# USING MACHINE LEARNING TO PREDICT INTEREST RATE CHANGES FROM FEDERAL RESERVE PROCEEDINGS

WRITTEN FOR CS 229: MACHINE LEARNING

INDIRA PURI

Abstract. Using Federal Reserve transcripts from 1982-2008, our goal is the following: sitting at a particular Federal Reserve meeting, can we predict what the Fed will do to rates (hold, raise, lower) at the next meeting, about one and a half months in the future? We show that predicting interest rate changes can be formulated as a text classification problem. Using this formulation, we can take advantage of classic machine learning text classification techniques. Using these techniques, our best model has an average out-sample accuracy of 73%, and an in-sample accuracy of 78%. In comparison, previous research has shown using publicly available economic indicators, including inflation, unemployment, financial indicators, and Federal Open Market Committee members' predictions of interest rates, one day before the decision (as opposed to one and a half months prior to decision) to predict changes in Fed policy only results in about 75% in-sample accuracy (Lapp et al., 2003), and that, in the private sector, from 1989-1993, the fed futures markets anticipated 41% of interest rate changes; from 1994-2000, that market anticipated 76% of interest rate changes (Lange et al., 2003). The remarkable fact is that, using only publicly available transcripts– no explicit economic indicators, nor prior knowledge about the members of FOMC, nor knowledge of financial markets or discussions with brokers – the computer via our classification algorithms is able to ballpark an expert (private sector) level of accuracy, and to surpass the prior level of accuracy using publicly available knowledge.

## 1. Introduction

US interest rates play a large role in the domestic and international economy. Eight times a year, when the Federal Open Market Committee meets to decide on interest rates, the world watches with baited breath. All else equal, higher interest rates on US bonds mean, for example, that developing countries' bonds look relatively less attractive to investors; that yields on company bonds must increase, i.e. it is more expensive for companies to raise money; that the rate an individual pays for a mortgage or student loan increases; and that, because higher interest rates discourage spending in the present, inflation should decrease. The interest rate the United States sets affects all facets of the economy, and has overseas repercussions as well; the attention paid to it is well-justified.

But these terms are vague. What is "the" interest rate, who in the United States sets it, and how do they do so? When commentators speak of "the" interest rate that the Federal Reserve sets, they usually refer to the federal funds rate. The federal funds rate is the rate at which banks lend money to each other overnight. A target for this rate by the Federal Open Market Committee (FOMC), which is made up of the seven members of the Board of Governors of the Federal Reserve and the twelve reserve bank presidents, eight times per year. The mechanisms of how the federal reserve achieves the target rate are not important to this paper; it suffices to know that the Federal Reserve sets a target federal funds rate, and that there are methods at the FOMC's disposal to ensure this target is met.

Given that the decision of the Federal Reserve as regards the fed funds rate is important to domestic and global players, it would be useful for many entities (banks, companies, developing countries, entrepreneurs, venture capitalists, etc.) to predict the target the Fed will set. The goal of this paper is to predict how the fed funds target rate changes, given transcripts from previous meetings[1]. In particular, we have as our predictor transcripts from the previous $n$ meetings (we explore which $n$ is optimal as part of our paper), and wish to predict whether the FOMC will raise, hold, or lower their fed funds target at the next meeting. For example, if we are in March 2008, we would wish to predict whether the FOMC will raise, hold, or lower the target relative to the current rate at their April 2008 meeting, using the transcripts that the Federal Reserve has already published.

Because our data pairs $(X, Y)$ are (transcripts at previous $n$ meetings, whether the Fed raised/lowered/held rate at next meeting), this can be viewed as a text classification problem, where our labels are "raised at next meeting (+1)", "held at next meeting (0)", or "lowered at next meeting (-1)". We may therefore utilize text classification machine learning techniques. In particular, we examine use Bayesian Logistic Regression, Naive Bayes, and Support Vector Machines (SVMs) to determine how well they can tackle this particular problem.

From a machine learning perspective, it would be interesting to also see how well neural networks perform on this problem. But, because this paper is geared towards an economics journal where econometrics techniques must be fully explained, and because neural nets are not yet fully understood even by the CS community, we eschew these newer techniques in favor of older, but still solid (we do outperform prior academic models in the field) techniques.

Predicting rates from transcripts is a challenging problem because we have (by machine learning standards) a very small dataset, and because we have to make important decisions on how to represent transcripts and which model evaluation metric to use. All are discussed at length in the methodology section.

Department of Economics, Stanford University. Department of Mathematics, Stanford University.

*Date*: December 16, 2016.

[1] Transcripts are detailed records of Federal Open Market Committee meetings. For discussion on the various types of records the Federal Reserve keeps, see `https://www.federalreserve.gov/monetarypolicy/fomc_historical.html`.

We do not set the number $n$ of previous transcripts because it is not clear what is the optimal number of prior transcripts to consider; for example, the Fed may have indicated three meetings ago something they were intending to do with interest rates this meeting. Because, in practice, the Fed's forward guidance seems to be considered most reliable until about one year out (Smith and Becker (2015)), we explore $n = 1, 2, ..., 8$.

Finally, we choose not to include explicit economic indicators. There are a few reasons for this. First, there are many economic indicators and deciding which indicators to use and how to incorporate them is itself a point of research (see ex. Lapp et al. (2003), Poole (2005)). The focus of this paper is on incorporating machine learning into macroeconomics; we do not wish to be bogged down by a secondary debate over which economic indicators to include and how. Second, because members of the FOMC discuss the economic indicators they believe are relevant at their meetings, the transcripts will include which economic indicators are important to the Fed as well as what their numbers are (ex. "U6 unemployment at 9.8%"). Adding explicit economic indicators is therefore extraneous. The computer should be able to deduce which, if any, indicator-level pairs are important to FOMC from processing the transcripts, so long as we process them intelligently.

The contribution of our research is twofold. First, we provide a means via which machine learning can be incorporated in macroeconomics to solve an important question. Second, our research can be used for prediction of interest rates far in advance. In particular, if our accuracy rate is high, our research provides a means via which the public may anticipate changes in the interest rate without extensive background or knowledge, because transcripts are publicly accessible. Since we predict interest rate changes about six weeks prior to decision, when no new economic indicator data has been released, our research also provides a method for institutions and private bodies to anticipate interest rate changes well in advance of their occurrence.

### 1.1. Related Literature.

To our knowledge, this is the first paper using machine learning to analyze the relationship between the interest rate and Fed transcripts. Our work is most closely related to previous literature on the predictability of the Fed. Lapp et al. (2003) attempts to predict the direction of Federal Reserve interest rate changes from 1979-1995 using publicly available data, including financial and macro indicators, and Fed members' own forecasts of these as stated in the Federal Reserve greenbooks. They find that publicly available data does not accurately predict federal reserve decisions. Sellon (2008) examines the accuracy of private sector surveys for numerical changes in the fed funds target rate, and finds that, on average, private sector surveys one quarter out are on average off by only 0.21 basis points. Lange et al. (2003) look at how well the fed funds futures markets predicts numerical interest rate changes, and find that from 1989-1993, the fed futures markets anticipated 41% of interest rate changes; from 1994-2000, the fed futures market anticipated 76% of interest rate changes. Besides these, there is a wealth of research looking at the language the Federal Reserve uses (Acosta (2015), Hansen et al. (2014)) and at the accuracy and measure of fed funds futures, especially as they relate to changes in Federal Reserve communications (Poole (2005), Piazzesi and Swanson (2008)).

## 2. Methodology

### 2.1. Data.

For target fed funds rate, we use the data series "target fed funds rate" provided by FRED, the Economic Research division of the Federal Reserve Bank of St. Louis. This series spans 1982-2008. We are grateful to Miguel Acosta, who has previously examined transparency of the federal reserve by comparing their different types of documentation via latent semantic analysis (Acosta, 2015), for providing us with transcripts already downloaded and processed from the federal reserve website, from 1976-2008.

### 2.2. Tools.

We use the scikit-learn library (Pedregosa, 2011) in Python.

### 2.3. Representing transcripts.

We use counts of tuples consisting of three words to represent a transcript. For example, if our transcript consists of the single sentence "Three percent inflation! Yes, three percent inflation", the set representing this document would be {(three, percent, inflation): 2 , (percent, inflation, yes): 1 , (yes, three, percent): 1}. Three words was chosen via hyperparameter tuning: in data evaluation, moving from one word to two words to three words maximum increases the cross-validation score, but moving from three words to four words or greater does not.

We considered using a common approach in natural language processing, which is using the presence or absence of single words and eliminating "stop words", words that in English tend to have neutral meaning ("the", "is", "will", so on), but reject this approach for three reasons.

First, using single word features does not capture the immense importance in this context of specific adjective-noun pairs. For example, consider a transcript which says "Inflation is high. The interest rate is low." and a transcript which says "The interest rate is high. Inflation is low." The former corresponds to a signal that the Fed may raise interest rates; the latter corresponds to a signal that the Fed may lower. These are two very different transcripts, but single-word features with part of speech tagging would represent both documents in precisely the same manner.

Second, in this context, neutral words play an important role. Saying "inflation is high" corresponds to a signal that, at present, inflation may be a concern. On the other hand, "inflation was high" is justification for a previous decision. It does not convey the message that, at present, inflation is a concern, and therefore the latter phrase has less bearing on the Federal Reserve's present decision. The difference is in the "neutral" words that the common NLP strategy seeks to omit.

Finally, because these transcripts represent discussion among the FOMC members, the number of times a word occurs, and not merely its presence or absence, has some significance. For example, a transcript in which "inflation is high" occurs 70 times is probably more likely to be one where Fed members are seriously considering raising rates than one in which the phrase occurs only once.

2.4. **Evaluation Metric.** Ideally we would have train, test, and validation sets, but because we have only 210 observations, we opt for leave-one-out cross-validation to assess the quality of a particular algorithm and set of parameters. We choose leave-one-out cross validation for three reasons. First, our dataset is sufficiently small by machine learning standards that to effectively use machine learning, we should be using as much data per fold as possible. Second, in practice, we would like to predict what the Fed will do at the next meeting: we only wish to make one prediction, so an average error estimate on one prediction is desirable. Third, any validation set would necessarily be quite small (around 10 observations), meaning that results on the validation set would have immensely high variance. Because we compare models on the cross-validation score, we should keep in mind that the final cross-validation score may be optimistic relative to true generalization error. To mitigate this issue, we perform hyperparameter tuning on three folds instead of the leave-one-out cross validation metric we use as our evaluation method; these approximate a test and validation set, respectively.

Because cross-validation can be done using many types of scores, the next decision is which score to use. A natural decision is accuracy. In this context, however, choosing to cross-validate on accuracy leads us astray because of skewness in the data. From 1982-2008, the fed choose to hold rates 104 times, to raise rates 50 times, and to lower rates 56 times. Because "hold" is represented more than the other two categories, a predictor that simply predicts "hold" for each meeting would obtain an accuracy rate of 50%.

But such a predictor is useless to us. We would prefer, for example, a predictor that guesses "raise" correctly 55% of the time, "lower" correctly 50% of the time, and "hold" correctly 45% of the time because such a classifier is using the information we give it to supply a weakly accurate guess. But, because "hold" is over-represented in the data, such a classifier would only have an accuracy rate of 48% – lower than the dumb strategy of guessing "hold" each time!

When selecting models, we should therefore use a metric other than accuracy. The usual fix in text classification problems with skewness in labels is to use the average F1 score to select models. The F1 score of a classifier is defined as follows:

For each category, define the precision as $\frac{\text{True positives}}{\text{True positives + False positives}}$ and define recall as $\frac{\text{True positives}}{\text{True positives + False negatives}}$. The F1 score is then the harmonic mean of precision and recall.

Intuitively, recall is the number of correct classifications to number of actual instances of the classification – it penalizes us for under-predicting a category. Precision is number of correct classifications to the number of predicted classifications – we do not wish to over-classify to a particular category (this gets rid of the dumb always-predict-hold classifier). Because we neither want to under-predict a particular category (which recall captures) nor over-predict a particular category (which precision captures), we take the harmonic mean of the two (the F1 score) to capture classification accuracy.

Therefore we compare models based on cross-validated average F1 score. When comparing our models to benchmarks derived from previous academic work, we will revert to discussing the accuracy of our best model for comparison purposes.

2.5. **Classification algorithms and mutual information score.** We examine the performance of classic text classification algorithms, Multinomial Naive Bayes (MNB), Support Vector Machines, and Bayesian Logistic Regression. We provide a brief description of each algorithm here. More detailed derivations with examples can be found at `http://cs229.stanford.edu/materials.html`.

Multinomial Naive Bayes builds a probability distribution for each classification, then classifies an unseen report in the category it is most likely to be in per the previously built probability distributions. Precisely, using the training dataset, MNB sets the probability of a tuple $t$ occurring in a document with classification $c$ as

$$p(t|c) = \frac{\lambda + \#\text{ times } t \text{ occurs in documents with classification } c}{\lambda * (\#\text{ of unique tuples in documents with classification } c) + \text{Cumulative } \#\text{ of tuples in documents with classification } c},$$

where $\lambda > 0$, the Laplace smoothing constant, penalizes overfitting when nonzero. The classification of a given document with tuples $t_1, .., t_k$ is then $\underset{c}{\operatorname{argmax}}\; p(c|t_1, .., t_k) \propto p(c) \prod_{i=1}^{k} p(t_i|c)$.

Before we discuss Support Vector Machines (SVMs) and Bayesian Logistic Regression, we remind the reader that, to a computer, our transcript representations are simply sets of vectors in some very high-dimensional space. Each axis of this space represents one feature: for example, a transcript {(inflation, is, high): 1, (unemployment, is, low): 3} could be represented as a set of vectors $\{(1, 1), (2, 3)\}$, by mapping each tuple to a unique number. Given binary labels $y = -1$ and $y = 1$, Support Vector Machines and Bayesian Logistic Regression take advantage of this representation to find a weight vector $w$ that minimizes $L(x, y; w) + R||w||^2$, where $R$ is a regularization term that reduces overfitting, $L(x, y; w)$ is hinge loss $\sum_i \max(0, 1 - y^{(i)}(w^T x^{(i)}))$ for SVMs, and $L(x, y; w)$ is logistic loss $\sum_i \log(1 + \exp(-y^{(i)} \theta^T x^{(i)}))$ for Bayesian Logistic Regression. With multiple categories, both SVMs and Bayesian Logistic Regression find weight vectors $w_c$ for each category $c$ by solving the prior minimization problem, then classify a new document $x^{(\text{new})}$ as $\underset{c}{\operatorname{argmax}}\; w_c^T x^{(\text{new})}$.

When performing feature selection, we use the mutual information score, common practice for text classification problems. The mutual information score between a feature and the set of classification labels measures how much information the former gives about the latter. For example, if in the training set a given tuple occurs 6 times in "hold" documents, 6 times in "raise" documents, and 6 times in "lower" documents, knowing that tuple tells us very little about which type of document we are looking at – the mutual information score will be low. On the other hand, when considering a new document, knowing the count of a tuple that we know appeared 3 times in "lower" documents, 30 times in "hold" documents, and 300 times in "raise" documents in the training set tells us much more about what the new document should be classified as. Formally, the mutual information score between two variables $T$ and $Y$ is $MI(T; Y) = \sum_{y \in Y} \sum_{t \in T} p(t, y) \log\left(\frac{p(t,y)}{p(t)p(y)}\right)$.

## 3. Results

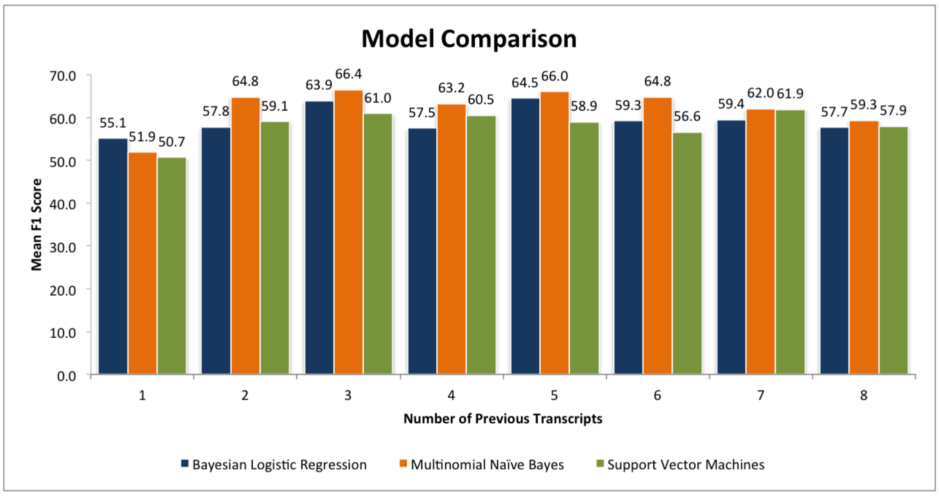

Figure 1. Model comparison for Bayesian Logistic Regression, SVMs and MNB.

3.1. **Best Classifier and Number of Previous Transcripts.** Models for the three different classification techniques with differing numbers of previous transcripts, each tuned to optimal hyperparameters, are shown in Figure 1. While each of the $n$ previous transcripts produce similar results, our best model is the one with the highest average F1 score, Multinomial Naive Bayes with $n = 3$ previous transcripts. This model has an average out-sample F1 score of 66.4.

The fact that the models tend to perform similarly and well above random (random would correspond to an F1 score of 0.32) is re-assuring: it tells us that Federal Reserve transcripts do have predictive power for interest rate changes, regardless of the model used.

3.2. **Feature Selection.** From the above data, Multinomial Naive Bayes with 3 previous transcripts is our best model. Let's see if we can improve it further.

We have 27,100 features. To reduce over-fitting and discard irrelevant features, we may use the mutual information score to select the top $k$ features, then determine which $k$ is best via cross-validation. Models with different numbers of features, each tuned to optimal hyperparameters, are presented in Figure 2 . It is easy to see that the best model is that with 15,000 features, which has an average out-sample F1 score of 72.7.

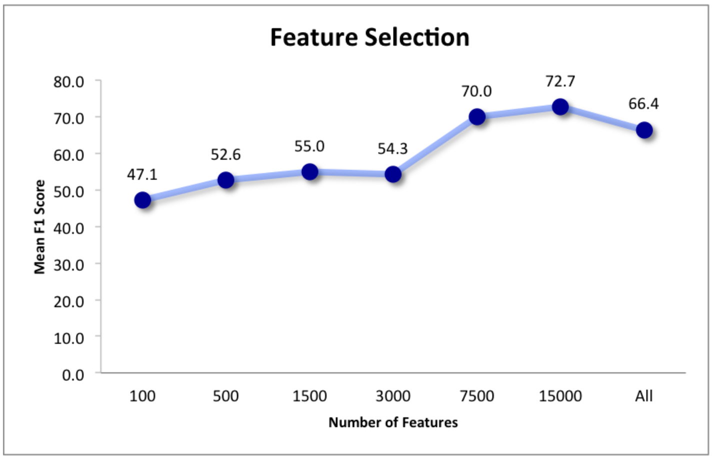

Figure 2. Feature selection for Multinomial Naive Bayes, with 3 previous transcripts.

3.3. **Best Model.** As can be seen from the previous two subsections, after comparing SVMs, Bayesian Logistic Regression, and Multinomial Naive Bayes, and performing feature selection, our best model is Multinomial Naive Bayes with 3 previous transcripts, using the 15,000 top features chosen via the mutual information score. The confusion matrix for this figure is presented in Figure 3. This model has an average out-sample F1 score of 72.7, a corresponding average out-sample accuracy of 73.1 percent, and an in-sample accuracy of 78.4 percent.

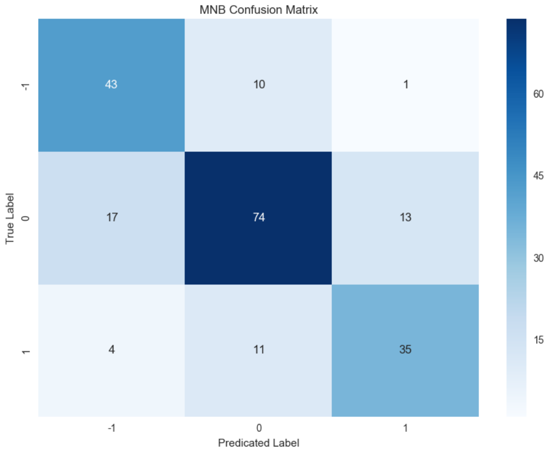

Figure 3. Confusion matrix for best-performing model, Multinomial Naive Bayes with three previous transcripts and 15,000 features.

3.4. **Benchmark.** Because this is the first paper to process Federal Reserve transcripts to predict interest rates, there is no analogous benchmark, but we can look at the success of similar problems:

(1) Models that predict interest rate changes using publicly available data
(2) The accuracy of aggregate market expectations, which are driven by sophisticated and informed traders

In the first category, Lapp et al. (2003) examine how well publicly available data does predicts the direction of Fed decisions. The differences in their setup relative to this paper are (1) Lapp et al. (2003) tries to predict the direction of Fed decisions one day in advance, as opposed to six weeks in advance like this paper. This means that Lapp et al. (2003) allows access to new economic indicators, which this paper does not do, and (2) that Lapp et al. (2003) uses publicly available economic indicators, and also tries using manually inferred Fed members' forecasts of those indicators, rather than having the computer read in and processing transcripts. The highest in-sample accuracy on any of the models in Lapp et al. (2003) is 75% (Lapp et al. (2003), Table 3). In comparison, the in-sample accuracy of our best model, Multinomial Naive Bayes with 3 previous transcripts, is 78%. The real test, of course, is out-sample accuracy, which Lapp et al. (2003) does not consider. Still, it is notable that our model outperforms one using explicit economic indicators in a shorter timeframe to announcement.

In the second category, Lange et al. (2003) shows that from 1989-1993, the fed funds futures markets anticipated 41% of interest rate changes, and from 1994-2000, 76% of interest rate changes. The fed funds futures market reflects the anticipations of informed market participants, including banks and hedge funds. The setup of Lange et al. (2003) is again not fully analogous to ours for two reasons: (1) their problem is slightly easier in that they look at prediction accuracy about one month in advance (as they consider futures) whereas ours is one and a half months in advance, so the fed funds market has access to slightly more information and (2) their problem is harder in that they look at numerical accuracy instead of directional accuracy. This means, for example, that if the fed funds futures market were predicting a 0.25 bp raise in the interest rate, and it actually rose by 0.50 bps, Lange et al. (2003) would consider the prediction inaccurate. Our model would consider the prediction accurate because both are a "raise" in the interest rate. Our average out-sample accuracy, obtained via leave-one-out cross validation, is 73%. We can therefore say we ballpark the level of private sector accuracy, but because the setup in Lange et al. (2003) is so different from ours, it is difficult to be more precise.

## 4. Conclusion

We use classic machine learning techniques to predict changes in interest rates using Federal Reserve transcripts as input. Our best model has in-sample accuracy of 78% and average out-sample accuracy of 73%. In comparison, previous research has shown using publicly available economic indicators one day before the decision (as opposed to two months prior to decision) to predict changes in Fed policy only results in about 75% in-sample accuracy (Lapp et al., 2003), and that, in the private sector, from 1989-1993, the fed futures markets anticipated 41% of interest rate changes; from 1994-2000, the fed futures markets anticipated 76% of interest rate changes (Lange et al., 2003). The remarkable fact is that, using only transcripts– no explicit economic indicators, nor prior knowledge about the members of FOMC, nor knowledge of financial markets or discussions with brokers – the computer via our classification algorithms surpass prior levels of accuracy using publicly available data and ballparks the level of accuracy of the private sector. Further research could include examining the accuracy of transcripts two meetings in advance (two months before decision) and looking at how well machine learning predicts numerical, as opposed to directional, interest rate changes.

## References

Acosta, M. 2015. *FOMC Response to Calls for Transparency*, Finance and Economics Discussion Series 2015-060. Washington: Board of Governors of the Federal Reserve System.

Hansen, S., M. McMahon, and A. Prat. 2014. *Transparency and deliberation within the FOMC: a computational linguistics approach*, CFM discussion paper series, CFM-DP2014-11. Centre For Macroeconomics.

Lange, J., B. Sack, and W. Whitesell. 2003. *Anticipations of Monetary Policy in Financial Markets*, Journal of Money, Credit and Banking **35**, 889–909.

Lapp, J., D.K. Pearce, and S. Luksanasut. 2003. *The Predictability of FOMC Decisions: Evidence from the Volcker and Greenspan Chairmanships*, Southern Economic Journal **70**, 312–327.

Poole, W. 2005. *How Predictable Is Fed Policy?*, Federal Reserve Bank of St. Louis Review **87**, 659–668.

Sellon, G.H. 2008. *Monetary Policy Transparency and Private Sector Forecasts: Evidence from Survey Data*, Federal Reserve Bank of Kansas City Economic Review.

Smith, A.L and T. Becker. 2015. *Has Forward Guidance Been Effective?*, Federal Reserve Bank of Kansas City Economic Review.

Piazzesi, M. and E.T. Swanson. 2008. *Futures prices as risk-adjusted forecasts of monetary policy*, Journal of Monetary Economics **55**, 677–691.

Pedregosa, F. and Varoquaux, G. and Gramfort. 2011. *Scikit-learn: Machine Learning in Python*, Journal of Machine Learning Research **12**, 2825–2830.