MGMTMFE407-2_HW1_Hyeuk Jung

Problem 1

Q1

$$\begin{aligned} E(r_t) &= E(\mu + \sigma * \varepsilon_t + J_t) \\ &= \mu + \sigma * E(\varepsilon_t) + E(J_t) \\ &= \mu + p * \mu_J \end{aligned}$$

$$\begin{aligned} Var(r_t) &= Var(\mu + \sigma * \varepsilon_t + J_t) \\ &= \sigma^2 * Var(\varepsilon_t) + Var(J_t) \\ &= \sigma^2 + \mu_J^2 * p(1-p) + \sigma_J^2 * p \end{aligned}$$

Used the cumulant generating function of the return model. Let X = $\mu + \sigma * \varepsilon_t \sim N(\mu, \sigma^2)$

$$\begin{aligned} K(r_t) &= ln(M_r(t)) = ln(M_X(t) * M_J(t)) = lnM_X(t) + lnM_J(t) \\ &= (\mu t + \frac{1}{2}\sigma^2 t^2) + ln(e^{\mu_j t} p + (1-p)) + ln(e^{\frac{1}{2}\sigma_J^2 t^2} p + (1-p)) \end{aligned}$$

$$\begin{aligned} Skew(r_t) &= S(\mu + \sigma * \varepsilon_t + J_t) \\ &= \frac{E[(r_t - \mu_r)^3]}{\sigma_r^3} \\ &= \frac{\frac{d^3}{dt^3} K(r_t)}{\sigma_r^3} \\ &= \frac{\mu_J^3 * p(1-p)^2}{(\sigma^2 + \mu_J^2 * p(1-p) + \sigma_J^2 * p)^{\frac{3}{2}}} \end{aligned}$$

$$\begin{aligned} Kurt(r_t) - 3 &= \frac{E[(r_t - \mu_r)^4]}{\sigma_r^4} - 3 \\ &= \frac{\frac{d^4}{dt^4} K(r_t)}{\sigma_r^4} - 3 \\ &= \frac{3 + \mu_J^4 p(1 - 7p + 8p^2 - 2p^3)}{(\sigma^2 + \mu_J^2 p(1-p) + \sigma_J^2 p)^2} - 3 \end{aligned}$$

Q2

Normal (continuous distribution) + Bernoulli (discrete distribution)

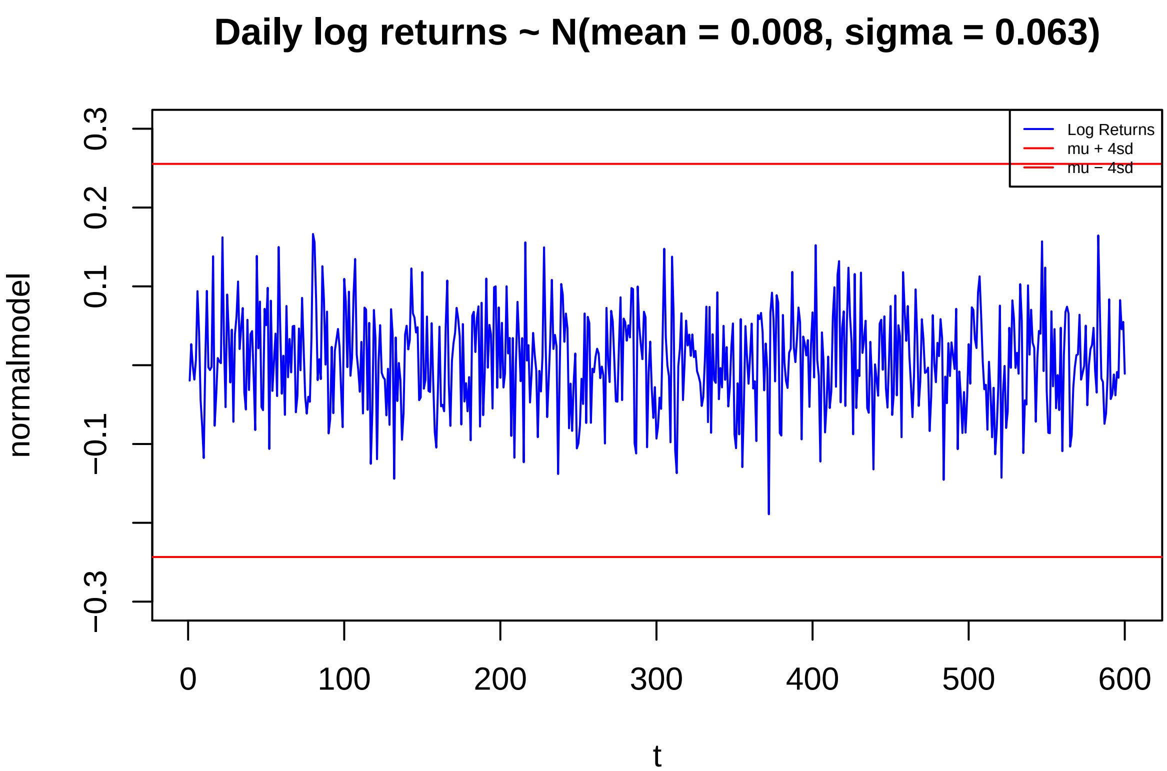
By combining both distribution, we can represent fat tails in to the model.

Q3

1) r ~ N(mean = 0.008, sd = 0.063), number of observations: 600 (50 years)

```r
# 1) r ~ N(mean = 0.008, sd = 0.063), number of observations: 600 (50 years)
r = 0.008; sigma = 0.063; t = 1:600
sim <- function(r, sigma, t) {
  #y = r + sigma*rnorm(length(t), sd = 1) # rnorm(): error term
  y = r + rnorm(length(t), sd = sigma)
}
normalmodel = sim(r, sigma, t)

#par(mfrow=c(1,2))
plot(t, normalmodel, pch=20, col="blue", ylim = c(-.3, .3), type = 'l')
abline(h = mean(normalmodel) + sd(normalmodel)*4, col = "red") # r + sigma*4
abline(h = mean(normalmodel) - sd(normalmodel)*4, col = "red") # r - sigma*4
title(paste("Daily log returns ~ N(mean = ", r,", sigma = ", sigma, ")", sep=""))
legend("topright", legend = c("Log Returns", "mu + 4sd", "mu - 4sd"), lty = 1, col = c("blue", "red", "
```

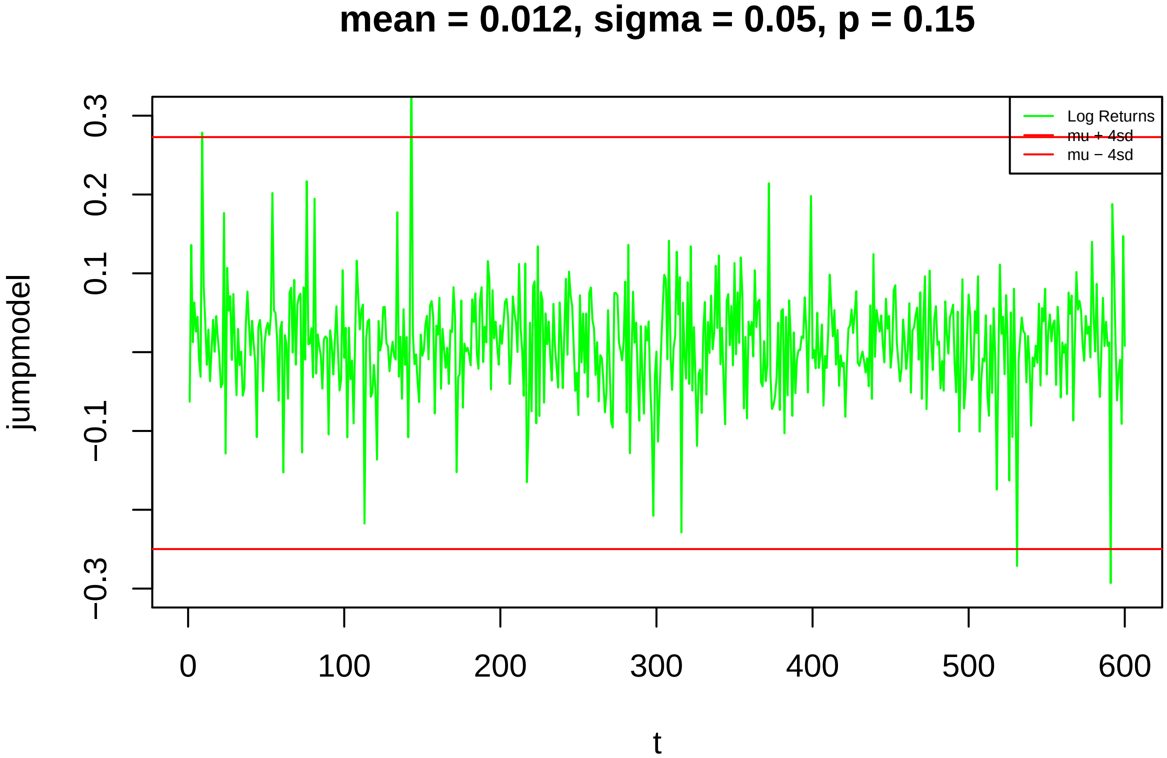


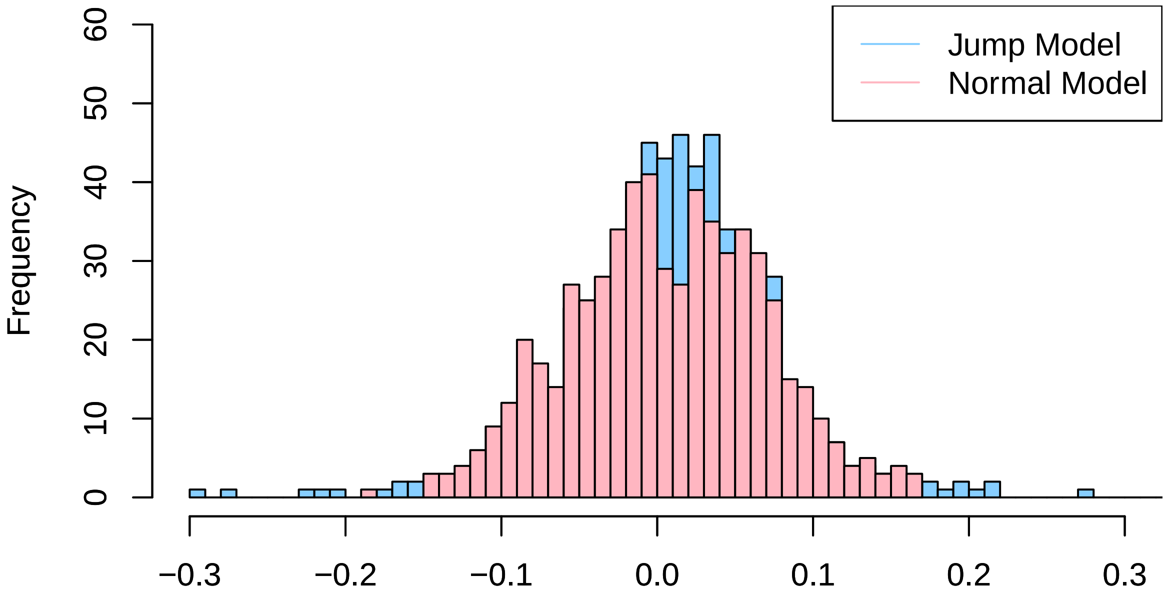
Compared to the data given in the lecture 1, all observations are within the +/- 4σ range. That being said, the simulation has no extreme case, which can be observed in the real market.

2) r = mean + sd*e_t + J_t

```r
# e_t ~ N(0, 1)
# J_t; mean = mean(J)*p, var = mean(J)^2*p*(1-p) + var(J)*p
mean = 0.012; sigma = 0.05; p = 0.15; mean_j = -0.03; sigma_j = 0.1
sim2 <- function(mean, sigma, p, mean_j, sigma_j, t) {
  y = mean + sigma*rnorm(length(t), sd = 1) + rbernoulli(length(t), p)*(mean_j + sigma_j*rnorm(length(t
}
jumpmodel = sim2(mean, sigma, p, mean_j, sigma_j, t)

#par(mfrow=c(1,2))
plot(t, jumpmodel, pch=20, col="green", ylim = c(-.3, .3), type = 'l')
abline(h = mean(jumpmodel) + sd(jumpmodel)*4, col = "red") # mean + mean_j*p + sigma*4
abline(h = mean(jumpmodel) - sd(jumpmodel)*4, col = "red")
title(paste("mean = ", mean,", sigma = ", sigma, ", p = ", p, sep=""))
legend("topright", legend = c("Log Returns", "mu + 4sd", "mu - 4sd"), lty = 1,
       col = c("green", "red", "red"), cex = .5)
```





Unconditional mean, var, skewness, and kurtosis of the jump model:

```
## [1] "mean =  0.011485825668092"
```

```
## [1] "variance =  0.00426936350050029"
```

```
## [1] "skewness =  -0.0553409184131869"
```

```
## [1] "kurtosis =  6.57962606301317"
```

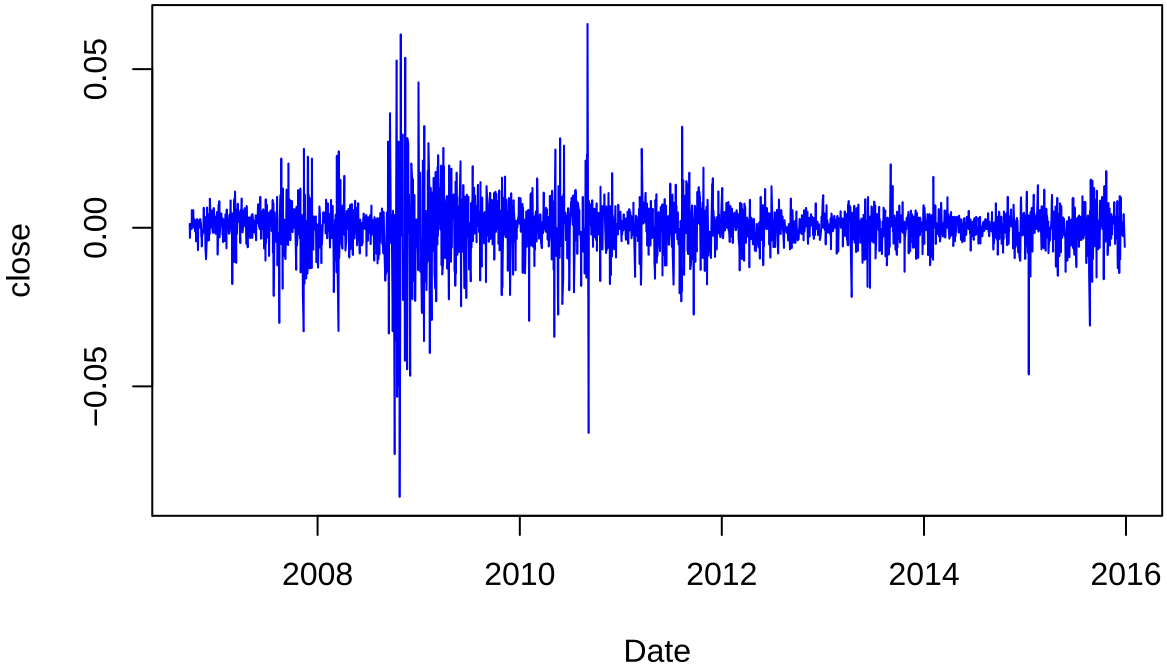
Compared to the simulation above, the simulated results are more silimar to the real market data. We can observe extreme cases reaching +/- 4σ range, which implies that the distribution of the result would have fat tail. However, we still cannot see volatility clustering.

Problem 2

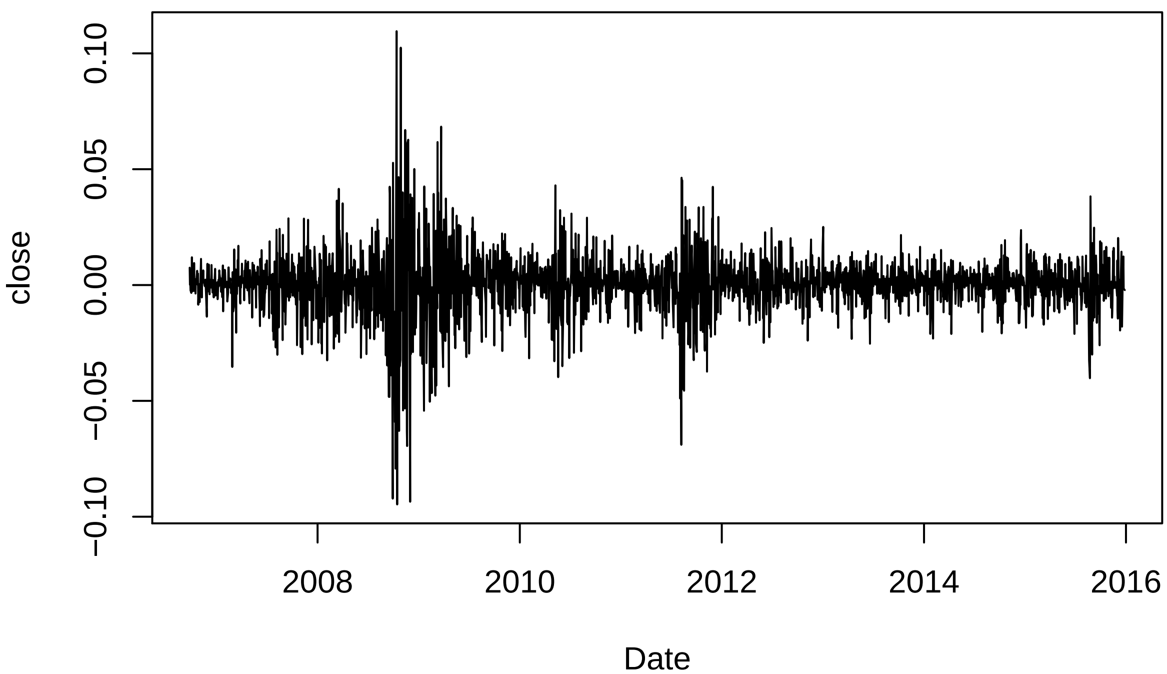
Q1. Visualizing the data

(a) Time series plots of the daily log-returns for DBV and GSPC

```r
with(dbv, plot(Date, close, type = 'l', col = "blue"))
```

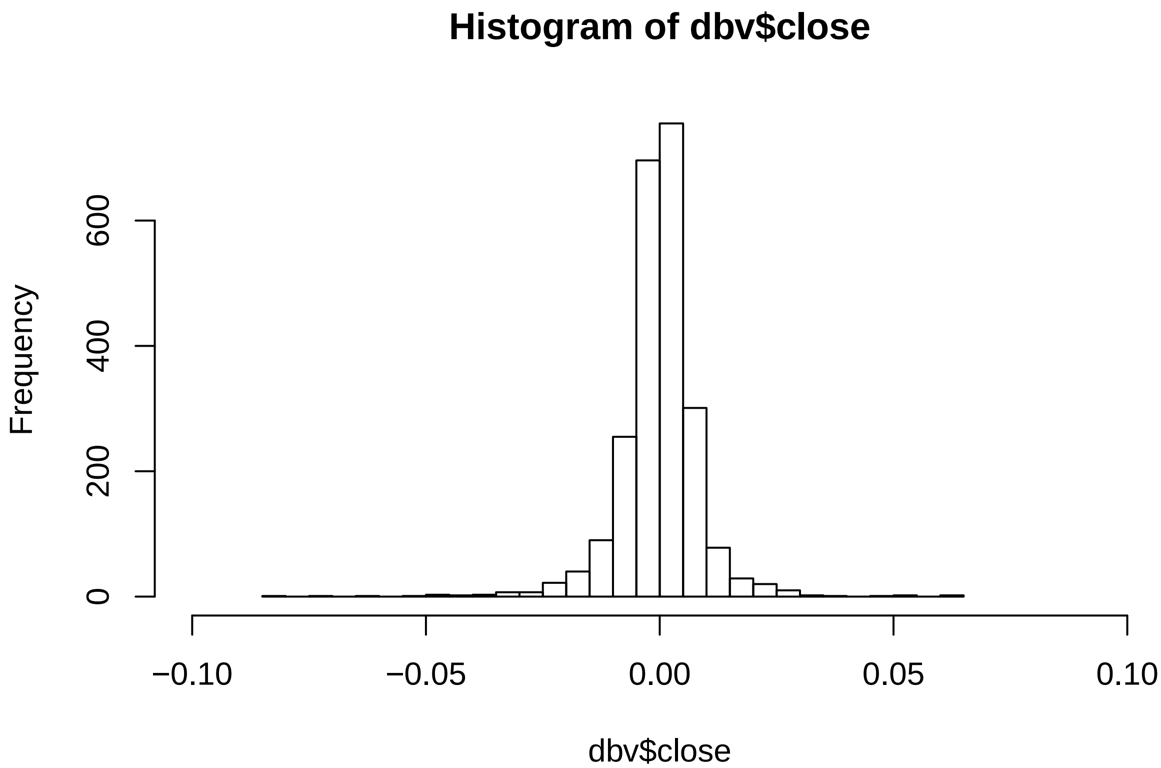



```r
with(gspc, plot(Date, close, type = 'l'))
```

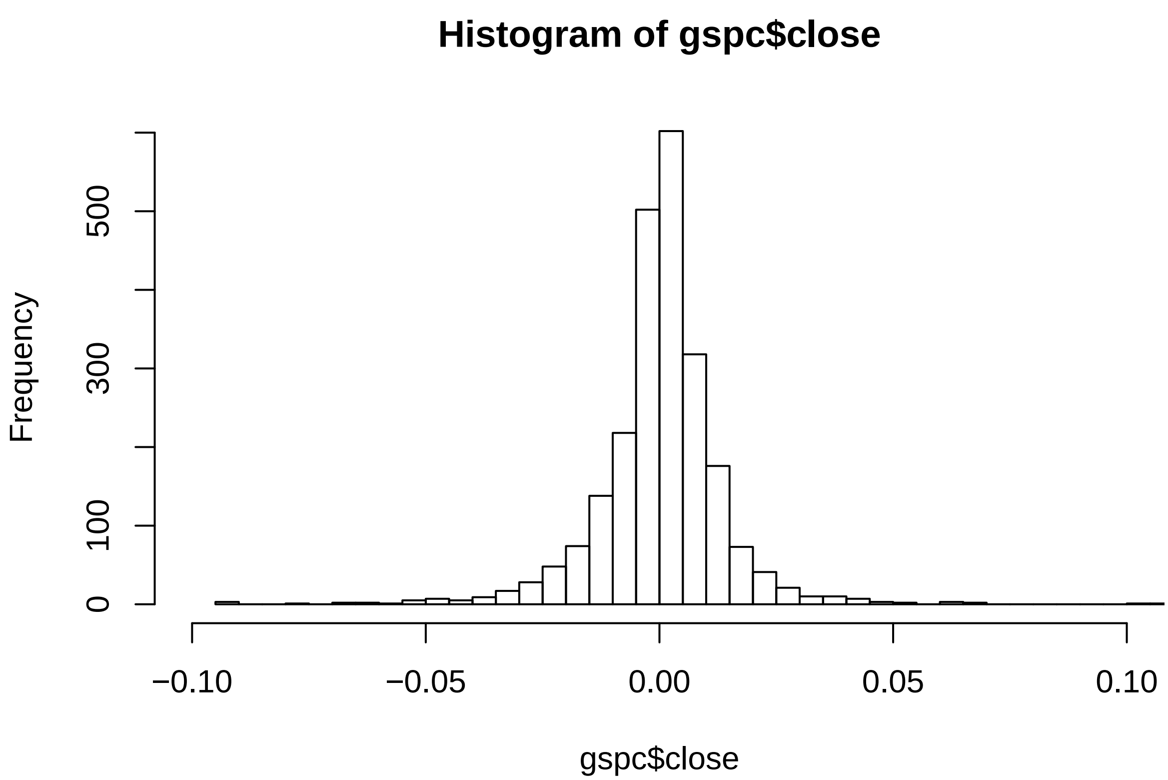


(b) Histograms of the daily log-returns for DBV and GSPC

```r
hist(dbv$close, breaks = 50, xlim = c(-.1, .1))
```




```r
hist(gspc$close, breaks = 50, xlim = c(-.1, .1))
```



Q2. Shape of Return Distribution

(a) H0: skewness = 0, a = 0.05

```r
# DBV (Deutsche Bank)
skew_dbv <- skewness(dbv$close)
skew_dbv_t <- skew_dbv / sqrt(6/length(dbv$close))
paste(abs(skew_dbv_t), ">", 1.96, ": reject null")
```

```
## [1] "16.6918720967581 > 1.96 : reject null"
```

```r
#pt(0.975, nrow(dbv)-1)

# GSPC (S&P 500)
skew_gspc <- skewness(gspc$close)
skew_gspc_t <- skew_gspc / sqrt(6/length(gspc$close))
paste(abs(skew_gspc_t), ">", 1.96, ": reject null")
```

```
## [1] "6.385640855269 > 1.96 : reject null"
```

(b) H0: excess kurtosis = 0, a = 0.05

```r
# DBV (Deutsche Bank)
kurt_dbv <- kurtosis(dbv$close) - 3
kurt_dbv_t <- kurt_dbv / sqrt(24/length(dbv$close))
paste(abs(kurt_dbv_t), ">", 1.96, ": reject null")
```

```
## [1] "133.181585624915 > 1.96 : reject null"
```

```r
# GSPC (S&P 500)
kurt_gspc <- kurtosis(gspc$close) - 3
kurt_gspc_t <- kurt_gspc / sqrt(24/length(gspc$close))
paste(abs(kurt_gspc_t), ">", 1.96, ": reject null")
```

```
## [1] "96.5398997596473 > 1.96 : reject null"
```

(c) Jarque-Bera test (H0: skewwness & excess kurtosis = 0 (log return ~ Normal), a = 0.05)

```r
#jb = ( skew_dbv^2 / (6/length(dbv$close)) ) + (kurt_dbv^2 / (24/length(dbv$close)))
# DBV (Deutsche Bank)
jb_dbv <- skew_dbv_t^2 + kurt_dbv_t^2
jb_dbv_chi <- qchisq(0.95, df = 2)
paste(jb_dbv, ">", jb_dbv_chi, ": reject null")
```

```
## [1] "18015.9533436611 > 5.99146454710798 : reject null"
```

```r
# GSPC (S&P 500)
jb_gspc <- skew_gspc_t^2 + kurt_gspc_t^2
jb_gspc_chi <- qchisq(0.95, df = 2)
paste(jb_gspc, ">", jb_gspc_chi, ": reject null")
```

```
## [1] "9360.72865473523 > 5.99146454710798 : reject null"
```

Q3.

	DBV (Deutsche Bank)	GSPC (S&P 500)
Skewness	-0.8470376	-0.3240426
t-test	-16.6918721	-6.3856409
Excess Kurtosis	13.5167363	9.7979339
t-test	133.1815856	96.5398998
JB-test	18015.9533437	9360.7286547
p-value JB	0.0000000	0.0000000

From the result, it is hard to say that log returns of both the DBV and S&P500 indices are following the normal distribution. For the DBV index, its negative skewness and high kurtosis shows that the index has been exposed to more extreme events compared to the S&P500.

Q4.

```r
# 1. scale the data to hold annual return = 0.2, annual sd = 0.4 (sharpe ratio = E(r)/sd = 0.5)
daily_target_r <- 0.2/252
daily_target_sd <- 0.4/sqrt(252)

# DBV
mean_dbv_daily <- mean(dbv$close)
sd_dbv_daily <- sd(dbv$close)
################################
# I'm not sure if this is the right way to scale the return and standard deviation
dbv$ret <- ((dbv$close - mean_dbv_daily)/sd_dbv_daily)* daily_target_sd + daily_target_r
################################
## Annual returns
dbv_annual <- apply.yearly(dbv[, .(Date, close)], function(x) prod(1 + x, na.rm = T) - 1)
#mean(dbv_annual) # -0.003420513
dbv_lever <- apply.yearly(dbv[, .(Date, ret)], function(x) prod(1 + x, na.rm = T) - 1)
#mean(dbv_lever) # 0.2006492

# GSPC
mean_gspc_daily <- mean(gspc$close)
sd_gspc_daily <- sd(gspc$close)
gspc$ret <- ((gspc$close - mean_gspc_daily)/sd_gspc_daily)* daily_target_sd + daily_target_r
## Annual returns
gspc_annual <- apply.yearly(gspc[, .(Date, close)], function(x) prod(1 + x, na.rm = T) - 1)
#mean(gspc_annual) # 0.04344666
gspc_lever <- apply.yearly(gspc[, .(Date, ret)], function(x) prod(1 + x, na.rm = T) - 1)
#mean(gspc_lever) # 0.1985398
```

	Original DBV	Leveraged DBV	Original GSPC	Leveraged GSPC
Daily Return	-0.0000147	0.0007937	0.0001882	0.0007937
Annual Return	-0.0034205	0.2006492	0.0434467	0.1985398
Standard Deviation	0.0087652	0.0251976	0.0134675	0.0251976
SD (annual)	0.1331622	0.4111677	0.1927480	0.3753094
Skewness	-0.8470376	-0.8470376	-0.3240426	-0.3240426
Kurtosis	16.5167363	16.5167363	12.7979339	12.7979339

Using these numbers, discuss the different nature of the risks you would face if you invested in equity or currency markets to achieve that target return (with the appropriate leverage).

While we can manipulate the return and standard deviation, it does not affect the skewness and the kurtosis of the distribution.

Q5.

```r
# Classic OLS (under normality)
out <- lm(dbv$close ~ gspc$close)
#summary(out)
ols <- summary(out)$coef[1:2, 1:2]

# Regression allowing non-normality and HAC
library(sandwich)
hac <- sqrt(diag(vcovHAC(out)))
```

	Standard OLS	Non-normality and HAC
Slope	0.0101102	0.0182741
Intercept	0.0001361	0.0001189

While we are assuming constant volatility for all residuals under homogeneity, heteroskedastic standard errors does not hold this assumption and accepts wider/narrower intervals of the residuals.

In general, residuals increases as the value of independent variable increases. Thus, the robust standard errors are larger than non-robust (homogeneity) standard errors, evaluating coefficients conservatively (as it decreases the t-stat of the coefficients).

In the result above, the robust standard error of the slope holds larger value while that of the intercept decreased.

Reason for smaller robust standard errors?