CarSalesTimeSeries

Subham Moda

2024-06-12

```
library(TSA)
```

```
##
## Attaching package: 'TSA'
```

```
## The following objects are masked from 'package:stats':
##
##     acf, arima
```

```
## The following object is masked from 'package:utils':
##
##     tar
```

```
library(tseries)
```

```
## Warning: package 'tseries' was built under R version 4.2.3
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

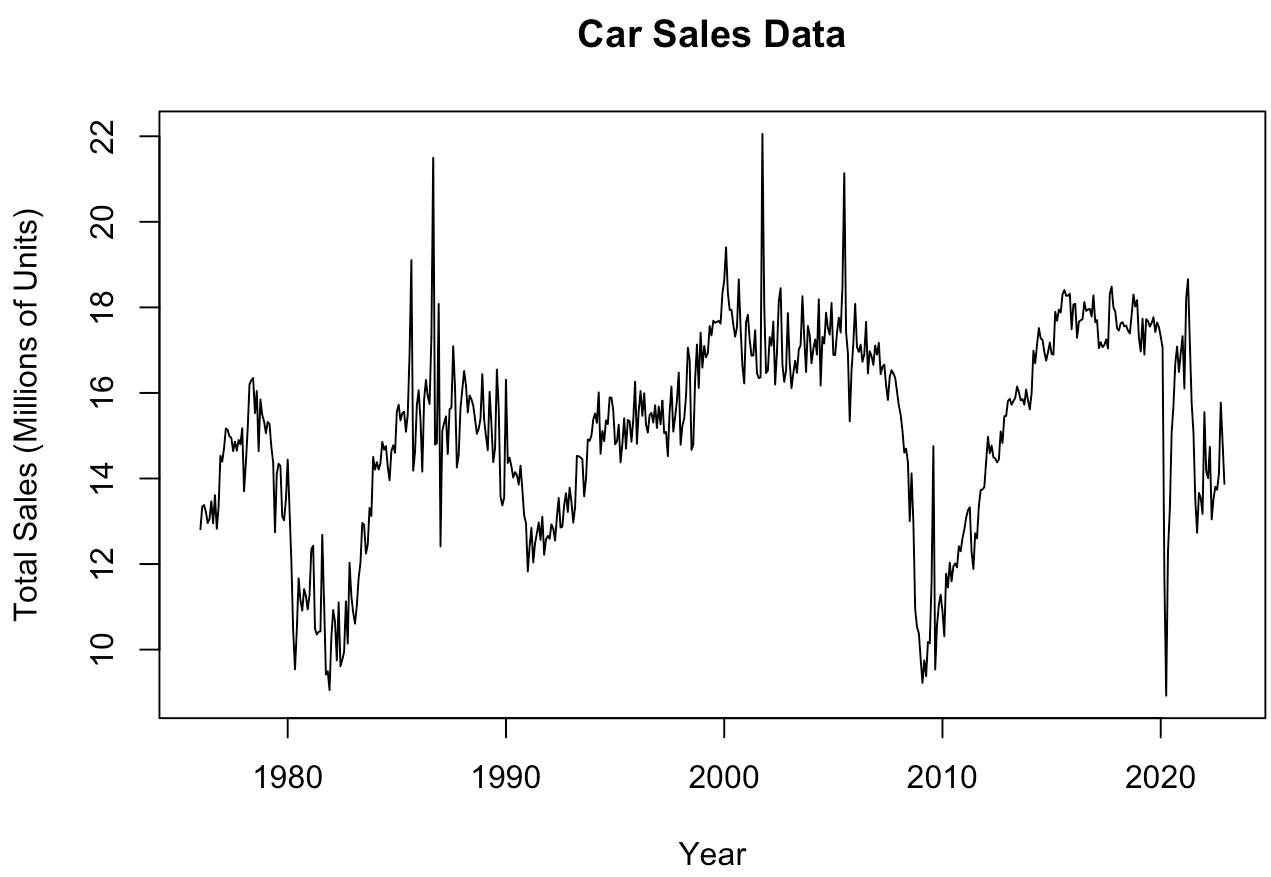
I've taken the cars sales data, consisting of monthly sales data of cars throughout United States. The data is dated from Jan 1976 to Dec 2022. I will try to fit a time series model and lastly predict the sales of cars for the next year, i.e. 2023.

```
ns_data <- read.csv('/Users/subhammoda/Documents/Projects/MA641_Project/TOTALSA.csv')
ns_data$TOTALSA = as.numeric(ns_data$TOTALSA)
ns_data$DATE = as.Date(ns_data$DATE)
ns_data <- ts(ns_data$TOTALSA[0:564], frequency = 12, start = c(1976, 1))
head(ns_data)
```

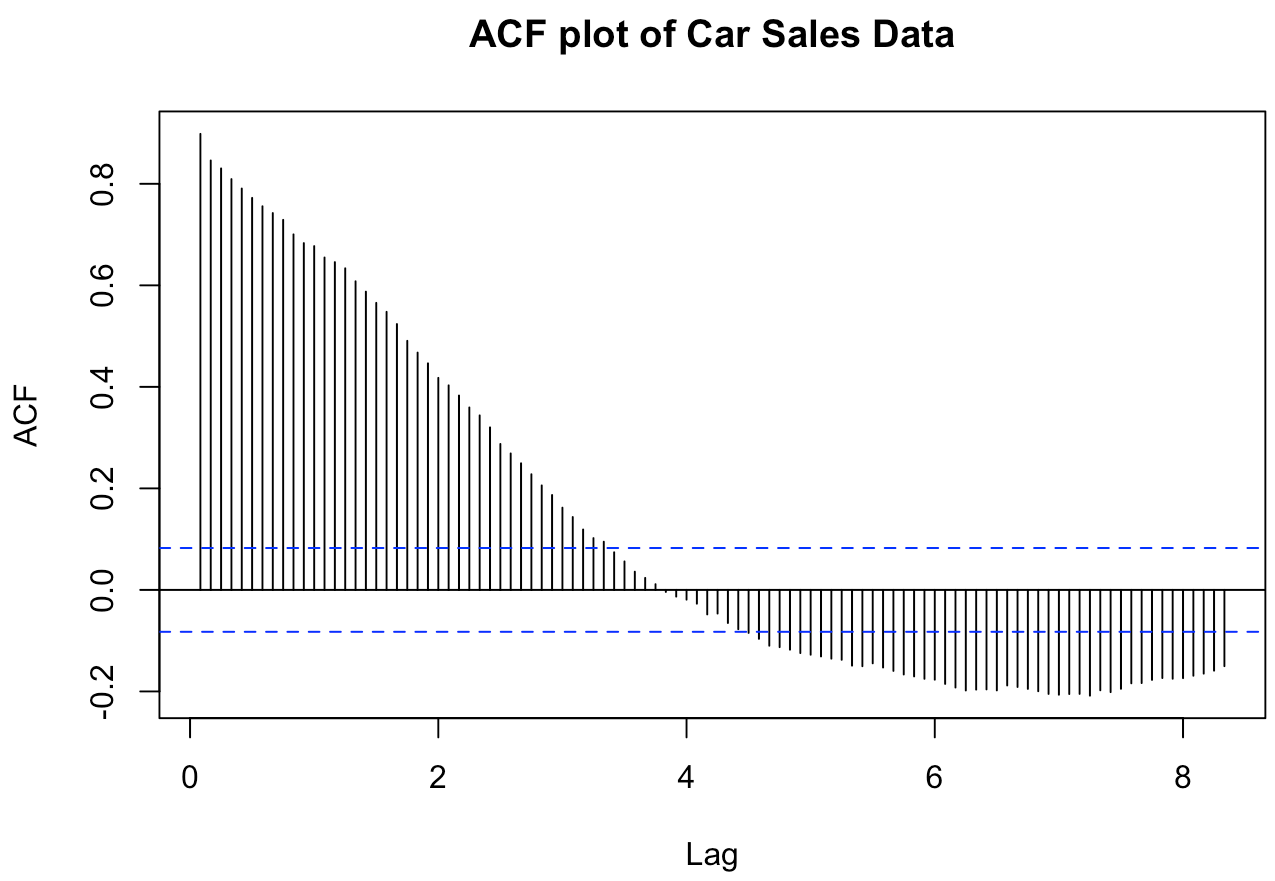
```
## [1] 12.814 13.340 13.378 13.223 12.962 13.051
```

The plot below shows the actual data.

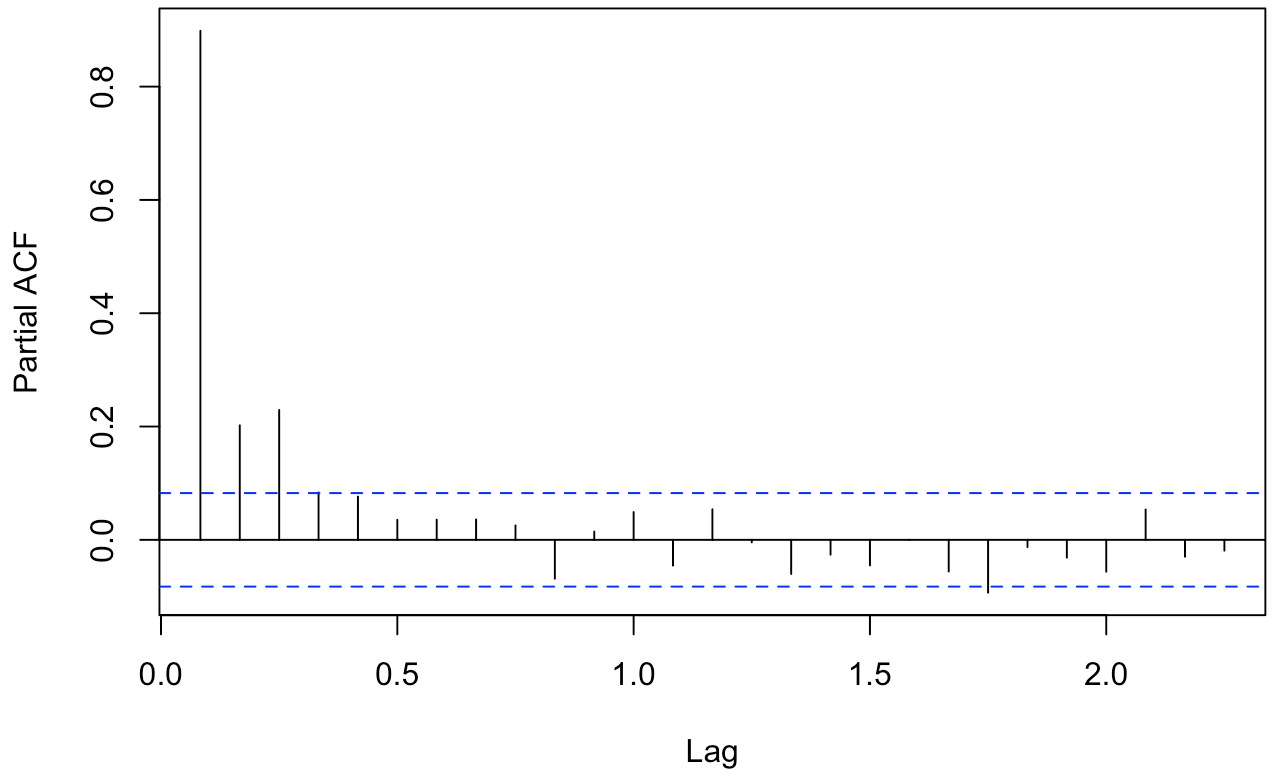
```
plot.ts(ns_data, type = 'l', ylab = 'Total Sales (Millions of Units)', xlab = 'Year', ma
in = "Car Sales Data")
```




```
acf(ns_data, lag.max = 100, main = "ACF plot of Car Sales Data")
```




```
pacf(ns_data, main = "PACF plot of Car Sales Data")
```



Check for stationarity using Dicky-Fuller Test.

H0: The time series is non-stationary.

H1: The time series is stationary.

```
adf.test(ns_data)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ns_data
## Dickey-Fuller = -2.4297, Lag order = 8, p-value = 0.3964
## alternative hypothesis: stationary
```

As the p-value is 0.3964 > 0.05, we fail to reject H0, which means that the data is not stationary.

In order to make the data stationary we transform the data by taking first difference and check for stationarity.

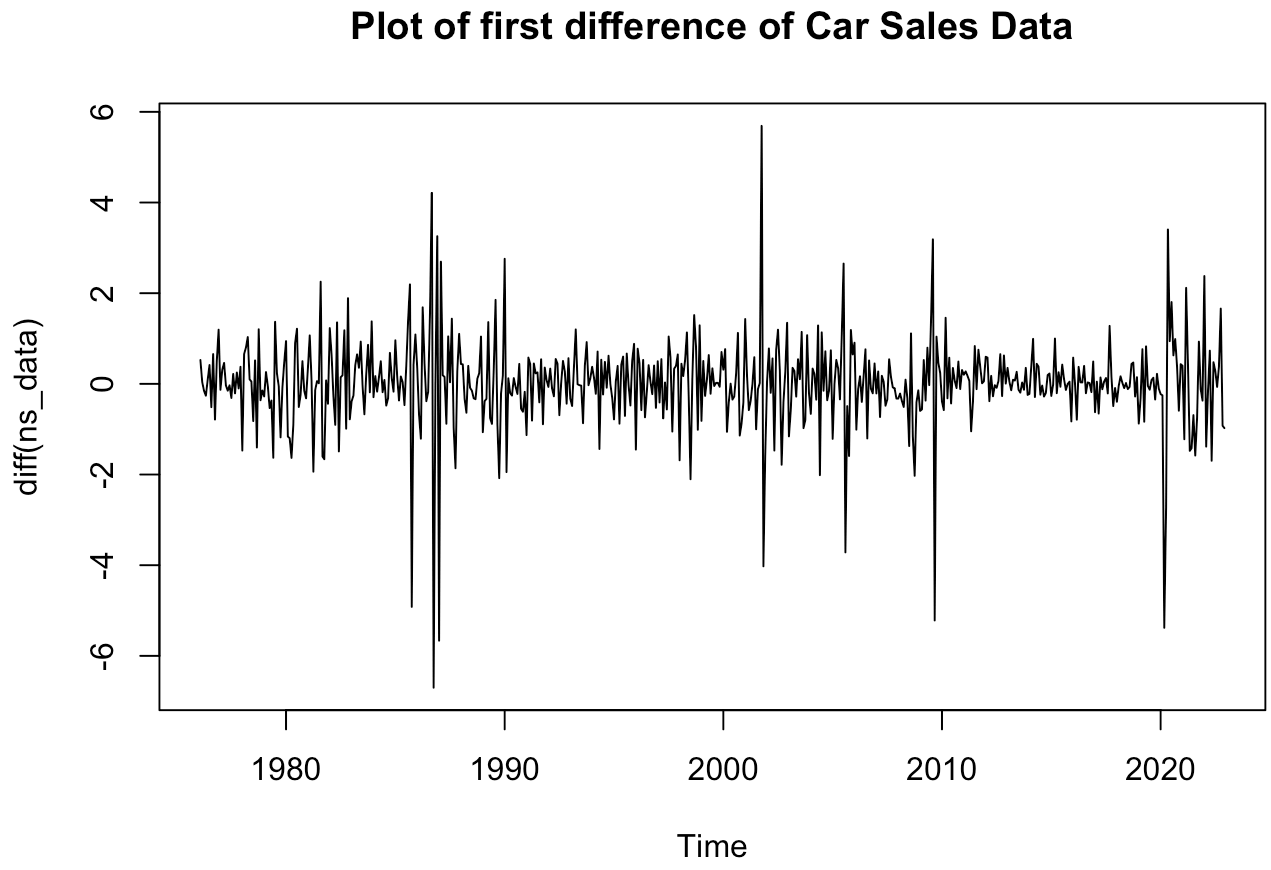
```
adf.test(diff(ns_data))
```

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

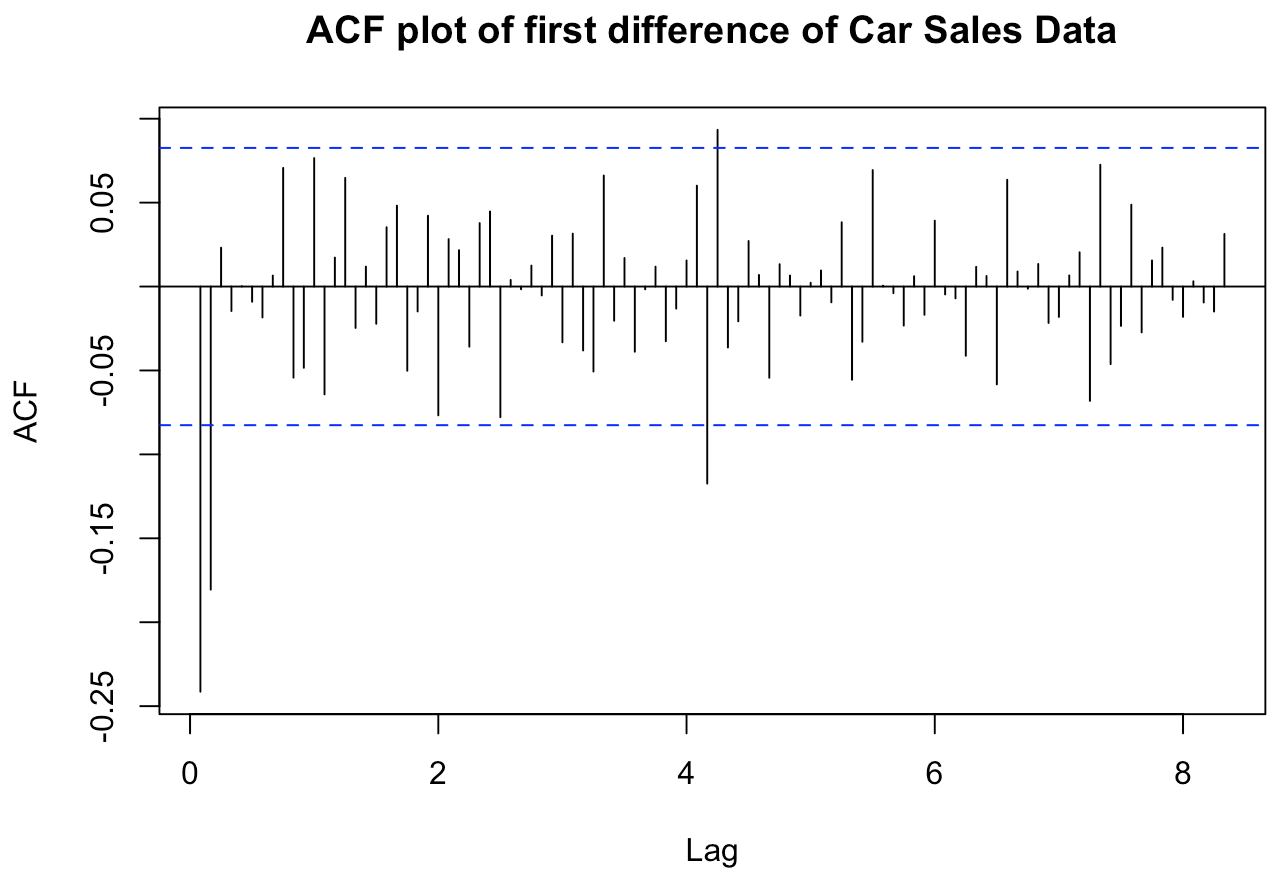
```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff(ns_data)
## Dickey-Fuller = -9.2541, Lag order = 8, p-value = 0.01
## alternative hypothesis: stationary
```

As the p-value is 0.01 < 0.05, we reject H0, the data is stationary.

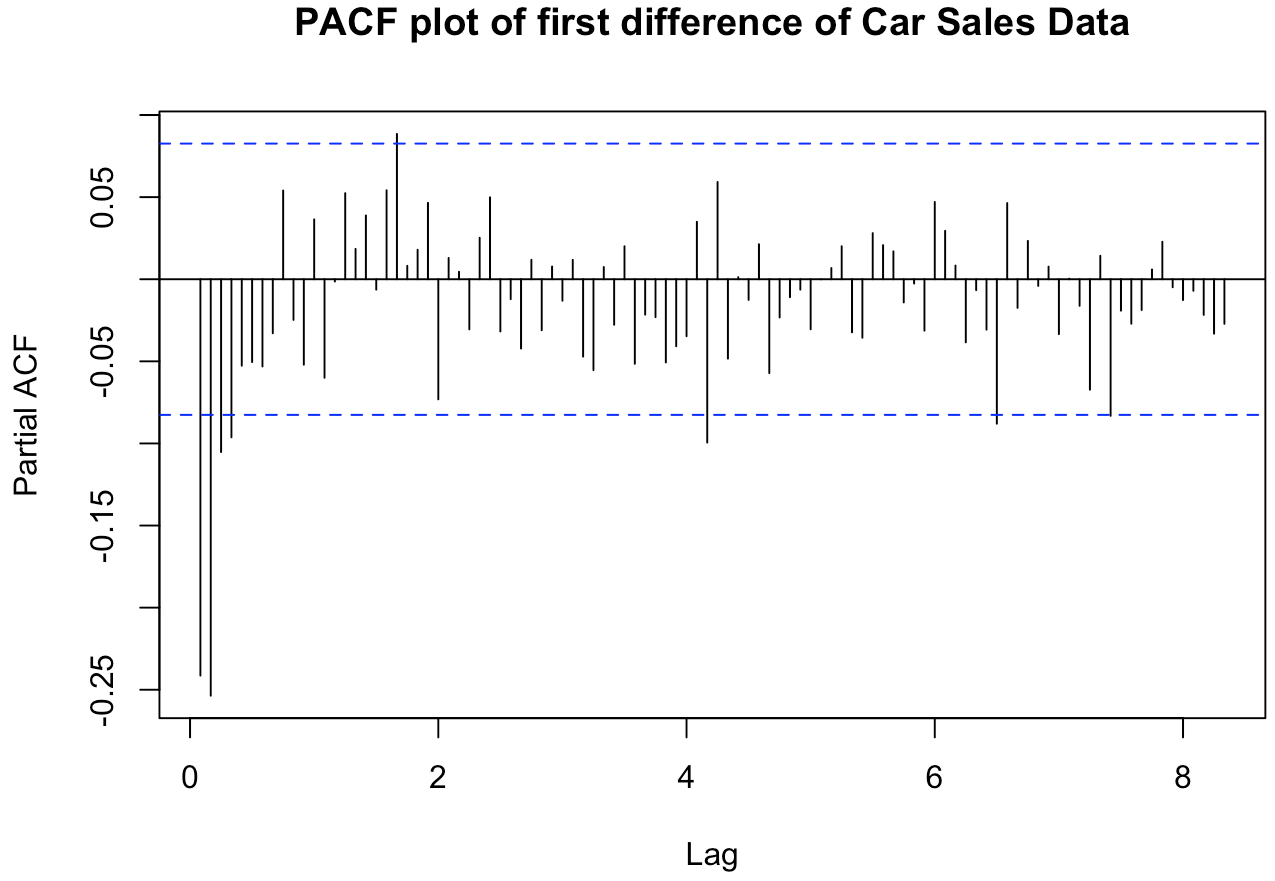
```
plot(diff(ns_data), type = 'l', main = "Plot of first difference of Car Sales Data")
```




```
acf(diff(ns_data), lag.max = 100, main = "ACF plot of first difference of Car Sales Dat
a")
```




```
pacf(diff(ns_data), lag.max = 100, main = "PACF plot of first difference of Car Sales Da
ta")
```




```
eacf(diff(ns_data))
```

```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x x o o o o o o o o o  o  o  o
## 1 x x o o o o o o o o o  o  o  o
## 2 x x o o o o o o o o o  o  o  o
## 3 x x x o o o o o o o o  o  o  o
## 4 x x x o o o o o o o o  o  o  o
## 5 x o x o o o o o o o o  o  o  o
## 6 x x o x o x o o o o o  o  o  o
## 7 x x o x x x o o o o o  o  o  o
```

Based on the ACF, PACF and EACF, we test for the following 4 models:-

1. ARIMA(0,1,2)
2. ARIMA(1,1,2)
3. ARIMA(2,1,2)
4. ARIMA(2,1,3)

Model 1 - ARIMA(0,1,2)

```
model1 = arima(ns_data,order=c(0,1,2))
model1
```

```
##
## Call:
## arima(x = ns_data, order = c(0, 1, 2))
##
## Coefficients:
##           ma1      ma2
##       -0.3484  -0.2084
## s.e.   0.0416   0.0420
##
## sigma^2 estimated as 0.8855:  log likelihood = -764.77,  aic = 1533.55
```

```
AIC(model1)
```

```
## [1] 1535.545
```

```
BIC(model1)
```

```
## [1] 1548.545
```

Model 2 - ARIMA(1,1,2)

```
model2 = arima(ns_data,order=c(1,1,2))
model2
```

```
##
## Call:
## arima(x = ns_data, order = c(1, 1, 2))
##
## Coefficients:
##           ar1      ma1      ma2
##       -0.0677  -0.2832  -0.2371
## s.e.   0.2215   0.2162   0.1000
##
## sigma^2 estimated as 0.8853:  log likelihood = -764.73,  aic = 1535.45
```

```
AIC(model2)
```

```
## [1] 1537.451
```

```
BIC(model2)
```

```
## [1] 1554.784
```

Model 3 - ARIMA(2,1,2)

```
model3 = arima(ns_data,order=c(2,1,2))
model3
```

```
##
## Call:
## arima(x = ns_data, order = c(2, 1, 2))
##
## Coefficients:
##           ar1     ar2      ma1      ma2
##       -0.2191  0.0815  -0.1314  -0.3667
## s.e.   0.3403  0.1688   0.3352   0.2642
##
## sigma^2 estimated as 0.885:  log likelihood = -764.61,  aic = 1537.22
```

```
AIC(model3)
```

```
## [1] 1539.22
```

```
BIC(model3)
```

```
## [1] 1560.887
```

Model 4 - ARIMA(2,1,3)

```
model4 = arima(ns_data,order=c(2,1,3))
model4
```

```
##
## Call:
## arima(x = ns_data, order = c(2, 1, 3))
##
## Coefficients:
##          ar1     ar2      ma1      ma2     ma3
##       0.0041  0.1029  -0.3547  -0.3094  0.0563
## s.e.  1.6975  0.2098   1.6977   0.5479  0.4019
##
## sigma^2 estimated as 0.8849:  log likelihood = -764.6,  aic = 1539.2
```

```
AIC(model4)
```

```
## [1] 1541.202
```

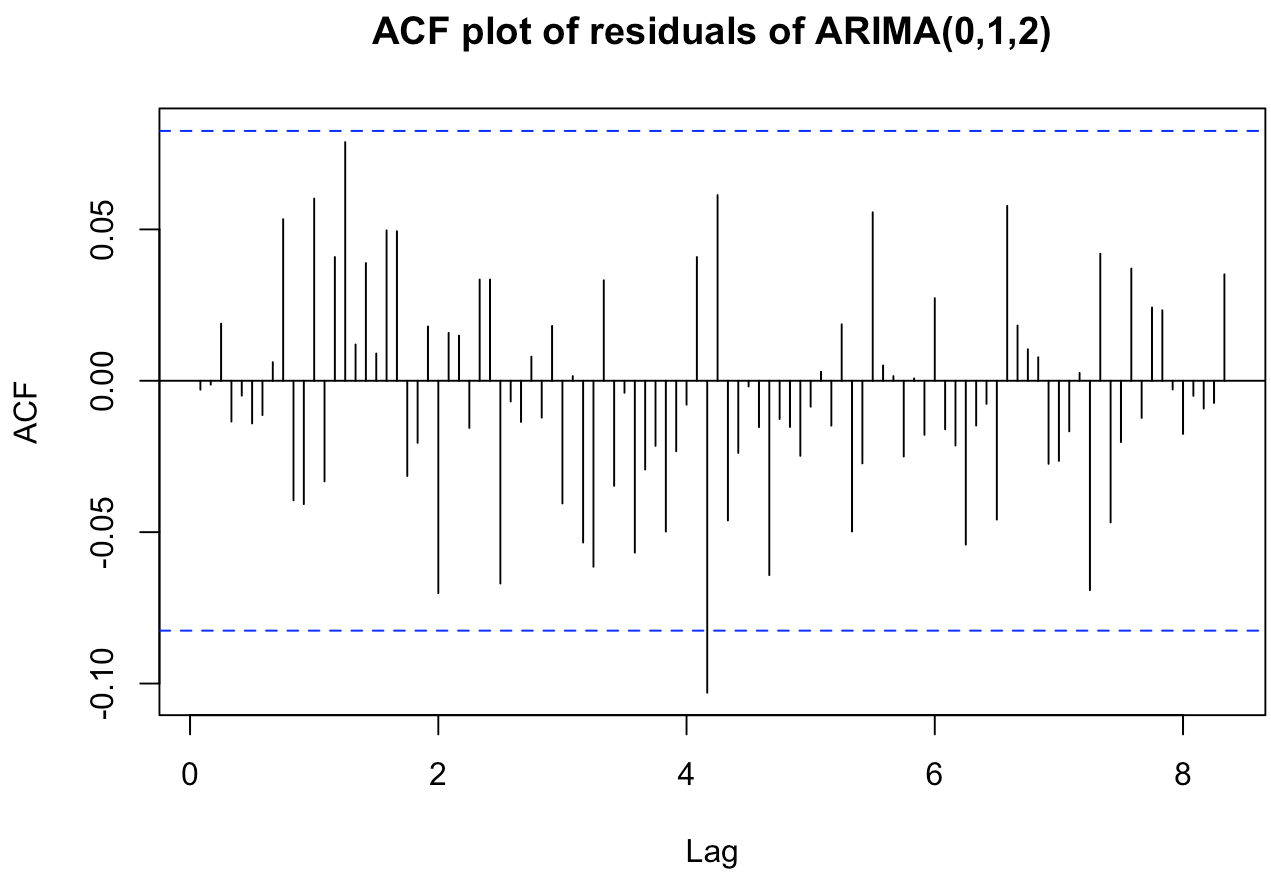
```
BIC(model4)
```

```
## [1] 1567.201
```

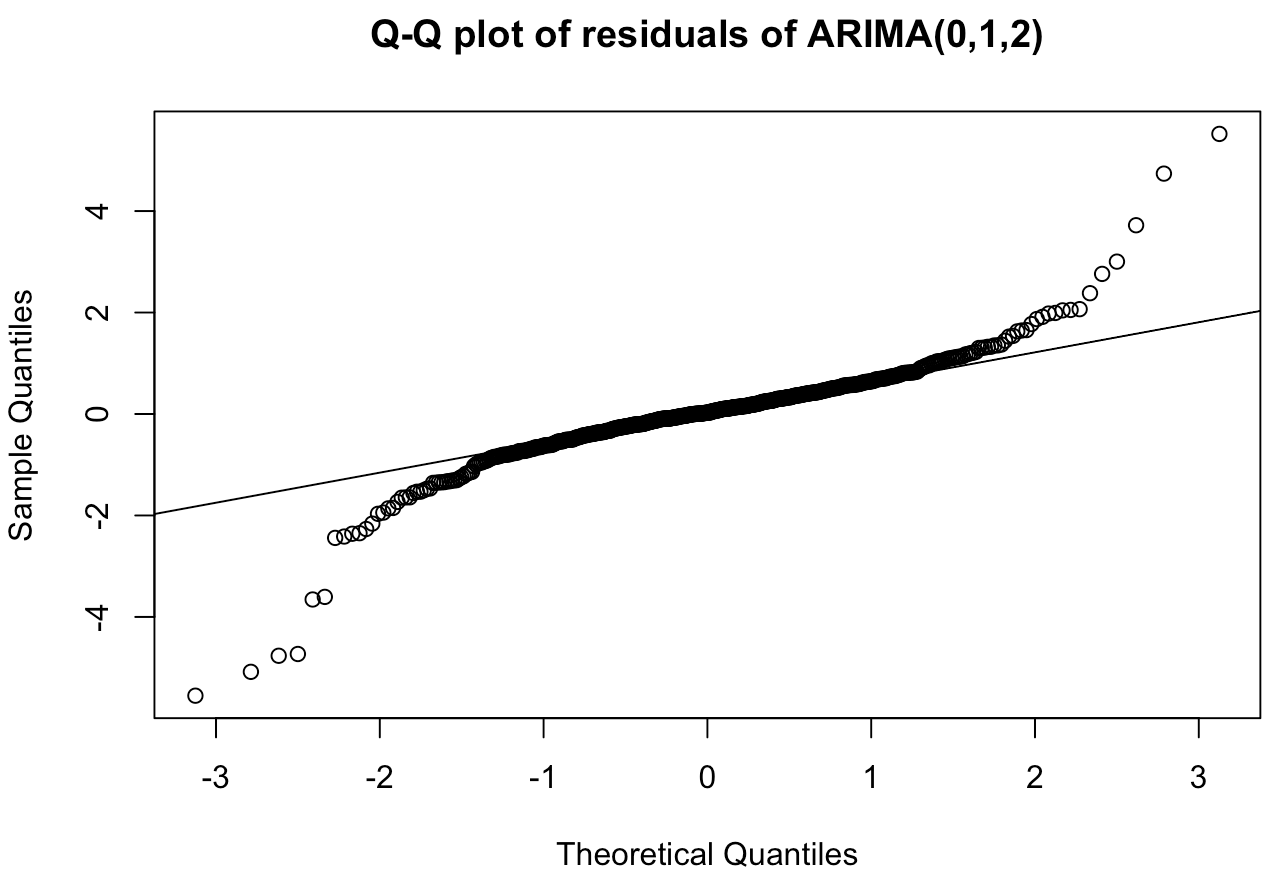
The best model for the above non-seasonal data is ARIMA(0,1,2) based on AIC and BIC values.

Residual Analysis

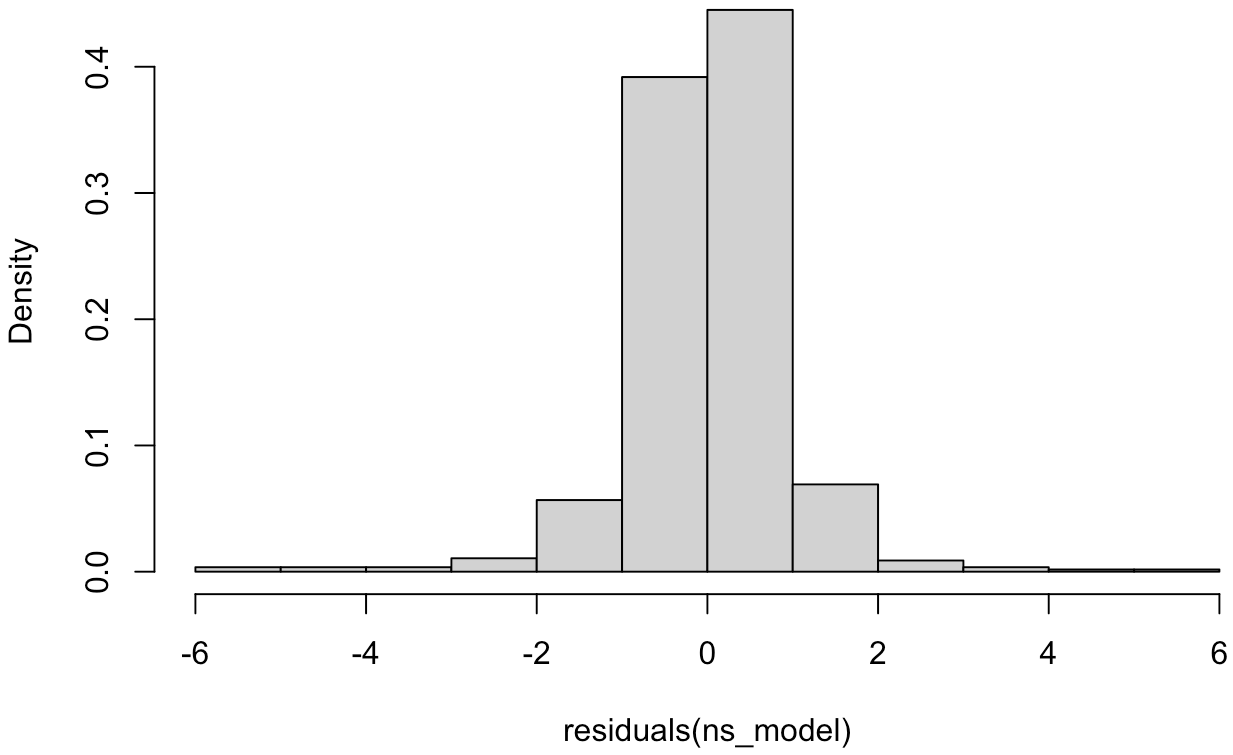
```
ns_model <- arima(ns_data,order=c(0,1,2))
acf(residuals(ns_model), lag.max = 100, main = "ACF plot of residuals of ARIMA(0,1,2)")
```




```
qqnorm(residuals(ns_model), main = "Q-Q plot of residuals of ARIMA(0,1,2)"); qqline(resi
duals(ns_model))
```




```
hist(residuals(ns_model), freq = FALSE, main = "Histogram of residuals of ARIMA(0,1,2)")
```




```
shapiro.test(residuals(ns_model))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(ns_model)
## W = 0.8711, p-value < 2.2e-16
```

From the Shapiro-Wilk test, the p-value of 2.2e-16 < 0.05, shows that the residual is not normal.

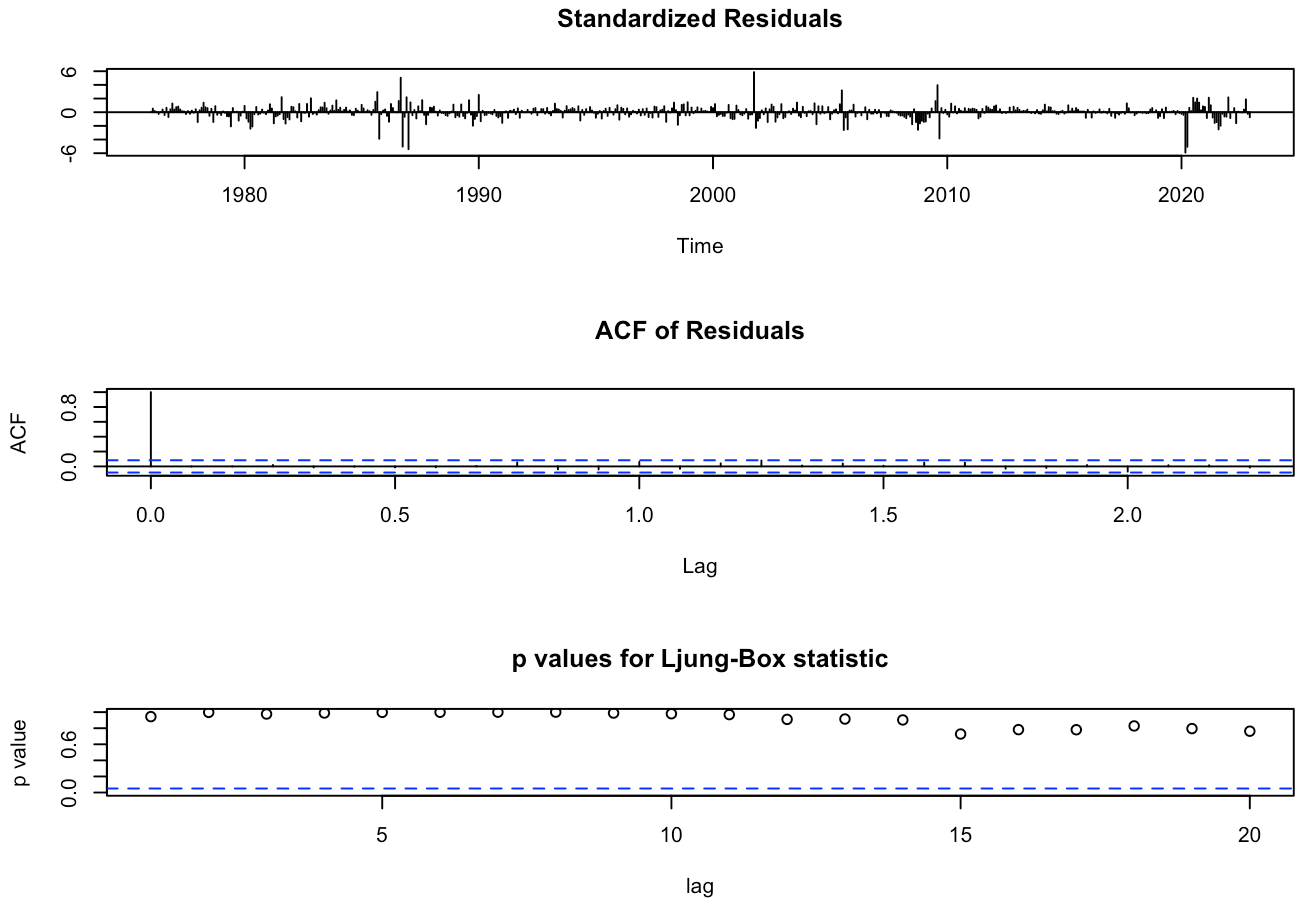
```
Box.test(residuals(ns_model), lag = 10, type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  residuals(ns_model)
## X-squared = 3.0699, df = 10, p-value = 0.9797
```

The Box-Ljung test, having p-value 0.9797 > 0.05, shows that the residuals are independent and identically distributed.

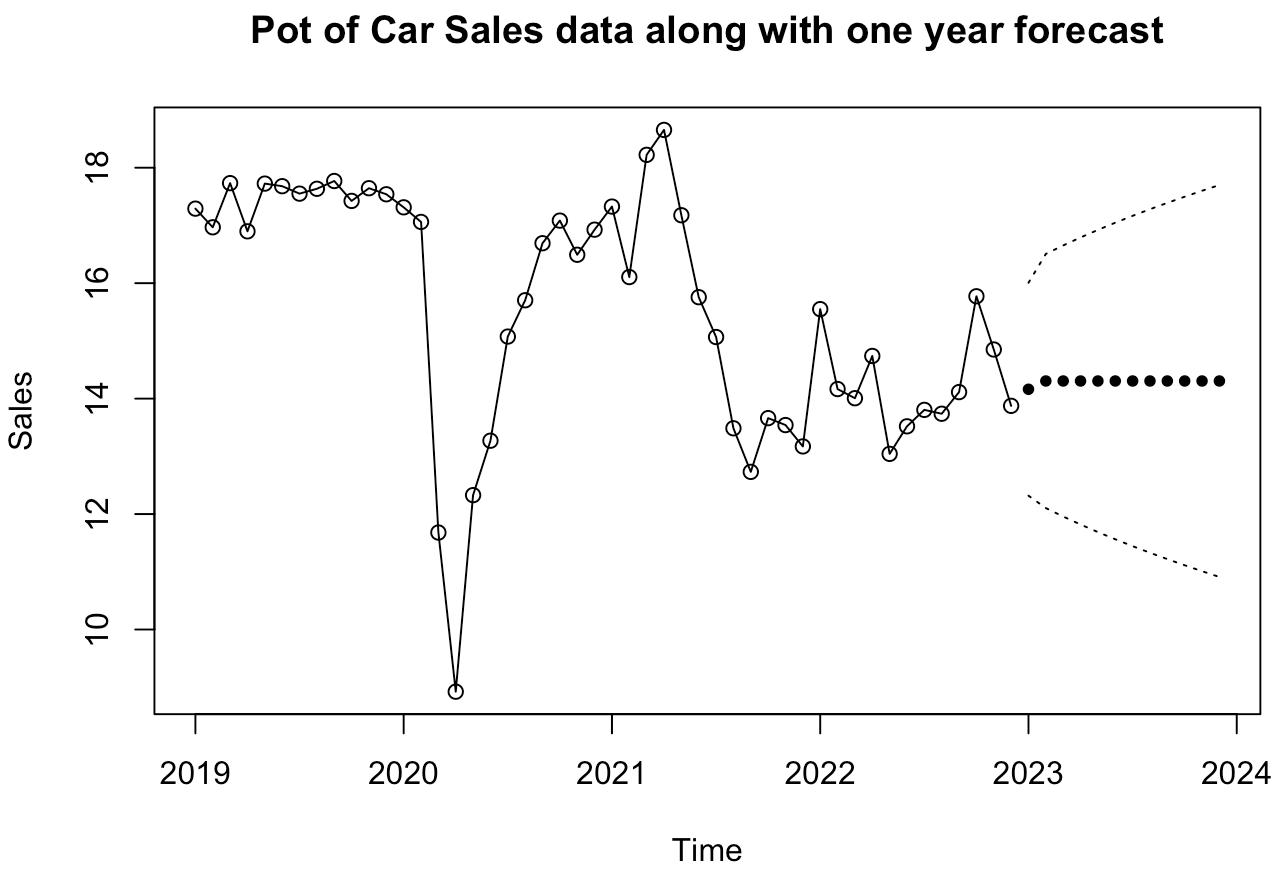
Diagnostic plot of ARIMA(0,1,2)

```
tsdiag(ns_model, gof.lag = 20)
```



Forecast

```
plot(ns_model, n1=c(2019,1), n.ahead=12,ylab='Sales',pch=20, main = "Pot of Car Sales da
ta along with one year forecast")
```



Conclusion

We can see that ARIMA(0,1,2) is a great fit to the data, and is able to forecast the Car Sales for 2023. The forecast seems to be a straight line since the ARIMA model tends to predict the approximate mean values, and gives a large confidence interval for the predicted values. If we see the confidence interval lines, they seem to have upper and lower limits around the recent trends.