

HW Unit 4

By: Qian Liu

15620161152269

1.

Fig 3

```
>libraries = c("ccgarch", "rmgarch", "xts", "zoo")
>lapply(libraries, function(x) if (!(x %in% installed.packages())) { install.packages(x) })
>lapply(libraries, library, quietly = TRUE, character.only = TRUE)

>load(file = "crix.RData")
>load(file = "ecrix.RData")
>load(file = "efcrix.RData")

>ecrix1 = zoo(ecrix, order.by = index(crix1))
>efcrix1 = zoo(efcrix, order.by = index(crix1))

>my.panel <- function(x, ...) {
  lines(x, ...)
  lines(ecrix1, col = "blue")
  lines(efcrix1, col = "red") }

>plot.zoo(crix1, plot.type = "multiple", type = "l", lwd = 1.5, panel = my.panel, main = "Indices in the CRIX family")
```

Fig 4

```
> 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_frame
> n=dim(x)
> a=seq(1,n[2],2)
> b=seq(2,n[2],2)
> date=t(x[1,a])
> price=t(x[1,b])
> ts.plot(price)
> ret=diff(log(price))
> ts.plot(ret)
```

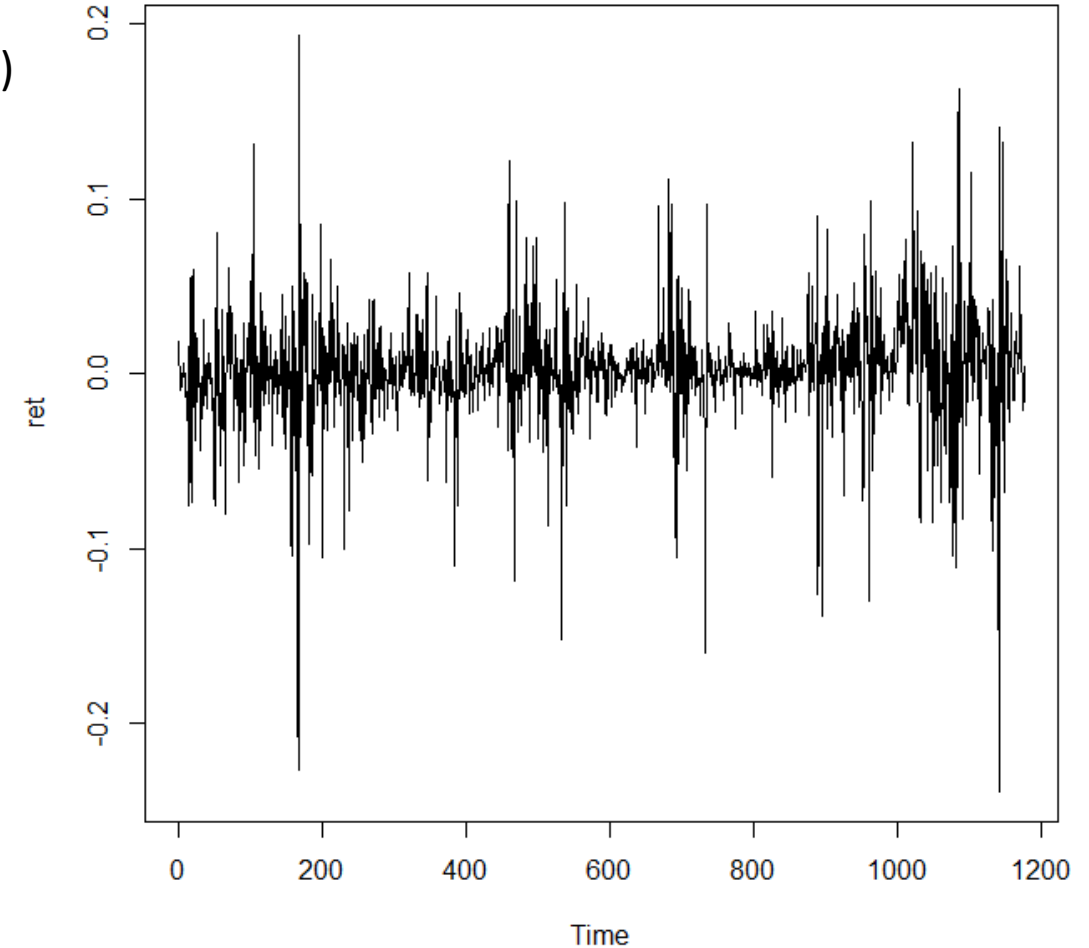


Fig 5

```
# install and load packages
```

```
>libraries = c("zoo", "tseries", "xts")
```

```
>lapply(libraries, function(x) if (!(x %in% installed.packages())) { install.packages(x) })
```

```
>lapply(libraries, library, quietly = TRUE, character.only = TRUE)
```

```
# load dataset
```

```
>load(file = "crix.RData")
```

```
# 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)
```

Fig 6

```
> 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_frame
> n=dim(x)
> a=seq(1,n[2],2)
> b=seq(2,n[2],2)
> date=t(x[1,a])
> price=t(x[1,b])
> ts.plot(price)
> ret=diff(log(price))
> ts.plot(ret)

# plot of pacf
> autopcorr=pacf(ret,lag.max=20,ylab="Sample Partial Autocorrelation",main=NA,ylim=c(-0.3,0.3),lwd=2)
> print(cbind(autopcorr$lag, autopcorr$acf))
```

2.

Fig 7

```
# arima202 predict
```

```
>fit202 = arima(ret, 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(ret))
```

```
>plot(ret, type = "l", xlim = c(0, 644), ylab = "log return", xlab = "days", lwd = 1.5)
```

```
>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)
```

3.

Fig 8

```
# install and load packages
>libraries = c("FinTS", "tseries")
>lapply(libraries, function(x) if (!(x %in% installed.packages())) { install.packages(x) })
>lapply(libraries, library, quietly = TRUE, character.only = TRUE)

>setwd()

>load(file = "crix.RData")
>Pr = as.numeric(crix)
>Da = factor(date1)
>crx = data.frame(Da, Pr)

# plot of crix return
>ret = diff(log(crx$Pr))
>Dare = factor(date1[-1])
>retts = data.frame(Dare, ret)
```