Time Series Econometrics Tutorial 3 2019

Exercise 1:

$$(\phi)L \; = 1 - \phi_1 L - \phi_2 L^2 = \left(1 - \frac{L}{z_1}\right)\left(1 - \frac{L}{z_2}\right)$$
$$= 1 - \frac{L(z_2 + z_{1)}}{z_1 z_2} + \frac{L^2}{z_1 z_2}$$

$$z_1 + z_2 = \frac{-\phi_1 + \sqrt{\phi_1^2 + 4\phi_2}}{2\phi_2} + \frac{-\phi_1 - \sqrt{\phi_1^2 + 4\phi_2}}{2\phi_2} = \frac{-2\phi_1}{2\phi_2} = \frac{-\phi_1}{\phi_2}$$

$$z_1 * z_2 = \frac{-\phi_1 + \sqrt{\phi_1^2 + 4\phi_2}}{2\phi_2} * \frac{-\phi_1 - \sqrt{\phi_1^2 + 4\phi_2}}{2\phi_2}$$
$$= \frac{1}{4\phi_2^2} * \phi_1^2 - \phi_1^2 - 4\phi_2 = \frac{-4\phi_2}{4\phi_2^2} = \frac{-1}{\phi_2}$$

$$= 1 - \frac{L\left(\frac{-\phi_1}{\phi_2}\right)}{\left(\frac{-1}{\phi_2}\right)} + \frac{L^2}{\left(\frac{-1}{\phi_2}\right)} = 1 - \frac{L - \phi_1}{-1} + L^2 \frac{\phi_2}{-1}$$

$$= 1 - \phi_1 L - \phi_2 L^2$$

Exercise 2:

Exercise 2. $z_t = 1.75 z_{t-1} - 0.5 z_{t-2} - 0.25 z_{t-3} + \varepsilon_t - 0.5\varepsilon_{t-1} + 0.25\varepsilon_{t-2}$

lag polynomial: $1 - 1.75z + 0.5z^2 + 0.25z^3 = 0$

$z = 1$ if the above were to $= 0$

$z - 1$ is divisor.

$\frac{1}{4}(z^2 + 3z - 4)$

$z - 1 \,\big)\, \frac{1}{4}(z^3 + 2z^2 - 7z + 4)$

$\frac{1}{4}(z^3 - z^2)$ $-$

$0 + 0 + 3z^2 - 7z + 4$

$3z^2 - 3z$ $-$

$-4z + 4$

$-4z + 4$ $-$

0

FINAL

remainder

$(z-1) \times \frac{1}{4} \times (z^2 + 3z - 4)$

$(z-1) \cdot \frac{1}{4} \cdot (z+4)(z-1)$ $p = -4$, $s = 3$ $4, -1$

$(z-1)^2 \cdot \left(\frac{z}{4} + 1\right)$

$\frac{z}{4} = -1$

$z = 1, 1$ $z = -4$

$\therefore$ ARIMA$(1, 2, 2)$

ARMA(3,2) Δ^2 ~~$z_t = 1.75 z_{t-1} - 0.5 z_{t-2} - 0.25 z_{t-3} + \varepsilon_t - 0.5\varepsilon_{t-1} + 0.25\varepsilon_{t-2}$~~.

~~$\Delta\Delta x_t = \Delta(1.75 z_{t-1} - 0.5z$~~

Note AR - Intregrater Order

Δ backwards.

Exercise 3:

R code:

```
#Exercise 3
DFCritVals <- function(samplesize) {
  #Values
  ba0.01 <- matrix(c(-2.56574, -2.2358, -3.627, 0),1,4)
  ba0.05 <- matrix(c(-1.94100, -0.2686, -3.365, 31.223),1,4)
  ba0.10 <- matrix(c(-1.61682, 0.2656, -2.714, 25.364),1,4)
  operator <- matrix(c(1, 1/(samplesize), 1/(samplesize^2),
1/(samplesize^3)),4,1)

  #Mechanics
  critstat0.01 <- ba0.01%*%operator
  critstat0.05 <- ba0.05%*%operator
  critstat0.10 <- ba0.10%*%operator

  #Output
  critstats <- round(c(critstat0.01, critstat0.05,
critstat0.10),4)
  return(critstats)
}
```

Exercise 4:

H_0: unit root $(\pi) = 0$
H_A: not a unit root $(\pi) < 0$

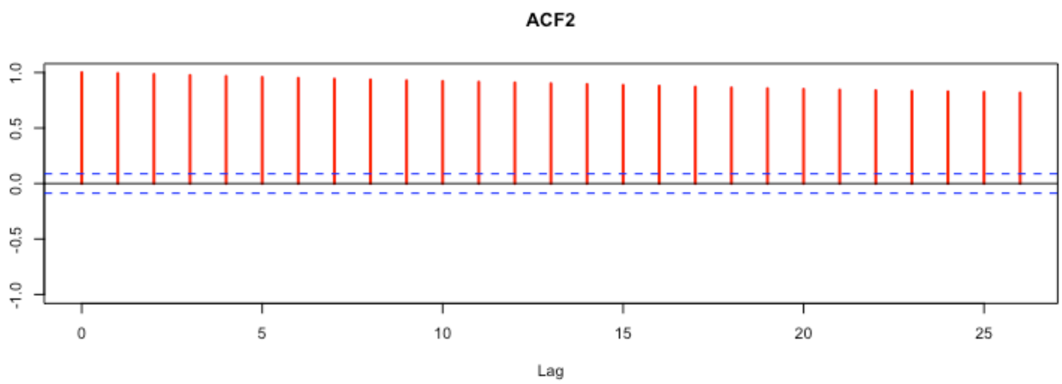
Time Series	Test-statistic	Critical Value (10pct)	P - value	Significant? (i.e. is the test-stat > the critical value)
x1_t	-9.4891	-1.62	<2.2e-16	Yes
x2_t	-1.1611	-1.62	0.2441	No
x3_t	-39.8115	-1.62	<2.2e-16	Yes
x4_t	-16.3106	-1.62	<2.2e-16	Yes
x5_t	-6.9297	-1.62	1.311e-11	Yes

x2_t is the unit root process. Looking at the statistical test (the Dickey-Fuller test), the test-statistic is greater than the critical value, for a one-sided test. This means that at the 10% significant level, $\widehat{(\pi)}$ is not statistically significant. Hence, we cannot reject the null hypothesis of a unit root.

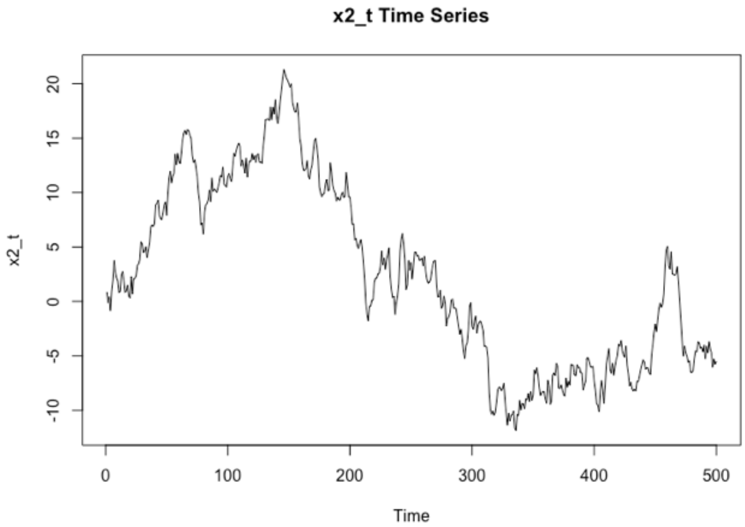
In addition to the t-test, the p-value for x2_t is also insignificant, confirming that we cannot reject the hypothesis of a unit root.

When the 1% critical value (-2.58) is taken into consideration, the process x5_t still results in us not rejecting the null hypothesis.

To further prove that x2_t is a unit root process, we look at the ACF function and compare it to the other ACF's (I will only put the ACF for x2_t to save from a large file size). It is spiking, but all of the values are significant. This means that the process is required to be integrated. If integration is required, then there are definitely unit roots.



Lastly, if we plot the process itself, x2_t exhibits a behaviour that does not resemble white noise, therefore it must be either a unit root process or a non-stationary process.



Considering the above results, it is certain that the process x2_t is a unit root process.

R code:

```
#Tutorial 3 R Code Luke Whitehill
#Exercise 4:
library(urca)

data1<-read.xlsx("Tutorial 3.xlsx", "Exercise 4", colIndex =
2)
data2<-read.xlsx("Tutorial 3.xlsx", "Exercise 4", colIndex =
3)
data3<-read.xlsx("Tutorial 3.xlsx", "Exercise 4", colIndex =
4)
data4<-read.xlsx("Tutorial 3.xlsx", "Exercise 4", colIndex =
5)
data5<-read.xlsx("Tutorial 3.xlsx", "Exercise 4", colIndex =
6)

x1<-ts(data = data1, start = 1, end = 500, frequency = 1)
x2<-ts(data = data2, start = 1, end = 500, frequency = 1)
x3<-ts(data = data3, start = 1, end = 500, frequency = 1)
x4<-ts(data = data4, start = 1, end = 500, frequency = 1)
x5<-ts(data = data5, start = 1, end = 500, frequency = 1)


summary(ur.df(x1, type="none", lags=0))
summary(ur.df(x2, type="none", lags=0))
summary(ur.df(x3, type="none", lags=0))
summary(ur.df(x4, type="none", lags=0))
summary(ur.df(x5, type="none", lags=0))

layout(matrix(c(1,1,2,2), ncol = 1))
acf2 <- acf(x1, main="ACF1", ylab="", ylim=c(-1, 1), lwd = 2,
col="red", ci.col="blue")
acf3 <- acf(x2, main="ACF2", ylab="", ylim=c(-1, 1), lwd = 2,
col="red", ci.col="blue")
acf4 <- acf(x3, main="ACF3", ylab="", ylim=c(-1, 1), lwd = 2,
col="red", ci.col="blue")
acf5 <- acf(x4, main="ACF4", ylab="", ylim=c(-1, 1), lwd = 2,
col="red", ci.col="blue")
acf6 <- acf(x5, main="ACF5", ylab="", ylim=c(-1, 1), lwd = 2,
col="red", ci.col="blue")


plot(x1, main = "x1_t Time Series")
plot(x2, main = "x2_t Time Series")
plot(x3, main = "x3_t Time Series")
plot(x4, main = "x4_t Time Series")
plot(x5, main = "x5_t Time Series")
```