Abstract to be written here. The abstract should not be too long and should provide the reader with a good understanding what you are writing about. Academic papers are not like novels where you keep the reader in suspense. To be effective in getting others to read your paper, be as open and concise about your findings here as possible. Ideally, upon reading your abstract, the reader should feel he / she must read your paper in entirety.

Table of Contents

# Introduction

References are to be made as follows: [Fama & French](#ref-fama1997) ([1997: 33](#ref-fama1997)) and [Grinold & Kahn](#ref-grinold2000) ([2000](#ref-grinold2000)) Such authors could also be referenced in brackets ([Grinold & Kahn, 2000](#ref-grinold2000)) and together [Grinold & Kahn](#ref-grinold2000) ([2000](#ref-grinold2000)). Source the reference code from scholar.google.com by clicking on ``cite’’ below article name. Then select BibTeX at the bottom of the Cite window, and proceed to copy and paste this code into your ref.bib file, located in the directory’s Tex folder. Open this file in Rstudio for ease of management, else open it in your preferred Tex environment. Add and manage your article details here for simplicity - once saved, it will self-adjust in your paper.

I suggest renaming the top line after @article, as done in the template ref.bib file, to something more intuitive for you to remember. Do not change the rest of the code. Also, be mindful of the fact that bib references from google scholar may at times be incorrect. Reference Latex forums for correct bibtex notation.

To reference a section, you have to set a label using ``\label’’ in R, and then reference it in-text as e.g. referencing a later section, Section .

Writing in Rmarkdown is surprizingly easy - see [this website](https://www.rstudio.com/wp-content/uploads/2015/03/rmarkdown-reference.pdf) cheatsheet for a summary on writing Rmd writing tips.

# Data

Notice how I used the curly brackets and dash to remove the numbering of the data section.

Discussion of data should be thorough with a table of statistics and ideally a figure.

In your tempalte folder, you will find a Data and a Code folder. In order to keep your data files neat, store all of them in your Data folder. Also, I strongly suggest keeping this Rmd file for writing and executing commands, not writing out long pieces of data-wrangling. In the example below, I simply create a ggplot template for scatter plot consistency. I suggest keeping all your data in a data folder.

# Literature review

The aim of this project is to determine whether U.S Presidential speeches have predictive power over stock market movements. A positive result would be powerful and could add to the ability of traders to accurately predict stock market movements. Broadly, machine learning has been selected as the method of modelling and the S&P 500 index has been chosen as representative of the ‘stock market’.

The undertaking of this project assumes 2 important conjectures. That there is a correlation between the linguistic factors[[1]](#footnote-23) of speech employed by U.S. presidents in their speeches and U.S. stock market movements; and that the prominence of speech factors can be quantified by using machine learning algorithms. The following literature review defends the two conjectures, confirms the novelty of the project, surveys methods of data cleaning that may be applicable and to discover which machine learning methods are most appropriate for this project.

## Conjecture defence

The following section references the findings of four peer-reviewed articles in defence of the conjectures made in section 1. More evidence exists and is reviewed in later sections but is not necessary to defend the conjectures made here.

### FOMC speeches and U.S. financial market reactions

Hayo, Kutan & Neuenkirch (2008) performed a generalized autoregressive conditionally heteroscedastic (GARCH) analysis of the relationship between Federal Open Market Committee (FOMC) speeches and the U.S. financial market. The analysis was quantitative on the market side and qualitative on the speech side. They found that FOMC speeches influence trader behaviour, but that this effect is both asymmetrical (negative impacts were larger than positive impacts) and non-uniform across trader type (bond markets were affected far more than financial and forex markets). Further, more formal modes of communication have a larger impact on both returns and conditional variance and more prominent speakers have a greater impact on bond markets. Finally, they found that volatility in 3- and 6-month T-bills was reduced on the day of a speech.

It was commented that heteroskedasticity left something to be desired when assessing the effects of monetary shocks. Further, it was found that speeches alone were not sufficient to create significant effects for financial markets. It is important that news agencies propagate the news for market repercussions to occur. In a brief, informal interview with a few bond traders it was discovered that they tended to “read monetary policy statements and listen to speeches by Greenspan (Bernanke) themselves. Other types of communications are rather neglected and the traders tend to rely on newswire information” (Hayo et al., 2008:27). Further, it was noted that news articles fail to take a neutral stance on the contents of speeches implying that the market effect may be distorted by the sentiment of news agencies.

This article shows that the sentiments of communications influence the effect on markets. Thus, analysing sentiment is an important factor in an accurate appraisal of the relationship between speeches and markets. The finding that speeches need to be propagated by news agencies in order to have significant impacts on financial markets is counter to the hypothesis of this project but the FOMC is less publicly scrutinized than the U.S. president which may render this finding inconsequential to this project.

### Political speeches and stock market outcomes

Maligkris (2017) demonstrates that the speeches given by U.S. presidential candidates directly influence the stock market, particularly during the early months of their campaigns. These speeches often contain information about potential presidents’ positions on policy changes and public issues. Thus, they can affect investor sentiment and in turn the stock market. The employed methodology was to analyse transcripts of presidential candidate speeches from the American Presidency Project archives and the U.S. Government Publishing Project from the 2004-2016 period according to the index developed in Baker, Bloom & Davis (2016) (explained in section 2.3). He shows, using regression analysis that there is an increase in excess market returns of 26 basis points following candidate speeches, however the direction and magnitude of this effect varies between candidates. He goes on to examine whether the difference in effect is due to heterogenous speech content. Finally, it was demonstrated that speeches laden with economic information tend to boost stock returns while also reducing volatility. Speeches with a negative tone have the opposite effect. The long run effect of speeches is dependent on market conditions.

This paper indicates that there is a correlation between presidential candidates’ speeches and stock movements. It is then reasonable to believe that there is a correlation between presidential speeches and stock market reactions. It also highlights that tone affects the relationship and thus that the sentiment of a speech is important. Further, the archives of the American Presidency Project and the U.S. Government Publishing Project are good sources for U.S. political speech transcripts – the potential predictive data.

### Measuring economic policy uncertainty

Baker, Bloom & Davis (2016) develop an Economic Policy Uncertainty Index based on the frequency of articles containing a trio of terms in 10 leading U.S. newspapers. These terms were: ‘ “economic” or “economy” ; “uncertain” or “uncertainty”; and one or more of “congress”, “deficit”, “Federal Reserve”, ‘legislation”, “regulation”, or “White House”’ (Baker et al., 2016:1594). The terms were selected over a period of 24 months during which more than 15 000 news articles were read by humans in an auditing process. The index proved to be quite accurate, spiking near the expected events, including wars, tight presidential elections, fiscal policy battles and terrorist activity.

They went on to show that their index had a strong relationship with other economic uncertainty measures and policy uncertainty measures. Further, congruence in uncertainty prediction was found between left- and right- leaning newspapers.

This article shows that language processing can be used to predict economic events, particularly economic uncertainty and economic policy uncertainty. However, the type of language processing proposed for this project differs significantly. While Baker et al. (2016) used man hours this project will use deep learning – the result of which is likely to be greater accuracy and reduced man hour expenditure, if it is correctly applied.

### Hope, change and financial markets: can Obama’s words drive the market?

Sazedj & Tavares (2011) asked whether the speeches made by former U.S. President Barack Obama affected stock market prices. By regressing the event of a speech during Obama’s first 11 months in office on the daily excess returns of the Dow Jones, the S&P 500 and the NASDAQ they found that the event of a speech had a generally insignificant effect on daily excesses. However, by regressing key terms contained in 43 speeches given during the first 11 months of Obama’s presidency it was found that the content of speeches can significantly affect daily excess returns nearly uniformly across all three indices. Notably, the NASDAQ’s correlation to the content of speeches was weaker – indicating that technology markets may be less susceptible to presidential rhetoric.

This paper highlights the relationship between the content of a presidential speech and stock market returns. This study was correlational rather than predictive in nature.

### Conjecture defence summary

As seen in section 2.1.2. and section 2.1.4. it is true that there is a correlation between the linguistic factors of speech employed by U.S. presidents in their speeches and U.S. stock market movements (Sazedj & Tavares, 2011; Maligkris, 2017). Further, section 2.1. and section 2.3. show that sentiment and other linguistic factors of speech can be quantified using machine learning methods (Hayo, Kutan & Neuenkirch, 2008; Baker, Bloom & Davis, 2016). Thus, both conjectures hold and further investigation is warranted. This is not the total of all evidence supporting these conjectures but is sufficient. Other articles reviewed here can be seen for further evidence.

## Novelty

Searching ‘sentiment analysis stock market’ on Google Scholar yields mostly articles linking Twitter data and the stock market. Searching ‘presidential speeches affect stock market’ on Google Scholar yields articles on the relationship between presidential speeches and the stock market but none using Machine Learning techniques. Khedr, Salama & Yaseen (2017) describe related work as including relating news or twitter data to stock market behaviour and prices and relating financial news to stock prices, but do not mention presidential speeches. Searching ‘machine learning S&P 500’ on Google Scholar yields an array of articles using machine learning as a technical analysis tool but most of these use other financial indicators and are not NLP based – bar one article analysing the effects of former U.S. president, Donald Trump’s tweets on the S&P 500 and the DJIA. Searching ‘political speech machine learning’ on Google Scholar yields articles that focus on political speeches but have no link to stock markets. Other searches yielded similar results, thus as far as can be told – this research is novel in nature.

## Data cleaning methods for NLP

Katre (2019), in his analysis of Indian political speeches, uses Natural language Toolkit (NLTK) and string methods to remove punctuation, HTML tags and English stopwords[[2]](#footnote-31) , as well as converting speeches to lowercase and tokenizing[[3]](#footnote-32) them. Zubair & Cios (2015), before correlating the sentiment in Reuters articles with S&P 500 movements, clean their text data by tokenizing it with NLTK. Kinyua et al. (2021) clean their twitter data (tweets from then U.S. president, Donald Trump) by deleting all tweets on days when the stock trading was closed, deleting all tweets that only contained standard stopwords and deleting all tweets that only contained URLs. Khedr, Salama & Yaseen (2017) tokenize, standardize by converting to lowercase, remove stopwords from and stem[[4]](#footnote-33) their textual data before processing the abbreviations (replacing abbreviations with the full phrase) and filtering out words that consist of two or less characters.

## Machine learning methods

The following section looks first at the literature informing the sentiment analysis space and then at the literature around stock market prediction in order to determine methods that suit the intersection between the two.

### Sentiment analysis methods

Ren, Wu & Liu (2019) analyse news articles at the sentence level by assigning a sentiment polarity (using software designed for the Chinese language) to each word followed by a sentiment score for each sentence in a document. Each document is then categorized and a sentiment score between -1 and 1 for all news for that day is generated. Zubair & Cios (2015) use the positive and negative valence categories from the Harvard General Enquirer (HGI) to assign each word in a Reuters news article a positive or negative label. They then sum the positives and negatives into a tuple and divide that tuple by the number of words in an article in order to create a vector that represents each news article. The vectors are organized into time series, normalized by dividing all vectors by the first vector, parsed through a Kalman filter and then correlated to S&P 500 returns using Pearson correlation (for both the positive and negative scalar in the vector). Kinyua et al., (2021) use the Valence Aware Dictionary for Sentiment Reasoning (VADER) to create a sentiment feature for former U.S. president Donald Trump’s tweets which was then used as a regression feature in linear, decision tree and random forest regressions. Khedr, Salama & Yaseen (2017) use N-gram (n=2) to extract key phrases from their corpus of news text data, then term-frequency inverse-document-frequency is used to determine the importance of those phrases within the corpus, and finally use a naïve-Bayes classifier to assign positive and negative labels to each news document. Purevdagva et al. (2020) use a variety of features present in both data and metadata to predict fake political speech. Two features relevant to this project were ‘speaker job’ and ‘context’ (press, direct or social) which were labelled using universal serial encoders. For the actual sentiment analysis they used the linguistic inquiry and word count (LIWC) tool to categorize and count words into emotional, cognitive and structural text components. Various further attempts to extract sentiment from the text did not yield increased prediction accuracies. They go on to use an extra tree classifier for feature selection and then support vector machine (SVM), multilayer perceptron, convolutional neural network, decision trees, fasttext and bidirectional encoder representations from transformers (BERT) for prediction with highest accuracy coming from the SVM. Dilai, Onukevych & Dilay, (2018) use SentiStrength – an automatic sentiment analysis tool - to compare the sentiment in speeches between former U.S. president Donald Trump and former Ukrainian president Petro Poroshenko.

### Stock market prediction methods

Ren, Wu & Liu (2019) use an SVM and fivefold cross validation approach to achieve a prediction accuracy of 98% when predicting fake news in political speech. They combined sentiment data and market indicators as their input data. Kinyua et al. (2021) use linear, decision tree and random forest regressions to predict S&P 500 and DJIA directional changes. Random forest regression performed best for both datasets. Khedr, Salama & Yaseen (2017) use open, high, low and close prices from their stock market data as features after labelling the change from the previous day as positive, negative or equal. Jiao & Jakubowicz (2017) extracted lag and window features from the S&P 500 and the global 8 index before running time series random forest, neural network and gradient boosted trees to predict movements of individual stocks in the S&P 500. Liu et al. (2016) used forward search feature selection to select features for SVM, naïve-Bayes, Gaussian discriminant analysis and logistic regression from a set of economic features including the crude oil daily return, currency exchange rates and major stock indices daily returns in order to forecast the S&P 500 movement.

## Summary of literature review

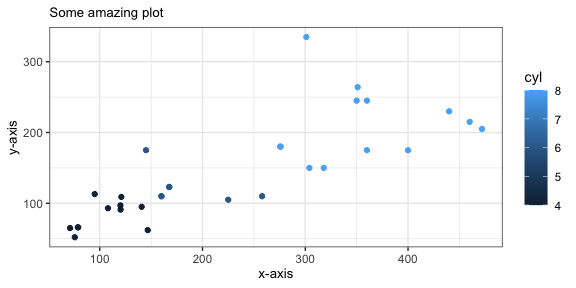
Table 2.1 represents the literature review in a condensed format which allows for easy comparison of the data and methods used and resulting accuracies. Table 2.2 represents the metadata, linked through ‘Paper number’ for Table 2.1.

Condensed literature review

| Paper number | ML Method | Cleaning method | Data type | Index predicted | Max ccuracy |
| --- | --- | --- | --- | --- | --- |
| 1 | Support vector machine with fivefold cross validation |  | Daily online stock reviews | SSE 50 | 0.98 |
| 1 | Support vector machine with rolling windows |  |  |  | 0.9 |
| 1 | Logistic regression with fivefold cross validation |  |  |  | 0.87 |
| 2 | Non-ML: | Tokenized and mined using the Harvard General Inquirer dictionary | Reuters textual data | S&P 500 | Corr: -0.91 |
| 3 | Random forest | Date validation, stop word and URL removal | Tweets | INDU | 0.98 |
| 3 | Decision tree |  |  |  | 0.97 |
| 3 | Logistic regression |  |  |  | 0.81 |
| 3 | Random forest |  |  | S&P 500 | 0.92 |
| 3 | Decision tree |  |  |  | 0.88 |
| 3 | Logistic regression |  |  |  | 0.77 |
| 4 | N-gram, TF-IDF, Naïve Bayes, K-NN | Tokenize, stopwords, stemming, abbreviation processing | News articles and financial reports | Three tech stocks | 0.9 |
| 5 | TS logistic regression | Feature extraction: lags, window features | Financial indicators | Individual S&P 500 stocks | 0.79 |
| 5 | TS random forest |  |  |  | 0.78 |
| 5 | TS neural network |  |  |  | 0.78 |
| 5 | TS gradient boosting |  |  |  | 0.78 |
| 6 | Logistic regression | Date validation forward search feature selection | Market indices, exchange rates | S&P 500 | 0.61 |
| 6 | Gaussian discriminant analysis |  |  |  | 0.61 |
| 6 | Naïve Bayes |  |  |  | 0.6 |
| 6 | Linear SVM |  |  |  | 0.6 |
| 6 | Radial Basis Function SVM |  |  |  | 0.63 |
| 6 | Polynomial SVM |  |  |  | 0.6 |
| 7 | SVM | Feature extraction and feature selection | Political speeches and metadata | Liar dataset | 0.74 |
| 7 | Multilayer perceptron |  |  |  | 0.55 |
| 7 | Convolutional neural network |  |  |  | 0.61 |
| 7 | Fasttext |  |  |  | 0.66 |
| 7 | BERT |  |  |  | 0.66 |

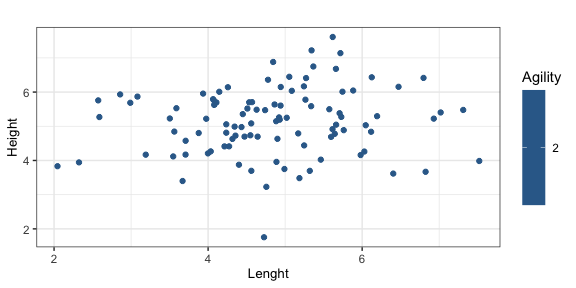
Table 2.1 metadata

| Paper number | Title | Citation | DOI |
| --- | --- | --- | --- |
| 1 | Forecasting Stock Market Movement Direction Using Sentiment Analysis and Support Vector Machine | (Ren, Wu & Liu, 2019) | 10.1109/JSYST.2018.2794462 |
| 2 | Extracting News Sentiment and Establishing its Relationship with the S&P 500 Index | (Zubair & Cios, 2015) | 10.1109/JSYST.2018.2794462 |
| 3 | An analysis of the impact of President Trump’s tweets on the DJIA and S&P 500 using machine learning and sentiment analysis | (Kinyua et al., 2021) | 10.1016/j.jbef.2020.100447 |
| 4 | Predicting Stock Market Behaviour using Data Mining Technique and News Sentiment Analysis | (Khedr, Salama & Yaseen, 2017) | 10.5815/ijisa.2017.07.03 |
| 5 | Predicting Stock Movement Direction with Machine Learning:an Extensive Study on S&P 500 Stocks | (Jiao & Jakubowicz, 2017 | 10.1109/BigData.2017.8258518 |
| 6 | Forecasting S&P 500 Stock Index Using Statistical Learning Models | (Liu et al., 2016) | 10.4236/ojs.2016.66086 |
| 7 | A machine-learning based framework for detection of fake political speech | (Purevdagva et al., 2020) | 10.1109/BigDataSE50710.2020.00019 |



Caption Here

To make your graphs look extra nice in latex world, you could use Tikz device. Replace dev - ‘png’ with ‘tikz’ in the chunk below. Notice this makes the build time longer and produces extra tex files - so if you are comfortable with this, set your device to Tikz and try it out:



Caption Here

To reference the plot above, add a ``\label’’ after the caption in the chunk heading, as done above. Then reference the plot as such: As can be seen, Figures and are excellent, with Figure being particularly aesthetically pleasing due to its device setting of Tikz. The nice thing now is that it correctly numbers all your figures (and sections or tables) and will update if it moves. The links are also dynamic.

I very strongly suggest using ggplot2 (ideally in combination with dplyr) using the ggtheme package to change the themes of your figures.

Also note the information that I have placed above the chunks in the code chunks for the figures. You can edit any of these easily - visit the Rmarkdown webpage for more information.

# Splitting a page

You can also very easily split a page using built-in Pandoc formatting. I comment this out in the code (as this has caused issues building the pdf for some users - which I presume to be a Pandoc issue), but you are welcome to try it out yourself by commenting out the following section in your Rmd file.

# Methodology

## Subsection

Ideally do not overuse subsections. It equates to bad writing.[[5]](#footnote-43)

## Math section

Equations should be written as such:

If you would like to see the equations as you type in Rmarkdown, use $ symbols instead (see this for yourself by adjusted the equation):

$$
\beta = \sum\_{i = 1}^{\infty}\frac{\alpha^2}{\sigma\_{t-1}^2} \\
\int\_{x = 1}^{\infty}x\_{i} = 1
$$

Note again the reference to equation . Writing nice math requires practice. Note I used a forward slashes to make a space in the equations. I can also align equations using **&**, and set to numbering only the first line. Now I will have to type ``begin equation’’ which is a native command. Here follows a more complicated equation:

Note that in I have aligned the equations by the equal signs. I also want only one tag, and I create spaces using ``quads’’.

See if you can figure out how to do complex math using the two examples provided in and .

# Results

Tables can be included as follows. Use the *xtable* (or kable) package for tables. Table placement = H implies Latex tries to place the table Here, and not on a new page (there are, however, very many ways to skin this cat. Luckily there are many forums online!).

To reference calculations **in text**, *do this:* From table we see the average value of mpg is 20.98.

Including tables that span across pages, use the following (note that I add below the table: ``continue on the next page’’). This is a neat way of splitting your table across a page.

Use the following default settings to build your own possibly long tables. Note that the following will fit on one page if it can, but cleanly spreads over multiple pages:

## Huxtable

Huxtable is a very nice package for making working with tables between Rmarkdown and Tex easier.

This cost some adjustment to the Tex templates to make it work, but it now works nicely.

See documentation for this package [here](https://hughjonesd.github.io/huxtable/huxtable.html). A particularly nice addition of this package is for making the printing of regression results a joy (see [here](https://hughjonesd.github.io/huxtable/huxtable.html#creating-a-regression-table)). Here follows an example:

If you are eager to use huxtable, comment out the Huxtable table in the Rmd template, and uncomment the colortbl package in your Rmd’s root.

Note that I do not include this in the ordinary template, as some latex users have complained it breaks when they build their Rmds (especially those using tidytex - I don’t have this problem as I have the full Miktex installed on mine). Up to you, but I strongly recommend installing the package manually and using huxtable. To make this work, uncomment the *Adding additional latex packages* part in yaml at the top of the Rmd file. Then comment out the huxtable example in the template below this line. Reknit, and enjoy.

**Table** : Regression Output

|  |  |  |  |
| --- | --- | --- | --- |
|  | Reg1 | Reg2 | Reg3 |
| (Intercept) | -2256.361 \*\*\* | 5763.668 \*\*\* | 4045.333 \*\*\* |
|  | (13.055) | (740.556) | (286.205) |
| carat | 7756.426 \*\*\* |  | 7765.141 \*\*\* |
|  | (14.067) |  | (14.009) |
| depth |  | -29.650 \* | -102.165 \*\*\* |
|  |  | (11.990) | (4.635) |
| N | 53940 | 53940 | 53940 |
| R2 | 0.849 | 0.000 | 0.851 |
| \*\*\* p < 0.001; \*\* p < 0.01; \* p < 0.05. | | | |

FYI - R also recently introduced the gt package, which is worthwhile exploring too.

# Lists

To add lists, simply using the following notation

* This is really simple
  + Just note the spaces here - writing in R you have to sometimes be pedantic about spaces…
* Note that Rmarkdown notation removes the pain of defining environments!

# Conclusion

I hope you find this template useful. Remember, stackoverflow is your friend - use it to find answers to questions. Feel free to write me a mail if you have any questions regarding the use of this package. To cite this package, simply type citation(“Texevier”) in Rstudio to get the citation for [Katzke](#ref-Texevier) ([2017](#ref-Texevier)) (Note that uncited references in your bibtex file will not be included in References).

# References

Fama, E.F. & French, K.R. 1997. Industry costs of equity. *Journal of financial economics*. 43(2):153–193.

Grinold, R.C. & Kahn, R.N. 2000. Active portfolio management.

Katzke, N.F. 2017. *Texevier: Package to create elsevier templates for rmarkdown*. Stellenbosch, South Africa: Bureau for Economic Research.

# Appendix

## Appendix A

Some appendix information here

## Appendix B

1. ‘Linguistic factors’ in this sense is intended to mean any and all patterns that can be detected in spoken language including verbiage, lexicon, tone, register, sentence length, word combination etc. [↑](#footnote-ref-23)
2. ‘Stopwords’ are words that commonly occur across all speech and therefore only create noise in the data. Some examples are ‘the’, ‘it’, ‘they’ and ‘and’. [↑](#footnote-ref-31)
3. ‘Tokenizing’ refers to the splitting of words into tokens that have linguistic importance, for example the words ‘terrorism’, ‘terrorist’ and ‘terror’ may all be tokenized to ‘terror’. Thus, the core concept of the word is captured while also simplifying the dataset. [↑](#footnote-ref-32)
4. ‘Stemming’ refers to the removal of suffixes to reduce the complexity of a dataset. [↑](#footnote-ref-33)
5. This is an example of a footnote by the way. Something that should also not be overused. [↑](#footnote-ref-43)