Figure & Code homework4

Presented by Xiaomeng Rao
ID Numb 15620161152244
Department of Finance, SOE

Figure 3

```
install.packages("rjson",repos="http://cran.us.r-project.org")
library("rjson")
json_file = http://crix.hu-berlin.de/data/crix.json
json_data = fromJSON(file=json_file)
crix_data_frame = as.data.frame(json_data)
x = crix_data_framen = dim(x)
a = seq(1,n[2],2)
b = seq(2,n[2],2)
date = t(x[1,a])
price = t(x[1,b])
plot(price)
dim(price)
ts.plot(price)
ret = diff( log(price) )
ts.plot( ret )
```

Figure 3

```
load(file = "C:/Users/Alex/Desktop/hw4/crix.RData")
load(file = "C:/Users/Alex/Desktop/hw4/ecrix.RData")
load(file = "C:/Users/Alex/Desktop/hw4/efcrix.RData")

# plot of crix
# plot(as.xts(crix), type="l", auto.grid=FALSE, main = NA)
plot(crix1, ylab = NA, xlab = NA)
```

Figure 3

Figure 3&4

```
# plot of crix return
ret   = diff(log(crix1))
# plot(as.xts(ret), type="l", auto.grid=FALSE, main = NA)
plot(ret, ylab = NA, xlab = NA)
```

Figure 4

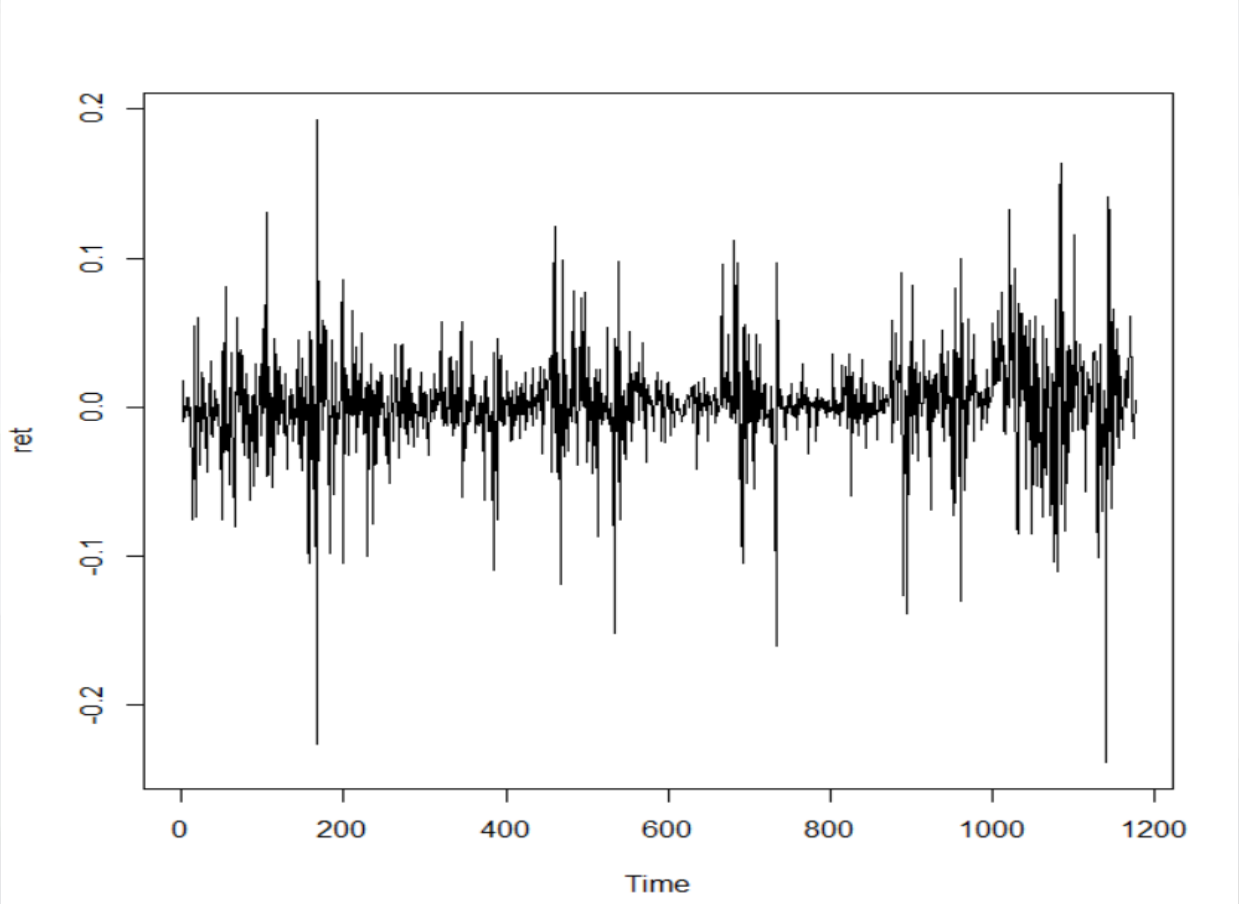

Figure 5

```
# histogram of returns
hist(ret, col = "grey", breaks = 20, freq = FALSE, ylim = c(0, 25), xlab = NA)
lines(density(ret), lwd = 2)
mu = mean(ret)
sigma = sd(ret)
x = seq(-4, 4, length = 100)
curve(dnorm(x, mean = mean(ret), sd = sd(ret)), add = TRUE, col = "darkblue",
    lwd = 2)
# qq-plot
qqnorm(ret)
qqline(ret, col = "blue", lwd = 3)
```

Figure 5

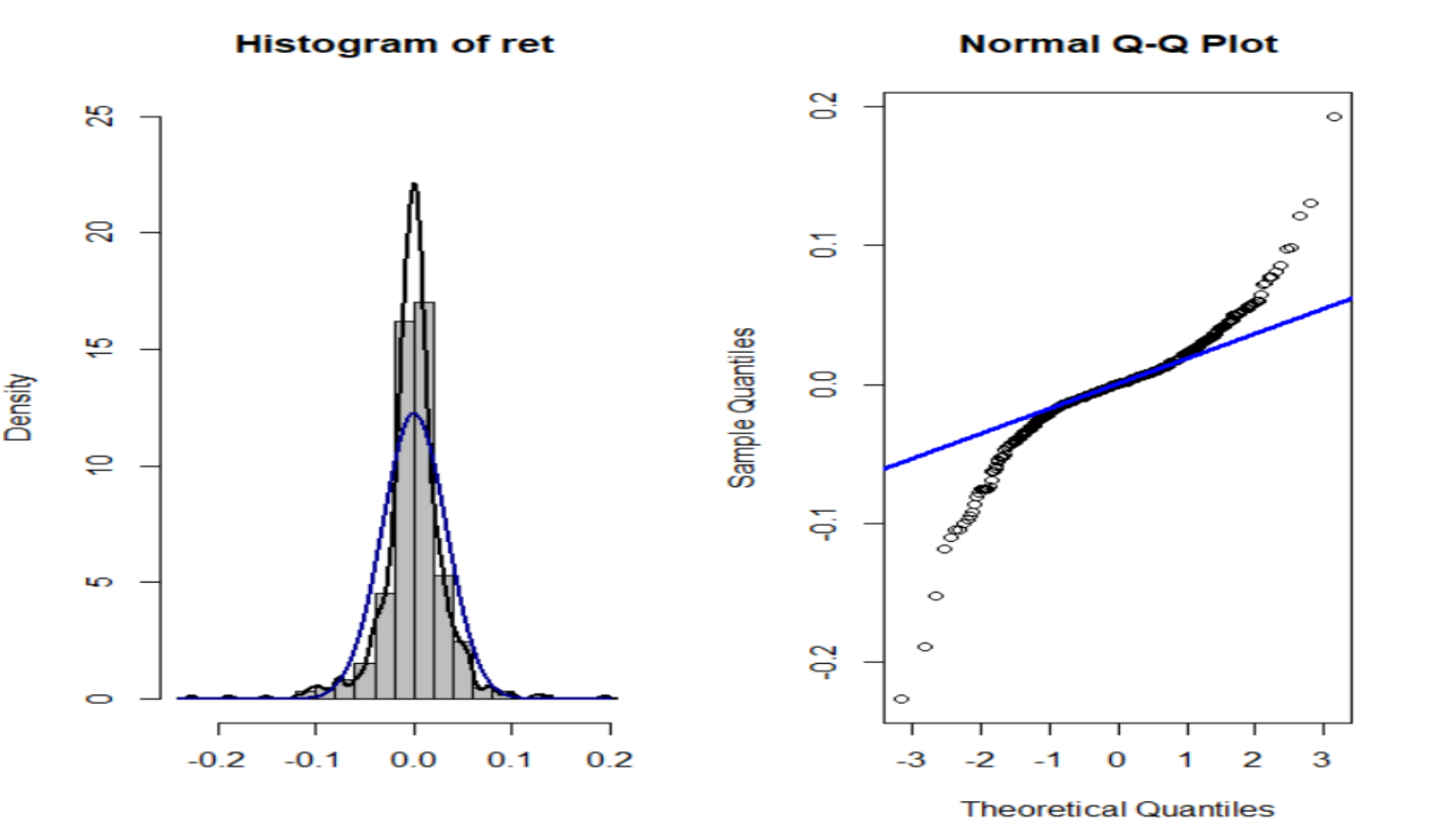
Histogram of ret
Density
0
5
10
15
20
25
-0.2
-0.1
0.0
0.1
0.2
Normal Q-Q Plot
Sample Quantiles
-0.2
-0.1
0.0
0.1
0.2
-3
-2
-1
0
1
2
3
Theoretical Quantiles

Figure 6&7

```
Box.test(crixreturn, type = "Ljung-Box", lag = 20)
adf.test(crixreturn, alternative = "stationary")
kpss.test(crixreturn, null = "Trend")
par(mfrow = c(1, 2))
autocorr = acf(crixreturn, lag.max = 20, ylab = "Sample Autocorrelation", main
= NA,
             lwd = 2, ylim = c(-0.3, 1))
print(cbind(autocorr$lag, autocorr$acf))
Box.test(crixreturn, type = "Ljung-Box", lag = 1, fitdf = 0)
Box.test(autocorr$acf, type = "Ljung-Box")
autopcorr = pacf(crixreturn, lag.max = 20, ylab = "Sample Partial
Autocorrelation",
               main = NA, ylim = c(-0.3, 0.3), lwd = 2)
```

Figure 6

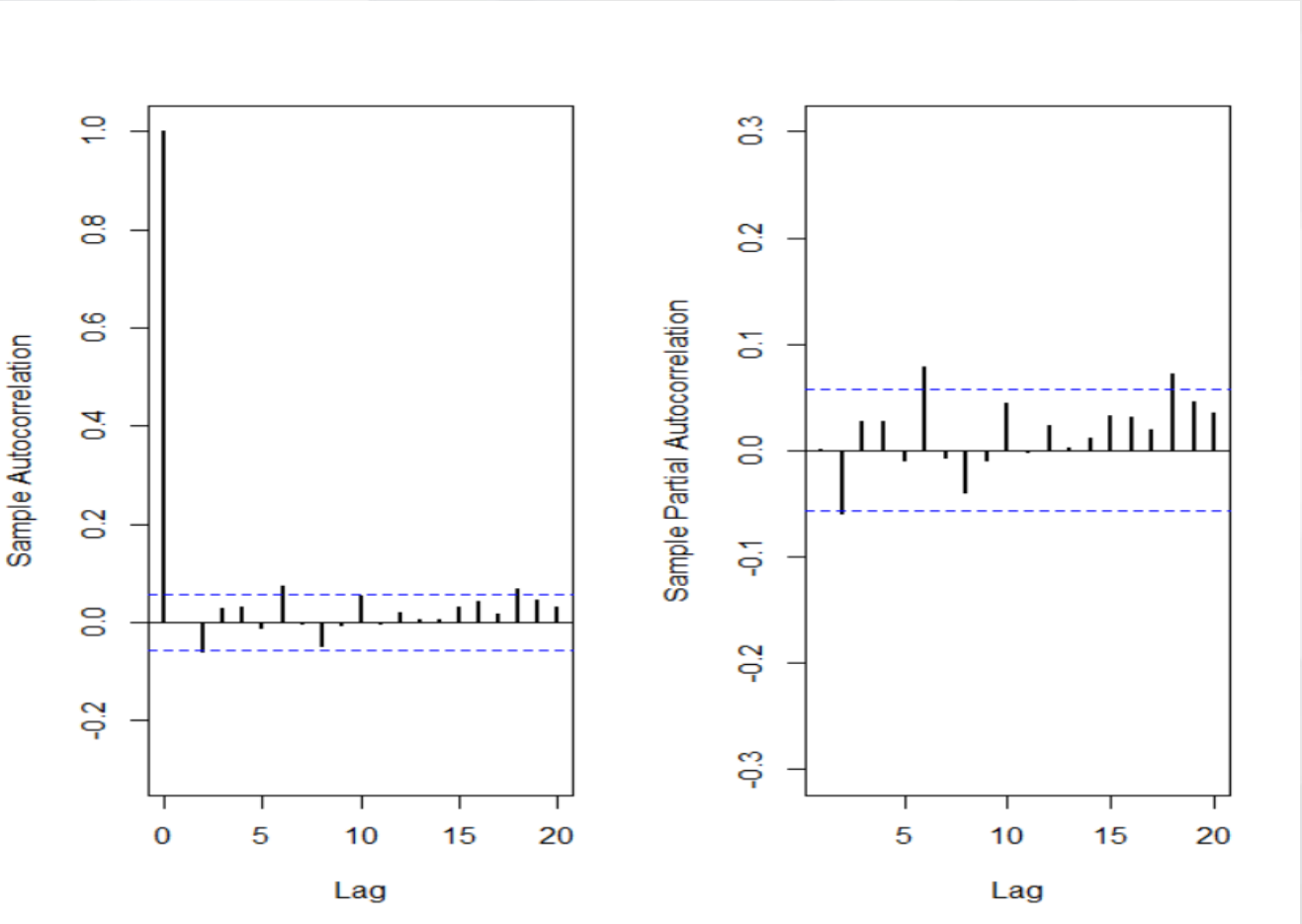
Sample Autocorrelation
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
0
5
10
15
20
Lag
Sample Partial Autocorrelation
0.3
0.2
0.1
0.0
-0.1
-0.2
-0.3
5
10
15
20
Lag

Figure 7

```
par(mfrow = c(1, 1))
auto.arima(crixreturn)
fit1 = arima(crixreturn, order = c(1, 0, 1))
tsdiag(fit1)
Box.test(fit1$residuals, lag = 1)

aic = matrix(NA, 6, 6)
for (p in 0:4) {
  for (q in 0:3) {
    a.p.q = arima(crixreturn, order = c(p, 0, q))
    aic.p.q = a.p.q$aic
    aic[p + 1, q + 1] = aic.p.q
}
}
aic
```

Figure 7

```
bic = matrix(NA, 6, 6)
for (p in 0:4) {
 for (q in 0:3) {
   b.p.q = arima(crixreturn, order = c(p, 0, q))
   bic.p.q = AIC(b.p.q, k = log(length(crixreturn)))
   bic[p + 1, q + 1] = bic.p.q
 }
}
bic
fit4 = arima(crixreturn, order = c(2, 0, 3))
tsdiag(fit4)
Box.test(fit4$residuals, lag = 1)
fitr4 = arima(crixreturn, order = c(2, 1, 3))
tsdiag(fitr4)
Box.test(fitr4$residuals, lag = 1)
fit202 = arima(crixreturn, order = c(2, 0, 2))
tsdiag(fit202)
tsdiag(fit4)
```

Figure 7

```
tsdiag(fitr4)
AIC(fit202, k = log(length(crixreturn)))
AIC(fit4, k = log(length(crixreturn)))
AIC(fitr4, k = log(length(crixreturn)))
fit202$aic
fit4$aic
fitr4$aic
fit202 = arima(crixreturn, order = c(2, 0, 2))
crpre = predict(fit202, n.ahead = 30)
dates = seq(as.Date("02/08/2014", format = "%d/%m/%Y"), by = "days", length =
length(crixreturn))
plot(crixreturn, type = "l", xlim = c(0, 1200), ylab = "log return", xlab = "days", lwd = 1)
lines(crpre$pred, col = "red", lwd = 3)
lines(crpre$pred + 2 * crpre$se, col = "red", lty = 3, lwd = 3)
lines(crpre$pred - 2 * crpre$se, col = "red", lty = 3, lwd = 3)
```

Figure7