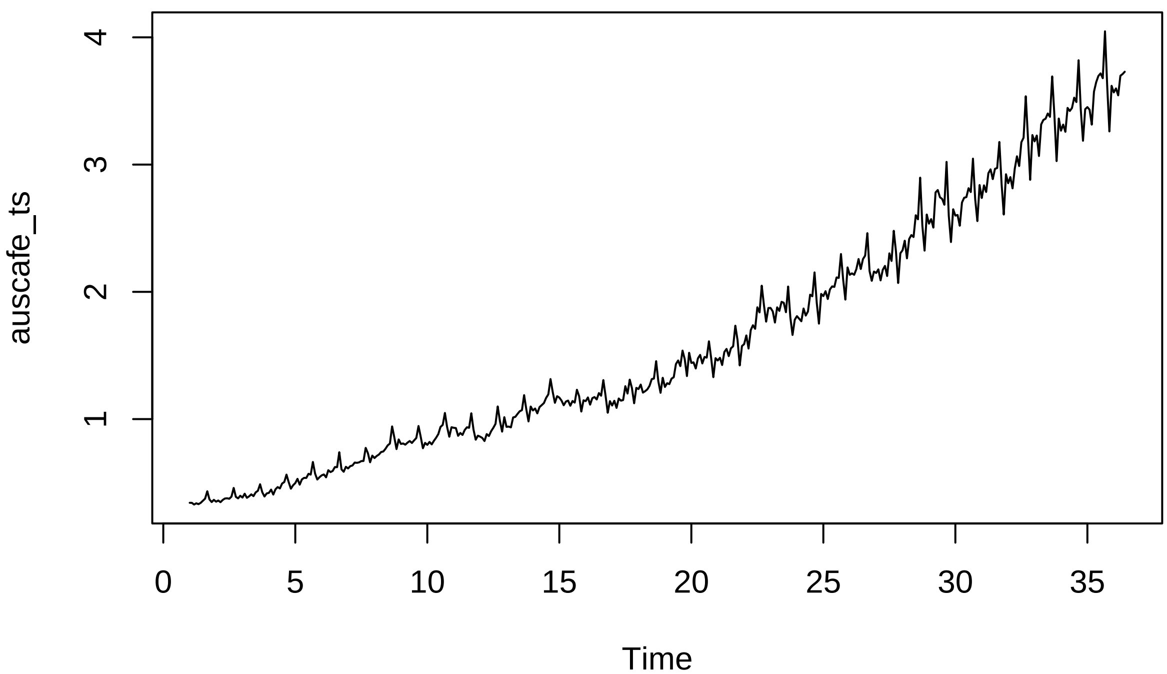
Exercise 3 (R)

Consider the total monthly expenditure on cafes, restaurants, and takeaway food services in Australia ($billion) for the sample April 1982 to September 2017 (data set auscafe).

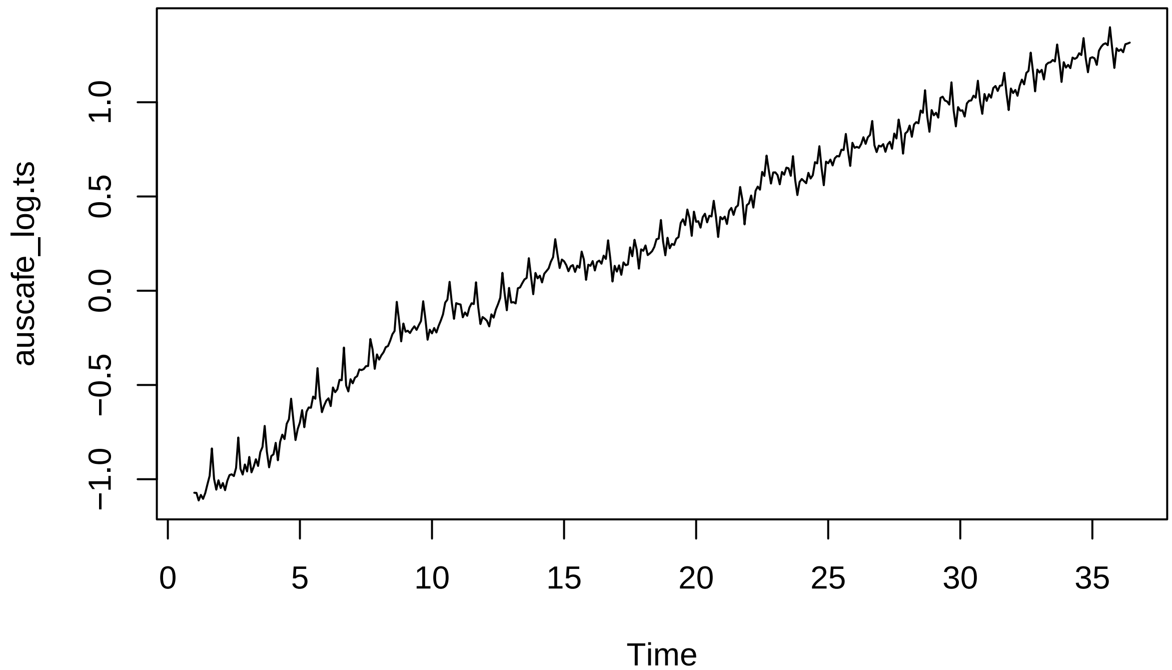
```
#lode the data

data(auscafe)
auscafe_ts <- ts(auscafe, frequency = 12)
plot(auscafe_ts)
```



3-a Do the data need transforming? If so, find a suitable transformation.

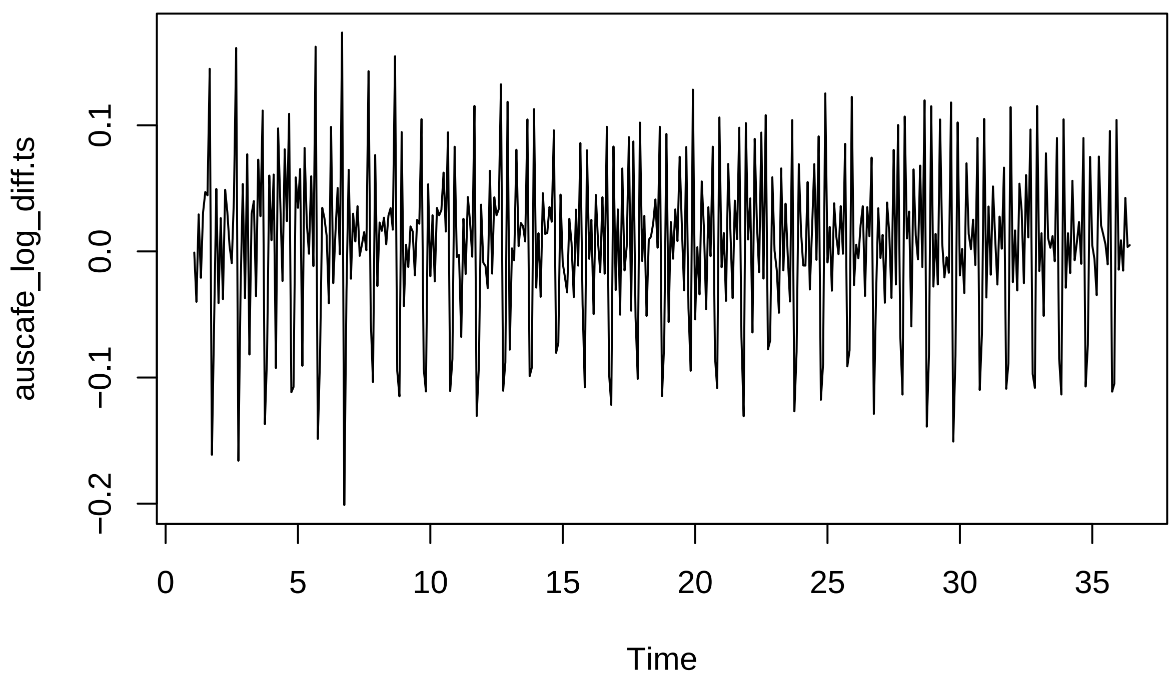
yes, the data need log transformation

```
auscafe_log.ts <- log(auscafe_ts)
plot(auscafe_log.ts)
```



3-b. Are the data stationary? If not, find an appropriate differencing which yields stationary data.

```
auscafe_log_diff.ts<- diff(auscafe_log.ts)
plot(auscafe_log_diff.ts)
```




```
#According to the graphy, it look staionary.


adf_result <- adf.test(auscafe_log_diff.ts)
kpss_result <- kpss.test(auscafe_log_diff.ts)

print(kpss_result)
```

```
##
##  KPSS Test for Level Stationarity
##
## data:  auscafe_log_diff.ts
## KPSS Level = 0.053631, Truncation lag parameter = 5, p-value = 0.1
```

```
print(adf_result)
```

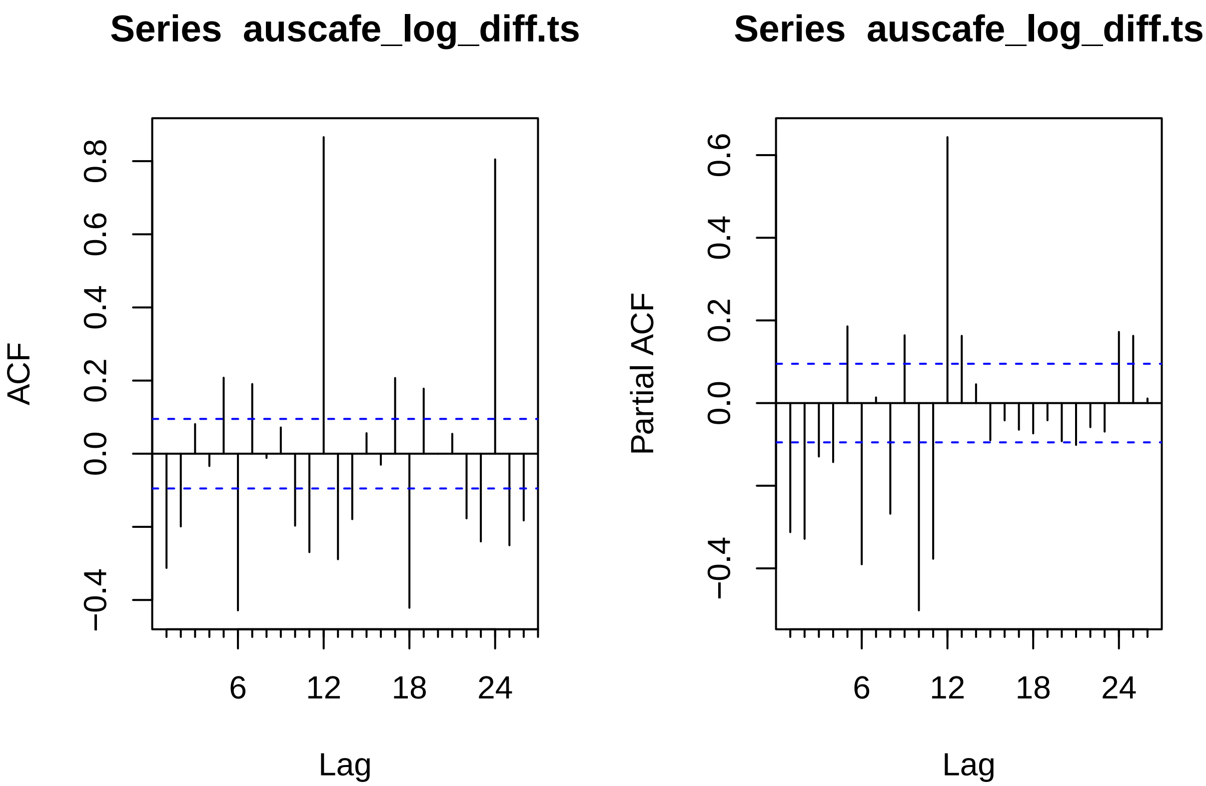
```
##
##  Augmented Dickey-Fuller Test
##
## data:  auscafe_log_diff.ts
## Dickey-Fuller = -13.17, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary
```

Both KPSS and ADF test suggest the data is stationary because both test the P-value is less the 0.05, we reject the null hypothesis and concluded the data is stationary.

Question 3-C

Identify a couple of ARIMA models that might be useful in describing the time series. Which of your models is the best according to their AIC values?

```
par(mfrow=c(1,2))
Acf(auscafe_log_diff.ts)
Pacf(auscafe_log_diff.ts)
```



According to the Acf and Pacf graph. The possible one are

ARIMA(2,1,1)(2,0,0)12 ARIMA(2,1,1)(1,0,0)12 ARIMA(2,1,1)(2,0,1)4 ARIMA(2,1,1)(1,0,1)4

ARIMA(2,1,0)(2,0,0)12 ARIMA(2,1,0)(1,0,0)4 ARIMA(2,1,0)(2,0,1)12 ARIMA(2,1,0)(1,0,1)12

ARIMA(1,1,1)(2,0,0)12 ARIMA(1,1,1)(1,0,0)12 ARIMA(1,1,1)(2,0,1)12 ARIMA(1,1,1)(1,0,1)12

ARIMA(1,1,0)(2,0,0)12 ARIMA(1,1,0)(1,0,0)12 ARIMA(1,1,0)(2,0,1)12 ARIMA(1,1,0)(1,0,1)12

```
model_211_200 <- Arima(auscafe_log_diff.ts, order = c(2, 1, 1), seasonal = list(order = c(2, 0, 0), per
model_211_100 <- Arima(auscafe_log_diff.ts, order = c(2, 1, 1), seasonal = list(order = c(1, 0, 0), per
model_211_201 <- Arima(auscafe_log_diff.ts, order = c(2, 1, 1), seasonal = list(order = c(2, 0, 1), per
model_211_101 <- Arima(auscafe_log_diff.ts, order = c(2, 1, 1), seasonal = list(order = c(1, 0, 1), per

model_210_200 <- Arima(auscafe_log_diff.ts, order = c(2, 1, 0), seasonal = list(order = c(2, 0, 0), per
model_210_100 <- Arima(auscafe_log_diff.ts, order = c(2, 1, 0), seasonal = list(order = c(1, 0, 0), per
model_210_201 <- Arima(auscafe_log_diff.ts, order = c(2, 1, 0), seasonal = list(order = c(2, 0, 1), per
model_210_101 <- Arima(auscafe_log_diff.ts, order = c(2, 1, 0), seasonal = list(order = c(1, 0, 1), per

model_111_200 <- Arima(auscafe_log_diff.ts, order = c(1, 1, 1), seasonal = list(order = c(2, 0, 0), per
model_111_100 <- Arima(auscafe_log_diff.ts, order = c(1, 1, 1), seasonal = list(order = c(1, 0, 0), per
model_111_201 <- Arima(auscafe_log_diff.ts, order = c(1, 1, 1), seasonal = list(order = c(2, 0, 1), per
model_111_101 <- Arima(auscafe_log_diff.ts, order = c(1, 1, 1), seasonal = list(order = c(1, 0, 1), per

model_110_200 <- Arima(auscafe_log_diff.ts, order = c(1, 1, 0), seasonal = list(order = c(2, 0, 0), per
model_110_100 <- Arima(auscafe_log_diff.ts, order = c(1, 1, 0), seasonal = list(order = c(1, 0, 0), per
model_110_201 <- Arima(auscafe_log_diff.ts, order = c(1, 1, 0), seasonal = list(order = c(2, 0, 1), per
model_110_101 <- Arima(auscafe_log_diff.ts, order = c(1, 1, 0), seasonal = list(order = c(1, 0, 1), per

aic_values <- c(
  model_211_200$aic, model_211_100$aic, model_211_201$aic, model_211_101$aic,
  model_210_200$aic, model_210_100$aic, model_210_201$aic, model_210_101$aic,
  model_111_200$aic, model_111_100$aic, model_111_201$aic, model_111_101$aic,
  model_110_200$aic, model_110_100$aic, model_110_201$aic, model_110_101$aic
)

names(aic_values) <- c(
  "ARIMA(2,1,1)(2,0,0)12", "ARIMA(2,1,1)(1,0,0)12", "ARIMA(2,1,1)(2,0,1)12", "ARIMA(2,1,1)(1,0,1)12",
  "ARIMA(2,1,0)(2,0,0)12", "ARIMA(2,1,0)(1,0,0)12", "ARIMA(2,1,0)(2,0,1)12", "ARIMA(2,1,0)(1,0,1)12",
  "ARIMA(1,1,1)(2,0,0)12", "ARIMA(1,1,1)(1,0,0)12", "ARIMA(1,1,1)(2,0,1)12", "ARIMA(1,1,1)(1,0,1)12",
  "ARIMA(1,1,0)(2,0,0)12", "ARIMA(1,1,0)(1,0,0)12", "ARIMA(1,1,0)(2,0,1)12", "ARIMA(1,1,0)(1,0,1)12"
)

#Print AIC values

print(aic_values)
```

```
## ARIMA(2,1,1)(2,0,0)12 ARIMA(2,1,1)(1,0,0)12 ARIMA(2,1,1)(2,0,1)12
##             -1841.924             -1787.680             -1922.105
## ARIMA(2,1,1)(1,0,1)12 ARIMA(2,1,0)(2,0,0)12 ARIMA(2,1,0)(1,0,0)12
##             -1919.246             -1706.970             -1660.584
## ARIMA(2,1,0)(2,0,1)12 ARIMA(2,1,0)(1,0,1)12 ARIMA(1,1,1)(2,0,0)12
##             -1788.670             -1782.162             -1833.396
## ARIMA(1,1,1)(1,0,0)12 ARIMA(1,1,1)(2,0,1)12 ARIMA(1,1,1)(1,0,1)12
##             -1783.692             -1914.838             -1909.332
## ARIMA(1,1,0)(2,0,0)12 ARIMA(1,1,0)(1,0,0)12 ARIMA(1,1,0)(2,0,1)12
##             -1577.531             -1545.815             -1662.560
## ARIMA(1,1,0)(1,0,1)12
##             -1644.659
```

```
aic_values_df <- as.data.frame(aic_values)
```

```r
print(aic_values_df)
```

```
##                        aic_values
## ARIMA(2,1,1)(2,0,0)12  -1841.924
## ARIMA(2,1,1)(1,0,0)12  -1787.680
## ARIMA(2,1,1)(2,0,1)12  -1922.105
## ARIMA(2,1,1)(1,0,1)12  -1919.246
## ARIMA(2,1,0)(2,0,0)12  -1706.970
## ARIMA(2,1,0)(1,0,0)12  -1660.584
## ARIMA(2,1,0)(2,0,1)12  -1788.670
## ARIMA(2,1,0)(1,0,1)12  -1782.162
## ARIMA(1,1,1)(2,0,0)12  -1833.396
## ARIMA(1,1,1)(1,0,0)12  -1783.692
## ARIMA(1,1,1)(2,0,1)12  -1914.838
## ARIMA(1,1,1)(1,0,1)12  -1909.332
## ARIMA(1,1,0)(2,0,0)12  -1577.531
## ARIMA(1,1,0)(1,0,0)12  -1545.815
## ARIMA(1,1,0)(2,0,1)12  -1662.560
## ARIMA(1,1,0)(1,0,1)12  -1644.659
```

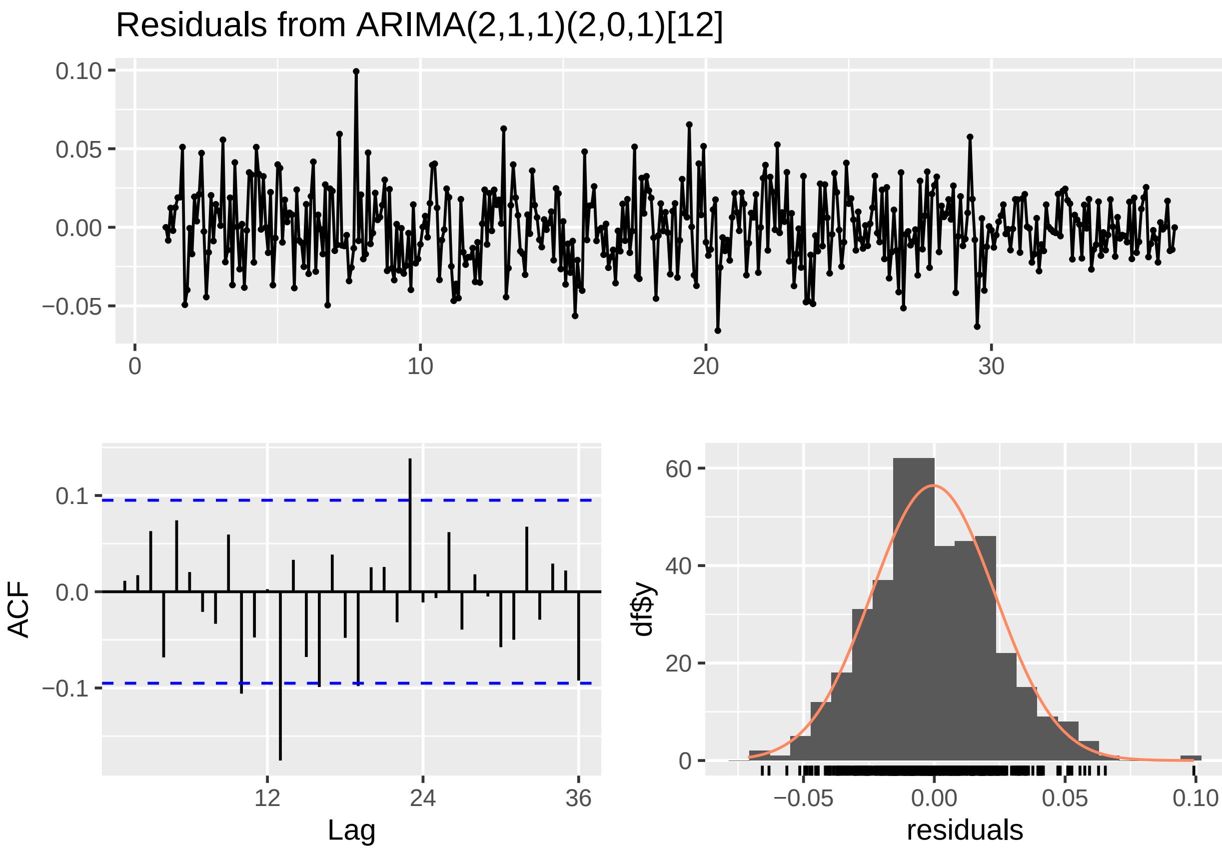
```r
min(aic_values_df)
```

```
## [1] -1922.105
```

ARIMA(2,1,1)(2,0,1)12 is the one with the smallest value.

d. Estimate the parameters of your best model and do diagnostic testing on the residuals. Do the residuals resemble white noise? If not, try to find another ARIMA model which fits better.

```r
The_best_model <- Arima(auscafe_log_diff.ts, order = c(2, 1, 1),
seasonal = list(order = c(2, 0, 1), period = 12))
checkresiduals(The_best_model)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(2,1,1)(2,0,1)[12]
## Q* = 50.732, df = 18, p-value = 5.847e-05
##
## Model df: 6.   Total lags used: 24
```

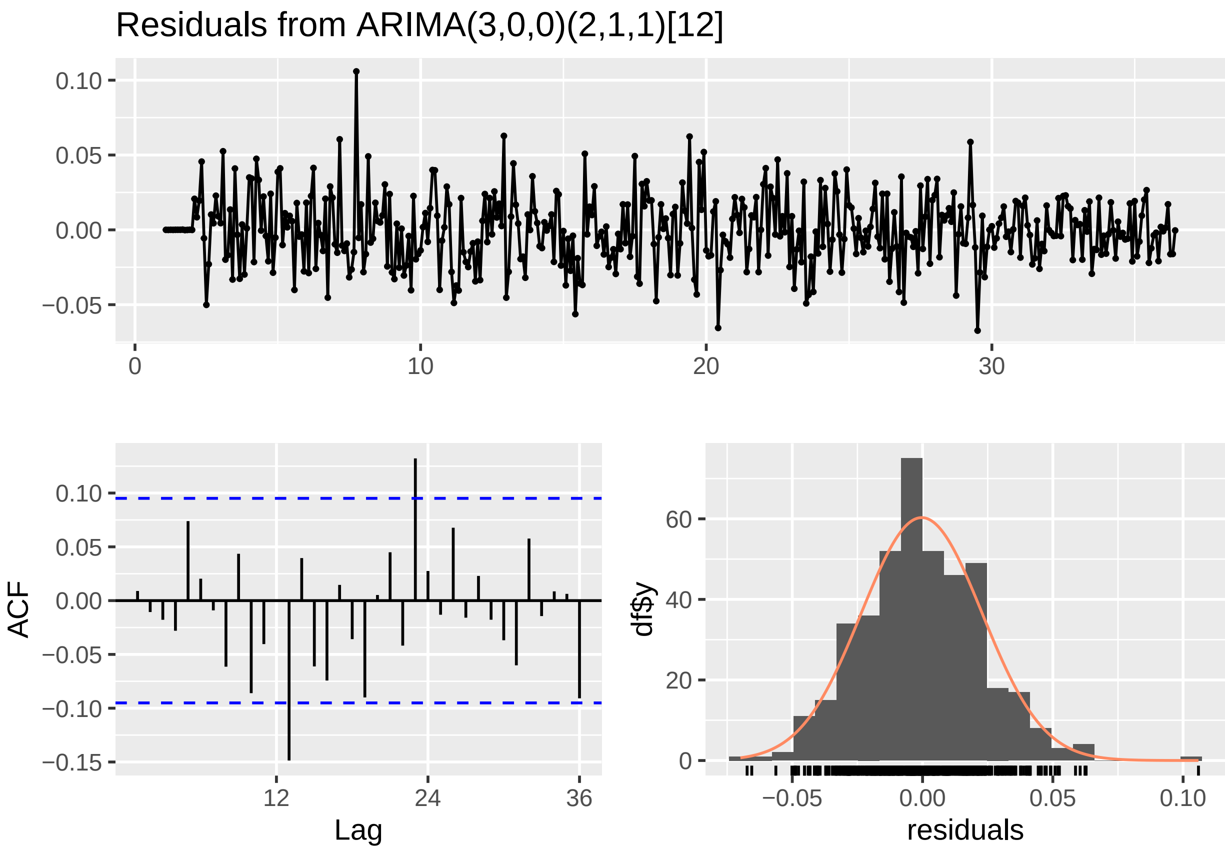
The residuaks seem like white noise.

f. Forecast the next 24 months of data using your preferred model.

```
auto.arima(auscafe_log_diff.ts)
```

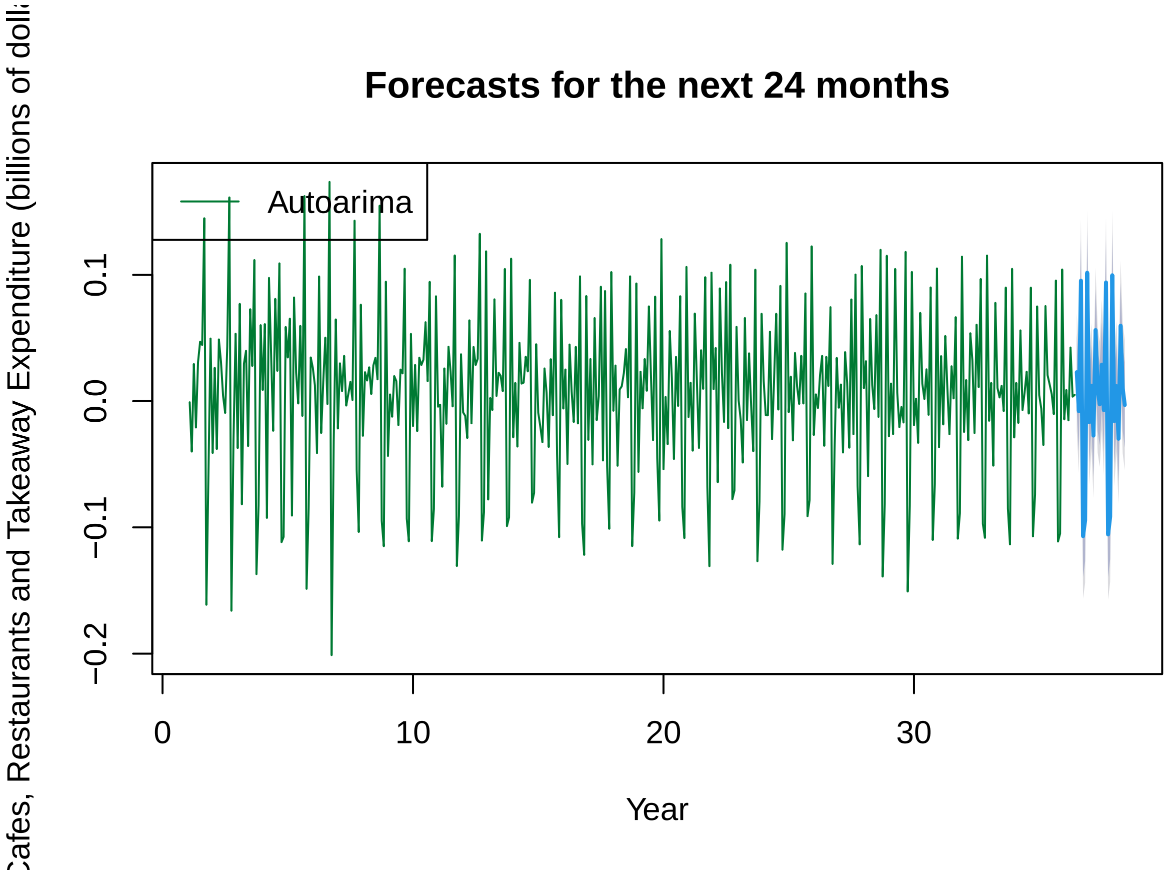
```
## Series: auscafe_log_diff.ts
## ARIMA(3,0,0)(2,1,1)[12]
##
## Coefficients:
##           ar1      ar2     ar3    sar1     sar2     sma1
##       -0.3412  -0.1094  0.0955  0.1216  -0.0530  -0.8302
## s.e.   0.0510   0.0521  0.0491  0.0646   0.0585   0.0432
##
## sigma^2 = 0.0005606:  log likelihood = 956.57
## AIC=-1899.13   AICc=-1898.85   BIC=-1870.97
```

```
The_best_model_better_best <- Arima(auscafe_log_diff.ts, order = c(3, 0, 0), seasonal = list(order = c(
checkresiduals(The_best_model_better_best)
```

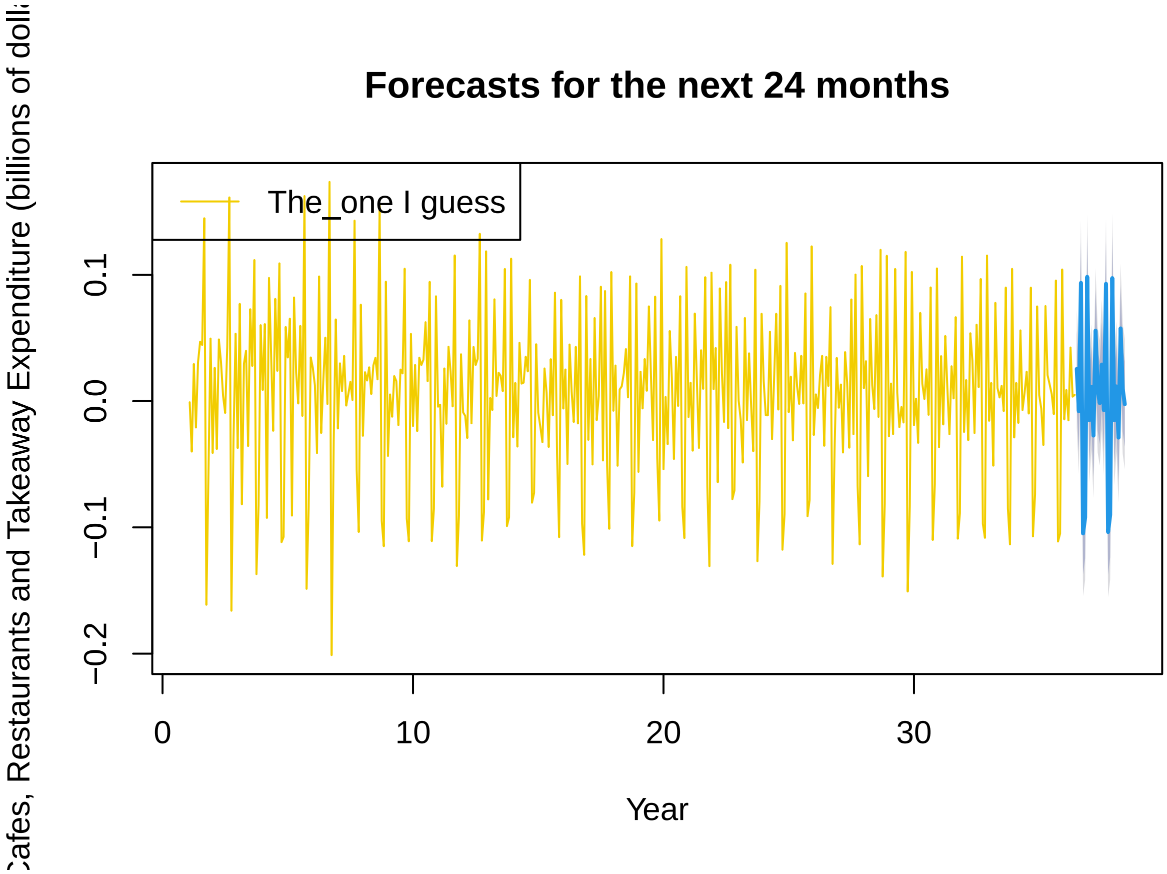



```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(3,0,0)(2,1,1)[12]
## Q* = 38.301, df = 18, p-value = 0.003534
##
## Model df: 6.   Total lags used: 24
```

```
forecast1 <- forecast(The_best_model_better_best, h = 24)
forecast2 <- forecast(The_best_model, h = 24)

#007A33 is ualberta green

plot(forecast1, type = "l", col = "#007A33", xlab = "Year", ylab = "Cafes, Restaurants and Takeaway Expe
legend("topleft", legend = c("Autoarima"), col = c("#007A33"), lty = 1)
```




```
plot(forecast2, type = "l", col = "#F2CD00", xlab = "Year", ylab = "Cafes, Restaurants and Takeaway Expe
legend("topleft", legend = c( "The_one I guess"), col = c( "#F2CD00"), lty = 1)
```




```
summary(forecast1)
```

```
##
## Forecast method: ARIMA(3,0,0)(2,1,1)[12]
##
## Model Information:
## Series: auscafe_log_diff.ts
## ARIMA(3,0,0)(2,1,1)[12]
##
## Coefficients:
##           ar1      ar2     ar3    sar1     sar2     sma1
##       -0.3412  -0.1094  0.0955  0.1216  -0.0530  -0.8302
## s.e.   0.0510   0.0521  0.0491  0.0646   0.0585   0.0432
##
## sigma^2 = 0.0005606:  log likelihood = 956.57
## AIC=-1899.13   AICc=-1898.85   BIC=-1870.97
##
## Error measures:
##                         ME       RMSE        MAE      MPE     MAPE      MASE
## Training set -0.0003351246 0.02316931 0.01795448 19.07737 100.7408 0.7738266
##                    ACF1
## Training set 0.00895241
##
## Forecasts:
##        Point Forecast         Lo 80        Hi 80        Lo 95       Hi 95
## Jul 36    0.022940206 -0.007401966  0.053282377 -0.023464141  0.06934455
```

```
## Aug 36   -0.008008323 -0.040067583  0.024050938 -0.057038731  0.04102209
## Sep 36    0.095408346  0.063348384  0.127468307  0.046376866  0.14443983
## Oct 36   -0.106950789 -0.139254299 -0.074647280 -0.156354744 -0.05754683
## Nov 36   -0.094368422 -0.126758238 -0.061978607 -0.143904371 -0.04483247
## Dec 36    0.101730416  0.069338213  0.134122619  0.052190817  0.15127002
## Jan 37   -0.016651617 -0.049047719  0.015744485 -0.066197180  0.03289395
## Feb 37    0.012283751 -0.020115340  0.044682842 -0.037266382  0.06183388
## Mar 37   -0.027273017 -0.059672380  0.005126346 -0.076823566  0.02227753
## Apr 37    0.056327077  0.023927675  0.088726480  0.006776467  0.10587769
## May 37    0.008682744 -0.023716743  0.041082231 -0.040867996  0.05823348
## Jun 37   -0.002554730 -0.034954233  0.029844774 -0.052105494  0.04699603
## Jul 37    0.028964528 -0.004620048  0.062549104 -0.022398649  0.08032770
## Aug 37   -0.007205776 -0.040926521  0.026514969 -0.058777205  0.04436565
## Sep 37    0.094012514  0.060291700  0.127733329  0.042440979  0.14558405
## Oct 37   -0.105623701 -0.139364212 -0.071883189 -0.157225361 -0.05402204
## Nov 37   -0.091480851 -0.125228418 -0.057733284 -0.143093300 -0.03986840
## Dec 37    0.099712869  0.065965103  0.133460634  0.048100115  0.15132562
## Jan 38   -0.015781227 -0.049529308  0.017966855 -0.067394463  0.03583201
## Feb 38    0.011893199 -0.021855124  0.045641521 -0.039720406  0.06350680
## Mar 38   -0.029789151 -0.063537495  0.003959194 -0.081402790  0.02182449
## Apr 38    0.059764863  0.026016515  0.093513210  0.008151219  0.11137851
## May 38    0.010207695 -0.023540660  0.043956050 -0.041405959  0.06182135
## Jun 38   -0.003030815 -0.036779171  0.030717541 -0.054644472  0.04858284
```

```
summary(forecast2)
```

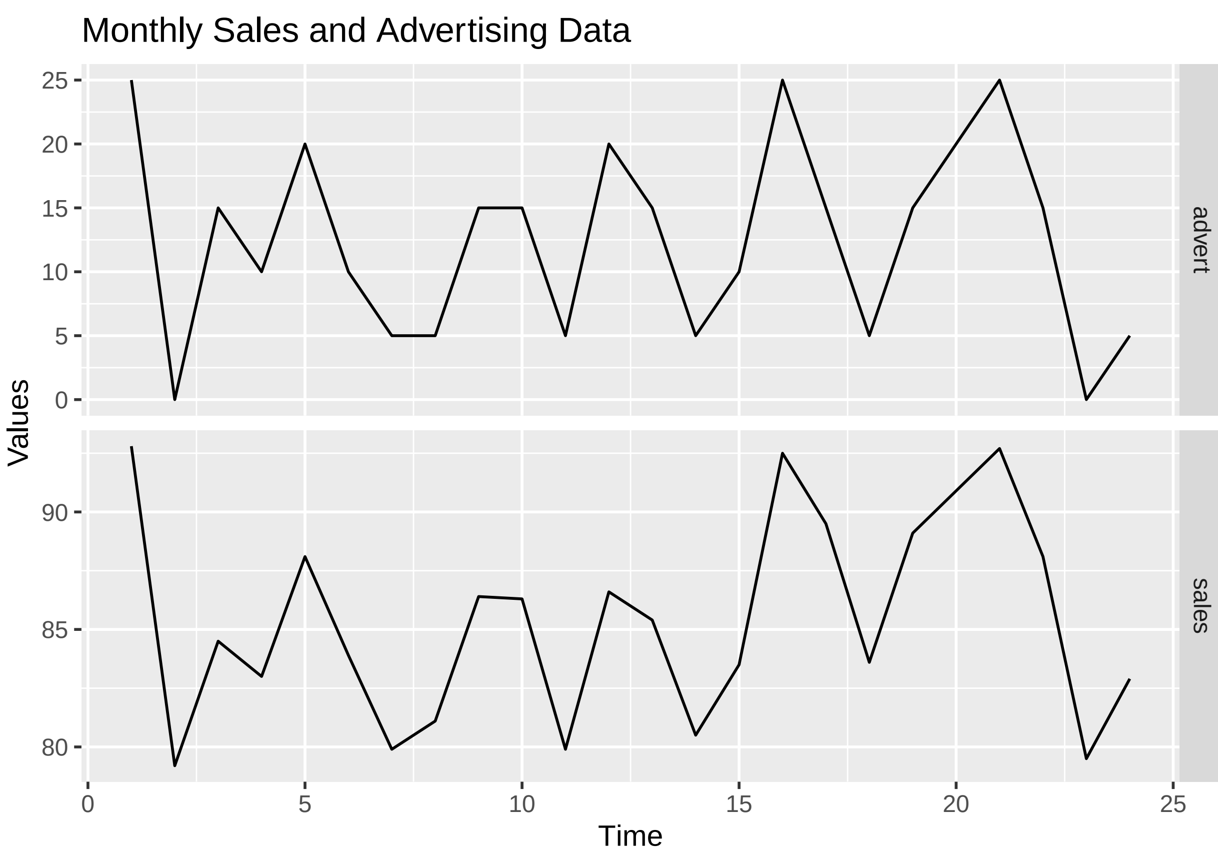
```
##
## Forecast method: ARIMA(2,1,1)(2,0,1)[12]
##
## Model Information:
## Series: auscafe_log_diff.ts
## ARIMA(2,1,1)(2,0,1)[12]
##
## Coefficients:
##           ar1      ar2     ma1    sar1     sar2     sma1
##       -0.3573  -0.1474  -1.000  1.1368  -0.1387  -0.8538
## s.e.   0.0632   0.0889   0.002  0.0278   0.0282   0.0230
##
## sigma^2 = 0.0005656:  log likelihood = 968.05
## AIC=-1922.11   AICc=-1921.84   BIC=-1893.76
##
## Error measures:
##                         ME       RMSE        MAE      MPE     MAPE      MASE
## Training set -0.0004411944 0.02358485 0.01861541 21.92912 100.9036 0.8023121
##                    ACF1
## Training set 0.01136266
##
## Forecasts:
##        Point Forecast        Lo 80        Hi 80        Lo 95       Hi 95
## Jul 36    0.025505128 -0.004973326  0.055983581 -0.021107645  0.07211790
## Aug 36   -0.008038876 -0.040403892  0.024326139 -0.057536896  0.04145914
## Sep 36    0.093635400  0.061264817  0.126005982  0.044128865  0.14314193
## Oct 36   -0.104772011 -0.137193858 -0.072350163 -0.154356948 -0.05518707
## Nov 36   -0.092026108 -0.124452794 -0.059599423 -0.141618445 -0.04243377
## Dec 36    0.098436920  0.066010167  0.130863673  0.048844481  0.14802936
```

```
## Jan 37   -0.015037901 -0.047464835  0.017389033 -0.064630617  0.03455482
## Feb 37    0.011336175 -0.021090754  0.043763104 -0.038256533  0.06092888
## Mar 37   -0.027249960 -0.059676889  0.005176969 -0.076842669  0.02234275
## Apr 37    0.055855081  0.023428151  0.088282011  0.006262371  0.10544779
## May 37    0.009028834 -0.023398096  0.041455764 -0.040563876  0.05862154
## Jun 37   -0.001529915 -0.033956845  0.030897016 -0.051122624  0.04806280
## Jul 37    0.028998636 -0.004556197  0.062553468 -0.022319052  0.08031632
## Aug 37   -0.007178460 -0.040874308  0.026517388 -0.058711812  0.04435489
## Sep 37    0.092905818  0.059209550  0.126602085  0.041371823  0.14443981
## Oct 37   -0.103641220 -0.137341460 -0.069940979 -0.155181290 -0.05210115
## Nov 37   -0.090013197 -0.123713802 -0.056312592 -0.141553825 -0.03847257
## Dec 37    0.097432064  0.063731455  0.131132674  0.045891430  0.14897270
## Jan 38   -0.015075724 -0.048776348  0.018624900 -0.066616380  0.03646493
## Feb 38    0.011676622 -0.022023982  0.045377226 -0.039864004  0.06321725
## Mar 38   -0.028855062 -0.062555666  0.004845542 -0.080395688  0.02268556
## Apr 38    0.057611199  0.023910595  0.091311803  0.006070573  0.10915183
## May 38    0.009784689 -0.023915915  0.043485293 -0.041755937  0.06132532
## Jun 38   -0.002418196 -0.036118800  0.031282408 -0.053958822  0.04912243
```

AIC and BIC are close and forecast almost identical.

4. Consider monthly sales and advertising data for an automotive parts company (data set advert).

```
autoplot(advert, facets = TRUE) +
  ggtitle("Monthly Sales and Advertising Data") +
  xlab("Time") +
  ylab("Values")
```



The facets=TRUE function allowed you compare two set of data side by side provide more straright forward veson on their relationship.

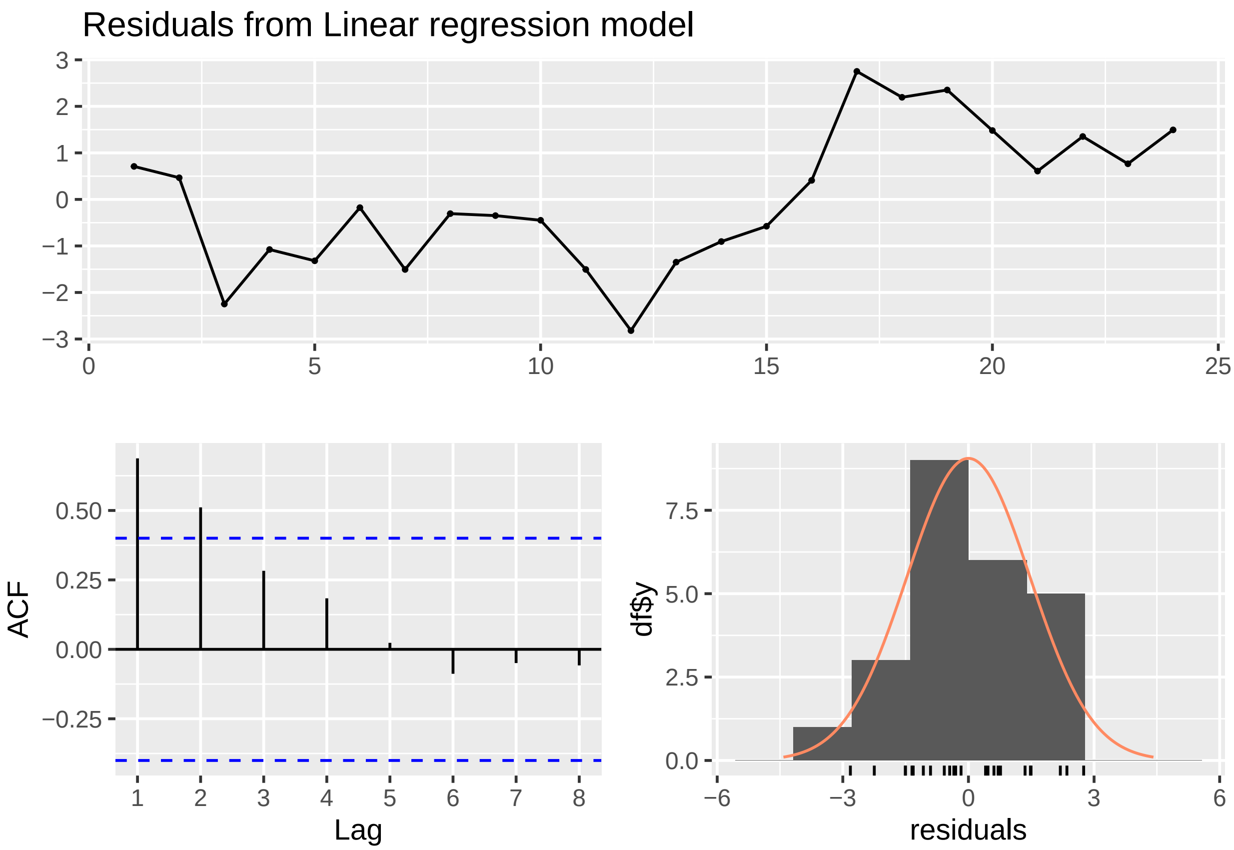
b. Fit a standard regression model y t = a + bx t + n t where y t denotes sales and x t denotes advertising using the tslm() function

```
regression_model <- tslm(sales ~ advert, data = advert)
summary(regression_model)
```

```
##
## Call:
## tslm(formula = sales ~ advert, data = advert)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8194 -1.1375 -0.2412  0.9123  2.7519
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 78.73426    0.59735  131.81  < 2e-16 ***
## advert       0.53426    0.04098   13.04 7.96e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.506 on 22 degrees of freedom
## Multiple R-squared:  0.8854, Adjusted R-squared:  0.8802
## F-statistic:   170 on 1 and 22 DF,  p-value: 7.955e-12
```

c. Show that the residuals have significant autocorrelation.

```
checkresiduals(regression_model)
```




```
##
##  Breusch-Godfrey test for serial correlation of order up to 5
##
## data:  Residuals from Linear regression model
## LM test = 12.498, df = 5, p-value = 0.02856
```

According the ACF graph, it show a consistent decresing trend, represent the residuals is significant auto correlations. The P value is greater than 0.01, represent there are autocorrlation excist.

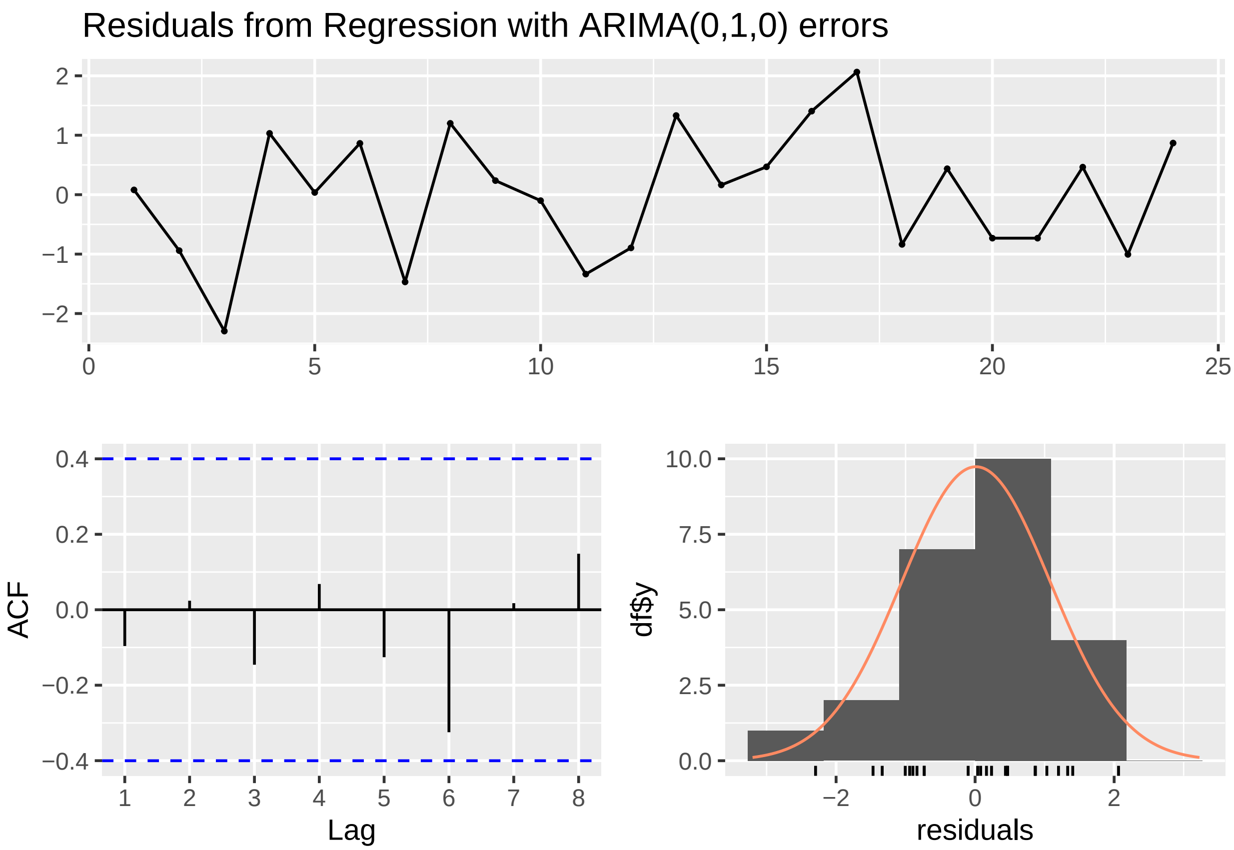
4-d e. Refit the model using auto.arima(). How much difference does the error model make to the estimated parameters? What ARIMA model for the errors is selected?

```r
advert.df <- as.data.frame(advert)
advert_arma <- auto.arima(advert.df$sales, xreg = advert.df$advert)
summary(advert_arma)
```

```
## Series: advert.df$sales
## Regression with ARIMA(0,1,0) errors
##
## Coefficients:
##         xreg
##       0.5063
## s.e.  0.0210
##
## sigma^2 = 1.201:  log likelihood = -34.22
## AIC=72.45   AICc=73.05   BIC=74.72
##
## Training set error measures:
##                        ME      RMSE       MAE         MPE      MAPE      MASE
```

```
## Training set 0.01279435 1.049041 0.8745732 -0.00247038 1.032833 0.189587
##                      ACF1
## Training set -0.09614401
```

```
checkresiduals(advert_arma)
```

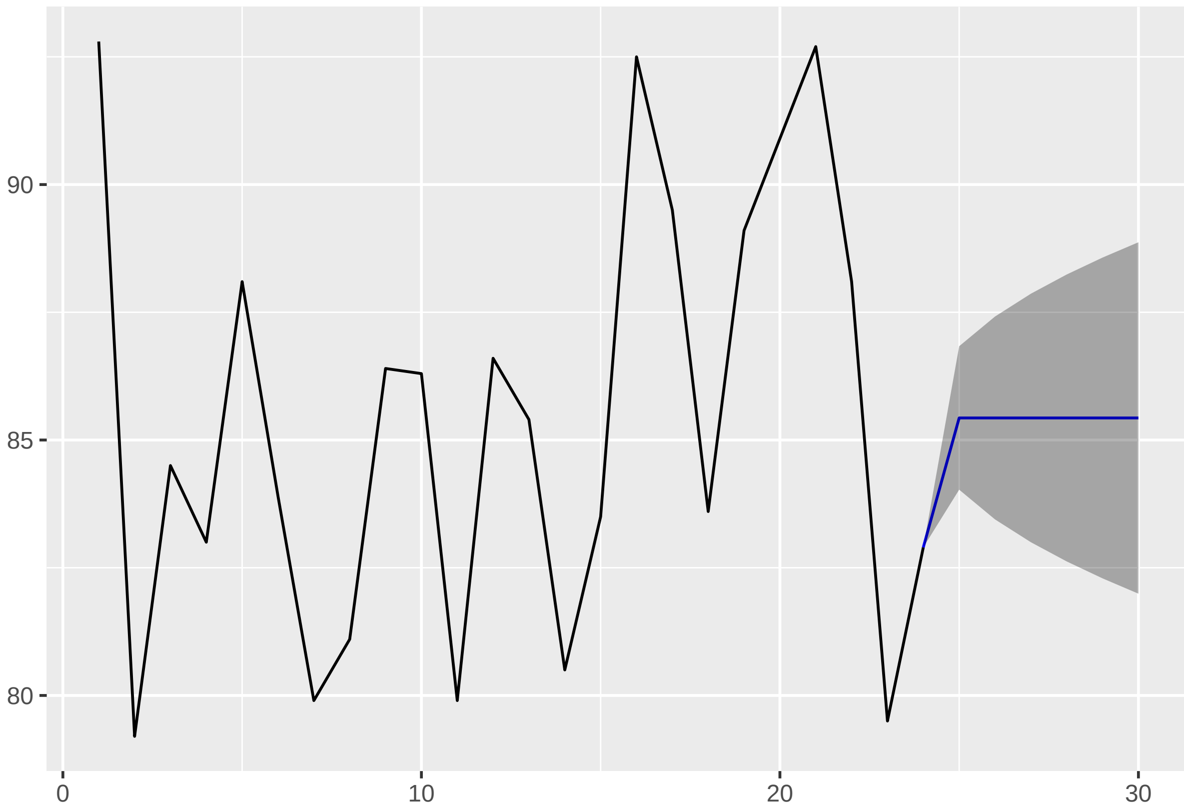



```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(0,1,0) errors
## Q* = 1.5622, df = 5, p-value = 0.9058
##
## Model df: 0.   Total lags used: 5
```

The estimated of coefficients for the adverstisigin variable are smaller from regression model. The auto-arima choice ARIMA(0,1,0)

4g

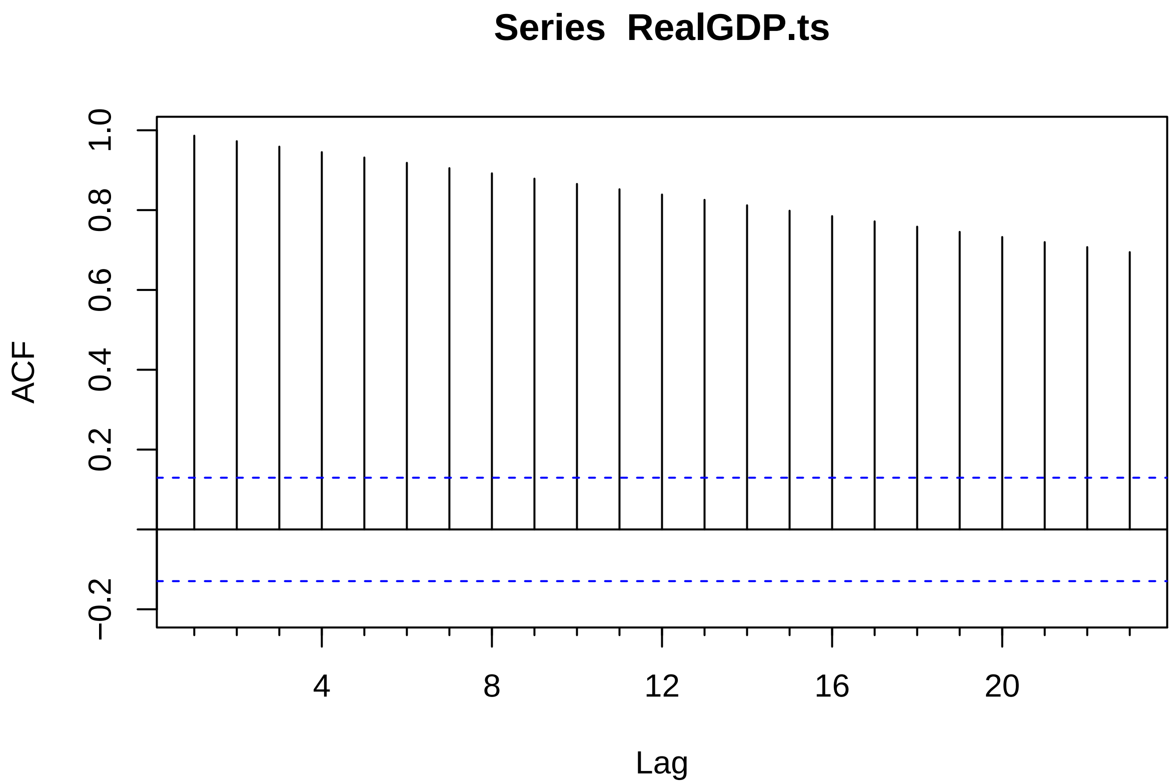
```
model_arima <- auto.arima(advert.df$sales, xreg = advert.df$advert)
sales_forecast <- forecast(model_arima, xreg = rep(10, 6))
autoplot(sales_forecast)
```



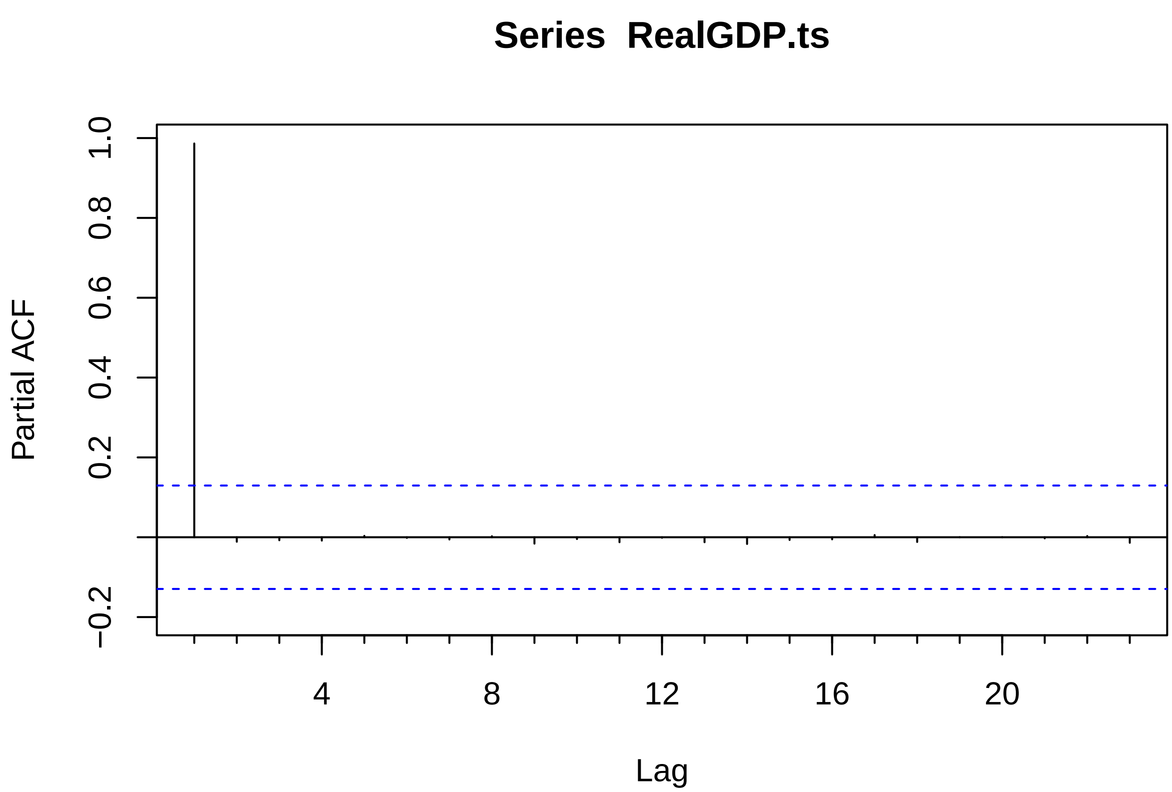
Q5:The file NAEXKP01CAQ661S.csv contains the series of quarterly real gross domestic product (RGDP) for Canada for the quarters 1961:Q1 to 2018:Q1, measured in millions of 2010 Canadian dollars and seasonally adjusted.

a. Use R to plot the series, the ACF, and PACF. Does the series appear to be stationary?

```
RealGDP <- read_excel("/Users/tie/SynologyDrive/nn/ECON-493-forcasting-economy/493 homework/Homewrok 4/I
    col_types = c("date", "numeric"))

#head(RealGDP, 5)

RealGDP.ts <- ts(RealGDP$NAEXKP01CAQ661S, start = 1961/01/01, end = 2018/01/01, frequency = 4)

Acf(RealGDP.ts)
```




```
Pacf(RealGDP.ts)
```



No, the data is not stationary.

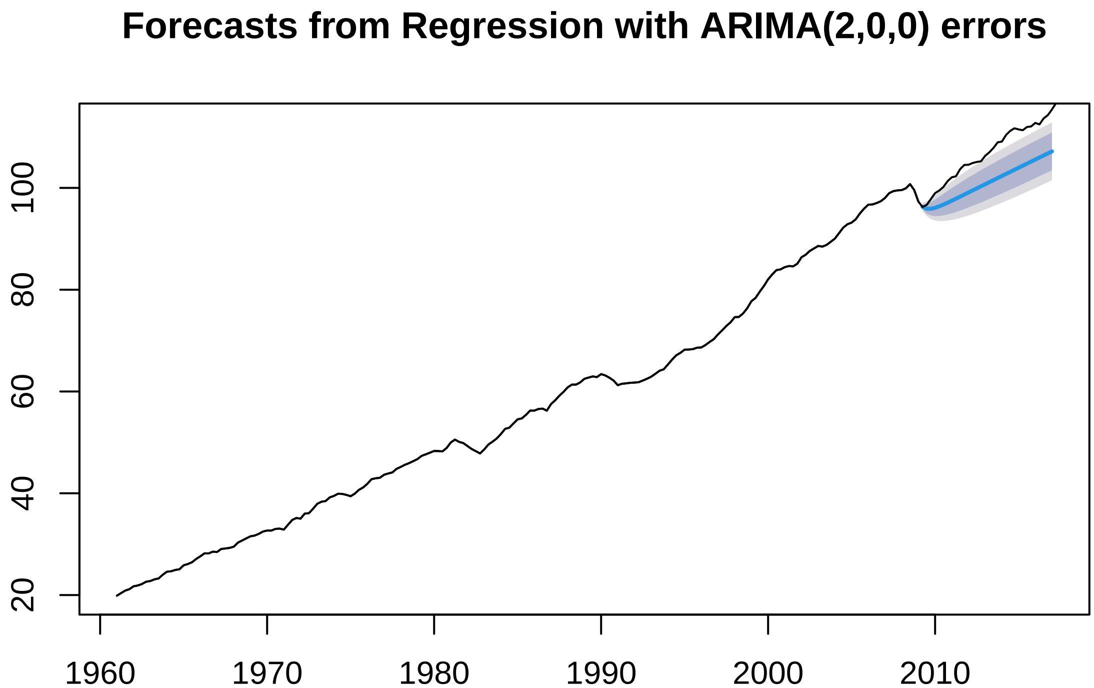
```r
# Subset the data

RealGDP_trainset <- window(RealGDP.ts, start = 1961/01/01, end = 2009/01/01)

trend <- seq_along(RealGDP_trainset)
(fit1 <- auto.arima(RealGDP_trainset, d=0, xreg=trend))
```

```
## Series: RealGDP_trainset
## Regression with ARIMA(2,0,0) errors
##
## Coefficients:
##          ar1      ar2  intercept    xreg
##       1.4957  -0.5154    17.8432  0.3975
## s.e.  0.0693   0.0698     2.3359  0.0196
##
## sigma^2 = 0.1719:  log likelihood = -104.03
## AIC=218.05   AICc=218.37   BIC=234.37
```


```r
trend <- seq_along(RealGDP_trainset)
(fit1 <- auto.arima(RealGDP_trainset, d=0, xreg=trend))
```

```
## Series: RealGDP_trainset
## Regression with ARIMA(2,0,0) errors
##
## Coefficients:
##          ar1      ar2  intercept    xreg
##       1.4957  -0.5154    17.8432  0.3975
## s.e.  0.0693   0.0698     2.3359  0.0196
##
## sigma^2 = 0.1719:  log likelihood = -104.03
## AIC=218.05   AICc=218.37   BIC=234.37
```


```r
phi1 <- coef(fit1)['ar1']
phi2 <- coef(fit1)['ar2']
intercept <- coef(fit1)['intercept']
slope <- coef(fit1)['xreg']
sigma2 <- fit1$sigma2

fc1 <- forecast(fit1, xreg=length(RealGDP_trainset) + 1:32)

plot(fc1)
lines(RealGDP.ts)
```



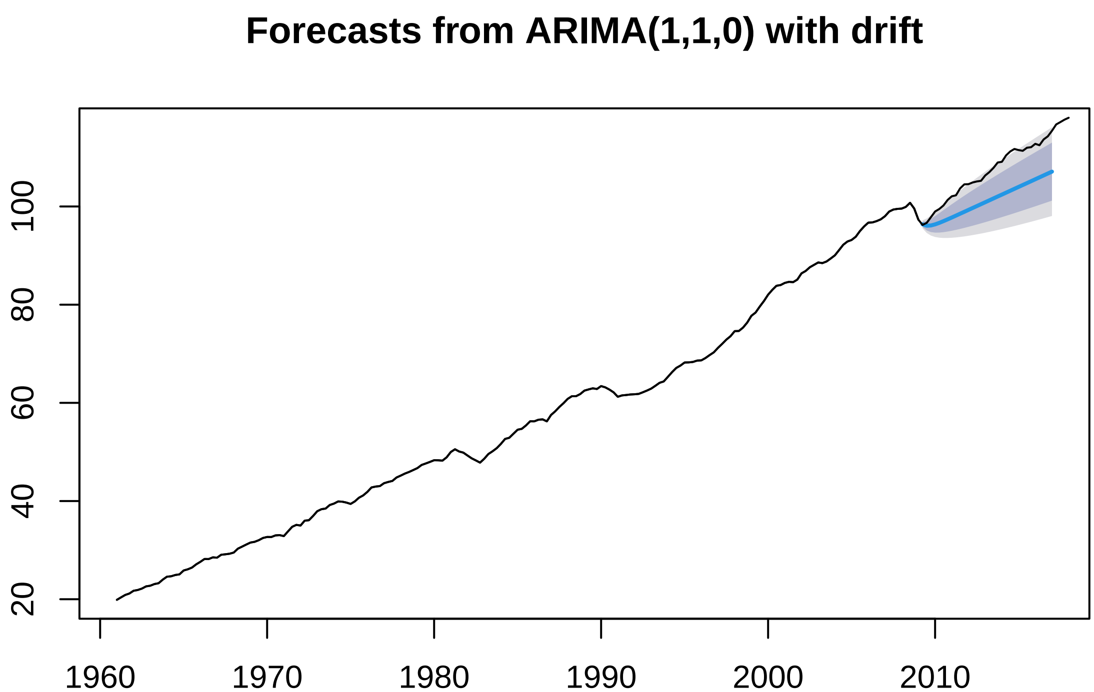
The graph show that the forecast are continue to up with a permanent negative shock around 2008 however the real data suggest the economy is backing to it normal growth rate. forecast is upward are date trend suggest and perdition error are hold relative consistent.

5-d Here we will fit a difference stationary model for the sample 1961Q1 to 2009Q4. Using the AIC, find the AR model that adequately describes the change RGDP. Make sure your model uses d = 1 and includes a drift. Motivate the steps that you take.

```
(fit2 <- auto.arima(RealGDP_trainset, d=1))
```

```
## Series: RealGDP_trainset
## ARIMA(1,1,0) with drift
##
## Coefficients:
##          ar1   drift
##       0.5018  0.3899
## s.e.  0.0700  0.0599
##
## sigma^2 = 0.1739:  log likelihood = -103.65
## AIC=213.3   AICc=213.43   BIC=223.08
```

```
#ARIMA(1,1,0

drift <- coef(fit2)['drift']
theta1 <- coef(fit2)['ma1']
sigma2 <- fit2$sigma2
fc2 <- forecast(fit2, h=32)

plot(fc2)
lines(RealGDP.ts)
```



According to the graphy, we can see that the forecaster values of real GDP follow an increasing trend with wider and wider predicted interview represent the increasing uncertainty in the fulture. The upward trend is consistent with the historical data. further more the predition error is wider in a faster speed than

5-e

```
tend_stationary_model_accuracy <- accuracy(fc1, RealGDP.ts)
Difference_stationary_model_accuracy <- accuracy(fc2,RealGDP.ts)


print(tend_stationary_model_accuracy)
```

```
##                        ME      RMSE       MAE         MPE      MAPE      MASE
## Training set -0.006690238 0.4102395 0.3192064 -0.02078465 0.6367834 0.1739996
## Test set      5.571224724 5.9495187 5.5736808  5.14220755 5.1447596 3.0382168
##                    ACF1 Theil's U
## Training set -0.00785481        NA
## Test set      0.82508292  7.521015
```

```
print(Difference_stationary_model_accuracy)
```

```
##                         ME      RMSE      MAE         MPE      MAPE      MASE
## Training set -0.0002640416 0.4137764 0.317921 -0.01191591 0.6306907 0.1732989
## Test set      5.4533739063 5.8675809 5.461262  5.02660559 5.0348019 2.9769371
##                      ACF1 Theil's U
## Training set -0.0004223652        NA
## Test set      0.8357206264  7.403863
```

According to the accuarcy function, the Difference_stationary_model_accuracy have the lower RMSE, therefore, Difference_stationary_mode is better model.