In this research project, the objective is to study if the State of the Union Address (SOTU) each year has a significant impact on the performance of the US stock market and some of its portfolios. Moreover, the sentimental preference of each address will be analyzed using sentiment analysis to see whether the sentimental preference has a strong impact on the stock returns.

There are **three main parts** done by the codes:

1. The first part – **CAPM Analysis (SOTU vs. non-SOTU).py**
2. The second part – **Sentiment Analysis.py**

(There is a Jupyter Notebook version – **Sentiment Analysis.ipynb** because it may be clearer to see the steps of training and testing neural networks on Jupyter Notebook)

1. The third part – **CAPM Analysis (Effect of Sentiment Preference).py**

(Running this part of codes might take a long time.)

The first part and the third part are similar and have used some same or similar functions. Hence, they are combined in the file - **CAPM Analysis (Combined).py**. As a result, one can run the second part first, and then directly run the combined codes.

Furthermore, a Jupyter Notebook of the combined version of part 1 and part 3 for interactive plotting – **CAPM Analysis Combined with Interactive Plotting.ipynb** has been made for people who are only interested in some particular portfolios. This Jupyter file can also allow researchers to save the Data Frame of specific portfolios to our Result file. I have run this Jupyter Notebook with the file name “Data\_Daily\_1.1-2.4.xlsx” and sheet name “2.1.”. The files “2.1.(Effect of Sentiment Preference).xlsx” and “2.1.(SOTU vs. non-SOTU).xlsx” are saved. You can change to what you are interested using the dropdown box.

**In the first part**, we use the daily stock return data which has been also divided into different portfolios according to different characteristics, such as market size, book-to-market values, and operating profitability. Then we separate these data into returns on days with a SOTU and without a SOTU. Using the Capital asset pricing model (CAPM) that  as well as linear regression, we are able get the values of , , and of different portfolios. By comparing these values on and not on SOTU days, we can get insights on how the SOTU affects the portfolios return and more specifically, how it affects , and . The results are plotted in bar plots and saved to the Result file.

From the results and graphs, we can find that the return on SOTU days is almost 10 times higher than the return on non-SOTU days. Moreover, there is more information if we look closer to each kind of portfolio. For example, for portfolios formed on sizes (see graph 2.1..png), portfolios comprised of smaller firms has higher , which is the abnormal return not captured by CAPM model with market return. On SOTU days, the is even much higher on portfolios of small firms compared with that of higher firms, indicating that the abnormal return account more for the higher returns on SOTU days. Another example is that for portfolios formed by operating profitability (see graph 2.3..png), portfolios of low profitability firms performs worse than those of high profitability on normal days without SOTU. However, on SOTU days, portfolios comprised of low profitability firms show higher expected return and high . Due to the space limit on this write up, further analysis on the result will not be displayed here, but one can always get insights from these graphs.

**The second part** is about doing sentiment analysis using the method of embedding and convolutional neural work (CNN). In this part, some packages including, nltk, re, Kaggle, genism, keras, sklearn are needed. We use the labelled movie reviews dataset to train our CNN model. The dataset has been downloaded in our Data file, but it can also be downloaded using Kaggle API with codes provided (need to uncomment). Common practices in natural language processing have been applied to clean the text data for training data, testing data, and prediction data. “word2vec-google-news-300” file has been used for embedding. Since it is a large vector file, downloading it requires some time and computer space. Therefore, executing this part of codes might take longer. However, as it has been run on my computer, the CNN details can be seen on the Jypyter Notebook provided sentiment and the sentiment result has been saved in the Data file. Hence, one can choose to skip running this part.

Next, a CNN model has been built up for sentiment prediction. The accuracy of this model is 85%. By breaking down each address into sentences, we are able to decide whether each sentence is positive or negative. A positive sentence will get a value of 1, and a negative sentence will get a 0. The final sentiment scores (between 0 to 1) of each address are the overall value of all sentences divided by the number of sentences. This result is saved to Data\sentiment.xlsx.

**In the third part**, we are going to use the sentiment scores we get from the second part in the model of , and compare the performance with our original model of . The of the linear regression will be a measurement of how well the model fits the data. Therefore, we draw the line plots of the using the two formulas above and see if adding the sentiment variable into the model will significantly increase the model performance.

According to the result we get, the of the two models are very close to each other among all portfolios. One possibility is that the sentiment preference may not be an important element to consider. However, another possibility is that the sentiment preference is not making a big difference due to the fact that the sentiment scores of all addresses are very close to each other. (Maybe it is because presidents being likely to deliver a relatively positive speech regardless of the real circumstances.)

**For future improvements:**

In this research, the model used in the first and third part is the basic CAPM model which only incorporates the excess market return as a factor. However, this model might be too simple. We will try to apply Fama-French 3-factor model, Fama-French 5-factor model, and even other models to have a better fitting.

In the second part, we used labelled movie review texts to train the CNN model. However, it might not be very accurate in deciding the presidents’ addresses. Hence, other training datasets can be explored with more relevant contexts. Moreover, more adjustments on variables such as epochs and batch size of the CNN model can be done to achieve a higher accuracy rate.