# Related work (Peter, Vijay, Andrew, Harrison)

We should just copy from original report. We don't need to update this

# Methodology (Peter, Vijay, Andrew, Harrison)

## Datasets, tools, main tasks, analytical thinking (Peter, Vijay, Andrew, Harrison)

**For each tool used (Peter, Vijay, Andrew, Harrison)**

We are using cross correlation to best figure out correlations between companies and their potential partnerships. There are some alternatives, like mutual information or dynamic time warping. We chose cross correlation because it was not computationally expensive for large datasets, and it was simpler.

**For each main task (Peter, Andrew, Harrison)**

Our main task was to try to find out whether two companies were correlated and had potential partnerships. While trying to solve this task we computed a correlation analysis. This was not the silver bullet we were looking for. It helped us to find companies that might have those partnerships, but it also highlighted competitors and those companies that had really strong runs recently. To find those partnerships we would need to add more models. We tried grouping more companies like groups of three. Which did help a little bit. We were able to find a correlation between McDonald’s and Coke’s stock prices. The only issue is we do not know if this is due to the current partnership between the two or whether it is because both companies sell junk food/drinks. Pepsi also has a high correlation with Mcdonalds so it could just be the latter.

This was done by first taking all the tags passed in and computing a full correlation analysis with the percentage stock data for each tag. This correlation analysis uses the numpy correlate function. We then convert the correlation data to its z score. We also map lag values onto the various correlation values. This was how we got the correlation value for two companies. When we went to three companies, we took every combination of two companies that we could get out of those three and averaged their max correlation values.

**Analytical thinking (Peter, Andrew, Harrison)**

In our correlation analysis, we expected to find two patterns. Which companies were correlated and how they potentially affected each other. The correlation analysis gave us groups of 2-3 companies that are correlated. We can then take this information and expand on it. This data only narrowed down the places for us to look.

# Data understanding, preprocessing, warehouse, modeling (Peter, Vijay, Andrew, Harrison)

**Data understanding (Peter, Andrew, Harrison)**

We started off with an exploratory analysis of the data. We created a volatility measure that found the standard deviation of the data. We then graphed it for all the companies. We also had a graph that showed all the correlation values for each company and the lag values those corresponded to. These helped us better understand the data we were working with and to make better models in the long run.

**Preprocessing (Peter, Andrew, Harrison)**

In the cross correlation analysis, we had to clean the data that was returned from the yfinance api. It returned data for Open, High, Low, Close, Adj Close, and Volume and we thought it best to only take the closing stock data for our analysis. If there were missing values, then we returned an error message because the api returned junk data. The stock data was then transformed into percentage change data. This allowed us to compare the stocks on the same playing field, since not all stocks have the same number of shares or the same price scale. We then through it through the numpy correlation function and then did z score normalization on that data. This was done to make it easier to compare correlation values.

# Evaluation (Peter, Andrew, Harrison)

**Metrics**

We tested our cross-correlation model by finding companies we would expect to be highly correlated and seeing whether they actually were correlated. This was how we figured out that data from the last five years is significantly better than last year for drawing conclusions. Our model is accurate but considering that we only have data from the past 5 years, it is not perfect.

**Results (figures, tables w/ proper labels)**

* What outputs did we get from the models?
  + The cross-correlation analysis outputted a table in the database that contained every combination of the input tags, the max correlation and at what lag value that correlation was at and the avg of the 3 largest correlation values. It also created some graphs to better display this information.
* What input did we give to the model?
  + The cross-correlation analysis pulled stock data from the finance api using tags given to it by the user.

**Interpretation**

With our correlation analysis, we were able to learn that the data can be used to find the most correlated stocks and use it in tandem with other models to try to find some sort of causation or pattern. If there is one.

Our findings contained some expected and unexpected values. From the cross correlation, we found some findings that were expected like Coca-cola and Pepsi or like Nvidia and Amd. We also found some unexpected results like Mcdonalds and Pepsi or Apple and Tesla were highly correlated.

# Discussions (Peter, Vijay, Andrew, Harrison)

**Lessons learned**

Learning about cross correlation analysis was interesting. It is a powerful tool that allows for analyzing whether different data sets are correlated or not. I thought it was especially interesting to find out about lag. In the context of stocks, it is very valuable to find out if one stock is doing good and how it is going to affect others in the future.

When working on the cross-correlation analysis, we got to experience the data mining principle of statistical significance. For instance, for the last year Tesla and Johnson and Johnson were the most correlated in their stocks. If we take the last five years into account, Johnson and Johnson is the least correlated company with Tesla. This shows that we can’t always take data at its face value and we always have to do some digging.

**What worked well/what didn’t**

We thought the correlation analysis worked pretty well. It allowed us to group companies that performed similarly. This could potentially allow us to use that data to make predictions on whether a stock might go up based on how its group is doing.

A part of the correlation analysis that went particularly badly was when we took an average of all the correlations at each lag point for two companies. I figured that it would give me a reasonable value that could lead me to believe that two companies were correlated. Turns out all average correlations were basically 0. We ended up only averaging the top 3 correlation values, but I don’t believe that it shows anything. Considering the fact that the second largest average is Nvidia and Visa and they have a fairly low max correlation.

**Directions for future work**

If we had another month on the project, we would like to implement a full apriori algorithm for grouping companies based on max correlation values. This would allow us to group companies in groups of more than 3. We would also try some other alternatives to cross correlation. Maybe one that is better at catching nonlinear patterns.

# Appendix (Peter, Vijay, Andrew, Harrison)

Andrew: I wrote all the code in the file companyCorrelator.ipynb. I helped the group to make the various versions of slides and the reports. I added the tables CrossCorrelationResults and Volatility to the database and generated all the data that went along with those. I wrote the code that computed the cross correlation analysis, volatility, and found groups of three for 25 different companies. I also did some analysis like finding mean, standard deviation, and quartiles on the data stored in the CrossCorrelationResults table. I also made a bunch of different graphs for the data. I also wrote a readme for companyCorrelator.ipynb.