Time Series HW3

40/40

B082040005 高念慈

2023-03-10

1. 20

Consider the daily stock returns of Apple from January 2020 to February 2023. The price data can be obtained by using R package quantmod.

```
# from 正常，to 少一天
getSymbols("AAPL",from="2020-01-01",to='2023-03-01')
```

```
## [1] "AAPL"
```

```
head(AAPL)
```

```
##            AAPL.Open AAPL.High AAPL.Low AAPL.Close AAPL.Volume AAPL.Adjusted
## 2020-01-02   74.0600   75.1500  73.7975    75.0875   135480400      73.44940
## 2020-01-03   74.2875   75.1450  74.1250    74.3575   146322800      72.73533
## 2020-01-06   73.4475   74.9900  73.1875    74.9500   118387200      73.31488
## 2020-01-07   74.9600   75.2250  74.3700    74.5975   108872000      72.97008
## 2020-01-08   74.2900   76.1100  74.2900    75.7975   132079200      74.14390
## 2020-01-09   76.8100   77.6075  76.5500    77.4075   170108400      75.71877
```

取出 adjust price

```
AAPL_adjust = AAPL$AAPL.Adjusted
```

a. Compute the sample mean, standard deviation, skewness and excess kurtosis of the log returns r_t.

- $r_t = ln(R_t + 1)$
- $r_t = ln(P_t) - ln(P_{t-1})$

Log Return : $r_t = ln(P_t) - ln(P_{t-1})$

```
AAPL_log_return = diff(log(AAPL_adjust))
head(AAPL_log_return)
```

```
##            AAPL.Adjusted
## 2020-01-02            NA
## 2020-01-03  -0.009769539
## 2020-01-06   0.007936367
## 2020-01-07  -0.004714027
## 2020-01-08   0.015958210
## 2020-01-09   0.021018349
```

- sample mean
- standard deviation
- skewness
- excess kurtosis

```
basicStats(AAPL_log_return)
```

```
##               AAPL.Adjusted
## nobs             795.000000
## NAs                1.000000
## Minimum           -0.137708
## Maximum            0.113158
## 1. Quartile       -0.010684
## 3. Quartile        0.014041
## Mean               0.000877
## Median             0.000746
## Sum                0.696621
## SE Mean            0.000815
## LCL Mean          -0.000722
## UCL Mean           0.002477
## Variance           0.000527
## Stdev              0.022961
## Skewness          -0.126218
## Kurtosis           4.122750
```

```
mean(AAPL_log_return,na.rm=T)      # 0.0008773567
```

```
## [1] 0.0008773565
```

```
sqrt(var(AAPL_log_return,na.rm=T)) # 0.02296092
```

```
##                 AAPL.Adjusted
## AAPL.Adjusted      0.02296093
```

```
skewness(AAPL_log_return,na.rm=T)  # -0.1262185
```

```
## [1] -0.1262181
## attr(,"method")
## [1] "moment"
```

```
kurtosis(AAPL_log_return,na.rm=T)  # 4.122718
```

```
## [1] 4.12275
## attr(,"method")
## [1] "excess"
```

b. Estimate the mean and standard deviation of the simple return R_t by assuming the log returns r_t follow a normal distribution $r_t \sim N(\mu, \sigma^2)$.

- $R_t = \frac{P_t - P_{t-1}}{P_{t-1}}$
- $R_t = e^{r_t} - 1$
- $r_t \sim N(\mu_{r_t}, \sigma^2_{r_t})$, then $e^{r_t} \sim lognormal(\mu_{r_t}, \sigma^2_{r_t})$
- $E(e^{r_t}) = e^{\mu_{r_t} + \frac{\sigma^2_{r_t}}{2}}$
- $E(e^{r_t} - 1) = E(R_t) = e^{\mu_{r_t} + \frac{\sigma^2_{r_t}}{2}} - 1$
- $Var(e^{r_t}) = e^{2\mu + \sigma^2}[e^{\sigma^2} - 1]$
- $Var(e^{r_t}) = Var(e^{r_t} - 1) = Var(R_t)$

```
mu = mean(AAPL_log_return,na.rm=T) # 0.0008773567
va = var(AAPL_log_return,na.rm=T)  # 0.000527204

# mean of the simple return
mean_R_t = exp(mu+va/2) - 1
mean_R_t                           # 0.00114161
```

```
##                AAPL.Adjusted
## AAPL.Adjusted     0.00114161
```

```
# standard deviation of the simple return
sd_R_t = sqrt(exp(2*mu+va)*(exp(va)-1))
sd_R_t                             # 0.02299017
```

```
##                AAPL.Adjusted
## AAPL.Adjusted     0.02299017
```

c. Compute the sample mean and the sample standard deviation of the simple return R_t. Compare the results of (b) and (c).

- $R_t = \frac{P_t - P_{t-1}}{P_{t-1}}$
- $R_t = e^{r_t} - 1$

simple return

```
AAPL_simple_return = exp(AAPL_log_return) - 1
head(AAPL_simple_return)
```

```
##            AAPL.Adjusted
## 2020-01-02            NA
## 2020-01-03  -0.009721972
## 2020-01-06   0.007967944
## 2020-01-07  -0.004702933
## 2020-01-08   0.016086222
## 2020-01-09   0.021240790
```

```
AAPL_simple_return2 = diff(AAPL_adjust)/lag(AAPL_adjust)  # lag 幫助往前除一天
head(AAPL_simple_return2)
```

```
##            AAPL.Adjusted
## 2020-01-02            NA
## 2020-01-03  -0.009721972
## 2020-01-06   0.007967944
## 2020-01-07  -0.004702933
## 2020-01-08   0.016086222
## 2020-01-09   0.021240790
```

the sample mean and the sample standard deviation

```
mean(AAPL_simple_return, na.rm = T)       # 0.00114107
```

```
## [1] 0.00114107
```

```
sqrt(var(AAPL_simple_return, na.rm = T))  # 0.02297127
```

```
##               AAPL.Adjusted
## AAPL.Adjusted    0.02297128
```

Estimate

- mean_R_t = 0.00114161
- sd_R_t = 0.02299017

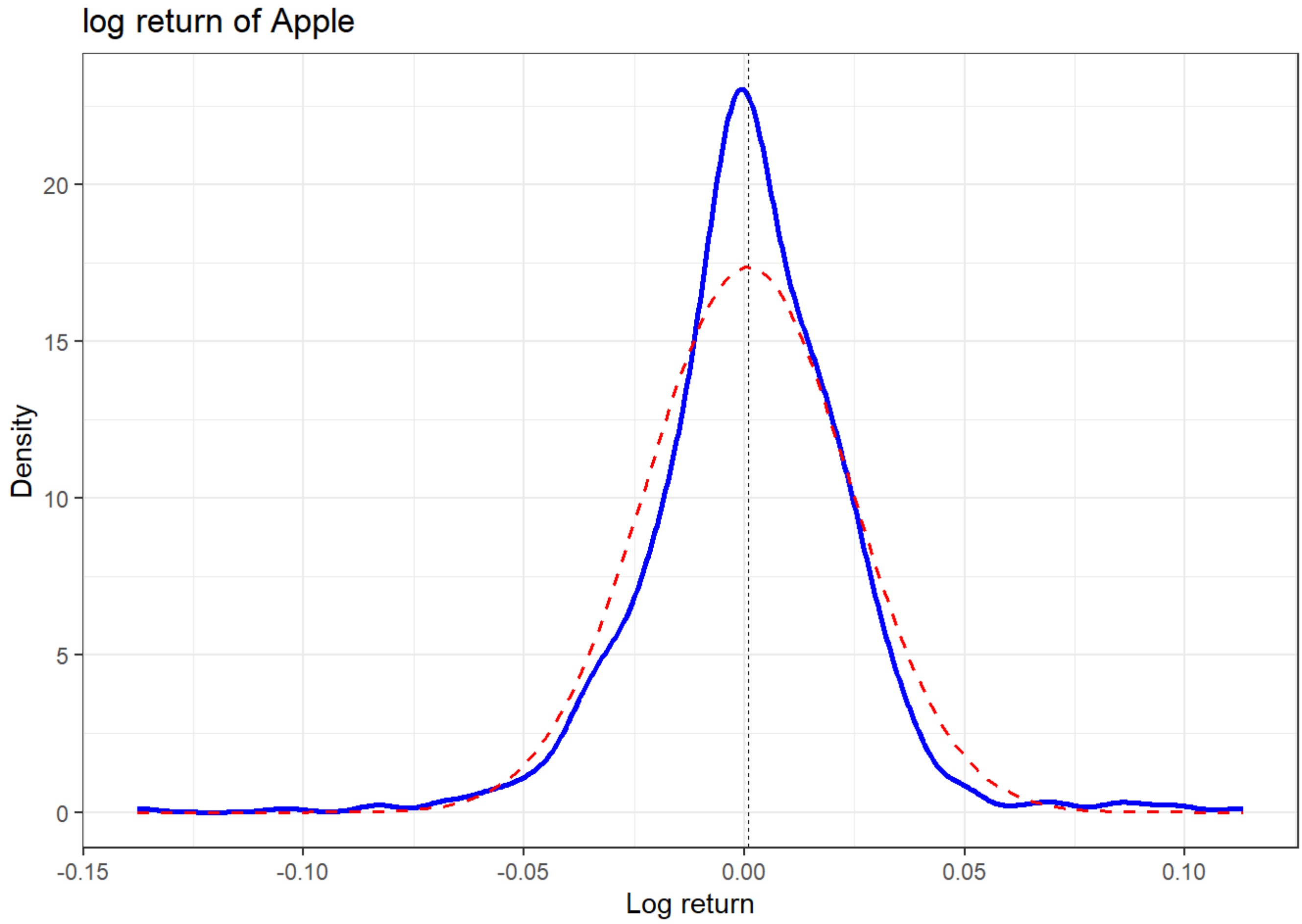
Compute

- mean = 0.00114107
- sd = 0.02297127
- 兩者結果幾乎一樣，r_t 可能真的符合常態假設

d. Find the kernel density estimator and normal density estimator
for the log return r_t and the simple return R_t respectively.
Compare the empirical kernel density and normal density for r_t and R_t
Plot the two estimated densities on the same graph.
(see Page 21 Figure 1.4 in the textbook)

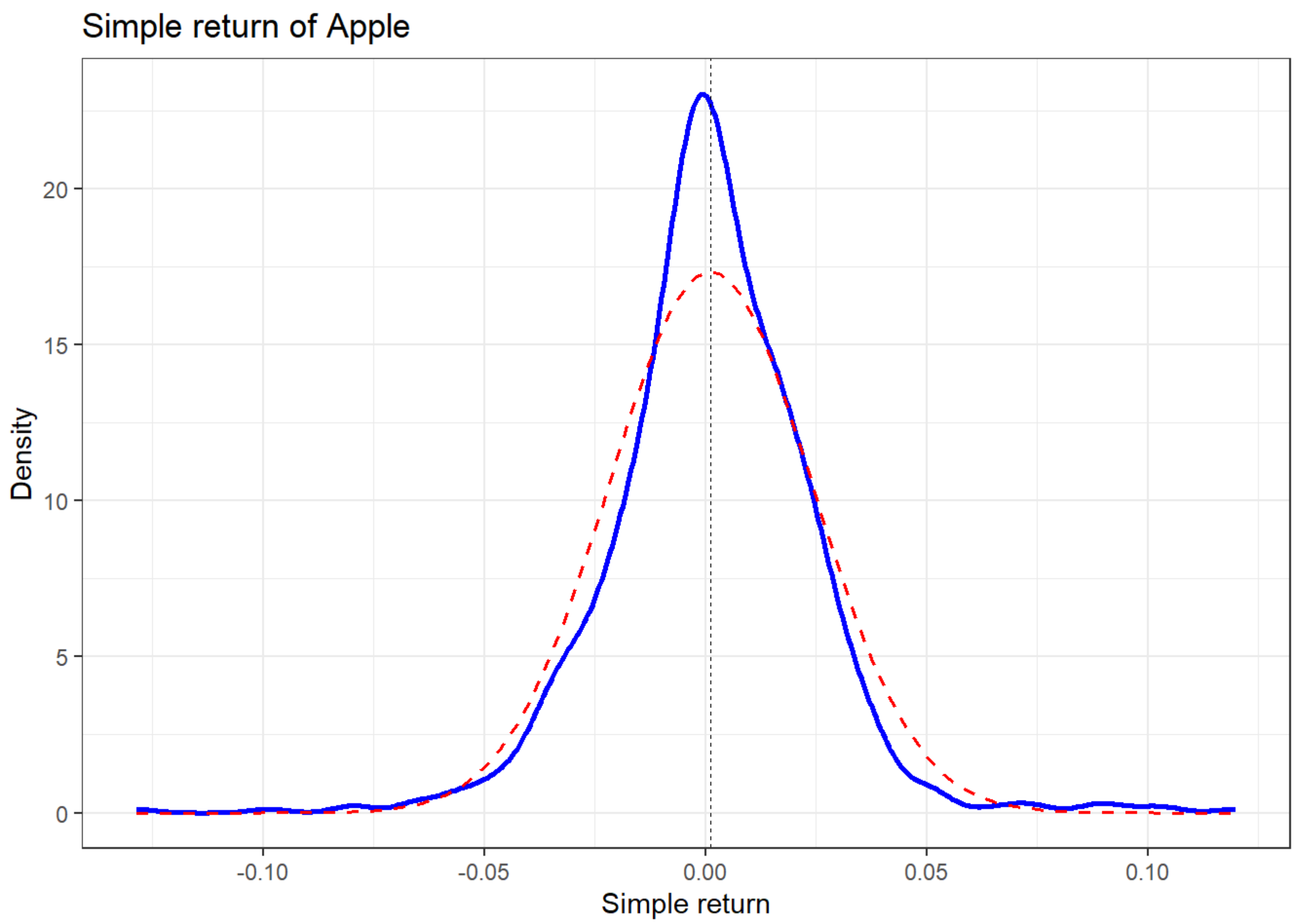
the log return

```
ggplot(na.omit(AAPL_log_return),aes(x=AAPL.Adjusted)) +
  geom_density(color = "blue",linewidth=1) +
  geom_vline(aes(xintercept = mu),color="black",linetype="dashed",linewidth=0.1) +
  theme_bw() +
  labs(title="log return of Apple",x="Log return",y="Density")+
  stat_function(fun = dnorm, args = list(mu, sqrt(va)),
                colour = "red",linewidth=0.6,linetype="dashed")
```



the simple return

```
ggplot(na.omit(AAPL_simple_return),aes(x=AAPL.Adjusted)) +
  geom_density(color = "blue",linewidth=1) +
  geom_vline(aes(xintercept = mean_R_t),color="black",linetype="dashed",linewidth=0.1) +
  theme_bw() +
  labs(title="Simple return of Apple",x="Simple return",y="Density")+
  stat_function(fun = dnorm, args = list(mean_R_t, sd_R_t),
                colour = "red",linewidth=0.6,linetype="dashed")
```



Compare the empirical kernel density and normal density for r_t and R_t

常態假設對 APPLE 股票的日回報是有問題的。
經驗密度函數在其均值附近有一個更高的峰值，
但尾部比相應的正態分佈更厚。

- **In other words, the empirical density function is taller and skinnier**
- **but with a wider support than the corresponding normal density**

2.

Consider the daily stock returns of Taiwan Semiconductor Manufacturing from January 2020 to February 2023.
The price data can be obtained by using R package quantmod.

```
# from 正常，to 少一天
getSymbols("TSM",from="2020-01-01",to='2023-03-01')
```

```
## [1] "TSM"
```

```
head(TSM)
```

```
##            TSM.Open TSM.High TSM.Low TSM.Close TSM.Volume TSM.Adjusted
## 2020-01-02    59.60    60.12   59.60     60.04    8432600     56.35028
## 2020-01-03    58.97    58.98   58.04     58.06   10546400     54.49196
## 2020-01-06    57.60    57.69   57.13     57.39    8897200     53.86313
## 2020-01-07    57.45    58.60   56.74     58.32    7444300     54.73599
## 2020-01-08    58.19    58.98   58.11     58.75    5381500     55.13955
## 2020-01-09    59.69    59.71   58.70     59.23    5112700     55.59006
```

取出 adjust price

```
TSM_adjust = TSM$TSM.Adjusted
```

a. Compute the sample mean, standard deviation, skewness and excess kurtosis of the log returns r_t.

- $r_t = ln(R_t + 1)$
- $r_t = ln(P_t) - ln(P_{t-1})$

Log Return : $r_t = ln(P_t) - ln(P_{t-1})$

```
TSM_log_return = diff(log(TSM_adjust))
head(TSM_log_return)
```

```
##            TSM.Adjusted
## 2020-01-02           NA
## 2020-01-03 -0.033534023
## 2020-01-06 -0.011606968
## 2020-01-07  0.016075191
## 2020-01-08  0.007345872
## 2020-01-09  0.008137055
```

- sample mean
- standard deviation
- skewness
- excess kurtosis

```
basicStats(TSM_log_return)
```

```
##                TSM.Adjusted
## nobs             795.000000
## NAs                1.000000
## Minimum           -0.151219
## Maximum            0.119135
## 1. Quartile       -0.014314
## 3. Quartile        0.014292
## Mean               0.000548
## Median            -0.000179
## Sum                0.435125
## SE Mean            0.000889
## LCL Mean          -0.001197
## UCL Mean           0.002293
## Variance           0.000628
## Stdev              0.025053
## Skewness           0.058385
## Kurtosis           2.890400
```

```
mean(TSM_log_return,na.rm=T)      # 0.0005480166
```

```
## [1] 0.0005480166
```

```
sqrt(var(TSM_log_return,na.rm=T)) # 0.02505303
```

```
##                TSM.Adjusted
## TSM.Adjusted   0.02505303
```

```
skewness(TSM_log_return,na.rm=T)  # 0.05838448
```

```
## [1] 0.05838536
## attr(,"method")
## [1] "moment"
```

```
kurtosis(TSM_log_return,na.rm=T)  # 2.890411
```

```
## [1] 2.8904
## attr(,"method")
## [1] "excess"
```

b. Estimate the mean and standard deviation of the simple return R_t by assuming the log returns r_t follow a normal distribution $r_t\ N(\mu, \sigma^2)$.

- $R_t = \frac{P_t - P_{t-1}}{P_{t-1}}$
- $R_t = e^{rt} - 1$
- $r_t\ N(\mu_{rt}, \sigma^2_{rt})$ · then $e^{rt}\ lognormal(\mu_{rt}, \sigma^2_{rt})$
- $E(e^{rt}) = e^{\mu_{rt} + \frac{\sigma^2_{rt}}{2}}$

- $E(e^{r_t} - 1) = E(R_t) = e^{\mu_{r_t} + \frac{\sigma_{r_t}^2}{2}} - 1$
- $Var(e^{r_t}) = e^{2\mu+\sigma^2}[e^{\sigma^2} - 1]$
- $Var(e^{r_t}) = Var(e^{r_t} - 1) = Var(R_t)$

```
mu2 = mean(TSM_log_return,na.rm=T) # 0.0005480166
va2 = var(TSM_log_return,na.rm=T)  # 0.0006276544

# mean of the simple return
mean_R_t2 = exp(mu2+va2/2) - 1
mean_R_t2                           # 0.0008622153
```

```
##              TSM.Adjusted
## TSM.Adjusted 0.0008622152
```

```
# standard deviation of the simple return
sd_R_t2 = sqrt(exp(2*mu2+va2)*(exp(va2)-1))
sd_R_t2                             # 0.02507857
```

```
##              TSM.Adjusted
## TSM.Adjusted   0.02507857
```

c. Compute the sample mean and the sample standard deviation of the simple return R_t. Compare the results of (b) and (c).

- $R_t = \frac{P_t - P_{t-1}}{P_{t-1}}$
- $R_t = e^{r_t} - 1$

simple return

```
TSM_simple_return = exp(TSM_log_return) - 1
head(TSM_simple_return)
```

```
##            TSM.Adjusted
## 2020-01-02           NA
## 2020-01-03 -0.032977990
## 2020-01-06 -0.011539867
## 2020-01-07  0.016205092
## 2020-01-08  0.007372919
## 2020-01-09  0.008170251
```

```
TSM_simple_return2 = diff(TSM_adjust)/lag(TSM_adjust)  # lag 幫助往前除一天
head(TSM_simple_return2)
```

```
##             TSM.Adjusted
## 2020-01-02           NA
## 2020-01-03 -0.032977990
## 2020-01-06 -0.011539867
## 2020-01-07  0.016205092
## 2020-01-08  0.007372919
## 2020-01-09  0.008170251
```

the sample mean and the sample standard deviation

```
mean(TSM_simple_return, na.rm = T)       # 0.00086202
```

```
## [1] 0.00086202
```

```
sqrt(var(TSM_simple_return, na.rm = T))  # 0.02510999
```

```
##               TSM.Adjusted
## TSM.Adjusted   0.02510999
```

Estimate

- mean_R_t2 = 0.0008622153
- sd_R_t2 = 0.02507857

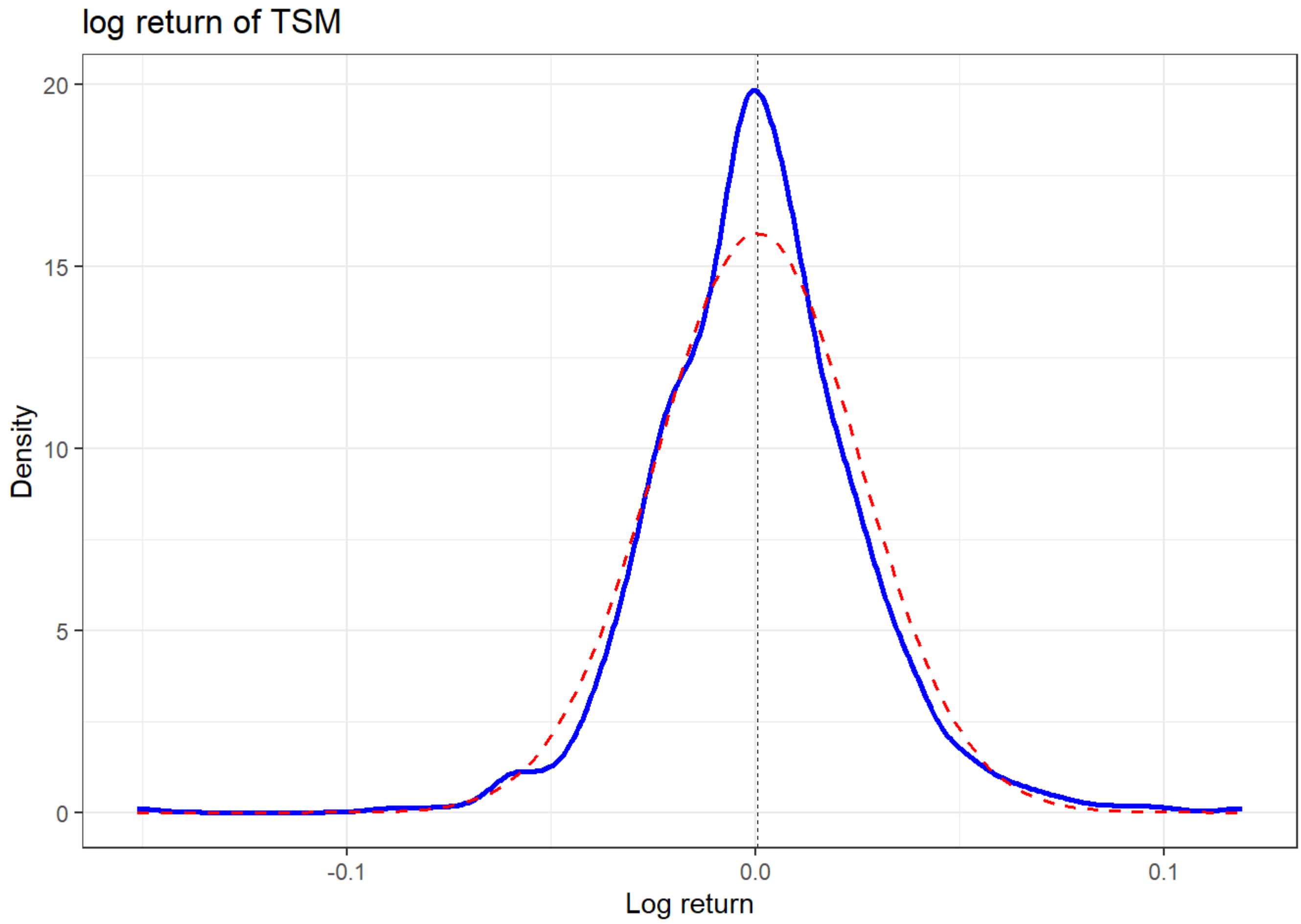
Compute

- mean = 0.00086202
- sd = 0.02510999
- 兩者結果幾乎一樣，r_t 可能真的符合常態假設

d. Find the kernel density estimator and normal density estimator
for the log return r_t and the simple return R_t respectively.
Compare the empirical kernel density and normal density for r_t and R_t
Plot the two estimated densities on the same graph.
(see Page 21 Figure 1.4 in the textbook)

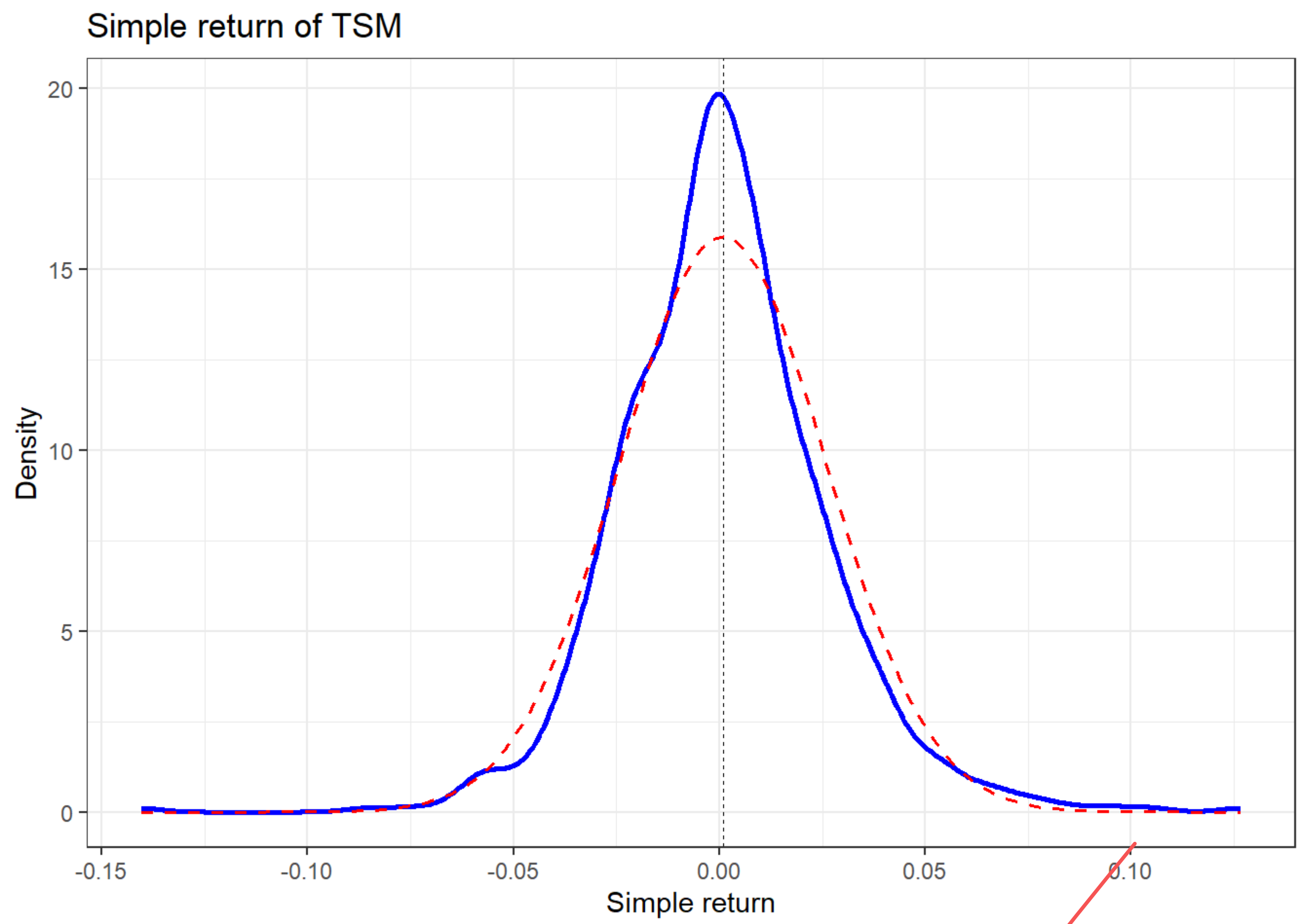
the log return

```
ggplot(na.omit(TSM_log_return),aes(x=TSM.Adjusted)) +
  geom_density(color = "blue",linewidth=1) +
  geom_vline(aes(xintercept = mu2),color="black",linetype="dashed",linewidth=0.1) +
  theme_bw() +
  labs(title="log return of TSM",x="Log return",y="Density")+
  stat_function(fun = dnorm, args = list(mu2, sqrt(va2)),
                colour = "red",linewidth=0.6,linetype="dashed")
```



the simple return

```
ggplot(na.omit(TSM_simple_return),aes(x=TSM.Adjusted)) +
  geom_density(color = "blue",linewidth=1) +
  geom_vline(aes(xintercept = mean_R_t2),color="black",linetype="dashed",linewidth=0.1) +
  theme_bw() +
  labs(title="Simple return of TSM",x="Simple return",y="Density")+
  stat_function(fun = dnorm, args = list(mean_R_t2, sd_R_t2),
                colour = "red",linewidth=0.6,linetype="dashed")
```



Compare the empirical kernel density and normal density for r_t and R_t

常態假設對 TSM 股票的日回報是有問題的。
經驗密度函數在其均值附近有一個更高的峰值，
但尾部比相應的正態分佈更厚。

- **In other words, the empirical density function is taller and skinnier**
- **but with a wider support than the corresponding normal density**