

732A62 Lab 2

Emil K Svensson & Rasmus Holm

2017-09-27

Assignment 1

a)

```
library(astsa)
library(kernlab)
library(TSA)
library(forecast)

set.seed(12345)
AR3 <- arima.sim(1000, model = list(order = c(3,0,0),
                                     ar = c(0.8, -0.2, 0.1)))

## The theoretical
AR3.pacf <- pacf(AR3, plot=F)
AR3.data <- ts.intersect(xt = AR3, x1 = lag(AR3, 1), x2 = lag(AR3, 2), x3 = lag(AR3, 3))

AR.lm <- resid(lm(xt ~ x1 + x2, data = AR3.data))
AR.lm.lag3 <- resid(lm(x3 ~ x1 + x2, data = AR3.data))

AR3.pacf[3]

##
## Partial autocorrelations of series 'AR3', by lag
##
##      3
## 0.117
cor(AR.lm, AR.lm.lag3)

## [1] 0.1146076
```

b)

```
set.seed(12345)
AR2 <- arima.sim(100, model = list(order = c(2,0,0),
                                     ar = c(0.8, 0.1)))

ar2.yw <- ar(AR2, order.max = 2, method = "yw", aic = FALSE)
ar2.ols <- ar(AR2, order.max = 2, method = "ols", aic = FALSE)
ar2.mle <- arima(AR2, order = c(2,0,0), method = "ML")

ar2.yw
```

```
##
## Call:
## ar(x = AR2, aic = FALSE, order.max = 2, method = "yw")
##
## Coefficients:
##      1      2
## 0.8029 0.1037
##
## Order selected 2  sigma^2 estimated as 1.267
```

```
ar2.ols
```

```
##
## Call:
## ar(x = AR2, aic = FALSE, order.max = 2, method = "ols")
##
## Coefficients:
##      1      2
## 0.8067 0.1205
##
## Intercept: -0.04401 (0.1074)
##
## Order selected 2  sigma^2 estimated as 1.129
```

```
ar2.mle
```

```
##
## Call:
## arima(x = AR2, order = c(2, 0, 0), method = "ML")
##
## Coefficients:
##           ar1      ar2  intercept
##      0.7967 0.1189    0.8290
## s.e. 0.0992 0.1000    1.1385
##
## sigma^2 estimated as 1.126: log likelihood = -148.71, aic = 303.41
```

Yes, the theoretical value for

$$\phi_2$$

is inside the confidence-intervall for the ML estimate.

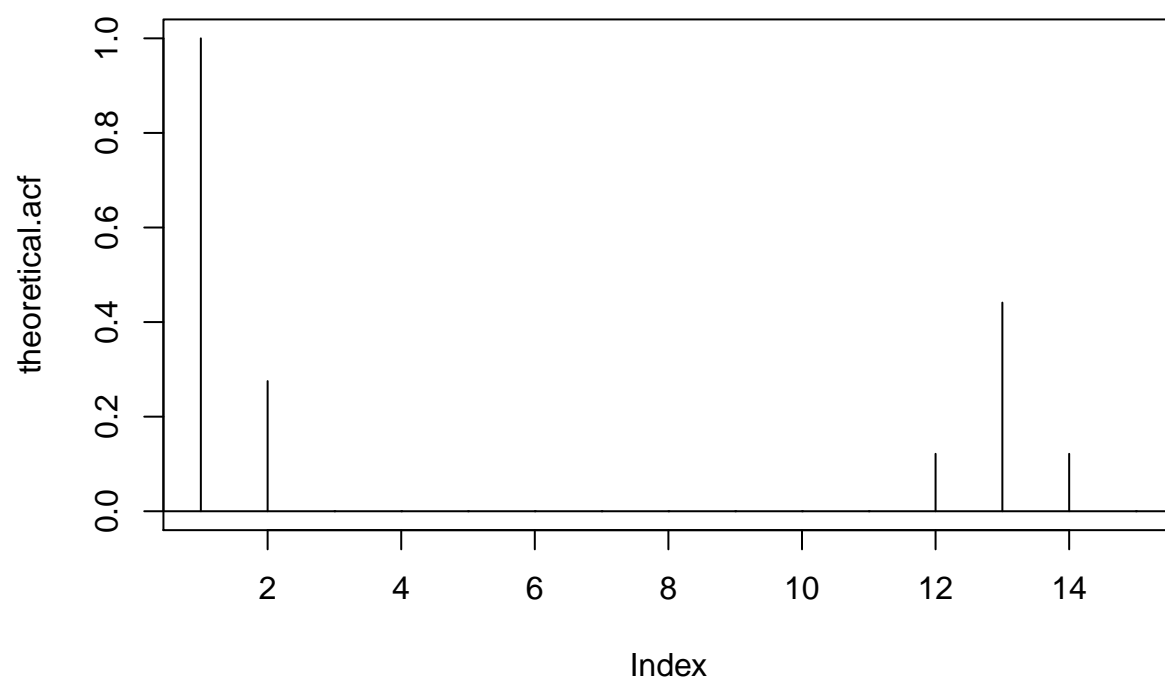
c)

```
set.seed(12345)
ma.coef <- c(0.3, rep(0, 10), 0.6)
ts4 <- arima.sim(n=200, model=list(order=c(0, 0, 12), ma = ma.coef))

theoretical.acf <- ARMAacf(ma=c(ma.coef, 0.3 * 0.6))
theoretical.pacf <- ARMAacf(ma=c(ma.coef, 0.3 * 0.6), pacf=TRUE)

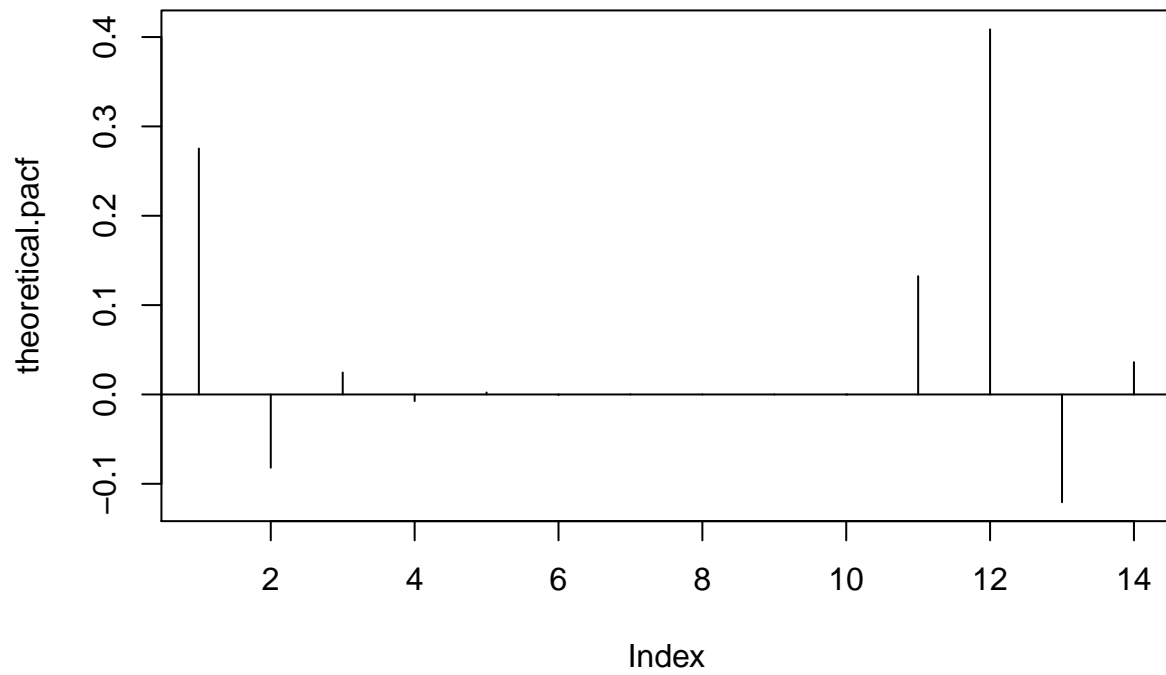
plot(theoretical.acf, type="h", main="Theoretical ACF")
abline(h=0)
```

Theoretical ACF

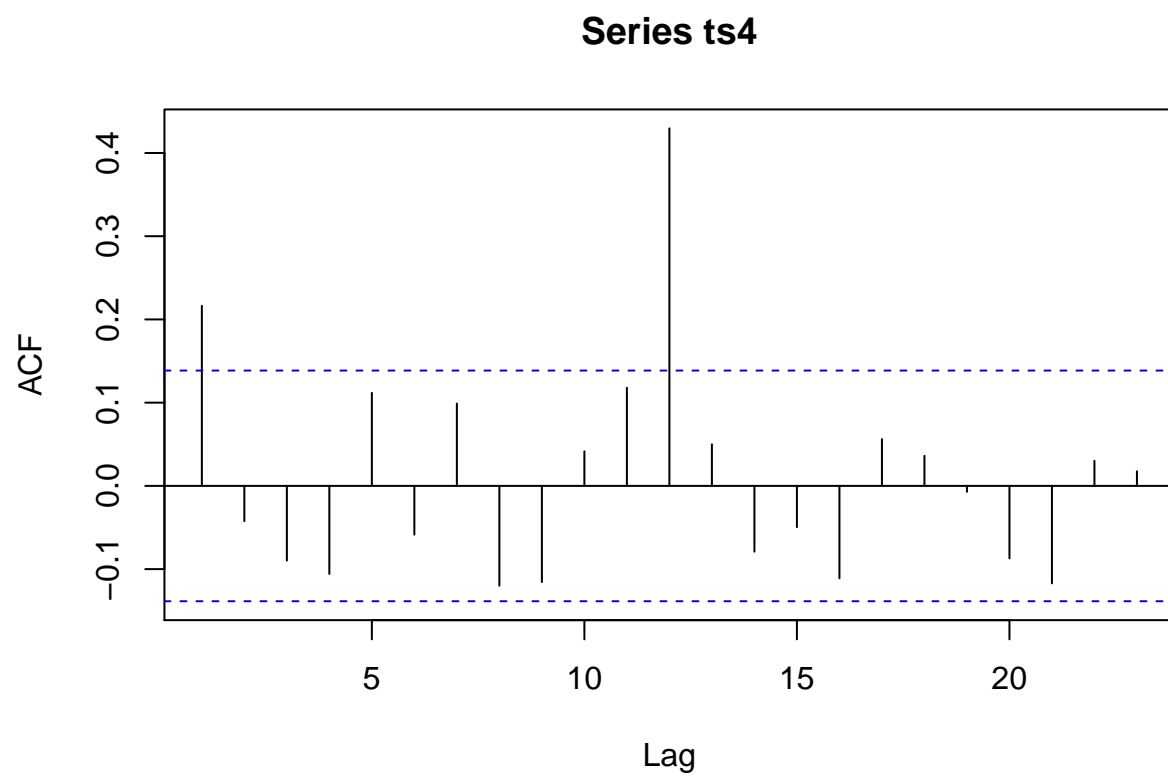


```
plot(theoretical.pacf, type="h", main="Theoretical PACF")  
abline(h=0)
```

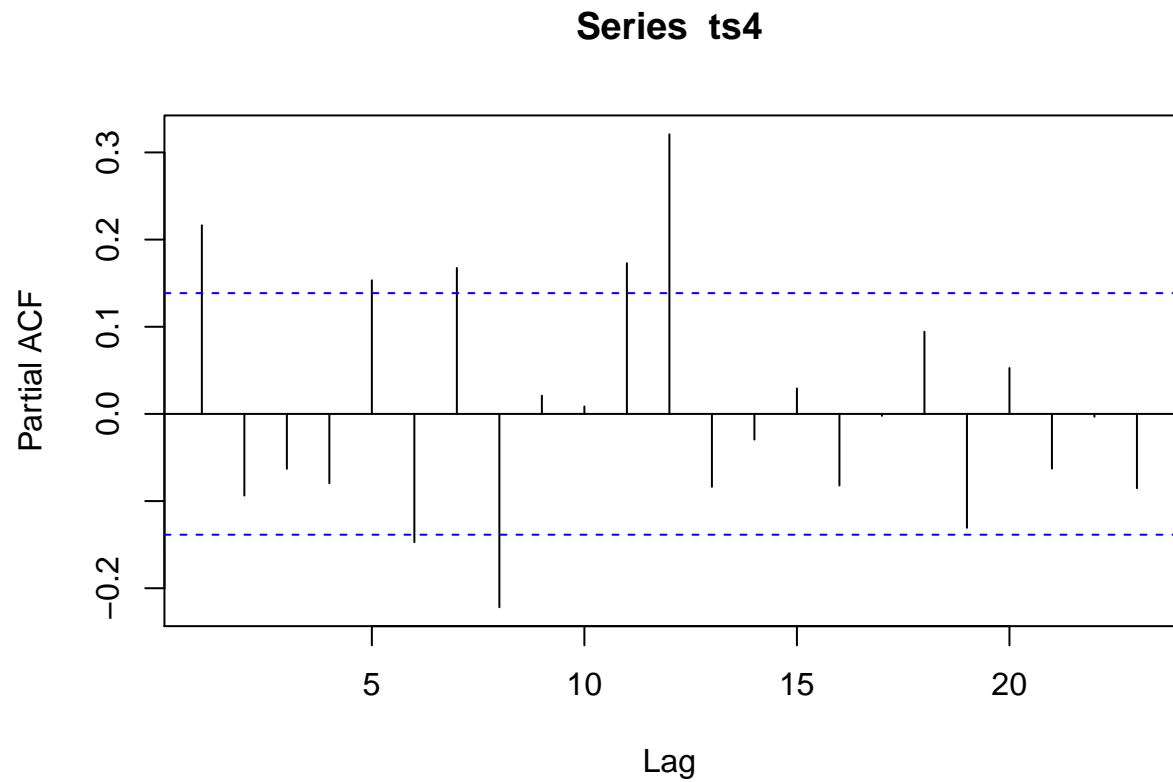
Theoretical PACF



```
empirical.acf <- acf(ts4)
```



```
empirical.pacf <- pacf(ts4)
```

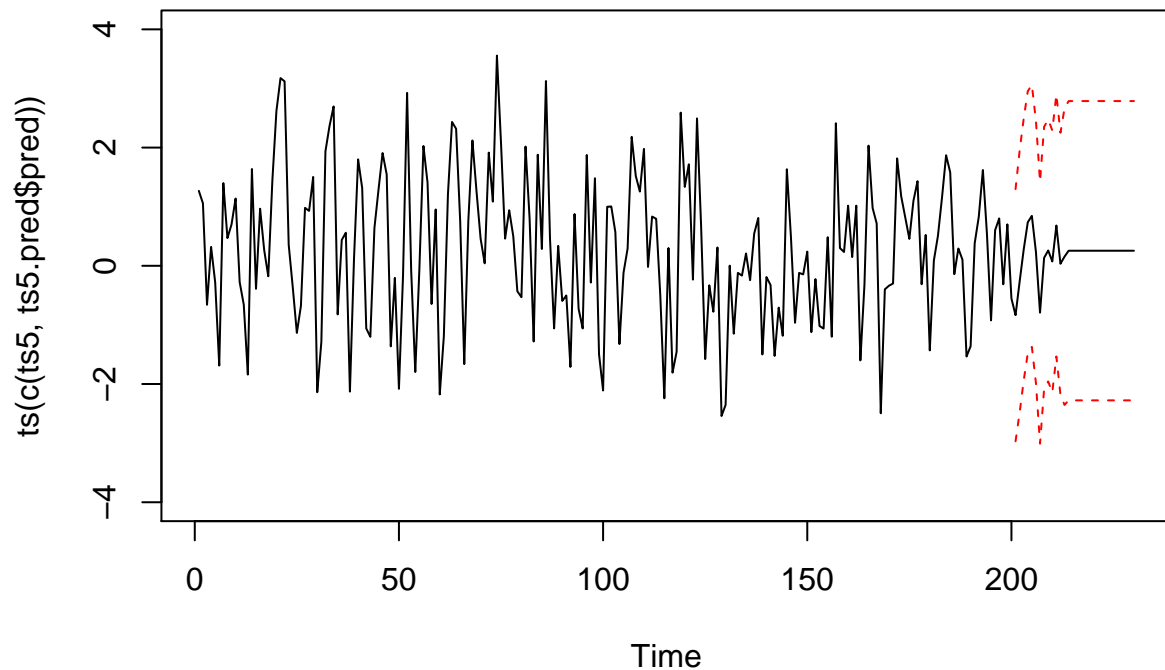


d)

```
set.seed(12345)
ma.coef <- c(0.3, rep(0, 10), 0.6)
ts5 <- arima.sim(n=200, model=list(order=c(0, 0, 12), ma = ma.coef))

ts5.fit <- arima(ts5, order=c(0, 0, 1), seasonal=list(order=c(0, 0, 1), period=12))
ts5.pred <- predict(ts5.fit, n.ahead=30, se.fit=TRUE)

plot(ts(c(ts5, ts5.pred$pred)), ylim=c(-4, 4))
lines(200 + 1:length(ts5.pred$pred), ts5.pred$pred + 1.96 * ts5.pred$se, lty=2, col="red")
lines(200 + 1:length(ts5.pred$pred), ts5.pred$pred - 1.96 * ts5.pred$se, lty=2, col="red")
```

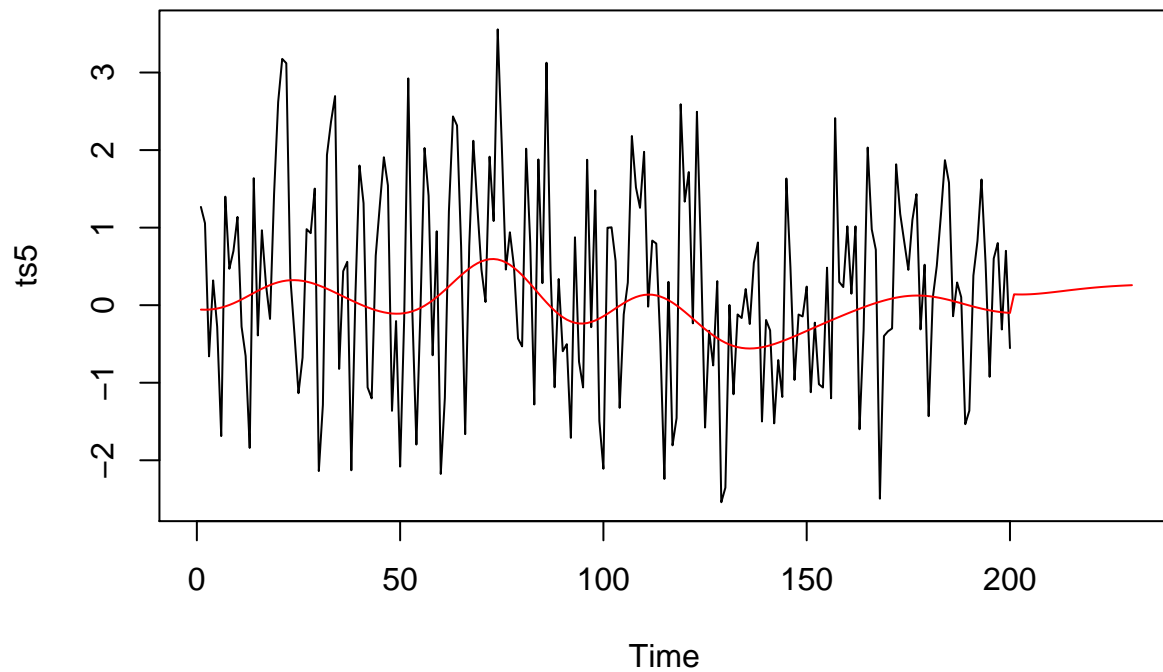


```
## gausspr.data <- ts.intersect(x=ts5, x1=lag(ts5, 1), x12=lag(ts5, 12), x13=lag(ts5, 13))
## gausspr.fit <- gausspr(x ~ ., gausspr.data)
## plot(ts5)
## lines(fitted(gausspr.fit), col="red", lwd=2)

gausspr.data <- data.frame(y=ts5, x=1:200)
gausspr.fit <- gausspr(y ~ x, gausspr.data)

## Using automatic sigma estimation (sigest) for RBF or laplace kernel
gausspr.pred <- predict(gfit, data.frame(x=201:230))

plot(ts5, xlim=c(0, 230))
lines(c(fitted(gausspr.fit), gausspr.pred), , col="red")
```



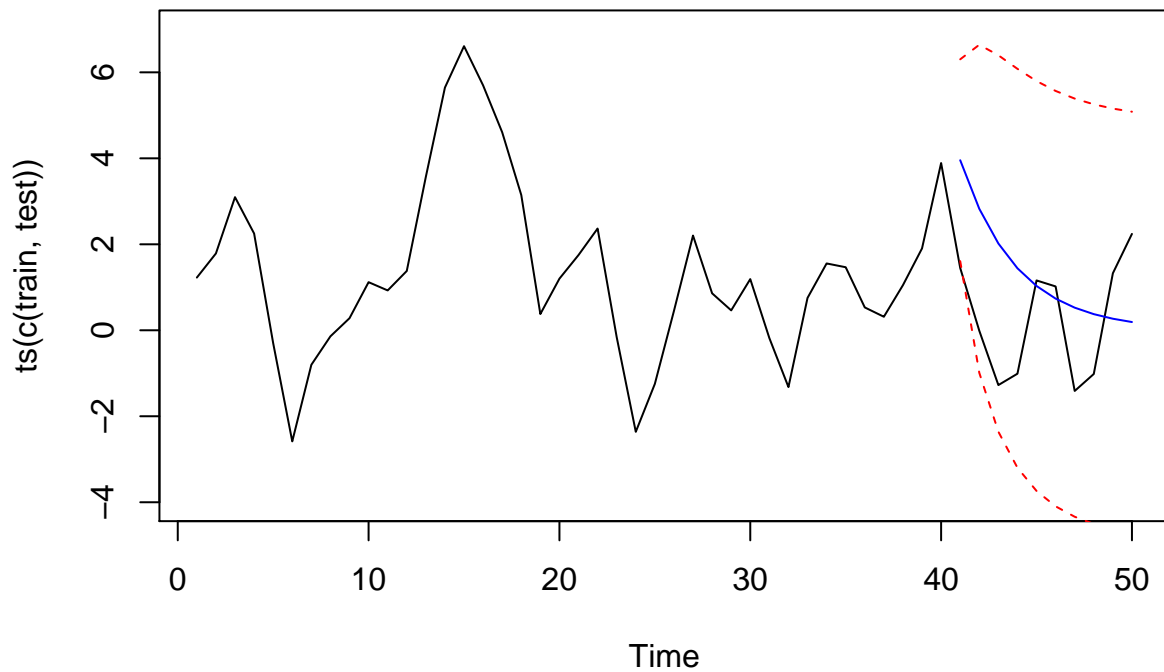
e)

```
set.seed(12345)
ts6 <- arima.sim(model=list(ma=c(0.5), ar=c(0.7)), n=50)

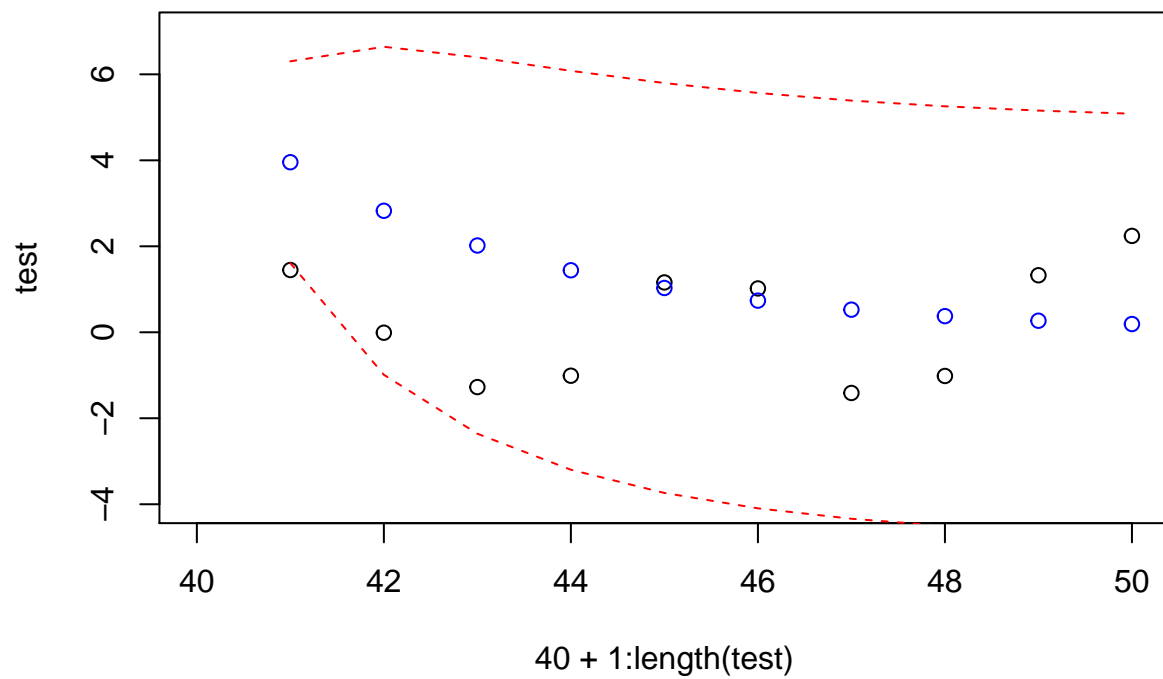
train <- ts(ts6[1:40])
test <- ts(ts6[41:50])

ts6.fit <- arima(train, order=c(1, 0, 1), include.mean = F)
ts6.pred <- predict(ts6.fit, n.ahead=10)

plot(ts(c(train, test)), ylim=c(-4, 7), type="l")
lines(40 + 1:length(test), ts6.pred$pred, col="blue")
lines(40 + 1:length(test), ts6.pred$pred + 1.96 * ts6.pred$se, lty=2, col="red")
lines(40 + 1:length(test), ts6.pred$pred - 1.96 * ts6.pred$se, lty=2, col="red")
```

```
plot(40 + 1:length(test), test, ylim=c(-4, 7), xlim=c(40, 50), type="p")
points(40 + 1:length(test), ts6.pred$pred, col="blue")
lines(40 + 1:length(test), ts6.pred$pred + 1.96 * ts6.pred$se, lty=2, col="red")
lines(40 + 1:length(test), ts6.pred$pred - 1.96 * ts6.pred$se, lty=2, col="red")
```

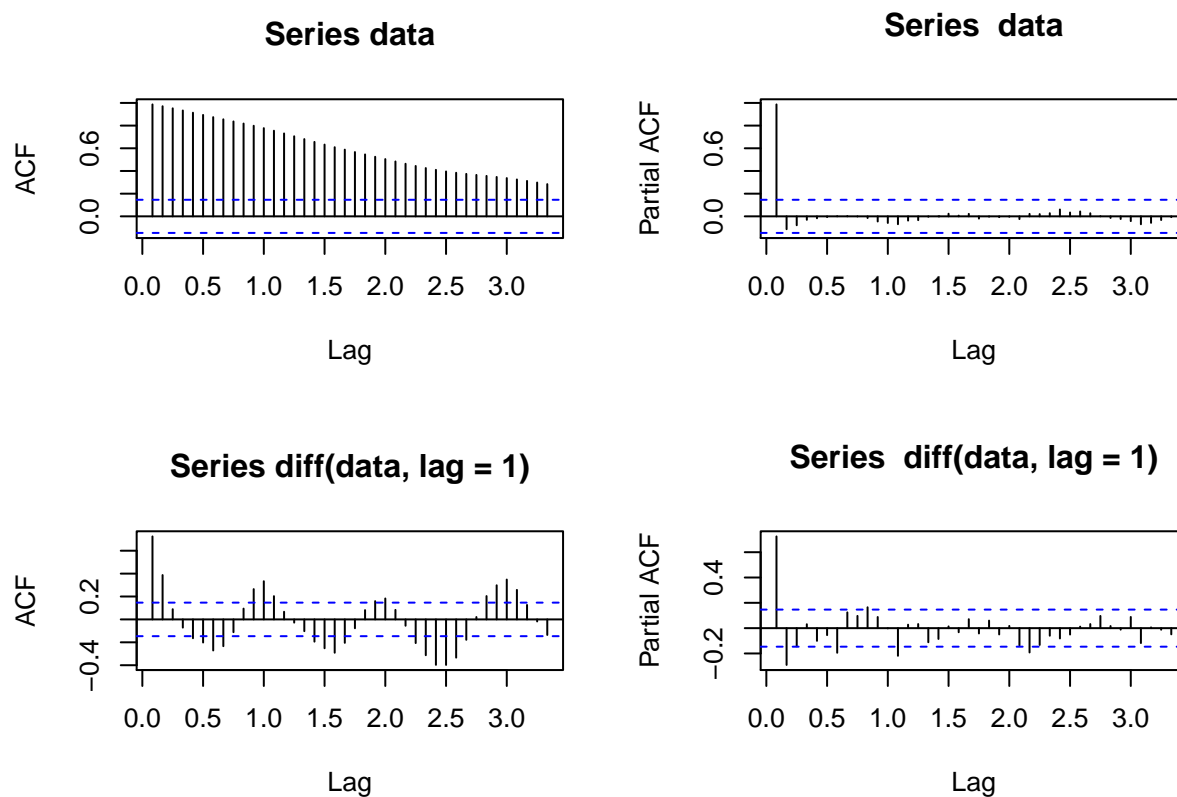


Assignment 2

```
assignment2 <- function(data){  
  old <- par(mfrow = c(2, 2))  
  acf(data, lag.max = 40)  
  pacf(data, lag.max = 40)  
  acf(diff(data, lag = 1), lag.max = 40)  
  pacf(diff(data, lag = 1), lag.max = 40)  
  par(old)  
}
```

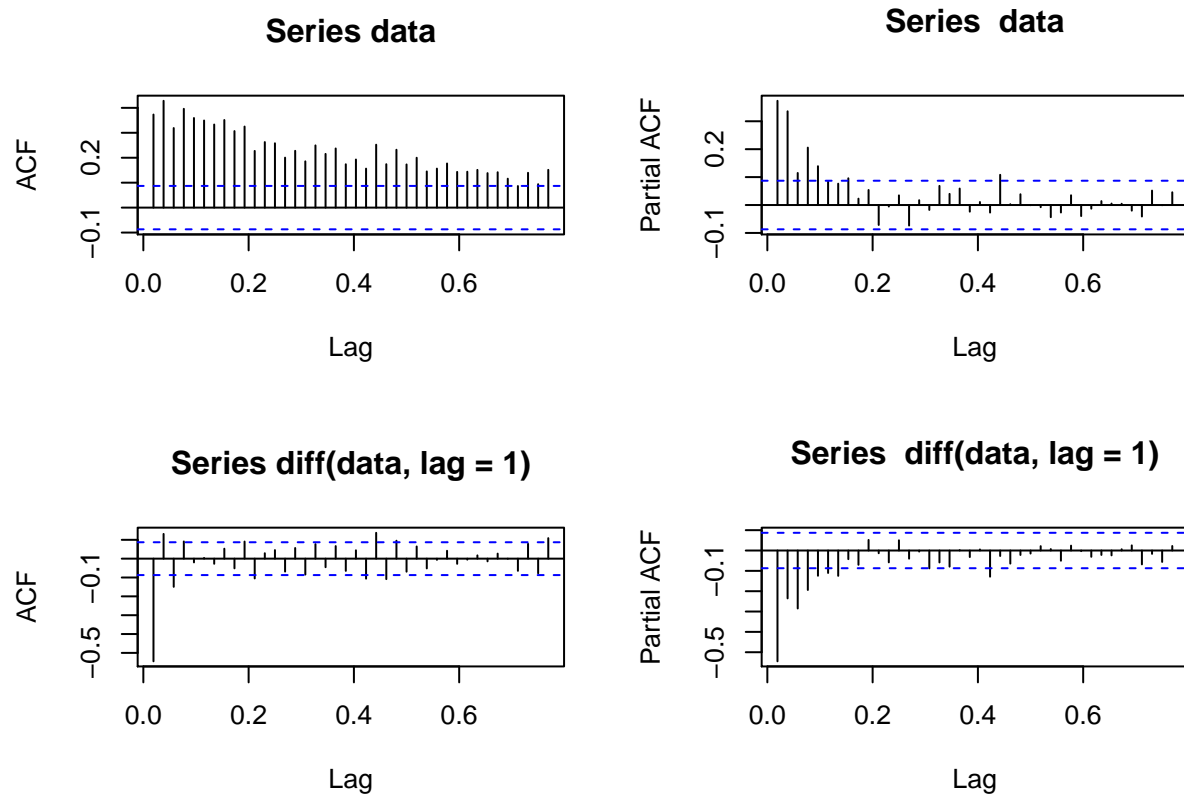
Chicken

```
assignment2(chicken)
```



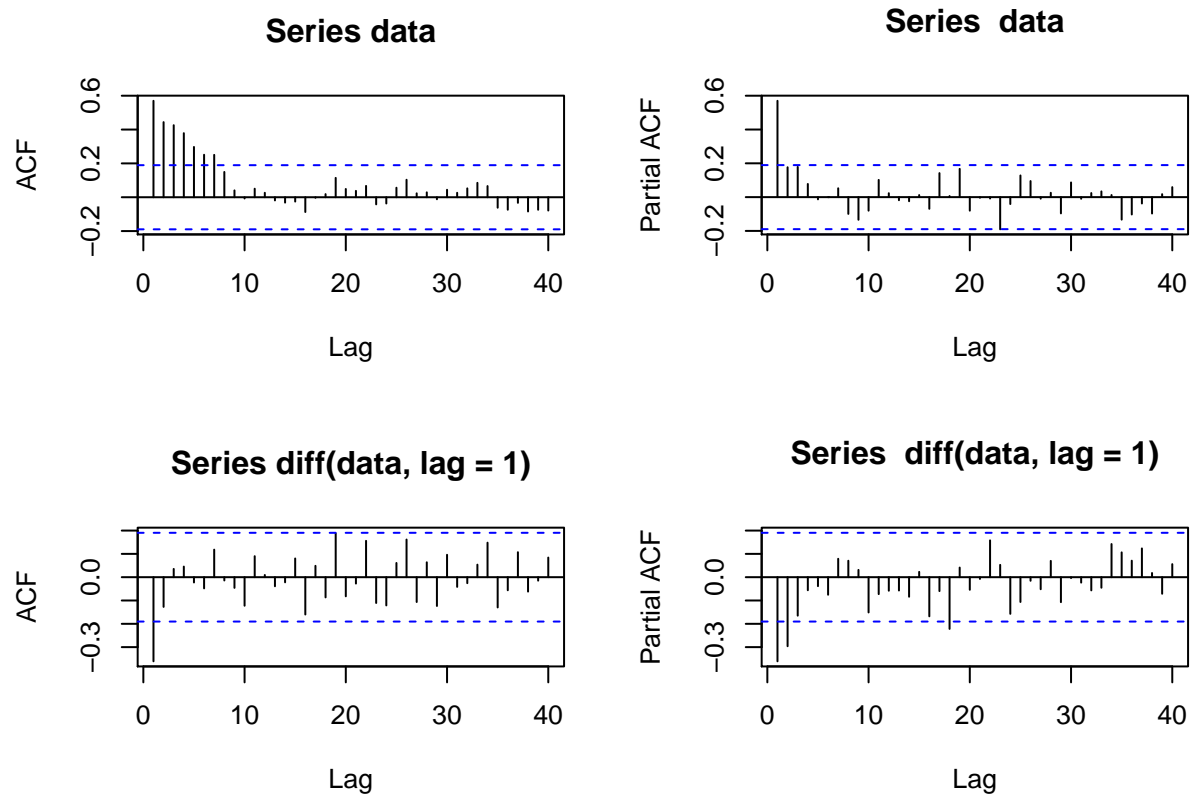
so2

```
assignment2(so2)
```



EQcount

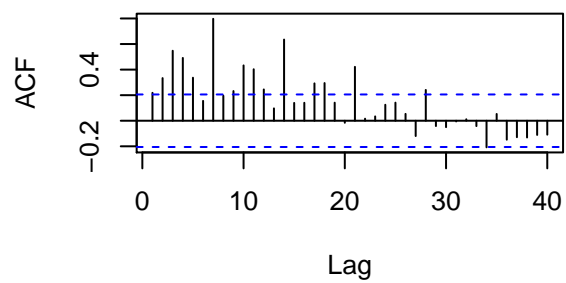
```
assignment2(EQcount)
```



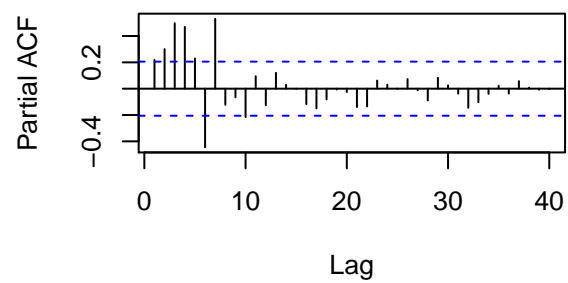
HCT

```
assignment2(HCT)
```

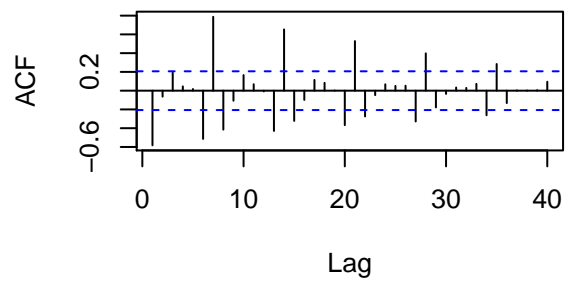
Series data



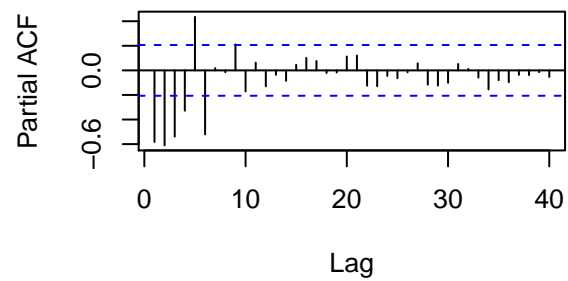
Series data



Series diff(data, lag = 1)



Series diff(data, lag = 1)



Assignment 3

```
plot_helper <- function(data, title) {
  old <- par(mfrow=c(4, 1))
  plot(data, main=title)
  acf(data, lag.max=40, main="")
  pacf(data, lag.max=40, main="")
  qqnorm(data, main="", las=1)
  qqline(data)
  par(old)
}

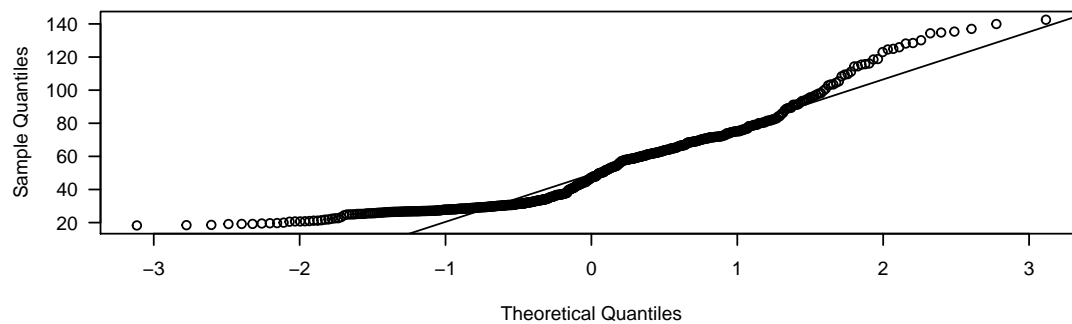
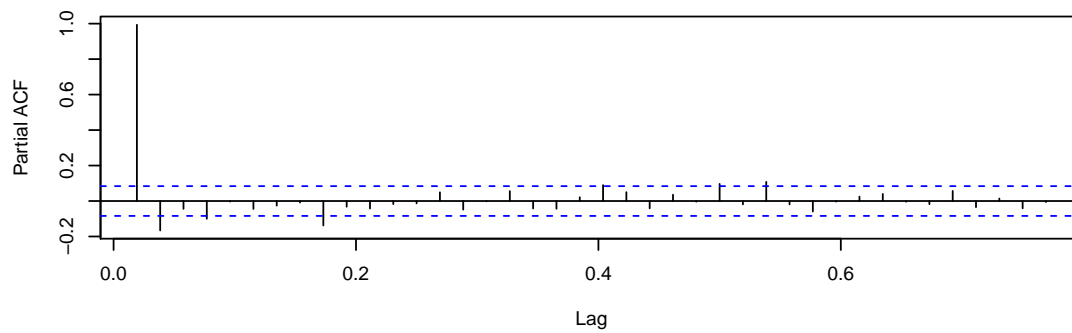
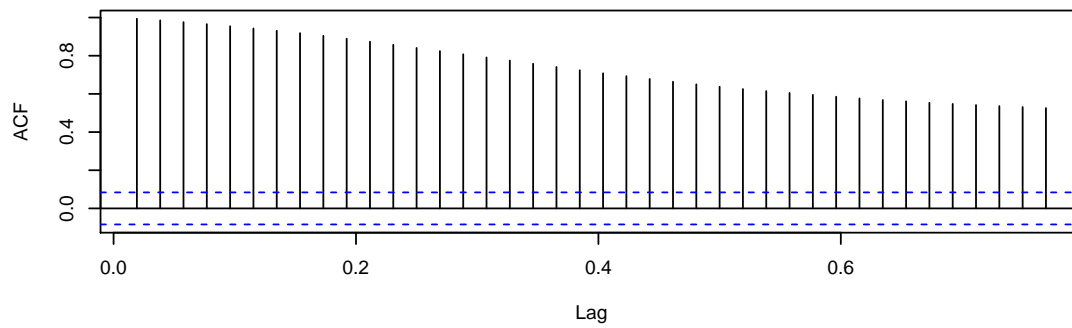
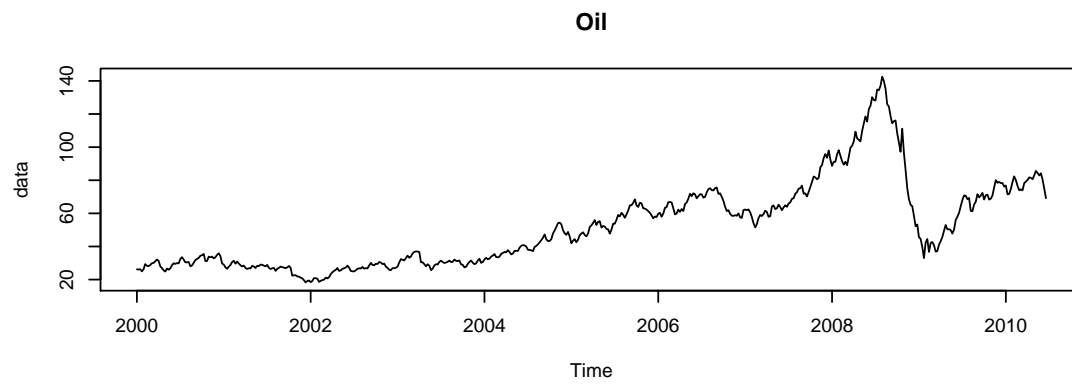
test_helper <- function(data) {
  print(Box.test(data, lag = 1, type = "Ljung-Box"))
  print(suppressWarnings(adf.test(data)))
  e <- eacf(data)
}

fit_plot <- function(model) {
  pred <- predict(model, n.ahead=20, se.fit=TRUE)
  upper_band <- pred$pred + 1.96 * pred$se
  lower_band <- pred$pred - 1.96 * pred$se

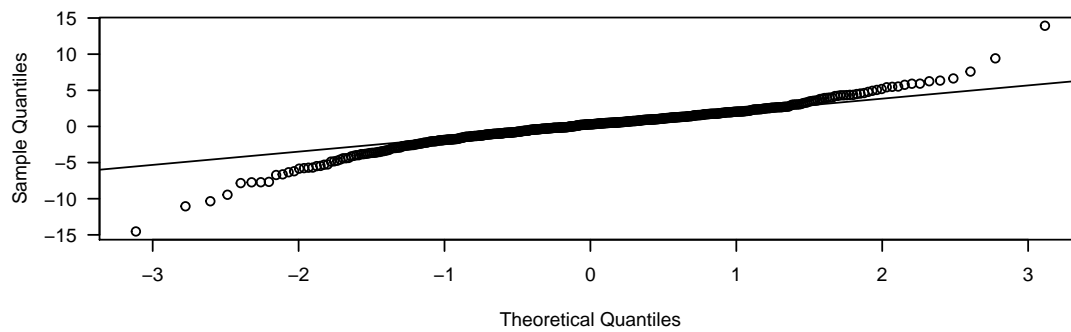
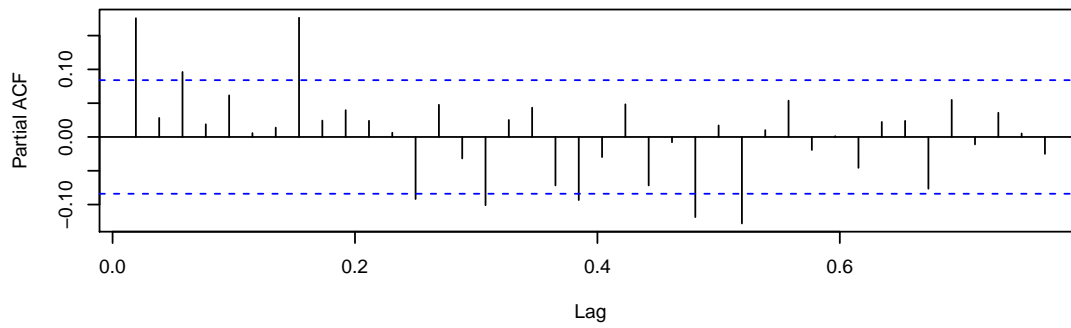
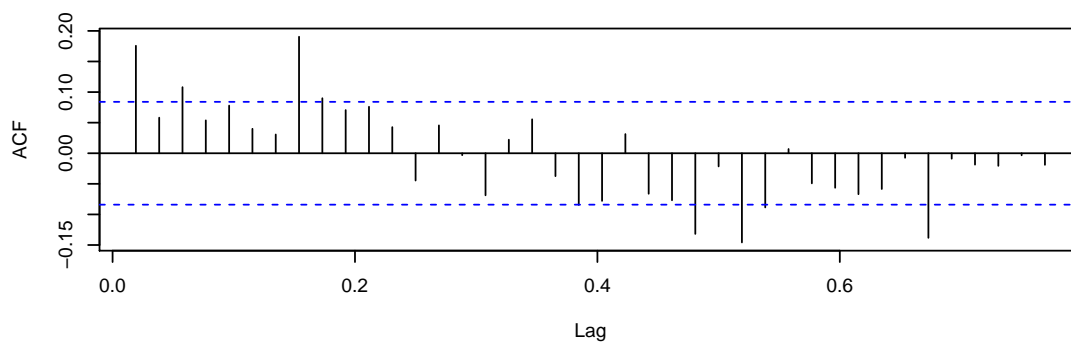
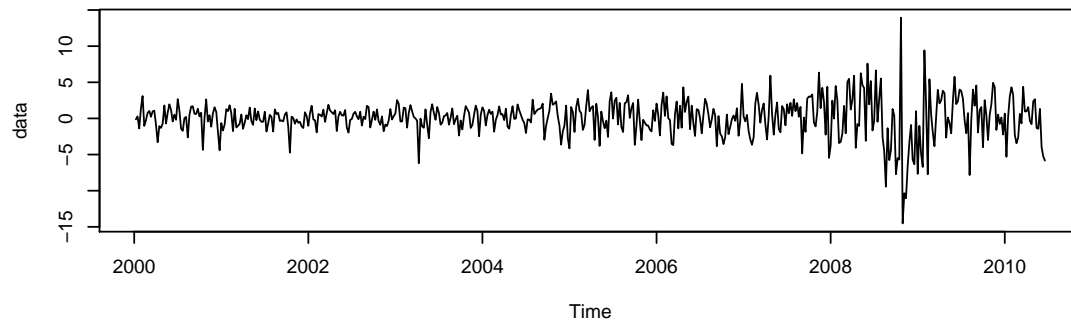
  plot(c(model$x, pred$pred), type="l", xlim=c(500, length(oil) + 20), ylim=c(min(lower_band), max(upper_band)),
       lines(length(oil) + 1:20, upper_band, lty=2, col="red")
       lines(length(oil) + 1:20, lower_band, lty=2, col="red"))
}
```

a)

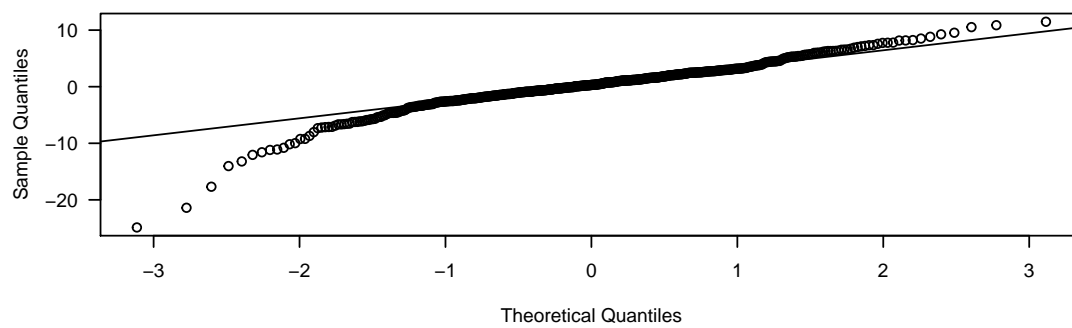
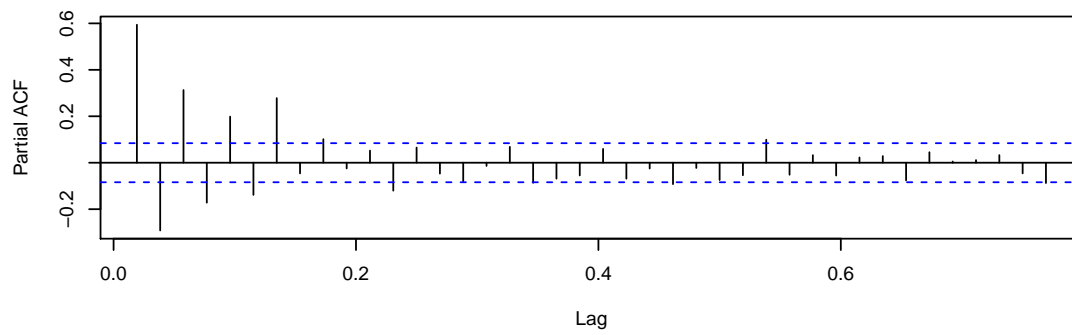
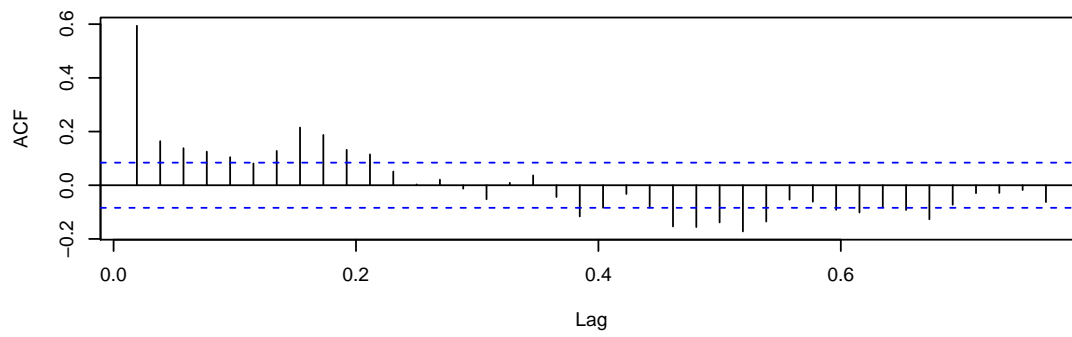
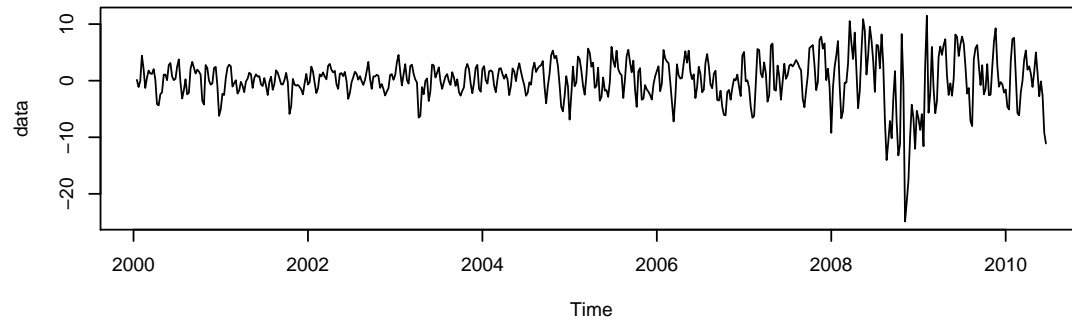
```
loil <- log(oil)
doil <- diff(oil)
ddoil <- diff(oil, 2)
dloil <- diff(loil)
ddloil <- diff(loil, 2)
```



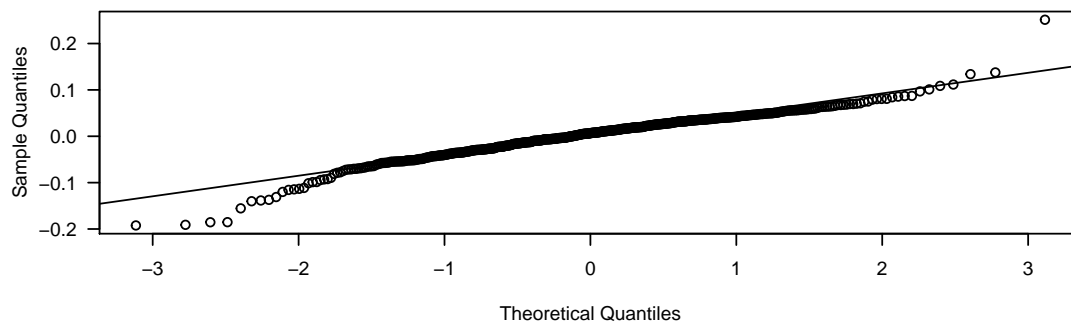
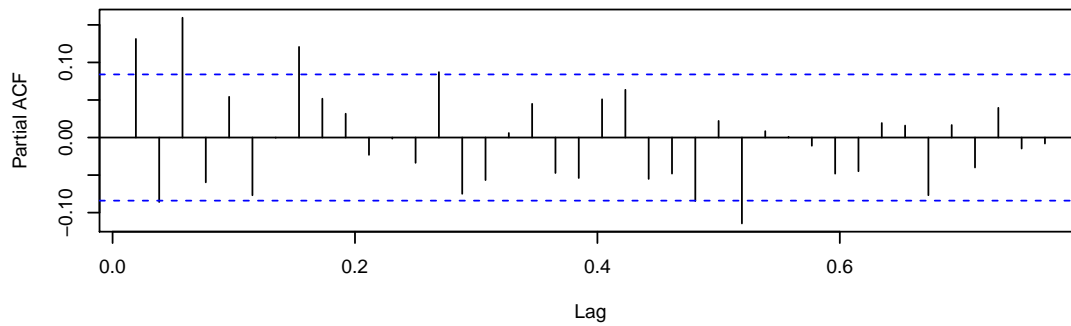
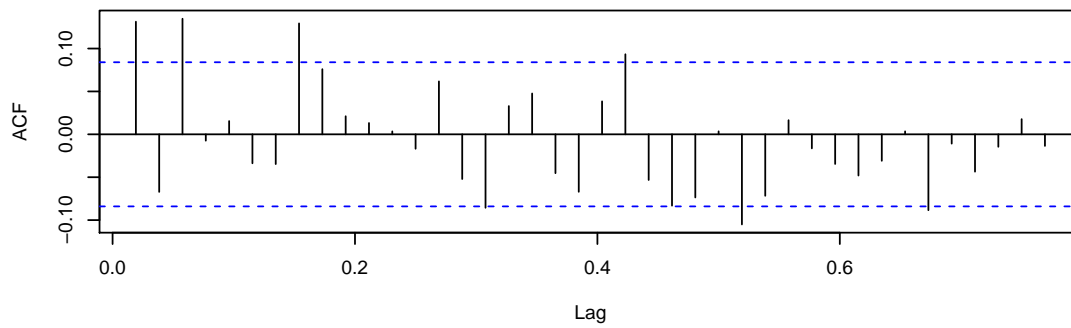
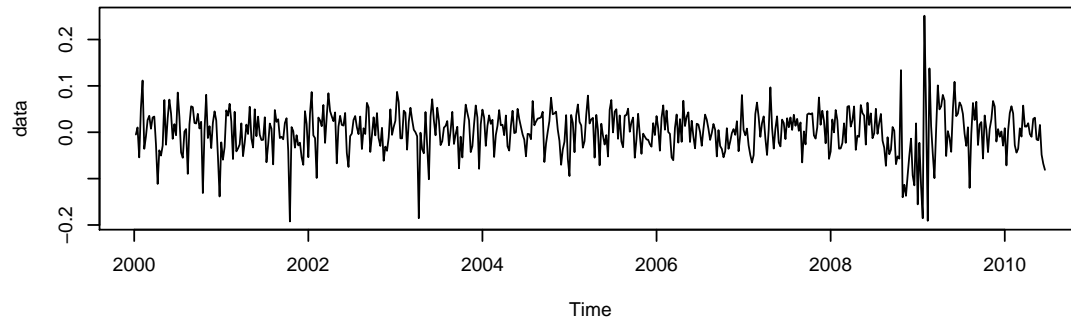
1 Difference Oil

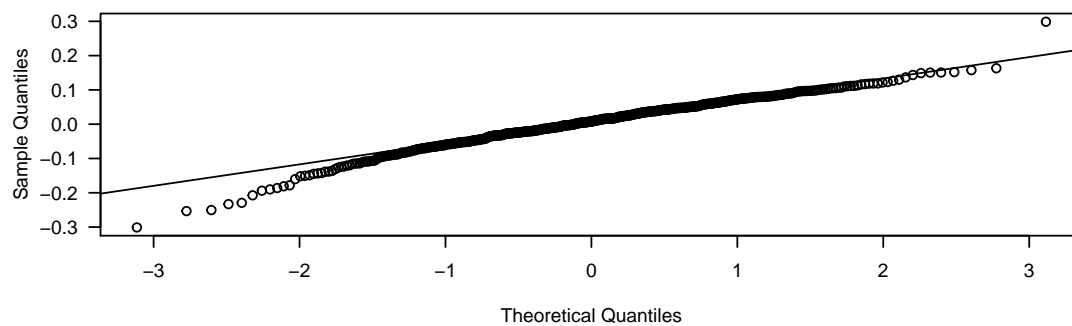
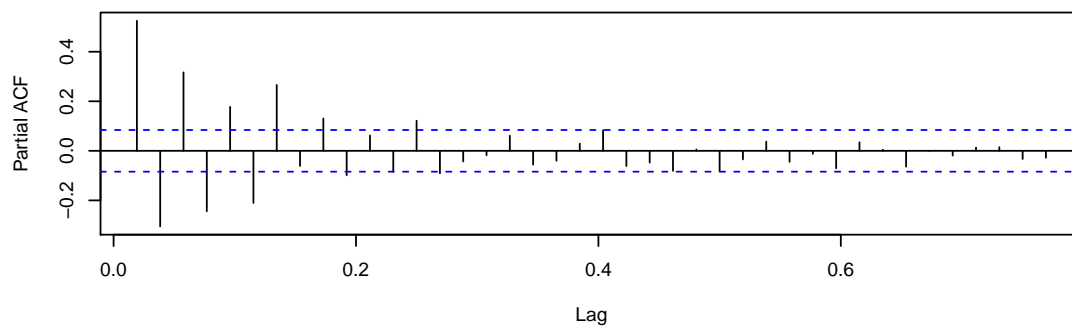
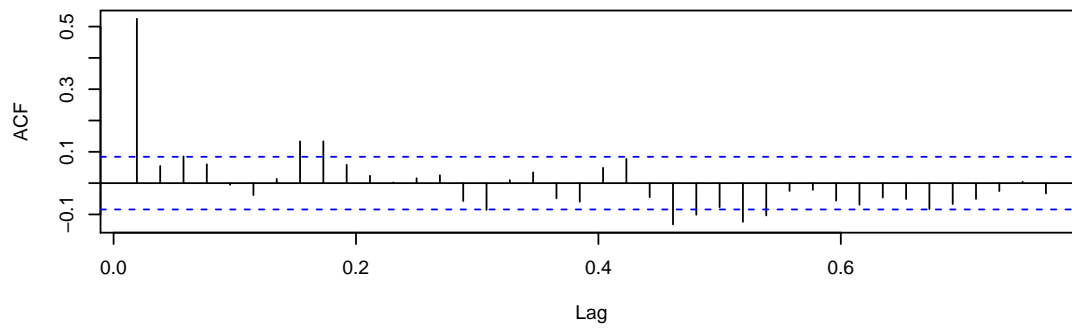
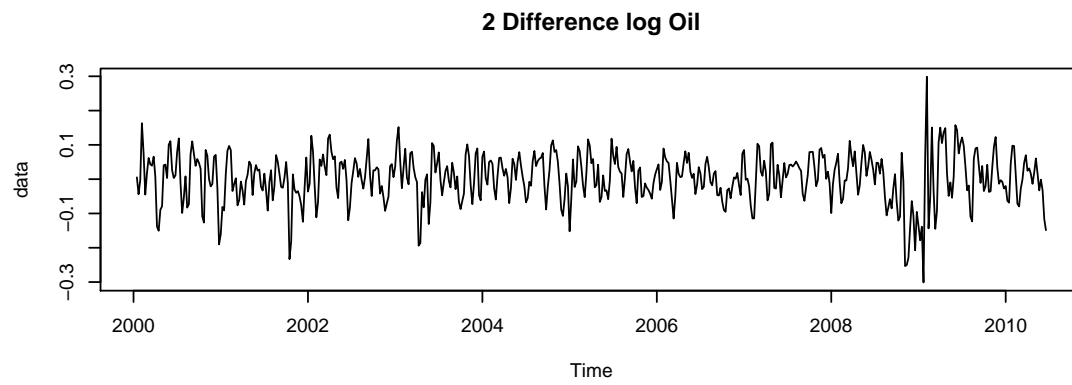


2 Difference Oil



1 Difference log Oil





```
test_helper(doil)
```

```
##
## Box-Ljung test
##
## data: data
## X-squared = 16.884, df = 1, p-value = 3.974e-05
##
## Augmented Dickey-Fuller Test
##
## data: data
## Dickey-Fuller = -5.3269, Lag order = 8, p-value = 0.01
## alternative hypothesis: stationary
##
## AR/MA
## 0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x o x o o o o x x o o o o o
## 1 x o o o o o o x o o o o o o
## 2 x x o o o o o x o o o o o o
## 3 x x x o o o o x o o o o o o
## 4 x x x o o o o x o o o o o o
## 5 x x o o o o o x o o o o o o
## 6 x o x o x o o x o o o o o o
## 7 o o x o x o x x o o o o o o
```

```
test_helper(ddoil)
```

```
##
## Box-Ljung test
##
## data: data
## X-squared = 192.72, df = 1, p-value < 2.2e-16
##
## Augmented Dickey-Fuller Test
##
## data: data
## Dickey-Fuller = -4.7773, Lag order = 8, p-value = 0.01
## alternative hypothesis: stationary
##
## AR/MA
## 0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x x x x x o x x x x x o o o
## 1 x x o o o o o x o o o o o o
## 2 x x x o o o o x x o x o o o
## 3 x x x o o o o x o o x o o o
## 4 x x o o o o o x x o o o o o
## 5 x x o x x x o x o o o o o o
## 6 x x o x x x x x o o o o o o
## 7 x x o x x x o x o x o o o o
```

```
test_helper(dloil)
```

```
##
```

```

## Box-Ljung test
##
## data: data
## X-squared = 9.4307, df = 1, p-value = 0.002134
##
##
## Augmented Dickey-Fuller Test
##
## data: data
## Dickey-Fuller = -6.3708, Lag order = 8, p-value = 0.01
## alternative hypothesis: stationary
##
## AR/MA
##  0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x o x o o o o x o o o o o o
## 1 x o x o o o o x o o o o o o
## 2 x x x o o o o x o o o o o o
## 3 x x x o o o o x o o o o o o
## 4 x o x o o o o x o o o o o o
## 5 x x x o x o o x o o o o o o
## 6 o x x o x x o x o o o o o x
## 7 o x x x x x x x o x o o o o

```

```
test_helper(ddloil)
```

```

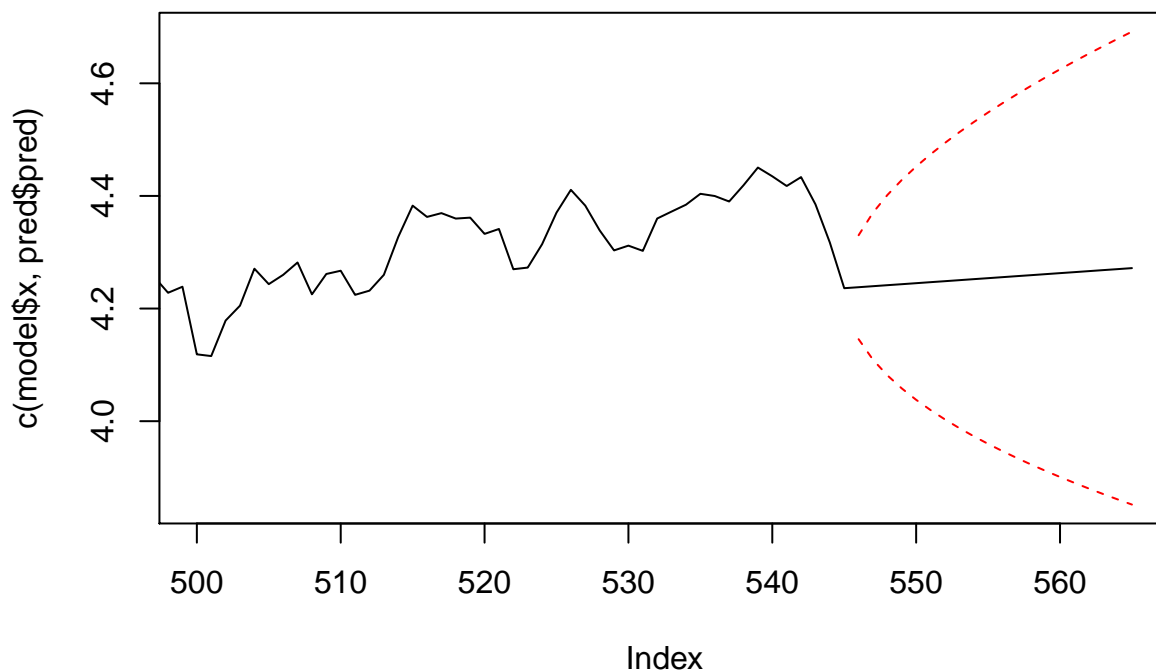
##
## Box-Ljung test
##
## data: data
## X-squared = 150.51, df = 1, p-value < 2.2e-16
##
##
## Augmented Dickey-Fuller Test
##
## data: data
## Dickey-Fuller = -5.6251, Lag order = 8, p-value = 0.01
## alternative hypothesis: stationary
##
## AR/MA
##  0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x o o o o o o x x o o o o o
## 1 x x o o o o o x x o o o o o
## 2 x x x o o o o x x o o o o o
## 3 x x x x o o o x x o o o o o
## 4 x x x x o o o x x o o o o o
## 5 x o x x x o o x x o o o o o
## 6 x o x x x x x x o x o o o o
## 7 x x x x x x x x o x o o o o

```

```
fit1 <- Arima(loil, order=c(0, 2, 1))
fit1
```

```
## Series: loil
## ARIMA(0,2,1)
##
## Coefficients:
##          ma1
##        -1.0000
## s.e.    0.0061
##
## sigma^2 estimated as 0.002213:  log likelihood=886.63
## AIC=-1769.26   AICc=-1769.24   BIC=-1760.67
```

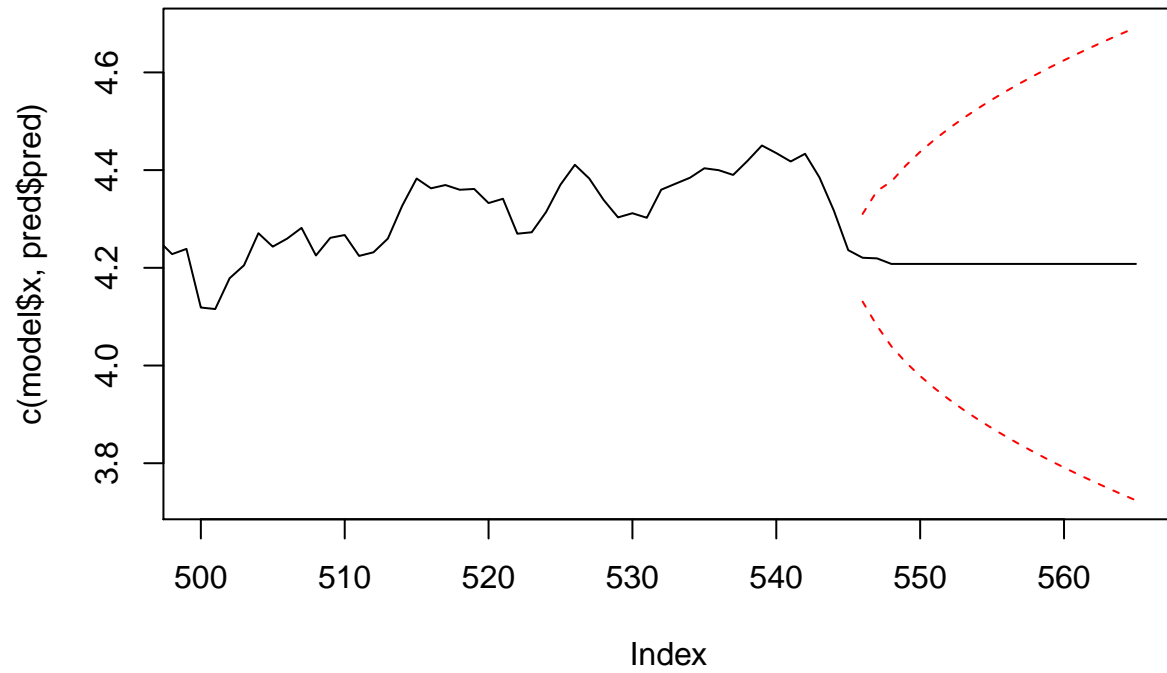
```
fit_plot(fit1)
```

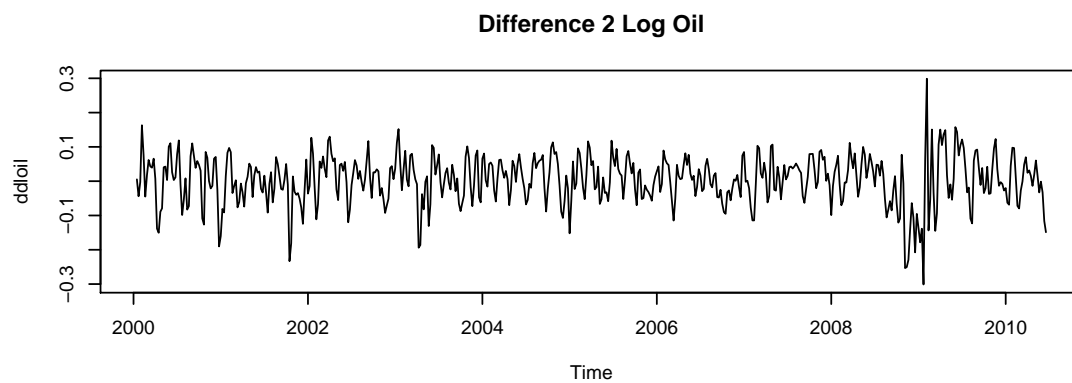
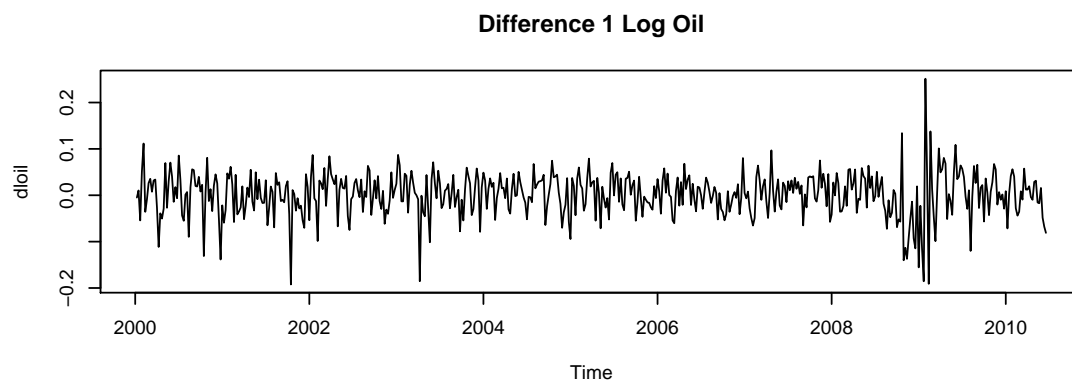
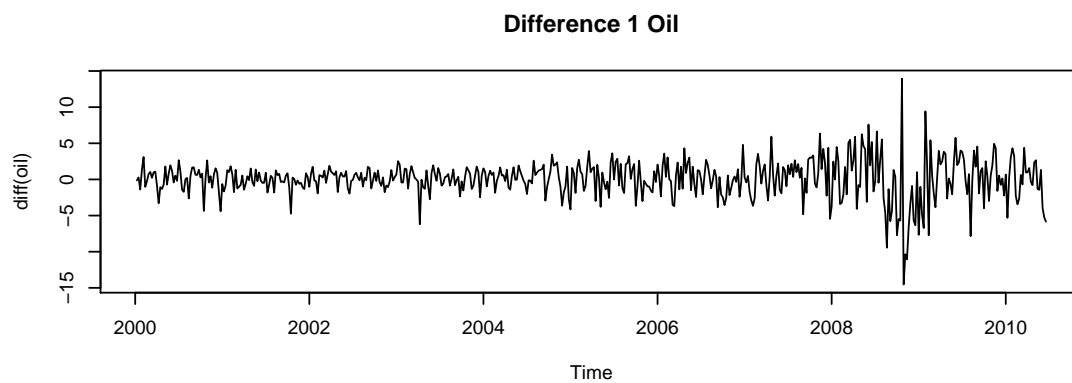
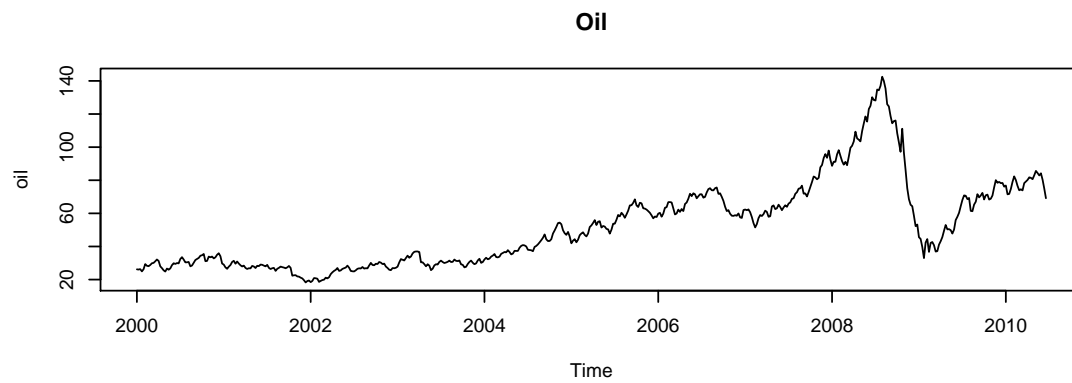


```
fit2 <- Arima(loil, order=c(0, 1, 3))
fit2
```

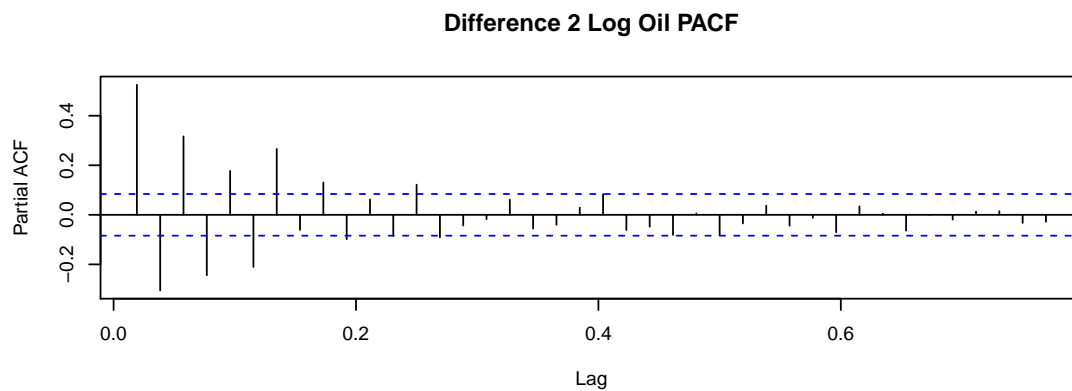
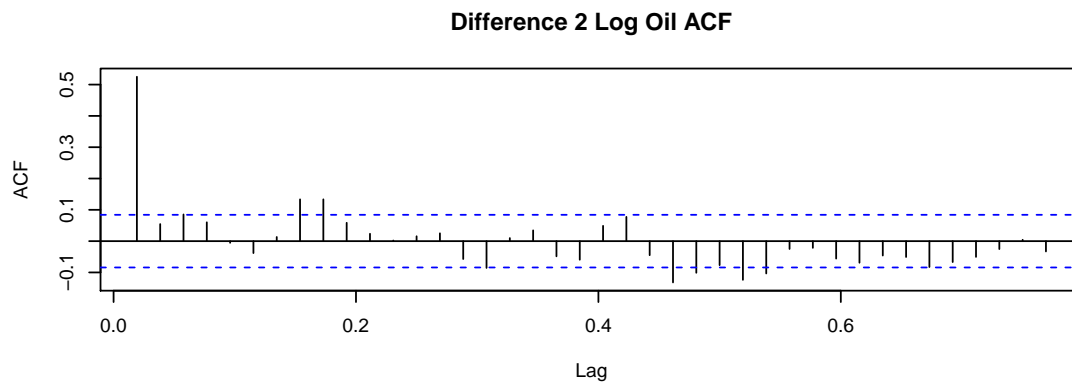
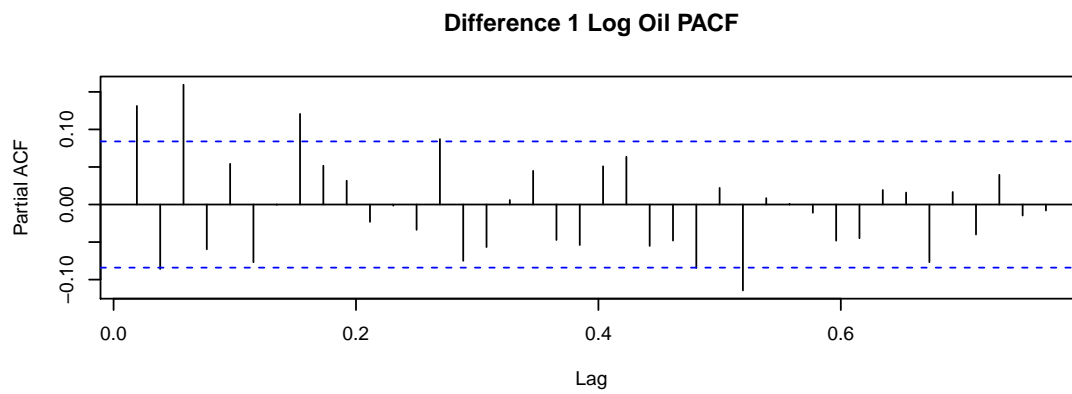
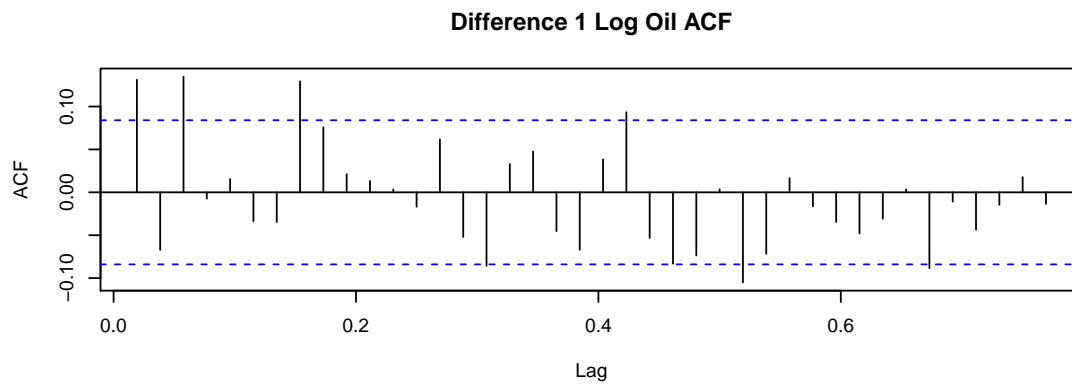
```
## Series: loil
## ARIMA(0,1,3)
##
## Coefficients:
##          ma1          ma2          ma3
##        0.1696       -0.0886       0.1458
## s.e.  0.0424       0.0424       0.0429
##
```

```
## sigma^2 estimated as 0.002094: log likelihood=907.41  
## AIC=-1806.83 AICc=-1806.75 BIC=-1789.63  
fit_plot(fit2)
```





Clearly difference log is the data we should work with. bla, bla, ...



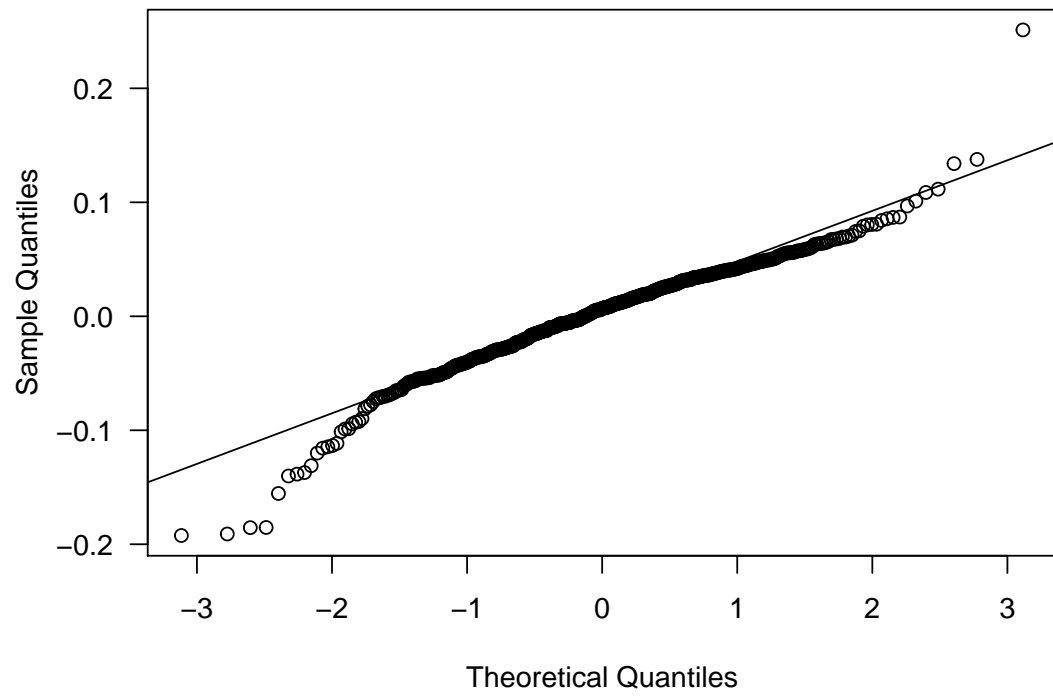
```
eacf(dloil)
```

```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x o x o o o o x o o o o o o
## 1 x o x o o o o x o o o o o o
## 2 x x x o o o o x o o o o o o
## 3 x x x o o o o x o o o o o o
## 4 x o x o o o o x o o o o o o
## 5 x x x o x o o x o o o o o o
## 6 o x x o x x o x o o o o o x
## 7 o x x x x x x x o x o o o o
```

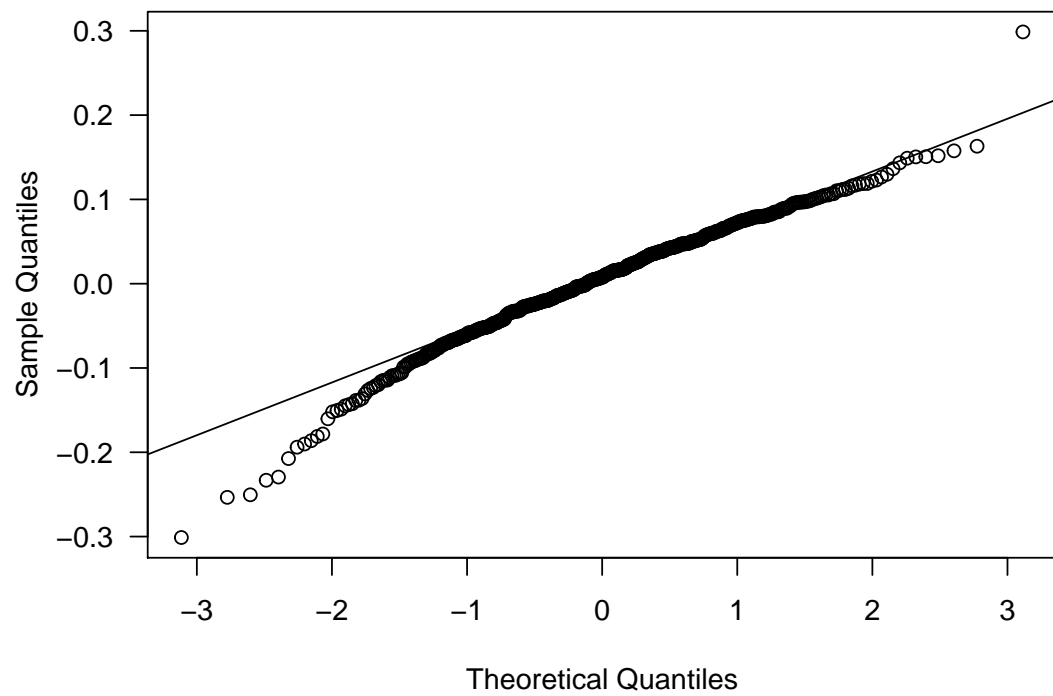
```
eacf(ddloil)
```

```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x o o o o o o x x o o o o o
## 1 x x o o o o o x x o o o o o
## 2 x x x o o o o x x o o o o o
## 3 x x x x o o o x x o o o o o
## 4 x x x x o o o x x o o o o o
## 5 x o x x x o o x x o o o o o
## 6 x o x x x x x x o x o o o o
## 7 x x x x x x x x o x o o o o
```

Difference 1 Log Oil



Difference 2 Log Oil



```
fit1 <- Arima(loil, order=c(1, 1, 1))
fit1
```

```
## Series: loil
## ARIMA(1,1,1)
##
## Coefficients:
##          ar1      ma1
##      -0.5253  0.7142
## s.e.   0.0872  0.0683
##
## sigma^2 estimated as 0.002112:  log likelihood=904.58
## AIC=-1803.15  AICc=-1803.11  BIC=-1790.25
```

```
fit2 <- Arima(loil, order=c(0, 1, 3))
fit2
```

```
## Series: loil
## ARIMA(0,1,3)
##
## Coefficients:
##          ma1      ma2      ma3
##      0.1696 -0.0886  0.1458
## s.e.  0.0424  0.0424  0.0429
##
## sigma^2 estimated as 0.002094:  log likelihood=907.41
## AIC=-1806.83  AICc=-1806.75  BIC=-1789.63
```

```
fit3 <- Arima(loil, order=c(0, 2, 1))
fit3
```

```
## Series: loil
## ARIMA(0,2,1)
##
## Coefficients:
##          ma1
##      -1.0000
## s.e.   0.0061
##
## sigma^2 estimated as 0.002213:  log likelihood=886.63
## AIC=-1769.26  AICc=-1769.24  BIC=-1760.67
```

```

complex_dist <- function(x) {
  sqrt(Re(x)^2 + Im(x)^2)
}

sapply(polyroot(c(1, -2, 1)), complex_dist)

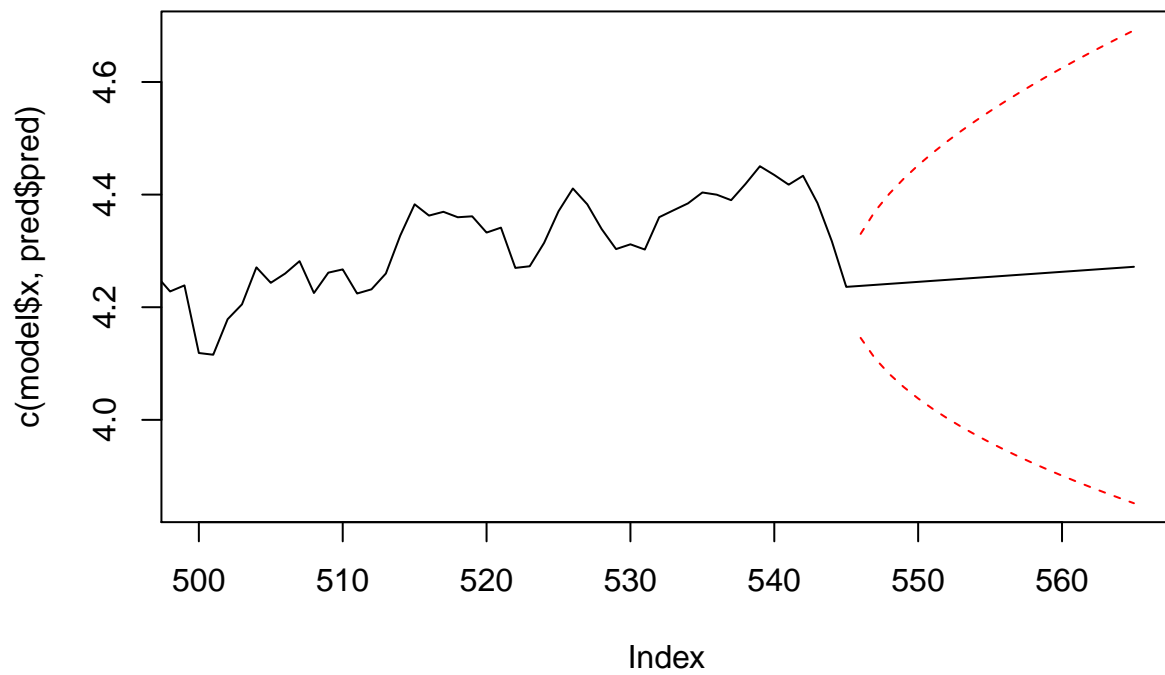
## [1] 1 1

sapply(polyroot(c(1, -1)), complex_dist)

## [1] 1

fit_plot(fit3)

```



b)