Final Project

Avery Loftin

11/29/2018

```
library(forecast)
WiscEmp <- read.csv("datasets/WiscEmp.csv")
names(WiscEmp) <- c("Month", "Employment")
WiscEmp.ts <- ts(data = WiscEmp$Employment, start = c(1961, 1), frequency = 12)
```

This dataset, retrieved from datamarket.com, represents 10,000 employed persons in Wisconsin from 1961 to 1975.

```
cat("Summary Statistics:\n")
```

```
## Summary Statistics:
```

```
summary(WiscEmp.ts)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   235.7   262.2   308.4   307.6   341.8   399.4
```

```
cat("Mean:", mean(WiscEmp.ts), "\n")
```

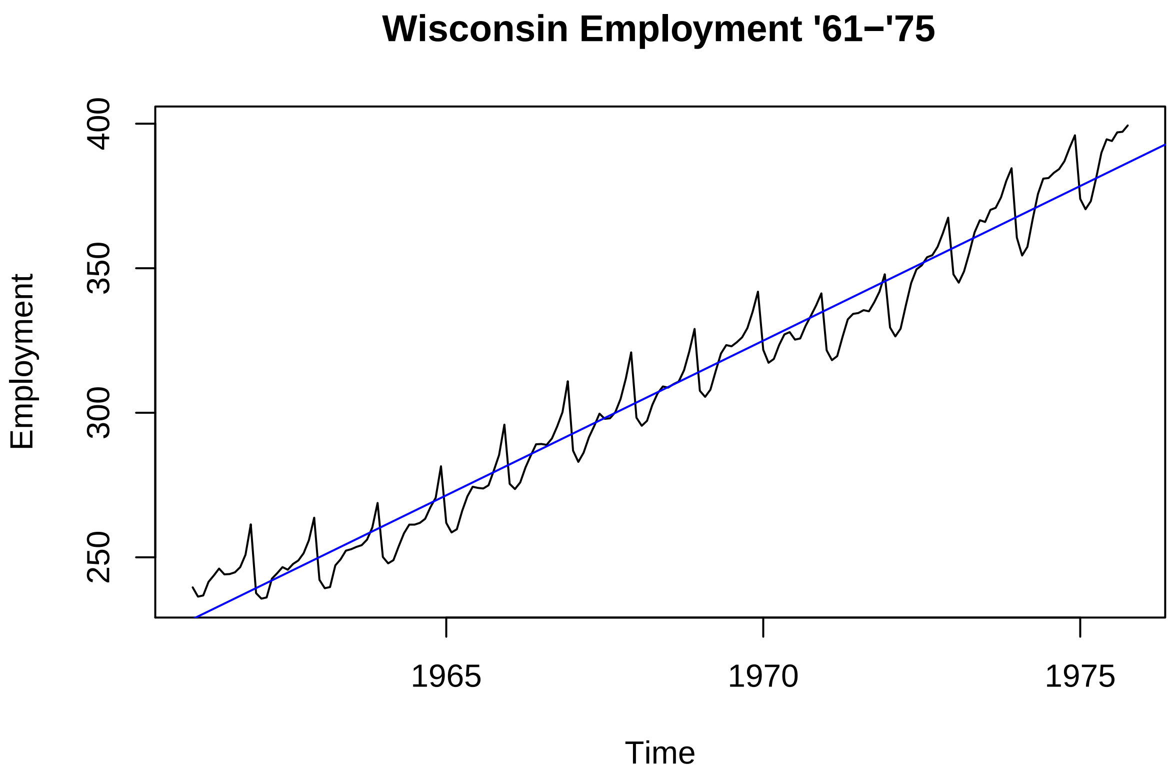
```
## Mean: 307.5584
```

```
cat("Standard Deviation:", sd(WiscEmp.ts), "\n")
```

```
## Standard Deviation: 46.76006
```

The mean employment from 1961 to 1975 was 3.076 million people, and the standard deviation was 467,606 people. These data have an upward trend due to the growth in overall population as well as annual seasonality peaking in December and hitting lows in Febuary. A linear regression model successfully captures the trend in these data, but fails to capture the seasonality.

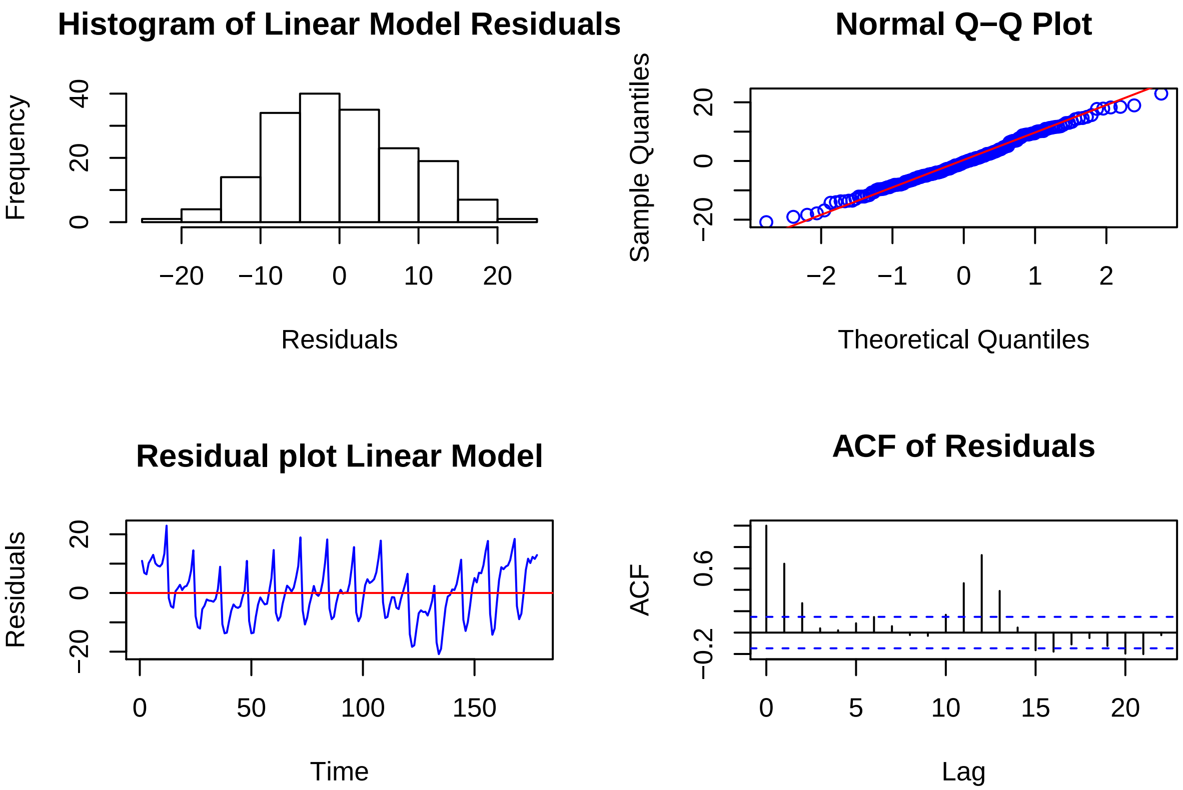
```
lm <- lm(WiscEmp.ts ~ time(WiscEmp.ts))

plot.ts(WiscEmp.ts, main="Wisconsin Employment '61-'75", ylab = "Employment")
abline(lm, col = "blue")
```



```r
Box.test(lm$residuals, type = "Ljung-Box")
```

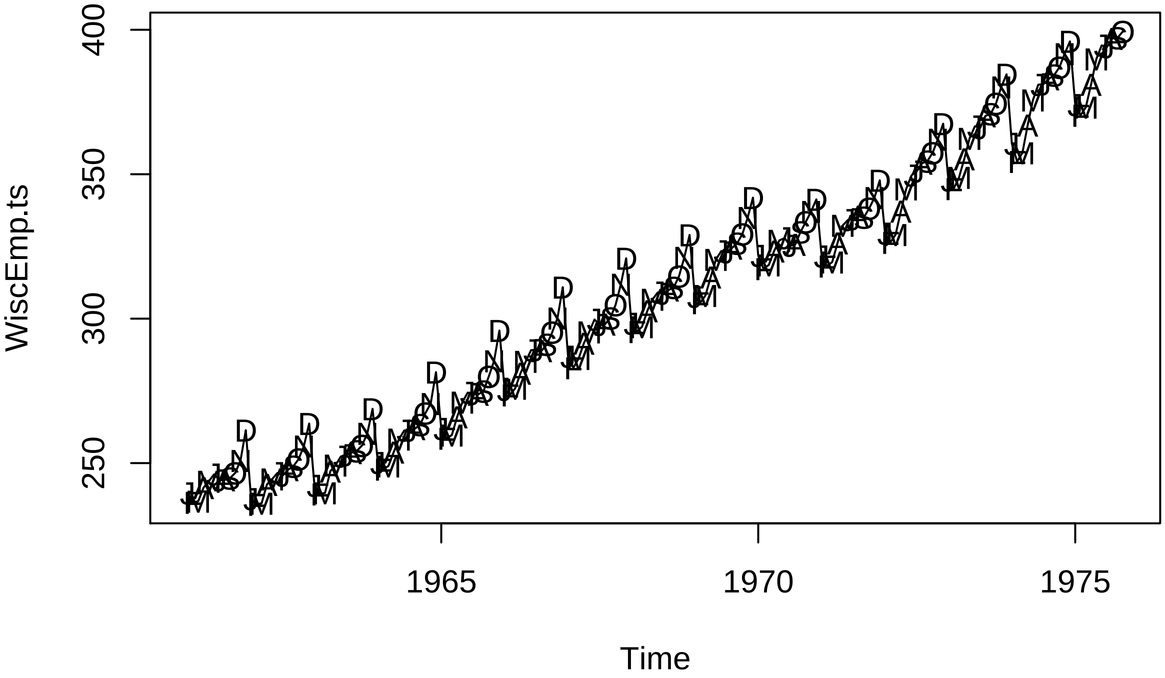
```
##
##  Box-Ljung test
##
## data:  lm$residuals
## X-squared = 75.133, df = 1, p-value < 2.2e-16
```

```r
par(mfrow=c(2,2))
hist(lm$residuals, main="Histogram of Linear Model Residuals", xlab="Residuals")
qqnorm(lm$residuals, col="blue")
qqline(lm$residuals, col="red")
plot.ts(lm$residuals, col="blue", main="Residual plot Linear Model", ylab = "Residuals")
abline(h=0, col="red")
acf(lm$residuals, main="ACF of Residuals")
```



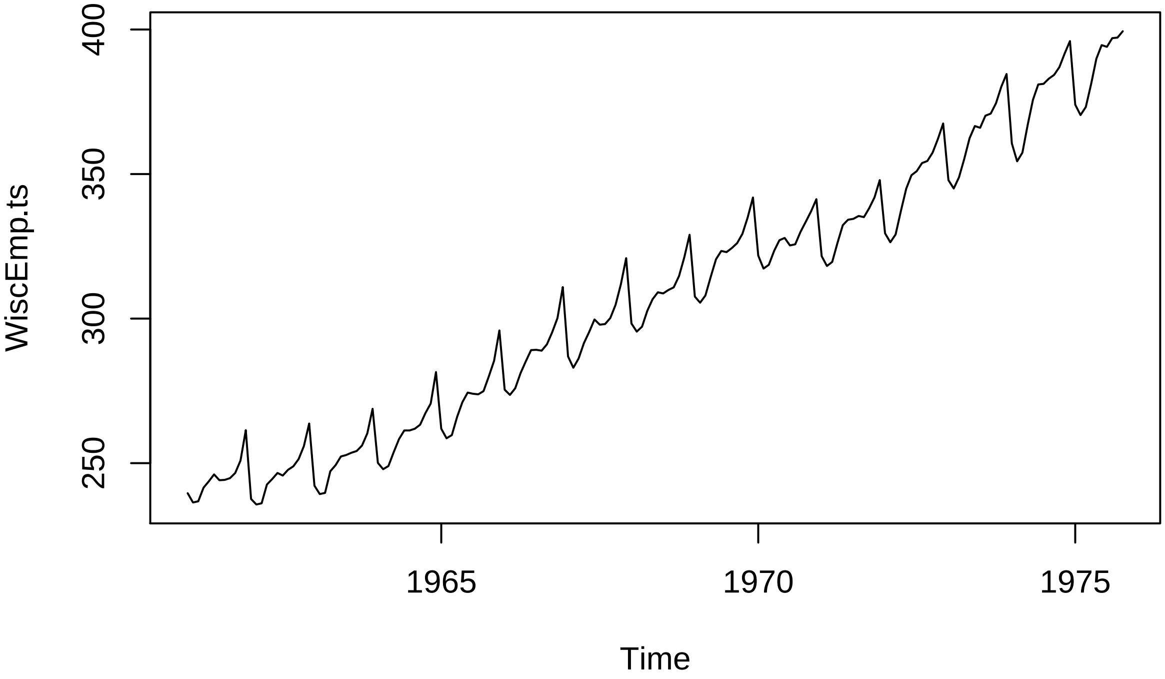
The Box-Ljung test resulted in a very low p-value of $2.2e^{-16}$, meaning that the model exhibits a lack of fit. The histogram looks relatively normal, however the QQ plot shows skewness towards the tails, and the residual plot depicts a clear trend rather than points randomly scattered about zero. Furthermore, the ACF of the residuals shows correlation and seasonality in the residuals. Therefore, the residuals fail to fit the assumption of normality.

```
plot.ts(WiscEmp.ts)
s <- c("J", "F", "M", "A", "M", "J", "J", "A", "S", "O", "N", "D")
points(y=WiscEmp.ts, x=time(WiscEmp.ts), pch=as.vector(s))
```

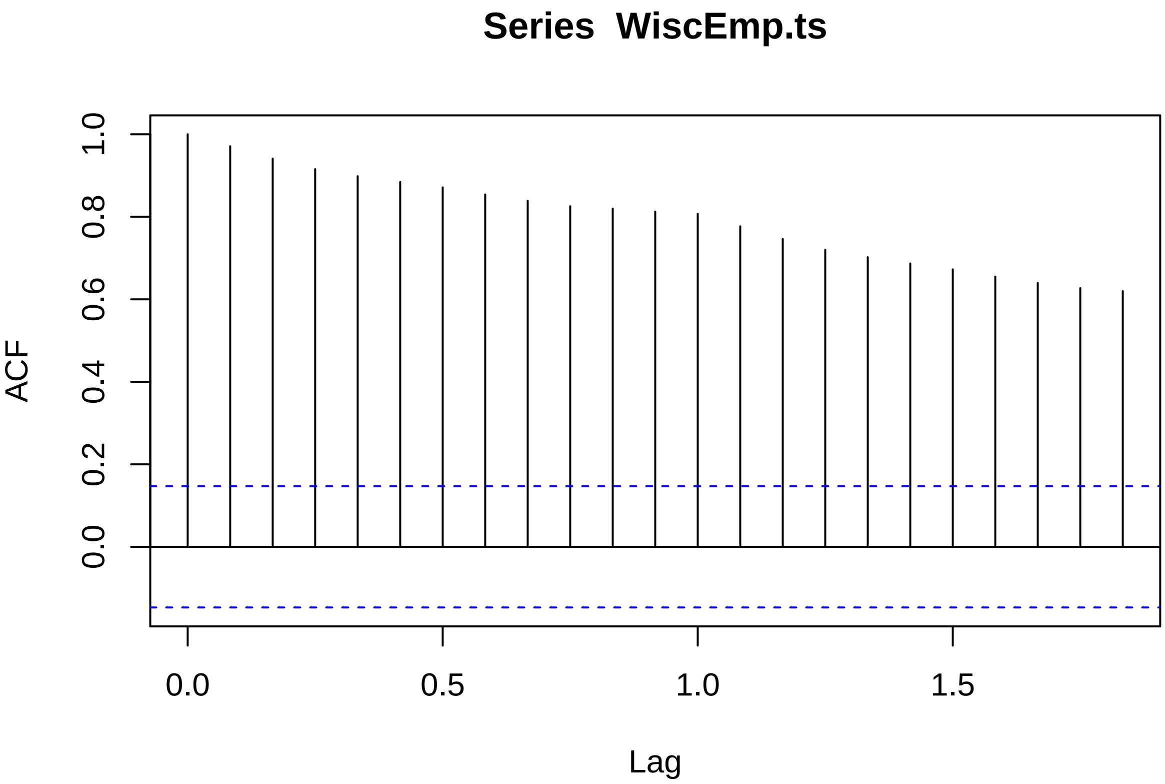


Using plotting symbols to highlight the months of the dataset, the seasonality becomes obvious with consistent peaks in december and troughs in February.

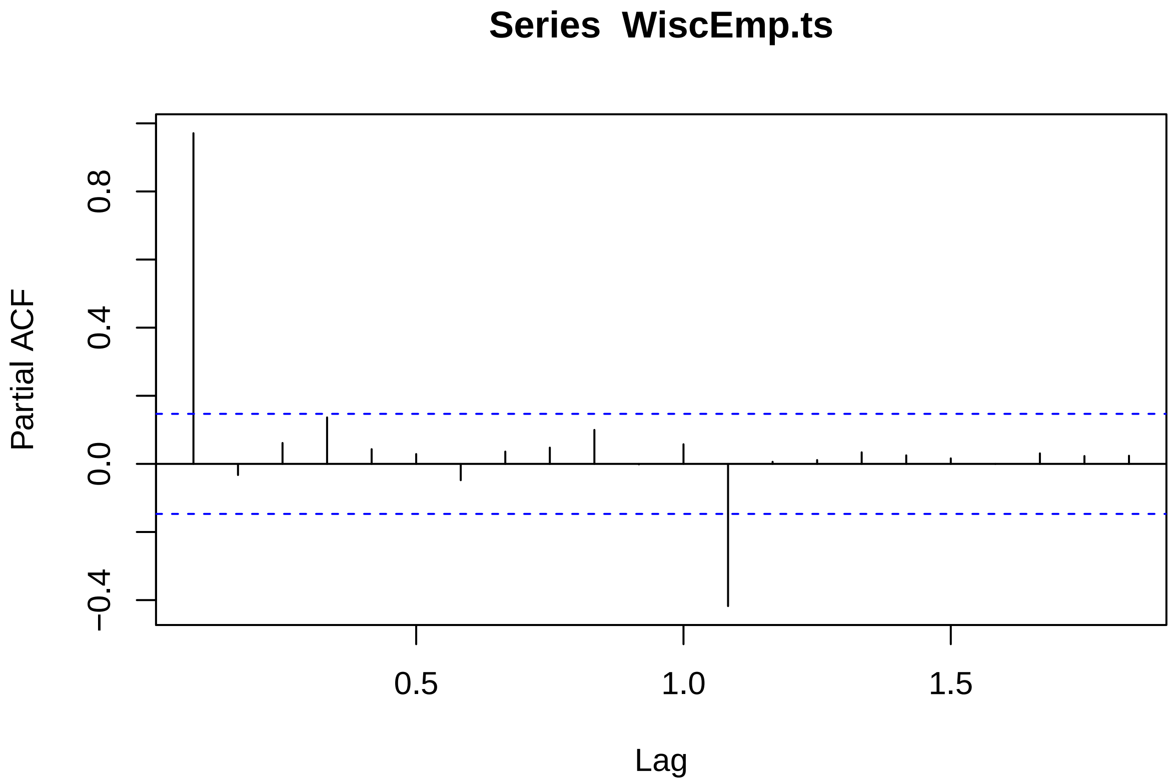
```
plot.ts(WiscEmp.ts)
```



```
acf(WiscEmp.ts)
```



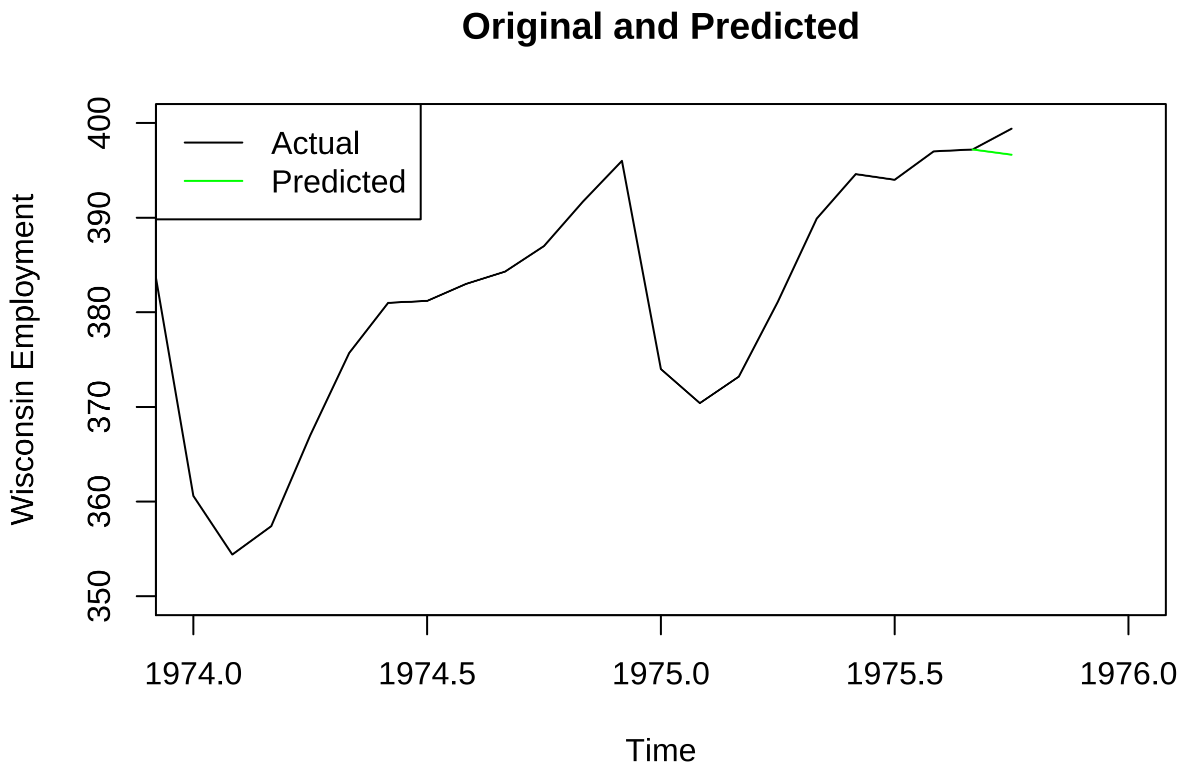
```
pacf(WiscEmp.ts)
```



The ACF of the dataset decays linearly, meaning the data are not stationary. The PACF is therefore not meaningful, since it's used when the ACF decays exponentially in order to know the number of significant lags in an AR model.

Next, an AR(1) model is fit to the dataset and is used to predict one month into the future. The resulting prediction has an error of 27,500 people from the actual value.

```
WiscEmp.train1 <- WiscEmp.ts[-length(WiscEmp.ts)]

AR1 <- arima(x = WiscEmp.train1, order = c(1,0,0))

lm.predict <- predict(AR1, 1)
lastAndPredict <- c(WiscEmp.train1[length(WiscEmp.train1)], lm.predict$pred)
lm.predict.full <- ts(data = lastAndPredict, start = c(1975, 9), frequency = 12)

plot.ts(WiscEmp.ts, ylim=c(350,400), xlim=c(1974, 1976), main="Original and Predicted", ylab="Wisconsin
lines(lm.predict.full, col="green")
legend("topleft", c("Actual", "Predicted"), col=c("black", "green"), lty=c(1,1))
```



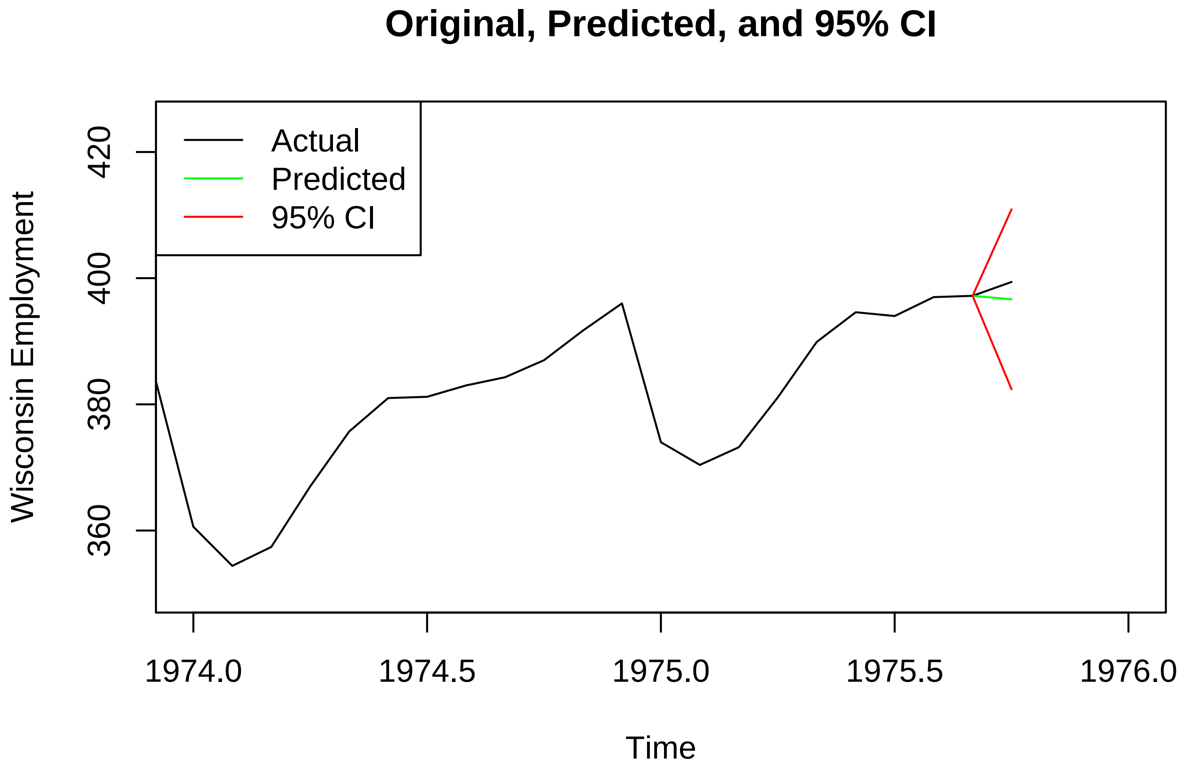
```
forecastError <- WiscEmp.ts[length(WiscEmp.ts)] - lm.predict$pred
cat("Forecast Error: ", forecastError)
```

```
## Forecast Error:  2.749787
```

Next, a 95% confidence interval is formed, and the actual value falls within this range.

```
predUC <- lm.predict$pred + 1.96 * lm.predict$se
predLC <- lm.predict$pred - 1.96 * lm.predict$se

predUC <- ts(c(WiscEmp.train1[length(WiscEmp.train1)], predUC), start=c(1975, 9), frequency=12)
predLC <- ts(c(WiscEmp.train1[length(WiscEmp.train1)], predLC), start=c(1975, 9), frequency=12)

plot.ts(WiscEmp.ts, main="Original, Predicted, and 95% CI", ylab="Wisconsin Employment", ylim=c(350,425
lines(lm.predict.full, col="green")
lines(predUC, col="red")
lines(predLC, col="red")
```

```r
legend("topleft", c("Actual", "Predicted", "95% CI"), col=c("black", "green", "red"), lty=c(1,1,1))
```



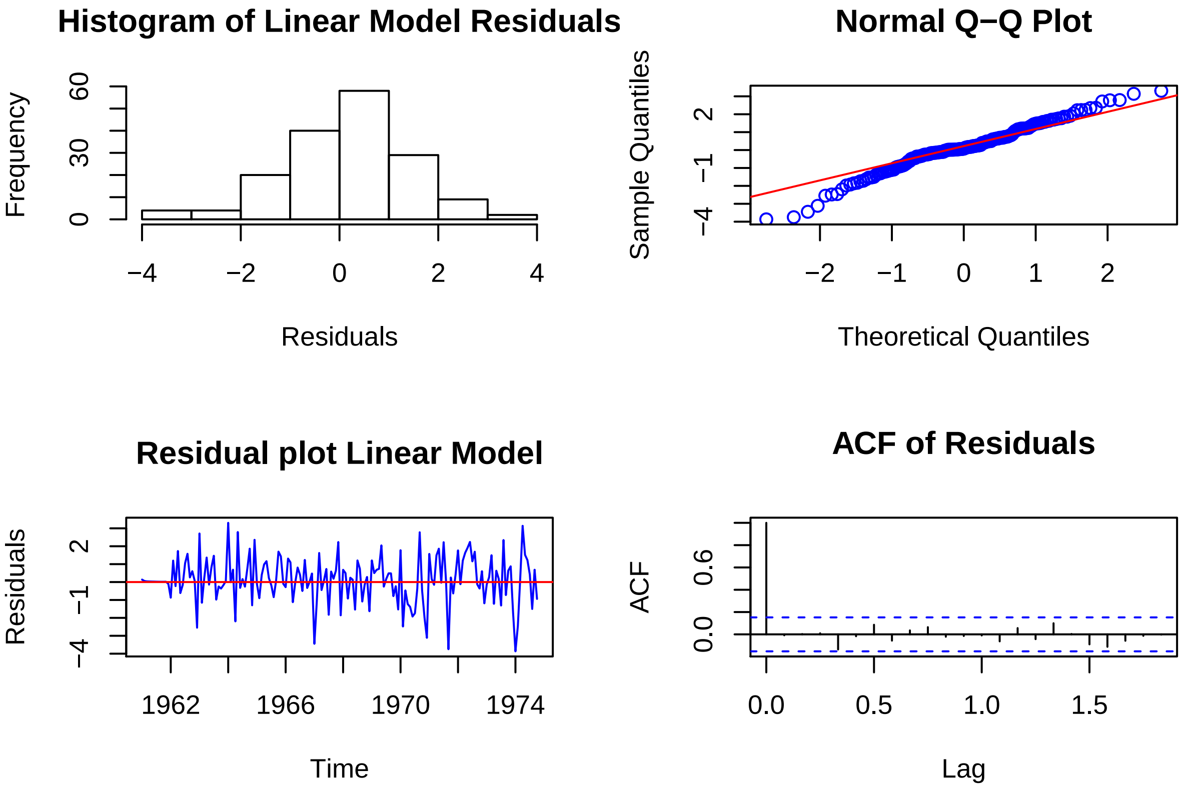
the auto.arima function finds a one time integrated MA(1) model best mimics the dataset. The error term is also integrated once and fit with an ARMA(1,1) model.

```r
WiscEmp.train <- WiscEmp.ts[1:(length(WiscEmp.ts)-12)]
WiscEmp.train <- ts(data = WiscEmp.train, start = c(1961, 1), frequency = 12)

WiscEmp.test <- WiscEmp.ts[(length(WiscEmp.ts)-11):length(WiscEmp.ts)]
WiscEmp.test <- ts(data = WiscEmp.test, start = c(1974, 11), frequency = 12)


model <- auto.arima(y = WiscEmp.train)
model
```

```
## Series: WiscEmp.train
## ARIMA(0,1,1)(1,1,1)[12]
##
## Coefficients:
##          ma1    sar1     sma1
##       0.1921  0.3868  -0.7256
## s.e.  0.0796  0.1565   0.1231
##
## sigma^2 estimated as 1.855:  log likelihood=-264.14
## AIC=536.27   AICc=536.54   BIC=548.39
```

```r
par(mfrow=c(2,2))
hist(model$residuals, main="Histogram of Linear Model Residuals", xlab="Residuals")
qqnorm(model$residuals, col="blue")
qqline(model$residuals, col="red")
plot.ts(model$residuals, col="blue", main="Residual plot Linear Model", ylab = "Residuals")
```

```r
abline(h=0, col="red")
acf(model$residuals, main="ACF of Residuals")
```



The residuals for the model show left skewness in the histogram. The QQ plot shows skewness on both tails and the residual plot does not depict consistent variance about zero. However, the ACF does not depict significant correlation between the residuals. Overall, the residuals fail to meet the normality assumptions, so further transformations are necessary.

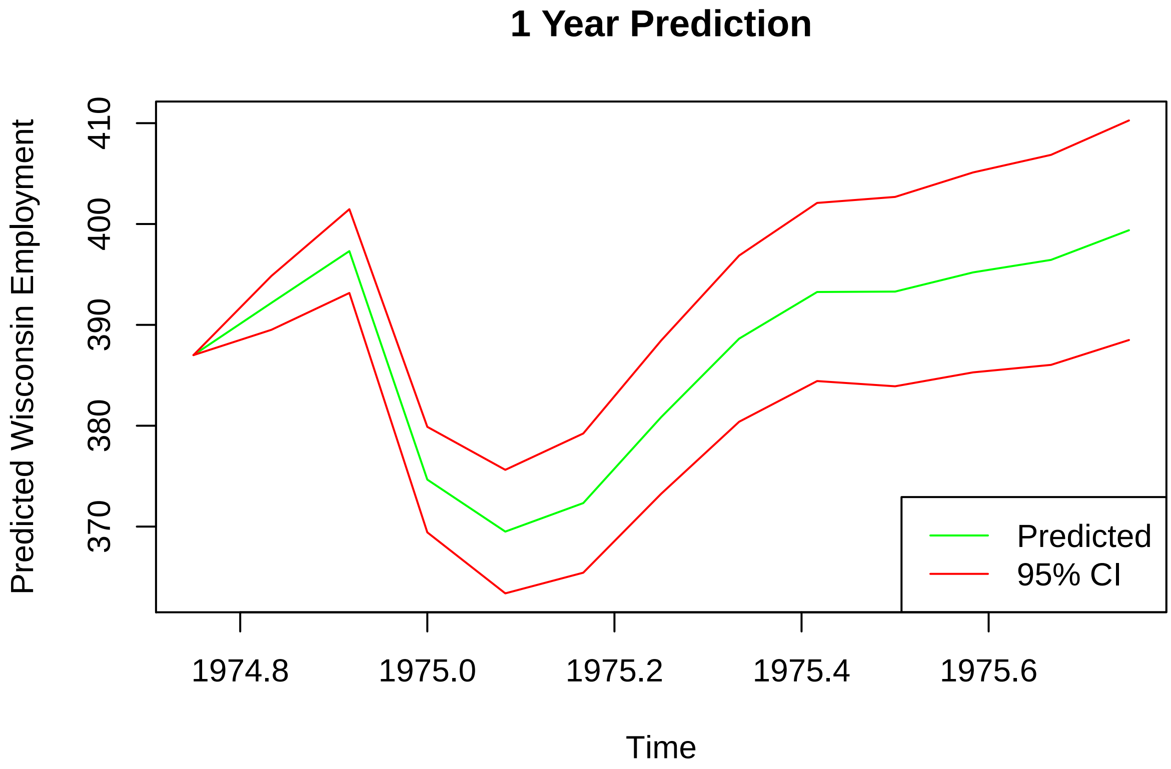
```r
Box.test(model$residuals, type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  model$residuals
## X-squared = 0.012489, df = 1, p-value = 0.911
```

The Box-Ljung test suggests that the model does not show a lack of fit, but the residuals do not appear to be drawn from a normal distribution. Next, this model is used to predict 1 year into the future and a 95% confidence interval is formed.

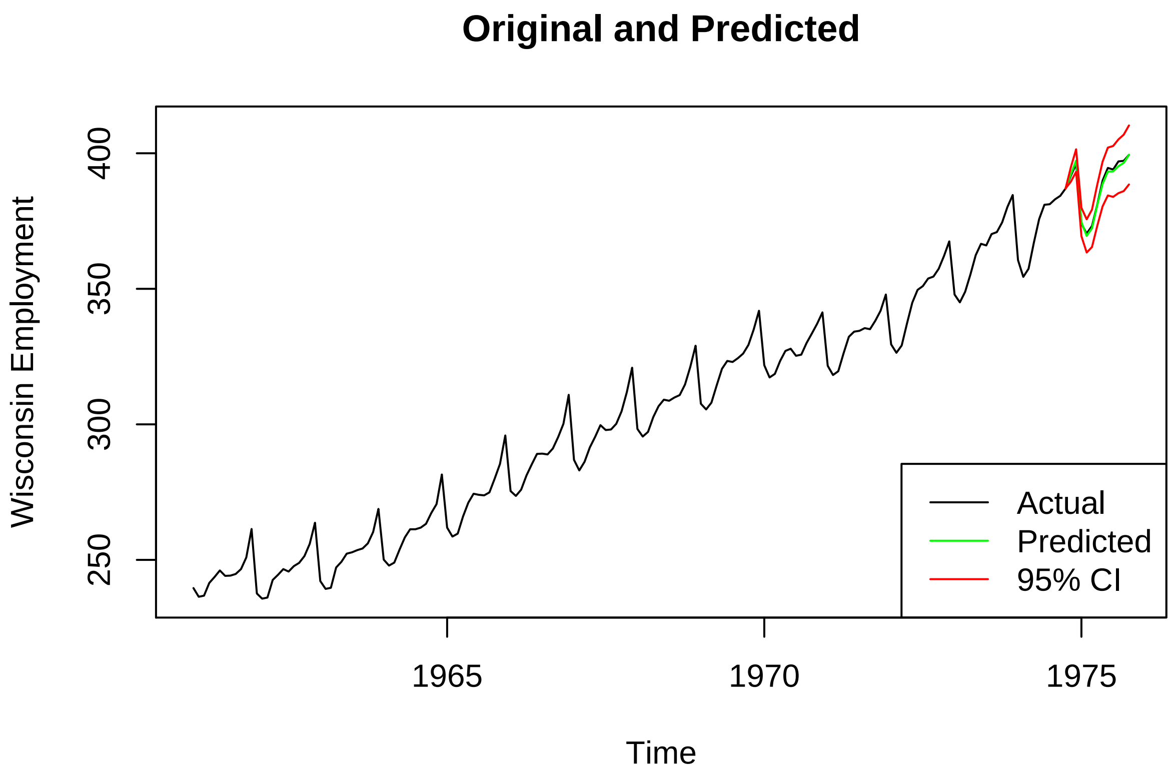
```r
model.predict <- predict(model, 12)

predUC <- model.predict$pred + 1.96 * model.predict$se
predLC <- model.predict$pred - 1.96 * model.predict$se

predUC <- ts(c(WiscEmp.train[length(WiscEmp.train)], predUC), start=c(1974, 10), frequency=12)
predLC <- ts(c(WiscEmp.train[length(WiscEmp.train)], predLC), start=c(1974, 10), frequency=12)

model.prediction <- c(WiscEmp.train[length(WiscEmp.train)], model.predict$pred)
model.prediction.ts <- ts(model.prediction, start=c(1974, 10), frequency = 12)
```

```
plot.ts(model.prediction.ts, main="1 Year Prediction", ylab="Predicted Wisconsin Employment", col="gree
lines(predUC, col="red")
lines(predLC, col="red")
legend("bottomright", c("Predicted", "95% CI"), col=c("green", "red"), lty=c(1,1))
```

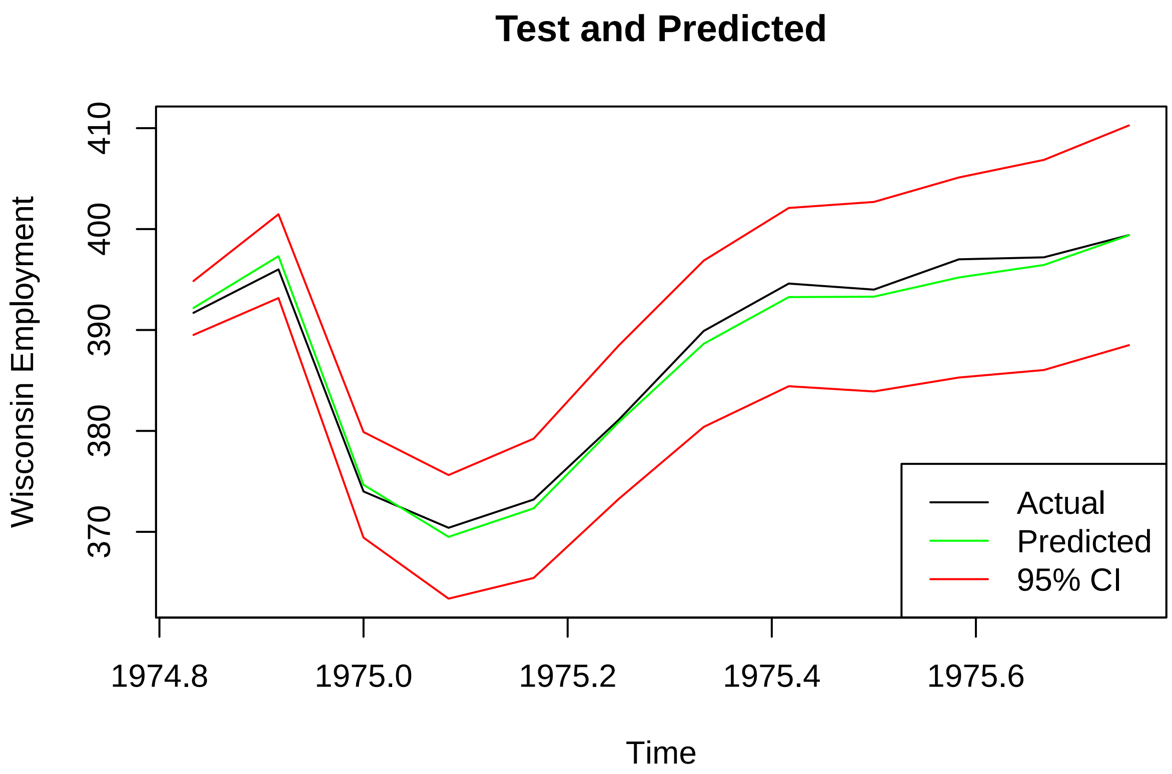


Plotting the prediction along with the actual values, The model appears to give incredibly accurate results.

```
plot.ts(WiscEmp.ts, main="Original and Predicted", ylab="Wisconsin Employment", ylim=c(min(WiscEmp.ts),
lines(model.prediction.ts, col="green")
lines(predUC, col="red")
lines(predLC, col="red")
legend("bottomright", c("Actual", "Predicted", "95% CI"), col=c("black", "green", "red"), lty=c(1,1,1))
```



```r
predUC2 <- ts(predUC[-1], start=c(1974, 11), frequency=12)
predLC2 <- ts(predLC[-1], start=c(1974, 11), frequency=12)

plot.ts(WiscEmp.test, main="Test and Predicted", ylab="Wisconsin Employment", ylim=c(min(predLC), max(pr
lines(model.predict$pred, col="green")
lines(predUC2, col="red")
lines(predLC2, col="red")
legend("bottomright", c("Actual", "Predicted", "95% CI"), col=c("black", "green", "red"), lty=c(1,1,1))
```



The mean squared error of only 9,000 people on the testing data further suggests that the model gives good predictions.

```
sqErr <- c()
for (i in 1:length(model.predict$pred))
{
  sqErr[i] <- (WiscEmp.test[i] - model.predict$pred[i])^2
}

MSE <- sum(sqErr)/length(sqErr)
MSE
```

```
## [1] 0.9757787
```

Using an ARIMA(0,1,1) model on the dataset along with an ARIMA(1,1,1) on the residuals provides very accurate predictions for the employment rate in Wisconsin.

ARIMA models can be used to successfully forecast time series datasets. In this example, the employment in Wisconsin was successfully modeled for a year with an error of only 9,000 people. This model could be employed to make predictions about future employment rates in the state and understand how employment changes over the course of a year.