Lecture 1: Overview

1 Time Series, Returns, and Data Visualization

A sequence of observations taken over time is called a *time series*. The frequency of the data often depends on the specific financial application. For instance, we can have daily data (open or close stock price data), annual data (inflation), quarterly data (unemployment), millisecond data (tick by tick stock prices).

Consider returns from investing in stocks. If you buy a stock at time t at price S_t, and the price of the stock at time $t+1$ is S_{t+1}, then the return on the investment is $\frac{S_{t+1}-S_t}{S_t}$. Table 1 displays the returns on the IBM stock price data from November, 30, 2016 through December 30, 2016. The Excel spreadsheet on Courseworks reports a longer sequence of returns for IBM stock price data. We can also do a scatterplot of the returns using the following set of R instructions:

Table 1: IBM Returns

Date	Returns
12/30/2016	-0.003
12/29/2016	-0.002
12/28/2016	0.008
12/27/2016	-0.002
12/23/2016	0
12/22/2016	0.002
12/21/2016	-0.007
12/20/2016	0.007
12/19/2016	-0.004
12/16/2016	0.013
12/15/2016	-0.006
12/14/2016	0.002
12/13/2016	-0.016
12/12/2016	0.006
12/9/2016	-0.009
12/8/2016	-0.002
12/7/2016	-0.026
12/6/2016	-0.003
12/5/2016	0.004
12/2/2016	-0.012
12/1/2016	0.019
11/30/2016	0.008

```
df = read.table("IBMreturns.csv", header = TRUE, sep = ",");
df$Date ← as.Date(df$Date,"%m/%d/%Y");
ggplot(data = df, aes(Date,Open)) + geom_point() + labs(x="Date",y="Returns",title="IBM Returns")).
```

The resulting output is provided in Figure 1. Similarly, we can plot the time series of Microsoft

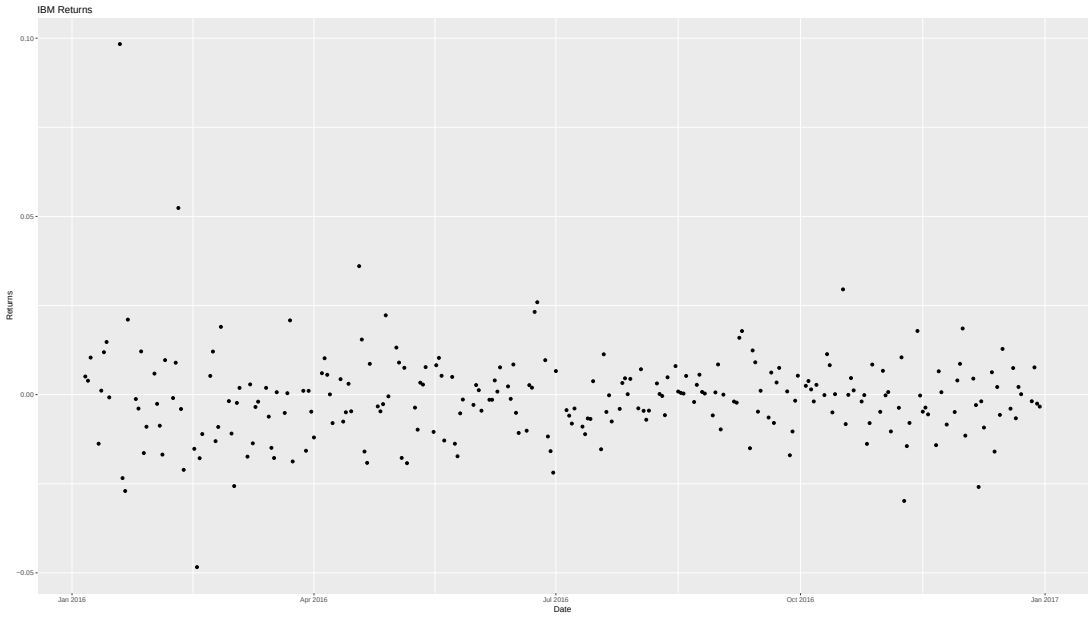

Figure 1: Scatterplot of IBM returns, computed using the time series of returns from January 6, 2016 through December 30, 2016

returns (msdf = read.table("msftreturns.csv", header = TRUE, sep = ",")). We use the same instructions as above for plotting, and plot the data in Figure 2.

An alternative means of representing data are histograms. A frequency histogram for both IBM and Microsoft returns is reported in Figure 3. The corresponding R instructions to plot histograms are given below.

```
hist(df$Open, breaks= 50, main="IBM");
hist(msdf$Open, breaks= 50, main="MSFT").
```

The histograms suggest that both time series seem to be centered around zero.

1.1 Summarizing data with a single numeric variable

In the previous section, we have analyzed data only visually through graphs. Suppose we are now interested in having a numerical summary of the data rather than just a graphical representation.

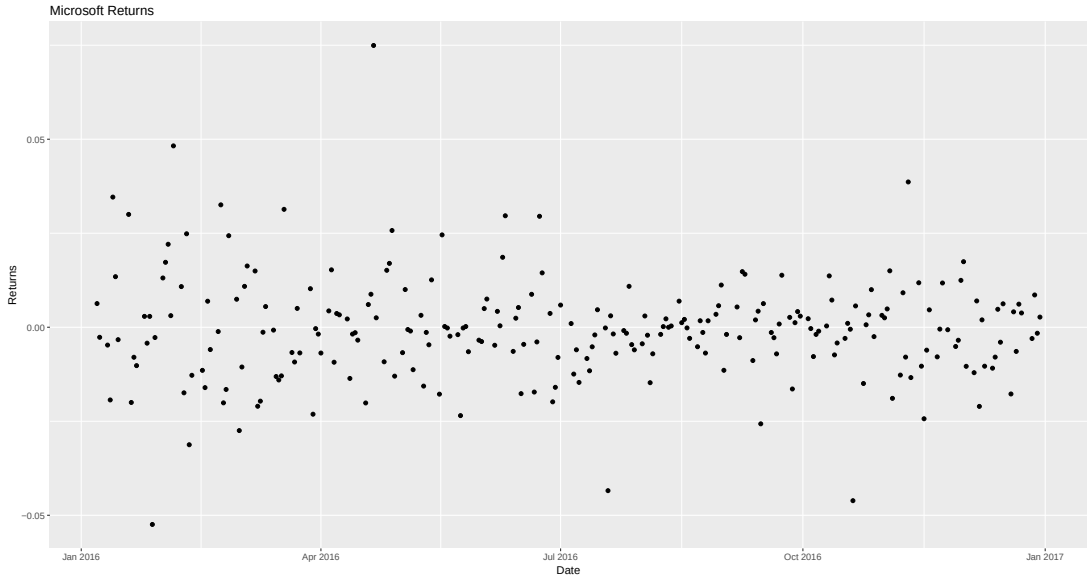

Figure 2: Scatterplot of Microsoft returns, computed using the time series of returns from January 6, 2016 through December 30, 2016

Two important summary statistics of any data set are: (1) the average value, and (2) how spread out the values of this variable are.

1.1.1 Sample Mean

A basic statistics which allows to summarize a time series with a single number is the sample mean. The sample mean of the Microsoft returns is –0.0006082268 (R instruction is mean(msdf$Open)). The sample mean of the IBM returns is -0.0007897998 (R instruction is mean(df$Open)). Despite the difference between these sample means being small, there is still a difference. It was hard to quantify this difference from the plots, because the difference is small compared to the variation in the data.

1.2 The Median

The median is a measure of central tendency. Arrange the data in ascending order. After this ranking, the middle number in the dataset is the median. If there is an even number of data points, then the median is the average of the two middle numbers. For example, consider the series 1,4,7,8,10. Then the median is 7. Then, consider the series 1,4,5,8. Then the median is 4.5.

Unlike the mean, the median is not affected by outliers. Next, let us see briefly why this is the case. Consider the dataset: 1,4,7,8,10. Then the median is 7, and the mean is 6. Consider the dataset 1,4,7,8,100. Then the median is 7, and the mean is 24. It is always good to report the median when there are extreme values in the data. For instance, think about the US household income in 2004. The median was $44,334. Of course, Bill Gates' income was around $50 billions

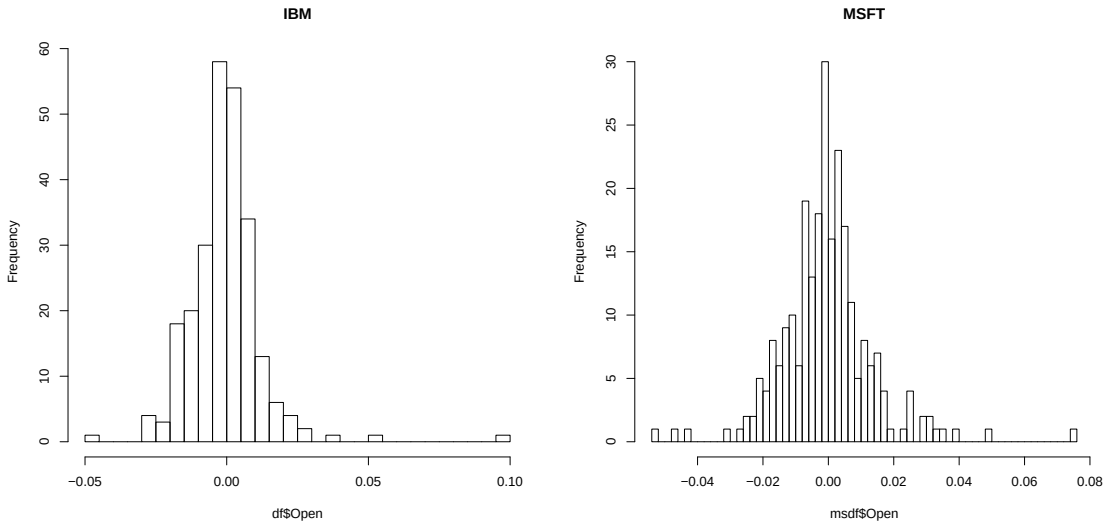

Figure 3: Histogram of IBM and Microsoft returns, computed using the time series of returns from January 6, 2016 through December 30, 2016

USD. So, Bill Gates' income is an outlier. The R command to compute the median of a data vector x is median(x). For instance, the median of MSFT returns, is median(msdf$Open).

1.3 Sample Variance

The sample mean provides a point estimate for the average entry of the vector x. Next, we want a numerical measure of variation or spread. Such a measure can be computed by averaging the differences $x_i - \bar{x}$, capturing the deviation of each entry from the sample mean. We need to be careful, and cannot just sum up these differences because the negative differences would offset the positive differences. To avoid this problem, we can square the differences and thus obtain the measure $\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$. For technical reasons (you will see why in the homework problem), the sample variance of the data is defined to be

$$s_x^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2,$$

i.e. dividing by $n-1$ and not by n. If n is large, there is little difference in dividing by n or by $n-1$. It is also helpful to have a measure of spread which is in the same units as the data. This is given by the sample standard deviation: $s_x = \sqrt{s_x^2}$. The units of the standard deviation are the same as those of the original data.

For our time series of IBM and Microsoft returns, a visual inspection appears to indicate that IBM returns are slightly less spread out than Microsoft returns. Computing the standard deviation for both series of returns (R instructions are sd(msdf$Open, na.rm=T), and sd(df$Open, na.rm=T),

respectively for Microsoft and IBM returns, we obtain respectively 0.01384712 and 0.01269863.[1] Hence, IBM returns are indeed less spread out than Microsoft returns.

1.3.1 Covariance and Correlation

The mean and standard deviation help us summarize numbers which are measurements of just one thing. A fundamentally different question is how one time series relates to another. For instance, how Microsoft returns relate to the DJIA returns? Are the returns on a mutual fund related to market returns? The relationship between two data vectors can be captured by the covariance and correlation. The sample covariance between x and y is

$$s_{xy} := \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x}) (y_i - \bar{y}) .$$

A positive covariance means that when one variable is above its average, the other one tends to be above its own average as well, and viceversa, when one variable is below its average, the other one tends to be below its own average as well. They move up and down together. A negative covariance means that when one moves up, the other tends to move down, i.e. they move in opposite directions. The R command to compute the covariance between Microsoft and IBM returns is cov(msdf$Open, df$Open). If we want to remove missing values, then we can do the following completedata <- na.omit(cbind(msdf$Open, df$Open)), and then cov(completedata)

The sample correlation between x and y is $r_{xy} = \frac{s_{xy}}{s_x s_y}$, so the correlation is just the covariance of the two vectors divided by the product of their standard deviations. Let us state some facts: the closer r_{xy} is to 1, the stronger the linear relationship is with a positive slope. That is, the largest values of y tend to be associated with the largest values of x (and vice-versa). The closer r_{xy} is to -1, the stronger the linear relationship is with a negative slope. That is, large values of y tend to be associated with small values of x (and vice-versa). The command to compute the Pearson correlation is cor(msdf$Open, df$Open).

[1] na.rm = TRUE indicates that potential missing values in the time series need to be ignored in the computation of the statistics, which in this case is the standard deviation.