

Canadian Unemployment

David Rucinski

April 1, 2019

```
# Source:
# Organization for Economic Co-operation and Development, Unemployment Rate: Aged 15 and Over: All Pers
# for Canada [LRUNTTTCAM156N], retrieved from FRED, Federal Reserve Bank of St. Louis;
# https://fred.stlouisfed.org/series/LRUNTTTCAM156N, June 11, 2019.
#
#
# OECD, "Main Economic Indicators - complete database", Main Economic Indicators
# (database),http://dx.doi.org/10.1787/data-00052-en April 1, 2019
# Copyright, 2016, OECD. Reprinted with permission.
```

Packages and Loading the Data

```
library("tseries")
library("astsa")
library("TSA")
library("fGarch")
library("forecast")

test <- read.csv("LRUNTTTCAM156N.csv", header = TRUE, stringsAsFactors = FALSE)
class(test)

[1] "data.frame"

unemploy.data <- ts(data=test$LRUNTTTCAM156N, frequency = 12, start=c(1960,1), end=c(2019,3))
class(unemploy.data)

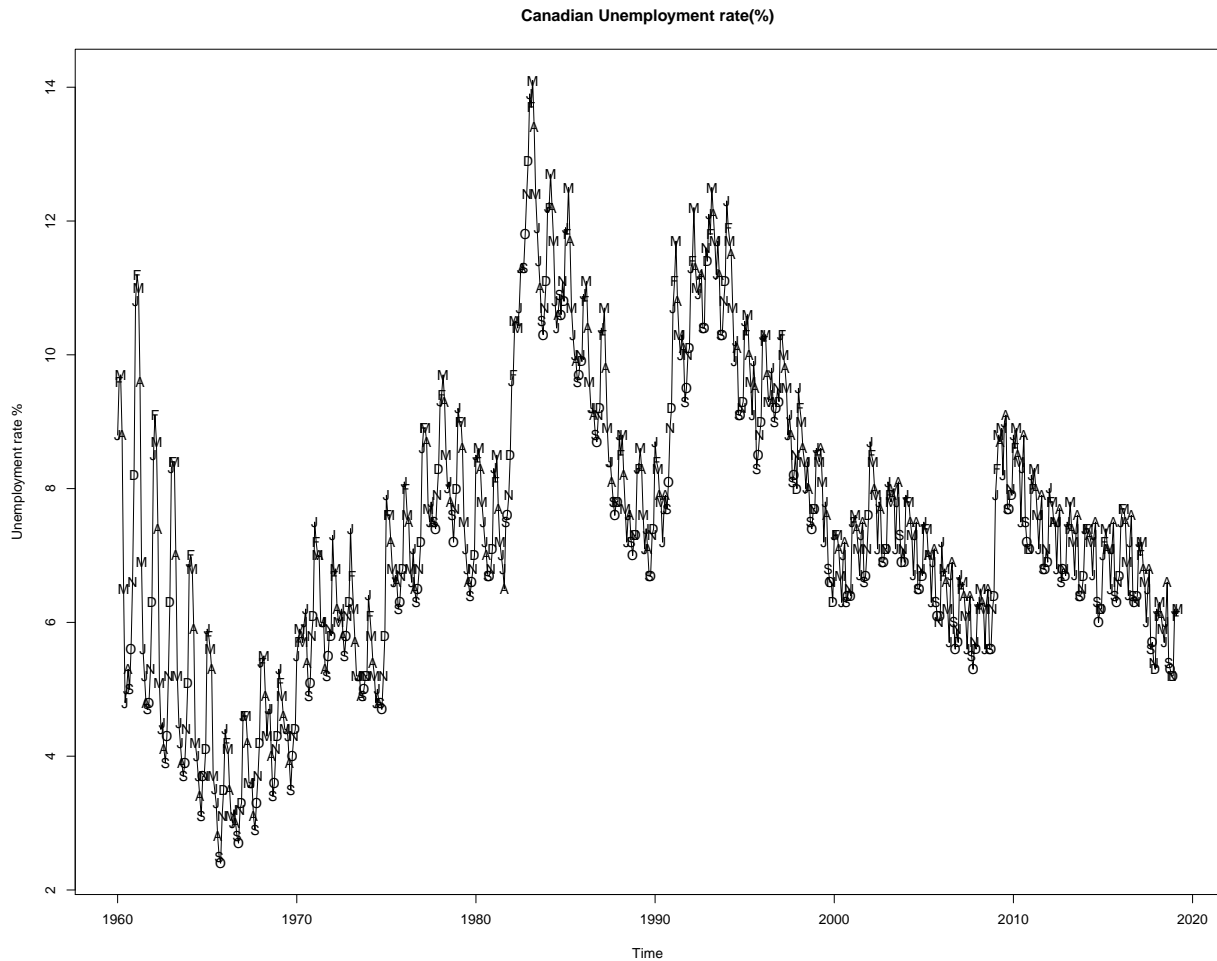
[1] "ts"

# colnames(unemploy.data) <- c("observation_date", "unemployment rate %")
rm(test)
```

Reading in the data from an excel file and converting to xts ran into some issues later as the xts data didn't work with certian functions. I grabbed a CVS file from the source website and created a ts object indexed appropriately.

A Look at the Data

```
plot(unemploy.data, ylab = "Unemployment rate %", main = "Canadian Unemployment rate(%)")
points(unemploy.data, pch = as.vector(season(unemploy.data)))
```



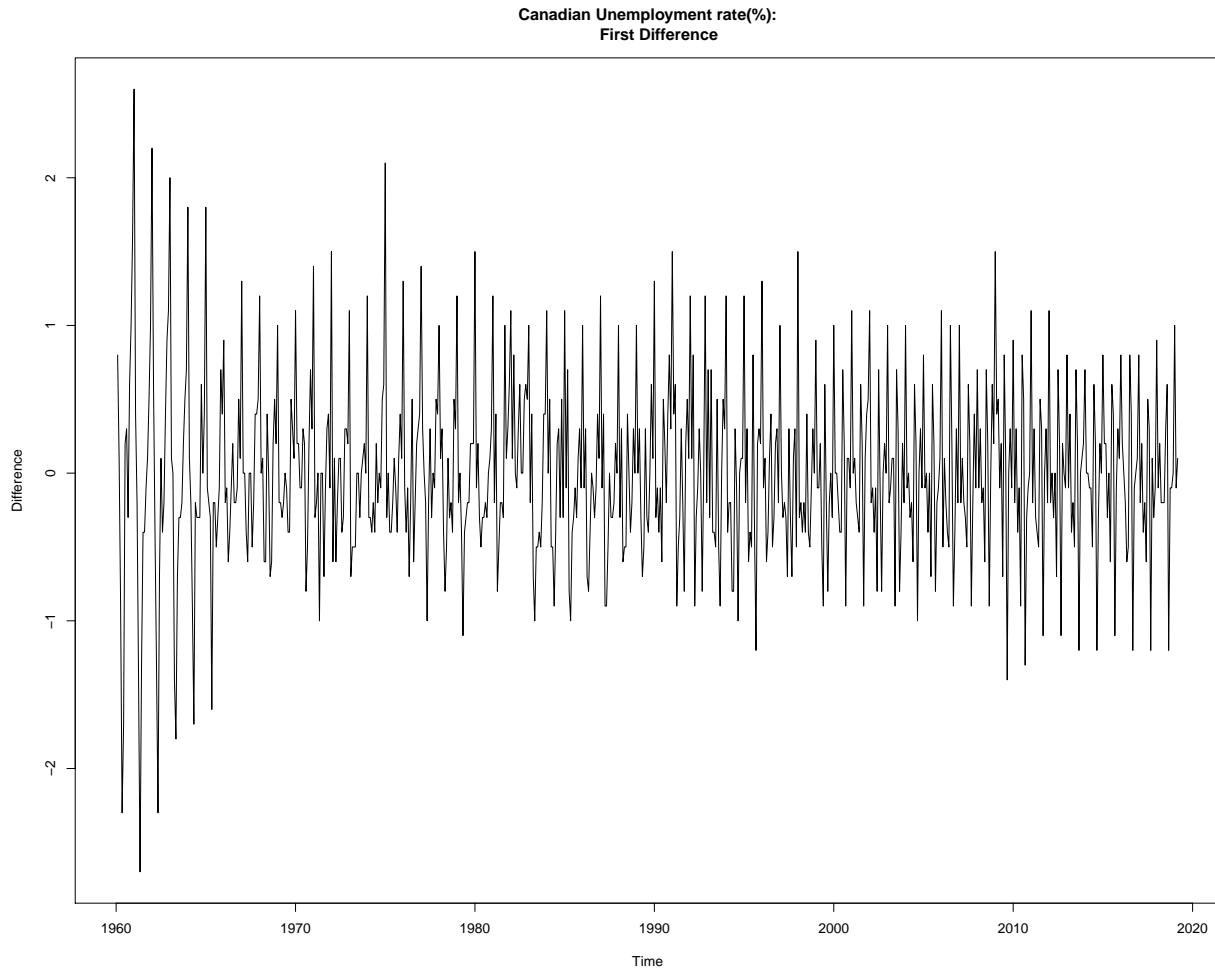
```
# ggplot to show recession on plot
# library(zoo)
# library(ggplot2)
#
# begin <- c(1960,1974,1981,1990,2007)
# end   <- c(1961,1975,1982,1992,2009)
#
# recess <- data.frame(begin,end)
#
# df_plot <- data.frame(x = unemploy.data, time = time(unemploy.data))
#
# ggplot(df_plot, aes(x = time, y = x )) +
#   geom_rect(data = recess,
#             aes(xmin = begin,
#                 xmax = end,
#                 ymin = -Inf,
#                 ymax = +Inf),
#             inherit.aes = FALSE, fill = "red", alpha = 0.2) +
#   geom_line()
```

First glance there's no constant variance or mean, we have a non-stationary process. We need to stabilize the variance and get something stationary. Since it is an unemployment rate does this include seasonal and

contract jobs? If there is a seasonal trend we should take a difference of some lag. Then fit a model and see if we can forecast.

First Difference Plot

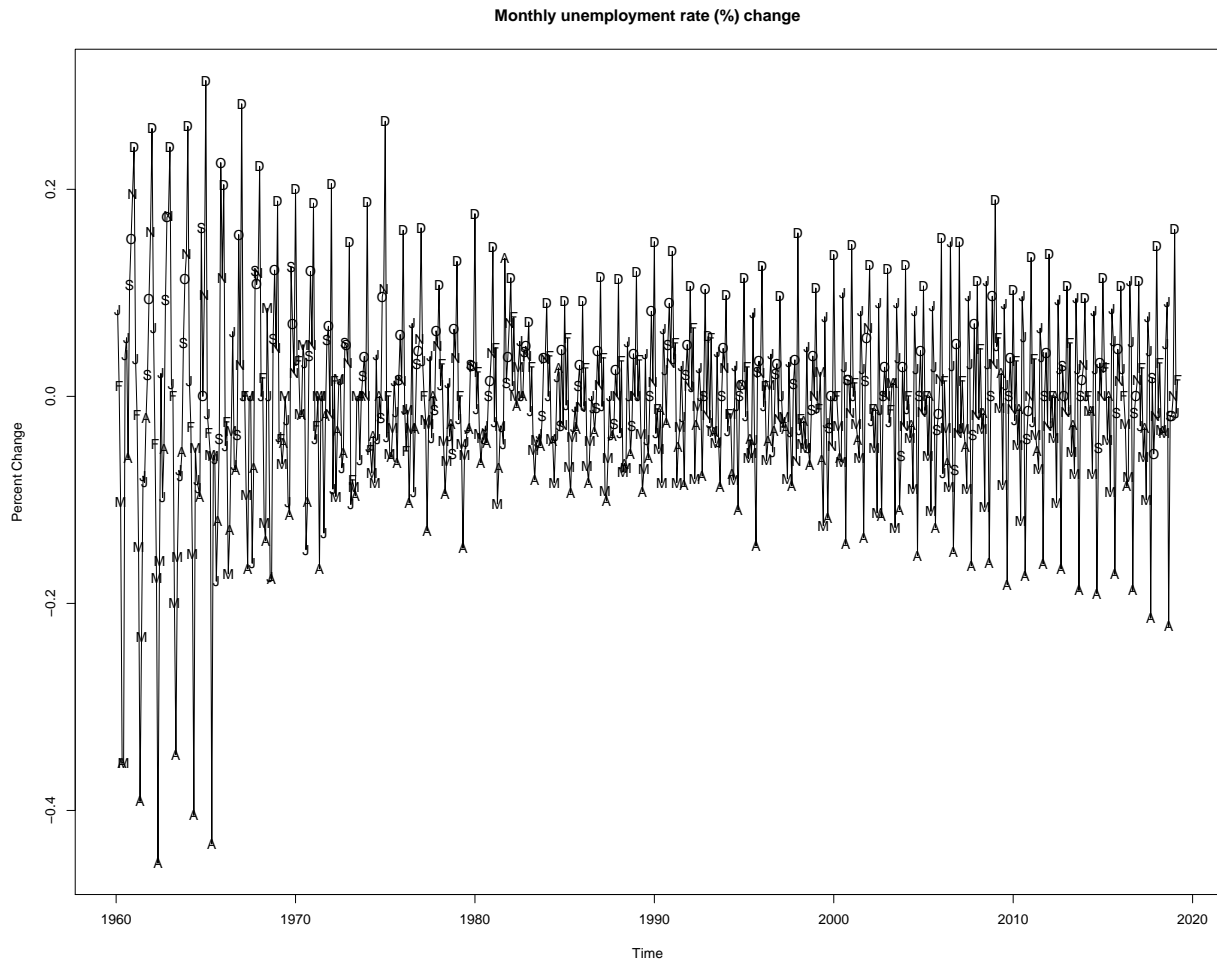
```
plot(diff(unemploy.data), ylab = "Difference", main = "Canadian Unemployment rate(%) :  
First Difference")
```



(First difference) Looks to be a bit more stable with constant mean, the variance still jumps. The peaks for the unemployment rate percent that really stick out are 1974, 1990, 2008. Which happen to be the dates of Canadian recessions.

Percent Change

```
percent.change <- diff(unemploy.data)/unemploy.data  
  
plot(percent.change, ylab = "Percent Change",  
      main = "Monthly unemployment rate (%) change")  
points(percent.change, pch = as.vector(season(unemploy.data)))
```



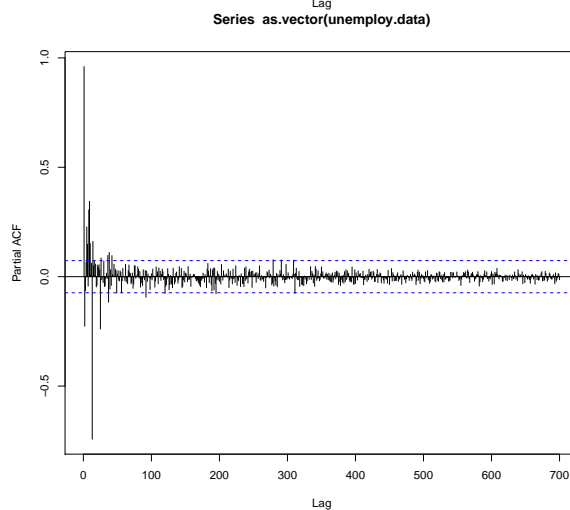
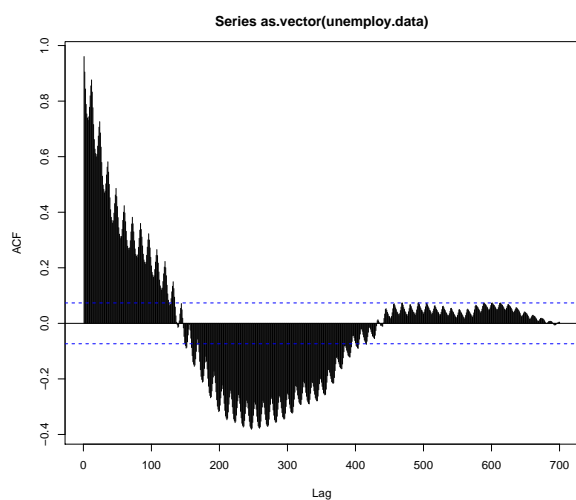
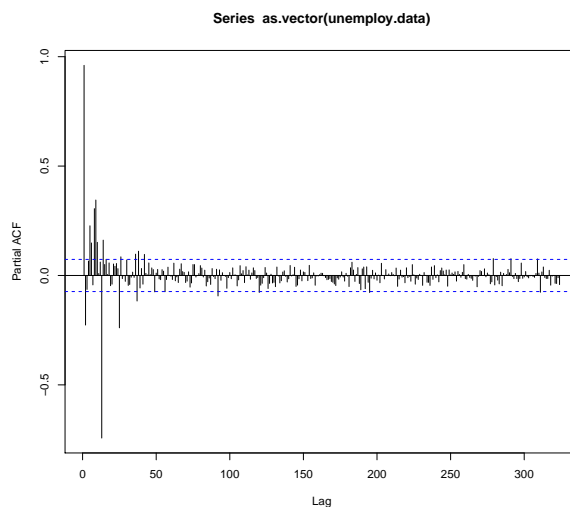
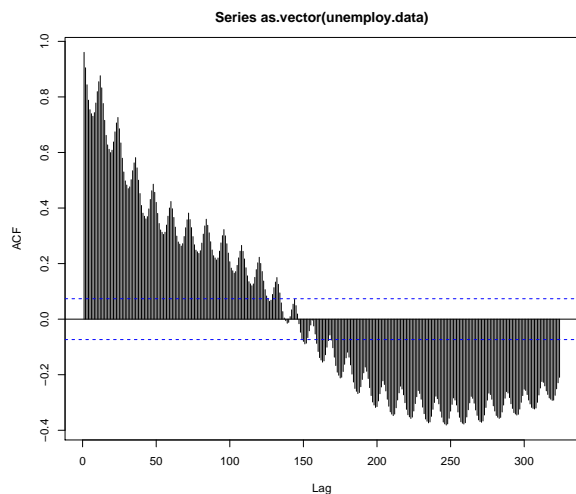
December has the greatest increase for unemployment rate and April has a decrease. A Particular trend we could say accounts for seasonal jobs, e.g. construction is stopped for the winter months.

ACF/PACF & Seasonal Testing

```
par(mfrow = c(2,2), mar = c(4,5,3,3))

acf(as.vector(unemploy.data), lag.max = 324)
pacf(as.vector(unemploy.data), lag.max = 324)

acf(as.vector(unemploy.data), lag.max = 700)
pacf(as.vector(unemploy.data), lag.max = 700)
```



```
month. <- season(unemploy.data)
model <- lm(unemploy.data~month.-1)
summary(model)
```

Call:

```
lm(formula = unemploy.data ~ month. - 1)
```

Residuals:

Min	1Q	Median	3Q	Max
-4.5085	-1.2135	-0.1593	1.0085	5.7271

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
month.January	8.3717	0.2664	31.42	<2e-16 ***
month.February	8.3000	0.2664	31.16	<2e-16 ***
month.March	8.4100	0.2664	31.57	<2e-16 ***
month.April	8.0085	0.2687	29.81	<2e-16 ***
month.May	7.4339	0.2687	27.67	<2e-16 ***
month.June	6.9983	0.2687	26.05	<2e-16 ***
month.July	7.2847	0.2687	27.11	<2e-16 ***
month.August	7.1763	0.2687	26.71	<2e-16 ***

```

month.September    6.6169      0.2687    24.63    <2e-16 ***
month.October      6.6593      0.2687    24.79    <2e-16 ***
month.November     7.0051      0.2687    26.07    <2e-16 ***
month.December     7.1729      0.2687    26.70    <2e-16 ***

```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.064 on 699 degrees of freedom
Multiple R-squared: 0.9304, Adjusted R-squared: 0.9292
F-statistic: 779.1 on 12 and 699 DF, p-value: < 2.2e-16

#plot(model) in the console will give inverse AR and MA roots on the unit circle [neat!]

Interesting, there are a lot less significant lags with the PACF. Certainly no simple models, the ACF seems to tail off in a cyclical fashion with a definite seasonal component. Let us check the first difference ACF/PACF.

ACF/PACF of the First Difference

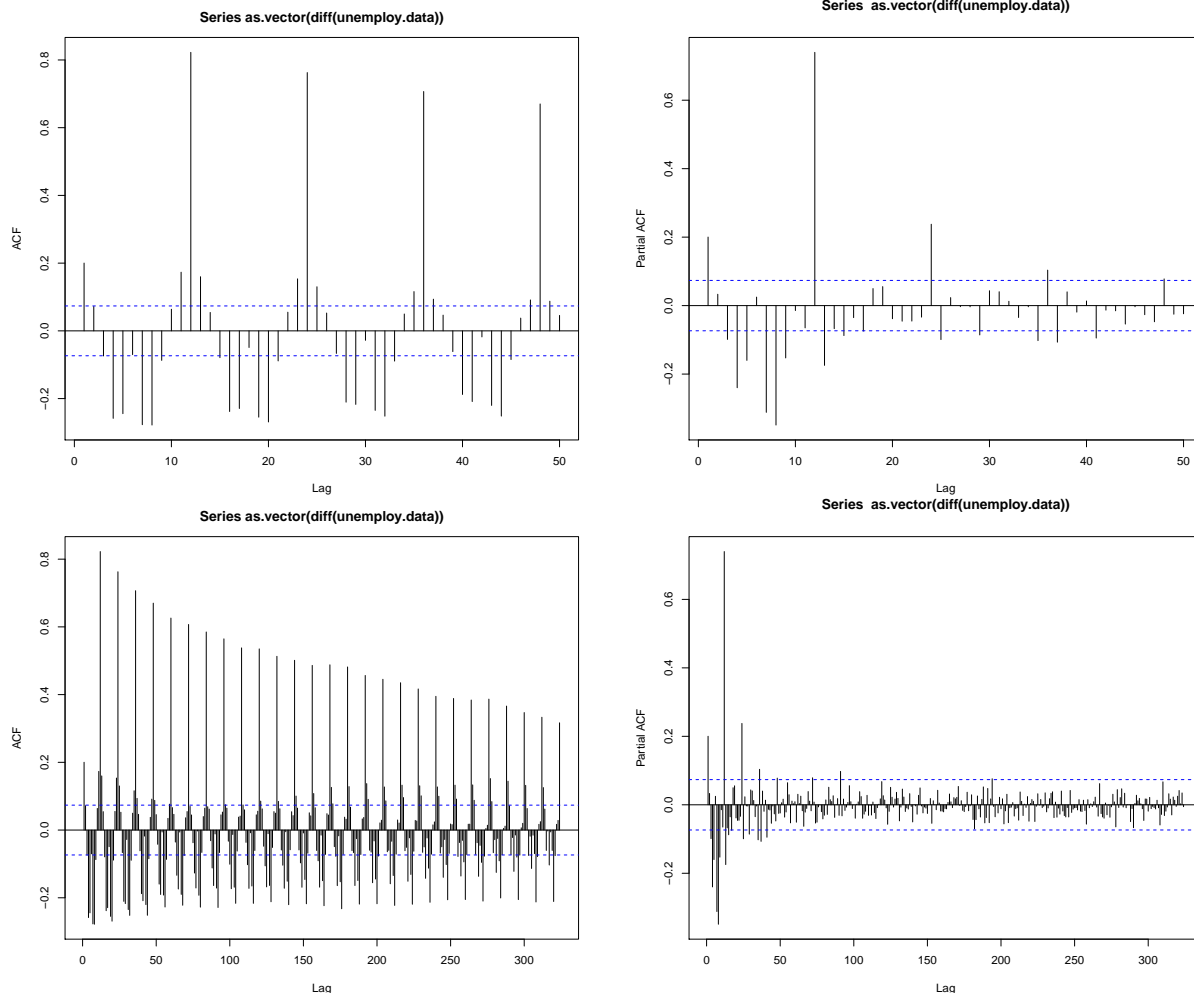
```

par(mfrow = c(2,2), mar = c(4,5,3,3))

acf(as.vector(diff(unemploy.data)), lag.max = 50)
pacf(as.vector(diff(unemploy.data)), lag.max = 50)

acf(as.vector(diff(unemploy.data)), lag.max = 324)
pacf(as.vector(diff(unemploy.data)), lag.max = 324)

```



Still tailing off in the ACF plot, though the cyclical pattern is removed. There is a seasonal trend present with a period of 12. PACF has some significant lags before 10, but there is also a significant seasonal trend present as well.

Trying Models

*# I tried a nested for loop to compute a large number of models, but could not save them
in an object to use a minimum function on the list. Also with some combinations there
were errors with infinity, so using previous ACF/PACF plots I over fit the data and worked down.*

```
arima(unemploy.data, order = c(1,1,1), seasonal = list(order = c(4,0,1), period = 12))$aic
```

```
[1] 189.399
```

```
arima(unemploy.data, order = c(1,1,1), seasonal = list(order = c(5,0,1), period = 12))$aic
```

```
[1] 191.2734
```

```
arima(unemploy.data, order = c(1,1,1), seasonal = list(order = c(5,0,0), period = 12))$aic
```

```
[1] 189.5465
```

```
arima(unemploy.data, order = c(1,1,1), seasonal = list(order = c(4,0,0), period = 12))$aic
```

```
[1] 187.6318
arima(unemploy.data, order = c(1,1,1), seasonal = list(order = c(3,0,0), period = 12))$aic

[1] 196.4065
arima(unemploy.data, order = c(1,1,1), seasonal = list(order = c(1,0,1), period = 12))$aic

[1] 184.8298
arima(unemploy.data, order = c(1,1,1), seasonal = list(order = c(1,0,2), period = 12))$aic

[1] 186.8295
# Taking a look at different models, and then choosing where Ljung-Box is not rejected
# we get : sarima(unemploy.data, 1,1,1, 1,0,1, S = 12)
```

Using minimum AIC to select a model AIC minimum is 184.82 so I will use that model, i.e. ARIMA(1,1,1)x(1,0,1)_{S=12}

Through some checking with the `sarima()` function I have found a handful of models with relatively low AIC and fail to reject Ljung-Box test. Ljung-box is testing to see if the residuals when grouped together behave as white noise at certain lags.

Fitting the Model

```
model <- arima(unemploy.data, order = c(1,1,1), seasonal = list(order = c(1,0,1),
                                                                period = 12))

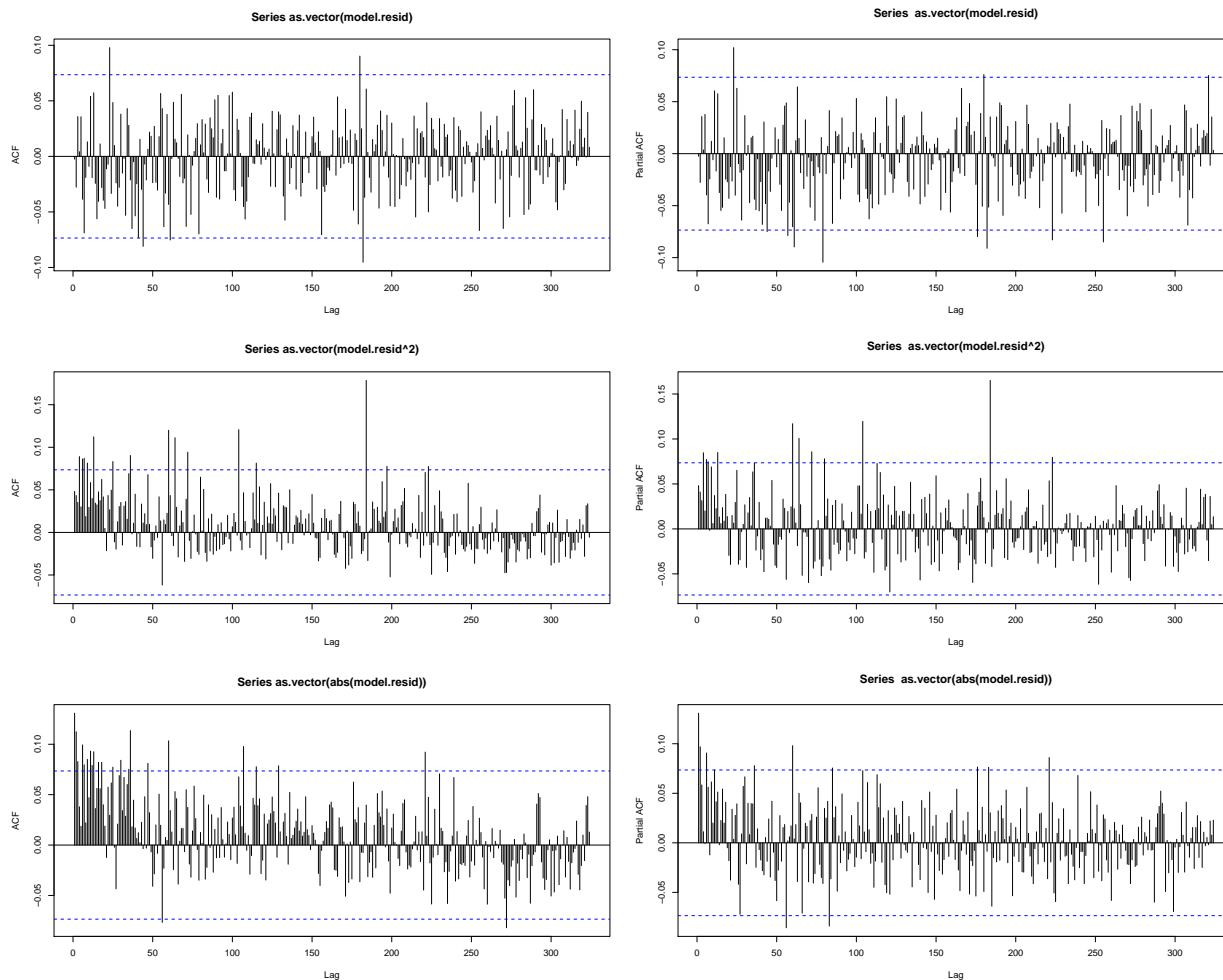
model.resid <- model$residuals

par(mfrow = c(3,2))

acf(as.vector(model.resid), lag.max = 324)
pacf(as.vector(model.resid), lag.max = 324)

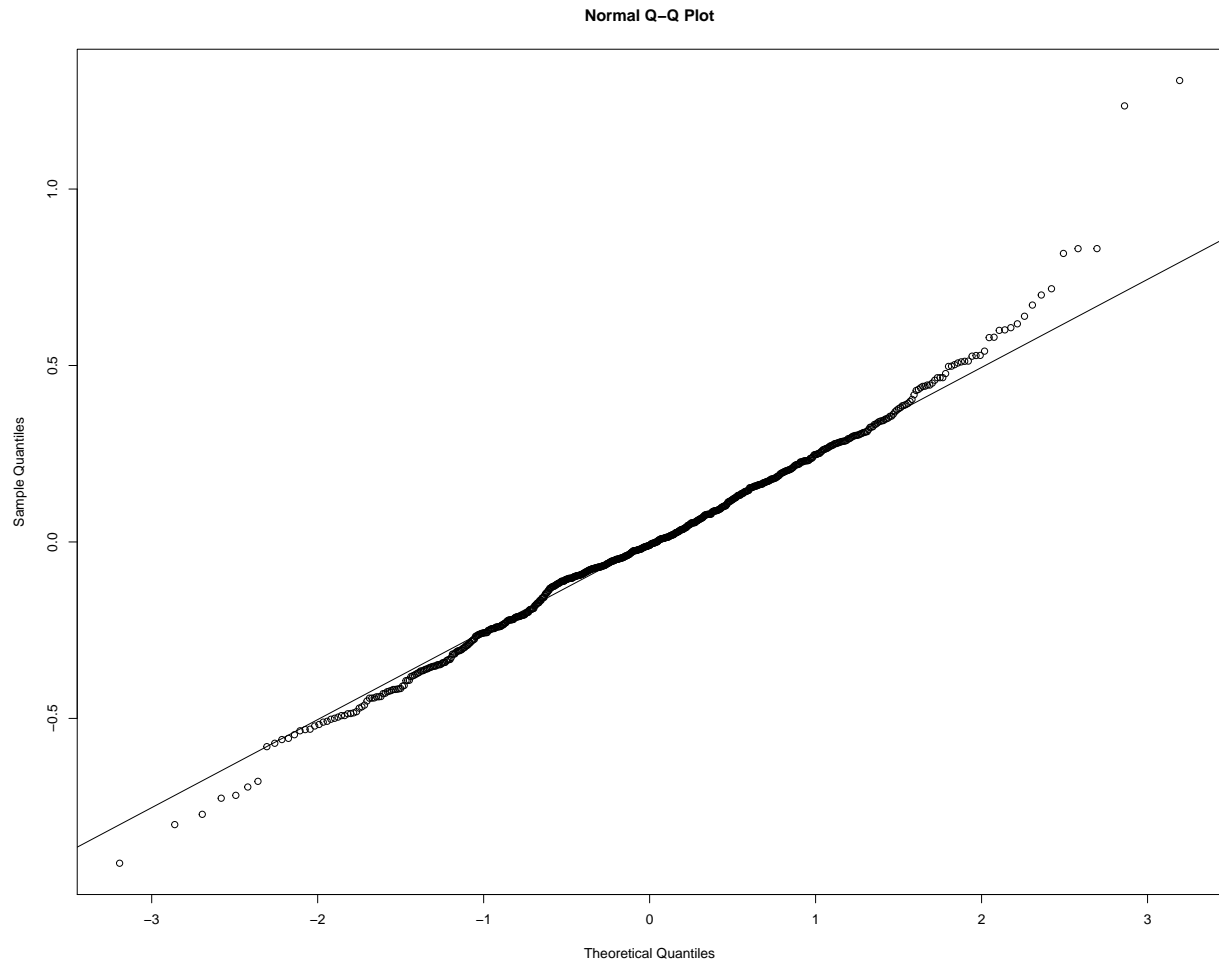
acf(as.vector(model.resid^2), lag.max = 324)
pacf(as.vector(model.resid^2), lag.max = 324)

acf(as.vector(abs(model.resid)), lag.max = 324)
pacf(as.vector(abs(model.resid)), lag.max = 324)
```

ACF of the residuals looks good with few significant lags, but with further investigation taking the absolute values and squared values of the residuals we see that they are not behaving as white noise. Thus the residuals will not be independent. Since the variance varies for the residuals we should probably fit a GARCH model.

```
qqnorm(model.resid)
qqline(model.resid)
```

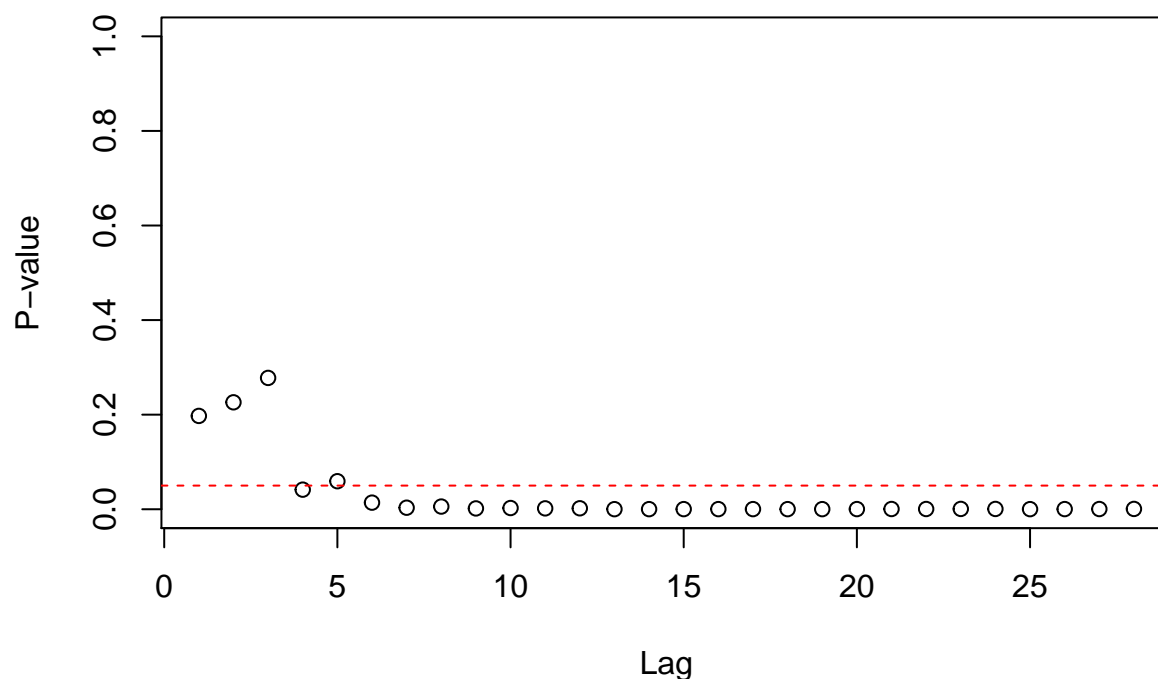


Residuals are approximately normal with tail behaviour, lets continue for fitting a GARCH model.

Testing for ARCH Effects

```
McLeod.Li.test(model, main = "McLeod-Li Test")
```

McLeod-Li Test



The null hypothesis of the McLeod-Li test is that there is no autoregressive conditional heteroskedasticity (ARCH) among the lags. Therefore we reject the null hypothesis at 5% level of significance, there are ARCH effects present.

Fitting a GARCH Model to Transformed Data

If we can fit a GARCH model and have the residuals behave as white noise, then we can account for the ARCH effects.

Checking the fit of GARCH(1,1)

```
summary(fit)
```

Call:

```
garch(x = diff(diff(unemploy.data), 12), order = c(1, 1))
```

Model:

```
GARCH(1,1)
```

Residuals:

Min	1Q	Median	3Q	Max
-2.9045	-0.7220	0.0000	0.6394	5.1424

Coefficient(s):

```

      Estimate Std. Error t value Pr(>|t|)
a0  0.004677   0.002455   1.905  0.05677 .
a1  0.076387   0.024099   3.170  0.00153 **
b1  0.873338   0.045812  19.063 < 2e-16 ***
---

```

```

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Diagnostic Tests:

Jarque Bera Test

data: Residuals

X-squared = 33.639, df = 2, p-value = 4.96e-08

Box-Ljung test

data: Squared.Residuals

X-squared = 0.0037817, df = 1, p-value = 0.951

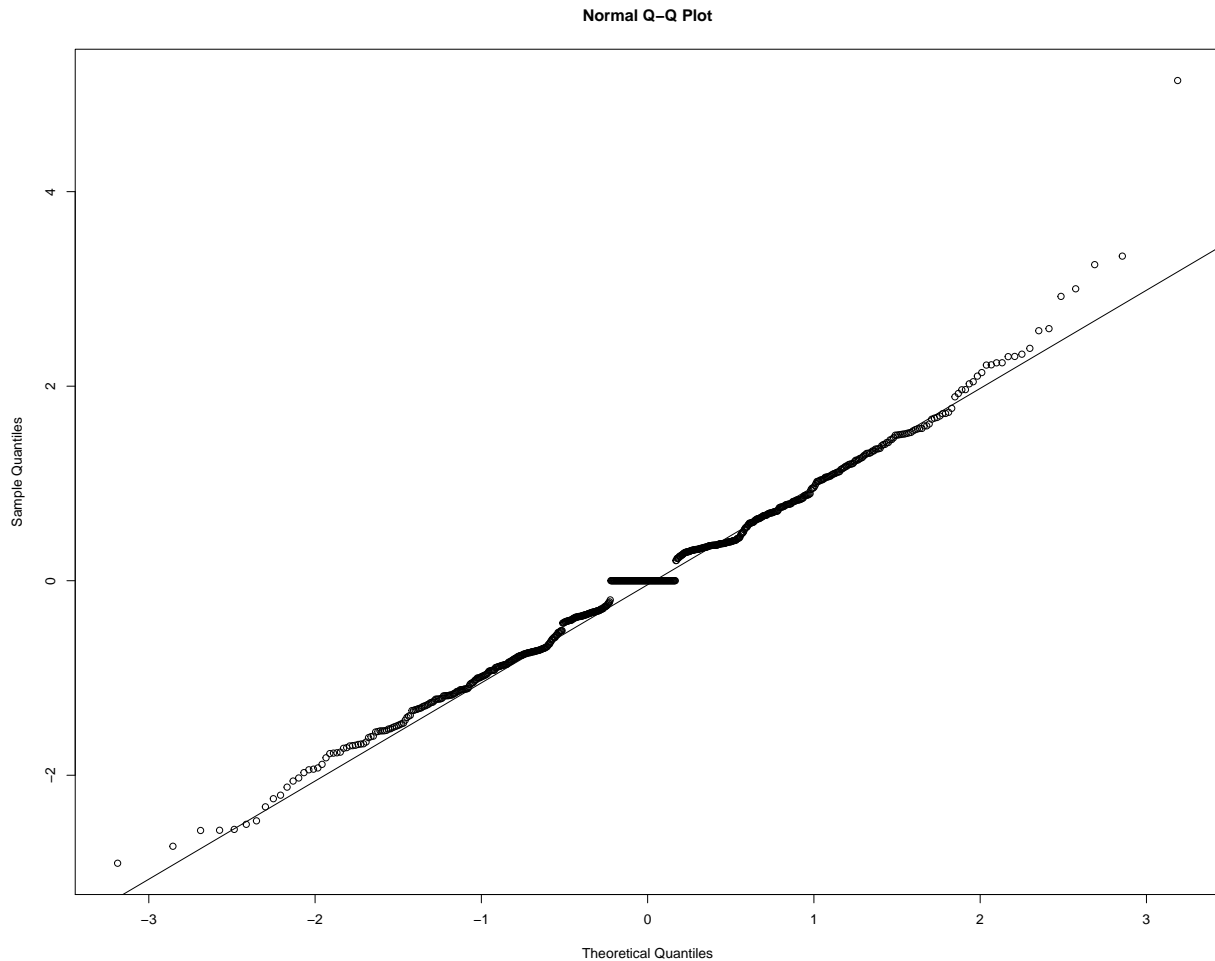
```
fit.resid <- fit$residuals
```

```
# see plot(fit), need some time to play with setting to get it right.
```

The GARCH(1,1) model seems to be significant for the seasonal first difference model.

Residual Analysis of GARCH(1,1) on Transformed Data

```
qqnorm(fit.resid)
qqline(fit.resid)
```



```
resid.fit.model <- fit.resid[-1]
```

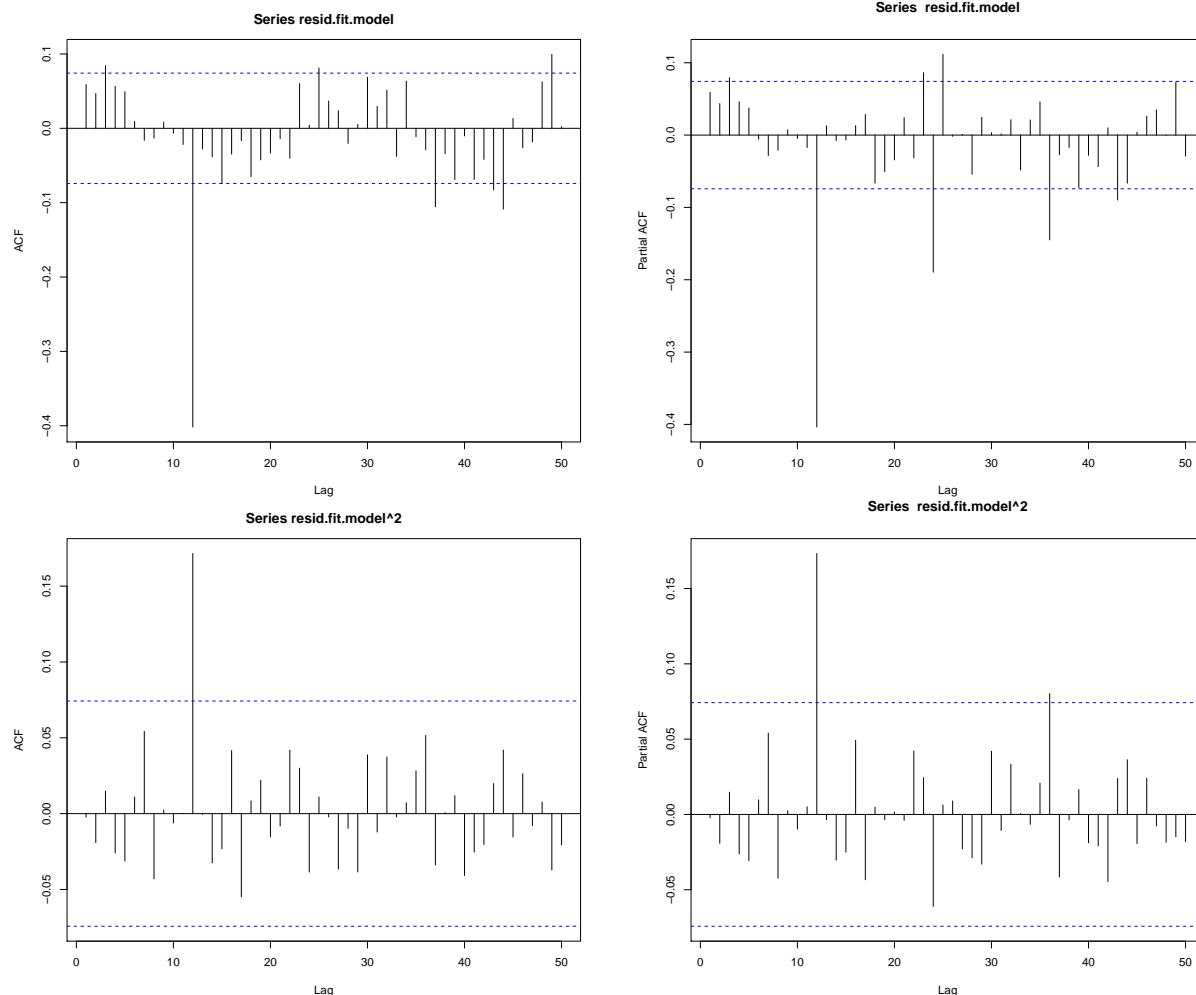
Very odd straight line horizontally but is approximately normal with tail behaviour, especially compared to a GARCH(1,1) on the original data or the first difference.

ACF/PACF of GARCH Residuals

```
par(mfrow = c(2,2), mar = c(4,5,3,3))

acf(resid.fit.model, lag.max = 50)
pacf(resid.fit.model, lag.max = 50)

acf(resid.fit.model^2, lag.max = 50)
pacf(resid.fit.model^2, lag.max = 50)
```



ACF seems fairly good with the exception at lag 1 & 12, the PACF on the other hand have a couple significant lags. Although on the squared residuals it does seem to behave more as white noise again with the exception of lag 12. As if the residuals themselves have a seasonal moving average (MA) component, so lets fit a ARMA + GARCH model.

Final Model with ARMA and GARCH

```
summary(out)
```

Title:

GARCH Modelling

Call:

```
garchFit(formula = ~arma(1, 1) + garch(1, 1), data = trans.data)
```

Mean and Variance Equation:

```
data ~ arma(1, 1) + garch(1, 1)
```

```
<environment: 0x000000001c5fe3d0>
```

```
[data = trans.data]
```

Conditional Distribution:

```
norm

Coefficient(s):
      mu      ar1      ma1      omega      alpha1      beta1
0.0014863  0.7488479 -0.6868358  0.0042723  0.0678341  0.8856721
```

```
Std. Errors:
based on Hessian
```

```
Error Analysis:
      Estimate Std. Error t value Pr(>|t|)
mu      0.001486  0.003383   0.439  0.66042
ar1     0.748848  0.096154   7.788 6.88e-15 ***
ma1    -0.686836  0.111796  -6.144 8.06e-10 ***
omega   0.004272  0.002253   1.896  0.05796 .
alpha1  0.067834  0.023643   2.869  0.00412 **
beta1   0.885672  0.040848  21.682 < 2e-16 ***
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Log Likelihood:
-143.9148      normalized: -0.2061817
```

```
Description:
Tue Jun 11 14:42:41 2019 by user: User
```

```
Standardised Residuals Tests:

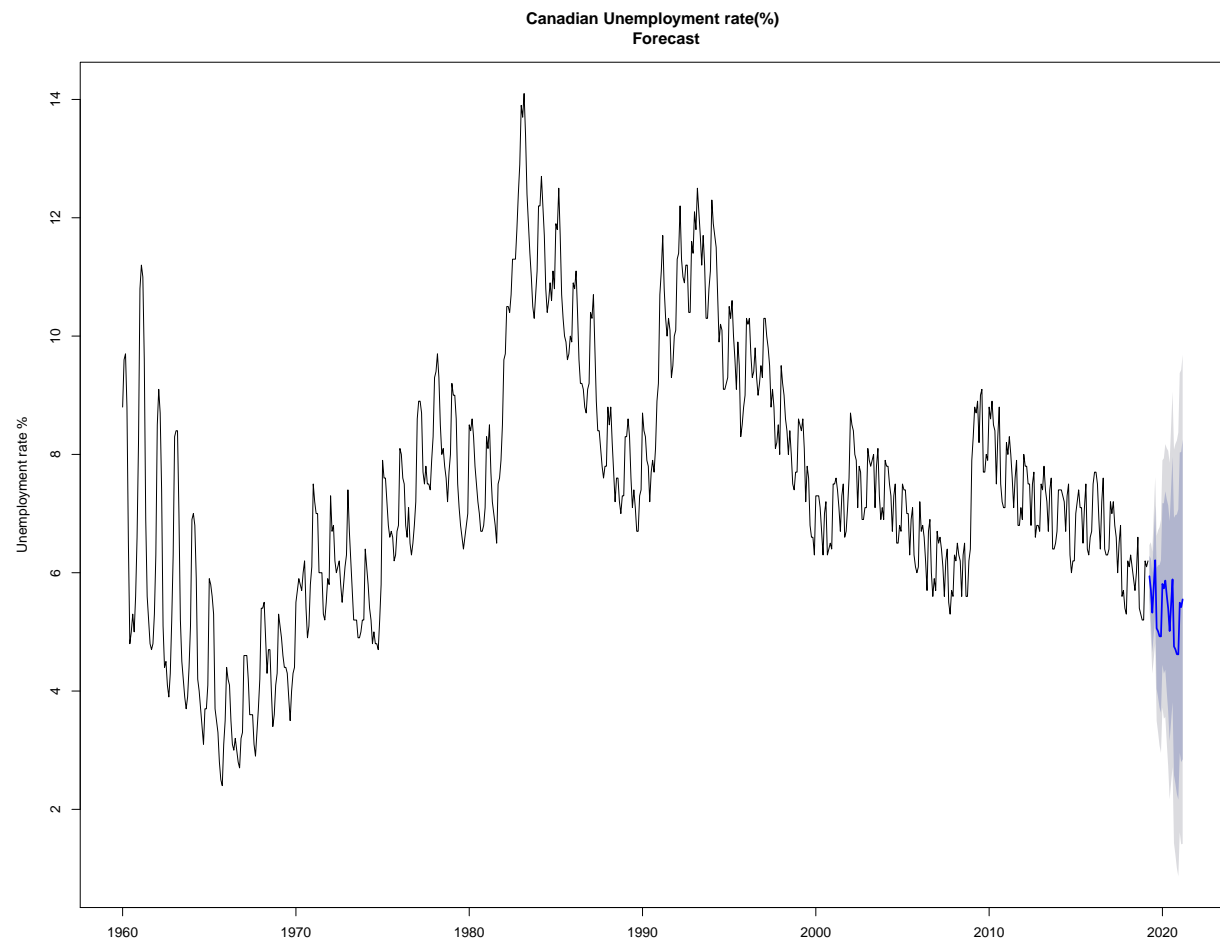
      Statistic p-Value
Jarque-Bera Test  R    Chi^2  33.02486  6.741293e-08
Shapiro-Wilk Test  R    W      0.991846  0.0006936932
Ljung-Box Test     R    Q(10)  4.038916  0.9455743
Ljung-Box Test     R    Q(15) 121.0769   0
Ljung-Box Test     R    Q(20) 124.6216   0
Ljung-Box Test     R^2  Q(10)  4.966406  0.8934108
Ljung-Box Test     R^2  Q(15) 34.28077  0.003109309
Ljung-Box Test     R^2  Q(20) 37.0585   0.01151444
LM Arch Test       R    TR^2   31.75959  0.001507002
```

```
Information Criterion Statistics:
      AIC      BIC      SIC      HQIC
0.4295554  0.4686519  0.4294093  0.4446705
```

All parameters are significant but the residual analysis of this model shows that they are not normal, and do not behave as white noise after lag 10. That is the residuals are not independent.

Although I could not fit a SARIMA + GARCH model, I would suggest to fit that model and test the residuals for white noise and normality. Since it seems to have a variance that is dependent on time, forecasting values may give a wide interval. There are also external factors that will impact the unemployment rate in Canada, recessions and innovation to name a couple. Thus the final model may be quite complex when including such factors.

```
plot(forecast(unemploy.data, model = model), ylab = "Unemployment rate %", main = "Canadian Unemployment
Forecast")
```



A forecast of the SARIMA, it may not be as accurate since this does not include the ARCH effects.