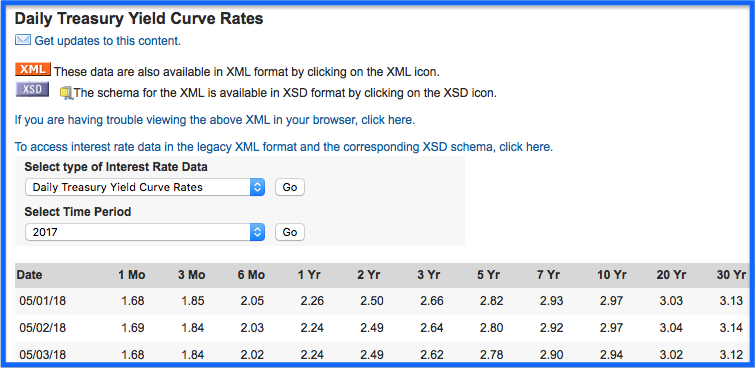
**Capstone Analytic Report and Research Proposal**

**Unit 4 / Lesson 2 / Project 2**

Analytic Report and Research Proposal by John Foxworthy

The dataset that I will be using are the Daily U.S. Treasury Yield Curve Rates and its significance is the global importance in the financial trading industry, data quality and its statistical significance. The figures below have daily dates as rows with maturity points as columns. For example, a ten year is 2.97 percent as of the 1st of May, 2018, which is the amount of the interest rate on a decade long loan. The data goes back to 1990 up until to the current date in 2018.



[Daily\_Treasury\_Yield\_Curve\_Rates](https://www.treasury.gov/resource-center/data-chart-center/interest-rates/Pages/TextView.aspx?data=yield)

Each of the figures represents hundreds of billions of U.S. dollar trades from participants from dozens of countries and have the impressive data attribute of a complete absence of stale data. Not some, but all figures, on each business day of the year, has at least one person and likely many people trying to influence the interest rate values to change. Specifically, the strong liquidity, in other words, the quantity of buyers finding sellers and vice versa is very high, in every trading day, in the year going back decades since the U.S. Dollar is the world’s reserve currency. According to organizations like the International Monetary Fund (IMF), 6 out of 10 or 7 out of 10 transactions by physical hand or electronically, involve the U.S. Dollar. There are more than 200 countries in the world and each have some type of exporting and importing business that does involve the U.S. Dollar. The more a country exports, the more their currency, not the U.S. Dollar, rises in value because of increased demand, thereby making the value of their exports more expensive. To counteract, countries buy U.S. Dollars to dampen their local currency appreciation and the most common format is in the U.S. loan or fixed income market. Borrow a month at 1.68% or a year at 2.26% in U.S. dollars on the 1st of May for your exporting business from the Daily Treasury Yield Curve, for instance. Not to mention, the U.S. economy is the largest country in the world so there are plenty of individuals and institutions that depend on this data to raise money for countless causes from a mortgage loan for a residential property to purchasing a new office building for an expanding company or local government institution.

Lastly, the source of this data is the U.S. Treasury department of the U.S. government. There are no impediments for the use of this data as it public from a legal perspective and officially published by the U.S. government on every business day. Daily Treasury Yield Curve is widely used by countless people and institutions for many causes and I choose it to forward my career in Data Science.

Plotting lines, histograms and other data visualizations from csv, xml or json files will not have an issues of daily rates data, but more importantly, the evolution throughout time over the past decades, would help in advanced statistical techniques in the coming months of the Thinkful Data Science course. Below is some code for the column maturity points of 1 month, 6 month, 1 year, 10 year and 30 year from a json file. There are roughly 250 business days in a given year so beyond the demonstration below there are about 6,750 business days for this single column only from 1990 to 2017. Furthermore, the 11 columns from the 1 month to the 30 year would be more than 75,000 interest rate daily data from 1990 to the current day. For the research report, we will use daily yield curve rates from 2014 to 2017, which is 1,000 data points per column maturity point for a total of 5,000.

# In[1]: import json, matplotlib.pyplot as plt, pandas as pd

%matplotlib inline

# In[2]: json.load((open('/Users/lacivert/2014\_2017.json')))

# In[3]: sample\_json\_df = pd.read\_json('/Users/lacivert/2014\_2017.json')

# In[4]: #assign DataFrame

df = pd.DataFrame(sample\_json\_df)

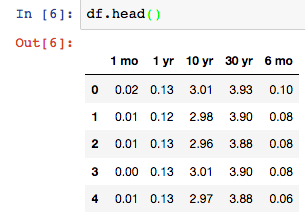
# In[5]: #list column names

list(df.columns.values)

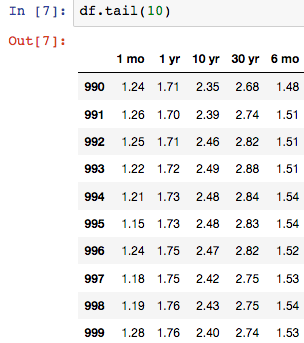
# Out[5]:['1 mo', '1 yr', '10 yr', '30 yr', '6 mo']

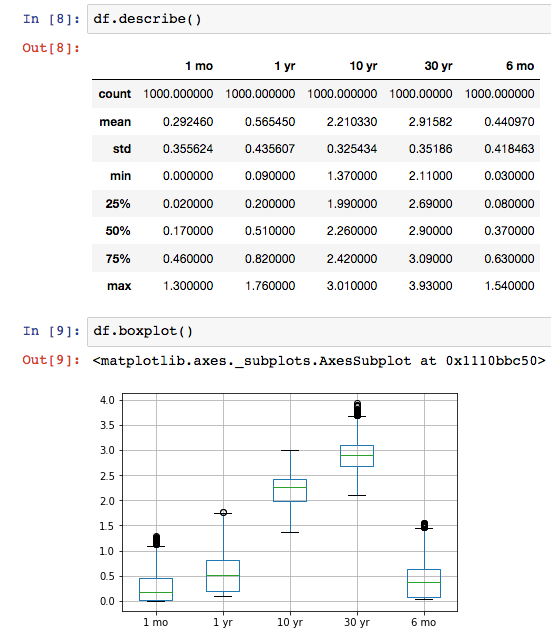
Looking forward, there are three analytical questions to answer . . .

1. How similar is the data set in terms of the minimum, maximum, average, quantiles and variation of the data set?
2. Does the data set cluster together over the 2014 to 2017 time period?
3. How much do the column maturity points vary with each other?



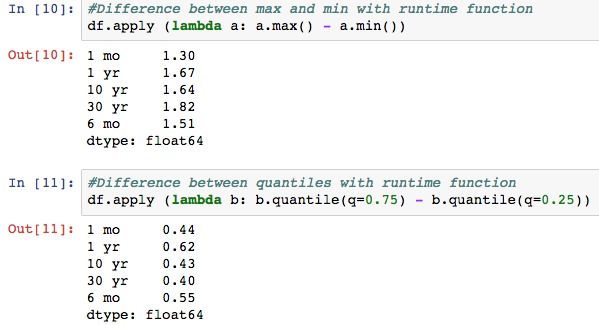
Continuing, we can take a look at the top part of the json file of 2014 and the bottom data points of 2017. We can see a progression of values starting from close to zero to moving away from zero over time. This is one characteristic of all the maturity points of the U.S. Treasury Yield Curve that continues with the descriptive statistics carrying most of this theme, but not all.





The mean, minimum, maximum and all quantiles are in ascending order yielding a similarity, but the variation around the mean, such as captured in one way, by the standard deviation is not similar. The one month has the lowest value in terms of the mean, minimum, maximum and the three expressed quantiles above. Next, the six month is the second lowest of the same metrics followed by the 1 year, 10 year and 30 year. As you increase the length of the loan, rates ascend, but vary differently as the dissimilarity in the standard deviation demonstrates the only exception of the common theme. The standard deviation is the highest among the 1 year, followed by the 6 month, then oddly the 1 month and 30 year are similar at 0.35, followed lastly by the 10 year. The box plot provides an illustration of the similarities with the boxes in each of the maturity points. The top of each box is the third quartile, the bottom is the first quartile and the green line in between is the median. The horizontal dash line at the very top above the boxes is the maximum and the bottom dash line is the minimum.

If you inspect the variation differences a bit more, then you will see there is no ascension of daily yield curve rate data when you difference the maximum and the minimum, and then separately again with the quantiles. The 1 month rates has the lowest difference in maximum and minimum, followed by the 6 month, but then the 1 year of 1.67 is more than the 10 year of 1.64. There is also no ascension when you difference the 75% and the 25% quantiles as the 1 year of 0.62 is the largest difference amount.



Secondly, to address clustering, there is a common theme across the time horizon of the data set. If you go back to page of 1, then you will see eleven column maturity points from 1 month to 30 years. We will see that both ends of the daily yield curve, cluster together. To demonstrate, let’s first assign the column maturity points by slicing the data frame.

# In[12]: one\_month, six\_month, one\_year

, ten\_year, thirty\_year =

df.loc[:,'1 mo'], df.loc[:,'6 mo'], df.loc[:,'1 yr']

, df.loc[:,'10 yr'], df.loc[:,'30 yr']

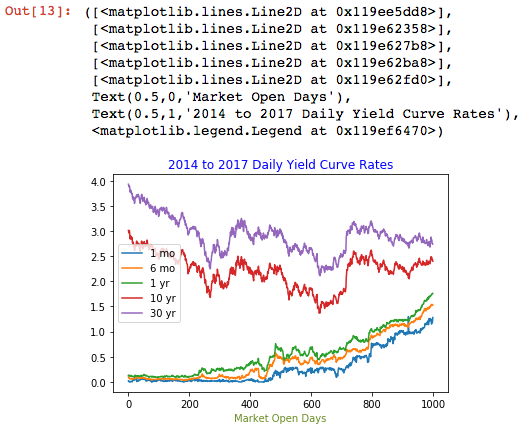
# In[13]: plt.plot(one\_month), plt.plot(six\_month), plt.plot(one\_year)

,plt.plot(ten\_year), plt.plot(thirty\_year)

, plt.xlabel('Market Open Days',color='olivedrab')

, plt.title('2014 to 2017 Daily Yield Curve Rates', color='blue')

,plt.legend()



From the line plot above we can see that the shorter maturity points of 1 month, 6 month and 1 year cluster together and differently from the longer maturity points of the 10 year and the 30 year. Vice versa on the longer maturity points, i.e. the 10 year and the 30 year clustering together and differing from the shorter maturity points. To illustrate another way, we can use a histogram that shows the overlapping of the dual clustering theme.

# In[14]: plt.hist(one\_month, histtype='step', label='1 Month')

,plt.hist(six\_month, histtype='step',label='6 Month')

,plt.hist(one\_year, histtype='step',label='1 Year')

,plt.hist(ten\_year, histtype='step',label='10 Year')

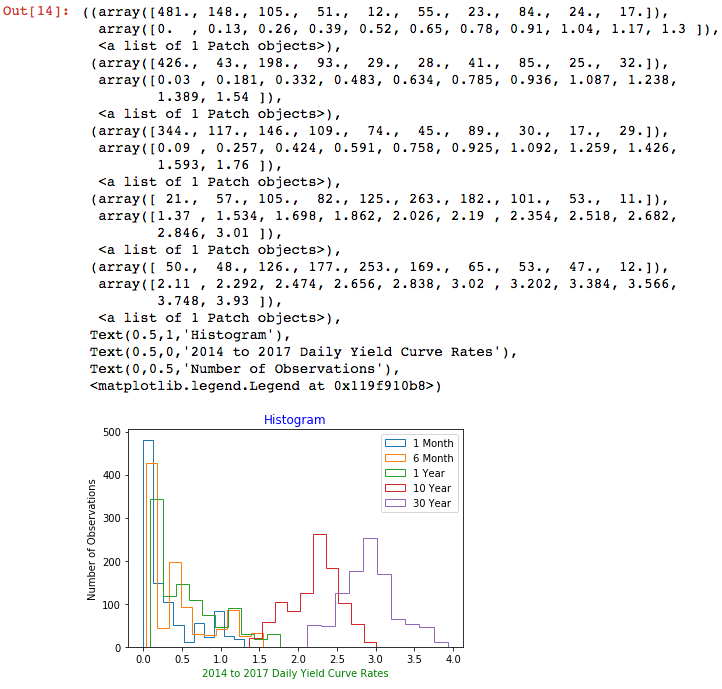
, plt.hist(thirty\_year, histtype='step',label='30 Year')

,plt.title('Histogram',color='blue')

, plt.xlabel('2014 to 2017 Daily Yield Curve Rates',color='green')

,plt.ylabel('Number of Observations')

,plt.legend()

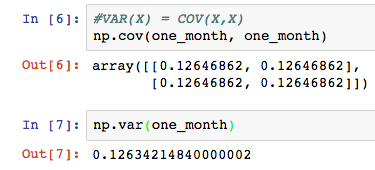


The shorter maturities of 1 month, 6 month and 1 year are all near zero on the left of the overlapping histograms as the longer maturities are away from daily zero yield curve rates.

Lastly, to address the last analytical question of how each of the maturites vary from each other, we look into covariance. To build a case, we begin with expected value leading to mean average, variance and then covariance. If we approach Daily Yield Curve Rates as random values, then buyers and sellers in the rate market have an expectation expressed as an expected value, i.e . All bids and offers in the rate market are equally likely as if it is a flip of a coin so a probability, , can be all be summed to be equal to one. If we take the expected value as the mean like on page 4, then the arithmetic mean can be calculated by the numpy mean function.

In turn, the values that vary around the mean such as variance are the calculated average squared difference of each value from the expected value or mean. To put it another way, the expected squared difference from the expected value or . Since individual probabilities are equal to each assigned individual value, then we can drop the probabilities of an equally weighted distribution to calculate the variance.

To understand how the column maturity point vectors vary with each other, the simplest approach to calculate covariance is comparing to itself, such as the one dimensional array of the one month maturity point. The covariance equals 0.126, just like the variance.



As we apply covariances to different maturity point vectors, then we will be able to see a pattern of clustering data versus the non - clustering.

#assigning covariances to short maturity cluster

In[8]: j, k, l = np.cov(one\_month,six\_month), np.cov(one\_month,one\_year), np.cov(six\_month,one\_year)

#assigning covariance to longer maturity cluster

In[9]: m = np.cov(ten\_year,thirty\_year)

#all maturity vectors that vary differently, i.e. not clustering

In[10]: t, u, v, w, x, y = np.cov(one\_month, ten\_year), np.cov(one\_month, thirty\_year),np.cov(six\_month, ten\_year), np.cov(six\_month, thirty\_year), np.cov(one\_year, ten\_year), np.cov(one\_year, thirty\_year)

In[11]: plt.scatter(j[:1], j[-1:],label='1 month 6 month', color='blue')

, plt.scatter(k[:1], k[-1:],label='1 month 1 year', color='magenta')

, plt.scatter(l[:1], l[-1:],label='6 month 1 year', color='red')

, plt.scatter(m[:1],m[-1:], label='10 year 30 year', color= 'black',marker='\*')

,plt.scatter(t[:1],t[-1:], label ='1 month 10 year', color='magenta',marker='x')

, plt.scatter(u[:1],u[-1:], label = '1 month 30 year', color='blue', marker='x')

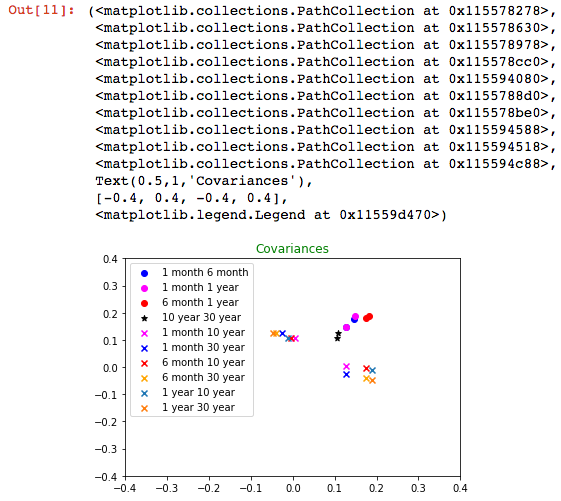
,plt.scatter(v[:1],v[-1:], label = '6 month 10 year', color='red', marker='x')

,plt.scatter(w[:1],w[-1:], label = '6 month 30 year', color='orange',marker='x')

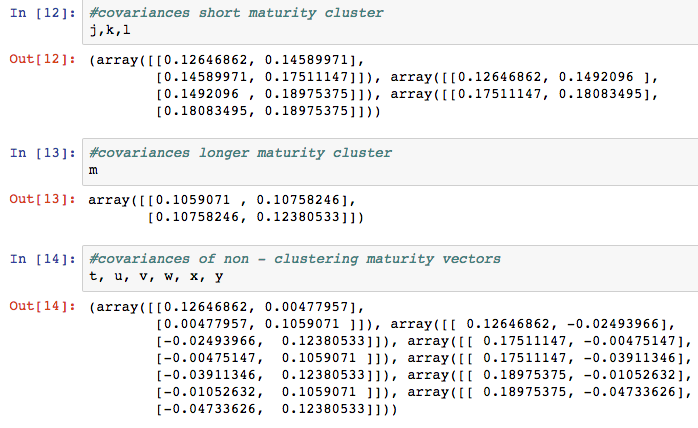
,plt.scatter(x[:1],x[-1:], label = '1 year 10 year', marker='x')

,plt.scatter(y[:1],y[-1:], label='1 year 30 year', marker = 'x') ,plt.title('Covariances',color='green')

,plt.axis([-0.40,0.40,-0.40,0.40]),plt.legend()



The shorter maturity cluster of 1 month, 6 month and 1 year have covariances with round dots, followed by the longer maturity cluster covariance of 10 year and 30 year with black stars. The clusters have a pattern of covariances from the bottom left to the top right on the scatter plot, which is the opposite of the non - clustering covariances. All of the covariances with an x shaped marker do not cluster and have a trend of upper left to lower right. The first variable of the non - clustering covariance are the shorter maturities of 1 month, 6 month and 1 year, followed by the second variable of the longer maturities of 10 year and 30 year.



Altogether, the selection of the Daily Yield Curve Data have both similarities and differences.

Maybe it is too early, but principal component analysis could be a viable candidate for my project as a statistical technique, but this is not my final answer for now in May of 2018. In conclusion, I look forward to demonstrate my net worth in the coming months . . .

John Foxworthy

Los Angeles, California