Lehigh University

Predicting the U.S. 10 Year Treasury Yield for Investors

Jesse Fisher, Quan Le, Mitch Fishbein, Prince E. Omuyeh, Devin Pombo

CSE 160 - Introduction to Data Science

Professor Davison

May, 10 2024

Executive Summary:

Our project and research sits at the intersection of data science and finance. During our project, we applied our new understanding of data science technologies and combined that with our fundamental understanding and background in finance to create a program that can help make predictions on what the price of 10 year U.S. treasury bonds will be in one month in the future. This was achieved by creating a linear regression model using supervised learning that took into account several economic factors such as inflation data, economic growth data, and more to assist in making predictions on the price of these bonds.

To begin, we collected data from different sources online and once we had all the data in RStudio, we used several different data cleansing techniques to replace NA values and combine and align the different data frames so we could begin to train the model. After the data was prepared, we trained the linear regression model using random guessing in order to gauge what mean average error (MAE) we needed to surpass to determine if our model was learning anything of value. We then took steps to attempt to improve the model using various techniques we learned throughout the semester including: adding more powerful features, creating non-linear variables, and combining the model. After each change we compared our new MAE to the original and performed analysis observing what was occurring using learning curves and fitting graphs to detect what next steps should be taken next to improve our model.

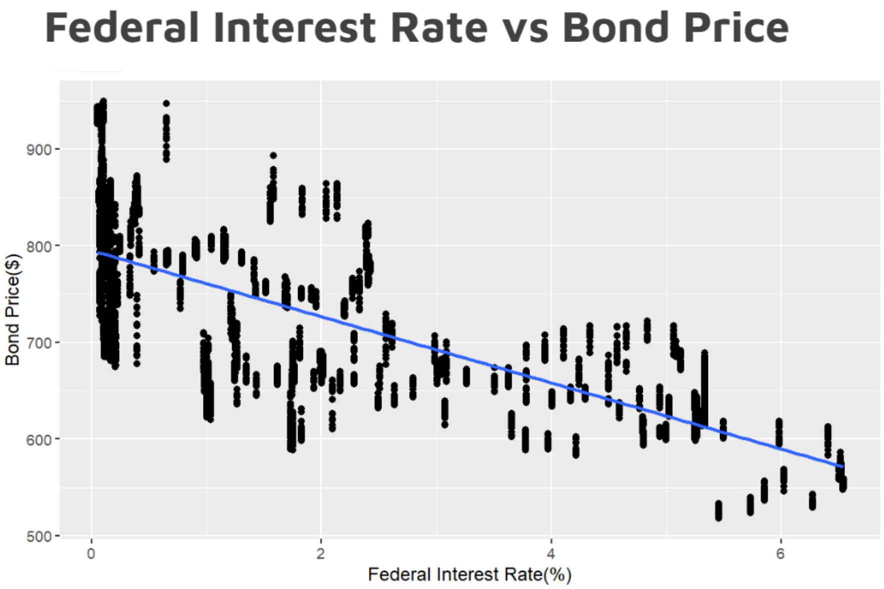
In the end, our model got a MAE that was considerably lower than the MAE from the model that guessed randomly. Therefore we concluded the steps we took to improve our model were successful since it learned something of value to help more accurately predict the future U.S. 10 year treasury price.

In the world of finance, there are several applicable applications where data science technology can be used to enhance several measures. It can aid businesses to forecast profits and expenses, provide analysis of their data, forecast important economic readings and measures, and assist in making predictions on the future value of assets. Our project will focus on the last item in that list: making predictions on what the future value of the U.S. 10 year treasury bond will be in one month from today. Our rationale for picking to predict U.S. treasury bonds for our project is because when compared to other financial assets, U.S. government bond yields have far fewer features than other financial assets affecting their price. Equity, for example, has many factors influencing prices such as news headlines affecting the economy and individual businesses making it more difficult to predict. With linear regression using supervised learning using a wide range of historical economic data influencing these bond prices, this goal was achieved. Our target customers are asset managers and institutional investors searching for a competitive advantage with our predictive model.

Bonds are a simple loan. There are two types of bonds: coupon bonds and zero coupon bonds. The type of bond our project will work with, and this paper will focus on, is a zero coupon bond – the U.S. 10 year treasury bill (10 year). This works by giving money in the present in return for a promise to pay more money back in the future. For example, assume the U.S. government needs to borrow money to fund fixing the highway infracture of the country. It decides it will be able to recoup the investment in 10 years to pay the money it borrowed back to investors so it issues 10 year treasury bills to the public. Treasury bills always give $1,000 when they mature or after the 10 years are up. In order for investors to consider buying this bond, the government must sell it at a discount. The price the government quotes investors today is $600. Simply stated, the government is offering investors to pay $600 today in exchange for $1,000 10 years from now. This is how this type of bond works and that is how the investor makes money. The way the government and investors come to agree on what the minimum price today should be to make it a worthwhile investment is from this formula:

$$\text{Present Value} = FV \frac{1}{(1+r)^{n}}$$

“FV” stands for future value,“n” is the number of years, and “r” is the federal interest rate. In our case, the future value is $1,000 and the number of years is 10. The federal interest rate, although it sounds like an interest rate received from the government, is not the effective interest rate the government is planning to pay for borrowing money. It is something completely different. The federal interest rate is the interest rate the federal reserve or the national bank decides. It is used as a tool to control the economy and inflation. When inflation is high and money is in high circulation. The federal reserve raises interest rates to make borrowing more expensive and which discourages borrowing. For example less mortgages for houses or loans to buy cars are taken out by the public. Consequently, because the public is spending less, businesses make less – the economy is beginning to slow down, demand lessens, and it stops growing as fast. Furthermore, because businesses are making less they have to lay off employees – the unemployment rate ticks up. People who are laid off cannot spend as much as before because they do not have income. Businesses make even less money because demand gets even smaller. It’s a vicious cycle. In this sense, this is how raising the federal interest rate slows down the economy and household spending. This interest rate is the “r” in our formula above and therefore if one could accurately predict future changes to “r”, one could profit from this information because as “r” changes, it has an effect on the present value of the bond today. “r” is in the denominator of the mathematical formula which indicates that it has an inverse relationship to the price today as illustrated below.



As the interest rate goes up, the price of the bond goes down, and vice versa. Lastly, for the remainder of this paper, bonds will not be referred to as having a price but rather yield to maturity (YTM) or "yield" for short. Many different bonds exist and not all of them have a $1,000 future value like this one does. Therefore as a more standardized way of quoting bond prices that have different future values, bonds are quoted in terms of the yield. Investors are shoppers for the best yield. In this sense, quoting bonds in terms of their yield is showing investors what they want and it is the standard. For example, going back to our $600 10 year treasury bond offering $1000 in 10 years example from before, for zero coupon bonds, the YTM is equal to the "r" in the formula from above. If you use the formula PV = FV/(1+r)^n and put in the variables we know: $600 = $1,000 / (1 + r) ^ 10, you can solve for "r" which is ~5.25%. Therefore, this bond would be quoted to you as offering 5.25% every year for 10 years. Notice the inverse relationship between price and yield. If the yield is increasing that is the same as saying the price is decreasing. This makes sense intuitively. If the price decreases from $600 to $500 on the same day it is bought, $1000 is still promised in the future regardless, however, if it is bought at the $500 price, the new "r" is 7.17% by using the formula from before.

Conceptually, the bond has to have a higher yield in order to approach or grow in value until it is worth the $1,000 future value over the same duration of 10 years. For clarification, YTM and federal interest rates are not always the same. For simplicity just understand that for the case of U.S. 10 year they are usually very close; however, this is not true for every bond. For the sake of conceptual understanding consider the following situation where they are always equal. For example, this price movement from $600 to $500 would only be possible if the federal reserve announced it was going to increase interest rates from 5.25% to 7.17%. Therefore if our model could accurately predict movements in the interest rate, one could have been informed to sell their bond for $600 before the interest rate announcement and buy it back for $500 after the announcement and pocket the $100 difference.

In order to predict the value of the 10 year yield in one month, using our financial domain knowledge we decided to use six features that are extremely relevant to predicting the likelihood of future changes to the interest rate and yields of these bonds: the federal reserve interest rate, U.S. 10 year treasury yield, economy growth rate (GDP), unemployment rate, inflation rate (CPI), U.S. 2 year treasury yield. These sets range all the way back to the 1950s. For training our model we decided to only use instances from 2000 to 2024 which accounted for over 6000 instances. After reading in these datasets to RStudio a lot of work was required to cleanse the data, remove NA values, and combine and align the datasets properly. In order to align the datasets, we used the merge function to merge them by date in chronological order. Because a lot of our datasets were only reported on a monthly or quarterly basis, we wrote code in a for-loop that extended the value given at a point in time downward to replace all the NAs beneath it until it encountered the next value. Therefore the instances had all the information available to them at that point in time. Lastly, some of the values were reported on days where U.S. treasuries were not traded and therefore these rows had NAs in the U.S. 10 year treasury column. These instances were removed using na.omit() because there is no way of knowing for certain what the correct yield should have been on that day. Finally, now that all our data was readily available to use in RStudio to train our model we wanted to know how well the model could perform if it were to guess randomly so we would know what our target MAE was to improve upon. We achieved this by creating a model that predicted the future yield in one month with the only dependent variable being the yield today. The yield today in a vacuum by itself without any other economic data does not offer any clue or context on what the price will be tomorrow and it most certainly does not indicate what it will be in one month from today. Before we trained the model we had to create our test and training sets. We first decided to split our data from 2000 to 2024 on January 1st of 2022 which made roughly a 90-10 split. Our reasoning behind this was that in order for our analysis to be successful we need to observe the idea of "Ceteris Paribus" – holding all other things constant and only changing one variable at a time to see the net effect of our experimentation. If we did a random 80-20 split each time, the MAE of each run could be different simply because the sample function chose a random subset for training that was better than the subset before it. However, after performing overfitting analysis we found that when the predict function was given the training data it gave a MAE of 0.1919 and the testing data gave a MAE of 0.2976 indicating that the model was memorizing some characteristics of the training set and not generalizing to the test set. Therefore, we decided to shift back towards the sample function using a 80-20 split and utilized the set.seed() function to guarantee that the split it chose every time was the same so we were still observing "Ceteris Paribus". This time the training and testing MAE were 0.2021 and 0.1998 respectively. The test set had less error – indicating overfitting did not occur. Therefore our MAE we will try to improve upon is 0.1998. If the model on average guesses the price in month with a mean absolute error of 0.1998, then that means on average it is within 0.1998 of the correct value.

Our first model we created simply used all the data we read in the first step: 10 year yield today, GDP, unemployment rate, CPI, 2 year yield today, and the federal interest rate. It had a slight improvement in MAE improving from 0.1998 to 0.1975. Our next idea was to add more powerful features. We used additional financial knowledge to create features that reflected the directional movements of the various variables every three, six, and twelve months. We also created non-linear variables like CPI squared and GDP times CPI. This improved it further and we achieved a MAE of 0.1907. Again we wanted to improve this value. Our next idea was to combine models. We were not sure if the model was learning efficiently or determining at a high level of accuracy whether the yield was moving upwards or downwards in a month's time. Whether the price would move upward or downwards was a binary classification task. So we created a new row that was the future yield minus the current yield. If it was positive, we put a 1 in the row, if it was negative, we put a 0. We decided it was not fair or proper practice to use this column of data for training our regression model because it contained information about the dependent variable that was 100% accurate. We would be essentially giving our model data that was 100% accurate about the future so information about the dependent variable would be used to train the dependent variable. Therefore we instead decided to use a decision tree to predict the value of this column. We trained and tested the tree on the same 80-20 split. We then tested the accuracy of the tree model on the validation set and found it to be 73% accurate. In order to get a column that had a prediction for every entry in the entire dataframe we reran the predict() function taking in the entire dataframe and found it to be 75% accurate. There could have been some slight overfitting since it was performing slightly better on the entire set which included the instances it was trained on, but nonetheless, we removed the column reflecting the data that was 100% accurate and we merged this 75% accurate data from our binary classification model to our data frame and re-trained our model. This time we saw a considerable improvement to MAE. MAE was now 0.1693 as opposed to the original 0.1998 we were trying to overcome which was a 15% improvement. Finally, we wondered if we were overloading the function with too many features and it was not able to pick out the best features so we decided to use a feature selection algorithm to aid us in narrowing down our features. Using the ref() a feature selection algorithm, It gave us 5 features out of the 22 features: the 10 year yield, the binary classification column, 2 year yield, CPI, and the 63 day percent change to GDP. However, when specifying these values to train our new model it actually increased MAE slightly from our previous value to 0.1739.

With the help of linear regression using supervised learning using a wide range of historical economic data influencing these bond yields, we created a model that predicts the future value of the U.S. 10 ten year treasury bill. This was achieved by using financial knowledge to identify data sets that are correlated with predicting the future value of bonds. Then we employed various data science techniques like creating more complex variables and introducing non-linear variables to reduce the MAE. Finally we combined our linear regression model with the output of a binary classification decision tree that reflected with 75% confidence the directional movement of the bond. All of these combined resulted in a MAE of 0.1693 which improved upon our "random guessing" model's MAE of 0.1998 by 0.0305: a 15% improvement. Treasuries and many other publicly traded assets future values are difficult to predict but we conclude that the reduction in MAE indicates our model and experimentation learned something of value by making it more accurate and reliable due to there being less error.

Appendix

Jesse Fisher

- Data Collection and Cleansing
- Training and Validation Test Set Logic
- Feature Selection Algorithm
- Financial Analysis from Finance Background

Quan Le

- Data Collection and Cleansing
- Non-linear Variable Testing

Mitch Fishbein

- Data Collection and Cleansing
- Non-linear Variable Testing

Prince E. Omuyeh

- Data Collection and Cleansing
- Binary Classification Model

Devin Pombo

- Data Collection and Cleansing
- Learning Curves and Fitting Graphs
- GGPlot Visualizations
- Financial Analysis from Finance Background