Fall 2019
FRE-GY 6883 Financial Computing
Course Team Projects

General Requirement

You are required to do class projects in groups (For Tuesday session, each team consist of 5 members; for Saturday session, each team will have 3 team members). You should elect one member of your group to be the team leader. Groups once formed cannot be changed midway through the project. The team lead is responsible to facilitate the planning of the project, and the entire team will plan the project under the guidance of your team leader. Planning involves identifying what should be done (tasks), who should do it (resources), when tasks should be done (time frames) and how tasks are best sequenced (dependencies).

Each team will submit PowerPoint slides and all the project files including source codes and executables tar/zipped to our course Web site 3 days before the presentation day (12/11/2019 11:55pm for Saturday Sections, and 12/14/2019 11:55pm for Tuesday Section). The project reports should include the Excel sheet about your research on S&P 500 stocks for their earnings, a drawing of project design, class declaration and data structures, program outputs and your PowerPoint slides. All the teams are requested to present and demonstrate their projects. Each team can resubmit one time on the presentation day. Your project will be judged by program efficiency, complexity and the success of your presentation. All the submission will be done by team lead.

Project Description
Evaluate the impact of quarterly earnings report on stock price movement

Programming Requirements:

- Use liburl to retrieve historical price data from Yahoo Finance: A function retrieves the adjusted close prices for selected S & P 500 stocks and SPY into memory.
- Create a set of classes such as class for stock to handle EPS estimate and price information.
- Use member functions or independent functions for all calculation. Overload a few arithmetic operators for vector/matrix.
- The stocks and their corresponding price information for each group should be stored in a STL map, with stock symbol as its keys.
- The expected AAR, AAR SD, and expected CAAR and CAAR SD for 3 groups are presented in a matrix. The row of the matrix is the group#, matrix columns are for AAR, AAR-SD, CAAR, CAAR-SD
- Use Excel Driver or gnuplot to show the CAAR from all 3 groups in one graph.

- Your program should be able to:
    - Retrieve historical price data for all selected stocks. Parse the retrieved data for dates and adjusted closing prices.
    - Calculate AAR, AAR-SD, CAAR CAAR-SD for each group
    - Populate the stock maps and AAR/CAAR matrix.
    - Show an Excel or gnuplot graph with CAAR for all 3 groups.
- Your program should have a menu of 5 options:
    - Retrieve historical price data for all stocks.
    - Pull information for one stock from one group.
    - Show AAR, AAR-SD, CAAR and CAAR-SD for one group.
    - Show the Excel or gnuplot graph with CAAR for all 3 groups.
    - Exit your program.

Calculation Details:

1. From Bloomberg terminal, use a query to pull 2019 1st quarter earnings releases (if a company's earning is far away from the 1st calendar quarter of 2019, select a quarterly earning close to 1st calendar quarter of 2019) for all S&P 500 stocks, sort and divide them into 3 groups:

    a. Calculate earnings surprise for each stock:

       Surprise % = (Actual EPS – EPS Estimate) ÷ EPS Estimate

    b. Sort all the surprises in ascending order, and split all the stocks into 3 groups with relatively equivalent numbers of stocks:
        i. Highest surprise group: Beat Estimate Group
        ii. Lowest surprise group: Miss Estimate Group
        iii. The rest stocks in between: Meet Estimate Group

2. Define day "zero" for a stock as the day the earning is announced.

3. Implement Bootstrapping:

    a. Sampling the stocks by randomly selecting 30 stocks from each group, total 90 stocks.

    b. Use libcurl lib to retrieve 61 days of historical prices for S&P500 stocks and ETF ticker SPY around the date of earning release (You could enhance our class example for this purpose).

    c. For each stock calculate the daily returns R_{it} for 30 days before the day "zero" and 30 days after: t = -30, -29,…-1, 0, 1,…, 29, 30:

R_{it} = ($Price_t$ – $Price_{t-1}$)/ $Price_{t-1}$

Using adjusted daily closing price for your calculation

d. Calculate the corresponding daily return R_{mt} for SPY for the same days.

e. Define abnormal returns as the difference AR_{it}= R_{it}–$R_{mt.}$

f. Calculate average daily abnormal returns for each group of stocks for all 60 reference days:

$$\mathrm{AARt} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{ARit}$$

g. Cumulate the returns on the first T days to CAAR:

$$\mathrm{CAAR} = \sum_{t=-30}^{T} \mathrm{ARRt},\ T = 30$$

h. Repeat steps a to g 30 times and then Calculate
   i. Average AAR and CAAR for 30 samplings for each group
   ii. Standard Deviation of AAR and CAAR for each group

4. Generate an Excel or gnuplot chart show the averaged CAAR of all three groups and discuss the impact the earning releases on their stock prices. Is there any conclusion you could draw from your project?

Project Tasks:

Task 1: Earnings research: sort stocks from S & P 500 into 3 groups based on their earnings and EPS Estimate from Bloomberg terminal.

Task 2: Project Design:

a) Create classes and data structure such as vectors, matrix and maps.
b) Figure out how to handle historical price retrieval from Yahoo Finance for all S&P 500 stocks and parse the retrieved data?
c) Figure out how to implement your Bootstrap algorithm?
d) Write member function or independent functions (with operator overloading) for all the calculation.

Task 3: Divide the project into modules and assign team members working on each module.
Task 4: Module Integration and Testing
Task 5: Presentation Preparation.