IE 423 Quality Engineering
Project Part 1, Due November 2, 2023

1. Introduction:

In the realm of financial markets, the movement of stock prices is influenced by various factors, both internal and external. As budding industrial engineers, the ability to manipulate, understand, and draw inferences from time series data is a crucial skill. This exercise aims to provide a platform to showcase your data manipulation skills in R/Python using Borsa İstanbul stock data while diving deep into quality engineering concepts.

2. Data:

You will retrieve your data from the provided link:
https://drive.google.com/drive/folders/1IMQmUOI_XNg5TSGZkbsglvIcoczSs3YR

The data comprises the closing prices of selected stocks from Borsa İstanbul. The frequency of the observations is 15 minutes. Under the shared folder, you will find the same information in two alternative formats, namely long and wide format. Long and wide formats are illustrated in Figures 1 and 2 respectively. You can choose one of them for your analyses. Note that the data is provided as csv files which are compressed with gzip.

You can find detailed information about these formats in the following links:

- https://www.statology.org/long-vs-wide-data/
- https://www.thedataschool.com.au/mipadmin/the-shape-of-data-long-vs-wide/
- https://libguides.princeton.edu/R-reshape
- https://stefvanbuuren.name/fimd/sec-longandwide.html

	short_name	timestamp	price
1:	AEFES	2012-09-17 09:45:00	22.3978
2:	AEFES	2012-09-17 10:00:00	22.3978
3:	AEFES	2012-09-17 10:15:00	22.3978
4:	AEFES	2012-09-17 10:30:00	22.3978
5:	AEFES	2012-09-17 10:45:00	22.5649

2848026:	ZOREN	2019-07-22 17:00:00	1.1800
2848027:	ZOREN	2019-07-22 17:15:00	1.1800
2848028:	ZOREN	2019-07-22 17:30:00	1.1800
2848029:	ZOREN	2019-07-22 17:45:00	1.1800
2848030:	ZOREN	2019-07-22 18:00:00	1.1700

Figure 1. The stock price data in long format

	timestamp	AEFES	AKBNK	AKSA	AKSEN	ALARK	ALBRK	ANACM
1:	2012-09-17 09:45:00	22.3978	5.2084	1.7102	3.87	1.4683	1.1356	1.0634
2:	2012-09-17 10:00:00	22.3978	5.1938	1.7066	3.86	1.4574	1.1275	1.0634
3:	2012-09-17 10:15:00	22.3978	5.2084	1.7102	NA	1.4610	1.1356	1.0679
4:	2012-09-17 10:30:00	22.3978	5.1938	1.7102	3.86	1.4537	1.1275	1.0679
5:	2012-09-17 10:45:00	22.5649	5.2084	1.7102	3.87	1.4574	1.1356	1.0725

50008:	2019-07-23 17:00:00	20.4800	7.7300	9.1400	2.47	3.2300	1.2100	2.8400
50009:	2019-07-23 17:15:00	20.5000	7.7200	9.1400	2.47	3.2200	1.2100	2.8400
50010:	2019-07-23 17:30:00	20.5000	7.7400	9.1300	2.46	3.2300	1.2100	2.8300
50011:	2019-07-23 17:45:00	20.4000	7.7000	9.1400	2.47	3.2400	1.2100	2.8200
50012:	2019-07-23 18:00:00	20.4600	7.7000	9.1400	2.47	3.2300	1.2000	2.8300

Figure 2. The stock price data in wide format

3. Requirements:

3.1. Software & Tools:

- R or Python environment for data manipulation and analysis.
- Relevant libraries/packages for time series data handling.
- Internet access to consult Google Trends or similar databases.

3.2. Prerequisite Knowledge:

- Basics of R or Python programming, especially related to data manipulation.
- Knowledge of sampling distribution, hypothesis testing, and confidence intervals.

4. Tasks:

4.1. Data Retrieval & Manipulation:

- Fetch the data using the link provided and load it into your R/Python environment. Organize data in chronological order, making sure the timestamps align for comparative analysis.
- Choose at least **6** stocks, ensuring:
  - They are from at least **3** different sectors (e.g., Banking, Industry, etc.).
  - The data spans at least **2** years for each index.

4.2. Identification of Outliers using Boxplots and 3-Sigma Limits:

- For each month in your data, generate a boxplot for the chosen indices. This visual tool will help you discern any outliers based on the Interquartile Range (IQR).
- As an alternative approach, employ the 3-sigma rule. Calculate the monthly mean and standard deviation for each series, and flag data points falling outside the range of (mean - 3standard deviation) and (mean + 3standard deviation) as outliers.
- Make sure to note these outliers and their respective year and months. For each identified outlier, shed light on potential reasons, possibly through preliminary research or a solid understanding of market dynamics.

4.3. Insights with Open Source Data:

Another interesting data source is Google Trends (http://trends.google.com/). The Wikipedia definition for Google Trends is:

"Google Trends is a public web facility of Google Inc., based on Google Search that shows how often a particular search-term is entered relative to the total search-volume across various regions of the world, and in various languages. The horizontal axis of the main graph represents time (starting from 2004), and the vertical is how often a term is searched for relative to the total number of searches, globally. Below the main graph, popularity is broken down by countries, regions, cities and language. Note that what Google calls "language", however, does not display the relative results of searches in different languages for the same term(s). It only displays the relative combined search volumes from all countries that share a particular language (see "flowers" vs "fleurs"). It is possible to refine the main graph by region and time period."

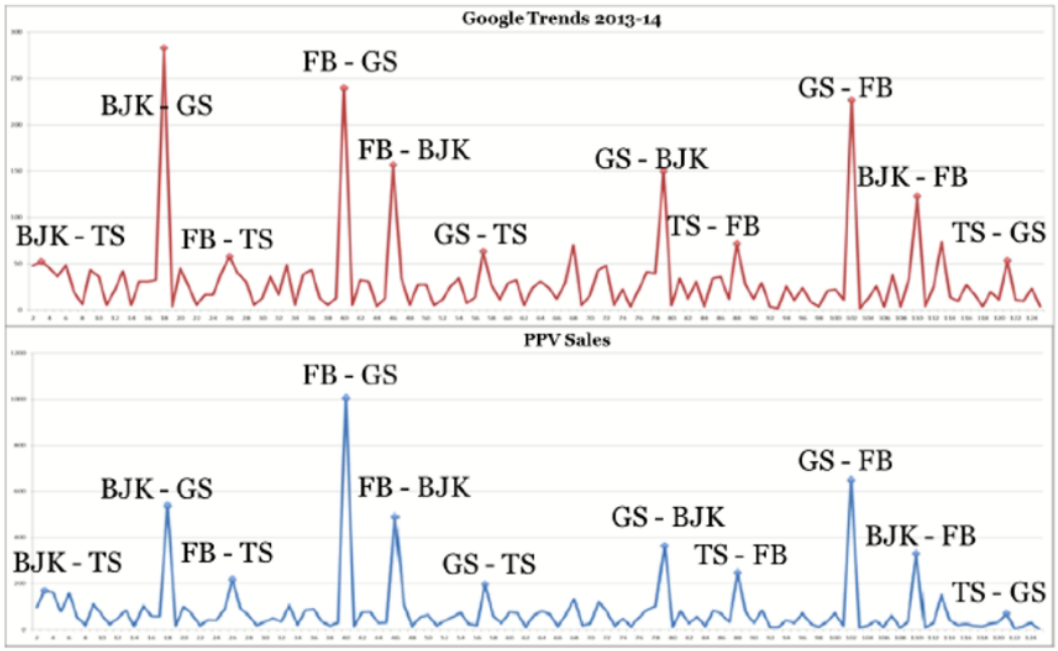

Figure 3. Search volume versus PPV sales

Sometimes it is interesting to check if search volume of a certain keyword has some relation to an event. For example, number of searches for "flu" can give you an idea about the number of people affected by the flu over the world. Another example is from a senior design project we have worked on last year. It was about understanding if Google Trends provide information about the number of pay-per-view (PPV) customers of Digiturk. To be more specific, Lig TV is a football channel of Digiturk holding rights to broadcast Turkish Super League live since 2001. It serves its customer through different channels such as the Internet platform "Digiturkplay" or satellite TV "Digiturk Plus". All

channels include a service called pay-per-view (PPV) providing customer to pay for the particular programs that they watch via private telecast. The aim here is to see if it is possible to forecast demand for PPV through the Internet search data, namely Google Trends Data. To achieve this, we obtained PPV data from Digiturk. For each soccer game, we obtained the daily search volume information for each team playing a game (i.e. if the game is between Beşiktaş and Fenerbahçe, we obtained the search volume for each team on the game day and sum these values to find out a measure that will quantify the popularity). The plot of the PPV sales versus the search volume information for each game is schematized in Figure 3. As you can see, Google Trends data is highly correlated with the sales.

Determine search keywords that can be relevant to the prices you consider in the earlier parts (i.e. if you are interested in AEFES you can check the search volume for "IST:AEFES").

- Use Google Trends to probe for events or relevant news coinciding with the dates of the detected outliers in your indices. Plot the line plot of the search volume and the data you obtained for each month. Note that you need to provide these plots for each measure of interest. You can also provide the scatter plot of the search volume and measure of interest to understand if there is a relationship between them.
- Systematically correlate any noteworthy findings between outlier data points and Google Trends' spikes or news items.

5. Report & Code Documentation:

Consolidate your results, visual aids included, into a detailed report. Wrap up your report with a summary, zooming in on the correlation between stock anomalies, sectoral influences, and real-world events.

Please solve the following exercises using R (http://www.r-project.org/) or Python (https://www.python.org/). You are expected to use GitHub Classroom and present your work as an html file (i.e. web page) on your progress journals. There are alternative ways to generate an html page for you work:

- A Jupyter Notebook including your codes and comments. This works for R and Python, to enable using R scripts in notebooks, please check:
  - https://docs.anaconda.com/anaconda/navigator/tutorials/r-lang/
  - https://medium.com/@kyleake/how-to-install-r-in-jupyter-with-irkernel-in-3-steps917519326e41

  Things are little easier if you install Anaconda (https://www.anaconda.com/). Please export your work to an html file. Please provide your *. ipynb file in your repository and a link to this file in your html report will help us a lot.

- A Markdown html document. This can be created using RMarkdown for R and Python. Markdown for Python

Note that html pages are just to describe how you approach to the exercises in the homework. They should include your codes. You are also required to provide your R/Python codes separately in the repository so that anybody can run it with minimal change in the code. This can be presented as the script file itself or your notebook file (the one with *.ipynb file extension).

The last and the most important thing to mention is that academic integrity is expected! Do not share your code (except the one in your progress journals). You are always free to discuss about tasks but your work must be implemented by yourself.

Please mention if you utilize Large Language Models (i.e. chatGPT, BART and etc). You are expected to provide your prompt and resulting response of these models at the end of your reports as Appendix if you have utilized any of these tools.

As a fundamental principle for any educational institution, academic integrity is highly valued and seriously regarded at Boğaziçi University.

Best of luck, and remember – the journey of analyzing is as essential as the conclusions drawn!