Assignment 4: Stationary Univariate ARMA models

Andrew G. Dunn

January 25, 2016

Andrew G. Dunn, Northwestern University Predictive Analytics Program

Prepared for PREDICT-413: Time Series Analysis and Forecasting.

Formatted using the LaTeX, via pandoc and R Markdown. References managed using pandoc-citeproc.

Setup

```r
require(fBasics)    # for calculations
require(fpp)        # for data
require(knitr)      # for table output
require(ggplot2)    # for graphing
require(ggfortify)  # for graphing time series
require(ggthemes)   # for graphing beautifully
require(gridExtra)  # for laying out graphs
```

Part 1

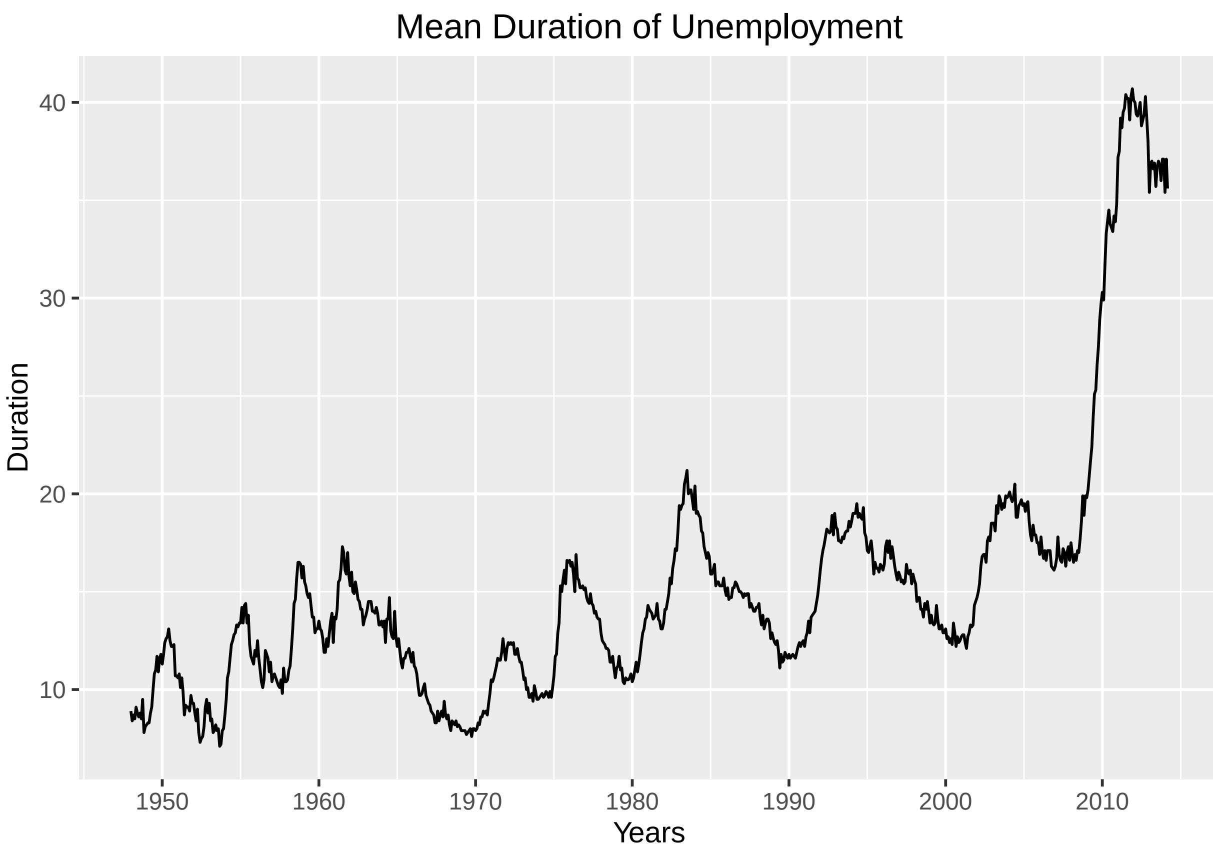
Unemployment rate is an important macroeconomic series. Equivalent importance is the duration of unemployment. Consider the mean duration of unemployment in the U.S. from January 1948 to March 2014. The duration is measured in weeks. The data re available from FRED of the Federal Reserve Bank of St. Louis, and also in `m-unempmean.txt`. The data were seasonally adjusted.

```r
d1 = read.table("data/m-unempmean.txt", header=T)
head(d1)
```

```
##   Year Mon Day Value
## 1 1948   1   1   8.9
## 2 1948   2   1   8.4
## 3 1948   3   1   8.7
## 4 1948   4   1   8.5
## 5 1948   5   1   9.1
## 6 1948   6   1   8.8
```

We'll visually examine the daa set to form initial impressions.

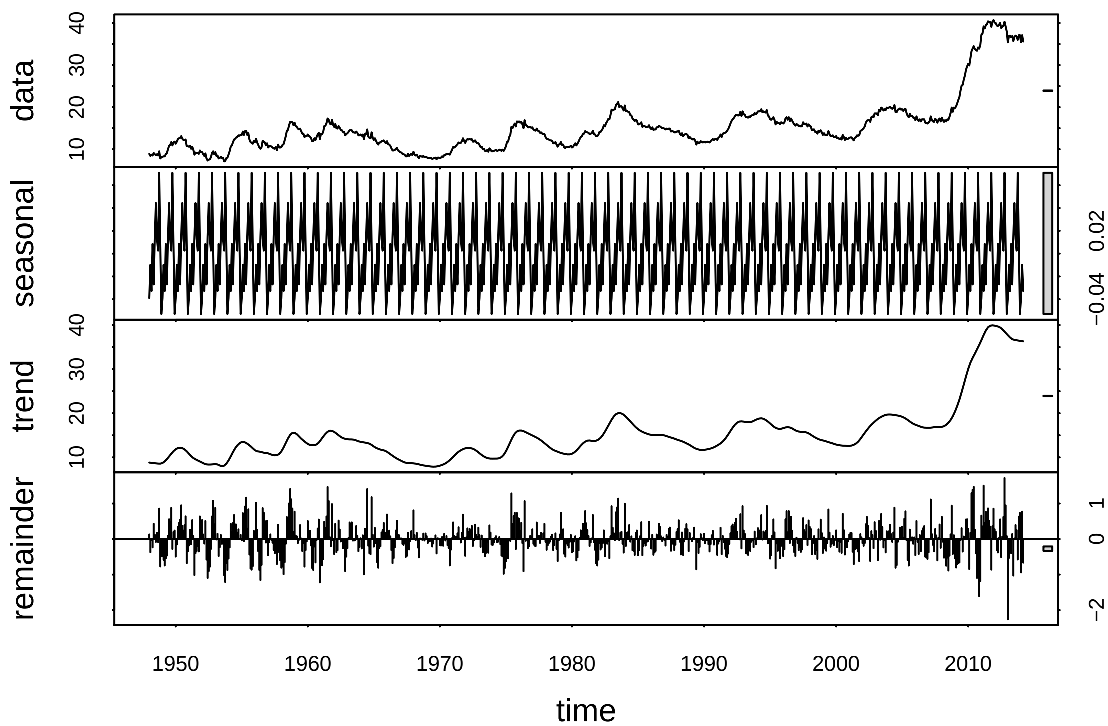
```
t1 = ts(d1$Value, start = 1948, frequency = 12)
autoplot(t1, main = "Mean Duration of Unemployment", ylab = "Duration", xlab = "Years")
```



We can see a slightly increasing trend that has indication of seasonality and cyclic characteristics. We also notice the large increase between 2009 and 2014.

We will perform an STL decomposition to investigate our suspicion that this data has a seasonal component.

```
unemp_stl = stl(t1, s.window="periodic")
plot(unemp_stl)
```



We see a seasonal component, which happens to be at a pretty high frequency.

Part A

Does the mean duration series have a unit root? Why?

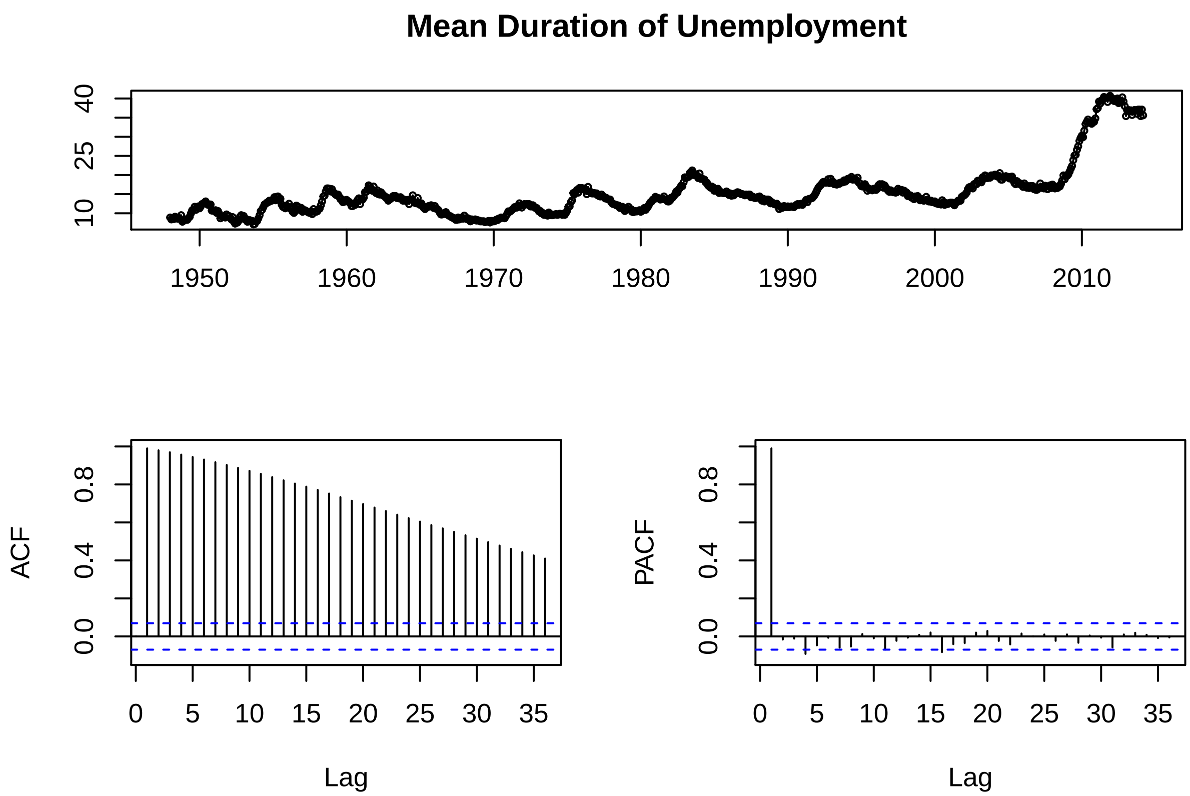
We will use the Augumented Dickey-Fuller Test, which computes the Augmented Dickey-Fuller test for the null that the time series has a unit root:

```r
adf.test(t1)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  t1
## Dickey-Fuller = -2.4134, Lag order = 9, p-value = 0.4033
## alternative hypothesis: stationary
```

Which, due to the p-value of `0.4033` meaning we fail to reject the null hypthothesis. This provides us with strong indication that the timeseries is non-stationary.

```r
tsdisplay(t1, main = "Mean Duration of Unemployment")
```

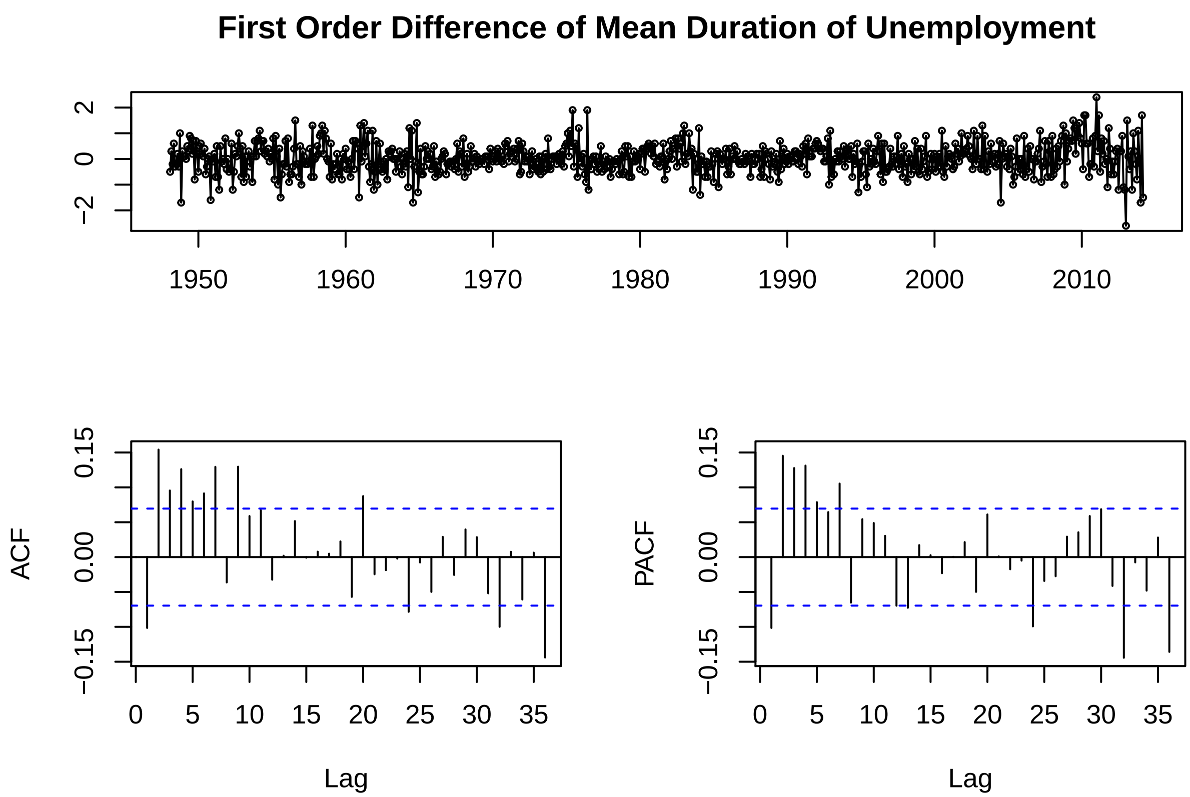


It's now obvious from the ACF that we're seeing a non-stationary process because the ACF is decreasing slowly. This leads us to want to examine the first order difference of the time series.

Part B

Focus on the change series of duration (e.g. the first differenced series). Denote the change series by r_t and let $E(r_t) = \mu$. Test $H_0 : \mu = 0$ versus the alternative $H_a : \mu \neq 0$. Draw conclusions.

```
dt1 = diff(t1)
tsdisplay(dt1, main = "First Order Difference of Mean Duration of Unemployment")
```



Well use a different test, the *Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test*, which reverses the hypothesis, so the null-hypothesis is that the data re stationary:

```
kpss.test(dt1)
```

```
## Warning in kpss.test(dt1): p-value greater than printed p-value
```

```
##
##  KPSS Test for Level Stationarity
##
## data:  dt1
## KPSS Level = 0.24219, Truncation lag parameter = 6, p-value = 0.1
```

Which, due to the p-value of `0.1` meaning we fail to reject the null hypothesis, we conclude from this test that the first order differencing of the time series is as far as we need to go (in terms of number of differencing).

We will also do a conventional t-test, which posits as the H_0 that the true mean is equal to zero.

```r
t.test(dt1)
```

```
##
##  One Sample t-test
##
## data:  dt1
## t = 1.6507, df = 793, p-value = 0.0992
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -0.006362152  0.073616560
## sample estimates:
## mean of x
## 0.0336272
```

Which, due to the p-value of `0.0992` we fail to reject the null hypothesis.

Part C

Build and AR model for the r_t series. Perform model checking using `gof = 24`. Is the model adequate? Why?

```
m1 = ar(dt1, method = "mle")
print(m1)
```

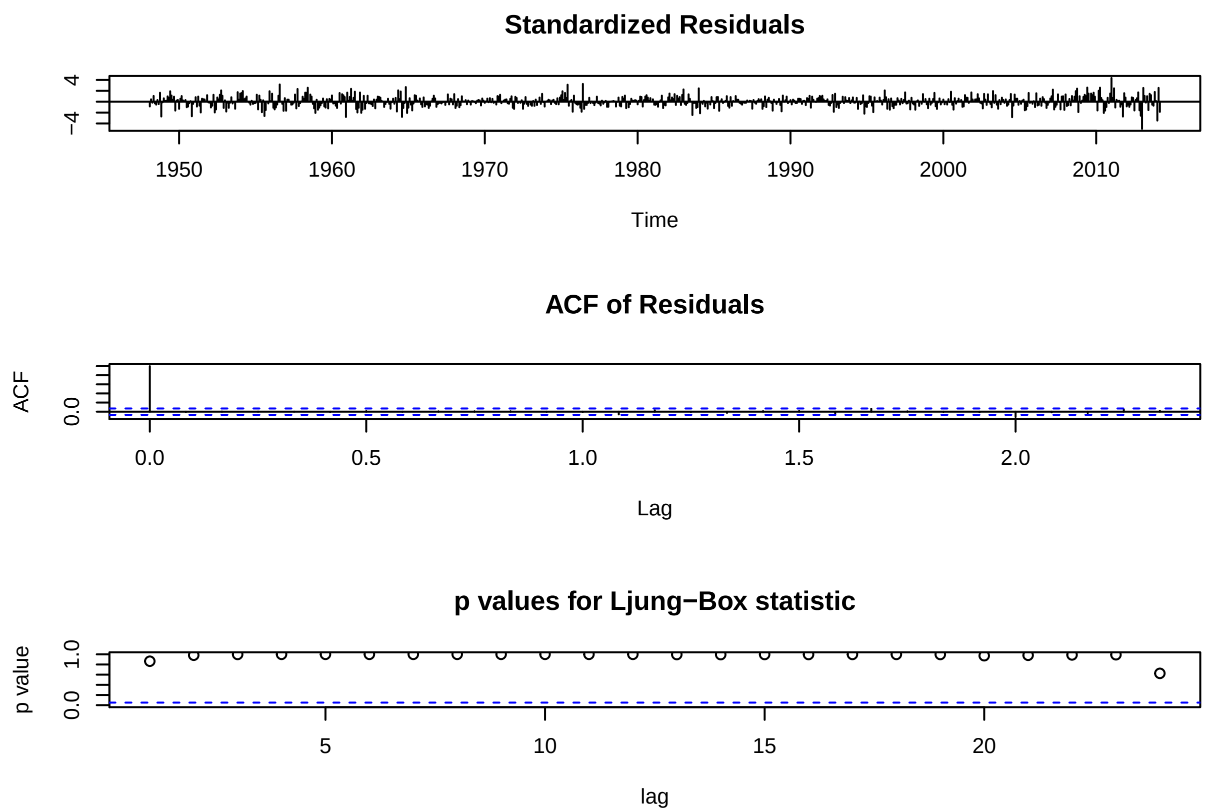
```
##
## Call:
## ar(x = dt1, method = "mle")
##
## Coefficients:
##       1        2        3        4        5        6        7        8
## -0.1364   0.1124   0.1071   0.1155   0.0729   0.0756   0.0894  -0.0618
##       9       10       11       12
##  0.0681   0.0605   0.0205  -0.0731
##
## Order selected 12  sigma^2 estimated as  0.2964
```

We'll now use the `arima` method to create an AR(12,0,0) model.

```
m2 = arima(dt1, order = c(12, 0, 0), include.mean = FALSE)
print(m2)
```

```
##
## Call:
## arima(x = dt1, order = c(12, 0, 0), include.mean = FALSE)
##
## Coefficients:
##           ar1     ar2     ar3     ar4     ar5     ar6     ar7      ar8
##       -0.1351  0.1134  0.1083  0.1165  0.0738  0.0770  0.0903  -0.0609
## s.e.   0.0355  0.0358  0.0361  0.0363  0.0365  0.0366  0.0365   0.0366
##          ar9    ar10    ar11     ar12
##       0.0687  0.0609  0.0228  -0.0723
## s.e.  0.0365  0.0366  0.0364   0.0362
##
## sigma^2 estimated as 0.2968:  log likelihood = -644.61,  aic = 1315.22
```

```
tsdiag(m2, gof.lag = 24)
```



An ACF plot of the residuals show all correlations within the threshold limits indicating that the residuals are behaving like white noise.

We will perform a Box-Pierce and Ljung-Box Test to compute a Ljung test statistic for examining the null hypothesis of independence given a time series. This is also known as a `portmanteau` test.

```
Box.test(m2$residuals, lag = 24, type = "Ljung")
```

```
##
##  Box-Ljung test
##
## data:  m2$residuals
## X-squared = 21.222, df = 24, p-value = 0.6256
```

This is testing to see if the residuals of the model look like white noise. The Ljung-Box test of the model residuals reveals a p-value that is not significant, we surmise that the model is adequate.

Part D

Write down the fitted AR model.

```r
print(m2)
```

```
##
## Call:
## arima(x = dt1, order = c(12, 0, 0), include.mean = FALSE)
##
## Coefficients:
##           ar1     ar2     ar3     ar4     ar5     ar6     ar7      ar8
##       -0.1351  0.1134  0.1083  0.1165  0.0738  0.0770  0.0903  -0.0609
## s.e.   0.0355  0.0358  0.0361  0.0363  0.0365  0.0366  0.0365   0.0366
##          ar9    ar10    ar11     ar12
##       0.0687  0.0609  0.0228  -0.0723
## s.e.  0.0365  0.0366  0.0364   0.0362
##
## sigma^2 estimated as 0.2968:  log likelihood = -644.61,  aic = 1315.22
```

$$\begin{aligned} y_t = &-0.1351 y_{t-1} + 0.1134 y_{t-2} + 0.1083 y_{t-3} + 0.1165 y_{t-4} + 0.0738 y_{t-5} + 0.077 y_{t-6} \\ &+ 0.0903 y_{t-7} - 0.0609 y_{t-8} + 0.0687 y_{t-9} + 0.0609 y_{t-10} + 0.0228 y_{t-11} - 0.0723 y_{t-12} + e_t \end{aligned} \tag{1}$$

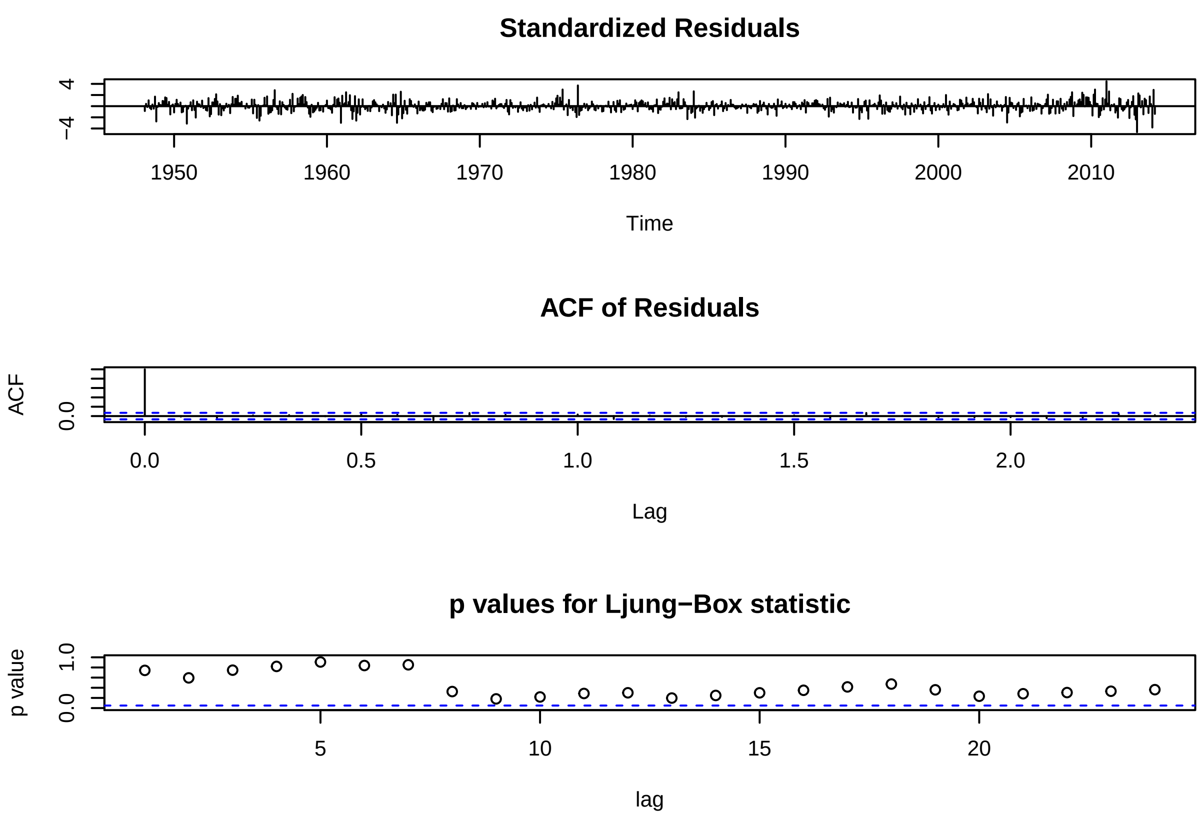
Part E

Fit a seasonal model for the r_t series using the command: `arima(r, order=(2,0,1), seasonal=list(order=c(1,0,1), period=12), include.mean = F)`. Perform model checking using `gof = 24`. Is the seasonal model adequate? Why?

```
m3 = arima(dt1, order = c(2, 0, 1), seasonal = list(order = c(1,0,1), period = 12), include.mean = FALS
print(m3)
```

```
##
## Call:
## arima(x = dt1, order = c(2, 0, 1), seasonal = list(order = c(1, 0, 1), period = 12),
##     include.mean = FALSE)
##
## Coefficients:
##          ar1     ar2      ma1    sar1     sma1
##       0.6538  0.2637  -0.8022  0.5662  -0.7429
## s.e.  0.0478  0.0360   0.0382  0.0755   0.0585
##
## sigma^2 estimated as 0.2926:  log likelihood = -639.43,  aic = 1290.85
```

```
tsdiag(m3, gof.lag = 24)
```



An ACF plot of the residuals show all correlations within the threshold limits indicating that the residuals are behaving like white noise.

We will perform a Box-Pierce and Ljung-Box Test to compute a Ljung test statistic for examining the null hypothesis of independence given a time series. This is also known as a `portmanteau` test.

```
Box.test(m3$residuals, lag = 24, type = "Ljung")
```

```
##
##  Box-Ljung test
##
## data:  m3$residuals
## X-squared = 25.813, df = 24, p-value = 0.3627
```

This is testing to see if the residuals of the model look like white noise. The Ljung-Box test of the model residuals reveals a p-value that is not significant, we surmise that the model is adequate.

Part F

Based on the in-sample fitting, which model is preferred? Why?

```r
accuracy(m2)
```

```
##                      ME     RMSE       MAE MPE MAPE      MASE         ACF1
## Training set 0.01766675 0.544767 0.4140733 NaN  Inf 0.6530631 -0.006053442
```

```r
accuracy(m3)
```

```
##                      ME      RMSE      MAE MPE MAPE     MASE        ACF1
## Training set 0.02208007 0.5409351 0.405764 NaN  Inf 0.639958 -0.01164725
```

The in-sample performance of the m2 (AR(12)) model is better than the m3 (AR(2,0,1) seasonal model.

Part G

Consider out-of-sample predictions. Use `t = 750` as the starting forecast origin. Which model is preferred based on the out-of-sample predictions?

```r
source("backtest.R")
backtest(m2, dt1, 750, 1, inc.mean = FALSE)
```

```
## [1] "RMSE of out-of-sample forecasts"
## [1] 0.9719466
## [1] "Mean absolute error of out-of-sample forecasts"
## [1] 0.7655835
```

```r
backtest(m3, dt1, 750, 1, inc.mean = FALSE)
```

```
## [1] "RMSE of out-of-sample forecasts"
## [1] 0.9440494
## [1] "Mean absolute error of out-of-sample forecasts"
## [1] 0.739132
```

It appears that in in-sample fitting the first model (AR(12)) has a higher RMSE.

Part 2

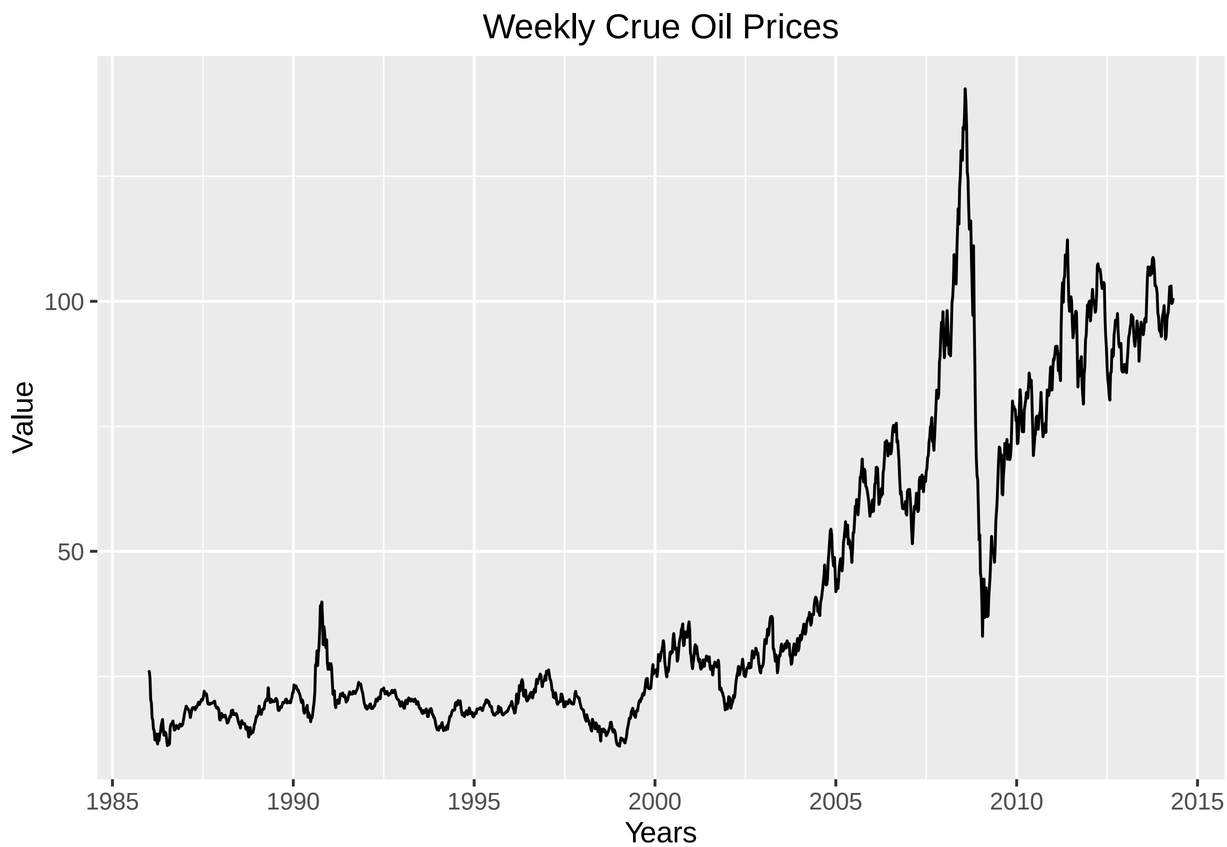
Consider the weekly crude oil prices: West Texas Intermediat (WTI), Crushing, Oklahoma. The data are available from FRED of the Federal Reserve Bank of St. Louis, and also in `w-coilwtico.txt`. The sample period is from January 3, 1986 to April 2, 2014.

```r
d2 = read.table("data/w-coilwtico.txt", header=T)
head(d2)
```

```
##   Year Mon Day Value
## 1 1986   1   3 25.78
## 2 1986   1  10 25.99
## 3 1986   1  17 24.57
## 4 1986   1  24 20.31
## 5 1986   1  31 19.69
## 6 1986   2   7 16.72
```

We'll visually examine the daa set to form initial impressions.

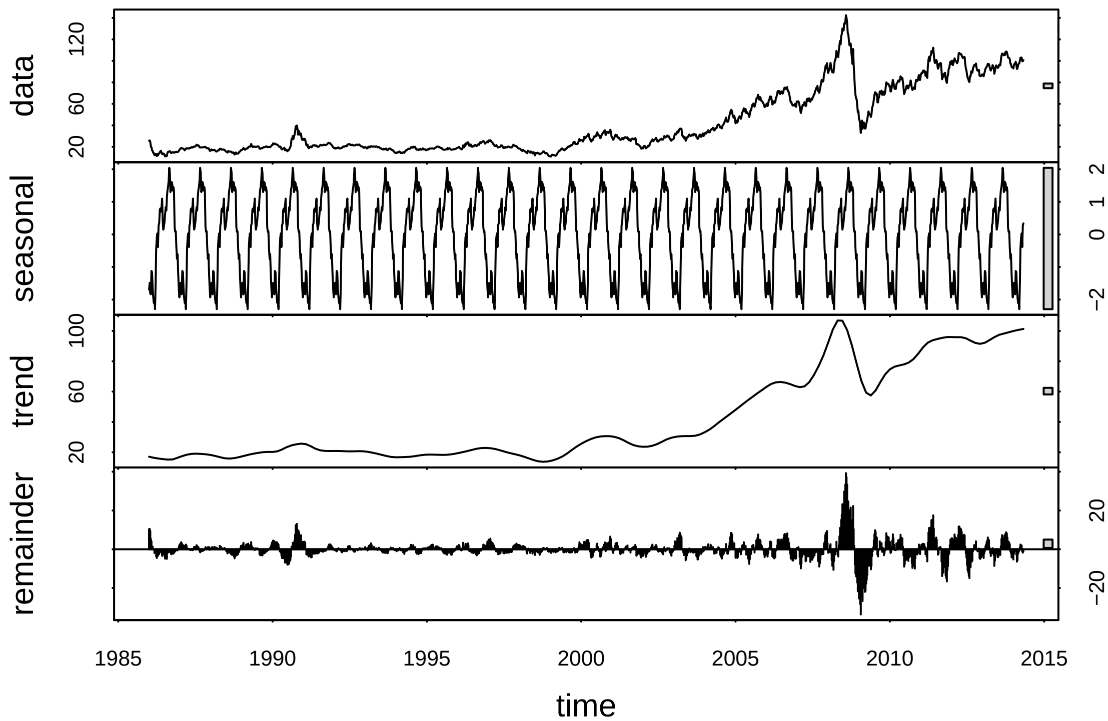
```r
t2 = ts(d2$Value, start = 1986, frequency = 52)
autoplot(t2, main = "Weekly Crue Oil Prices", ylab = "Value", xlab = "Years")
```



We can see a slightly increasing trend that has indication of seasonality and cyclic characteristics. We also notice the large increase between 2009 and 2014.

We will perform an STL decomposition to investigate our suspicion that this data has a seasonal component.

```
oil_stl = stl(t2, s.window="periodic")
plot(oil_stl)
```



We see a seasonal component. We also notice an interesting remainder in the time of high volatility.

Part A

Let r_t be the growth series (e.g. the first difference of log oil proces). Is there a serial correlation in the r_t series?

```
ldt2 = diff(log(t2))
```

We will perform a Box-Pierce and Ljung-Box Test to compute a Ljung test statistic for examining the null hypothesis of independence given a time series. This is also known as a `portmanteau` test.

```
Box.test(ldt2, type = "Ljung")
```

```
##
##  Box-Ljung test
##
## data:  ldt2
## X-squared = 14.079, df = 1, p-value = 0.0001753
```

From the p-value of `0.0001753` we must reject H_0. This is an indicator that there are some significant serial correlations at the 5% level for the first order difference series.

Part B

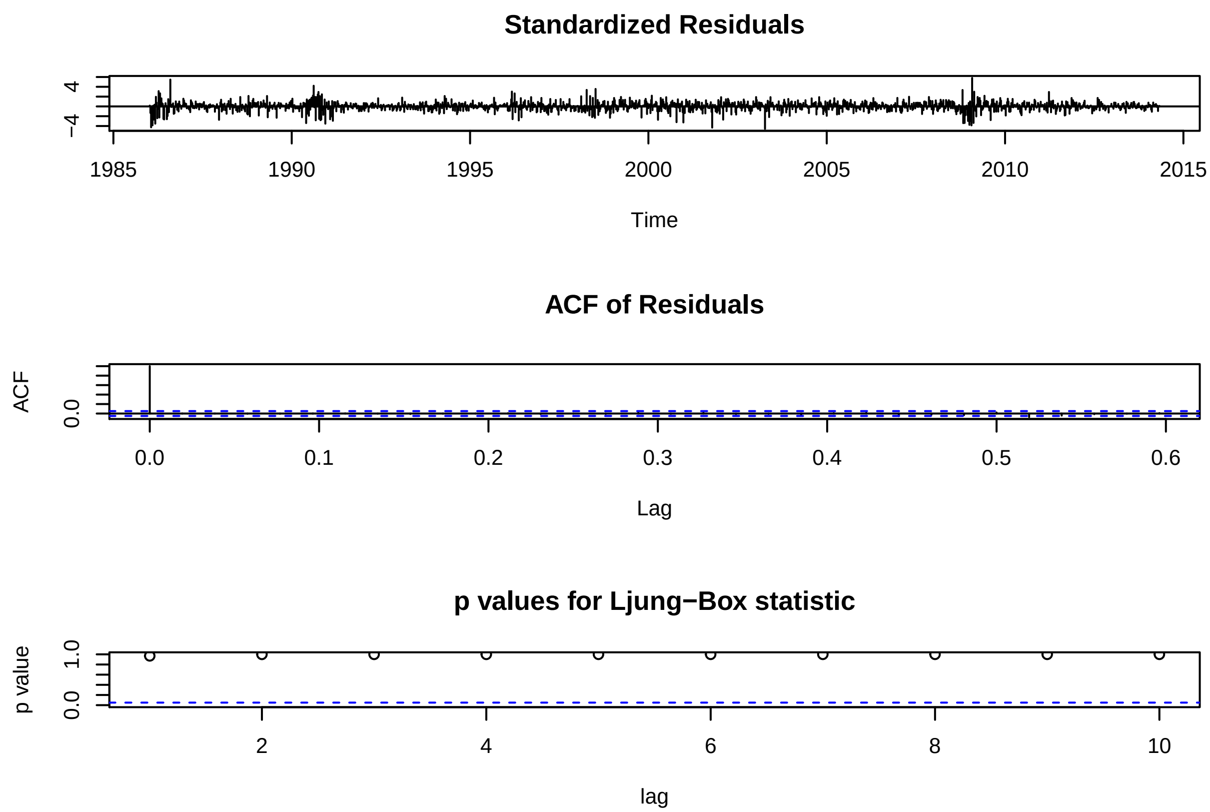
Build an AR model for r_t. Check the adequacy of the model, and write down the model.

```r
m4 = ar(ldt2, type="mle")
m5 = arima(ldt2, order=c(16, 0, 0))
print(m5)
```

```
##
## Call:
## arima(x = ldt2, order = c(16, 0, 0))
##
## Coefficients:
##          ar1      ar2     ar3     ar4      ar5      ar6      ar7     ar8
##       0.1073  -0.0492  0.1115  0.0364  -0.0225  -0.0210  -0.0321  0.1034
## s.e.  0.0260   0.0261  0.0263  0.0264   0.0265   0.0264   0.0265  0.0265
##           ar9    ar10     ar11    ar12     ar13    ar14     ar15     ar16
##       -0.0078  0.0275  -0.1022  0.0266  -0.0016  0.0654  -0.0307  -0.0568
## s.e.   0.0267  0.0267   0.0267  0.0268   0.0268  0.0268   0.0268   0.0267
##       intercept
##          0.0009
## s.e.     0.0013
##
## sigma^2 estimated as 0.001811:  log likelihood = 2559.96,  aic = -5083.91
```

$$
\begin{aligned}
y_t = 0.1067y_{t-1} - 0.0485y_{t-2} + 0.1098y_{t-3} + 0.0353y_{t-4} \\
- 0.0227y_{t-5} - 0.0228_{t-6} - 0.0307y_{t-7} + 0.0993y_{t-8} \\
- 0.0047y_{t-9} + 0.0229y_{t-10} - 0.0975y_{t-11} + 0.0233y_{t-12} \\
+ 0.0011y_{t-13} + 0.0625y_{t-14} - 0.0266y_{t-15} - 0.0571y_{t-16} + e_t \quad (2)
\end{aligned}
$$

```r
tsdiag(m5)
```



An ACF plot of the residuals show all correlations within the threshold limits indicating that the residuals are behaving like white noise.

We will perform a Box-Pierce and Ljung-Box Test to compute a Ljung test statistic for examining the null hypothesis of independence given a time series. This is also known as a `portmanteau` test.

```
Box.test(m5$residuals, type = "Ljung")
```

```
##
##  Box-Ljung test
##
## data:  m5$residuals
## X-squared = 0.0014717, df = 1, p-value = 0.9694
```

This is testing to see if the residuals of the model look like white noise. The Ljung-Box test of the model residuals reveals a p-value that is not significant, we surmise that the model is adequate.

Part C

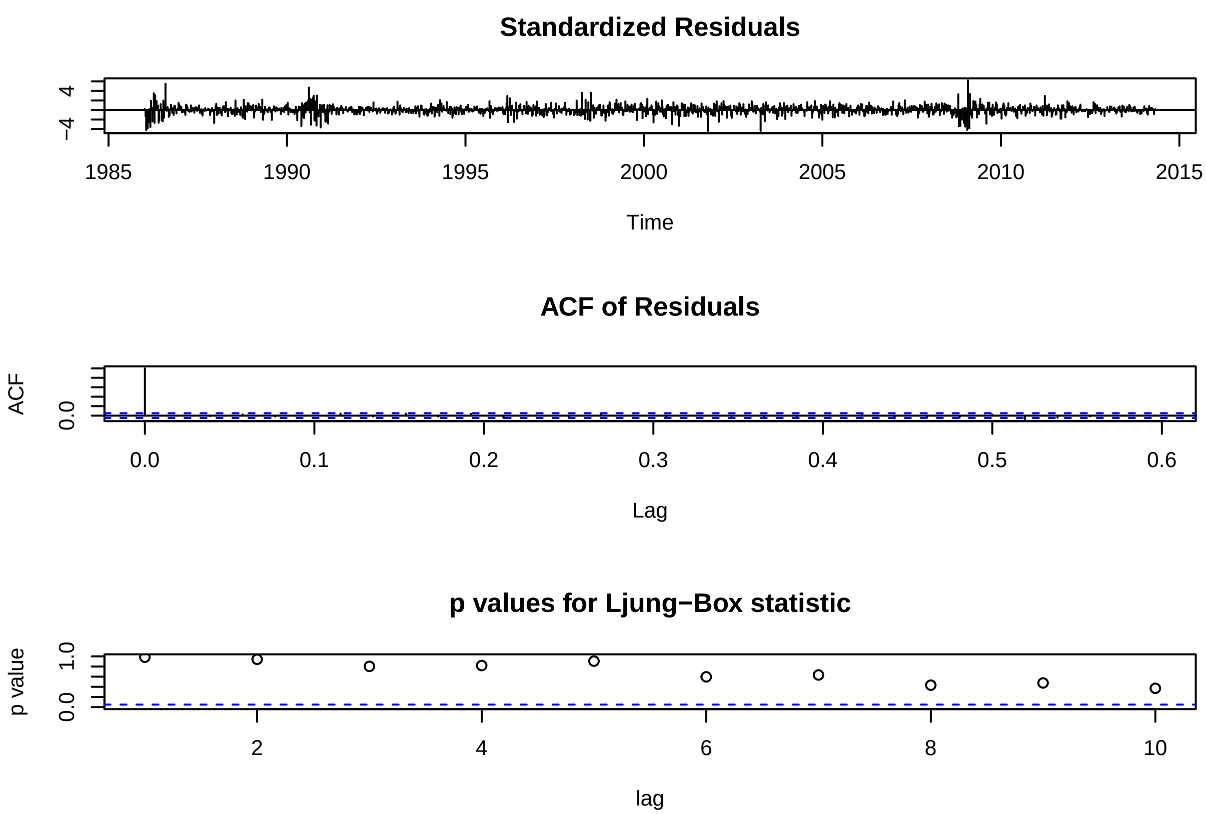
Fit another model to r_t using the following command: `arima(r, order=c(3,0,2), include.mean = F)` This is an ARIMA(3,0,2) model, write down the model. Based on in-sample fitting, which model is preferred?

```
m6 = arima(ldt2, order=c(3, 0, 2), include.mean = FALSE)
print(m6)
```

```
##
## Call:
## arima(x = ldt2, order = c(3, 0, 2), include.mean = FALSE)
##
## Coefficients:
##          ar1      ar2     ar3      ma1     ma2
##       0.5664  -0.8548  0.1689  -0.4680  0.7753
## s.e.  0.0934   0.0681  0.0270   0.0931  0.0680
##
## sigma^2 estimated as 0.001845:  log likelihood = 2546.4,  aic = -5080.8
```

$$y_t = 0.5664y_{t-1} - 0.8548y_{t-2} + 0.1689y_{t-3} + e_t - 0.4680e_{t-1} + 0.7753e_{t-2}$$

```
tsdiag(m6)
```



```r
accuracy(m5)
```

```
##                        ME       RMSE        MAE MPE MAPE      MASE
## Training set 3.746386e-06 0.04255399 0.03098926 NaN  Inf 0.7432954
##                     ACF1
## Training set 0.000998543
```

```r
accuracy(m6)
```

```
##                        ME       RMSE        MAE MPE MAPE      MASE
## Training set 0.0007899324 0.04295061 0.03105575 NaN  Inf 0.7448904
##                      ACF1
## Training set 0.0005752462
```

It appears that the m6 model (AR(3,0,2)) has better accuracy than our MLE found AR(16) model.