PSTAT 174 Final Project

Employment Levels: Time Series Analysis

Authors:
Kai Dong
Jiale Liang
Mujie Wang
Ying Yan
Tong Zhao

Supervised By:
Professor Sudeep Bapat

Contents

Abstract

The importance of Employment levels in the world today for any economy cannot be overemphasized. Governments and world leaders are becoming aware of the increasing importance of predicting the levels of employment levels to be able to develop better policies for the subjects under them for informed fiscal planning and resource allocation. This project intends to predict the monthly employment levels in Wisconsin, USA using a select method of time series analysis. Basing our prediction on a stationary data set, we use AIC and BIC to build a suitable SARIMA model. This is made possible by a combination of transformations and differencing of the dataset containing the original time series, and thereafter performing a diagnosis check and spectral analysis to establish suitability of a final model. Using the selected model, we made a forecast for the next 10 months on the dataset that is for the months of November 1978 through June 1979. As we shall see in the analyses, the values predicted using the model are well with the 95% confidence interval and close to the values in the original dataset.

Introduction

Employment levels are an important factor in the growth and wellbeing of any economy in the world. Citizens want to get jobs they are trained to do or have expertise in, which translates to them getting a source of income to maintain a certain kind of lifestyle. Government authorities at any level wish to know what the future as far as employment is concerned looks like in order to better prepare themselves for it. They would wish to get this kind of insight in order to be able to develop policies, both fiscal and social, to be able to cushion its citizens from the negative impacts these employment levels may have on them or better utilize the opportunities that better future prospects present. We have chosen this dataset because it contained the different components of a time series and is from a time in the past when there were many factors playing out in the determining the state of the American economy, and by extension that of Wisconsin. It also afforded us the opportunity to compare the values we predicted with the actual values of that time, thus testing the suitability of the prediction model. Apart from that, it was a considerably large enough dataset of 178 observations which we felt was a large enough sample from which to build a model.

An initial plot of the original dataset gives us a general idea of what the time series looks like, in this case we observe an upward trend pointing to the increasing employment levels throughout the duration under consideration. We also observe the existence of a seasonal component in the time series spread over uniform intervals of the time series, which we would expect on this kind of labor data given the inherent effect of the distinct weather seasons experience in Wisconsin.

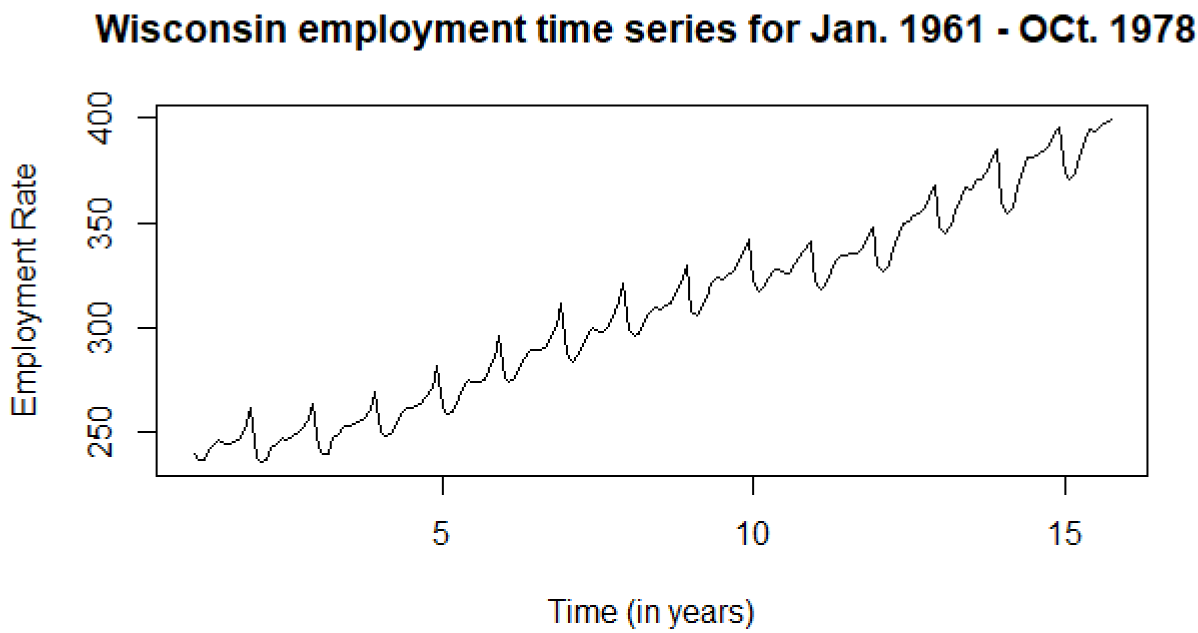
Data Exploration

Initial Exploration

The time series dataset we used was obtained from an online source, it is a time series of two variables – time (in months and years) and employment rate recorded as number of people in employment. It is made up of 178 observations recorded from January 1961 to October 1978, from which we cut observations recorded for 1978 for later use as a testing set for the values we will predict using the model we shall develop. The remaining dataset contains 168 observations and we will use it as a training set for our model.

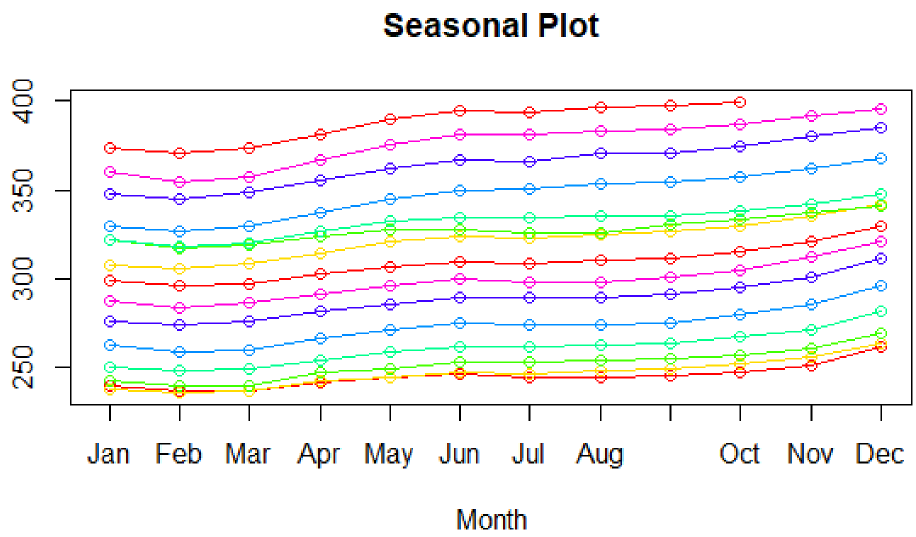
The initial plot was intended to give insight into the data to help us make better choice of model for our time series. It is a plot of the time series with x-axis bearing the time factor and y-axis the employment rate as shown in Figure 1 below.

Figure 1: Plot of the original data



The plot above clearly indicates a data set that is seasonal in nature as we shall examine further in the following sections. The time series is upward sloping with a distinct large spike present in every season recorded. To have a clearer picture of the behavior of the dataset's seasonality, we did a seasonal plot.

Figure 2



We observe from Figure 2 above that the month of December records the highest employment rate for the year while February and March generally record the lowest rates. With the information above alone we cannot confidently conclude that there is in fact seasonality that is significant in the data until we decompose it into the individual components.

Decomposition of the Time Series

In time series analysis there are three components one may come across in the course of analysis: seasonal, trend and random. The data we are working with may have any of the three components, a combination of two or all of them.

Figure 3

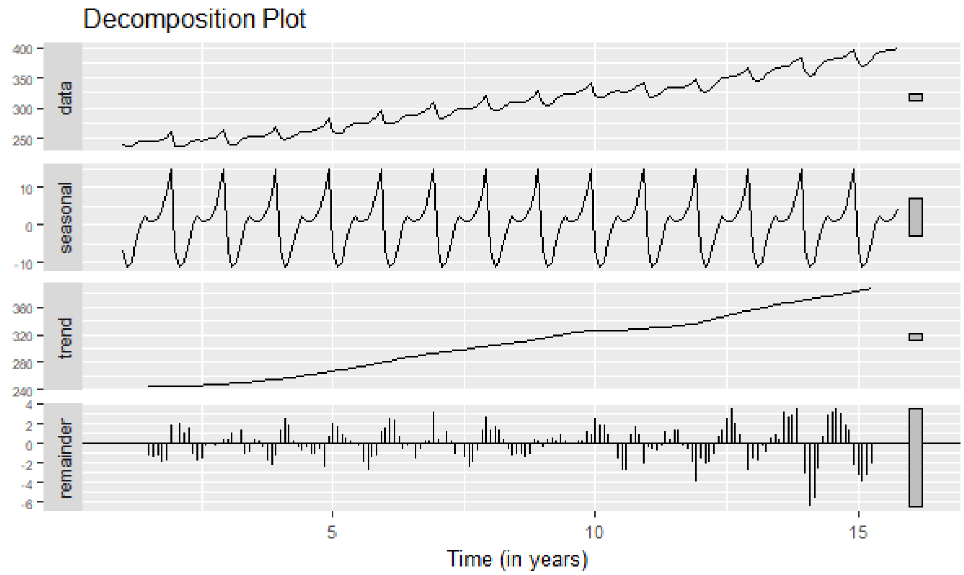


Figure 3 above shows us there exists seasonality in the data with a pattern, a small spike then a relative stability for a short time before a bigger spike can be seen in the month of December. The trend component is clear, a gradual upward movement, and the random component is also present as shall be seen more clearly after making the data stationary in the following steps. Considering all the factors above, we conclude that the data is not stationary and we should, therefore, make it stationary through either transformation, differencing or a combination of the two. This should make the dataset ready for further steps towards building a suitable model.

Transformation

In the following analyses we shall find a way of making the data stationary and then make the variance of the data set stable. This will help us remove the seasonal component that is evident in the data we are working with.

Stabilizing the Variance

A Box-Cox transformation is often used stabilize variance a dataset we may be interested in but before doing the transformation, it is always advisable to examine if it will be of benefit to do the transformation. We plot a Box-Cox graph with an established confidence limit, if 0 is within the range of the limit we do not perform the transformation and vice versa.

Figure 4

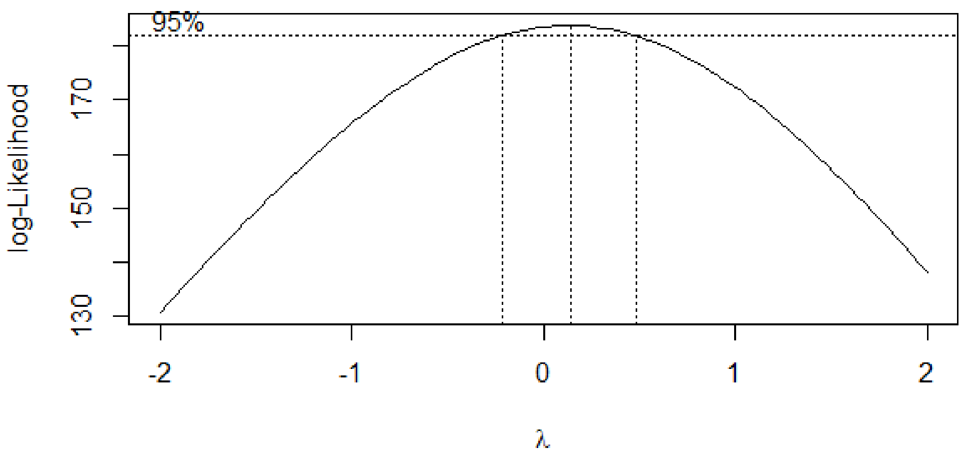
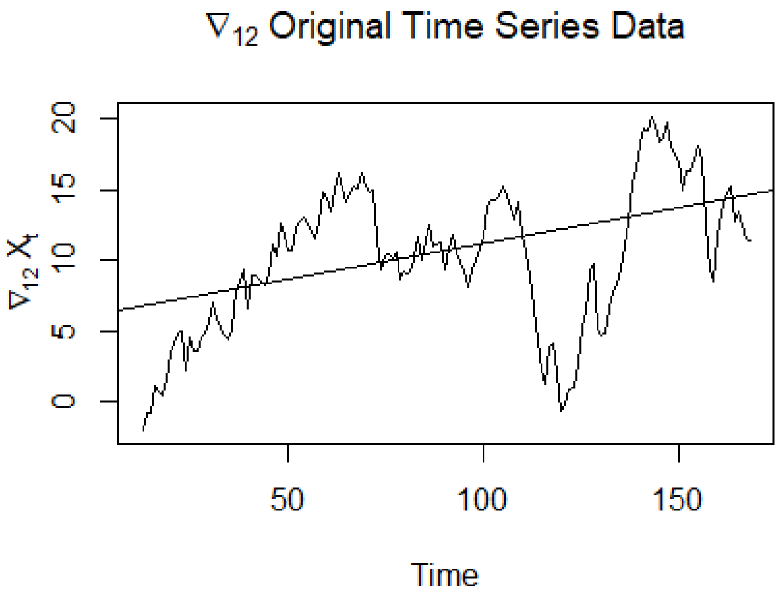


Figure 4 above shows that 0 is contained in the 95% confidence interval of the Box-Cox graph, meaning the variance is stable and the transformation is not helpful in this case.

Making the Time Series Stationary

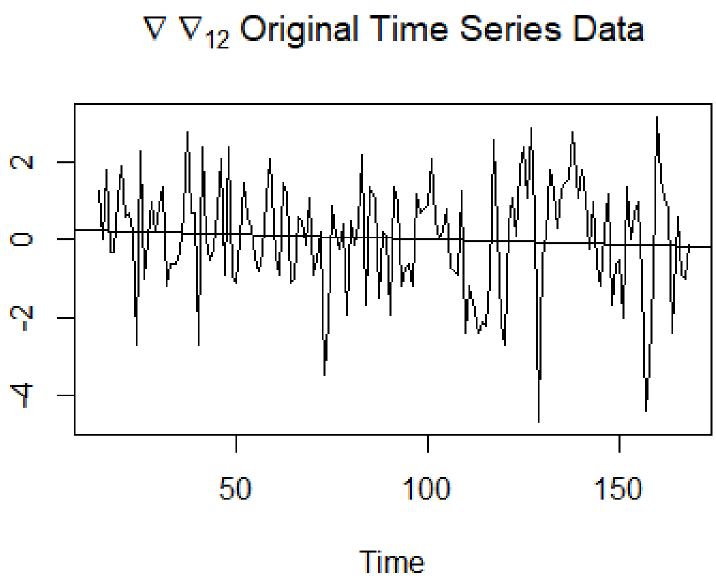
From the initial insights we already know that the data has annual seasonality, that is a season happens for 12 months before it goes and another one comes. We assign the seasonal component a period 12 to remove the seasonality as shown in Figure 5 below.

Figure 5



The variance obtained for differenced data is 25.88961 and there is a trend with a significant positive gradient which we have to remove from the data to help us do further analyses on stabilized data. Differencing again at lag = 1 removes the trend component as seen in Figure 6 below. The variance of the second-degree differenced data is 2.104166, which is less than the variance of the first-degree differenced data above. The trend line of the second differenced data has a slight negative gradient when it should have ideally been horizontal. We future difference the data for a third time at lag 1 in the hope of removing this slight trend but realize the variance of the three-times differenced data is 3.447368 which is higher than that of twice-differenced data. This is an indication of over differencing, so we work with the data set that was differenced one time at lag 1 as above.

Figure 6



To confirm our conclusion, we perform a Dickey-Fuller test. The hypothesis is stated thus:

$$H_0\text{: The time series is not stationary}$$

$$H_1\text{: The time series is stationary}$$

The calculated p-value is 0.01, which is less than the printed p-value so we reject the null hypothesis and conclude with 95% confidence that the data is stationary.

Identifying and Estimating Models

The initial plot of the employment rates in Wisconsin indicates seasonality in the period under consideration. Considering there also is a trend component to the data, a SARIMA model would be appropriate. The model has the following parameters:

SARIMA (p, d, q) x (P, D, Q)

with;

p being the order of an AR process without seasonality;
d being the non-seasonal differencing of the time series;
q being the order of MA process without seasonality;
P being the order of the AR process with seasonality;
D being seasonal differencing;
Q being the order of seasonal AR process;
s being the period of seasonality

The time series data of Wisconsin employment between the years 1961 and 1978 has annual seasonality, so our s = 12 since the data was recorded monthly. As indicated above, we differenced the data once at lag = 12 to remove the seasonal component then differenced it again at lag = 1 to remove the trend component. This means d = 1 and D = 1 and we should proceed to find the missing values that are P, Q, p and q. We use ACF and PACF plots to help us find these values.

Initial Identification

ACF and PACF of the time series from which seasonality and trend have been removed as seen in Figure 7 below help us identify P and Q at lags 12, 24, 36, … From the plot we note that P = 0 & Q = 1. To find p and q we check the ACF/PACF plots in Figure 8 and note that they both tail off after lag 4. This is an ARMA(p,q) process of *max(p,q)* = 4 and therefore conclude that p,q both assume values between 0 and 4. We consider models of the combinations of the terms p, q, P, and Q identified above.

Figure 7

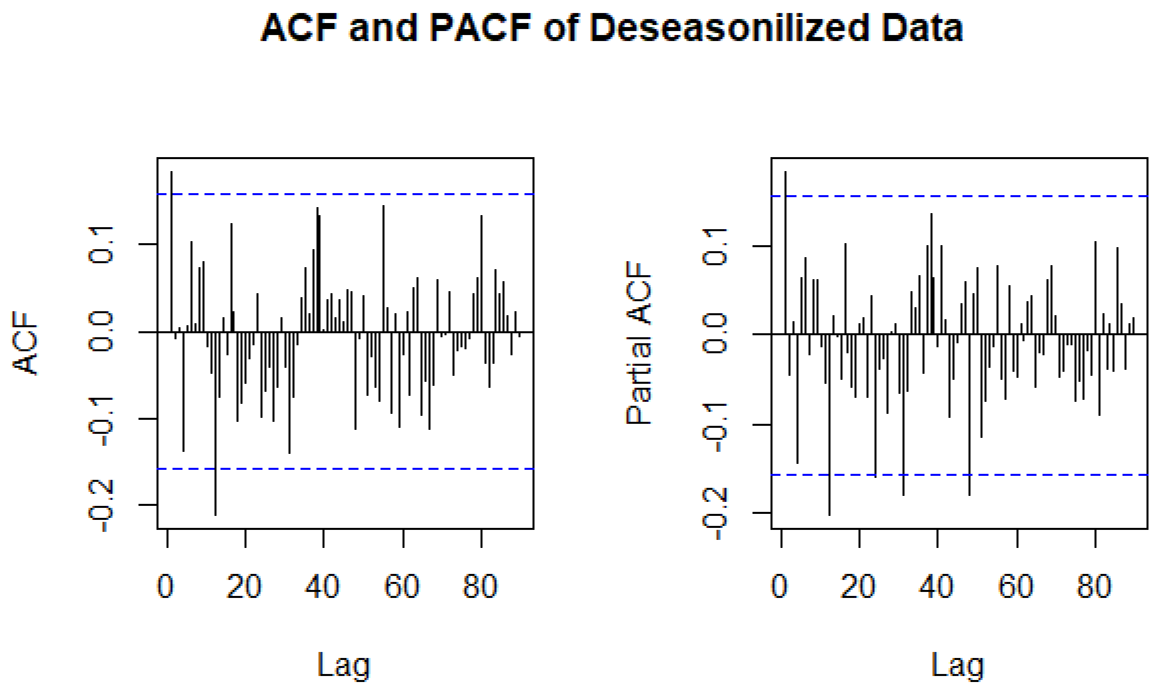
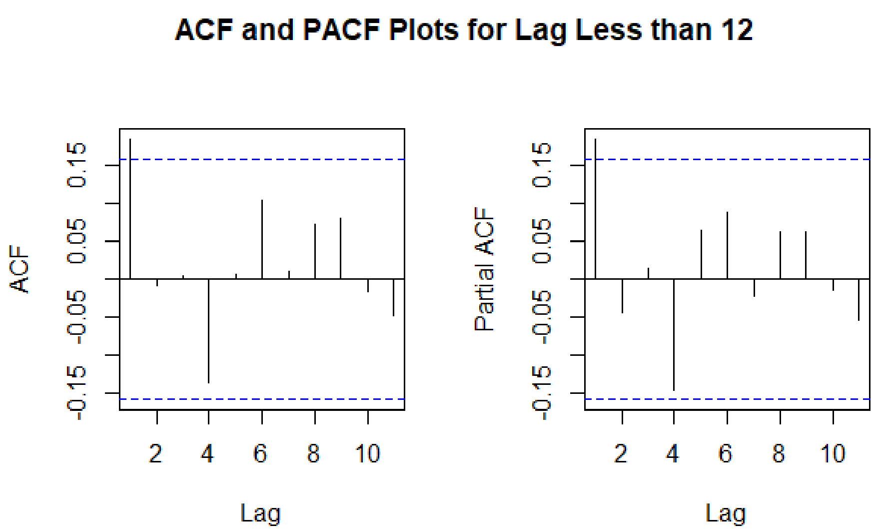


Figure 8



Selection of a Suitable Model

The AICc and BIC tables below help us choose the best models for our data. AICc identifies the two best models as those with p = 0 and q = 1 and BIC identifies the model with the smallest value as that with p = 0 and q = 0.

Table 1

	MA(0)	MA(1)	MA(2)
AR(0)	1.676326	1.657190	1.669567
AR(1)	1.658358	1.669415	1.681317
AR(2)	1.670059	1.680915	1.693797

AICc

Table 2

	MA(0)	MA(1)	MA(2)
AR(0)	0.6825830	0.6816039	0.7119863
AR(1)	0.6827720	0.7118351	0.7415873
AR(2)	0.7124787	0.7411861	0.7717618

BIC

The two possible models we consider are SARIMA (0, 1, 1) * $(0, 1, 1)_{12}$ and SARIMA (0, 1, 0) × $(0, 1, 1)_{12}$

Estimating the Model

We fit the models we have selected in the earlier stages and use maximum likelihood to estimate the coefficients for either of the model as in Table 4 below.

Table 3

	Model 1	Model 2
MA(1)	0.1818	--
SMA(1)	-0.3691	-0.3746

The two models we have are;

Model 1: SARIMA (0, 1, 1) * (0, 1, 1)

$X_t = (1 + 0.1818B) * (1 - 0.3691B^{12})*Z_t$; and

Model 2: SARIMA (0, 1, 1) x (0, 1, 1)

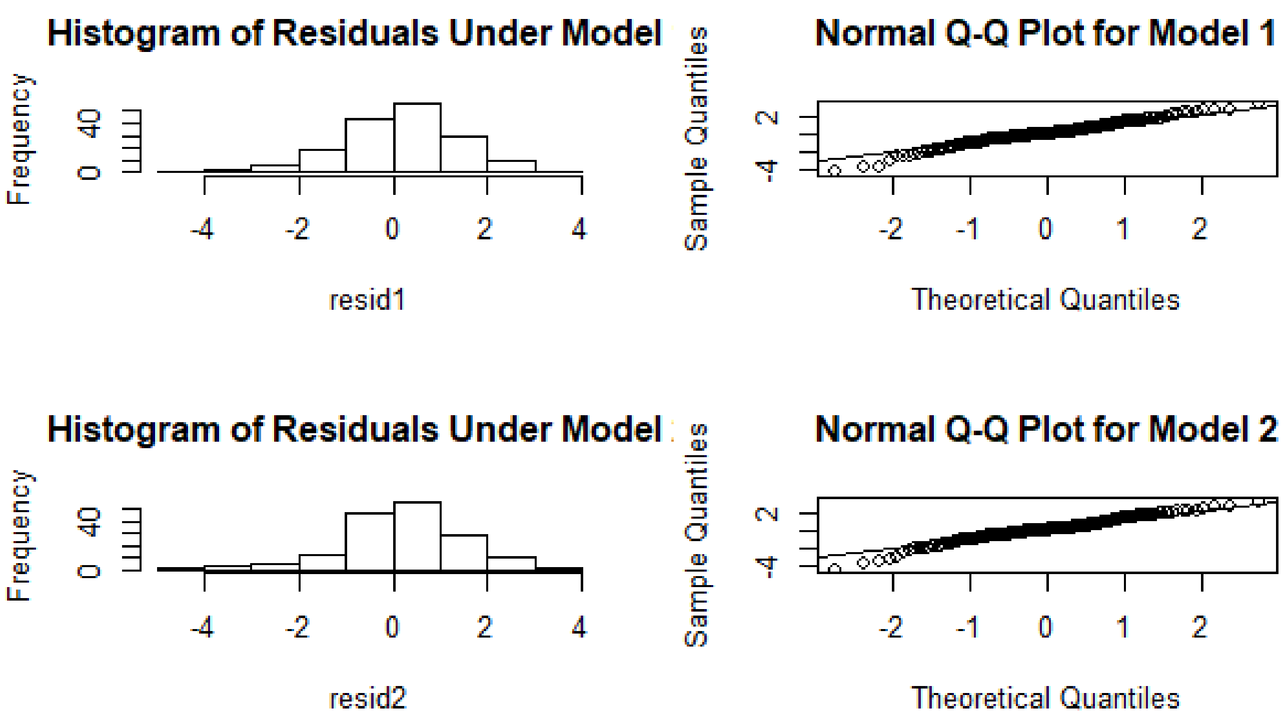
$X_t = (1 - 0.3746^{12})Z_t$

Diagnostics

Checking for Normality

This step helps us see if the residuals of the data we have are normally distributed, which is made possible with the help of a normal Q-Q plot as in the Figure 9 below.

Figure 9



Both models exude a roughly symmetrical normal curved histogram as seen in the figure above. Moreover, the points lie well on the line as in the plot.

The hypothesis we are testing is:

H_0: Residuals are normally distributed

H_1: Residuals are not normally distributed

A Shapiro Wilk Test evaluated at alpha 0.05 examines the assumption and gives us results as in the Table 4 below.

Table 4

	W.Statistics	**P-value**
Model 1	0.9823828	0.031510416

Model 2	0.9823828	0.004746233

Checking for Independence (Serial Correlation)

We perform a Ljung-Box and Box-Pierce tests at 0.05 alpha level to be sure that the residuals of our time series data do not serially correlate with lag observations. The hypothesis we are testing is:

H_0: Residuals serially uncorrelated with lagged values

H_1: Residuals not serially uncorrelated with lagged values

Table 5

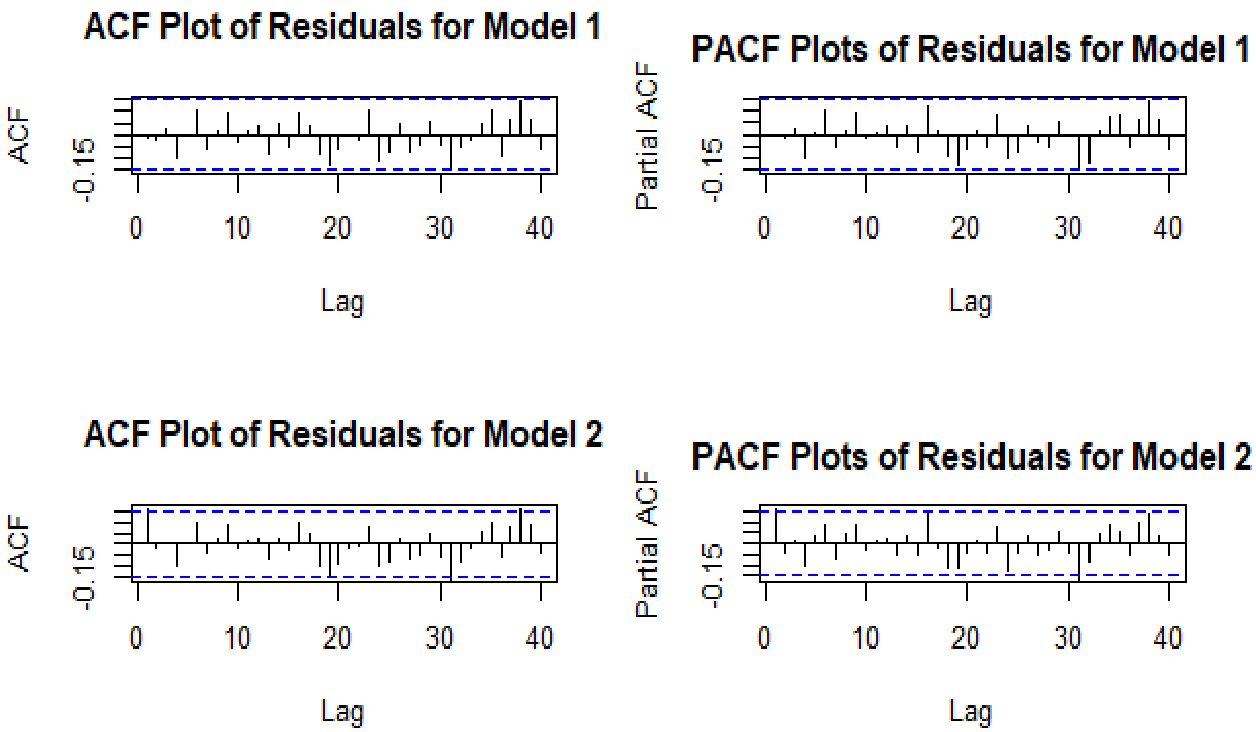
	P-Value (Model 1)	P-Value (Model 2)
Box-Pierce	0.774629	0.4646126
Ljung-Box	0.743154	0.4330845

The p values are all greater than 0.05 for both models so do not reject H_0 and the conclusion is that the residuals are indeed serially uncorrelated with their lag observations.

Checking for Constant Variance

We better check for constant variance by plotting ACF and PACF plots for both models as is in Figure 10 below.

Figure 10



From the plot we see that the values lies within the 95% confidence limits for white noise. In our case, most values lie within this limit, so both models are efficient in the prediction of future data points.

Spectral Analysis

Plotting Periodograms for the Time Series

Figure 11 below shows a periodogram of the time series after removing the trend and seasonal components.

Figure 11

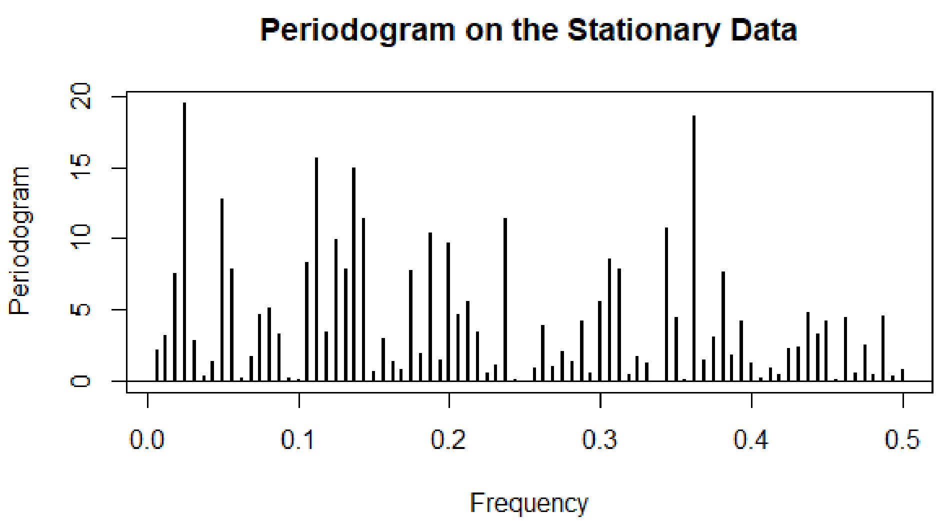
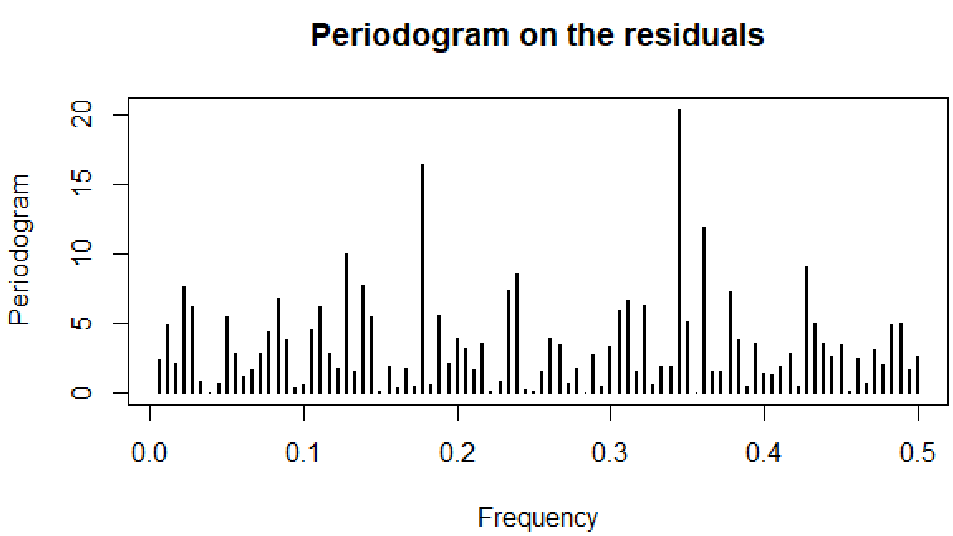


Figure 12



The plot shows relative importance of frequencies and may help explain the seasonality of the time series.

$$x_t = \mu + \sum_{j=1}^{k} (A_j \cos 2\pi v_j t) + (B_j \sin 2\pi v_j t) \text{ , } where\ v = frequency$$

The four most significant frequencies are v_1 = 0.12903226, v_2 = 0.02580645, v_3 = 0.36129032 and v_4 = 0.05161290. We then use a linear regression model and find that our constant, $\mu = 0.08645161$, and our A_js and B_js are as indicated in Table 6.

Table 6

j	Aj	Bj
1	0.19528095	0.50167317
2	0.36383655	0.32420953
3	-0.11650238	0.44772310
4	-0.34090319	-0.17457664

Our model may then be summarized as:

$$\begin{aligned} x_t \approx\ & 0.08645161 + 0.19528095 \cos(2\pi * 0.12903226\text{t}) \\ & + 0.50167317 \sin(2\pi * 0.12903226\text{t}) + 0.36383655 \cos(2\pi * 0.02580645) \\ & + 0.32420953 \sin(2\pi * 0.02580645) - 0.11650238 \cos(2\pi * 0.36129032\ ) \\ & + 0.44772310 \sin(2\pi * 0.36129032\ ) - 0.34090319 \cos(2\pi * 0.05161290) \\ & - 0.17457664 \sin(2\pi * 0.05161290) \end{aligned}$$

X_t is the variable in our time series data which has had the components of trend and seasonality removed.

Figure 13

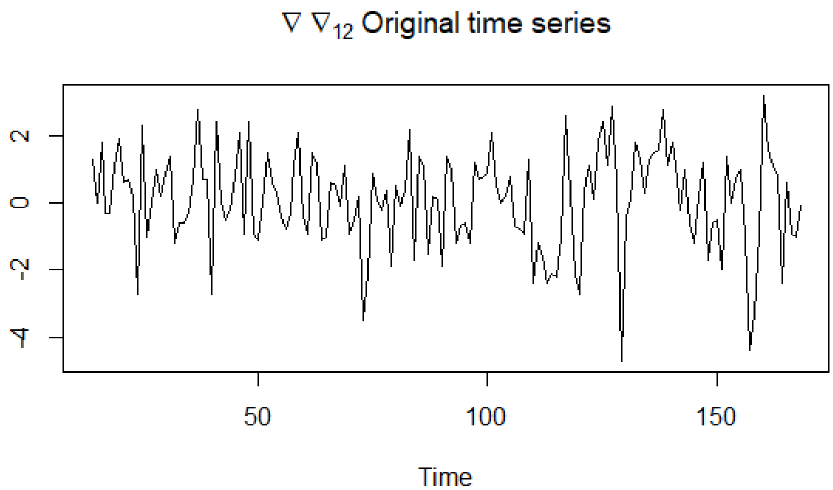
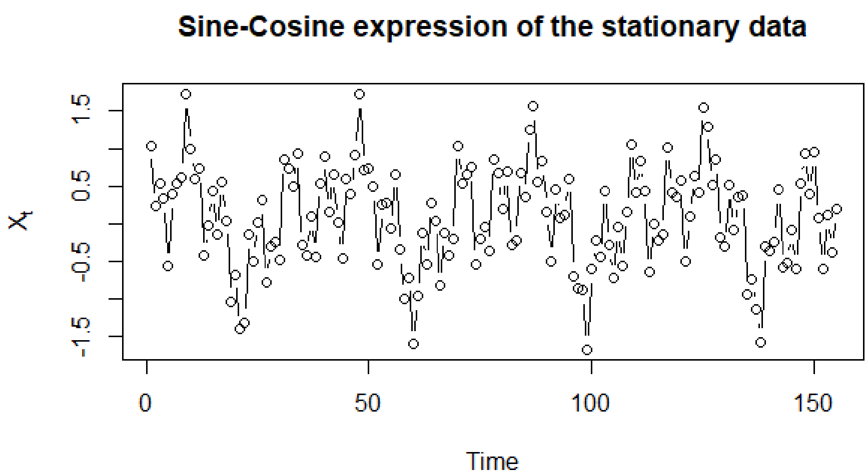


Figure 14



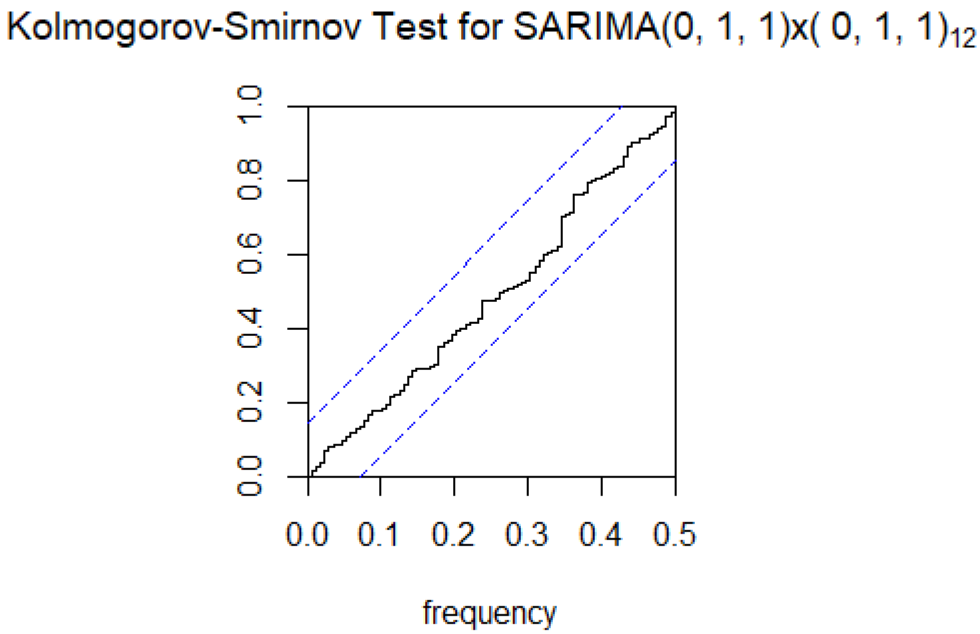
Fisher's Test

Fisher's test helps us see if there exist hidden periodicities in the data whose frequencies are unspecified. The calculated value is 0.1523968 > 0.05, an indication that the model does not have any hidden periodicities with unspecified frequencies.

Kolmogorov-Smirnov Test

This test is done on the residuals of our selected model to see if there exist Gaussian white noise.

Figure 15



None of the plotted values in Figure 11 above lies outside of the limits because they are white noise.

Forecasting using the Model

Forecasting was the main objective of developing a time series model in our case, so we use the selected model to get future values for the months of November 1978 through August 1979 on a monthly interval. Figures 16 and 17 below show the forecast values as red circles and the blue lines indicate the 95% boundaries for the values.

Figure 16

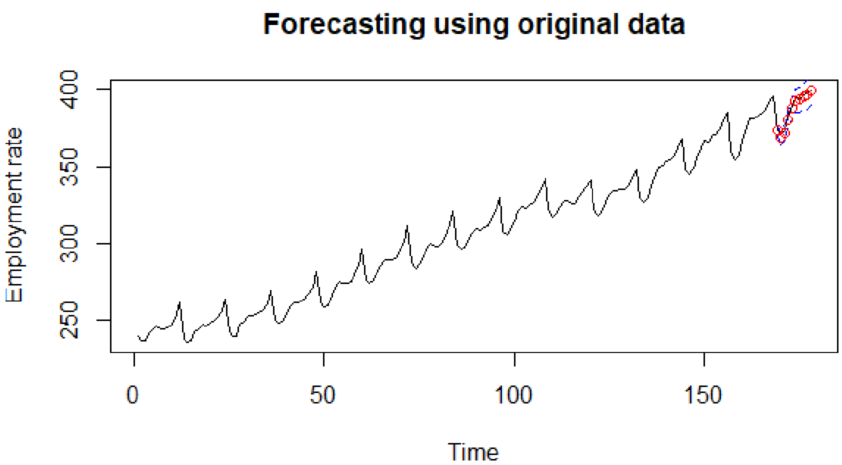
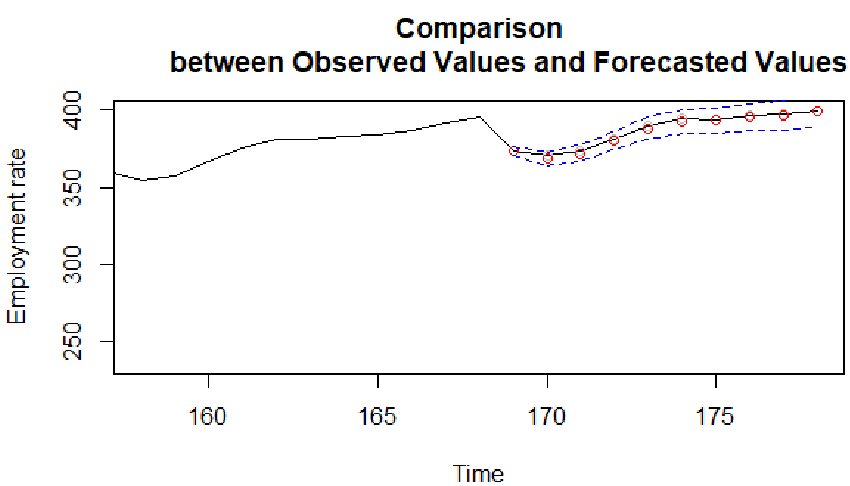


Figure 17



The general outlook of the curve of the forecast values is similar to that of the original data. In fact, the observed values shown alongside the forecast values are not far from the forecast plot points.

`

Conclusion

Employment and availability of employment opportunities are very vital for the growth of any economy. Employment data available to government agencies is a powerful tool for future planning as far as growth of employment is concerned, it enables policy and decision makers to make informed decisions and better policies. The main goal was to model a time series for the monthly employment data of Wisconsin for the months between January 1961 and October 1978 then forecast future values of the time series. The data showed elements of trend, seasonality and randomness as seen in the plot of the initial data as a time series. Employment levels peaked in December and slumped in the months of February and March, this replicated for all the years covered by the data. There was an upward trend in the data that we later removed by differencing at lag = 1 to help us further decompose the dataset to help us build a better model.

Removing the seasonal and trend components helped us build models from the resulting stationary time series dataset, from which we selected one model that best fit our data. The model we selected is a SARIMA (0, 1, 1) * (0, 1, 1)$_{12}$:

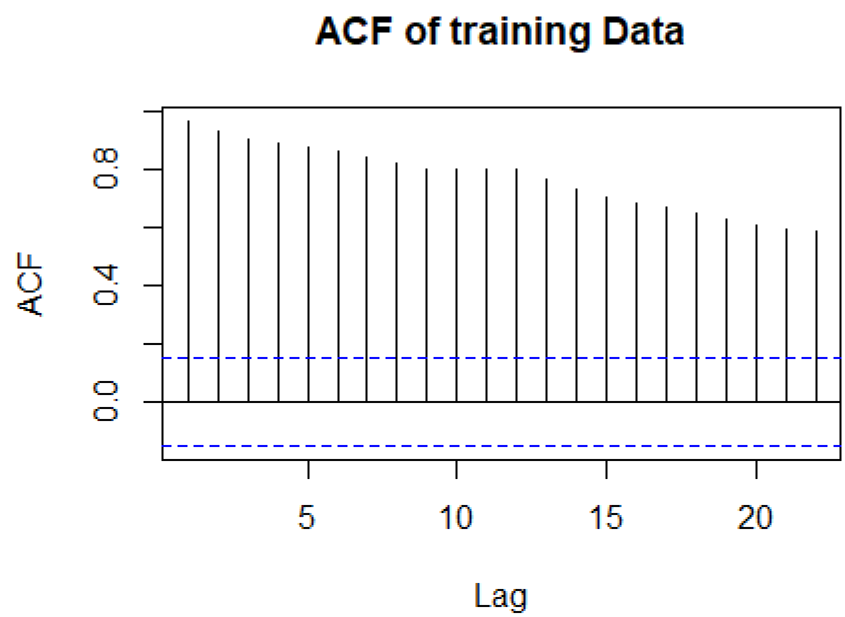
$$X_t = (1 + 0.1818B) * (1 - 0.3691B^{12})Z_t$$

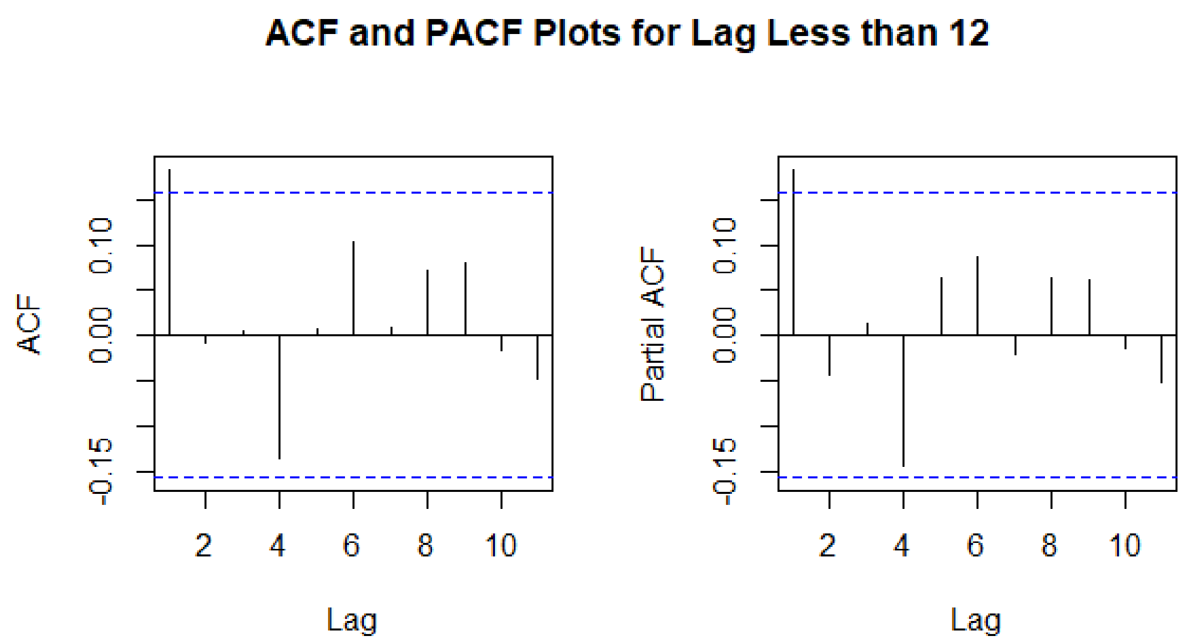
The forecast values for this model fall well within the 95% confidence interval and close to the true values when compared. This is proof that the model is indeed feasible.

References

Wisconsin employment time series, Jan. 1961 - Oct. 1975. (n.d.). Retrieved from https://datamarket.com/data/set/22l8/wisconsin-employment-time-series-trade-jan-1961-oct-1975#!ds=22l8&display=line

Appendix


ACF of training Data
ACF
0.0
0.4
0.8
5
10
15
20
Lag


ACF and PACF Plots for Lag Less than 12
ACF
-0.15
0.00
0.10
2
4
6
8
10
Lag
Partial ACF
-0.15
0.00
0.10
2
4
6
8
10
Lag

```
Call:
lm(formula = employmentdiff12diff1 ~ a1 + a2 + a3 + a4 + a5 + a6 + a7 + a8)

Residuals:
    Min      1Q  Median      3Q     Max
-4.5626 -0.7704  0.1579  0.7626  3.2903

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.08645    0.10620   0.814  0.41693
a1           0.19528    0.15019   1.300  0.19556
a2           0.50167    0.15019   3.340  0.00106 **
a3           0.36384    0.15019   2.423  0.01664 *
a4           0.32421    0.15019   2.159  0.03251 *
a5          -0.11650    0.15019  -0.776  0.43917
a6           0.44772    0.15019   2.981  0.00337 **
a7          -0.34090    0.15019  -2.270  0.02468 *
a8          -0.17458    0.15019  -1.162  0.24697
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.322 on 146 degrees of freedom
Multiple R-squared:  0.2124,	Adjusted R-squared:  0.1692
F-statistic: 4.921 on 8 and 146 DF,  p-value: 2.118e-05
```

R Codes

```
setwd("C:/Users/Bisiny/Desktop")
library(astsa)
library(tseries)
library(MASS)
library(forecast)
library(TSA)
library(GeneCycle)
library(ggplot2)

#plot the original data
employment = read.csv("C:/Users/Bisiny/Desktop/tsdata.csv")
employmentts = ts(employment[ , 2 ] , frequency=12)
employmentts
```

```
plot(employmentts, xlab="Time (in years)" , ylab="Employment Rate" , main="Wisconsin
employment time series for Jan. 1961 - OCt. 1978 " ) #seasonality, s=12

#seasonal plot
seasonplot(employmentts, 12, col=rainbow(7), year.labels=FALSE, main=" Seasonal Plot " )

#decomposed plot
decomposed = decompose(employmentts)
autoplot(decomposed , main="Decomposition Plot") +
  theme(axis.text.y = element_text(size =6), text = element_text(size = 12))+
  xlab ( "Time (in years) " )

#to check if stabilizing variance is helpful
transformed = boxcox(employmentts~as.numeric(1:length(employmentts)))
#the Box-Cox plot contains 0 in the 95% confidence interval range
#we do not perform the Box-Cox transformation

#make a training set by dropping the last 10 observations
training = ts(employmentts[1:(length(employmentts)-10)])
plot(training, xlab = "Time", ylab = "", main = "Employment training set") #plot the training data
set
acf(training, main = "ACF of training Data")

#removing the seasonal component
employmentdiff12 = diff(training, lag = 12)
plot(employmentdiff12, xlab = "Time", ylab = "", main = expression(nabla[12]~"Original time
series"))
title(ylab = expression(nabla[12]~X[t]), line = 2)
abline(lm(employmentdiff12~as.numeric(1:length(employmentdiff12))))
```

```
var(employmentdiff12) #variance = 25.88961

#removing the trend component
employmentdiff12diff1 = diff(employmentdiff12, lag = 1)
plot(employmentdiff12diff1, xlab = "Time", ylab = "", main =
expression(nabla~nabla[12]~"Original time series"))
abline(lm(employmentdiff12diff1~as.numeric(1:length(employmentdiff12diff1))))
var(employmentdiff12diff1) #variance = 2.104166
title(ylab = expression(nabla~nabla[12]~X[t]), line = 2)

#difference again
employmentdiff12diff2 = diff(employmentdiff12diff1, lag = 1)
var(employmentdiff12diff2) #variance = 3.447368
#variance increased from 2.104166 to 3.447368
#further differencing will be overdifferencing
#Therefore, our D = 1 and d = 1

#checking to see if X[t] is stationary or not
adf.test(employmentdiff12diff1)

#Model identification
#identifying P, Q
op = par(mfrow=c(1,2))
acf(employmentdiff12diff1, lag.max = 60, main = "")
pacf(employmentdiff12diff1, lag.max = 60, main ="")
title(main = "ACF and PACF of deseasonilized time series", outer = TRUE, line = -1)
par(op)
#check acf and pacf at lag = 12, 24, 36, 48, ...
 #Q=1
```

```
  #P=0

#identify p,q
op <- par(mfrow=c(1,2))
acf(employmentdiff12diff1, lag.max = 11, main="")
pacf(employmentdiff12diff1, lag.max = 11, main="")
title(main = "ACF and PACF for lags less than 12", outer = TRUE, line = -1)
par(op)

#at lag 1,2,3,...,11
#q = 0 or 2
#p = 0 or 2
#test combinations of p & q in 0 to 2

#model selection by AIC
AICc = numeric()
for (p in 0:2) {
  for (q in 0:2) {
    AICc = c(AICc, sarima(training, p, 1, q, 0, 1, 1, 12, details = FALSE)$AICc)
  }
}
AICc = matrix(AICc,nrow = 3, byrow = TRUE)
rownames(AICc)=c("p(0)","p(1)","p(2)")
colnames(AICc)=c("q(0)","q(1)","q(2)")
AICc
AICc=data.frame(AICc)
aic = setNames(AICc,c("q(0)","q(1)","q(2)"))
#p(0),q(1) smallest
#p(1),q(0) second smallest
```

```
BIC = numeric()
for (p in 0:2) {
  for (q in 0:2) {
    BIC = c(BIC, sarima(training, p, 1, q, 0, 1, 1, 12, details = FALSE)$BIC)
  }
}
BIC = matrix(BIC,nrow = 3, byrow = TRUE)
rownames(BIC)=c("p(0)","p(1)","p(2)")
colnames(BIC)=c("q(0)","q(1)","q(2)")
BIC
BIC=data.frame(BIC)
bic = setNames(BIC,c("q(0)","q(1)","q(2)"))
#p(0),q(1) smallest
#p(0),q(0) second smallest

#Based on AICc and BIC , select two models
#Model 1, SARIMA (0,1,1)*(0,1,1)[12]
#Model 2, SARIMA (0,1,0)*(0,1,1)[12]

#Estimating and fitting using MLE
#MODEL 1 : p = 0 and q = 1
fita = arima(training, order = c(0,1,1), seasonal = list(order=c(0,1,1),period=12),method = "ML")
fita
#MODEL 2
fitb = arima(training, order = c(0,1,0), seasonal = list(order=c(0,1,1),period=12),method = "ML")
fitb

#Diagnostics
##Normality
```

```
resida = residuals(fita) #residuals for M1
residb = residuals(fitb) #residuals for M2
op = par(mfrow = c(2,2))
hist(resida, xlab = "Residuals", main="Histogram of Residuals of Model 1")
qqnorm(resida, main="Normal Q-Q Plot for Model 1")
qqline(resida)
hist(residb ,xlab = "Residuals", main="Histogram of Residuals for Model 2")
qqnorm(residb , main="Normal Q-Q Plot for Model 2")
qqline(residb)
par(op)

#Shapiro test for Model 1 and 2
shap = matrix(c(shapiro.test(resida)$statistic, shapiro.test(resida)$p.value,
shapiro.test(residb)$statistic, shapiro.test(residb)$p.value),nrow = 2, byrow = T)
rownames(shap) = c("Model 1", "Model 2")
colnames(shap) = c("W Statistic", "P-value")
shap = data.frame(shap)

#diagnostics for independence and correlation
b1 = Box.test(resida, lag = 12, type = "Box-Pierce", fitdf = 2)$p.value
b1 #0.774629>0.05
b2 = Box.test(resida, lag = 12, type = "Ljung-Box", fitdf = 2)$p.value
b2 #0.743154>0.05
b3 = Box.test(residb, lag = 12, type = "Box-Pierce", fitdf = 2)$p.value
b3 #0.4646126>0.05
b4 = Box.test(residb, lag = 12, type = "Ljung-Box", fitdf = 2)$p.value
b4 #0.4330845>0.05

#Box-Pierce and Ljung-Box testing
```

```
boxT = matrix(c(b1,b2,b3,b4), nrow = 2, byrow = FALSE)
rownames(boxT)=c("Box-Pierce", "Ljung-Box")
colnames(boxT)=c("Model1 P-Value", "Model2 P-Value")
par(mfrow=c (2, 2))

#ACF and PACF for residuals of model 1
acf(resida, main = "ACF Plot of Residuals for Model 1", lag.max = 50)
pacf(resida, main = "", lag.max = 50)
title(main = "PACF Plots of Residuals for Model 1", outer=FALSE, line =1)

#ACF and PACF of residuals for model 2
acf(residb, main = "ACF Plot of Residuals for Model 2", lag.max = 50)
pacf(residb, main = "", lag.max = 50)
title(main = "PACF Plots of Residuals for Model 2", outer=FALSE, line =1)
par(op)

#forecasting based on the selected model
pred.tr = predict(fit1, n.ahead = 10)
U.tr = pred.tr$pred + 2*pred.tr$se # upper bound for the C.I.
L.tr = pred.tr$pred - 2*pred.tr$se # lower bound for the C.I.

pred.orig = pred.tr$pred #predicting future values
U = U.tr #upper confidence boundary
L = L.tr #lower confidence boundary
#Plotting forecast values on the data set
employment2 = ts(employment[,2])
ts.plot(employment2, xlim = c(1, length(employment2)),
main = "Forecasting using original data", ylab = "Employment rate")
lines(U, col = "red", lty = "solid")
```

```
lines(L, col = "red", lty = "solid")
points((length(training)+1):(length(training)+10), pred.orig, col = "blue")

#zooming to have a closer look at the forecast data
ts.plot(employment2, xlim = c(length(employment2)-20, length(employment2)), main =
"Comparison
    between observed and forecast values", ylab = "Employment rate")
points((length(training)+1):(length(training)+10), employment2[169:178], col = "grey")
points((length(training)+1):(length(training)+10), pred.orig, col = "blue")
lines((length(training)+1):(length(training)+10), U, lty = 2, col = "red")
lines((length(training)+1):(length(training)+10), L, lty = 2, col = "red")

#Spectral Analysis
#Periodogram of the stationary data
TSA::periodogram(employmentdiff12diff1, main = "Periodogram of stationary time series")
abline(h=0)

#checking dominant frequencies and periodicities
p1 = periodogram(employmentdiff12diff1)
freqW = p1$freq[order(p1$spec, decreasing = TRUE)][1:4]
freqW #frequencies with top 6 highest densities


t = 1:155
w = 2*pi*t
a1 = cos(w*freqW[1])
a2 = sin(w*freqW[1])
a3 = cos(w*freqW[2])
a4 = sin(w*freqW[2])
```

```
a5 = cos(w*freqW[3])
a6 = sin(w*freqW[3])
a7 = cos(w*freqW[4])
a8 = sin(w*freqW[4])

z = lm(employmentdiff12diff1~a1+a2+a3+a4+a5+a6+a7+a8)
summary(z)
#sine-cosine plots
learn = lm(employmentdiff12diff1~a1+a2+a3+a4+a5+a6+a7+a8)$coeff
y = learn[1] + learn[2]*a1 + learn[3]*a2 + learn[4]*a3 + learn[5]*a4 + learn[6]*a5 + learn[7]*a6
+ learn[8]*a7 + learn[9]*a8
plot(t,y, type = "b",xlab = "Time", ylab = expression(X[t]), main = "Sine-Cosine plot of the
stationary data")
plot(employmentdiff12diff1, xlab = "Time", ylab = "", main =
expression(nabla~nabla[12]~"Original time series"))
title(ylab = expression(X[t]), line = 2)

#periodogram of residual data
TSA::periodogram(resida, main ="Periodogram of the residuals")

#Kolmogorov-Smirnov Test
cpgram(resida, main = expression("Kolmogorov-Smirnov Test for SARIMA(0, 1, 1)x( 0, 1, 1)"
[12]))

#Fisher's test
fisher.g.test(resida) #0.1523968>0.05 no hidden periodicities
```