Predicting Earning Surprises Report

Group 1C

Carlos Perez, Ryan Garg, Junchi Wang, Rory Thomas, Fengze Yang

Oct 2022

ANCORA · IMPARO

MONASH University

Contributions

- Ryan Garg - 33154325
  - Investigated relevant modelling techniques for the problem
  - Created kNN classifier model to investigate connections
  - Researched the financial domain and context of the problem

- Carlos Weffer Perez - 33154414
  - Researched the financial domain and context of the problem
  - Performed exploratory analysis
  - Curated a kNN regressor for the data

- Rory Thomas - 33154368
  - Completed data pre-processing, wrangling and cleaning
  - Performed exploratory analysis
  - Generated meaningful conclusions for the project
  - Edited the final report

- Fengze Yang - 33047995
  - Worked on Linear Regression models
  - Looked into many aspects of linear models to concretely prove/disprove their relevance
  - Explore the external factors that have influenced stock price predictability over the past few years

- Junchi Wang - 30351197
  - Worked on Linear Regression models
  - Looked into many aspects of linear models to concretely prove/disprove their relevance

Context and Background

Purpose

The purpose of this project was to investigate and explore financial data to prove or disprove a hypothesis given to the project team that had worked in the past. The hypothesis given was:

> *“When a company has a positive earnings surprise, its stock price will increase. And when a company has a negative earnings surprise, its stock price will decrease.”*

Context

As this project was centred around the financial domain and more specifically earnings per share some context is required to understand it. Most of the niche topics used in this report are as follows:

- **Earnings Per Share (EPS):** Publicly traded companies (companies with stock available on the stock market) have to release an earnings report every quarter. This earning report contains a statistic called earnings per share which is “a company's net profit divided by the number of common shares it has outstanding”. This is often a good measure of a company’s value and thus has a positive relationship with share price.

- **Earning Surprise:** This connection to share price means investors are always trying to predict a company’s earnings per share. When the predictions are incorrect there is an aptly named earning surprise. If this is positive, the predicted EPS was less than the actual EPS and vice versa.

- **S.U.E Score:** Standing for “Standardized unexpected earnings, this was the main metric used to describe earning surprises in our data. It can be calculated using the formula shown in Figure 1. (Corporate Finance Institute, 2019)

$$SUE = \frac{EPS - EPS_{estimated}}{Standard\ Deviation}$$

Figure 1: Formula for S.U.E Score

Data Pre-Processing

At the start of this project, a zip folder full of datasets and supporting information for each of these datasets was provided. The parts of this data set that required direct Python-based usage were the following:

- A data frame containing earning reports data for the top 200 retail companies and one containing price data for the same companies
- A data frame containing earning reports data for the top 200 banks and one containing price data for the same companies

One issue that arose from how the data was provided was that up until this point, only CSV file manipulation had been covered in ADS1002 and the aforementioned data came in the form of sheets within XLSX files (“Microsoft Excel Open XML Spreadsheet”)(Gavin, 2018) as shown in figure 2. As a result, some preprocessing was required.

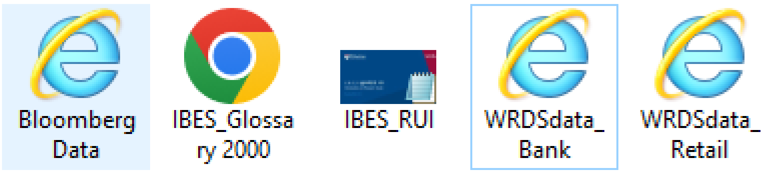
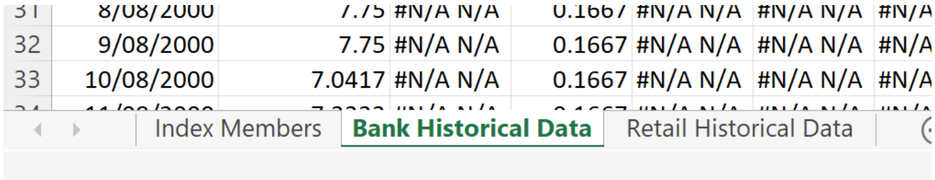


Figure 2: The data provided (WRDS files had earning reports, "Bloomberg Data" had price data)

"Bloomberg Data"

This workbook contained the data on the stock price for both the top 200 performing banks and the top 200 performing retail companies. This price data was split across two sheets with a sheet devoted to each sector and then an additional sheet containing ticker information. This is shown in Figure 3.



31	8/08/2000	7.75	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A
32	9/08/2000	7.75	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A
33	10/08/2000	7.0417	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A

Figure 3: The Sheets within the "Bloomberg Data.xlsx" document

This was then converted into a python-friendly format through the following process which is shown in figure 4. First, the desired sheet was right-clicked and the "Move or Copy" dialogue box was filled out to move the sheets to their own Excel workbook. Then, in the file menu, the new workbook was exported as an appropriately named CSV file.

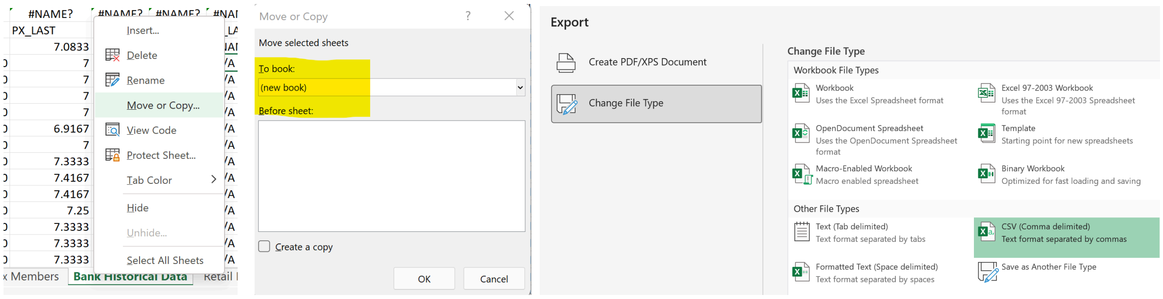


Figure 4: Steps for converting datasets to CSV files

These CSV files could then be loaded into and manipulated in Python without any hassle.

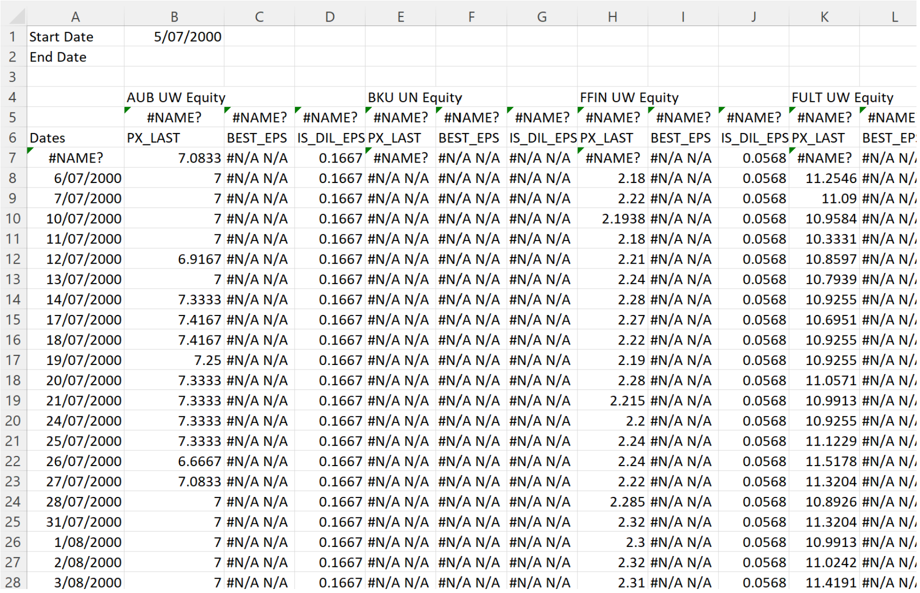
"WRDSdata_Bank" and "WRDSdata_Retail"

The "WRDSdata_Bank.xlsx" and "WRDSdata_Retail.xlsx" files were composed of a single sheet with the data frame we wanted in it. As a result, it was far easier to pre-process this data and make it suitable for use in Python. All that needed to be done was to export each XLSX file as a

CSV through the Excel “file” menu as shown in the third pane of Figure 4. After exporting these files were then fit for usage in Python.

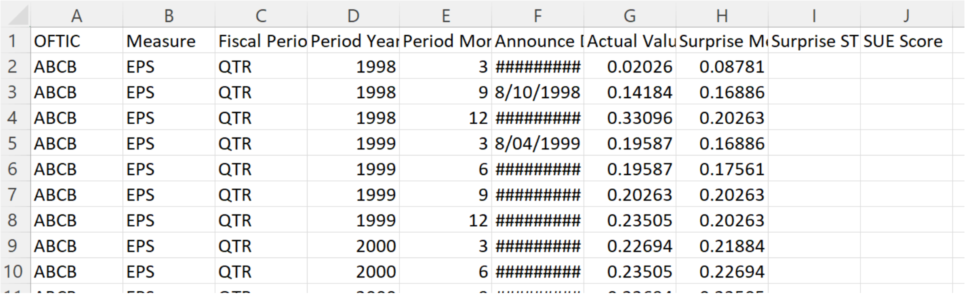
Data Cleaning and Wrangling

Like many data science projects, once the data was in Python it still needed significant modification to be used for analytical purposes. A major contributor to this is that the project was about comparing the effects of the S.U.E score with the price change, however, these two metrics were in separate data frames. To clarify, price data was stored as “PX_LAST” in the price data frames shown in Figure 5 and the S.U.E score was stored as “S.U.E Score” in the earning report data frames shown in Figure 6. The best solution found for bringing these two variables together was migrating the price data for each earning report into the earning reports data frame by referencing the price data in the prices data frame. This would prove to be more intuitive than alternative techniques involving merging or joining the data frames.



	A	B	C	D	E	F	G	H	I	J	K	L
1	Start Date	5/07/2000										
2	End Date											
3												
4		AUB UW Equity			BKU UN Equity			FFIN UW Equity			FULT UW Equity	
5		#NAME?	#NAME?	#NAME?	#NAME?	#NAME?	#NAME?	#NAME?	#NAME?	#NAME?	#NAME?	#NAME
6	Dates	PX_LAST	BEST_EPS	IS_DIL_EPS	PX_LAST	BEST_EPS	IS_DIL_EPS	PX_LAST	BEST_EPS	IS_DIL_EPS	PX_LAST	BEST_EP
7	#NAME?	7.0833	#N/A N/A	0.1667	#NAME?	#N/A N/A	#N/A N/A	#NAME?	#N/A N/A	0.0568	#NAME?	#N/A N/
8	6/07/2000	7	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.18	#N/A N/A	0.0568	11.2546	#N/A N/
9	7/07/2000	7	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.22	#N/A N/A	0.0568	11.09	#N/A N/
10	10/07/2000	7	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.1938	#N/A N/A	0.0568	10.9584	#N/A N/
11	11/07/2000	7	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.18	#N/A N/A	0.0568	10.3331	#N/A N/
12	12/07/2000	6.9167	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.21	#N/A N/A	0.0568	10.8597	#N/A N/
13	13/07/2000	7	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.24	#N/A N/A	0.0568	10.7939	#N/A N/
14	14/07/2000	7.3333	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.28	#N/A N/A	0.0568	10.9255	#N/A N/
15	17/07/2000	7.4167	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.27	#N/A N/A	0.0568	10.6951	#N/A N/
16	18/07/2000	7.4167	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.22	#N/A N/A	0.0568	10.9255	#N/A N/
17	19/07/2000	7.25	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.19	#N/A N/A	0.0568	10.9255	#N/A N/
18	20/07/2000	7.3333	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.28	#N/A N/A	0.0568	11.0571	#N/A N/
19	21/07/2000	7.3333	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.215	#N/A N/A	0.0568	10.9913	#N/A N/
20	24/07/2000	7.3333	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.2	#N/A N/A	0.0568	10.9255	#N/A N/
21	25/07/2000	7.3333	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.24	#N/A N/A	0.0568	11.1229	#N/A N/
22	26/07/2000	6.6667	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.24	#N/A N/A	0.0568	11.5178	#N/A N/
23	27/07/2000	7.0833	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.22	#N/A N/A	0.0568	11.3204	#N/A N/
24	28/07/2000	7	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.285	#N/A N/A	0.0568	10.8926	#N/A N/
25	31/07/2000	7	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.32	#N/A N/A	0.0568	11.3204	#N/A N/
26	1/08/2000	7	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.3	#N/A N/A	0.0568	10.9913	#N/A N/
27	2/08/2000	7	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.32	#N/A N/A	0.0568	11.0242	#N/A N/
28	3/08/2000	7	#N/A N/A	0.1667	#N/A N/A	#N/A N/A	#N/A N/A	2.31	#N/A N/A	0.0568	11.4191	#N/A N/

Figure 5: Raw Price Data



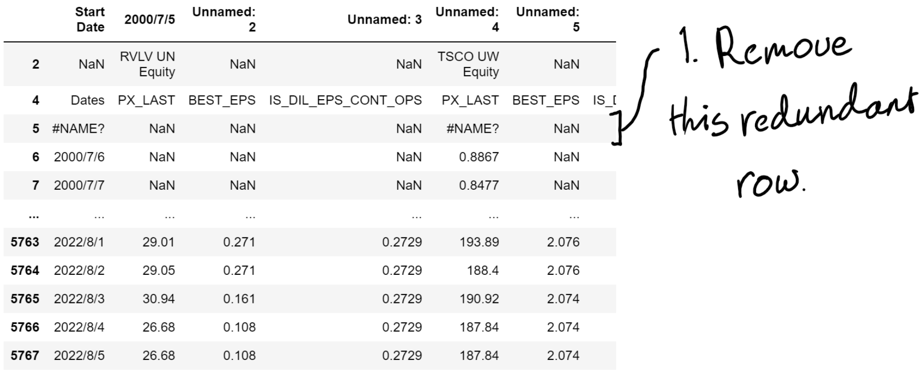
	A	B	C	D	E	F	G	H	I	J
1	OFTIC	Measure	Fiscal Perio	Period Year	Period Mor	Announce [	Actual Valu	Surprise M	Surprise ST	SUE Score
2	ABCB	EPS	QTR	1998	3	#########	0.02026	0.08781		
3	ABCB	EPS	QTR	1998	9	8/10/1998	0.14184	0.16886		
4	ABCB	EPS	QTR	1998	12	#########	0.33096	0.20263		
5	ABCB	EPS	QTR	1999	3	8/04/1999	0.19587	0.16886		
6	ABCB	EPS	QTR	1999	6	#########	0.19587	0.17561		
7	ABCB	EPS	QTR	1999	9	#########	0.20263	0.20263		
8	ABCB	EPS	QTR	1999	12	#########	0.23505	0.20263		
9	ABCB	EPS	QTR	2000	3	#########	0.22694	0.21884		
10	ABCB	EPS	QTR	2000	6	#########	0.23505	0.22694		

Figure 6: Raw Earning Reports data

Cleaning the Prices Data Frames

As we plan to reference prices from the prices data frame and insert them into the earning reports data frame, we need to be able to easily call a company's stock price on a given day. This being said figure 5 shows that this was not necessarily the case with the raw data. As a result, it needed to undergo some data cleaning.

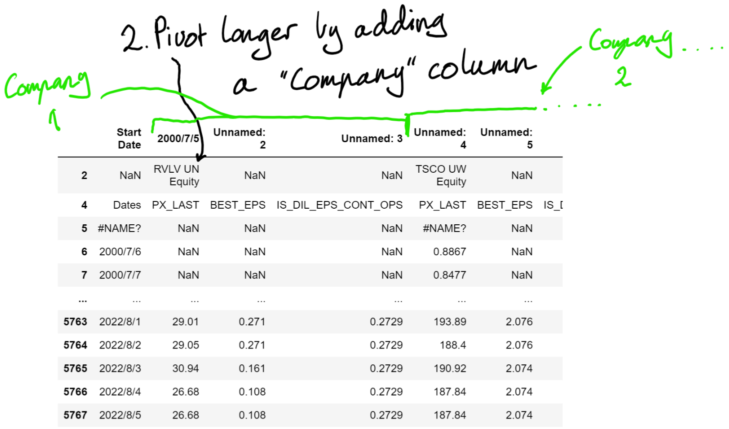
The first thing done to the dataset was removing redundant rows. These came in the form of rows containing nothing other than name errors and Nan's as shown in figure 7.



	Start Date	2000/7/5	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5	
2	NaN	RVLV UN Equity	NaN	NaN	TSCO UW Equity	NaN	
4	Dates	PX_LAST	BEST_EPS	IS_DIL_EPS_CONT_OPS	PX_LAST	BEST_EPS	IS_[
5	#NAME?	NaN	NaN	NaN	#NAME?	NaN	
6	2000/7/6	NaN	NaN	NaN	0.8867	NaN	
7	2000/7/7	NaN	NaN	NaN	0.8477	NaN	
...	...	...	...	...	...	...	
5763	2022/8/1	29.01	0.271	0.2729	193.89	2.076	
5764	2022/8/2	29.05	0.271	0.2729	188.4	2.076	
5765	2022/8/3	30.94	0.161	0.2729	190.92	2.074	
5766	2022/8/4	26.68	0.108	0.2729	187.84	2.074	
5767	2022/8/5	26.68	0.108	0.2729	187.84	2.074	

Figure 7: Redundant rows in price data

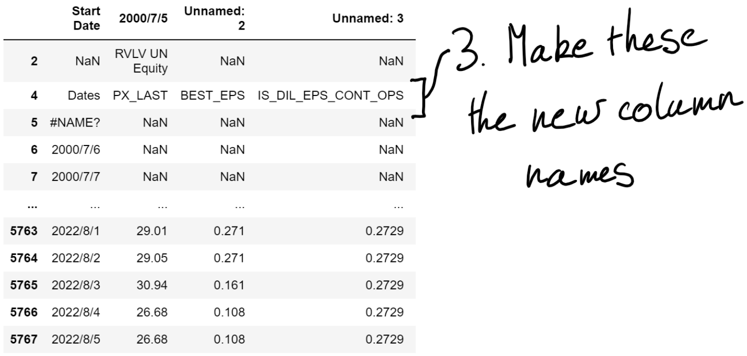
The second thing was then pivoting the data into a longer format. As shown in Figure 8 the data is currently arranged such that left to right the PX_LAST, BEST_EPS and IS_DIL_EPS_CONT_OPS variables repeat for each company. Intuitively this makes it more difficult to call a specific company stock price in comparison to if PX_LAST, BEST_EPS and IS_DIL_EPS_CONT_OPS were column variables and then there was another column variable for the company that the data was referring to.



	Start Date	2000/7/5	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5	
2	NaN	RVLV UN Equity	NaN	NaN	TSCO UW Equity	NaN	
4	Dates	PX_LAST	BEST_EPS	IS_DIL_EPS_CONT_OPS	PX_LAST	BEST_EPS	IS_[
5	#NAME?	NaN	NaN	NaN	#NAME?	NaN	
6	2000/7/6	NaN	NaN	NaN	0.8867	NaN	
7	2000/7/7	NaN	NaN	NaN	0.8477	NaN	
...	...	...	...	...	...	...	
5763	2022/8/1	29.01	0.271	0.2729	193.89	2.076	
5764	2022/8/2	29.05	0.271	0.2729	188.4	2.076	
5765	2022/8/3	30.94	0.161	0.2729	190.92	2.074	
5766	2022/8/4	26.68	0.108	0.2729	187.84	2.074	
5767	2022/8/5	26.68	0.108	0.2729	187.84	2.074	

Figure 8: Pivoting the data into a longer format

The last thing done was making PX_LAST, BEST_EPS and IS_DIL_EPS_CONT_OPS all variable names so that we could easily identify and call the price column of the frame.

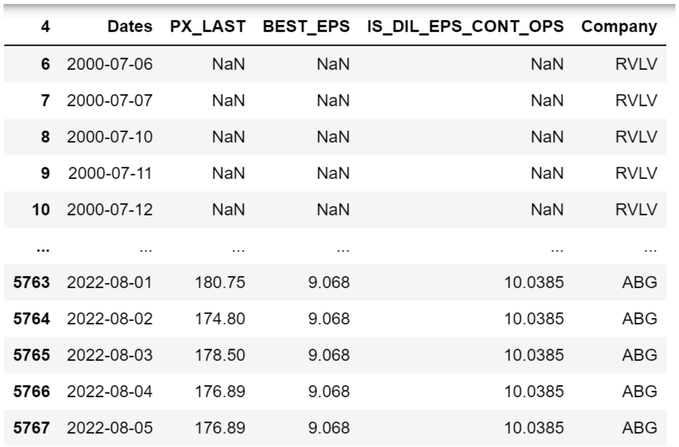


	Start Date	2000/7/5	Unnamed: 2	Unnamed: 3
2	NaN	RVLV UN Equity	NaN	NaN
4	Dates	PX_LAST	BEST_EPS	IS_DIL_EPS_CONT_OPS
5	#NAME?	NaN	NaN	NaN
6	2000/7/6	NaN	NaN	NaN
7	2000/7/7	NaN	NaN	NaN
...	...	...	...	...
5763	2022/8/1	29.01	0.271	0.2729
5764	2022/8/2	29.05	0.271	0.2729
5765	2022/8/3	30.94	0.161	0.2729
5766	2022/8/4	26.68	0.108	0.2729
5767	2022/8/5	26.68	0.108	0.2729

3. Make these the new column names

Figure 9: Creating appropriate column names

After all of this cleaning, we ended the result shown in Figure 10. This data frame made it a lot easier to find specific company stock prices on specific days and thus made the process of migrating price data into the earning reports data frame a lot easier.



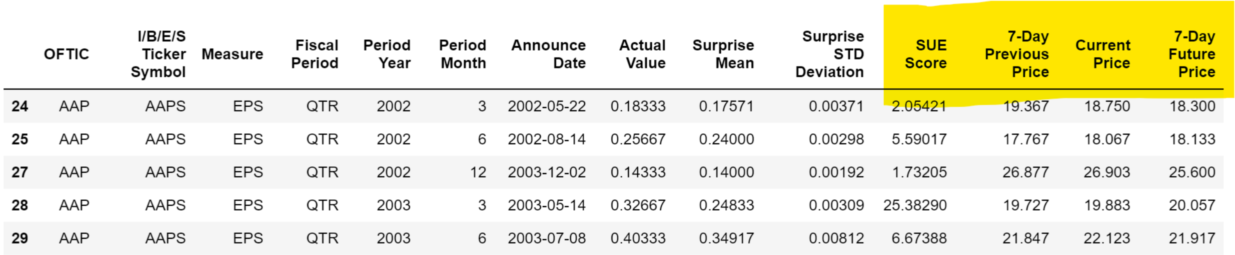
4	Dates	PX_LAST	BEST_EPS	IS_DIL_EPS_CONT_OPS	Company
6	2000-07-06	NaN	NaN	NaN	RVLV
7	2000-07-07	NaN	NaN	NaN	RVLV
8	2000-07-10	NaN	NaN	NaN	RVLV
9	2000-07-11	NaN	NaN	NaN	RVLV
10	2000-07-12	NaN	NaN	NaN	RVLV
...	...	...	...	...	...
5763	2022-08-01	180.75	9.068	10.0385	ABG
5764	2022-08-02	174.80	9.068	10.0385	ABG
5765	2022-08-03	178.50	9.068	10.0385	ABG
5766	2022-08-04	176.89	9.068	10.0385	ABG
5767	2022-08-05	176.89	9.068	10.0385	ABG

Figure 10: The final prices data frame

It is worth noting that we didn’t have to deal with the NaNs as most of them represented the period when the company in question did not exist. As non-existent companies don’t release earning reports we wouldn’t have the issue of NaNs making it into our analysis as the earning reports data frame would only require price data around the date of the earning report's release.

Migrating Price Data

Now that calling specific stock prices was easy, migrating price data into the earning reports data frame became a relatively simple task. All that was involved was iterating over each observation in the earning report data frame, subsetting the prices data frame by the company that released the earning report and then pulling the price on the day of the earning report’s release. This along with the price 7 days prior and 7 days in the future then populated three new variables in the dataset with the earning reports as shown in Figure 11.



	OFTIC	I/B/E/S Ticker Symbol	Measure	Fiscal Period	Period Year	Period Month	Announce Date	Actual Value	Surprise Mean	Surprise STD Deviation	SUE Score	7-Day Previous Price	Current Price	7-Day Future Price
24	AAP	AAPS	EPS	QTR	2002	3	2002-05-22	0.18333	0.17571	0.00371	2.05421	19.367	18.750	18.300
25	AAP	AAPS	EPS	QTR	2002	6	2002-08-14	0.25667	0.24000	0.00298	5.59017	17.767	18.067	18.133
27	AAP	AAPS	EPS	QTR	2002	12	2003-12-02	0.14333	0.14000	0.00192	1.73205	26.877	26.903	25.600
28	AAP	AAPS	EPS	QTR	2003	3	2003-05-14	0.32667	0.24833	0.00309	25.38290	19.727	19.883	20.057
29	AAP	AAPS	EPS	QTR	2003	6	2003-07-08	0.40333	0.34917	0.00812	6.67388	21.847	22.123	21.917

Figure 11: New variables in earning reports data set

From here some more variables were derived from these statistics and then data analysis could begin. The new variables were:

- Price Change: The future price minus the current price
- Gradient Change: Price change minus the difference between the current price and the 7-day previous price
- Price Change Sign: 1 if the price change was positive, -1 if the price change was negative and 0 if the price change was 0.
- S.U.E Score Sign: Same as price change variable but for S.U.E score

Exploratory Data Analysis

Before beginning any modelling we had a look around with our new data frames in order to get familiar with the data they contained. This involved performing simple analytical processes such as investigating the shape of the data and looking at correlations amongst other things.

The shape of the Data

To reiterate, the hypothesis we are investigating is that a positive earnings surprise leads to an increase in stock price and a negative earning surprise leads to a drop in stock price. This hypothesis details a relationship that may be visible if S.U.E scores and Price change were plotted against each other. As a result, the S.U.E score was plotted against price change to produce the scatter plot in Figure 12.

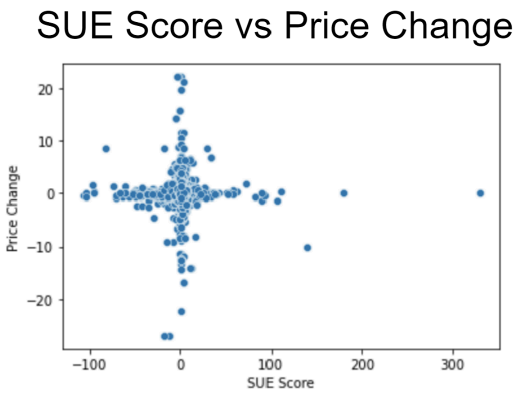


Figure 12: Sue Score vs Price Change for all data

If the hypothesis was true what we'd expect to see is most of the data points lying in quadrants 1 and 3 as this is where the S.U.E score and price change have the same sign. An example of what this would look like is shown in Figure 15. The scatter plot in Figure xx fails to show this, however, this relationship may have been hidden behind the large concentration of data points concentrated around the origin. Therefore, this scatter plot isn't that informative and we need to move on to more sophisticated techniques.

Linear Correlations

A way of circumventing the issues that arise from looking for trends in busy data is to look at the linear correlations in the data. This is because a linear correlation statistic is simple and easy to read regardless of how much overlapping data there is. As a result of this, we investigated the linear correlation between the S.U.E score and Price change through the following heatmap shown in Figure 13.

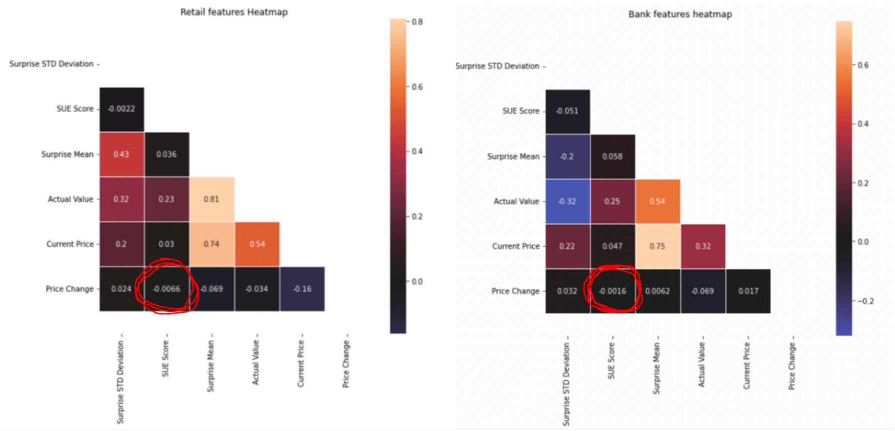


Figure 13: Linear Correlations between S.U.E score and Price Change

If the hypothesis was true we would expect to find at least moderate positive values around the tune of 0.2. Figure 13 shows that this was not the case as the linear correlation coefficients in the banking and retail sections were -0.0016 and -0.0066 respectively. This indicates that basic methods cannot indicate any significant linear correlation between the two variables and to learn more about these variables' linear correlation we need to use more advanced techniques such as linear regression.

Variable Changes Overtime

The project team was advised in the beginning stages of the project that a real trader had found that the hypothesis seemed to hold true previously, however, in recent years its apparent legitimacy declined. As a result, part of the exploratory analysis involved investigating the variance in price change over time. One way we did this was by graphing price changes over time as shown in figure 14.

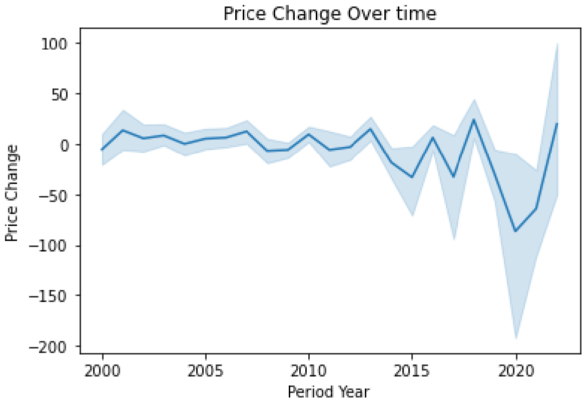


Figure 14: Graph of Price change over time.

In Figure 14 the opaque blue line represents the mean price change and the height of the transparent blue represents the variance in a price change. The variance for price change before 2010 was 0.78 and the variance after 2010 was 21.61 as shown in the graph. Although not outright proving this does suggest that the theory that the hypothesis has become less true over time has some validity. This is because whatever is causing this increase in variance may have also resulted in a decline in the hypothesis' effectiveness as a stock trading strategy.

Linear Regression Modelling

As mentioned in the exploratory analysis, if the hypothesis was true it is expected that the data would follow a similar shape to the sketch in Figure 15. Although, this sketch doesn't strictly show a linear relationship between SUE score and price change it indicates that a linear model would at least have some accuracy in predicting price change from SUE score.

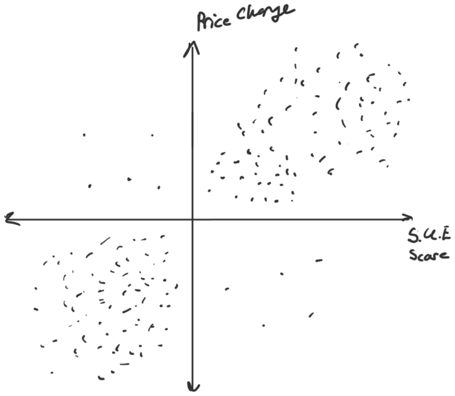
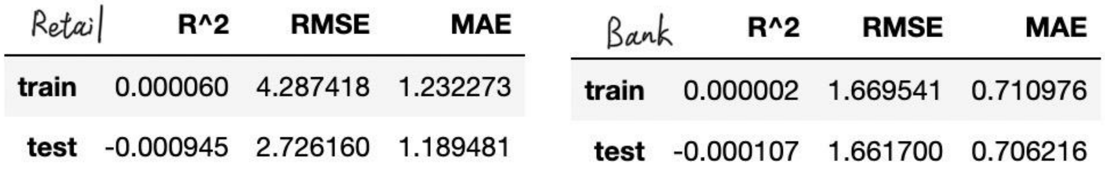


Figure 15: Price Change vs. SUE score if the hypothesis was true

Due to the subtle linearity in the hypothesis, linear regression modelling was experimented with to find a connection between the two variables. To do this both the retail and bank datasets were split 80:20 into training and testing sets and modelled using the sklearn Linear Regression model. The resultant error metrics from these models are shown in Figure 16. As shown the models performed very poorly with R-squared scores within 0.001 on either side of zero. These incredibly low R-squared scores indicate that within a linear model SUE score was unable to explain much, if any, of the variance in price change.

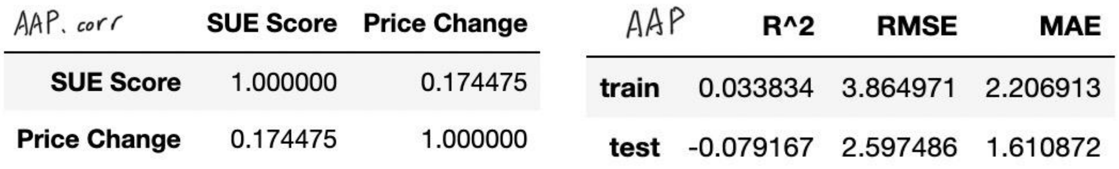


Retail	R^2	RMSE	MAE
train	0.000060	4.287418	1.232273
test	-0.000945	2.726160	1.189481

Bank	R^2	RMSE	MAE
train	0.000002	1.669541	0.710976
test	-0.000107	1.661700	0.706216

Figure 16: Error metrics for SUE score vs Price change regression

As shown the models performed very poorly with R-squared scores of 0.000002 and 0.0006. These incredibly low R-squared scores indicate that within a linear model, the SUE score was unable to explain much, if any, of the variance in a price change. Further investigation was then conducted by looking at linear regression modelling for specific companies. One such company was Advance Auto Parts Inc. (AAP) and the correlation table and linear modelling error metrics for are shown in Figure 17.



AAP. corr	SUE Score	Price Change
SUE Score	1.000000	0.174475
Price Change	0.174475	1.000000

AAP	R^2	RMSE	MAE
train	0.033834	3.864971	2.206913
test	-0.079167	2.597486	1.610872

Figure 17: AAP correlation table and linear regression error metrics

These results were a substantial improvement over what we had received previously as they included significantly higher correlations and much higher R-squared scores. Despite this R squared was still very low, however, these results did indicate that the connection between SUE score and price change may be related to the specific company. We investigated this indication further by creating a linear model for each company and then graphing their R-squared scores as shown in Figure 18.

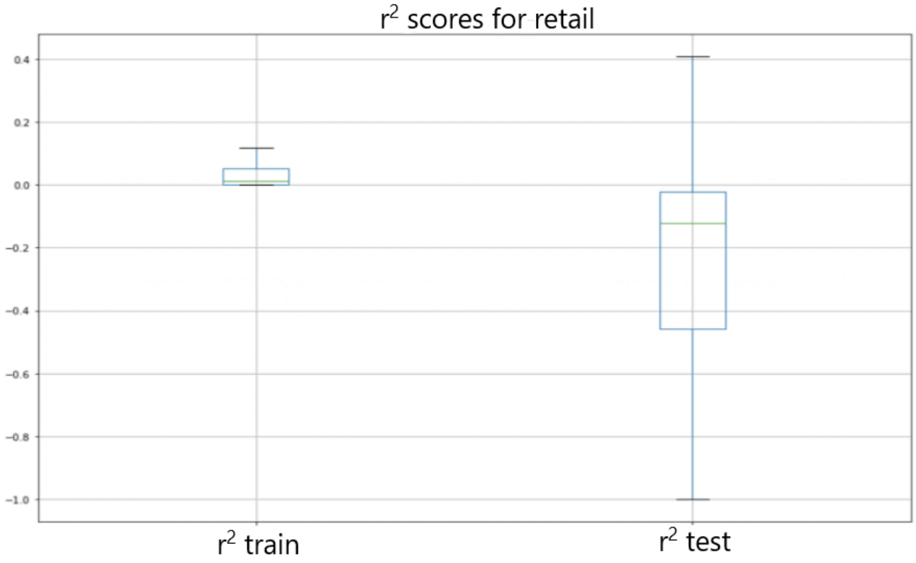


Figure 18: Distribution of linear model scores by company

As the boxplot (outliers removed) shows, the mean training score is very close to zero, and most (over 75%) testing scores are lower than zero. This meant that to most retailers, the linear models generated made worse predictions than just using the mean in most cases. As a result, the tentative conclusion was drawn that there is likely to be no linear relationship between SUE Score and Price Change.

kNN Modelling

Although the results from experimenting with linear regression largely pointed towards the hypothesis being false, research into other modelling techniques began. This was in the hopes that other modelling techniques could provide more insight into the relationship between S.U.E Score and price change.

Due to there being significant evidence against a linear relationship between stock price change and S.U.E score, kNN modelling was investigated due to its effectiveness when working with non-linear relationships. The kNN model works by comparing a data point to its “neighbours” and then classifying it by the majority class closest to the unclassified data point. It is useful when the data is non-linear and the points are well-defined. Some of the hyperparameters that needed to be tuned were the k-value, which decides how many neighbours the model considers, as well as the training and testing split, which is often around 80:20.

kNN Regressor

The first of these investigations was the creation of a kNN regressor that would attempt to predict price change from the S.U.E score. Creating this regressor required some preparation. Firstly, extreme outliers have the potential to confuse kNN regressors and as a result, they were removed from the training and testing data as our chance of predicting extreme outliers was unlikely anyway. This removal is shown in Figure 19 where the dots circled in red represent some of the extreme outliers that were removed.

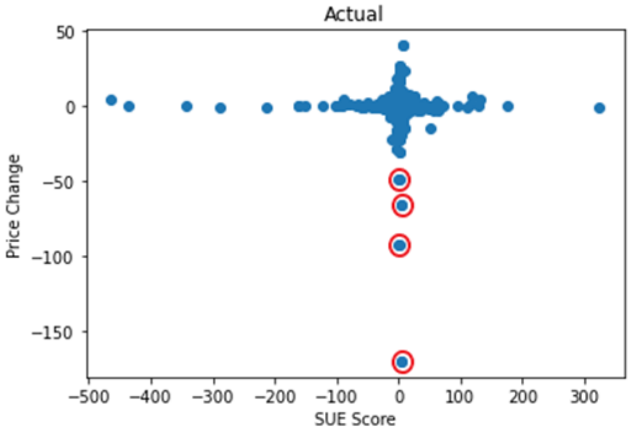


Figure 19: Outliers to be removed before kNN regressor creation

After this, the optimal number of neighbours to use for the regressor was then investigated by creating knn regressors for each number of neighbours and recording their mean squared error as shown by the line plot in Figure 20. The optimal number of neighbours was found to be around 100 neighbours.

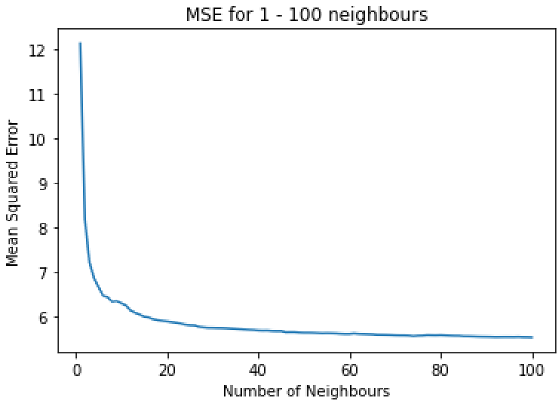


Figure 20: Comparison of the number of neighbours vs MSE for kNN regressor models

From here the final kNN regressor model was created using a 70:30 training to test split and the previously found optimal number of neighbours. The predictions this model made are compared to the actual spread of data in Figure 21

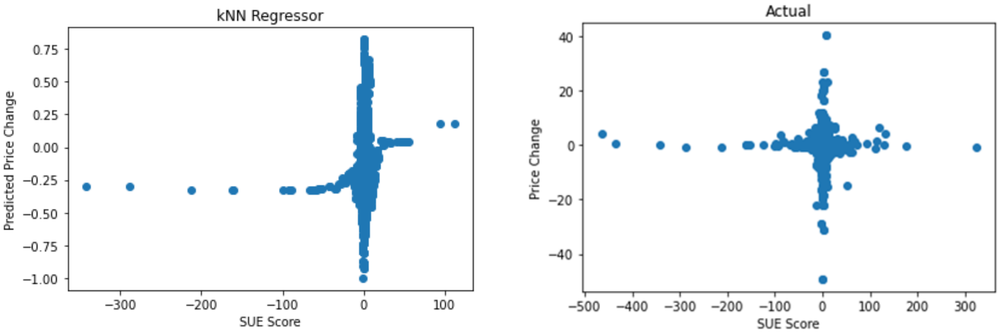


Figure 21: Comparison between kNN regressor predictions and actual data spread

This comparison in conjunction with the low mean squared score illustrated that whilst the relationship between S.U.E Score and price change might not be as linear and straightforward as proposed in the hypothesis, there may still be a reliable way to predict changes in stock price from the S.U.E score. This being said, even after removing the extreme outliers, there was still a high amount of variance at S.U.E Score = 0 which caused the model to struggle in predicting Price change when the S.U.E score was near that value.

kNN Classifier:

After the kNN regressor hinted at a more complex relationship between S.U.E score and price change this relationship was explored through a more simple approach. This involved utilising a kNN classifier model that predicts a categorical variable, rather than attempting the more complicated task of predicting a continuous response variable with the regressor.

For this kNN classifier, two variables were chosen. The independent variable was the S.U.E. score, and the dependent variable was the Price Change Sign. Price Change Sign was the result of categorising the Price Change variable, in order to work with the classifier modelling technique. Price Change Sign had 3 different states, “Positive”, “Negative”, and “No Change”, which indicated the direction of the price change between two dates.

A kNN classifier with three neighbours was created with an 80:20 training testing split and its confusion matrix is shown in figure 22. The confusion matrix is a 3x3, which represents the 3 classes that could be potentially classified. The diagonal of the confusion matrix should have the largest values, in order to indicate a good model. This is the case with this classifier, as the majority of the values are on this diagonal.

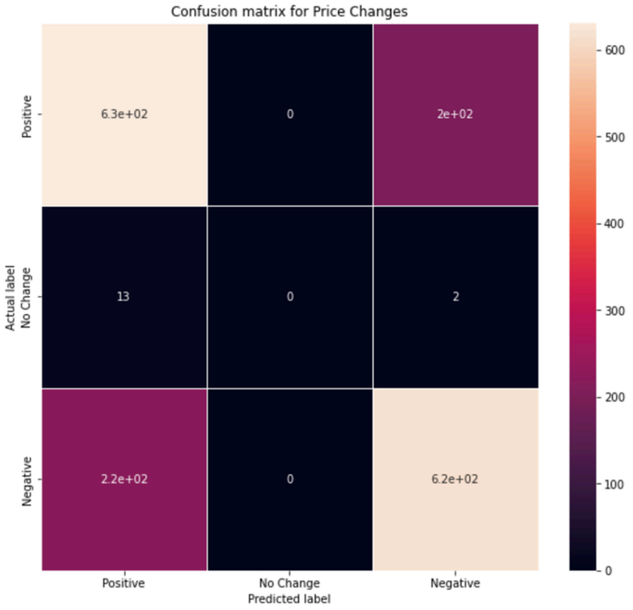


Figure 22: kNN classification model's confusion matrix

When looking at the result metrics of this kNN classifier, the model had an overall accuracy of 74%. Other numerical results for the recall and precision scores are outlined in Figure 23.

Category	Recall Score	Precision Score	F1-Score
Positive	0.73	0.76	0.74
No Change	0	0	0
Negative	0.75	0.74	0.74

Figure 23: Result metrics for kNN Classifier

The “No Change” category had the value 0 for all metrics since the model was unable to predict anything in this class. This was a consequence of the imbalance between the “Positive” and “Negative” classes, compared to the “No Change” class. To mitigate this, the additional parameter of “weights = ‘uniform’” was utilised, however, this imbalance was too great to be avoided. The recall and precision scores for the “Positive” and “Negative” classes were moderately strong, as well as the f1-scores. As the f1-score is the harmonic mean of the precision and recall score it can be used as a simple metric that combines both precision and recall. The value for the F1-score for both “Positive” and “Negative” was 0.74, which indicated that the k-NN model was moderately accurate overall when predicting these two classes. Some potential improvements for the future would be to make multiple k-NN classifiers in different periods, to assess the predictability of stock price changes in different periods, and to assess the validity of the hypothesis in the past.

Conclusions

Addressing the Hypothesis

After all the research conducted on the topic, the project had an inconclusive stance on the hypothesis. This result is largely due to the duality in findings. Firstly, the hypothesis infers that there is at least some linear relationship between SUE score and price change, however, there were no findings to support this. The low model R squared scores, near zero correlations and incoherent shape of the data all suggest that there was no linear relationship between SUE score and price change. Therefore, we can’t say that the hypothesis is true, however, due to the success of the kNN classifier’s high accuracy scores we also can’t say it’s false. The kNN classifier indicated that there may be some predictable way to forecast price change using SUE score thus indicating that there may be some validity to the hypothesis. Furthermore, the kNN classifier couldn’t be used as outright proof for the hypothesis as the relationship it proved wasn’t necessarily the relationship outlined in the hypothesis. Resultantly, the chance of it being spurious was too high for it to be considered a rigorous argument.

Overall, this result makes sense as financial markets are very complex, making them unpredictable. Firstly, company stock prices and the stock market as a whole can be affected by world events such as war and civil unrest, natural disasters and disease. These effects often occur in a chain reaction. Secondly, inflation cannot be avoided. To combat inflation, the Federal Reserve frequently modifies interest rates, which is one of the more predictable consequences of the stock market. Historically, low inflation has strongly correlated negatively with valuations. Third, a company's stocks frequently follow the market and their sector or other related industries. Much of a stock’s movement is influenced by the interaction of broader market and industry movements. Finally, changes in exchange rates can raise or lower the cost of doing business in a nation, which can have an impact on the stock prices of companies conducting business overseas.

Summary of Other Findings

In completing the analysis of the project some other findings were:

- There seemed to be no difference between the two sectors (banking and retail) as they both had equally low performance in linear testing.
- There has been an increase in the variance of price changes which may be the cause of the perceived decline in the hypothesis’ validity.

Areas for further research

A way to conduct more advanced research on this topic would be to look at it on a more case-by-case basis. This may look like investigating specific companies or periods to look for patterns in correlation. As explained, financial markets are very complex and therefore, looking at the problem through a more specific lens may provide insight into how exogenous factors impact the hypothesis.

References:

[1] CFI Team. (2021, January 5). Unexpected Earnings. Corporate Finance Institute. https://corporatefinanceinstitute.com/resources/knowledge/accounting/unexpected-earnings/

[2] Corporate Finance Institute. (2019, August 24). Unexpected Earnings. Retrieved October 21, 2022, from Corporate Finance Institute website: https://corporatefinanceinstitute.com/resources/knowledge/accounting/unexpected-earnings/

[3] Dwivedi, R. (2020, April 23). How Does K-nearest Neighbor Work in Machine Learning Classification Problem? AnalyticSteps. https://www.analyticssteps.com/blogs/how-does-k-nearest-neighbor-works-machine-learning-classification-problem

[4] Earnings Per Share (EPS): What It Means and How to Calculate It. (2022). Retrieved October 21, 2022, from Investopedia website: https://www.investopedia.com/terms/e/eps.asp

[5] Gavin, B. (2018, October 26). What is an XLSX File (and How Do I Open One)? Retrieved October 21, 2022, from How-To Geek website: https://www.howtogeek.com/392333/what-is-an-xlsx-file-and-how-do-i-open-one/

[6] Publicly Traded Company: Definition, How It Works, and Examples. (2022). Retrieved October 21, 2022, from Investopedia website: https://www.investopedia.com/terms/p/publiccompany.asp