Zid>_project1.py`:

1. Set the correct expressions for the constants `SRCDIR`, `TICKERS`, and `FF_CSV`, which contain the file location of the source directory, the `TICKERS.txt` file, and the `ff_daily.csv` file, respectively.
2. Set the correct expressions for the variables `SRC_COLS`, `SRC_COL_DTYPES`, and `SRC_COL_WIDTHS`. These represent the source column names, a dictionary from column name to Pandas `dtype`, and the column widths, respectively. To complete this part, you will need the information in the `README.txt` file.
3. Complete the indicated part of the function `get_tics`. This function reads a file with tickers and returns a list with formatted tickers.
4. Complete the indicated part of the function `dat_to_df`. This function reads a stock price data file and returns a Pandas DataFrame. This function is used by the function `mk_prc_df`, which you will also need to complete.
5. Complete the indicated part of the function `mk_prc_df`. The output of this function must meet the one specified in the docstring. This function will call the `dat_to_df` function you completed above.

Part 2: Computing Abnormal Returns and Preparing Analysts' Recommendation Data

For this part of the project, you will compute abnormal returns and prepare the recommendations contained in each `<tic>_rec.csv` file.

1. Complete the indicated parts of the function `mk_aret_df`. This function will produce a dataframe with the daily abnormal returns for each ticker. Abnormal returns are defined as the stock's return minus the market return. Market returns are provided in the `ff_daily.csv` file.
2. Complete the indicated parts of the function `read_rec_csv`. This function takes a ticker `<tic>` and returns a dataframe with the contents of the corresponding `<tic>_rec.csv` file.
3. Complete the indicated parts of the function `proc_rec_df`. This function will subset the dataframe from `read_rec_csv` to include only upgrades and downgrades from at most 30 firms, selected based on the number of recommendations.

Part 3: Performing the Event Study

1. Complete the indicated parts of the function `mk_event_df`, which creates a dataframe with each event which will be included in the study. An event will be either a downgrade or upgrade for a firm in a given day.
2. Complete the docstring of the function `mk_ret_dates`. You should try (but are not required) to follow docstring style used for other functions in this module. You do not need to write a 'Notes' section.
3. Complete the function `mk_ret_dates_by_group`, which is used by the `mk_ret_dates` function above.
4. Perform the event study by running the function `main`.