Problem 5: Computational Finance - Modelling Stock prices

Deepmalya Dutta

Following piece of code download the prices of TCS since 2007

```
library(quantmod)
```

```
## Loading required package: xts
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
## Loading required package: TTR
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
getSymbols('TCS.NS')
```

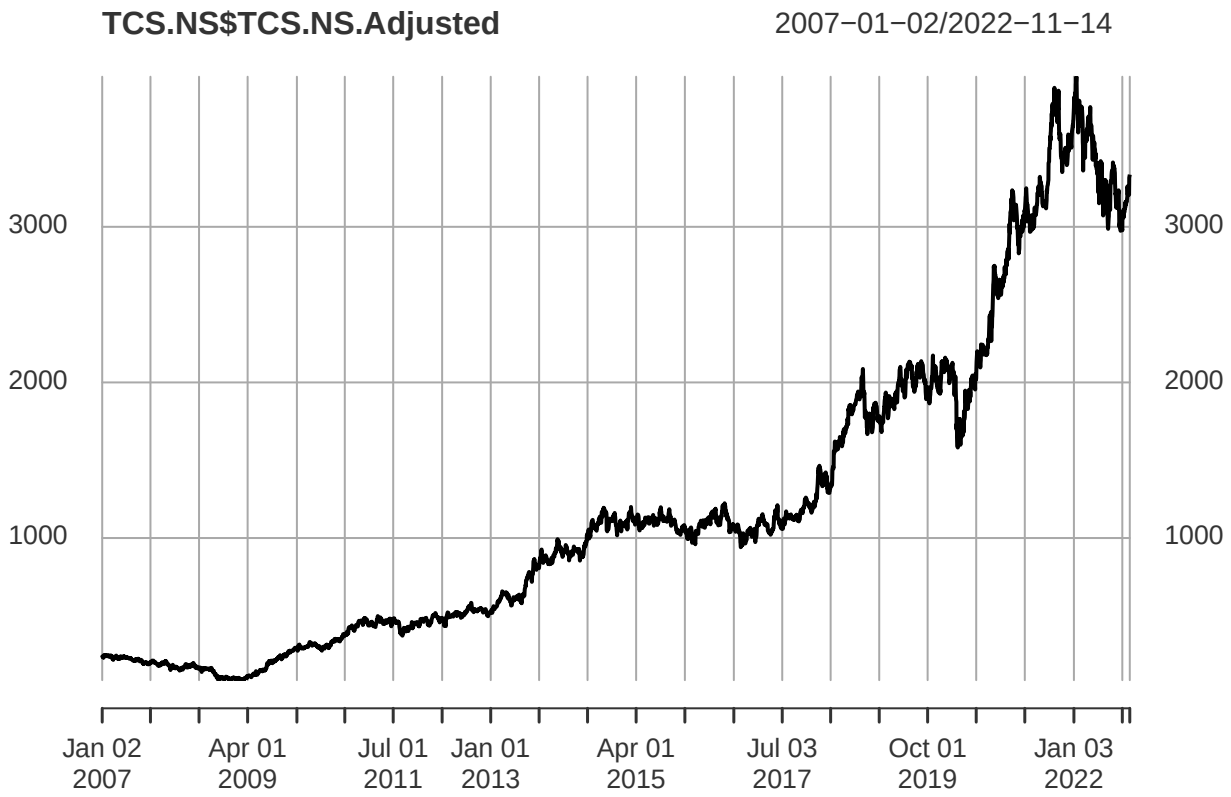
```
## [1] "TCS.NS"
```

```
tail(TCS.NS)
```

```
##            TCS.NS.Open TCS.NS.High TCS.NS.Low TCS.NS.Close TCS.NS.Volume
## 2022-11-04      3217.0     3220.05    3166.15      3217.40       1464013
## 2022-11-07      3229.0     3242.80    3195.10      3233.70       1474498
## 2022-11-09      3249.8     3249.80    3201.65      3216.05       1162267
## 2022-11-10      3170.0     3225.00    3170.00      3205.65       1573092
## 2022-11-11      3269.6     3341.60    3255.05      3315.95       3265394
## 2022-11-14      3324.0     3349.00    3309.00      3335.50       1342074
##            TCS.NS.Adjusted
## 2022-11-04         3217.40
## 2022-11-07         3233.70
## 2022-11-09         3216.05
## 2022-11-10         3205.65
## 2022-11-11         3315.95
## 2022-11-14         3335.50
```

Plot the adjusted close prices of TCS

```
plot(TCS.NS$TCS.NS.Adjusted)
```



Download the data of market index Nifty50. The Nifty 50 index indicates how the over all market has done over the similar period.

```
getSymbols('^NSEI')
```

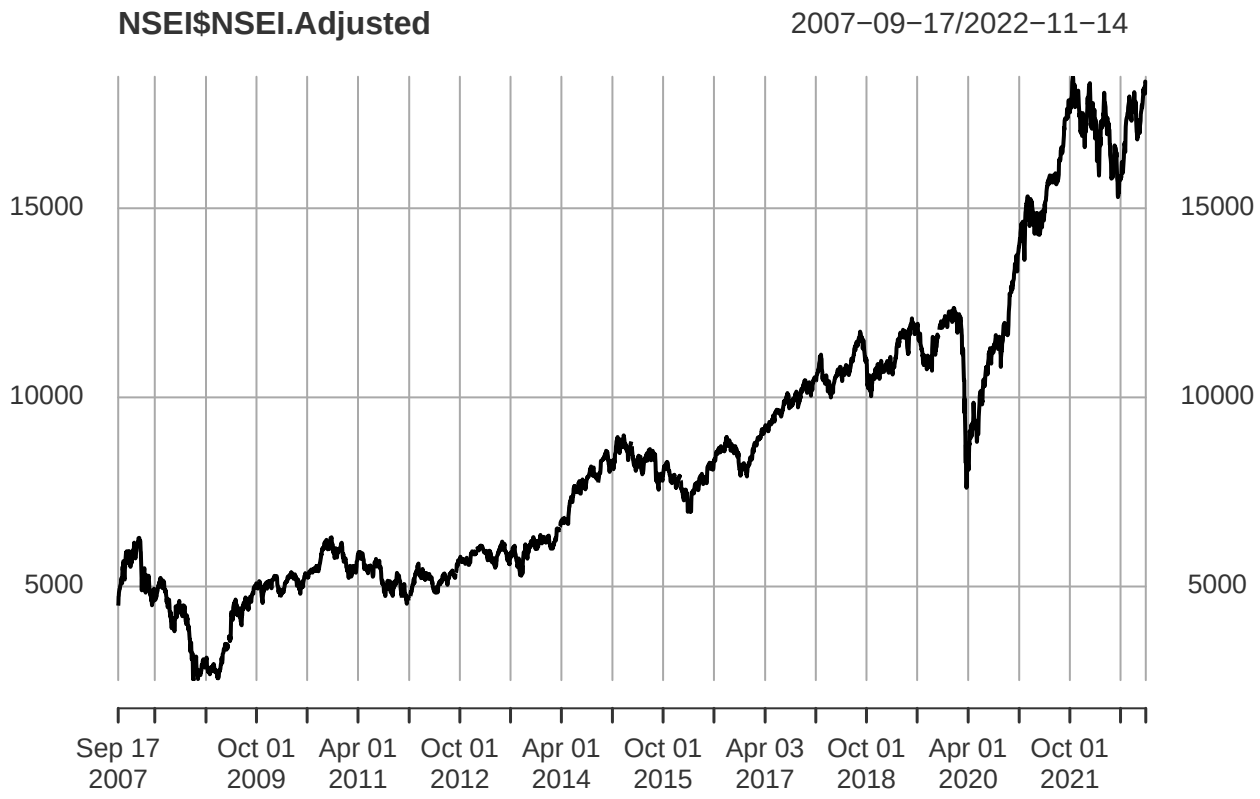
```
## [1] "^NSEI"
```

```
tail(NSEI)
```

```
##            NSEI.Open NSEI.High NSEI.Low NSEI.Close NSEI.Volume NSEI.Adjusted
## 2022-11-04  18053.40  18135.10 18017.15   18117.15      267900      18117.15
## 2022-11-07  18211.75  18255.50 18064.75   18202.80      314800      18202.80
## 2022-11-09  18288.25  18296.40 18117.50   18157.00      307200      18157.00
## 2022-11-10  18044.35  18103.10 17969.40   18028.20      256500      18028.20
## 2022-11-11  18272.35  18362.30 18259.35   18349.70      378500      18349.70
## 2022-11-14  18376.40  18399.45 18311.40   18329.15      301400      18329.15
```

Plot the adjusted close value of Nifty50

```
plot(NSEI$NSEI.Adjusted)
```



Log-Return

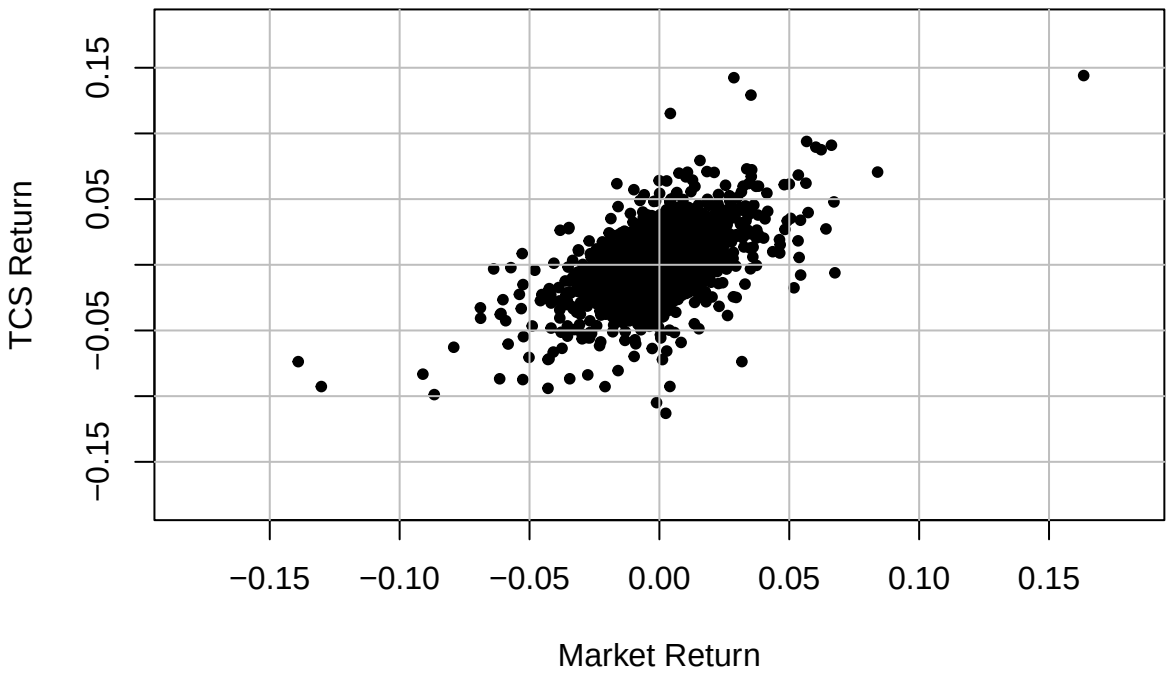
We calculate the daily log-return, where log-return is defined as

$$r_t = \log(P_t) - \log(P_{t-1}) = \Delta \log(P_t),$$

where P_t is the closing price of the stock on t^{th} day.

```
TCS_rt = diff(log(TCS.NS$TCS.NS.Adjusted))
Nifty_rt = diff(log(NSEI$NSEI.Adjusted))
retrn = cbind.xts(TCS_rt,Nifty_rt)
retrn = na.omit(data.frame(retrn))

plot(retrn$NSEI.Adjusted,retrn$TCS.NS.Adjusted
     ,pch=20
     ,xlab='Market Return'
     ,ylab='TCS Return'
     ,xlim=c(-0.18,0.18)
     ,ylim=c(-0.18,0.18))
grid(col='grey',lty=1)
```



- Consider the following model:

$$r_t^{TCS} = \alpha + \beta r_t^{Nifty} + \varepsilon,$$

where $\mathbb{E}(\varepsilon) = 0$ and $\mathbb{V}ar(\varepsilon) = \sigma^2$.

1. Estimate the parameters of the models $\theta = (\alpha, \beta, \sigma)$ using the method of moments type plug-in estimator discussed in the class.

```
m_x <- mean(retrn$NSEI.Adjusted)
m_y <- mean(retrn$TCS.NS.Adjusted)
s_x <- sd(retrn$NSEI.Adjusted)
s_y <- sd(retrn$TCS.NS.Adjusted)
r <- cor(retrn$TCS.NS.Adjusted, retrn$NSEI.Adjusted)

a0 <- m_y -r*s_y/s_x*m_x
b0 <- r*s_y/s_x

return <- retrn
return$pred_TCS <- a0 + b0*return$NSEI.Adjusted
e <- return$TCS.NS.Adjusted - return$pred_TCS
sigma0 <- sd(e)

print(c("The parameters of the model estimated using method of moments are:",
↪  paste("alpha = ", a0,", beta = ", b0, ", sigma =", sigma0)))
```

```
## [1] "The parameters of the model estimated using method of moments are:"
```

```
## [2] "alpha =  0.000462824161300306 , beta =  0.74368401407465 , sigma = 0.0161846571879184"
```

2. Estimate the parameters using the `lm` built-in function of `R`. Note that `lm` using the OLS method.

```
ols <- lm(TCS.NS.Adjusted~NSEI.Adjusted,data = retrn)
a1 <- ols$coefficients[1]
b1 <- ols$coefficients[2]
sigma1 <- sd(ols$residuals)

print(c("The parameters of the model estimated using `lm` built-in function of `R` are:",
↪  paste("alpha = ", a1,", beta = ", b1, ", sigma =", sigma1)))
```

```
## [1] "The parameters of the model estimated using `lm` built-in function of `R` are:"
## [2] "alpha =  0.000462824161300303 , beta =  0.743684014074649 , sigma = 0.0161846571879184"
```

3. Fill-up the following table

Parameters	Method of Moments	OLS
α		
β		
σ		

Solution:

Parameters	Method of Moments	OLS
α	0.000462823035007132	0.000462823035007131
β	0.743684314730296	0.743684314730293
σ	0.0161846620846538	0.0161846620846538

4. If the current value of Nifty is 18000 and it goes up to 18200. The current value of TCS is Rs. 3200/-. How much you can expect TCS price to go up?

```
nifty_current <- 18000
nifty_future <- 18200
TCS_current <- 3200
nifty_return <- log(nifty_future) - log(nifty_current)
TCS_pred_return <- predict(ols, data.frame(NSEI.Adjusted = c(nifty_return)))
print(paste("The TCS price would go up to", round(exp(TCS_pred_return)*TCS_current)))
```

```
## [1] "The TCS price would go up to 3228"
```