Assignment 4 STA457H1F

Mattea Busato

December 4, 2024

1 Exercise - Daily stock prices for Barrick Gold

1.1 Fitting ARIMA models

Firstly, we conduct the Augmented Dickey-Fuller test on the data to see if the time series is non-stationary. The test, performed with different lag values, yields p-values significantly higher than the threshold of 0.05. For this reason, we have no evidence for rejecting the null hypothesis and we can assume that the data is non-stationary.

We can now confidently proceed with fitting ARIMA(0,1,1) and ARIMA (0,1,2) models to the data without the worry of over-differencing stationary data.

We fit both models to the data and then we plot the residuals. The residuals do not look like white noise from the plots. This would mean that the models do not fit the data too well and that there is still substantial unexplained variance in the residuals.

To assess the validity of our intuition we conduct the Ljung-Box test using lag=10 (chosen by common rule of thumb):

· For the residuals of the ARIMA(0,1,1) model we obtain a p-value of 0.0003061 which is lower than the threshold of 0.05, we can then reject the null hypothesis that the residuals are white noise.

· For the residuals of the ARIMA(0,1,2) model we obtain a p-value of 0.09712 which is higher than the threshold of 0.05, we then retain the null hypothesis.

This test shows us that ARIMA(0,1,2) fits better the data, since its residuals are closer to white noise than ARIMA(0,1,1).

To double-check this result we conduct an AIC test. We obtain values of -24438.09 for ARIMA(0,1,1) and of -24450.7 for ARIMA(0,1,2), which tells us that indeed the latte model fits the data better.

1.2 Fitting ARCH models

We now fit models ARCH(1), ARCH(2), ARCH(3), ARCH(4) and ARCH(5) to the residuals from the ARIMA(0,1,2) model we have found in the previous section. We observe the following AIC values:

· -4.503860 for ARCH(1)
· -4.533235 for ARCH(2)
· -4.562443 for ARCH(3)
· -4.576758 for ARCH(4)
· -4.588344 for ARCH(5)

This tells us that ARCH(5) is the model that fits best the residuals of the previously found ARIMA(0,1,2) model.

1.3 Fitting GARCH models

We now fit models GARCH(1,1), GARCH(1,2) and GARCH(1,3) to the residuals from the ARIMA(0,1,2) model we have found in the first section. We observe the following AIC values:

· -4.664868 for GARCH(1,1)
· -4.665924 for GARCH(1,2)
· -4.665919 for GARCH(1,3)

The AIC values show that these newly fitted models are better than all the previous ARCH models. In particular, GARCH(1,2) shows the lowest AIC value out of all the models we have fitted for the residuals. The GARCH models generally outperform ARCH models due to their ability to model conditional variance more flexibly by incorporating lagged conditional variance terms. The results we have obtained in the last section were to be expected.

2 Exercise - Stationary bivariate process

2.1 Derive the cross-covariance

We recall that the cross-covariance function $\gamma_{xy}(s)$ is related to the cross-spectrum $f_{xy}(\omega)$ by

$$f_{xy}(\omega) = \sum_{s=-\infty}^{\infty} \gamma_{xy}(s) exp(2\pi i \omega s) = \int_{-\infty}^{\infty} \gamma_{xy}(s) exp(2\pi i \omega s) \delta_{s \in Z}(s) ds$$

where δ is the Kronecker-delta function defined as

$$\delta_{s \in Z}(s) = \begin{cases} 0 \; for \; s \notin Z \\ \infty \; for \; s \in Z \end{cases} \tag{1}$$

and the cross-covariance is defined as

$$\gamma_{xy}(s) exp(2\pi i \omega s) = Cov(x_t, x_{t+s})$$

The cross-spectrum function $f_{xy}(\omega)$ is the Fourier transform of the cross-covariance function γ_{xy}. The Fourier inversion formula states that given the following Fourier transform

$$\hat{f}(\xi) = \int f(t) e^{-it\xi} dt$$

the inverse formula is

$$f(t) = \frac{1}{2\pi} \int \hat{f}(\xi) e^{it\xi} d\xi$$

From the Inversion Theorem, we can directly derive the cross-covariance function

$$\gamma_{xy}(s) = \frac{1}{2\pi} \int f_{xy}(\omega) exp(-2\pi i \omega s) 2\pi d\omega$$

The cross-spectrum attains values in the interval [0,1], hence:

$$\gamma_{xy}(s) = \int_0^1 f_{xy}(\omega) exp(-2\pi i \omega s) d\omega$$

2.2 Predict Y_t using $\{X_t\}$

We aim to predict Y_t using a linear combination of X_t

$$\hat{Y}_t = \sum_{s=-\infty}^{\infty} a_s X_{t-s}$$

where the a_s's are chosen to minimize the prediction MSE

$$MSE = E[(Y_t - \hat{Y}_t)^2] = E[(Y_t - \sum_{s=-\infty}^{\infty} a_s X_{t-s})^2]$$

We minimize the MSE by setting the partial derivative with respect to each a_s to zero:

$$\frac{\partial MSE}{\partial a_s} = 0$$

Expanding the MSE:

$$MSE = E[Y_t^2] - 2 \sum_{s=-\infty}^{\infty} a_s E[Y_t X_{t-s}] + \sum_{s=-\infty}^{\infty} \sum_{s'=-\infty}^{\infty} a_s a_{s'} E[X_{t-s} X_{t-s'}]$$

Taking the derivative with respect to a_u:

$$\frac{\partial MSE}{\partial a_u} = -2E[Y_t X_{t-u}] + 2\sum_{s=-\infty}^{\infty} a_s E[X_{t-s}X_{t-u}]$$

Setting the derivative to zero:

$$E[Y_t X_{t-u}] = \sum_{s=-\infty}^{\infty} a_s E[X_{t-s}X_{t-u}]$$

Using the cross-covariance function and autocovariance function:

$$\gamma_{xy} = E[Y_t X_{t-u}]$$

$$\gamma_x(u-s) = E[X_{t-s}X_{t-u}]$$

the equation becomes:

$$\sum_{s=-\infty}^{\infty} a_s \gamma_x(u-s) = \gamma_{xy}(u)$$

for U$\in$ Z as wanted.

2.3 Determine the transfer function $\Gamma(\omega)$

We recall that the autocovariance functions $\gamma_x(s)$ is related to the spectral density $f_x(\omega)$ by

$$f_x(\omega) = \sum_{s=-\infty}^{\infty} \gamma_x(s) exp(-2\pi i\omega s)$$

Similarly, the cross-covariance function $\gamma_{xy}(s)$ is related to the cross-spectrum $f_{xy}(\omega)$ by

$$f_{xy}(\omega) = \sum_{s=-\infty}^{\infty} \gamma_{xy}(s) exp(-2\pi i\omega s)$$

Starting from the equation we have previously derived

$$\sum_{s=-\infty}^{\infty} a_s \gamma_x(u-s) = \gamma_{xy}(u)$$

we multiply both sides by $exp(2\pi i\omega u)$:

$$\sum_{s=-\infty}^{\infty} a_s \gamma_x(u-s) exp(2\pi i\omega u) = \gamma_{xy}(u) exp(2\pi i\omega u)$$

We sum both sides over u from $-\infty$ to ∞:

$$\sum_{u=-\infty}^{\infty}(\sum_{s=-\infty}^{\infty} a_s \gamma_x(u-s)) exp(2\pi i\omega u) = \sum_{u=-\infty}^{\infty} \gamma_{xy}(u) exp(2\pi i\omega u)$$

We can then derive:

$$\sum_{u=-\infty}^{\infty}\sum_{s=-\infty}^{\infty} a_s \gamma_x(u-s) exp(2\pi i\omega u) = \sum_{v=-\infty}^{\infty} \gamma_{xy}(v) exp(-2\pi i\omega v)$$

$$\sum_{s=-\infty}^{\infty} a_s(\sum_{u=-\infty}^{\infty} \gamma_x(u-s) exp(2\pi i\omega u)) = f_{xy}(\omega)$$

$$\sum_{s=-\infty}^{\infty} a_s exp(2\pi i\omega s) f_x(\omega) = f_{xy}(\omega)$$

$$\Gamma(\omega) f_x(\omega) = f_{xy}(\omega)$$

$$\Gamma(\omega) = \frac{f_{xy}(\omega)}{f_x(\omega)}$$

2.4 Values for a_s for $s = -20, ..., 20$

We derive the transfer function with the equation we have previously derived:

$$\Gamma(\omega) = \frac{f_{xy}(\omega)}{f_x(\omega)} = \frac{(\omega - 1/2)^4}{4(\omega - 1/2)^2} = \frac{1}{4}(\omega - 1/2)^2$$

To find the a_s we find the inverse of the transfer function which is

$$= \int_{-1/2}^{1/2} \frac{1}{4}(\omega - 1/2)^2 exp(2\pi i \omega s) d\omega$$

The coefficients are seen in Figure 1

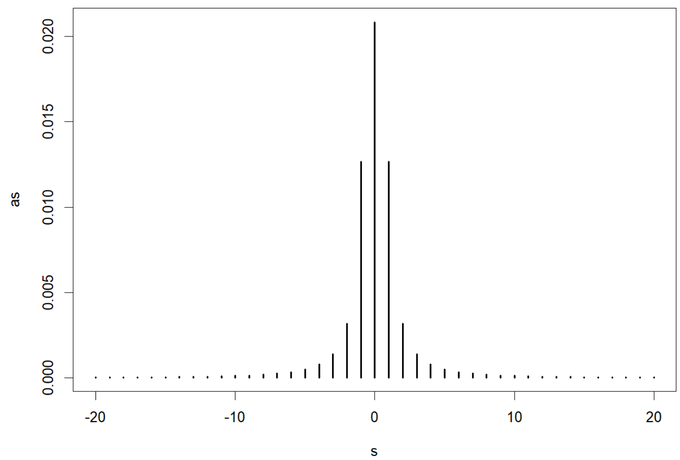

Figure 1:

A RScript for Exercise 1

Figure 2

B RScript for Exercise 2

Figure 3

```r
library(tseries)
library(fGarch)
prices <- scan(file.choose())

# 1) ARIMA models
# ADF tests
adf.test(prices, k=1)
adf.test(prices, k=2)
adf.test(prices, k=5)
adf.test(prices, k=10)

# ARIMA(0,1,1)
model1arima <- arima(prices,order=c(0,1,1))
res1arima <- model1arima$residuals
plot(res1arima)
Box.test(res1arima, lag=10, type="Ljung")
model1arima$aic

# ARIMA(0,1,2)
model2arima <- arima(prices,order=c(0,1,2))
res2arima <- model2arima$residuals
plot(res2arima)
Box.test(res2arima, lag=10, type="Ljung")
model2arima$aic

# 2) ARCH models
model1arch <- garchFit(res2arima~garch(1,0),data=res2arima,trace=F)
summary(model1arch)

model2arch <- garchFit(res2arima~garch(2,0),data=res2arima,trace=F)
summary(model2arch)

model3arch <- garchFit(res2arima~garch(3,0),data=res2arima,trace=F)
summary(model3arch)

model4arch <- garchFit(res2arima~garch(4,0),data=res2arima,trace=F)
summary(model4arch)

model5arch <- garchFit(res2arima~garch(5,0),data=res2arima,trace=F)
summary(model5arch)

# 3) GARCH models
model1garch <- garchFit(res2arima~garch(1,1),data=res2arima,trace=F)
summary(model1garch)

model2garch <- garchFit(res2arima~garch(1,2),data=res2arima,trace=F)
summary(model2garch)

model3garch <- garchFit(res2arima~garch(1,3),data=res2arima,trace=F)
summary(model3garch)
```

Figure 2:

```r
M <- 2048
omega <- c(0:(M-1))/M
gamma <- ((omega - 0.5)^2) / 4
as <- Re(fft(gamma, inverse = TRUE))/M
as_centered <- c(as[(M-20+1):M], as[1:21])
s <- -20:20
plot(s, as_centered, ylab="as", type="h", lwd=2)
```

Figure 3: