Lab3_Initial_Legg

Sue Yang, Michelle Kim, Legg Yeung

July 28, 2017

```
rm(list = ls())
library(moments)
library(psych)
```

```
## Warning: package 'psych' was built under R version 3.3.3
```

```
unem.data = read.csv("UNRATENSA.csv", header = T)
auto.data = read.csv("TOTALNSA.csv", header = T)
```

EDA

Data Overview

```
str(unem.data)
```

```
## 'data.frame':    834 obs. of  2 variables:
##  $ DATE     : Factor w/ 834 levels "1948-01-01","1948-02-01",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ UNRATENSA: num  4 4.7 4.5 4 3.4 3.9 3.9 3.6 3.4 2.9 ...
```

```
str(auto.data)
```

```
## 'data.frame':    498 obs. of  2 variables:
##  $ DATE    : Factor w/ 498 levels "1976-01-01","1976-02-01",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ TOTALNSA: num  885 995 1244 1191 1203 ...
```

```
cbind(head(unem.data),tail(unem.data))
```

```
##         DATE UNRATENSA       DATE UNRATENSA
## 1 1948-01-01       4.0 2017-01-01       5.1
## 2 1948-02-01       4.7 2017-02-01       4.9
## 3 1948-03-01       4.5 2017-03-01       4.6
## 4 1948-04-01       4.0 2017-04-01       4.1
## 5 1948-05-01       3.4 2017-05-01       4.1
## 6 1948-06-01       3.9 2017-06-01       4.5
```

```
cbind(head(auto.data),tail(auto.data))
```

```
##         DATE TOTALNSA       DATE TOTALNSA
## 1 1976-01-01    885.2 2017-01-01   1164.3
## 2 1976-02-01    994.7 2017-02-01   1352.1
## 3 1976-03-01   1243.6 2017-03-01   1582.7
## 4 1976-04-01   1191.2 2017-04-01   1449.7
## 5 1976-05-01   1203.2 2017-05-01   1544.1
## 6 1976-06-01   1254.7 2017-06-01   1500.6
```

```
nrow(unem.data) - nrow(auto.data)
```

```
## [1] 336
```

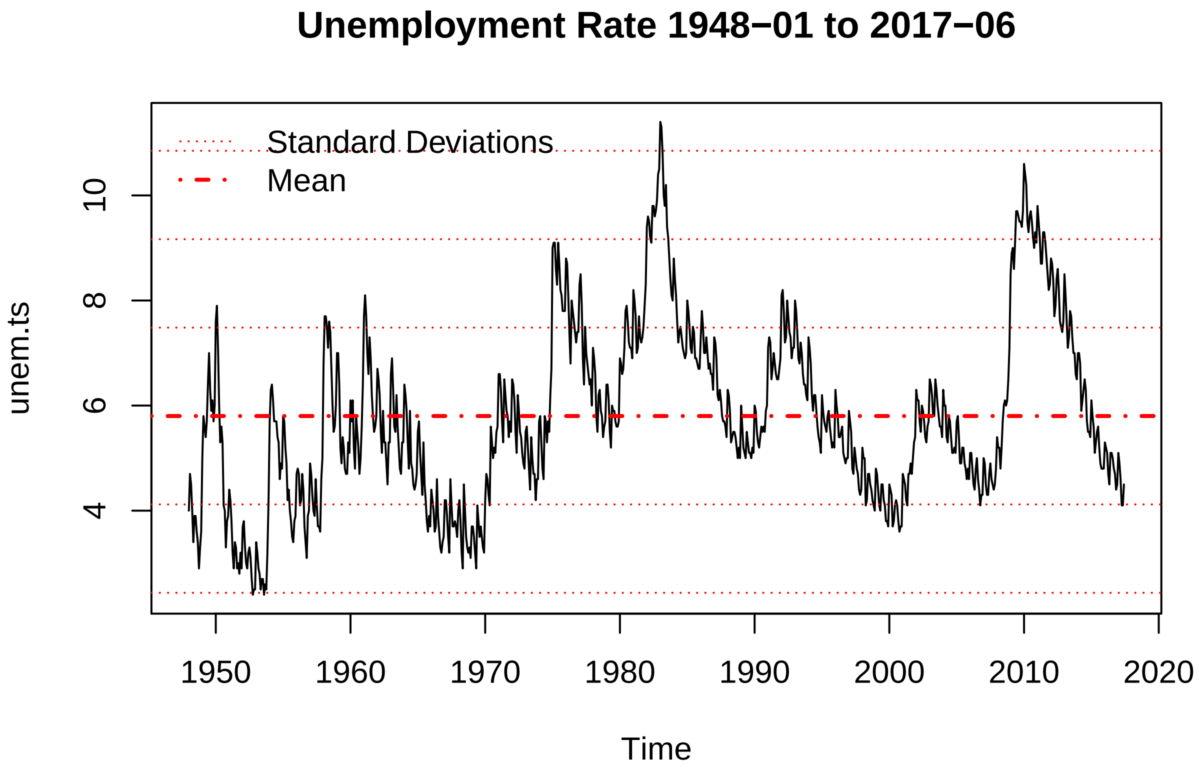
Both datasets are time indexed, accompanied with a key variable of interests. With UNRATENSA, the key variable refers to unemployment rate. With TOTALNSA, the key variable refers to car sale. Both time series presents monthly data. UNRATENSA has 834 observations and starts from 1948-01-01. TOTALNSA has 498 observations and starts from 1976-01-01. Both ends at 2017-06. UNRATENSA has 28 more years, that is 336 more observations than TOTALNSA.

Time series Overview

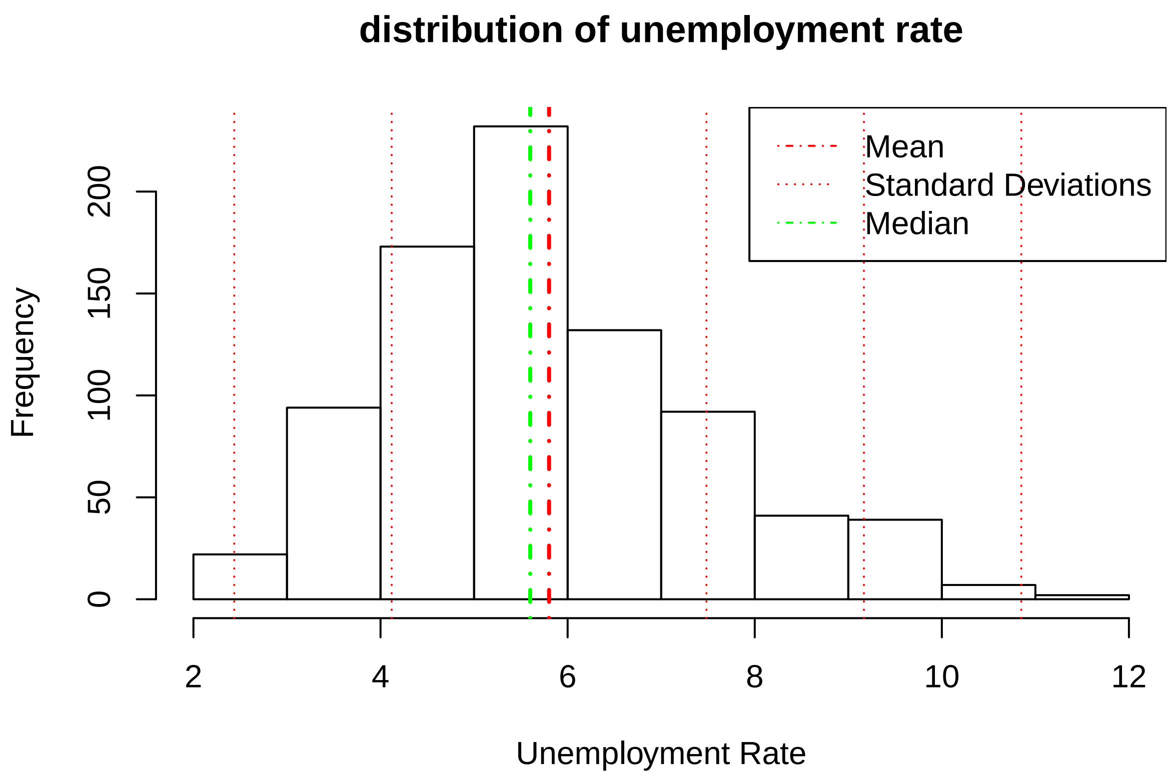
Time plots and Histograms

```
unem.ts = ts(unem.data$UNRATENSA, frequency = 12, start = c(1948,1))
auto.ts = ts(auto.data$TOTALNSA, frequency = 12, start = c(1976,1))
```

```
ts.plot(unem.ts); title("Unemployment Rate 1948-01 to 2017-06");
abline(h = mean(unem.ts), col = "red", lty = "dotdash", lwd = 2)
abline(h = c((mean(unem.ts) + sd(unem.ts)),
             (mean(unem.ts) + 2*sd(unem.ts)),
             (mean(unem.ts) + 3*sd(unem.ts))),
       col = "red", lty = "dotted", lwd = 1)
abline(h = c((mean(unem.ts) - sd(unem.ts)),
             (mean(unem.ts) - 2*sd(unem.ts))),
       col = "red", lty = "dotted", lwd = 1)

#abline(lm(unem.ts~time(unem.ts)), lty = "dotted", col = "blue")
legend("topleft", c("Standard Deviations","Mean"),
       col = c("red","red"),
       lty = c("dotted", "dotdash"), bty = "n",
       lwd = c(1,2))
```



```
hist(unem.data$UNRATENSA, main = "distribution of unemployment rate",
     xlab = "Unemployment Rate")

abline(v = mean(unem.data$UNRATENSA), col = "red", lty = "dotdash"
       , lwd = 2)
abline(v = c((mean(unem.data$UNRATENSA) + sd(unem.data$UNRATENSA)),
             (mean(unem.data$UNRATENSA) + 2*sd(unem.data$UNRATENSA)),
             (mean(unem.data$UNRATENSA) + 3*sd(unem.data$UNRATENSA))),
       col = "red", lty = "dotted", lwd = 1)
abline(v = c((mean(unem.data$UNRATENSA) - sd(unem.data$UNRATENSA)),
             (mean(unem.data$UNRATENSA) - 2*sd(unem.data$UNRATENSA))),
       col = "red", lty = "dotted", lwd = 1)

abline(v = median(unem.data$UNRATENSA), col = "green", lty = "dotdash"
       , lwd = 2)

legend("topright", c("Mean","Standard Deviations", "Median"),
       col = c("red","red","green"),
       lty = c("dotdash", "dotted","dotdash"))
```



```
unem.data[unem.data$UNRATENSA > mean(unem.data$UNRATENSA) + 2.5*sd(unem.data$UNRATENSA),]
```

```
##           DATE UNRATENSA
## 419 1982-11-01      10.4
## 420 1982-12-01      10.5
## 421 1983-01-01      11.4
## 422 1983-02-01      11.3
## 423 1983-03-01      10.8
## 426 1983-06-01      10.2
## 745 2010-01-01      10.6
## 746 2010-02-01      10.4
## 747 2010-03-01      10.2
```

```
# precentage of observations beyond 2, 2.5 and 3 sd
nrow(unem.data[unem.data$UNRATENSA > mean(unem.data$UNRATENSA)
               + 2*sd(unem.data$UNRATENSA),])/nrow(unem.data)
```

```
## [1] 0.04916067
```

```
nrow(unem.data[unem.data$UNRATENSA > mean(unem.data$UNRATENSA)
               + 2.5*sd(unem.data$UNRATENSA),])/nrow(unem.data)
```

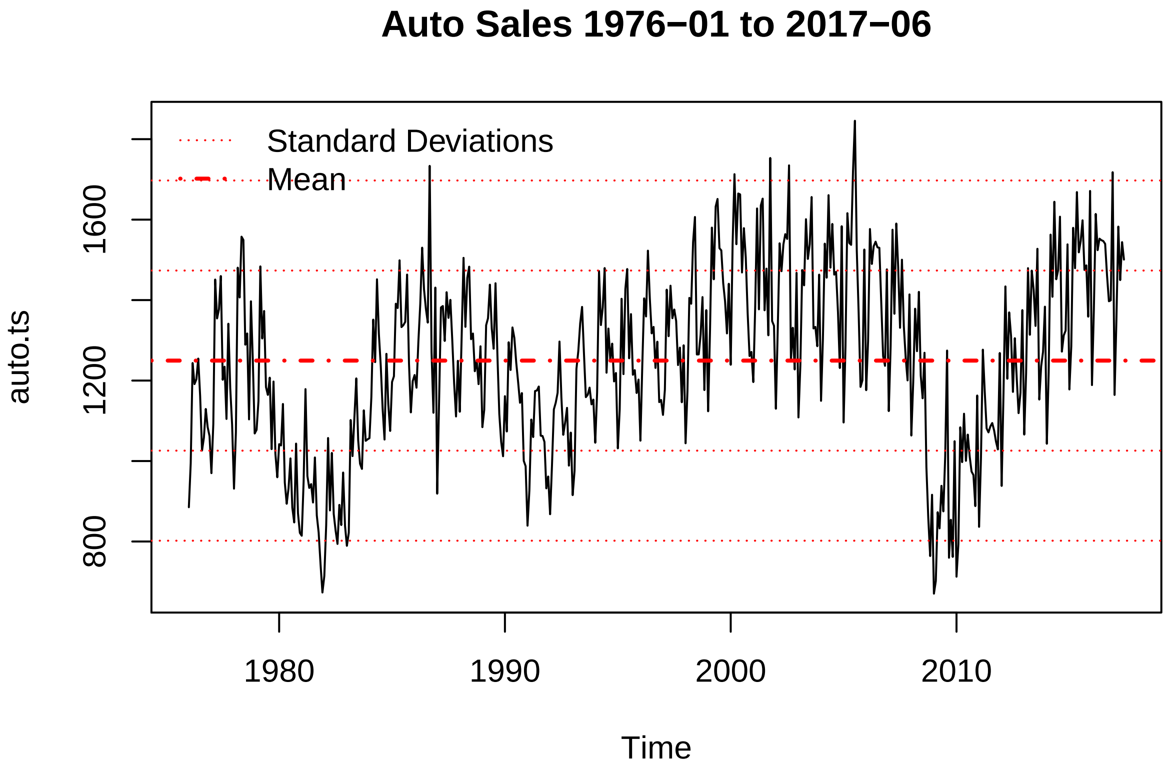
```
## [1] 0.01079137
```

```
nrow(unem.data[unem.data$UNRATENSA > mean(unem.data$UNRATENSA)
               + 3*sd(unem.data$UNRATENSA),])/nrow(unem.data)
```

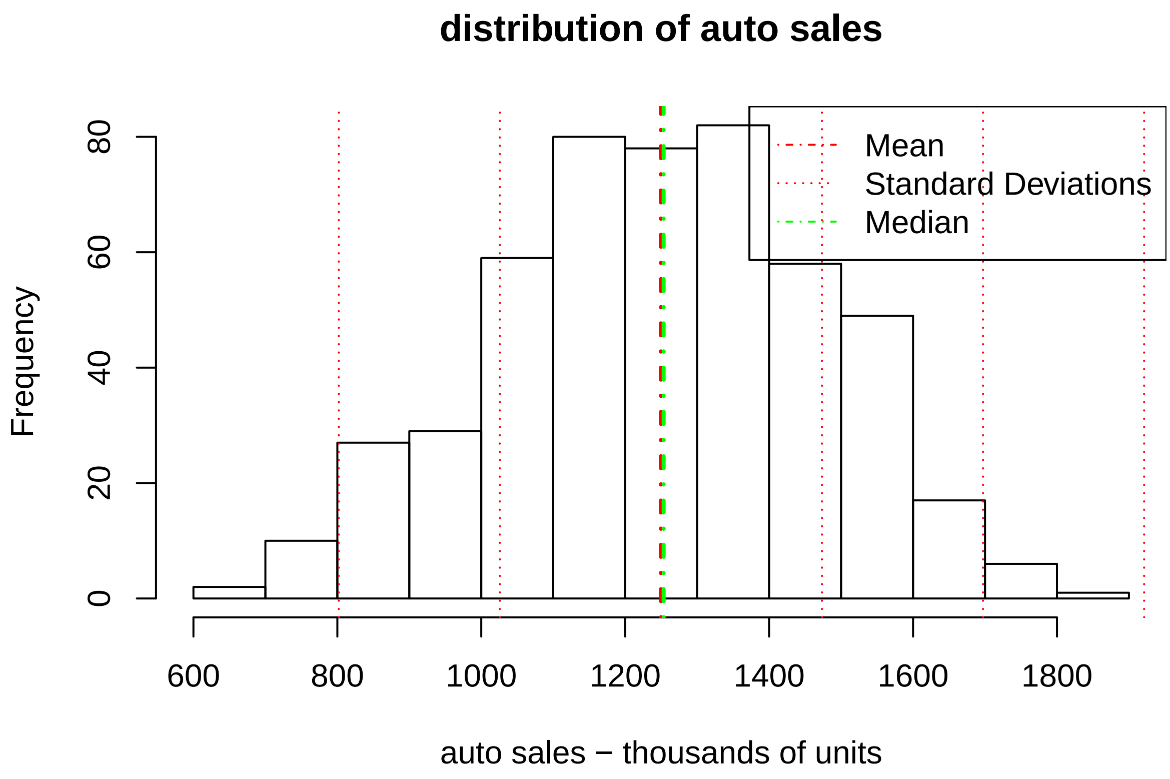
```
## [1] 0.002398082
```

From the above time plot of unemployment, we see clear persistency of the observations. That is, when observations are above or below the mean, they tend to stay so for a while. The overall trend seem to climb slowly upward. Almost 5% of the observations lie 2 standard deviations above the mean, which is made obvious in the right skewed histogram as well. We isolated these observations, the 6 in 1982 and 1983 probably correspond to the early 1980s recessions which officially ended in November 1982. The 3 in 2010 probably correspond to the late 2000s recession which officially ended in June 2009.

```
ts.plot(auto.ts); title("Auto Sales 1976-01 to 2017-06")
abline(h = mean(auto.ts), col = "red", lty = "dotdash", lwd = 2)
abline(h = c((mean(auto.ts) + sd(auto.ts)),
             (mean(auto.ts) + 2*sd(auto.ts)),
             (mean(auto.ts) + 3*sd(auto.ts))),
       col = "red", lty = "dotted", lwd = 1)
abline(h = c((mean(auto.ts) - sd(auto.ts)),
             (mean(auto.ts) - 2*sd(auto.ts))),
       col = "red", lty = "dotted", lwd = 1)

#abline(lm(auto.ts~time(auto.ts)), lty = "dotted", col = "blue")
legend("topleft", c("Standard Deviations","Mean"),
       col = c("red","red"),
       lty = c("dotted", "dotdash"), bty = "n",
       lwd = c(1,2))
```



```
hist(auto.data$TOTALNSA, main = "distribution of auto sales",
     xlab = "auto sales - thousands of units")

abline(v = mean(auto.data$TOTALNSA), col = "red", lty = "dotdash"
       , lwd = 2)
abline(v = c((mean(auto.data$TOTALNSA) + sd(auto.data$TOTALNSA)),
             (mean(auto.data$TOTALNSA) + 2*sd(auto.data$TOTALNSA)),
             (mean(auto.data$TOTALNSA) + 3*sd(auto.data$TOTALNSA))),
       col = "red", lty = "dotted", lwd = 1)
abline(v = c((mean(auto.data$TOTALNSA) - sd(auto.data$TOTALNSA)),
             (mean(auto.data$TOTALNSA) - 2*sd(auto.data$TOTALNSA))),
       col = "red", lty = "dotted", lwd = 1)

abline(v = median(auto.data$TOTALNSA), col = "green", lty = "dotdash"
       , lwd = 2)

legend("topright", c("Mean","Standard Deviations", "Median"),
       col = c("red","red","green"),
       lty = c("dotdash", "dotted","dotdash"))
```



```
auto.data[auto.data$TOTALNSA < mean(auto.data$TOTALNSA) - 2*sd(auto.data$TOTALNSA),]
```

```
##           DATE TOTALNSA
## 71  1981-11-01    743.0
## 72  1981-12-01    673.2
## 73  1982-01-01    714.4
## 80  1982-08-01    794.0
## 85  1983-01-01    789.3
## 395 2008-11-01    763.9
## 397 2009-01-01    670.4
## 398 2009-02-01    701.7
## 405 2009-09-01    759.6
## 407 2009-11-01    761.8
## 409 2010-01-01    712.5
## 410 2010-02-01    793.3
```

```
head(auto.data[auto.data$TOTALNSA > mean(auto.data$TOTALNSA) + 2*sd(auto.data$TOTALNSA),],1)
```

```
##           DATE TOTALNSA
## 129 1986-09-01   1733.6
```

```
# precentage of observations beyond 2 sd
nrow(auto.data[auto.data$TOTALNSA > mean(auto.data$TOTALNSA)
               + 2*sd(auto.data$TOTALNSA),])/nrow(auto.data)
```

```
## [1] 0.01405622
```

```
nrow(auto.data[auto.data$TOTALNSA < mean(auto.data$TOTALNSA)
               - 2*sd(auto.data$TOTALNSA),])/nrow(auto.data)
```

```
## [1] 0.02409639
```

From the above time plot of auto sales, we see some but weaker persistency than the unemployment series. The overall trend doesn't seem to climb upward or downward. There are more noticeable seasonal patterns than the unemployment series. The histogram is more symmetric and normal. Less than 1.5% observations lie 2 standard deviations above the mean and less than 2.5% observations lie 2 standard deviations below the mean. We isolated these observations, the 5 in 1982 and 1983 and the 5 in 2008 and 2009 probably correspond to the two aforementioned recessions. Notice that these recession related observations in auto sales tend to happen a few months before those in unemployment. There is an unusual spike in 1986, probably attributed to oil prices dropping in half that year.

```
summary(unem.data$UNRATENSA)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.400   4.700   5.600   5.801   6.900  11.400
```

```
summary(auto.data$TOTALNSA)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   670.4  1095.0  1253.0  1250.0  1407.0  1846.0
```

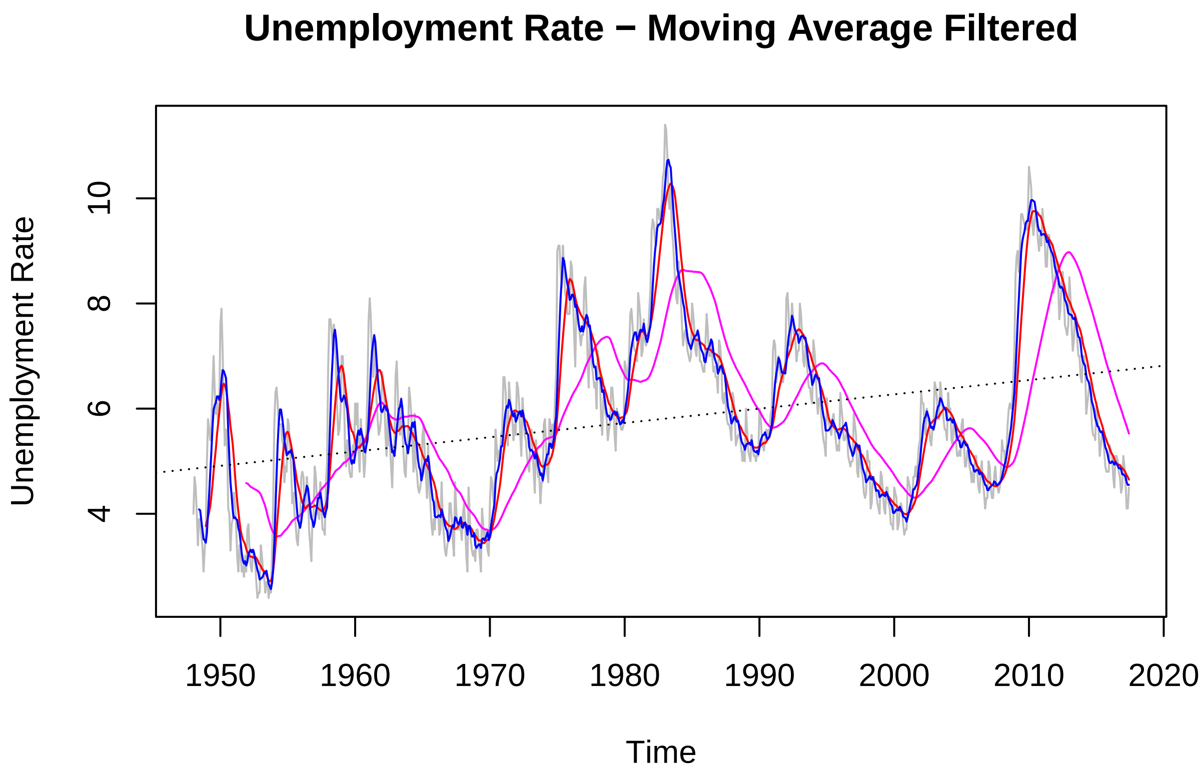
```
moments::kurtosis(unem.data$UNRATENSA)
```

```
## [1] 3.069337
```

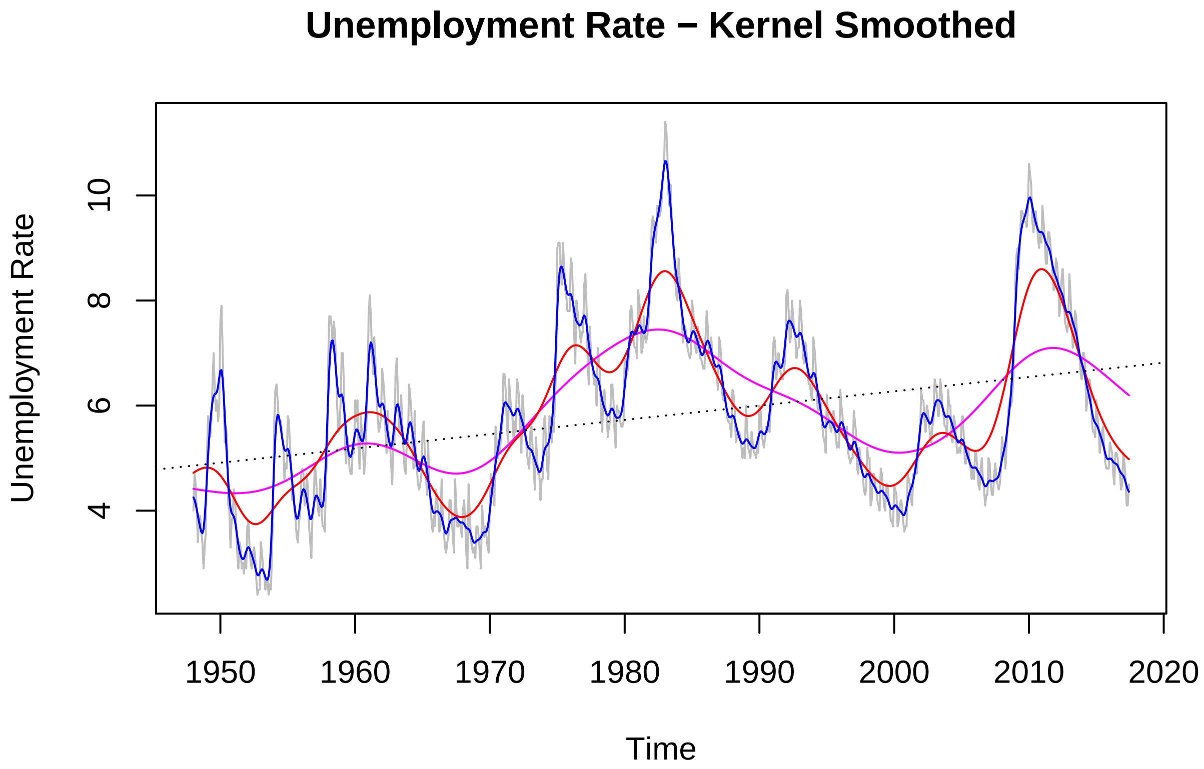
Trend Examination with Smoothers and Decomposition - Unemployment

In this section, we apply smoothers and decomposition to gain some initial insights about the overall trend, seasonality and noise behavior of both series.

```
# Moving Average Filter
unem.ma.smooth.4year = filter(unem.ts, sides = 1, rep(1/48, 48))
unem.ma.smooth.annual = filter(unem.ts, sides = 1, rep(1/12, 12))
unem.ma.smooth.halfyear = filter(unem.ts, sides = 1, rep(1/6, 6))

# Make plot
plot(unem.ts, col = "gray", ylab = "Unemployment Rate",
     main = "Unemployment Rate - Moving Average Filtered")
lines(unem.ma.smooth.4year, col = "magenta")
lines(unem.ma.smooth.annual, col = "red")
lines(unem.ma.smooth.halfyear, col = "blue")
abline(lm(unem.ts~time(unem.ts)), lty = "dotted", col = "black")
```



```
# Kernel smoothing
unem.k.smooth.widest = ksmooth(time(unem.ts),
                               unem.ts, kernel = c("normal"),
                               bandwidth = 10)

unem.k.smooth.wide = ksmooth(time(unem.ts),
                             unem.ts, kernel = c("normal"),
                             bandwidth = 4)

unem.k.smooth.narrow = ksmooth(time(unem.ts),
                                 unem.ts, kernel = c("normal"),
                                 bandwidth = 0.5)

# Make plot
plot(unem.ts, col = "gray", ylab = "Unemployment Rate",
     main = "Unemployment Rate - Kernel Smoothed")
lines(unem.k.smooth.widest$x, unem.k.smooth.widest$y, col = "magenta")
lines(unem.k.smooth.wide$x, unem.k.smooth.wide$y, col = "red")
lines(unem.k.smooth.narrow$x, unem.k.smooth.narrow$y, col = "blue")
abline(lm(unem.ts~time(unem.ts)), lty = "dotted", col = "black")
```



The asymmetric moving average filters above average over the past 6 months, 12 months and 4 years. The symmetric kernel smoothers attempted 3 bandwidths. In either case, the smoothed series resemble behavior of random walks with drift, evidence by the gradually widened variance and slowly increasing mean towards the right. Formulation for the two smoothers are given here:

- One-sided Moving Average Smoother

$$m_t = \frac{1}{n+1}\sum_{j=0}^{n} x_{t-j}$$

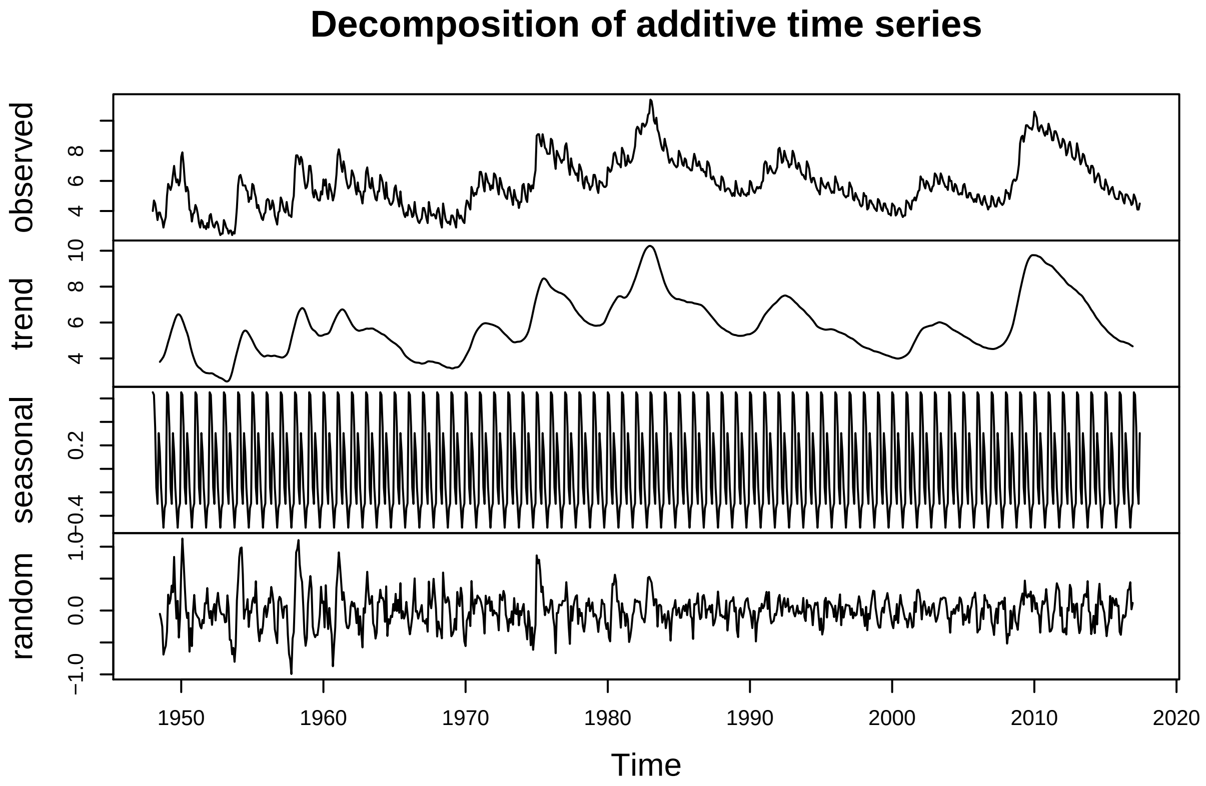
where m_t refers to the trend we hope to study, x_t is the raw series, n is the number of past months to include in the averaging.

- Symmertic Kernel Smoother (symmetric moving average smoother with probability density weight function applied)

$$m_t = \sum_{i=1}^{n} w_i(t)x_{t_i}$$

where density weight function $w_i(t)$ is a function of the kernel function and $w_i(t) = K(\frac{t-t_i}{b})/\sum_{j=1}^{n} K(\frac{t-t_j}{b})$. b is the bandwidth which we manipulate. The blue kernel curve used $b = 0.5$ to correspond approximately smoothing over about half a year. The smoother curves were generated using higher bandwidth.

```r
plot(decompose(unem.ts, type = "additive"))
```



The decomposed trend series resembles that of the moving average and kernel filters. Here the growing variance of the trend is more noticeable, as well as the gradual upward trend. A regular, annual seasonality series was isolated out from the raw series. The random component series is clearly non-stationary. Its variance diminishes over time featured by dramatic spikes before 1965. Clearly, an OLS model would not be the right choice, we should not disregard the non-stationary trend and random components.

The techniques assumes that the raw series is composed of a clear trend, seasonality and random component. The model imposed is:

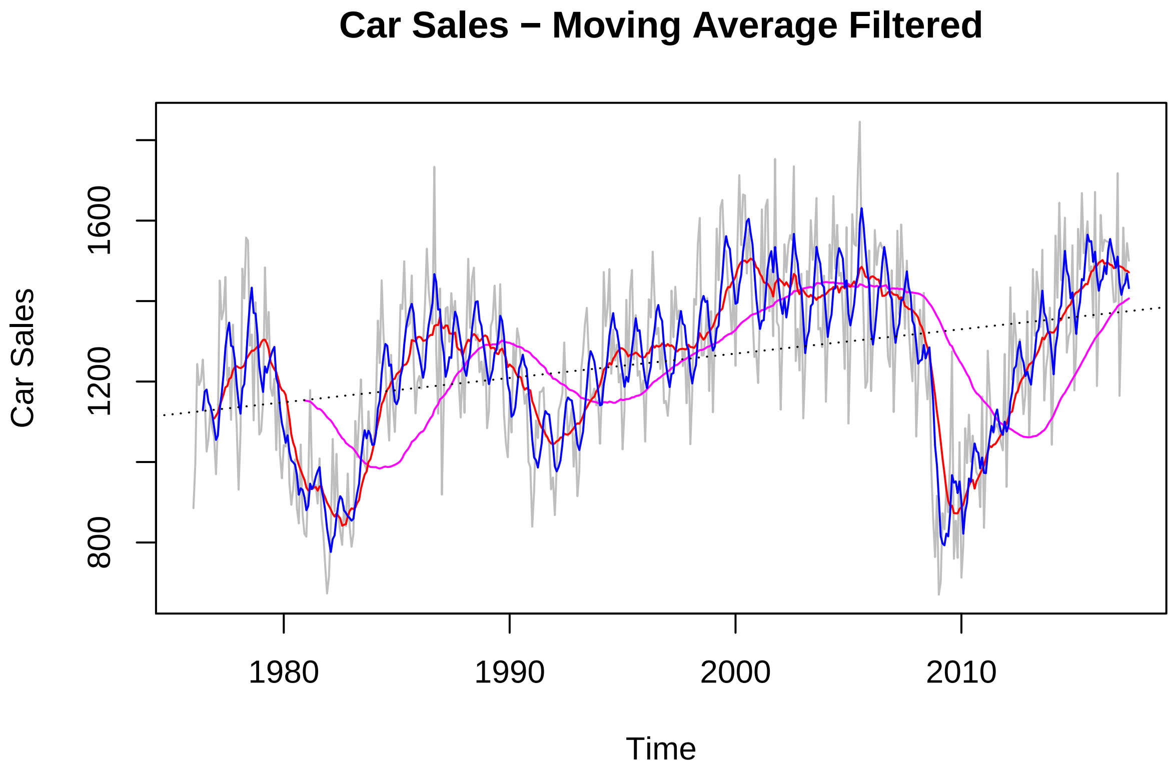
$$X_t = M_t + S_t + N_t$$

where X_t is the raw series, M_t is the trend, S_t is the seasonal component and N_t is the random component.

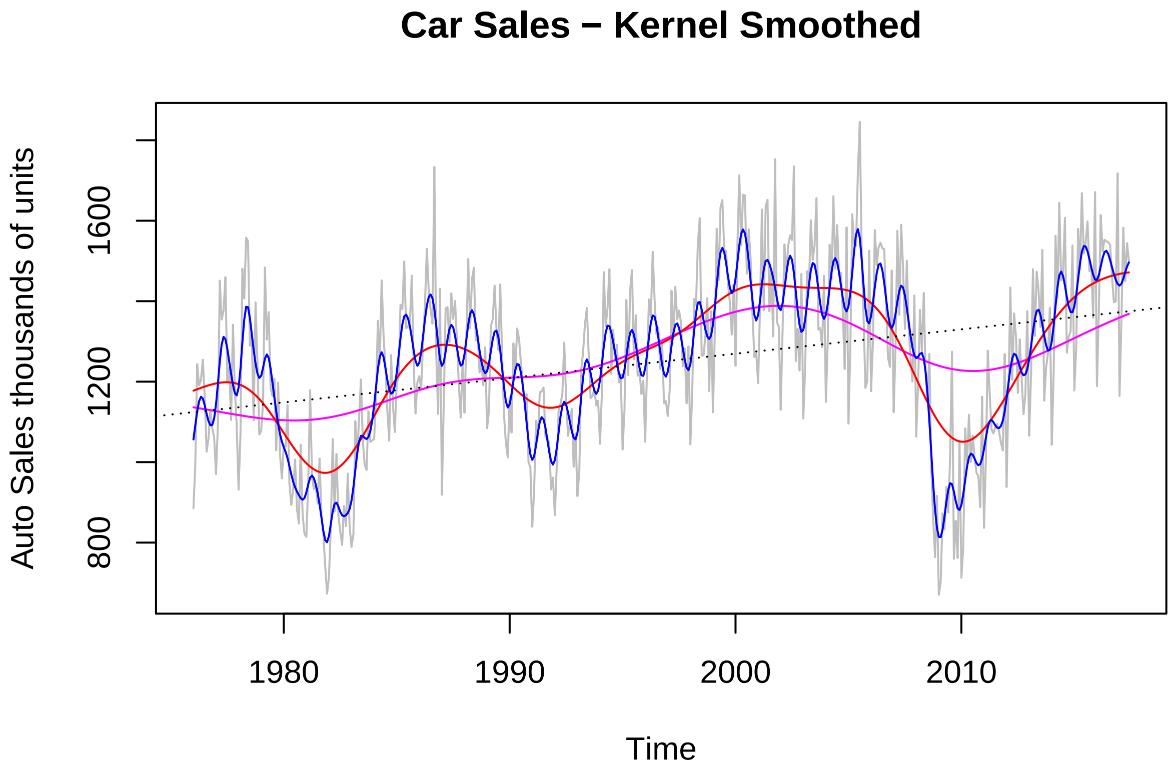
Trend Examination with Smoothers and Decomposition - Unemployment

```
# Moving Average Filter
auto.ma.smooth.4year = filter(auto.ts, sides = 1, rep(1/60, 60))
auto.ma.smooth.annual = filter(auto.ts, sides = 1, rep(1/12, 12))
auto.ma.smooth.halfyear = filter(auto.ts, sides = 1, rep(1/6, 6))

# Make plot
plot(auto.ts, col = "gray", ylab = "Car Sales",
     main = "Car Sales - Moving Average Filtered")
lines(auto.ma.smooth.4year, col = "magenta")
lines(auto.ma.smooth.annual, col = "red")
```

```r
lines(auto.ma.smooth.halfyear, col = "blue")
abline(lm(auto.ts~time(auto.ts)), lty = "dotted", col = "black")
```

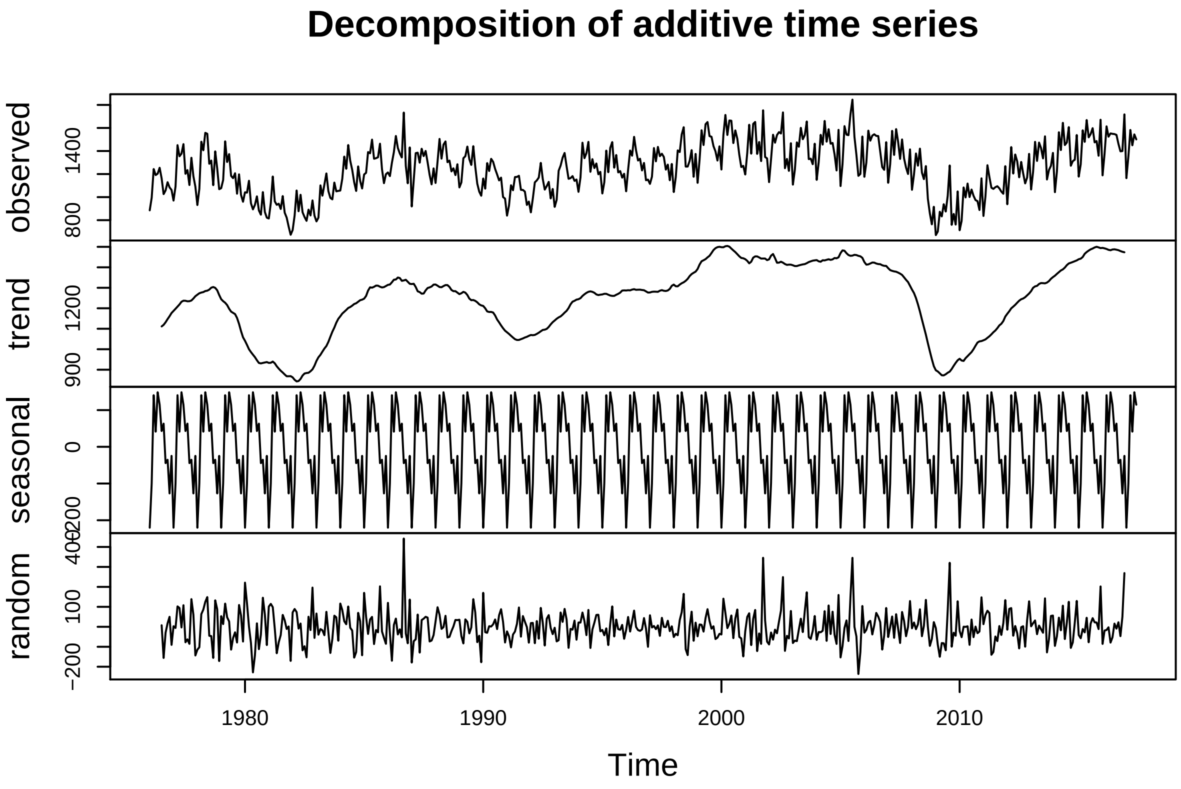


```r
# Kernel smoothing
auto.k.smooth.widest = ksmooth(time(auto.ts),
                                auto.ts, kernel = c("normal"),
                                bandwidth = 10)

auto.k.smooth.wide = ksmooth(time(auto.ts),
                                auto.ts, kernel = c("normal"),
                                bandwidth = 4)

auto.k.smooth.narrow = ksmooth(time(auto.ts),
                                  auto.ts, kernel = c("normal"),
                                  bandwidth = 0.5)

# Make plot
plot(auto.ts, col = "gray", ylab = "Auto Sales thousands of units",
     main = "Car Sales - Kernel Smoothed")
lines(auto.k.smooth.widest$x, auto.k.smooth.widest$y, col = "magenta")
lines(auto.k.smooth.wide$x, auto.k.smooth.wide$y, col = "red")
lines(auto.k.smooth.narrow$x, auto.k.smooth.narrow$y, col = "blue")
abline(lm(auto.ts~time(auto.ts)), lty = "dotted", col = "black")
```



Compared to the unemployment series, the auto sale series exhibit less downhill-uphill-downhill behavior, and appear to be relatively stable between the two aforementioned recessions in early 1980s and late 2000s. It resembles less of a random walk and drift is not entirely apparent. Seasonal pattern is also stronger.

```
plot(decompose(auto.ts, type = "additive"))
```

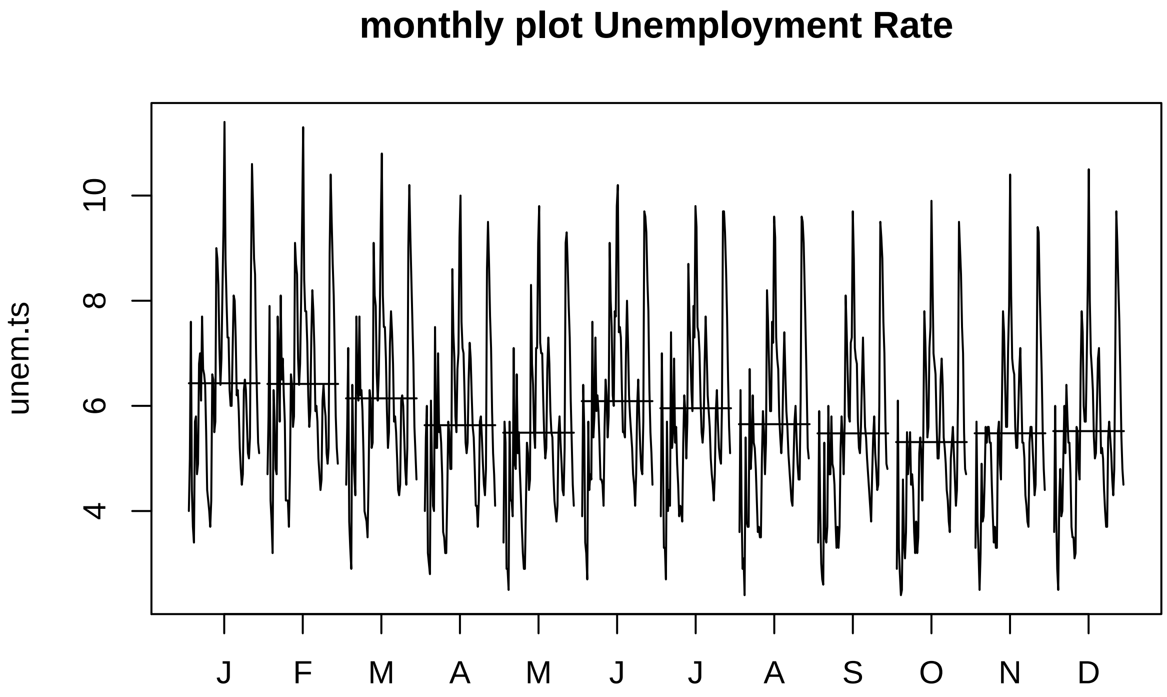


The trend component observed in the decomposed series concur with our intuitions from the smoothed series. With the seasonal component taken out, the random component show a spike in mid-1980s and several more in 2000s, both of which may correspond to respective oil price fluctuations. In general, the random component series shows higher variance before mid-1980s than after, which we have also observed in the unemployment series. The decomposed components are clearly not stationary, which against suggest that OLS is not appropriate.

Establish Stationarity

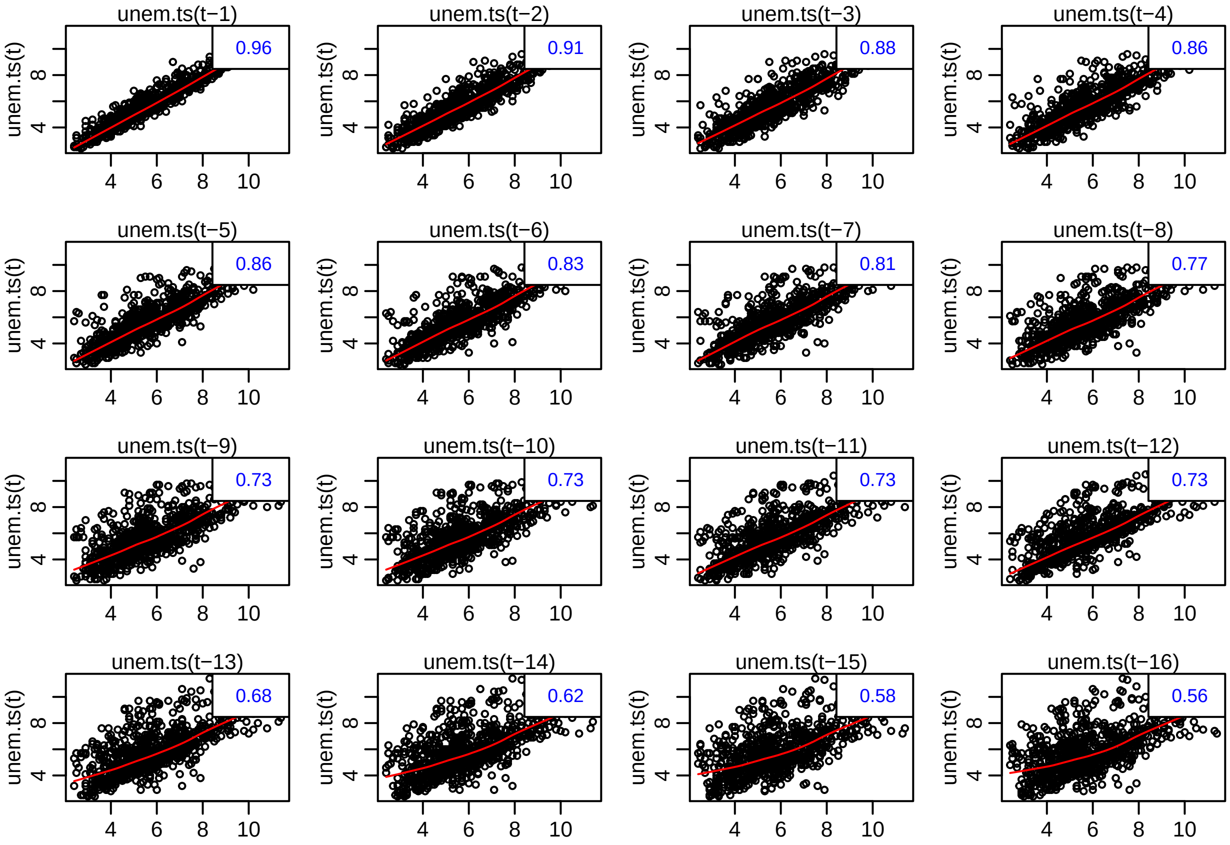
Seasonality and Unit Root Investigation – unemployment rate

```r
monthplot(unem.ts);title("monthly plot Unemployment Rate")
```



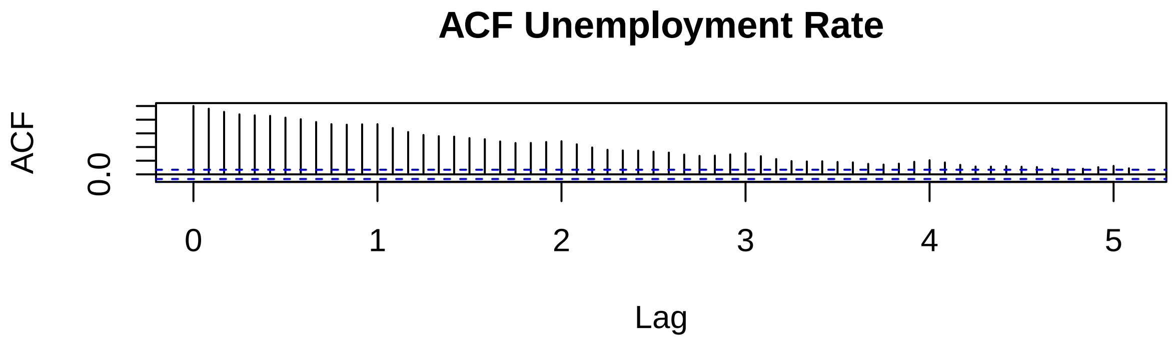
The monthly plot show some seasonal pattern at the mean, but the range of unemployment variation for each month also overlap a lot with other months. So the seasonal pattern exists but is not very strong. We see that unemployment rate tend to be a little higher in the beginning and middle of each year followed by mild gradual decrease.

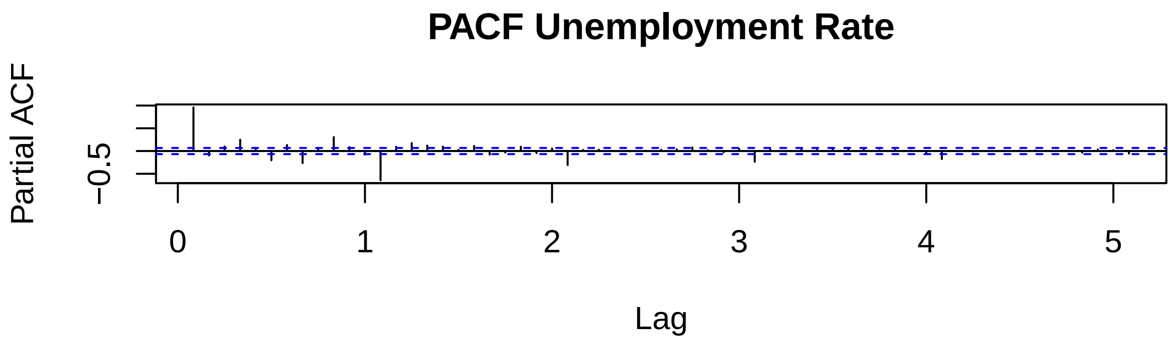
```
astsa::lag1.plot(unem.ts,16)
```



The above scatterplot shows that the series is most correlated with its first lag, then correlations grow weaker and the point clouds becomes more scattered with larger lags without any sign of picking up at lag 12. This plot itself doesn't suggest seasonal effects.

```r
par(mfrow = c(2,1))
acf(unem.ts, lag.max = 61, main = ""); title("ACF Unemployment Rate")
pacf(unem.ts, lag.max = 61, main = ""); title("PACF Unemployment Rate")
```





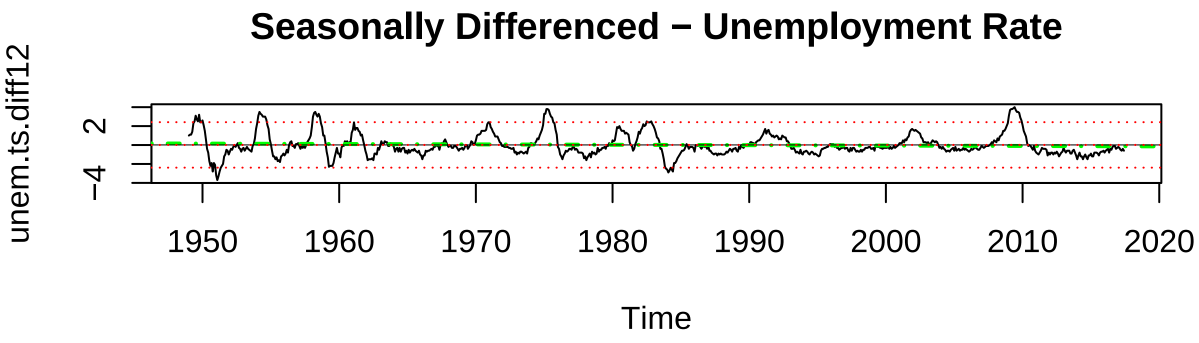
The acf shows slow decay from lag 0 coupled with a sharp drop of pacf after lag 1. This is a sign of an AR(1) process. We see minor local maxima on the acf at lag 12, 24, 36 and 48 which indicate seasonal effects. The significant, negative pacfs which tails off at lag 13, 25, 37, 49 suggest seasonal AR or MA components. Some smaller significant pacf values within the first 12 lags suggest either some AR processes or an MA process in that range.

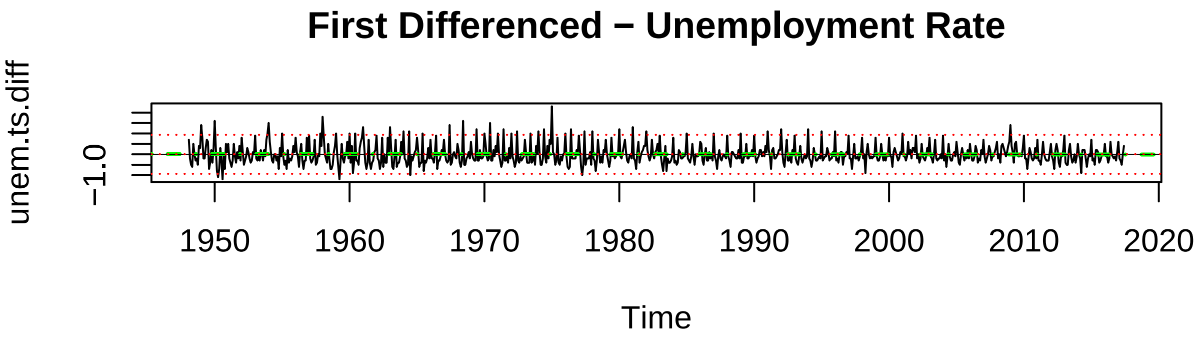
Although there are some seasonal effects in the unemployment series, they don't seem strong enough to require seasonal differencing. To help us determine, we compare the seasonal differenced and first differenced series and compare their behaviors using time and autocorrelation plots.

```r
unem.ts.diff12 = diff(unem.ts, lag = 12)
unem.ts.diff = diff(unem.ts, lag = 1)
```

```r
par(mfrow = c(2,1))

ts.plot(unem.ts.diff12)
abline(lm(unem.ts.diff12 ~ time(unem.ts.diff12)),
       col = "green", lty = "dotdash", lwd = 2)
abline(h = mean(unem.ts.diff12), lwd = 0.5)
abline(h = c(mean(unem.ts.diff12), mean(unem.ts.diff12) +
               2 * sd(unem.ts.diff12)), col = "red",
       lwd = 1, lty = "dotted")
abline(h = c(mean(unem.ts.diff12), mean(unem.ts.diff12) -
               2 * sd(unem.ts.diff12)), col = "red",
       lwd = 1, lty = "dotted")
title("Seasonally Differenced - Unemployment Rate")
```

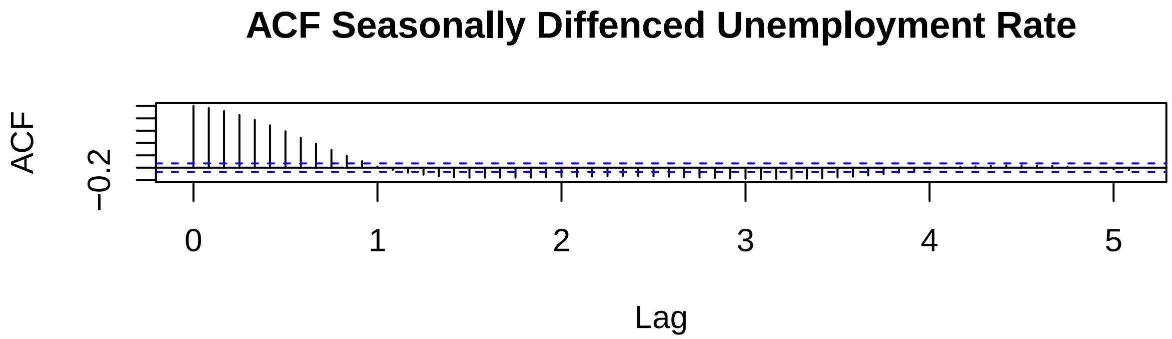
```
ts.plot(unem.ts.diff)
abline(lm(unem.ts.diff ~ time(unem.ts.diff)),
       col = "green", lty = "dotdash", lwd = 2)
abline(h = mean(unem.ts.diff), lwd = 0.5)
abline(h = c(mean(unem.ts.diff), mean(unem.ts.diff) +
               2 * sd(unem.ts.diff)), col = "red",
       lwd = 1, lty = "dotted")
abline(h = c(mean(unem.ts.diff), mean(unem.ts.diff) -
               2 * sd(unem.ts.diff)), col = "red",
       lwd = 1, lty = "dotted")
title("First Differenced - Unemployment Rate")
```

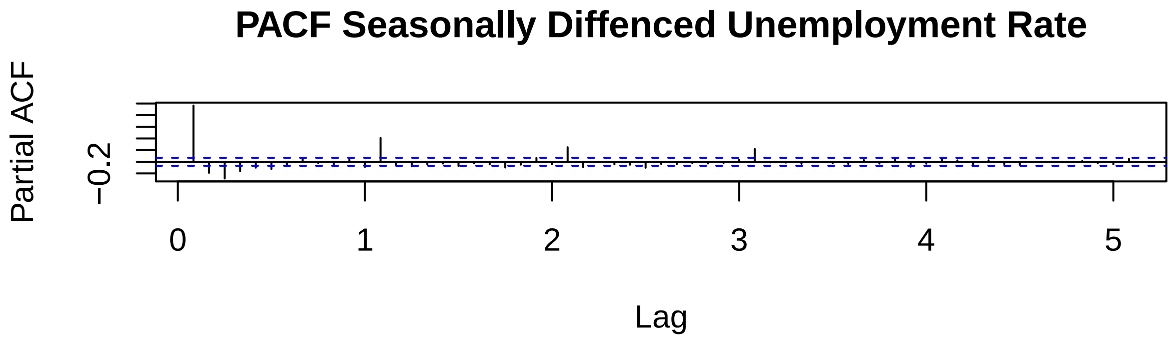




The seasonally differenced series retained most of the random walk like behavior in the raw series and appear more persistent before mid 1980s. The first differenced series eliminated most random walk and persistent behaviors but variance is generally larger before 1980. There is still a slight downward trend (green regression line) with the seasonal differenced series. (red line refers to 2 standard deviation marks)

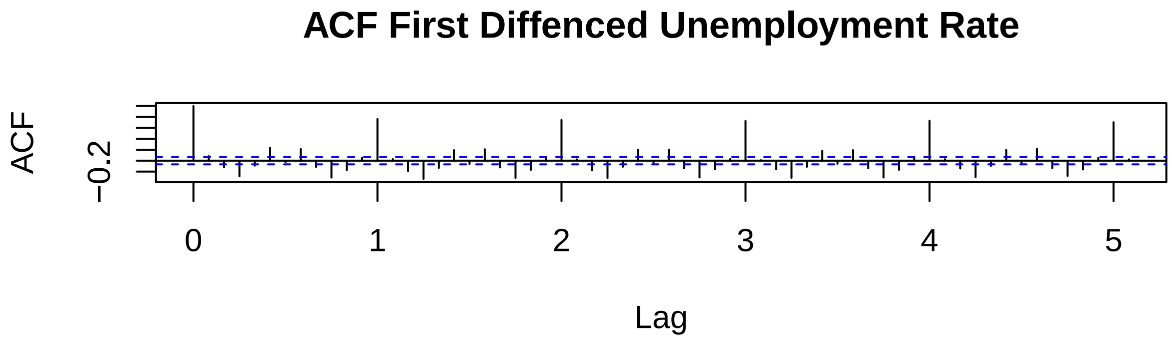
```
par(mfrow = c(2,1))
acf(unem.ts.diff12, lag.max = 61, main = "")
title("ACF Seasonally Diffenced Unemployment Rate")
pacf(unem.ts.diff12, lag.max = 61, main = "")
title("PACF Seasonally Diffenced Unemployment Rate")
```

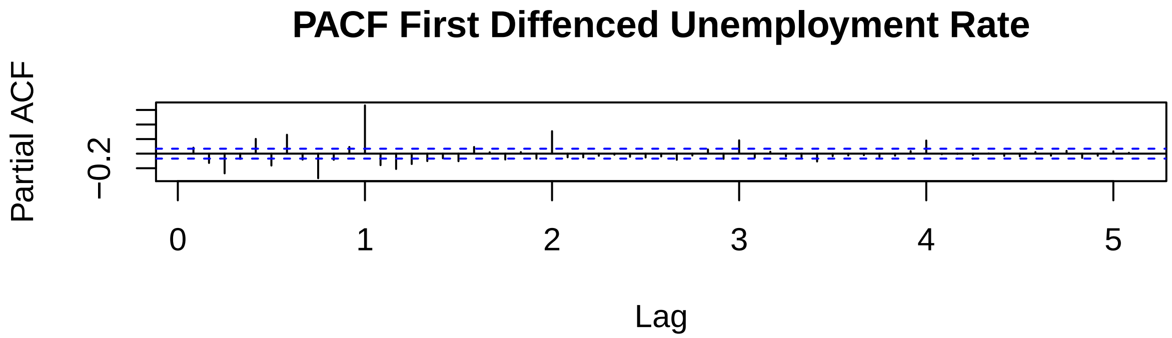




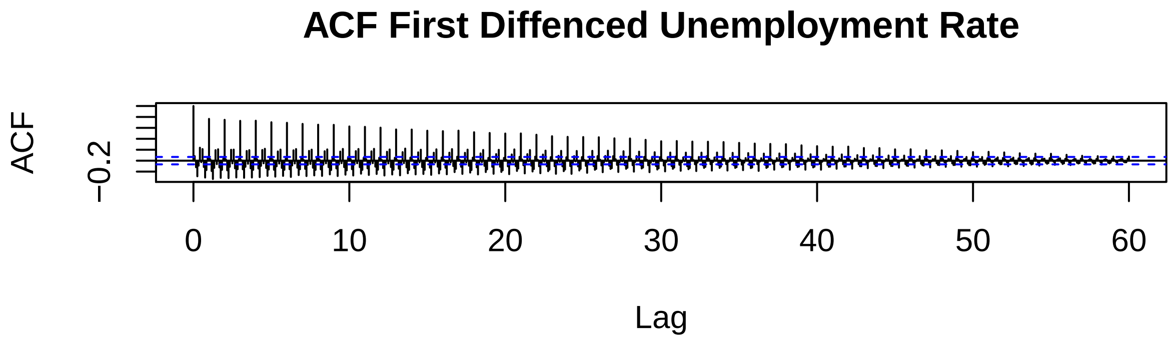
Above plots: In the seasonal differenced series, we still see a significant pacf at lag 1 followed by a sharp drop and gradual decay in the acf. Notice the significant pacfs in lags 13, 25, 37 as well. This suggests an AR(1) process in combination with seasonal AR(3), or in combination with some seasonal ARMA processes. Notice a couple significant pacfs between lag 2 to 4, they can be AR(2-4) components but hard to tell at this stage. We may entertain the model is SARIMA(4,0,q)(3,1,Q) from these plots.

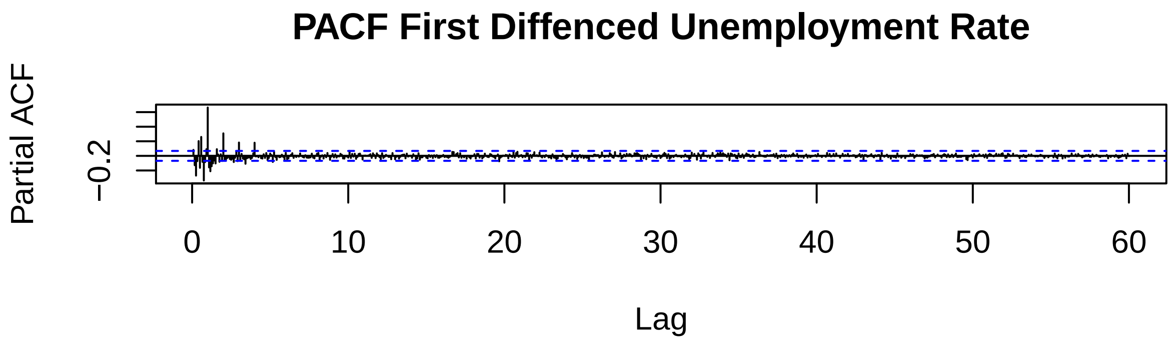
```r
par(mfrow = c(2,1))
acf(unem.ts.diff, lag.max = 61, main = "")
title("ACF First Diffenced Unemployment Rate")
pacf(unem.ts.diff, lag.max = 61, main = "")
title("PACF First Diffenced Unemployment Rate")
```





```
par(mfrow = c(2,1))
acf(unem.ts.diff, lag.max = 720, main = "")
title("ACF First Diffenced Unemployment Rate")
pacf(unem.ts.diff, lag.max = 720, main = "")
title("PACF First Diffenced Unemployment Rate")
```





Above plots: In the first differenced series, the plots some periodic behavior, the extended plot show that acf tails off until lag 720, which suggest either a seasonal long memory or seasonal integrated process (both SAR(1)) in combination with some sesonal AR(2-4) components or in combination with a seasonal MA component. The pacf show some significance between lag 2-9 without a little echo in the acf lag 13-16, which can come from some MA processes but it's hard to say at this stage. We simply assume some ARMA processes present. We may entertain the model as a SARIMA(9,1,q)(4,0,Q) using these plots.

Performing both differencing procedures may over-difference the series. To determine how much is enough, we perform unit root tests below to check for stationarity. Augmented Dicky Fuller Test and Phillips Perron Tests are performed, with the following test hypotheses:

- Ho : The series has a unit root
- Ha : The series is stationary

In the ADF test, our null hypothesis assume that the process is a random walk with drift and some AR(p) components,$x_t = \beta_0 + \phi x_{t-1} + \sum_{j=1}^{p-1} \psi_j x_{t-j} + w_t$, where β_0 represents the drift and $\phi = 1$. We are essentially testing the null hypothesis $\gamma = 0$ in the differenced series $\nabla x_t = \gamma x_{t-1} + \sum_{j=1}^{p-1} \psi_j \nabla x_{t-j} + w_t$ since $\gamma = \phi - 1$. Under the alternative hypothesis $\gamma < 0$. On the other hand, the Philips Perron test, our null hypothesis assume a model $x_t = \beta_0 + \rho_1 x_{t-1} + u_t$, where non-parametric correction is applied on ρ to correct for serial correlation in u_t already. We are essential testing of $\rho = 1$ in the null hypothesis.

```
tseries::adf.test(unem.ts)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  unem.ts
## Dickey-Fuller = -2.5911, Lag order = 9, p-value = 0.3281
## alternative hypothesis: stationary
```

```r
tseries::pp.test(unem.ts)
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  unem.ts
## Dickey-Fuller Z(alpha) = -28.648, Truncation lag parameter = 6,
## p-value = 0.01093
## alternative hypothesis: stationary
```

```r
tseries::adf.test(unem.ts.diff)
```

```
## Warning in tseries::adf.test(unem.ts.diff): p-value smaller than printed p-
## value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  unem.ts.diff
## Dickey-Fuller = -13.13, Lag order = 9, p-value = 0.01
## alternative hypothesis: stationary
```

```r
tseries::pp.test(unem.ts.diff)
```

```
## Warning in tseries::pp.test(unem.ts.diff): p-value smaller than printed p-
## value
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  unem.ts.diff
## Dickey-Fuller Z(alpha) = -602.1, Truncation lag parameter = 6,
## p-value = 0.01
## alternative hypothesis: stationary
```

```r
tseries::adf.test(unem.ts.diff12)
```

```
## Warning in tseries::adf.test(unem.ts.diff12): p-value smaller than printed
## p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  unem.ts.diff12
## Dickey-Fuller = -8.1444, Lag order = 9, p-value = 0.01
## alternative hypothesis: stationary
```

```r
tseries::pp.test(unem.ts.diff12)
```

```
## Warning in tseries::pp.test(unem.ts.diff12): p-value smaller than printed
## p-value
```

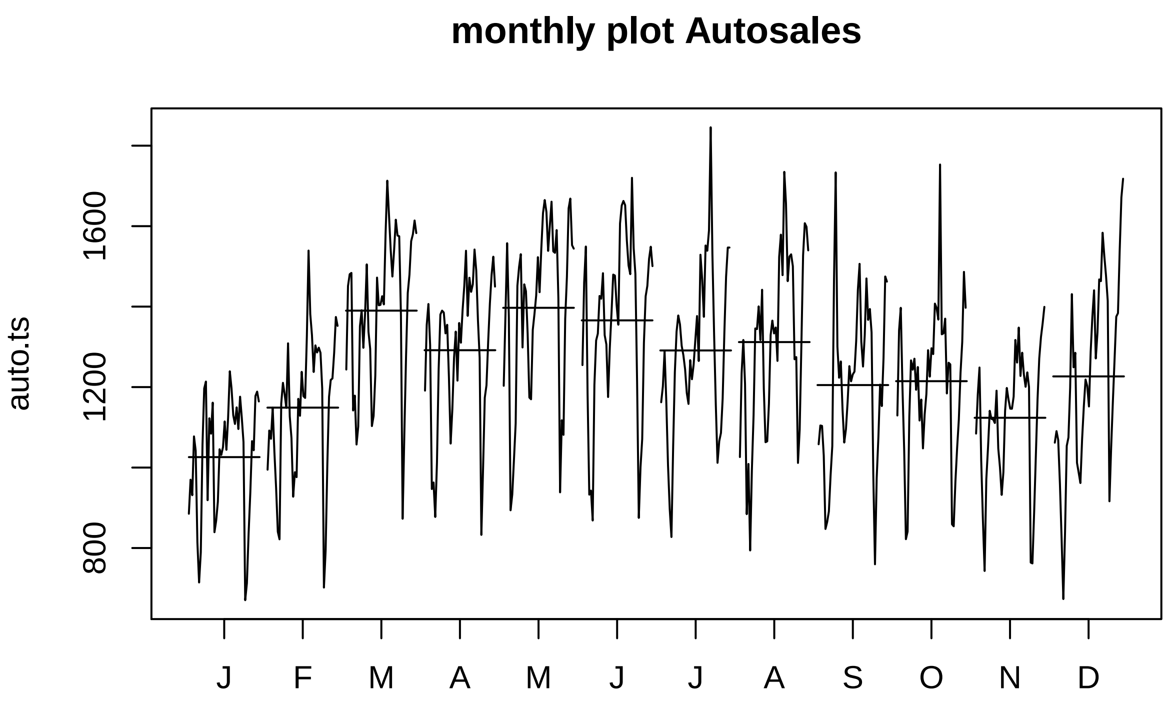
```
##
##  Phillips-Perron Unit Root Test
##
## data:  unem.ts.diff12
## Dickey-Fuller Z(alpha) = -60.968, Truncation lag parameter = 6,
## p-value = 0.01
## alternative hypothesis: stationary
```

As expected with the raw series, the ADF test failed to reject the null hypothesis that the series contains a unit root. The PP test results contradicts but we acknowledge that the test mechanism is different and that some early literature (Davidson and Mackinnon 2004) showed that ADF performs better in finite sample than PP test. All the tests rejected the null hypotheses that either the first differenced or seasonal differenced series contains a unit root and supports stationarities. We should not need to difference the raw series with lag 12 in addition to lag 1, or vice versa.

Unemployment series stationarity conclusion: The month plot, scatterplot matrix and autocorrelation plots suggest some but weak seasonality of our raw series, along with strong random walk behavior. The time plots, acf and pacf plots of the seasonal differenced series strongly suggest AR(1) process and that of the first differenced series strongly suggest seasonal AR(1) process. The unit root test suggested that we don't need to do both differencing. We may give slightly higher preference to a first differenced series with a long memory seasonal AR(1) component, but should still consider a seasonal differenced series as a candidate.

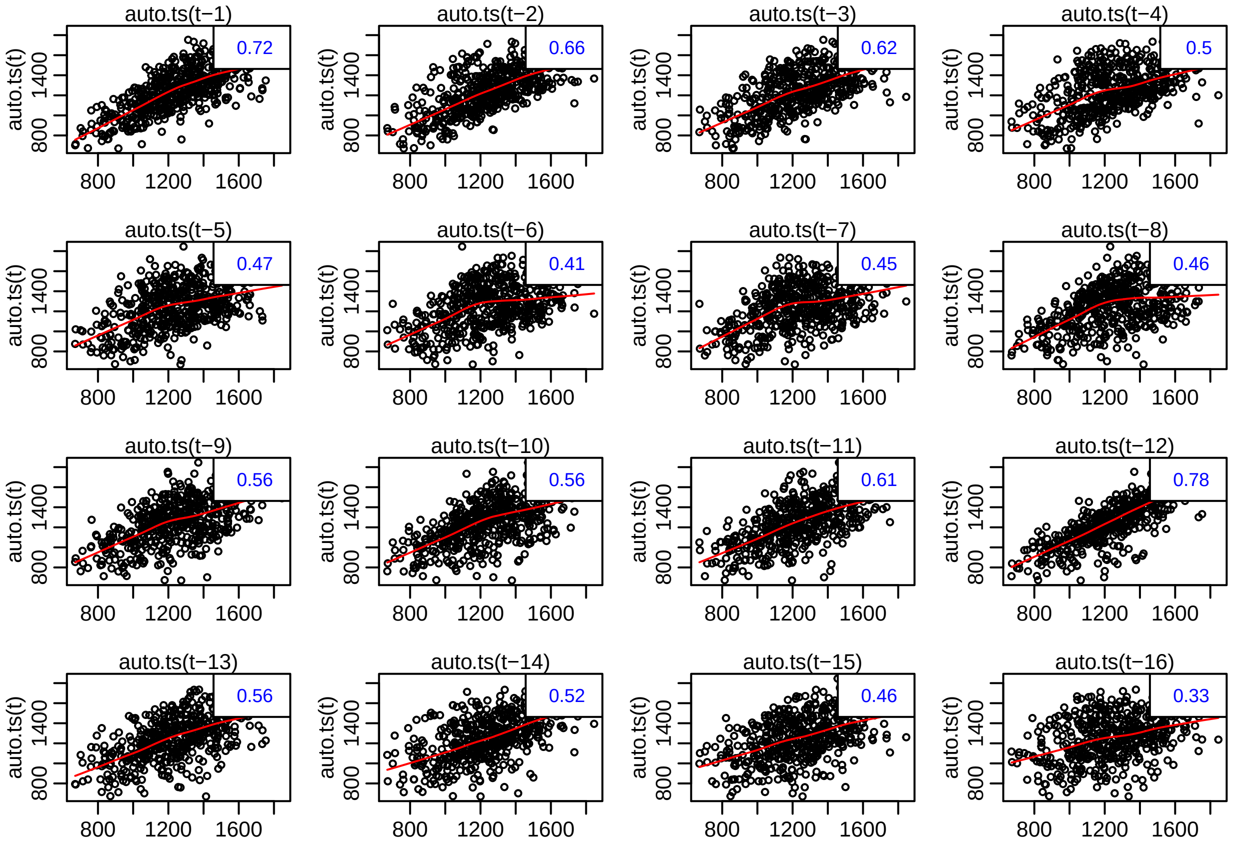
Seasonality and Unit Root Investigation – unemployment rate

```
monthplot(auto.ts);title("monthly plot Autosales")
```



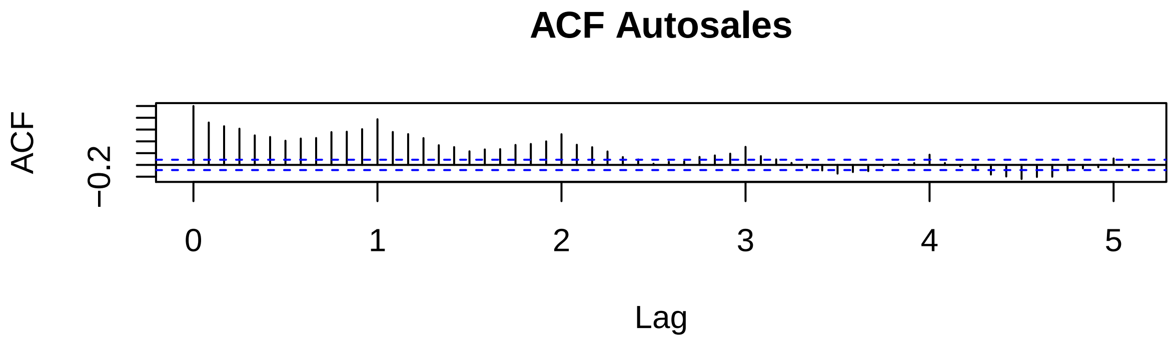
Compared to that of unemployment rate, monthly plot of auto sales show much more discrete values in the mean. So the seasonal pattern is much stronger. We see that autosales tend to be noticeably lower in the first two months, picks up sharply in March, steps up and down for the rest of the year. Fall sales are generally lower than summer sales.

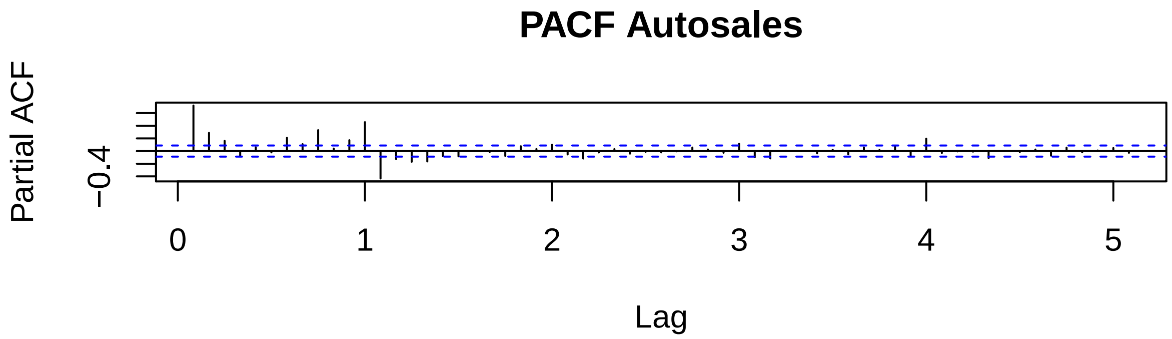
```
astsa::lag1.plot(auto.ts,16)
```



Scatterplot matrix of auto sales with its own lags display clear seasonal dependence. Autocorrelation is highest with lag 12, that is this month last year, even more so compared to lag 1. The opposite patterns was observed in the unemployment rates series, when autocorrelation is highest with lag 1 and then gradually declines towards higher lags.

```r
par(mfrow = c(2,1))
acf(auto.ts, lag.max = 61, main = ""); title("ACF Autosales")
pacf(auto.ts, lag.max = 61, main = ""); title("PACF Autosales")
```





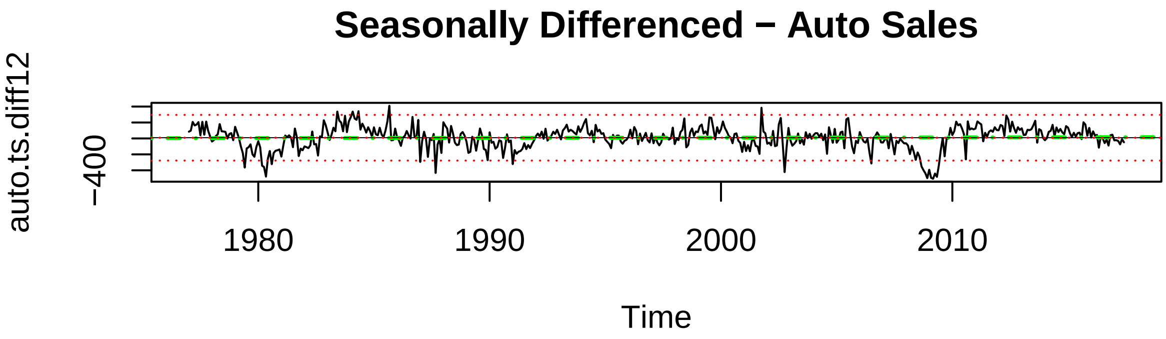
Similar to the unemployment series, the acf shows slow decay from lag 0, a sharp drop of pacf after lag 1 and some local acf maximums at lag 12, 24, 36 and 48. Unlike the unemployment series, the acf decay ends earlier at lag 36. The local maxima have much stronger profiles. Some significant pacf values occur before lag 12 which suggest either some AR or MA components(effects on acf plot is not discernable) in that range.

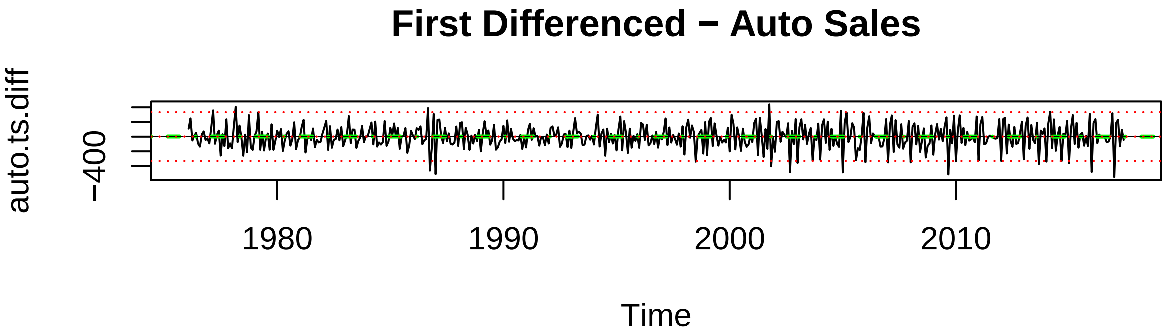
Based strong seasonal pattern in the above three plots, seasonal differencing would be useful in achieving stationarity. To help us determine, we again compare the seasonal differenced and first differenced series and compare their behaviors using time and autocorrelation plots.

```
auto.ts.diff12 = diff(auto.ts, lag = 12)
auto.ts.diff = diff(auto.ts, lag = 1)
```

```
par(mfrow = c(2,1))

ts.plot(auto.ts.diff12)
abline(lm(auto.ts.diff12 ~ time(auto.ts.diff12)),
       col = "green", lty = "dotdash", lwd = 2)
abline(h = mean(auto.ts.diff12), lwd = 0.5)
abline(h = c(mean(auto.ts.diff12), mean(auto.ts.diff12) +
               2 * sd(auto.ts.diff12)), col = "red",
       lwd = 1, lty = "dotted")
abline(h = c(mean(auto.ts.diff12), mean(auto.ts.diff12) -
               2 * sd(auto.ts.diff12)), col = "red",
       lwd = 1, lty = "dotted")
title("Seasonally Differenced - Auto Sales")

ts.plot(auto.ts.diff)
```

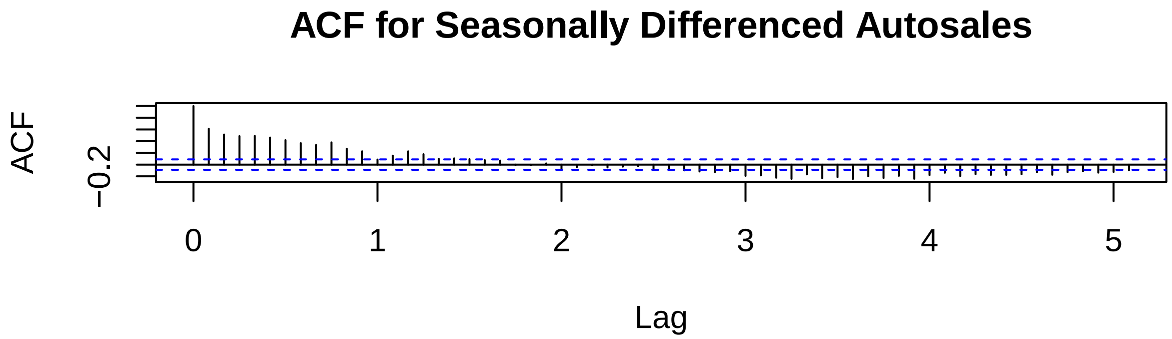
```
abline(lm(auto.ts.diff ~ time(auto.ts.diff)),
       col = "green", lty = "dotdash", lwd = 2)
abline(h = mean(auto.ts.diff), lwd = 0.5)
abline(h = c(mean(auto.ts.diff), mean(auto.ts.diff) +
               2 * sd(auto.ts.diff)), col = "red",
       lwd = 1, lty = "dotted")
abline(h = c(mean(auto.ts.diff), mean(auto.ts.diff) -
               2 * sd(auto.ts.diff)), col = "red",
       lwd = 1, lty = "dotted")
title("First Differenced - Auto Sales")
```

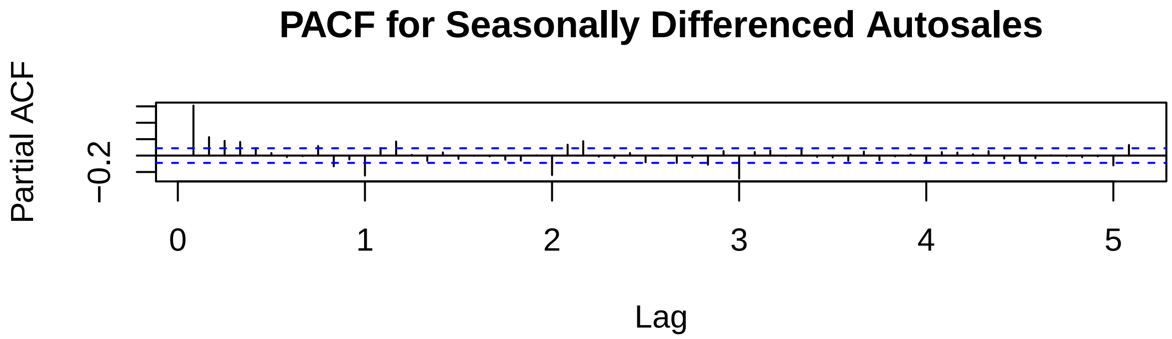




The seasonally differenced series noticeably removed the strong seasonal patterns and some degree of persistency from the raw series. The first differenced series removed most persistencies from the raw series but appear clustered regularly at seasonally intervals. Both series removed the upward trend in the raw series effectively.

```
par(mfrow = c(2,1))
acf(auto.ts.diff12, lag.max = 61, main = "")
title("ACF for Seasonally Differenced Autosales")
pacf(auto.ts.diff12, lag.max = 61, main = "")
title("PACF for Seasonally Differenced Autosales")
```





Above plots: Similar to unmployment rate, in the seasonal differenced serie of auto sales, we still see a significant pacf at lag 1 followed by a sharp drop and gradual decay in the acf. Notice the significant pacfs in lag 12, 24 and 36. This suggests an AR(1) process in combination with seasonal AR(3) or some seasonal MA component. Also notice a few significant pacfs at lags 2-4. We may entertain the model as a SARIMA(4,0,q)(3,1,Q) using these plots. It doesn't seem like an additional first differencing is necessary.

We perform unit root tests below to check for stationarities if the raw, firsr differenced and seasonal differenced series for auto sales. Augmented Dicky Fuller Test and Phillips Perron Tests are performed, with the following test hypotheses:

- Ho : The series has a unit root
- Ha : The series is stationary

```
tseries::adf.test(auto.ts)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  auto.ts
## Dickey-Fuller = -3.5662, Lag order = 7, p-value = 0.03595
## alternative hypothesis: stationary
```

```
tseries::pp.test(auto.ts)
```

```
## Warning in tseries::pp.test(auto.ts): p-value smaller than printed p-value
```

```
##
##  Phillips-Perron Unit Root Test
##
```

```
## data:  auto.ts
## Dickey-Fuller Z(alpha) = -157.6, Truncation lag parameter = 5,
## p-value = 0.01
## alternative hypothesis: stationary
```

```
tseries::adf.test(auto.ts.diff)
```

```
## Warning in tseries::adf.test(auto.ts.diff): p-value smaller than printed p-
## value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  auto.ts.diff
## Dickey-Fuller = -16.651, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary
```

```
tseries::pp.test(auto.ts.diff)
```

```
## Warning in tseries::pp.test(auto.ts.diff): p-value smaller than printed p-
## value
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  auto.ts.diff
## Dickey-Fuller Z(alpha) = -611.4, Truncation lag parameter = 5,
## p-value = 0.01
## alternative hypothesis: stationary
```

```
tseries::adf.test(auto.ts.diff12)
```

```
## Warning in tseries::adf.test(auto.ts.diff12): p-value smaller than printed
## p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  auto.ts.diff12
## Dickey-Fuller = -4.1696, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary
```

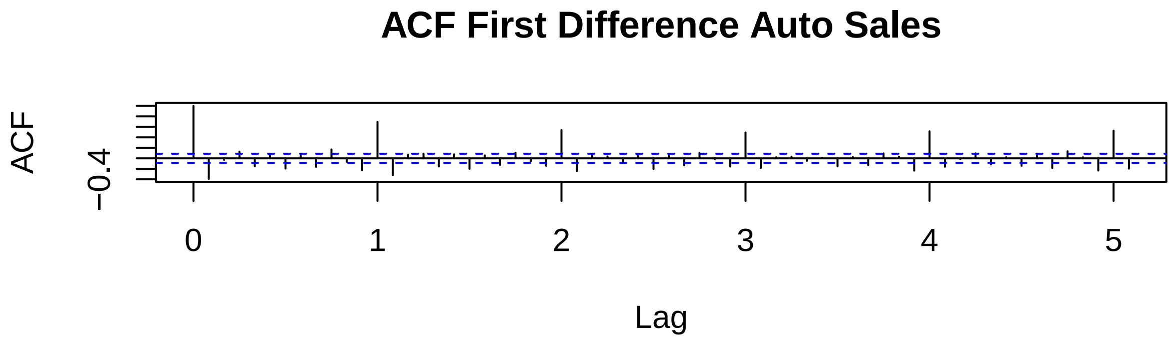
```
tseries::pp.test(auto.ts.diff12)
```

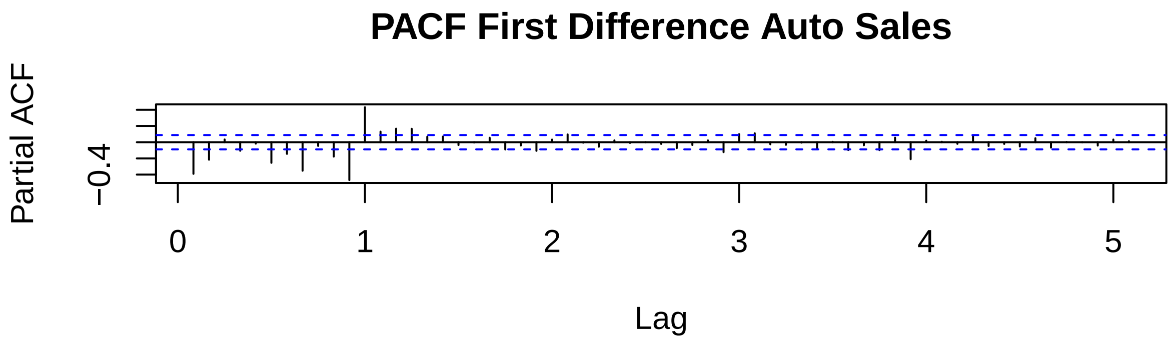
```
## Warning in tseries::pp.test(auto.ts.diff12): p-value smaller than printed
## p-value
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  auto.ts.diff12
## Dickey-Fuller Z(alpha) = -199.97, Truncation lag parameter = 5,
## p-value = 0.01
## alternative hypothesis: stationary
```

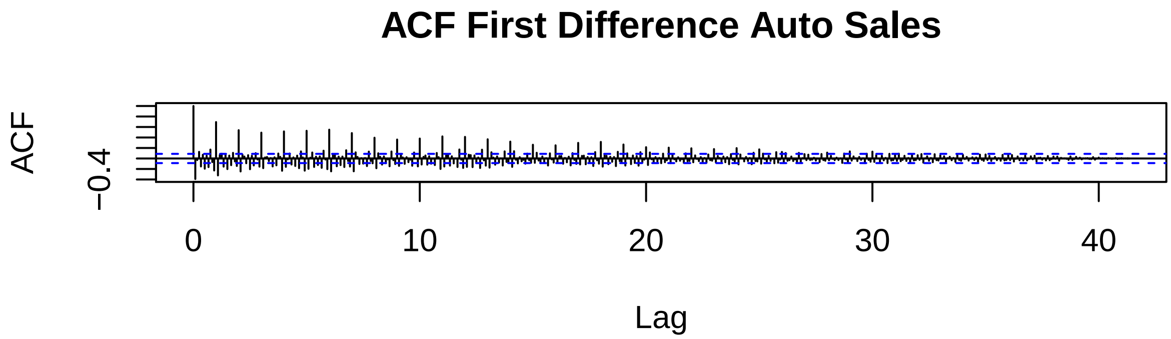
All tests rejected the null hypotheses for all three series. Notice that the ADF test p-value for the raw series is closer to the critical cut off of 0.05. There can be a very weak chance that the raw series contains a unit root. Again we should not apply first or both differencing procedures to the raw series to avoid over-differencing. Below, we examine the acf and pacf for the first differenced series to confirm our intuition.

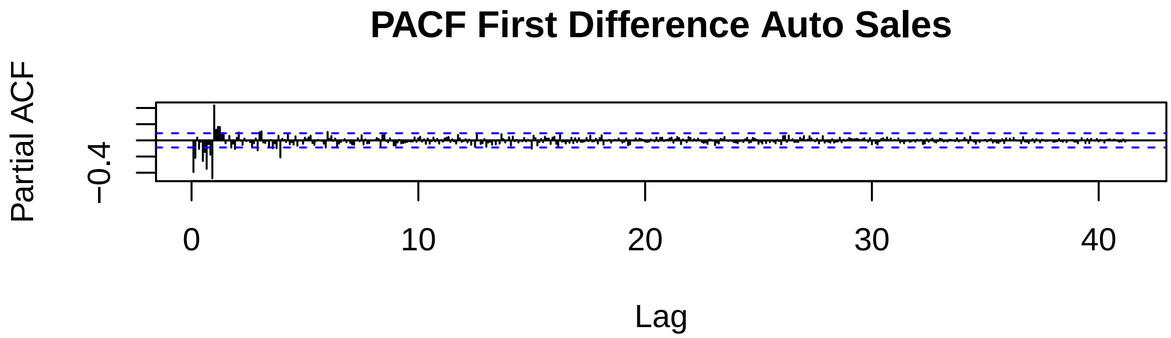
```r
par(mfrow = c(2,1))
acf(auto.ts.diff, lag.max = 61, main = "")
title("ACF First Difference Auto Sales")
pacf(auto.ts.diff, lag.max = 61, main = "")
title("PACF First Difference Auto Sales")
```





```r
par(mfrow = c(2,1))
acf(auto.ts.diff, lag.max = 720, main = "")
title("ACF First Difference Auto Sales")
pacf(auto.ts.diff, lag.max = 720, main = "")
title("PACF First Difference Auto Sales")
```





Above plots: Similar to unmployment rate, in the first differenced serie of auto sales, the acf tails off very slowly until lag 360. Unlike that of the unemployment series, the seasonal ripples on the acf appear much stronger. This plots suggest a strong seasonal AR(1) component and some AR(p) components before lag 12.

Auto sales series stationarity conclusion: The month plot, scatterplot matrix and autocorrelation plots suggest strong seasonality of our raw series. The time plots, acf and pacf plots of the seasonal differenced series suggest an AR(1) component and that of the first differenced series suggests a strong seasonal AR(1) process. The unit root tests suggest that we don't need to do first or both differencing. We may give higher preference to a seasonal differenced series with an AR(1) component, but may still entertain a first differenced series as a candidate.

Examine Bivariate Relationship

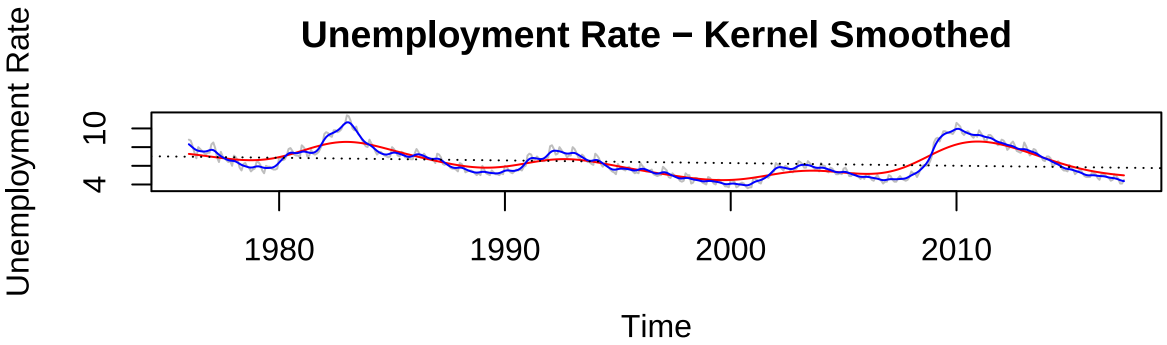
Overview and cross-correlation

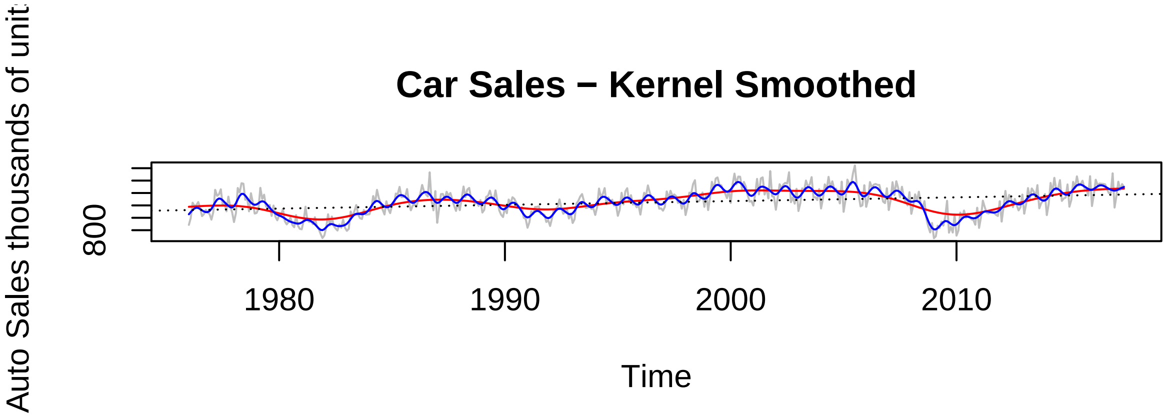
```
# Intersect the series
combined.raw = ts.intersect(unem.ts, auto.ts)
unem.intersect.ts = combined.raw[,1]
auto.intersect.ts = combined.raw[,2]

head(combined.raw);tail(combined.raw)
```

```
##      unem.ts auto.ts
## [1,]     8.8   885.2
## [2,]     8.7   994.7
## [3,]     8.1  1243.6
## [4,]     7.4  1191.2
## [5,]     6.8  1203.2
## [6,]     8.0  1254.7
```

```
##         unem.ts auto.ts
## [493,]     5.1  1164.3
## [494,]     4.9  1352.1
## [495,]     4.6  1582.7
## [496,]     4.1  1449.7
## [497,]     4.1  1544.1
## [498,]     4.5  1500.6
```

```r
par(mfrow = c(2,1))

# Kernel smoothing
unem.k.smooth.widest = ksmooth(time(unem.intersect.ts),
                             unem.intersect.ts, kernel = c("normal"),
                             bandwidth = 10)

unem.k.smooth.wide = ksmooth(time(unem.intersect.ts),
                             unem.intersect.ts, kernel = c("normal"),
                             bandwidth = 4)

unem.k.smooth.narrow = ksmooth(time(unem.intersect.ts),
                               unem.intersect.ts, kernel = c("normal"),
                               bandwidth = 0.5)

# Make plot
plot(unem.intersect.ts, col = "gray", ylab = "Unemployment Rate",
     main = "Unemployment Rate - Kernel Smoothed")
lines(unem.k.smooth.wide$x, unem.k.smooth.wide$y, col = "red")
lines(unem.k.smooth.narrow$x, unem.k.smooth.narrow$y, col = "blue")
abline(lm(unem.intersect.ts~time(unem.intersect.ts)), lty = "dotted", col = "black")

# Kernel smoothing
auto.k.smooth.widest = ksmooth(time(auto.ts),
                             auto.ts, kernel = c("normal"),
                             bandwidth = 10)

auto.k.smooth.wide = ksmooth(time(auto.ts),
                             auto.ts, kernel = c("normal"),
                             bandwidth = 4)

auto.k.smooth.narrow = ksmooth(time(auto.ts),
                               auto.ts, kernel = c("normal"),
                               bandwidth = 0.5)

# Make plot
plot(auto.ts, col = "gray", ylab = "Auto Sales thousands of units",
     main = "Car Sales - Kernel Smoothed")
lines(auto.k.smooth.wide$x, auto.k.smooth.wide$y, col = "red")
lines(auto.k.smooth.narrow$x, auto.k.smooth.narrow$y, col = "blue")
abline(lm(auto.ts~time(auto.ts)), lty = "dotted", col = "black")
```

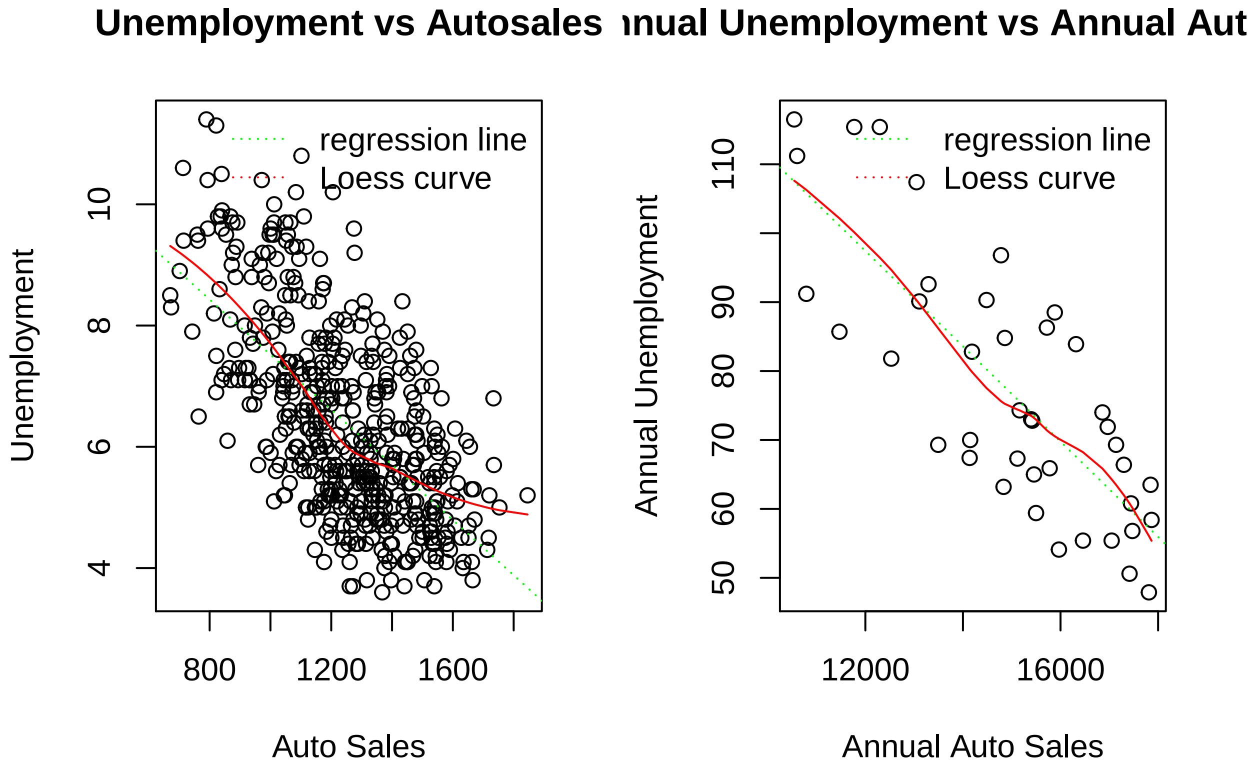




It appears that between 1976 to early 1990s and between late 2000s and 2017, a crest in auto sales would be echoed with a trough in unemployment rate a few months later and a trough in auto sales would be echoed by a crest in unemployment rate a few months later. The unemployment series has a slight downward trend over the whole time interval while the opposite is true for auto sales.

```
# Function for scatterplot with Loess Curve and regression curve
scatter.loess.lm.plot = function(y, x, xlab, ylab, title){

  plot(x = x, y = y, xlab = xlab, ylab = ylab)
  title(title)

  abline(lm(y~x), col = "green", lty = "dotted")

  order.pred = order(x)
  smooth.stand = loess(formula = y~x,
                       weights = rep(1,length(x)))
  lines(x = x[order.pred],
        y = predict(smooth.stand)[order.pred],
        lty = "solid", col = "red")

  legend("topright",
         legend = c("regression line", "Loess curve"),
         col = c("green", "red"),
         lty = "dotted", "solid", bty = "n")
}
```

```
par(mfrow = c(1,2))

scatter.loess.lm.plot(y = unem.intersect.ts,
                      x = auto.intersect.ts,
                      xlab = "Auto Sales",
                      ylab = "Unemployment",
                      title = "Unemployment vs Autosales")

scatter.loess.lm.plot(y = as.vector(aggregate(unem.intersect.ts)),
                      x = as.vector(aggregate(auto.intersect.ts)),
                      xlab = "Annual Auto Sales",
                      ylab = "Annual Unemployment",
                      title = "Annual Unemployment vs Annual Autosales")
```



```
cat("Corr(Unemployment, Autosales):", cor(auto.intersect.ts,unem.intersect.ts))
```

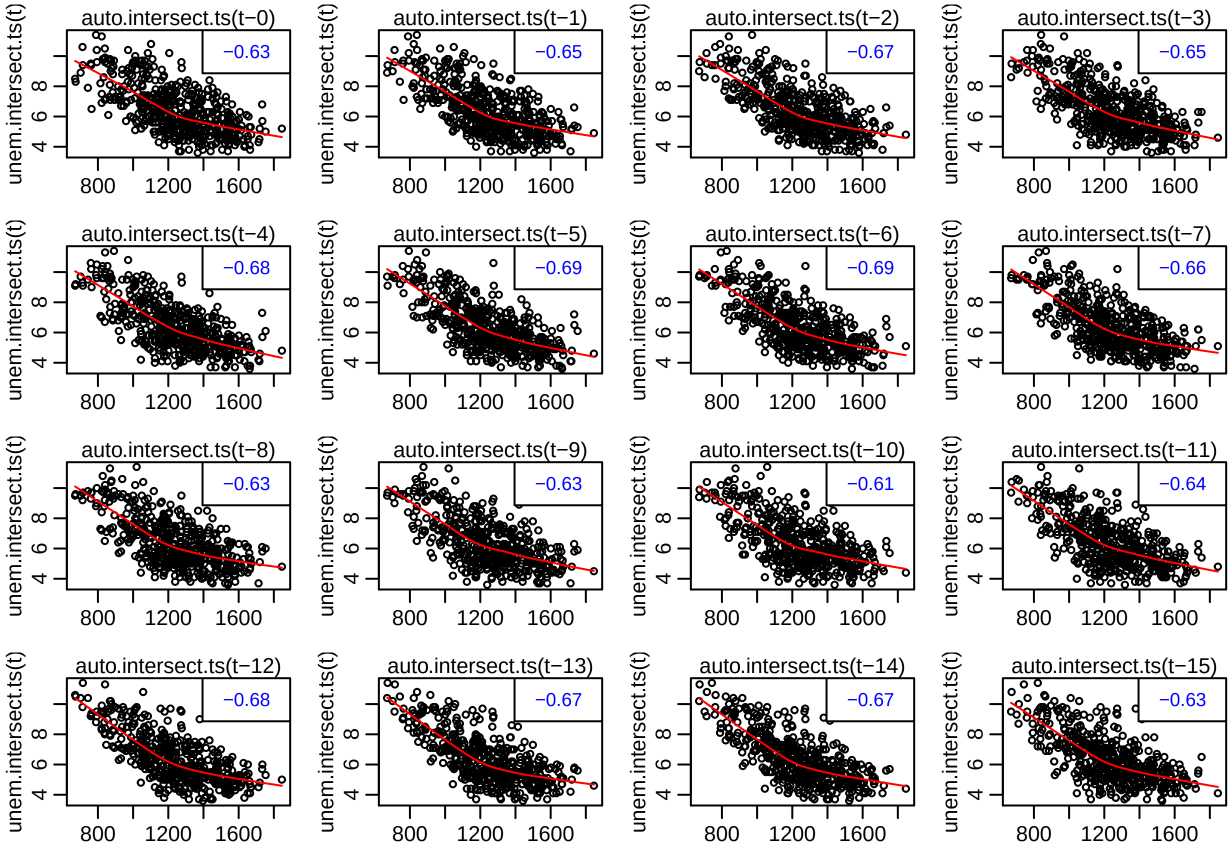
```
## Corr(Unemployment, Autosales): -0.6342763
```

```
cat("Corr(Annual Unemployment, Annual Autosales):", cor(aggregate(auto.intersect.ts),aggregate(unem.inte
```

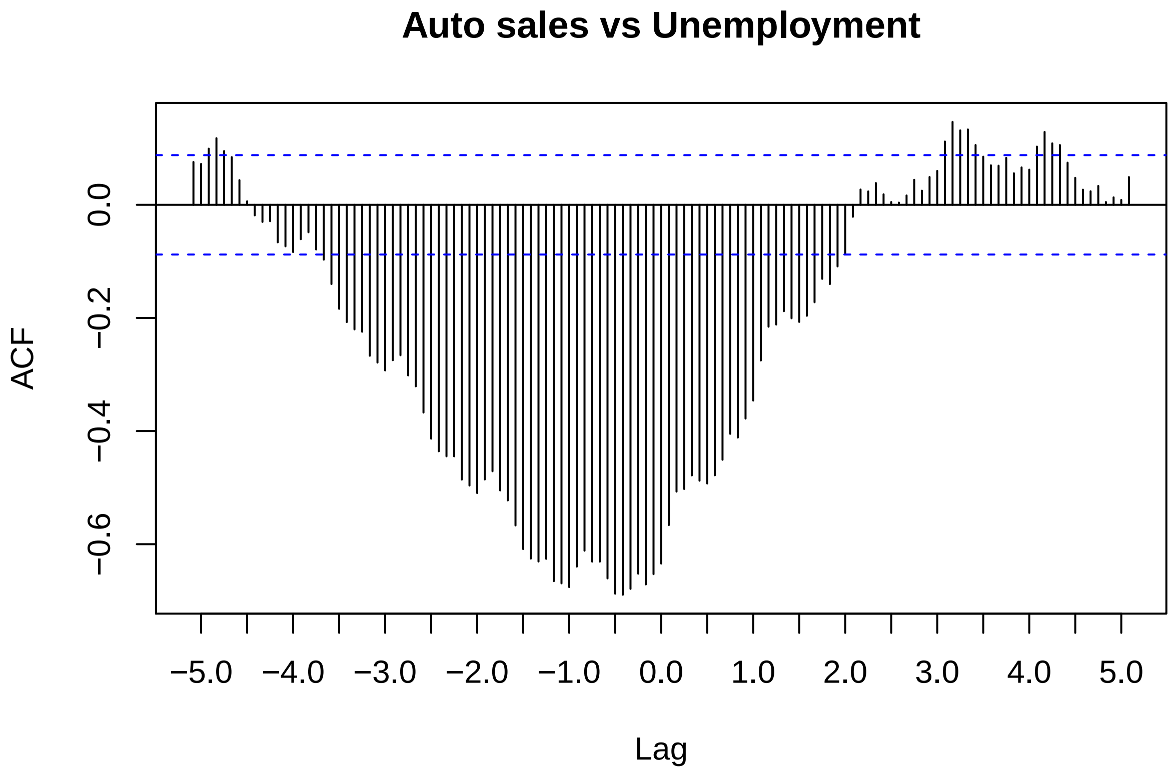
```
## Corr(Annual Unemployment, Annual Autosales): -0.7891484
```

Disregarding time-dependent variations, the two series show some non-linear relationship in addition to overall negative correlation. The annually aggregated series show stronger and more linear correlation. It seems that elimination of seasonal granularity may “hide” otherwise non-linear relationships, which can be depicted by the scatterplot matrix below. Correlation of unemployment series against auto sales series remains moderate for more than 12 lags, and it peaks at lag 5 and 6.

```r
astsa::lag2.plot(auto.intersect.ts, unem.intersect.ts,15)
```



```r
#par(mfrow = c(3,1))
#acf( unem.intersect.ts, lag.max = 61, main = "")
#title("Unemployment")
#acf(auto.intersect.ts, lag.max = 61, main = "")
#title("Auto Sales")
ccf(auto.intersect.ts, unem.intersect.ts, main = "",
    lag.max = 61, xaxt = "n")
axis(side = 1, at = seq(-5,5,0.5))
title("Auto sales vs Unemployment")
```



The ccf plot above assess the cross-correlation $\hat{\rho}_{xy}(h) = \frac{\hat{\gamma}_{xy}(h)}{\sqrt{\hat{\gamma}_x(0)\hat{\gamma}_y(0)}}$ based on the cross-covariance function $\hat{\gamma}_{xy}(h) = n^{-1}\sum_{t=1}^{n-h}(x_{t+h} - \bar{x})(y_t - \bar{y})$, where x_t refers to the auto sales series, y_t refers to the unemployment series, h refers to the number of lags in the auto sales series and n refers to the number of monthly observations considered. The ccf occurs mostly at the negative lags and peaks somewhere at lag 5 to 6, this indicate that auto sales series clearly leads the unemployment series.

Testing for cointegration

Based on the unit root test results conducted in the univariate section and the moderate, negative linear correlation observed just above. There is some chance that our two series is cointegrated. We conduct the Phillips-Ouliaris Cointegration Test here with the null hypothesis:

- Ho: The two series are not cointegrated
- Ha: the two series are cointegrated
- The po.test results provides evidence that the series are cointegrated since the null hypothesis is rejected at the 1% level.

```
tseries::po.test(cbind(auto.intersect.ts, unem.intersect.ts))
```

```
## Warning in tseries::po.test(cbind(auto.intersect.ts, unem.intersect.ts)):
## p-value smaller than printed p-value
```

```
##
##  Phillips-Ouliaris Cointegration Test
##
## data:  cbind(auto.intersect.ts, unem.intersect.ts)
## Phillips-Ouliaris demeaned = -234.14, Truncation lag parameter =
## 4, p-value = 0.01
```

If there truely exists a linear combination between the two series that is stationary, regressing unemployment series on auto sales should produce residuals that are some what stationary(though it can still be time dependent). We conduct such a model below to examine the residual series.

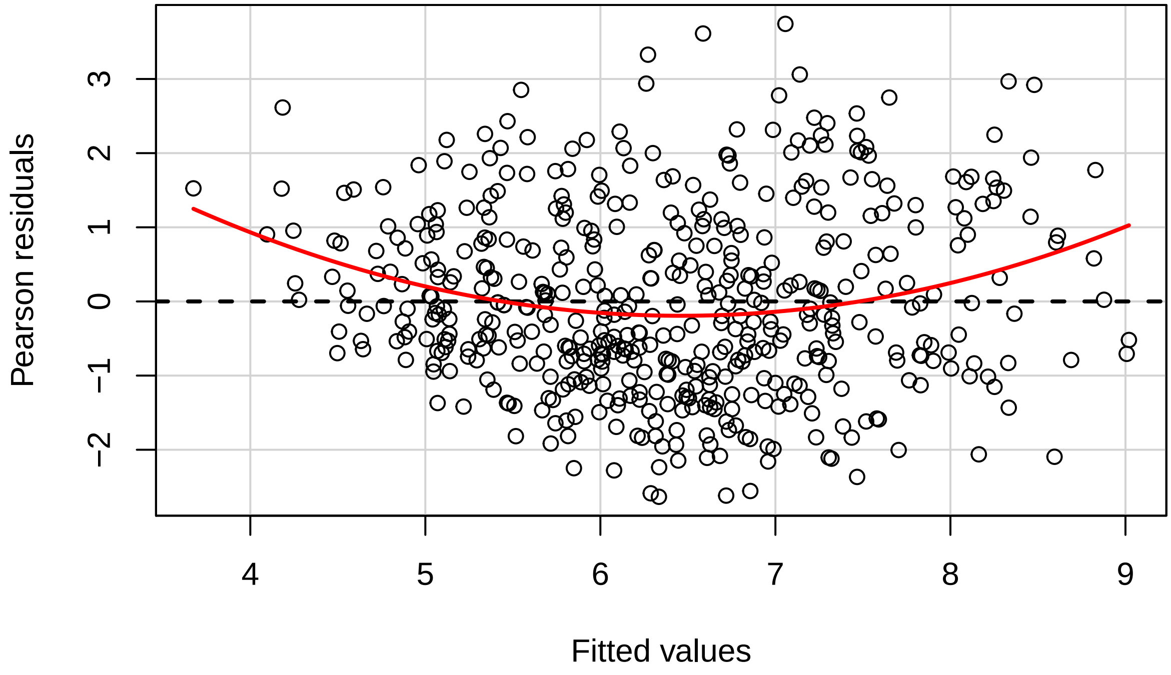
```
unem.auto.lm = lm(unem.intersect.ts~auto.intersect.ts)
#summary(unem.auto.lm)
unem.auto.lm$coefficients[2]
```

```
## auto.intersect.ts
##      -0.004547657
```

```
as.numeric(unem.auto.lm$coefficients[2]/sd(unem.intersect.ts))
```

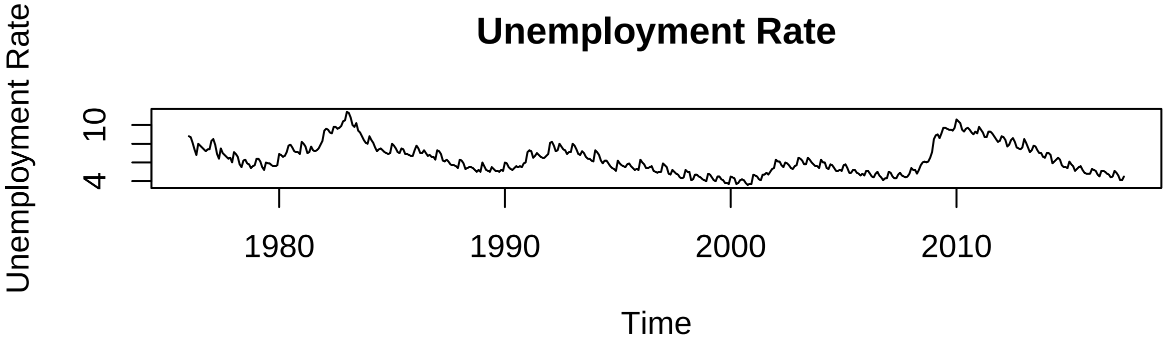
```
## [1] -0.002833825
```

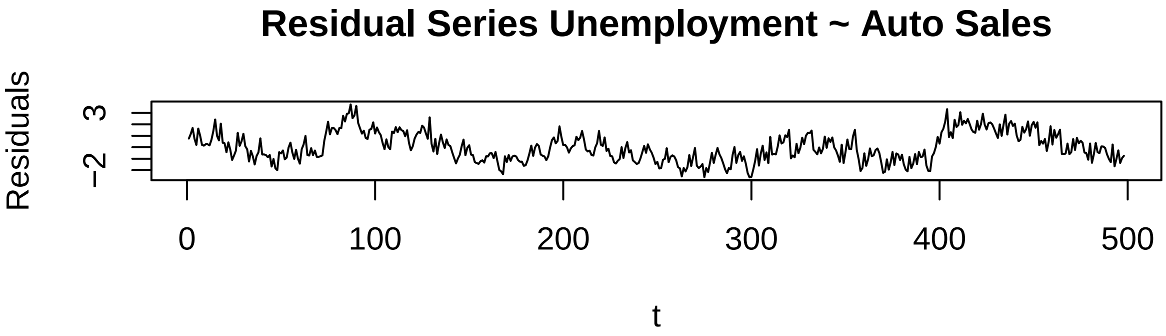
```
car::residualPlot(unem.auto.lm)
```



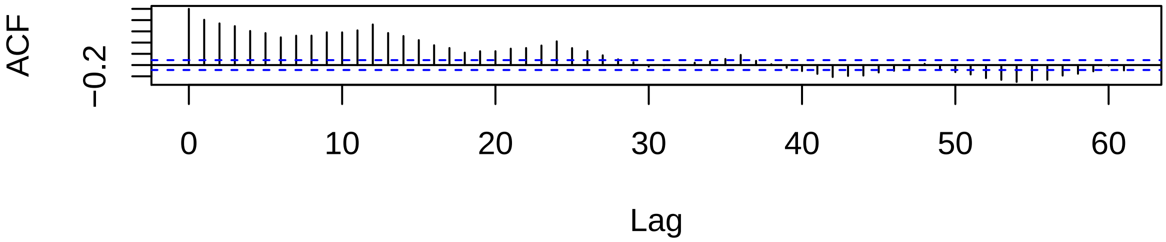
From the regression output above, the coefficient estimate for autosales is not practically significant(less than 1% standard deviation of unemployment rate). The residual plot also show strong curvature. Atuosales at lag 0 doesn't effectively explain variations in unemployment in expectation.

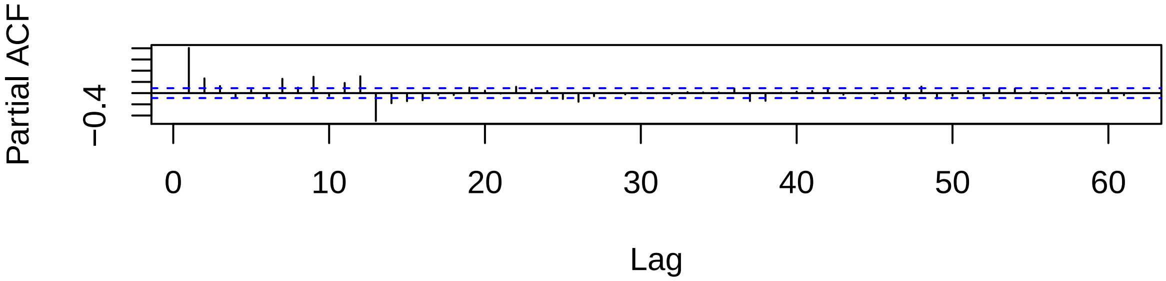
```
par(mfrow = c(2,1))
ts.plot(unem.intersect.ts, main = "",
        ylab = "Unemployment Rate")
title("Unemployment Rate")
unem.auto.lm.res = resid(unem.auto.lm)
plot(unem.auto.lm.res, xlab = "t", ylab = "Residuals",
     lty = 1, pch = 1, type = "l")
title("Residual Series Unemployment ~ Auto Sales")
```





```
par(mfrow = c(2,1))
acf(unem.auto.lm.res,61, main = "")
pacf(unem.auto.lm.res,61, main = "")
```





From the time plots and acf plots, the residual series still picks up most of the random walk behaviors in the unemployment rate. The acf and pacf plots still show evidence of a strong AR(1) process. Contradicting the cointegration test results, these residual plots don't fully support the existence of linear combination between the two series that is entirely stationary.

We conclude the bivariate investigation that there is clearly linear relationship between the unemployment and autosales, but the two are not necessarily cointegrated. A VAR model that regress unemployment on lags of auto sales is probably more appropriate than cointegration specific models, like VECM.