Homework #5

Le Wang

Instruction: Do all the following empirical exercises using R. Turn in your R markdown file with answers and supporting tables and graphs, if any. Refer to the R output whenever appropriate when discussing your results. It does not matter which method you use to plot a time series variable.

Question 1.[Learn How to Calculate Value at Risk (VaR)]

1. Go to Yahoo Finance and download the historical stock prices data for Google. In what follows, we will focus on the adjusted prices (the 6th column of the data)
2. Construct a new variable, daily returns (based on the formulae $\frac{(P_t - P_{t-1})}{P_{t-1}}$).
3. Based on the distribution for the daily returns, calculate the VaR (value at risk) for daily returns at 95 percent confidence level.
4. Based on your results in (3), calculate the VaR in the next day for $10,000 at 95 confidence level. Interpret the value that you obtain.
5. Cacculate the VaR in the next day for $10,000 at 99 percent confidence level. Interpret the VaR that you obtain.
6. Suppose that your client does not tell you what risk he or she is willing to tolerate. What you can do is to calculate the VaR at every confidence level, say, from 1 to 99 percent confidence level and plot it. This is called **cumulative risk profile**.
7. Suppose that there was a fundamental change in Google's operations in 2012 (change of the CEO perhaps), which in turn led to a systematic change in the distribution of daily returns. So, we will focus on the post-2012 data. Again, repeat the analysis above for this subset of the data and report the VaR in the next day for $10,000 at 95 confidence level. Interpret the value that you obtain.
8. Calculate the VaR in the next **week** for $10,000 at 95 confidence level. Interpret the value that you obtain. Use two different approaches
   a. Use the built-in command in `R`
   b. Use the T rule discussed in class.

Question 2. [Learn How to Obtain and Program Certain Characteristics]

As Alex rightly noted after class, we took a hard way to calculate sample averages. This is because we would like to know how we can derive sample estimators based on a theoretical concept, when there is no way to do so. Let's look at one example. Suppose that we are interested in the following statistic

$$\sum_{j=1}^{J} x^j \frac{[f(x^j)]^2}{\sum_{j=1}^{J}[f(x^j)]^2}$$

where x^j is a distinct value of variable X, and $f(x^j)$ is the corresponding probability.In total, we have J distinct values. This is a weighted mean of x, but with the weight being much larger for those with larger probabilities. We cannot simply estimate this number by taking the simplistic approach to sum up all the values and then divide it by N, the number of observations. But we can use the definition above to program it. Let me walk you through this process. You will see that we will put together to use many different commands that we learned before.

Run the following code in your R to generate the sample data first.

```r
set.seed(123456)
x <- sample(1:6, 1000, replace = TRUE)
```

1. Let's find out what the unique values are by using `unique()`, and call it `x.unique`

```r
x.unique <-
```

2. Lets' sort our unique values, and call it `x.sorted`

```r
x.sorted <-
```

3. Calculate the probability mass function (then we automatically have $f(x^j)$ for every x^j), and call it `x.pmf`.

```r
x.pmf <-
```

4. Generate a new vector that is $[f(x^j)]^2$, the square of each probability, and call it `x.pmf2`.

```r
x.pmf2 <-
```

5. Calculate the sum $\sum_{j=1}^{J}[f(x^j)]^2$ in the denominator, and call it `x.pmf2sum`

```r
x.pmf2sum <-
```

6. Now sum up the (element-wise) product of every distinct value, x^j and $\frac{[f(x^j)]^2}{\sum_{j=1}^{J}[f(x^j)]^2}$