MA641_MeetPatel_Project

Code

2023-18-12

Hide

```
library(tseries)
```

```
Warning: package ‘tseries’ was built under R version 4.2.3Registered S3 method overwritt
en by 'quantmod':
  method            from
  as.zoo.data.frame zoo

    ‘tseries’ version: 0.10-55

    ‘tseries’ is a package for time series analysis and computational finance.

    See ‘library(help="tseries")’ for details.
```

Introduction

Time series analysis is a method for analyzing data in order to spot trends and predict what will happen in the future. I will carry out time series analysis on two types of data i.e. seasonal and non-seasonal data. This project will provide a procedure to analyze and fit a time series model in R. Part A covers analysis and forecast of Closing Price of Schodinger Stock Data. Part B covers analysis and forecast of Catfish Sales in United States. The data comprises of catfish sales on monthly level. I’ve followed the Box-Jenkins approach in the project in order to fit an appropriate time series model.

Methodology

I follow Box-Jenkins Models to tackle the time-series data and fit an appropriate model to the data. The Box-Jenkins Model comprises of six steps that needs to be followed.

1. Stationarity
2. Estimating Models
3. Parameter Redundancy
4. Parameter Estimation
5. Residual Analysis
6. Forecast

Step 1: Stationarity: To check if the data is stationary, if the data is stationary we can move to the next step, else we need to make the data stationary using Differencing, Detrending or Transformation. To check stationarity we perform Dicky Fuller Test.

Step 2: Estimating Models: We estimate the p and q values of ARIMA model, based on the ACF and PACF plots on the stationary data. We also use EACF plot to estimate the models.

Step 3: Parameter Redundancy: We work with all the estimated models. We fit the model to all the combinations of estimated p,d,q values.

Step 4: Parameter Estimation: Once we fit all the models, we compare the models and check the loglikelihood, AIC and BIC value. We select the model with lowest AIC and BIC values, and lower number of parameters. We can selected the model with slightly higher AIC or BIC, if it reduces the number of parameters in the model significantly.

Step 5: Residual Analysis: Based on the model that we find to be the best fit, we perform analysis on the residuals of the model. We plot the ACF plot to check if the residuals are uncorrelated. We check the normality of the residuals by plotting Q-Q plot, histogram and performing Shapiro-Wilk Test. We perform Ljung-Box Test to know if the residual is white noise or not.

Step 6: Forecast: The final step of Time Series Analysis, is to forecast data for the future. We fit the best model we found above on the original data and forecast the future values.

Part A: Non-Seasonal Data

For Non-Seasonal Data, I've taken the Schodinger Stock Data, consisting of daily Closing Price. The data is dated from Feb 2020 to Dec 2023. I will try to fit a time series model and lastly predict the closing price of the next few days.

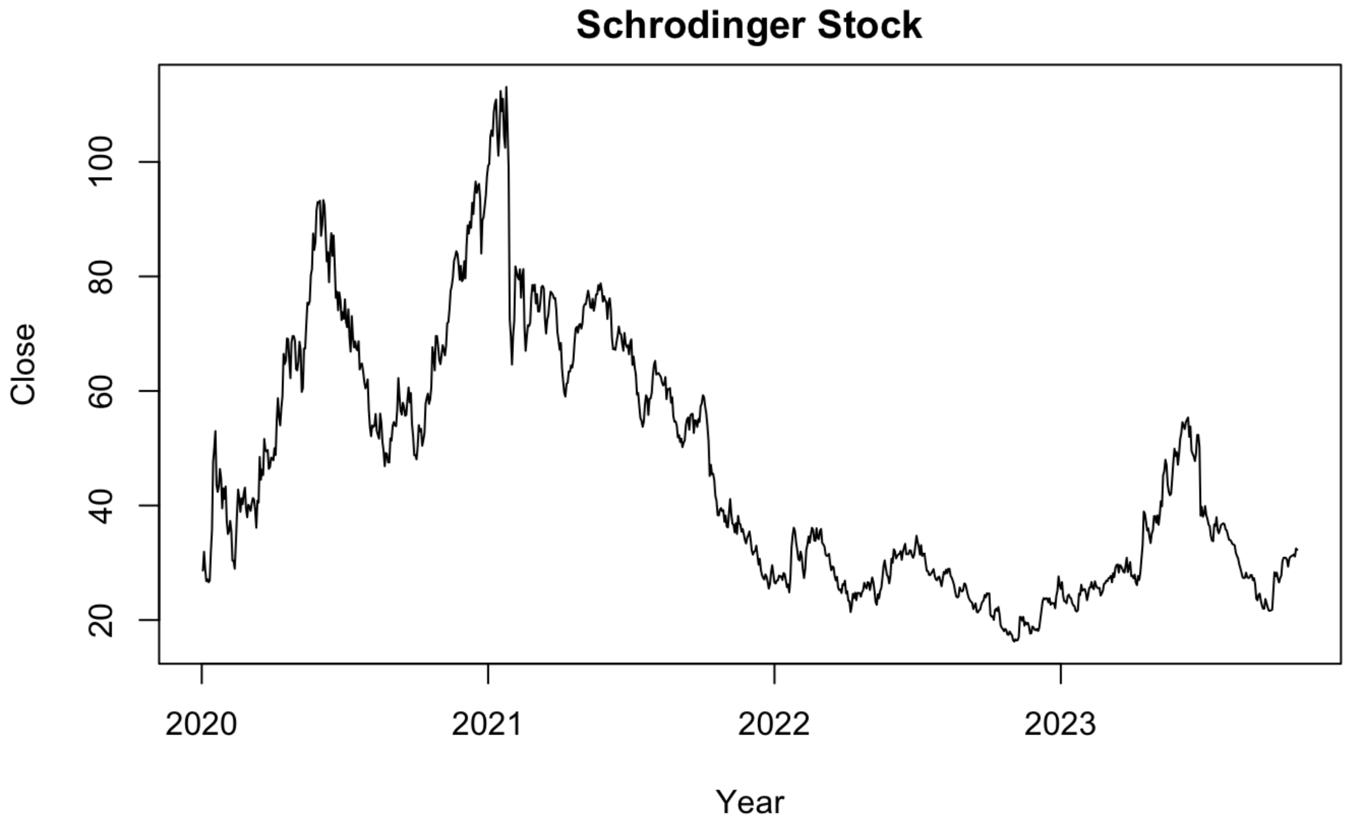
date <chr>	open <dbl>	high <dbl>	low <dbl>	close <dbl>	volume <dbl>
2020-02-06	26.0000	31.4000	25.5000	28.640	7624541
2020-02-07	30.4500	34.1500	29.3100	31.920	3225299
2020-02-10	32.3800	33.4500	28.1100	28.940	2007709
2020-02-11	28.7500	29.5197	26.6500	26.790	1253919
2020-02-12	27.3300	28.2160	26.6600	27.130	1510572
2020-02-13	27.0800	27.5500	26.0600	26.600	595058
2020-02-14	26.6100	27.1000	26.2400	26.900	564797
2020-02-18	27.1700	32.1600	27.0000	31.900	1509702
2020-02-19	32.2800	36.4142	32.2800	35.720	1626915
2020-02-20	36.4000	48.7300	32.8400	47.620	6922284

1-10 of 964 rows Previous **1** 2 3 4 5 6 ... 97 Next

Hide

```
head(ts_data)
```

```
[1] 28.64 31.92 28.94 26.79 27.13 26.60
```



Check for stationarity using Dicky-Fuller Test.

H0: The time series is non-stationary.

H1: The time series is stationary.

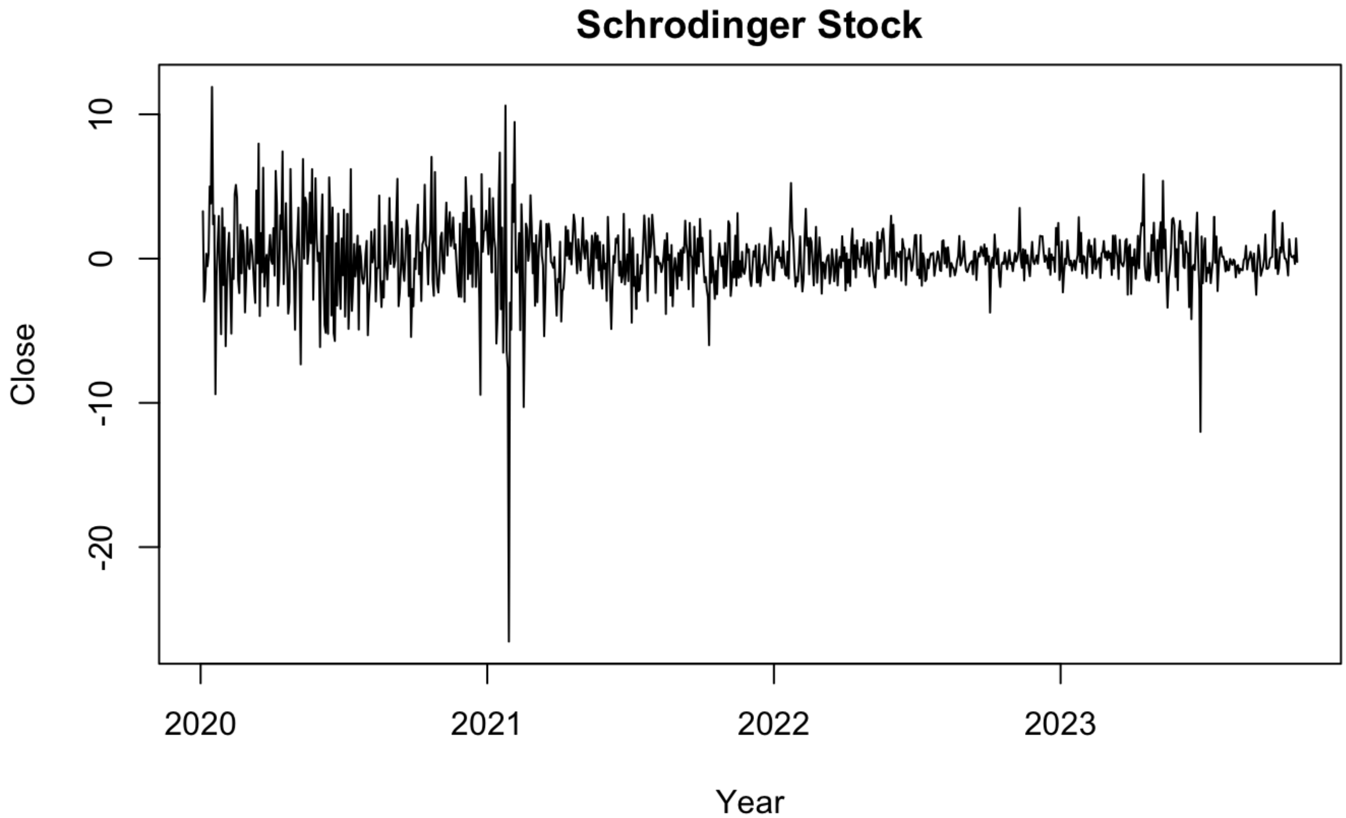
Hide

```
adf.test(ts_data)
```

```

	Augmented Dickey-Fuller Test

data:  ts_data
Dickey-Fuller = -2.3289, Lag order = 9, p-value = 0.4391
alternative hypothesis: stationary
```

Since p-value is 0.4391 > 0.05, we fail to reject H0, the data is not stationary.

Since, the data is not stationary we will take difference of the series to make it stationary.



Check for stationarity using Dicky-Fuller Test.

H0: The time series is non-stationary.

H1: The time series is stationary.

Hide

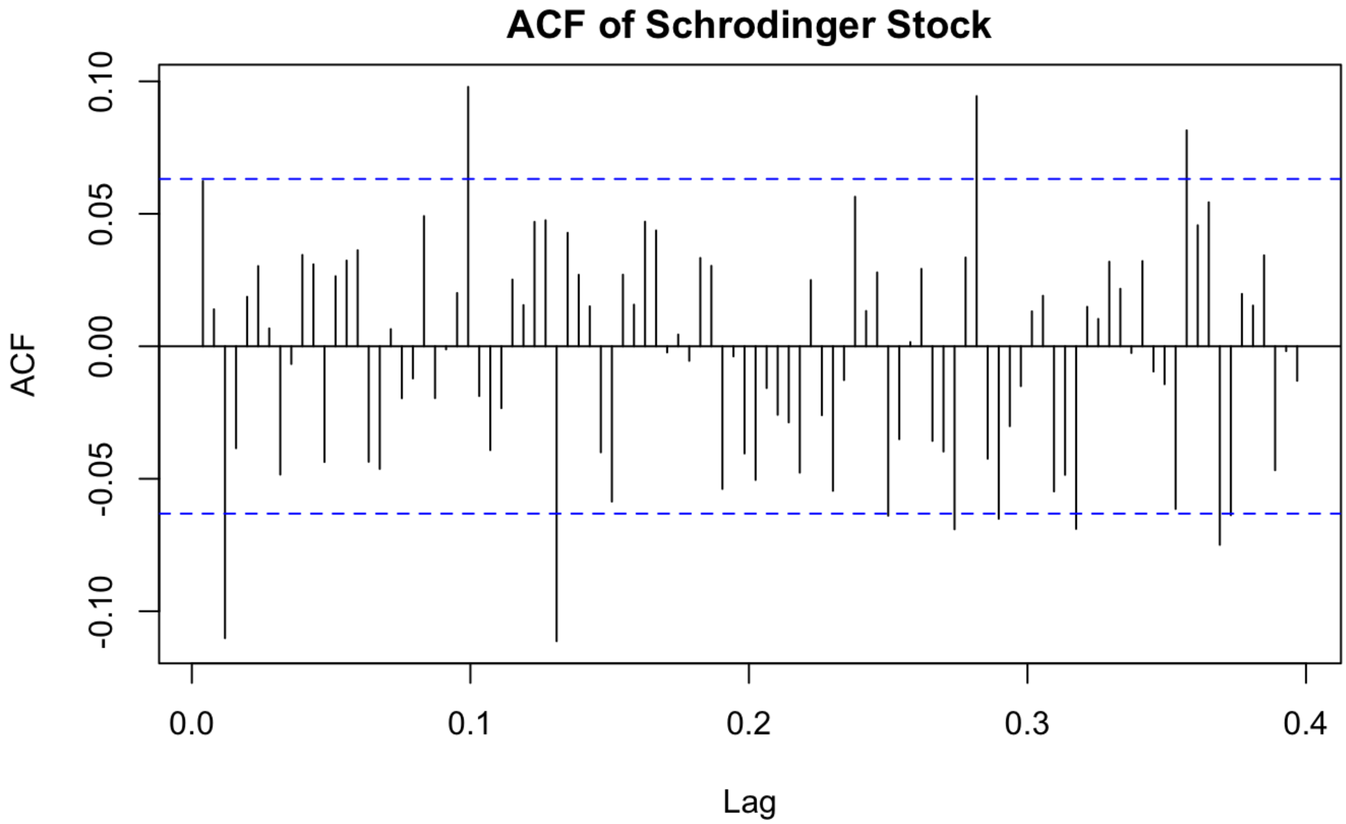
```
adf.test(ts_diff_data)
```

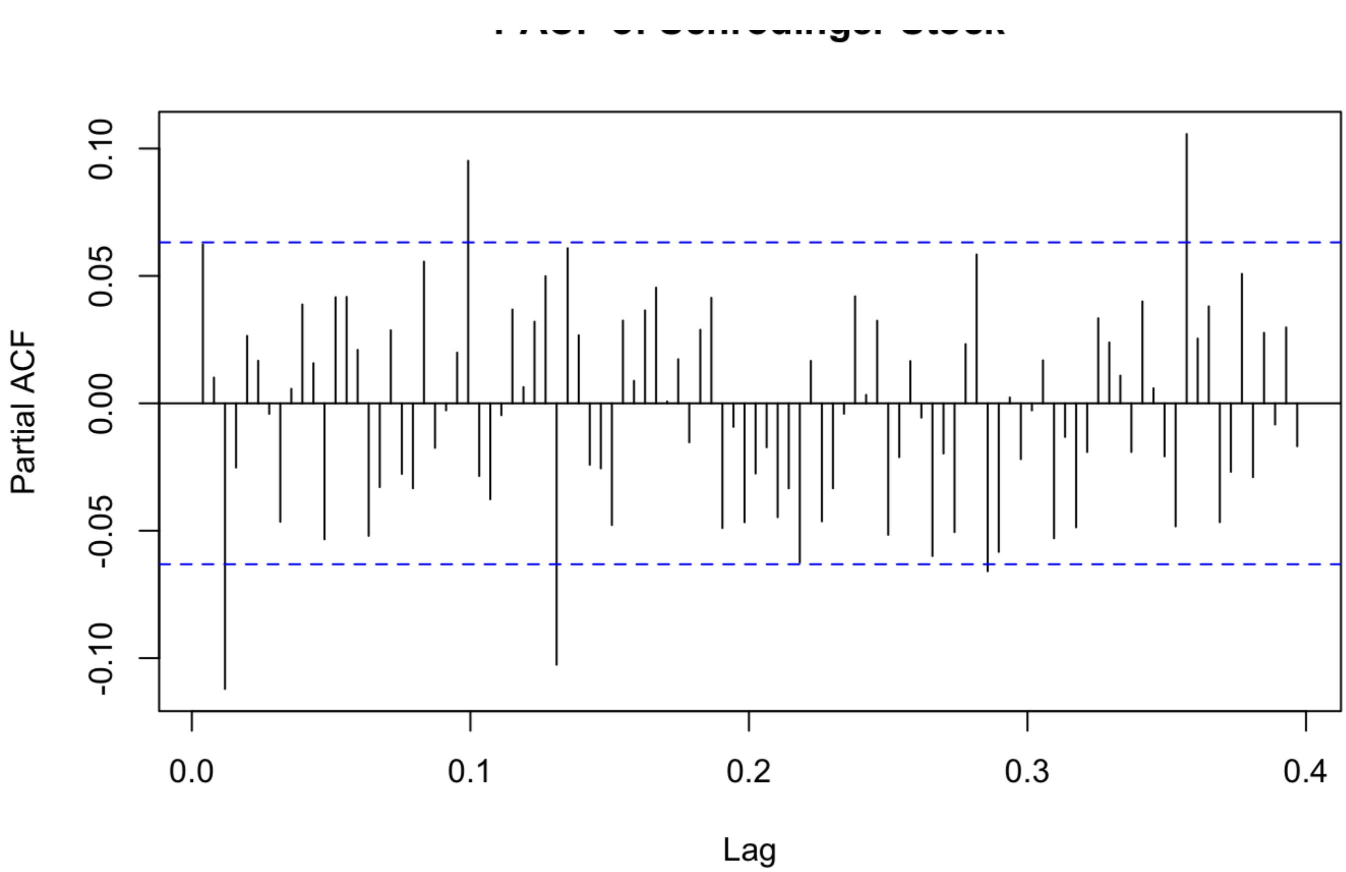
```
Warning: p-value smaller than printed p-value
```

```

	Augmented Dickey-Fuller Test

data:  ts_diff_data
Dickey-Fuller = -10.113, Lag order = 9, p-value = 0.01
alternative hypothesis: stationary
```

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

The data is stationary we will plot ACF and PACF.



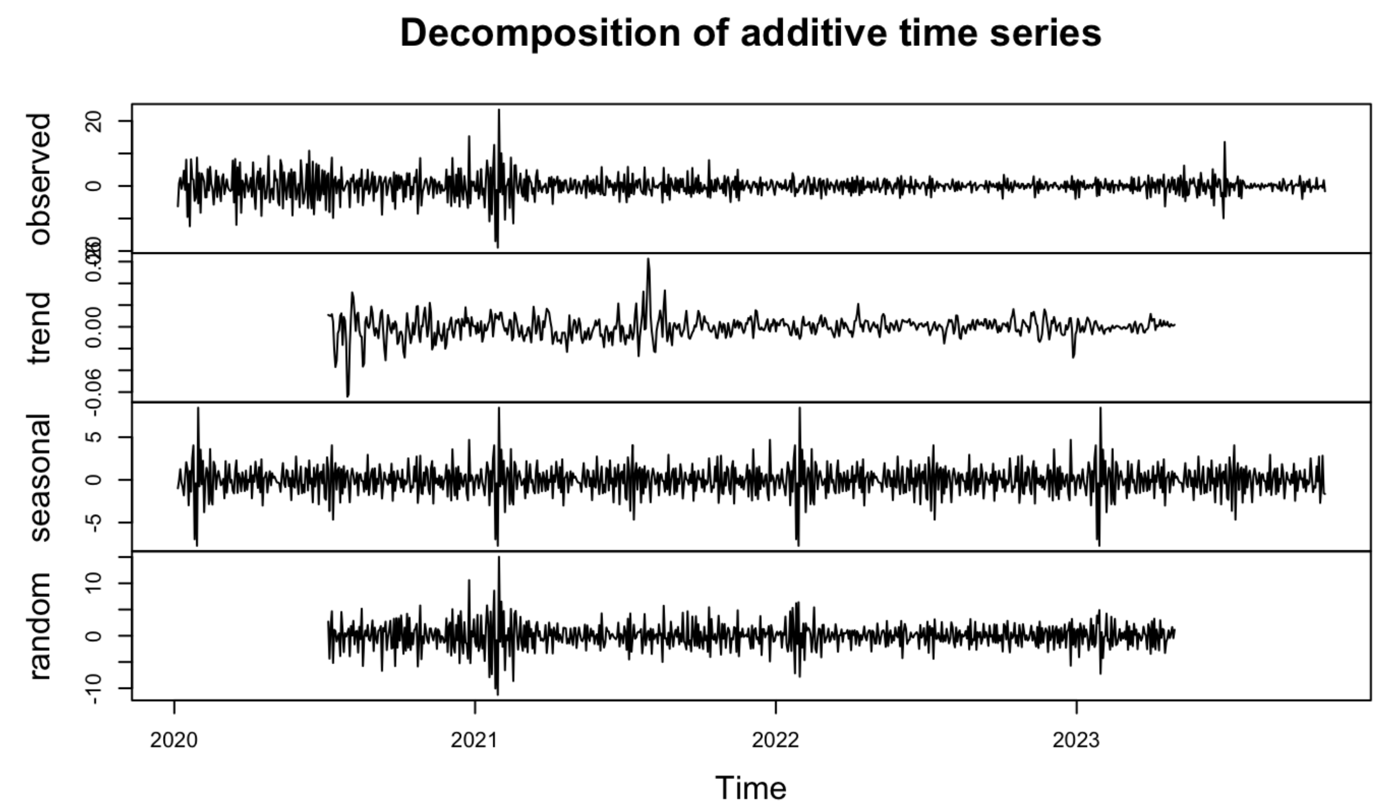


Hide

```
eacf(ts_diff_data)
```

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

Based on the ACF, PACF and EACF, we test for the following 4 models:- 1. ARIMA(0,1,3) 2. ARIMA(2,1,3) 3. ARIMA(3,1,3) 4. ARIMA(4,1,3) 5. ARIMA(5,1,3)



Hide

```
model1
```

```
Call:
arima(x = ts_diff_data, order = c(0, 1, 3))

Coefficients:
          ma1      ma2      ma3
      -0.9310  -0.0367  -0.0323
s.e.   0.0336   0.0409   0.0349

sigma^2 estimated as 5.681:  log likelihood = -2203.9,  aic = 4413.81
```

Hide

```
AIC(model1)
```

```
[1] 4415.809
```

Hide

```
BIC(model1)
```

```
[1] 4435.285
```

Hide

```
model2 = arima(ts_diff_data,order=c(2,1,3))
model2
```

```
Call:
arima(x = ts_diff_data, order = c(2, 1, 3))

Coefficients:
         ar1      ar2      ma1     ma2      ma3
      0.2124  -0.8105  -1.1370  0.9796  -0.8426
s.e.  0.0881   0.0733   0.0779  0.0959   0.0714

sigma^2 estimated as 5.587:  log likelihood = -2196.01,  aic = 4402.03
```

Hide

```
AIC(model2)
```

```
[1] 4404.029
```

Hide

```
BIC(model2)
```

```
[1] 4433.243
```

Hide

```
model3 = arima(ts_diff_data,order=c(3,1,3))
model3
```

```
Call:
arima(x = ts_diff_data, order = c(3, 1, 3))

Coefficients:
         ar1      ar2      ar3      ma1     ma2      ma3
      0.2963  -0.7067  -0.0479  -1.2353  0.9604  -0.7251
s.e.  0.1476   0.1095   0.0397   0.1456  0.1459   0.1153

sigma^2 estimated as 5.582:  log likelihood = -2195.61,  aic = 4403.22
```

Hide

```
AIC(model3)
```

```
[1] 4405.222
```

Hide

```
BIC(model3)
```

```
[1] 4439.305
```

Hide

```
model4 = arima(ts_diff_data,order=c(4,1,3))
model4
```

```
Call:
arima(x = ts_diff_data, order = c(4, 1, 3))

Coefficients:
```

```
Warning: NaNs produced
```

```
          ar1     ar2      ar3      ar4      ma1      ma2      ma3
      -0.4969  0.0401  -0.0990  -0.0874  -0.4414  -0.5444  -0.0142
s.e.      NaN  0.0761   0.0388      NaN      NaN      NaN   0.1409

sigma^2 estimated as 5.607:  log likelihood = -2197.73,  aic = 4409.46
```

Hide

```
AIC(model4)
```

```
[1] 4411.463
```

Hide

```
BIC(model4)
```

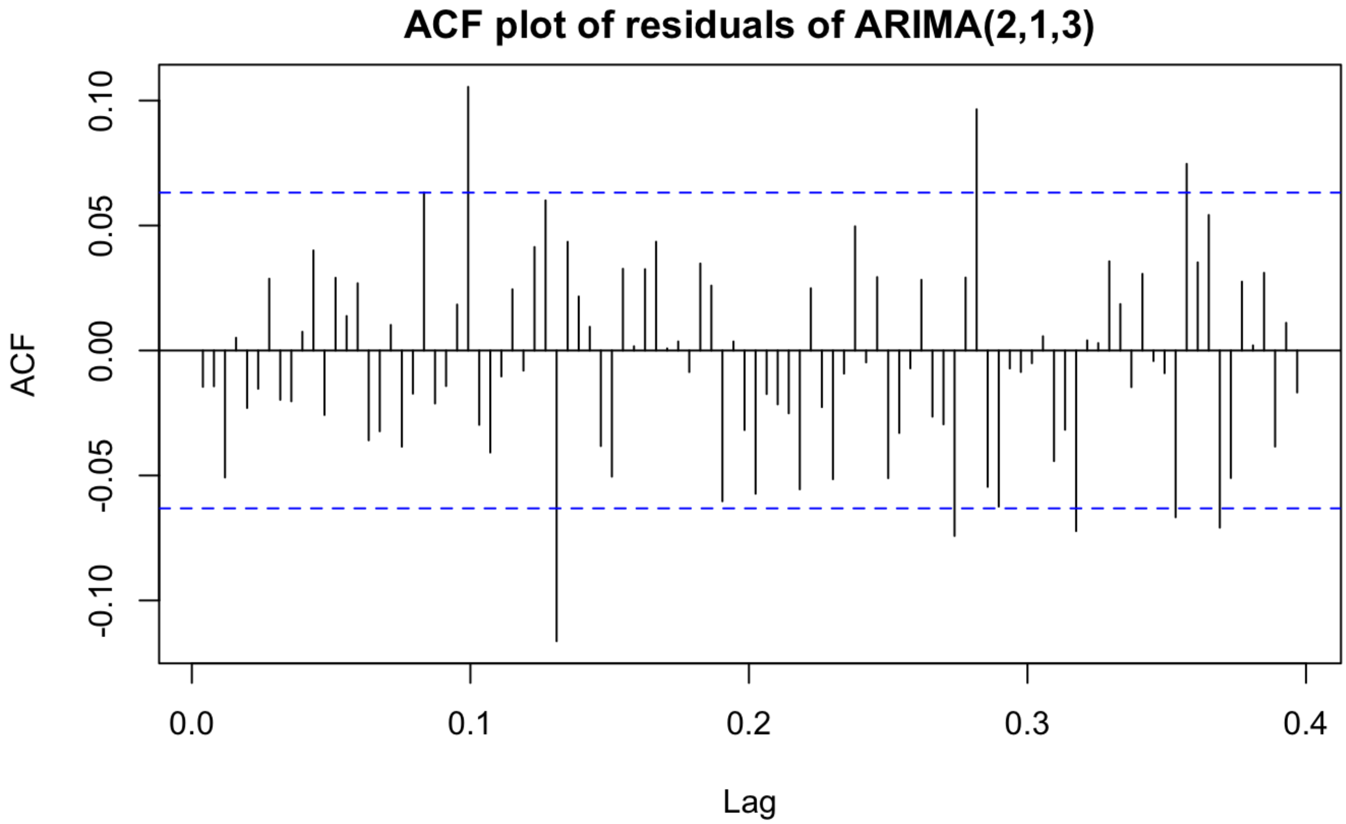
```
[1] 4450.416
```

The best model for the above non-seasonal data is ARIMA(2,1,3) based on AIC and BIC values.

Residual Analysis

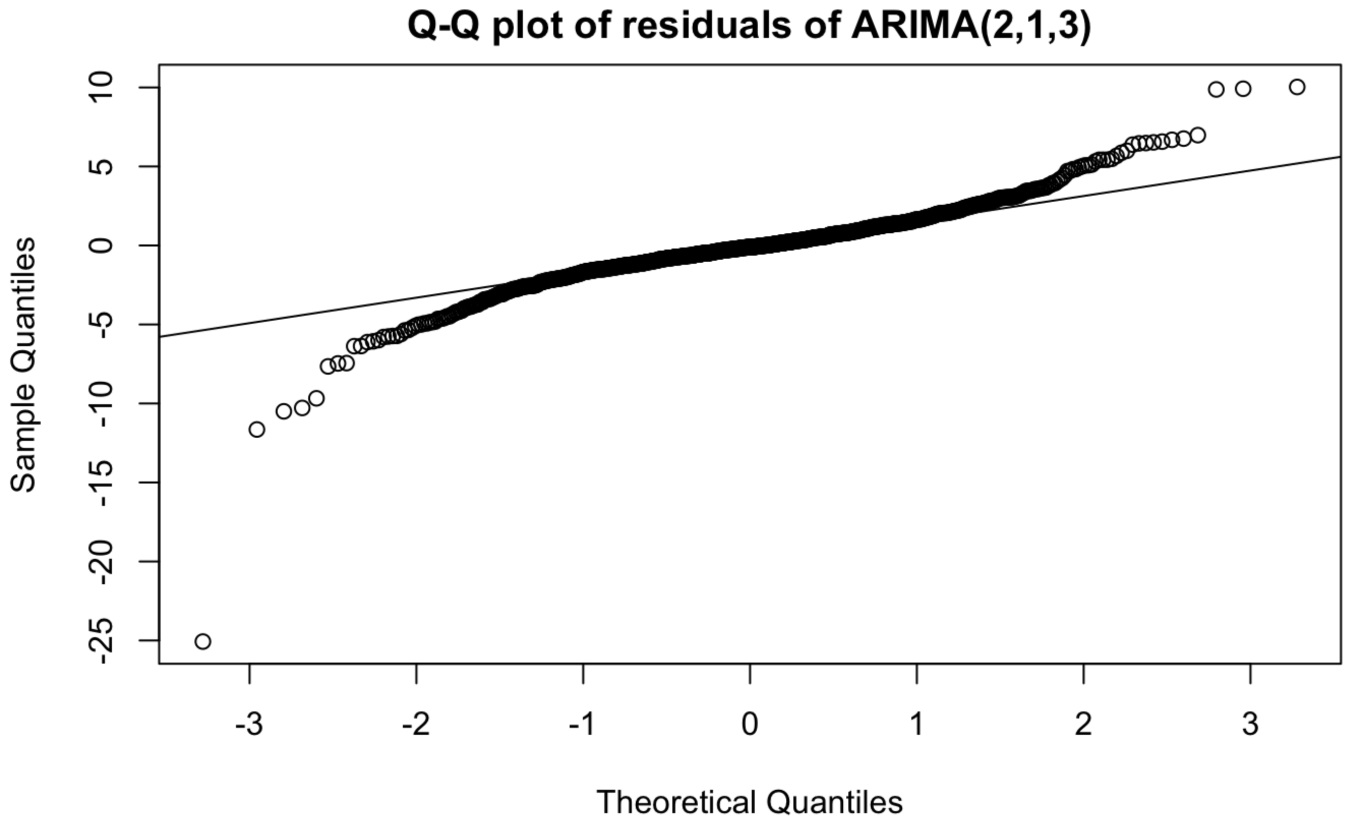
Hide

```
selected_model <- arima(ts_diff_data,order=c(2,1,3))
acf(residuals(selected_model), lag.max = 100, main ="ACF plot of residuals of ARIMA(2,1,
3)")
```



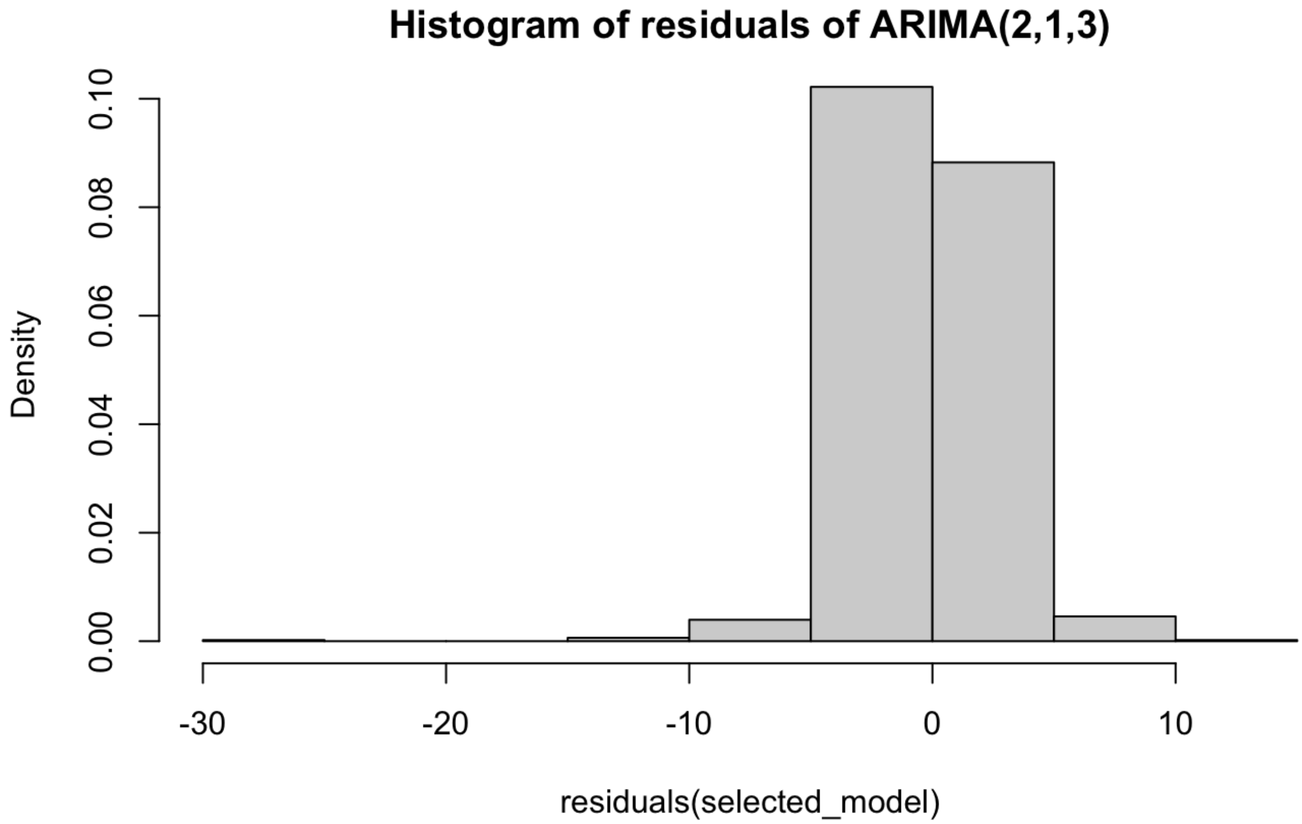
Hide

```
qqnorm(residuals(selected_model), main = "Q-Q plot of residuals of ARIMA(2,1,3)"); qqlin
e(residuals(selected_model))
```



Hide

```
hist(residuals(selected_model), freq = FALSE, main = "Histogram of residuals of ARIMA(2,
1,3)")
```



Hide

```
shapiro.test(residuals(selected_model))
```

```

	Shapiro-Wilk normality test

data:  residuals(selected_model)
W = 0.89282, 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.

Hide

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

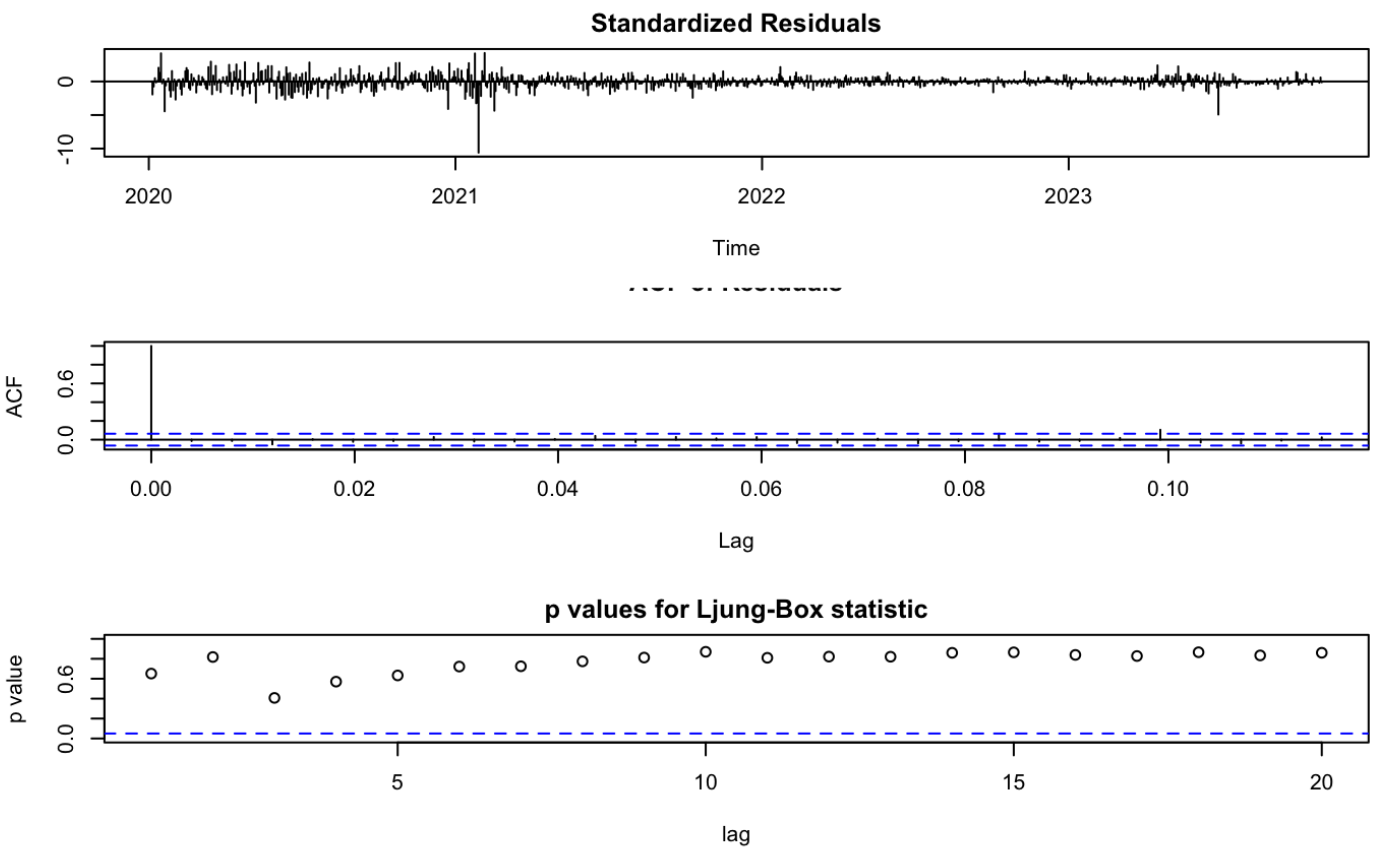
```

	Box-Ljung test

data:  residuals(selected_model)
X-squared = 5.3009, df = 10, p-value = 0.8702
```

The Box-Ljung test, having p-value 0.8702 > 0.05, shows that the residuals are independent and identically distributed.

Diagnostic plot of ARIMA(2,1,3)

Hide

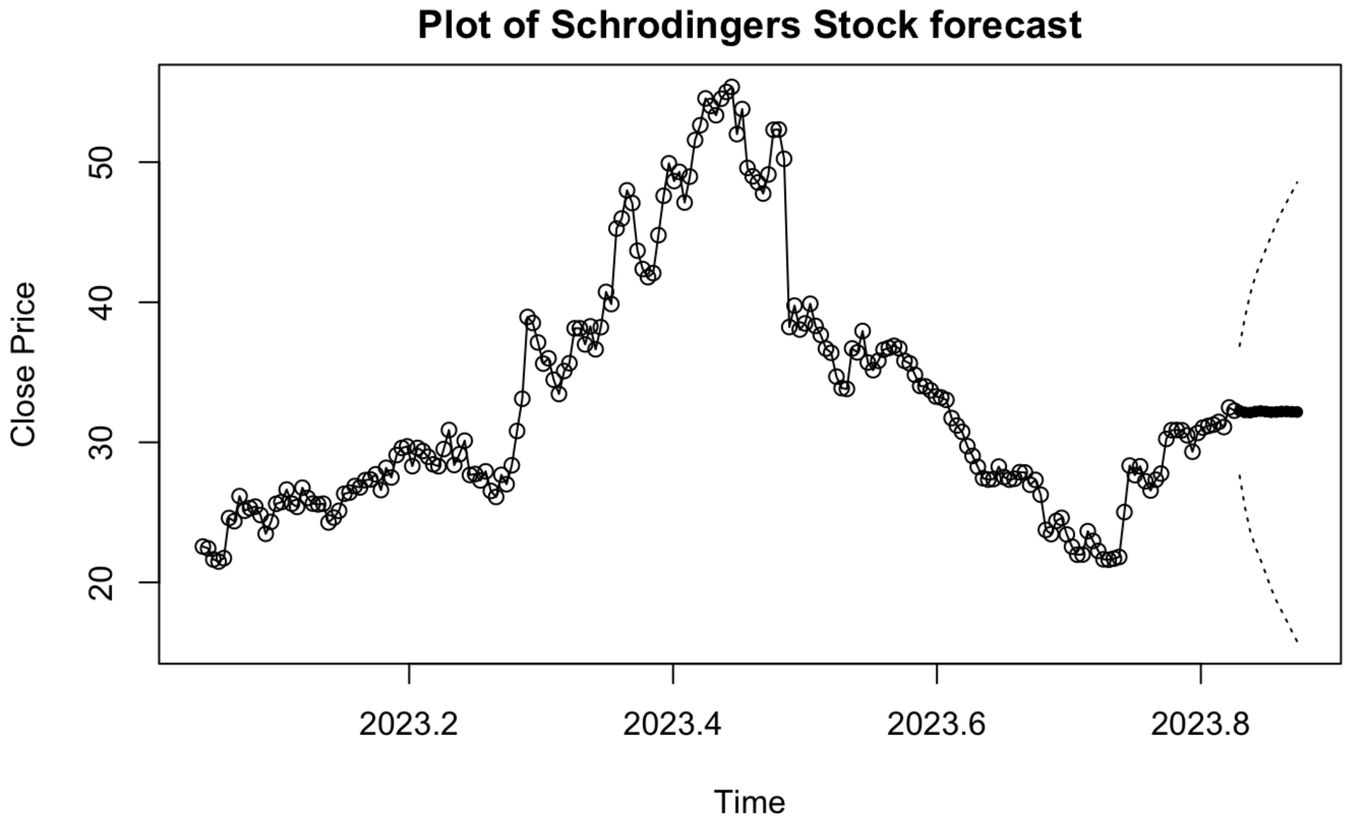
```
tsdiag(selected_model, gof.lag = 20)
```



Forecast

Hide

```
selected_model <- arima(ts_data,order=c(2,1,3))
plot(selected_model, n1=c(2023,12), n.ahead=12, ylab='Close Price',pch=20, main = "Plot
of Schrodingers Stock forecast")
```



Conclusion

We can see that ARIMA(2,1,3) is not a great fit to the data, and is not able to forecast the Closing Prices. 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. As observed in the ACF plot of residual there are still significant lag, meaning there are still trends that we fail to capture. We might get better results using GARCH models.

Part B: Seasonal Data

For Seasonal Data, I've taken Catfish sales data for United States, which has the monthly data for the Catfish sales in US from 1986 to 2012. I will fit the data to a time series model and lastly predict the Catfish sales for future years.

X <int>	Year <int>	Month <chr>	Value <chr>	Date <chr>
0	1986	Jan	9,034	1986-01-01
27	1986	Feb	9,596	1986-02-01
54	1986	Mar	10,558	1986-03-01
81	1986	Apr	9,002	1986-04-01
108	1986	May	9,239	1986-05-01
135	1986	Jun	8,951	1986-06-01
162	1986	Jul	9,668	1986-07-01

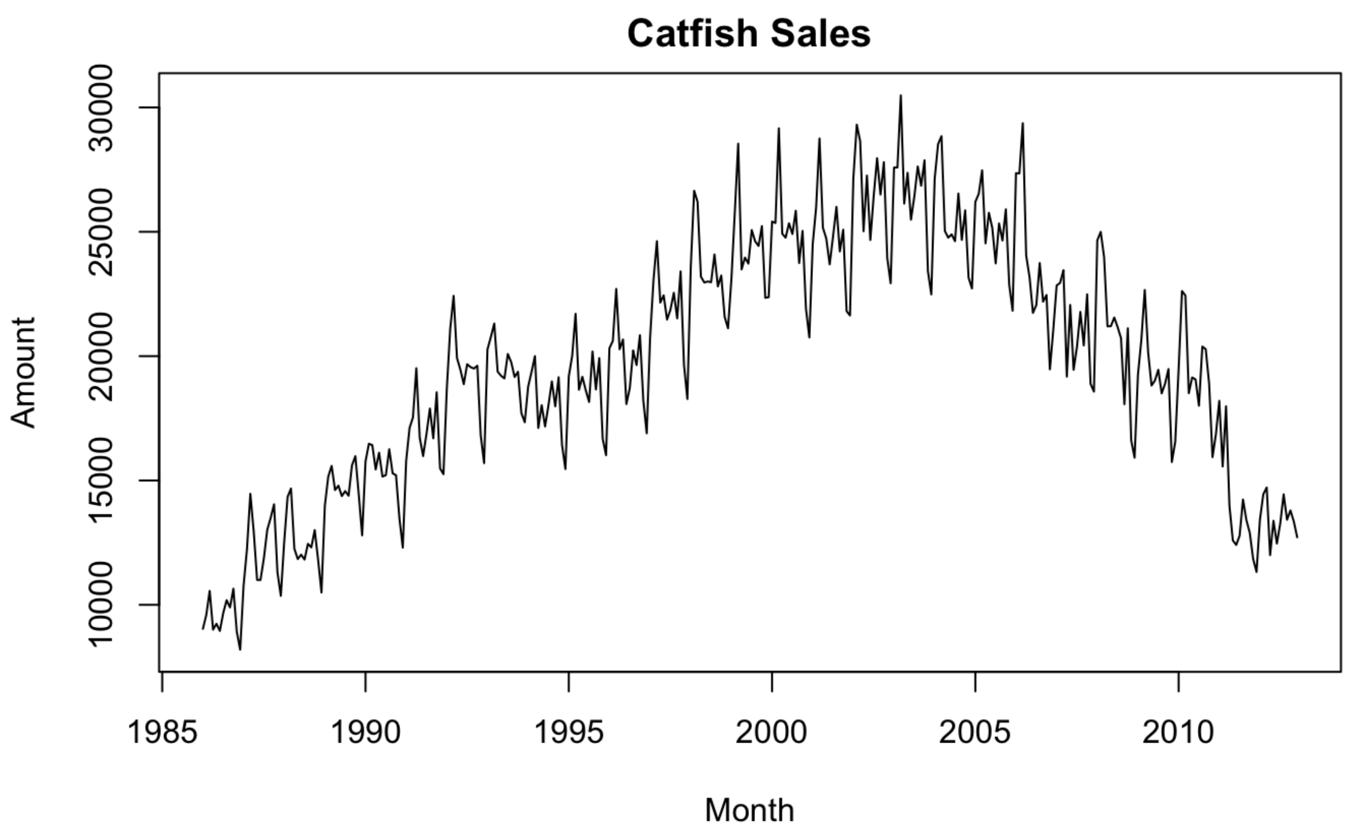
X <int>	Year <int>	Month <chr>	Value <chr>	Date <chr>
189	1986	Aug	10,188	1986-08-01
216	1986	Sep	9,896	1986-09-01
243	1986	Oct	10,649	1986-10-01

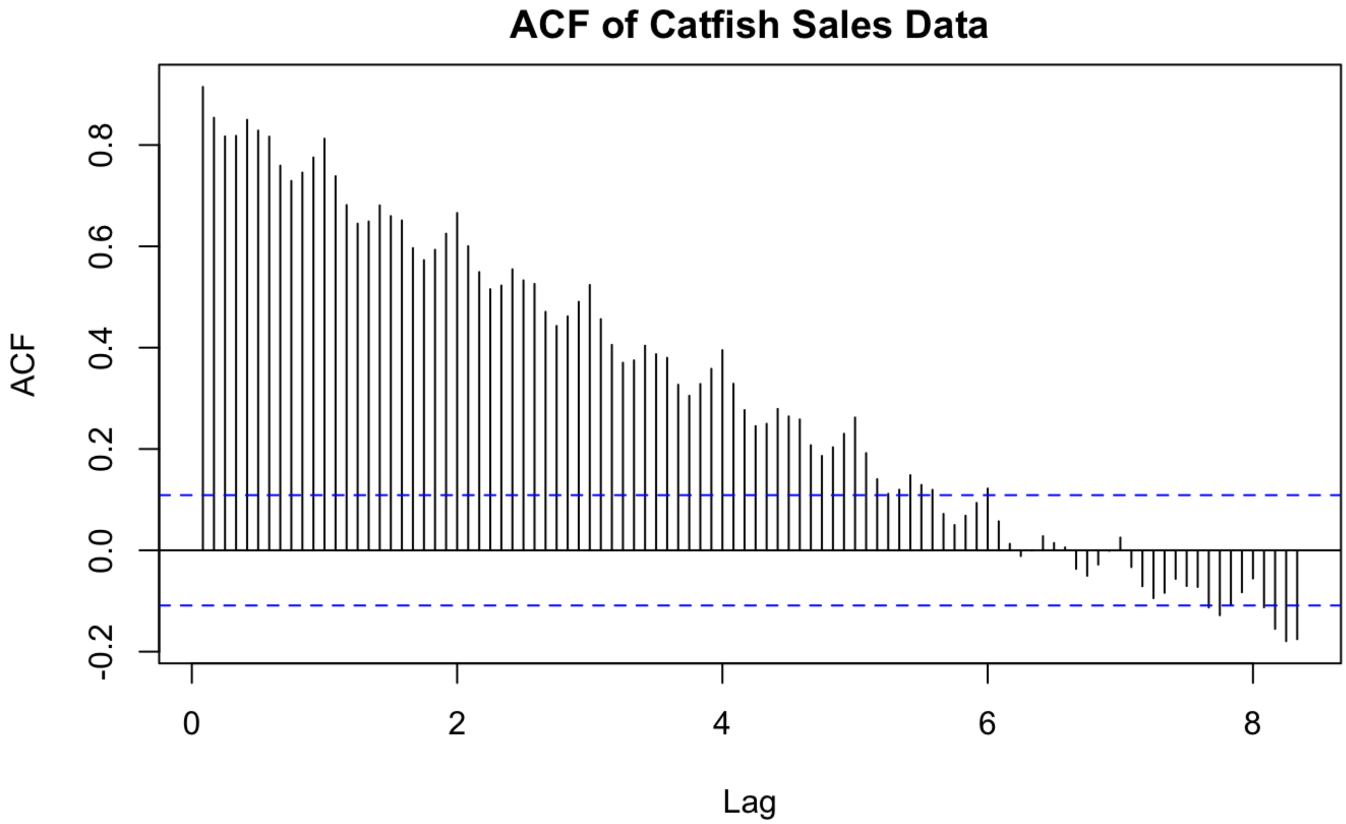
1-10 of 324 rows Previous **1** 2 3 4 5 6 ... 33 Next

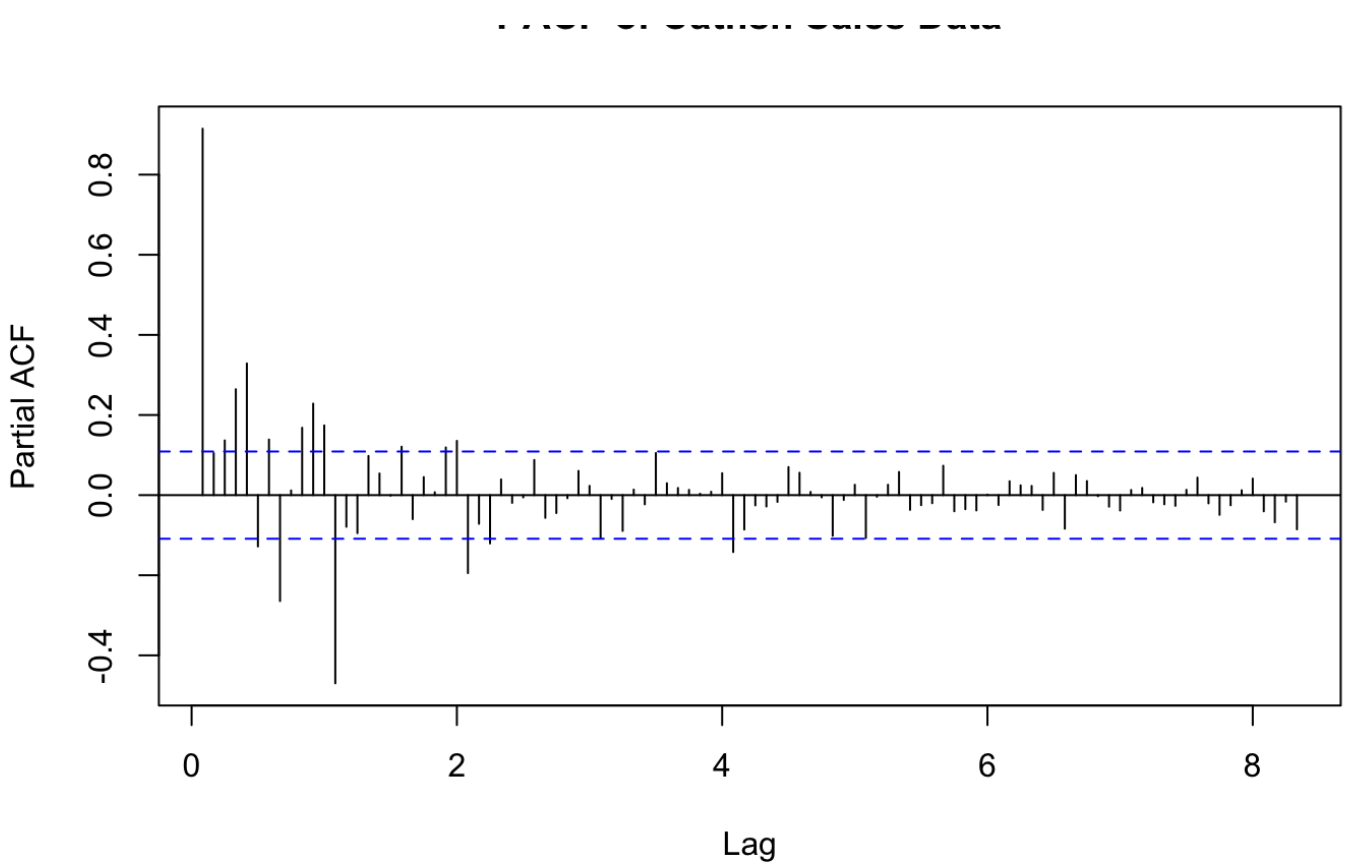
Hide

```
head(ts_s_data)
```

```
[1]  9034  9596 10558  9002  9239  8951
```

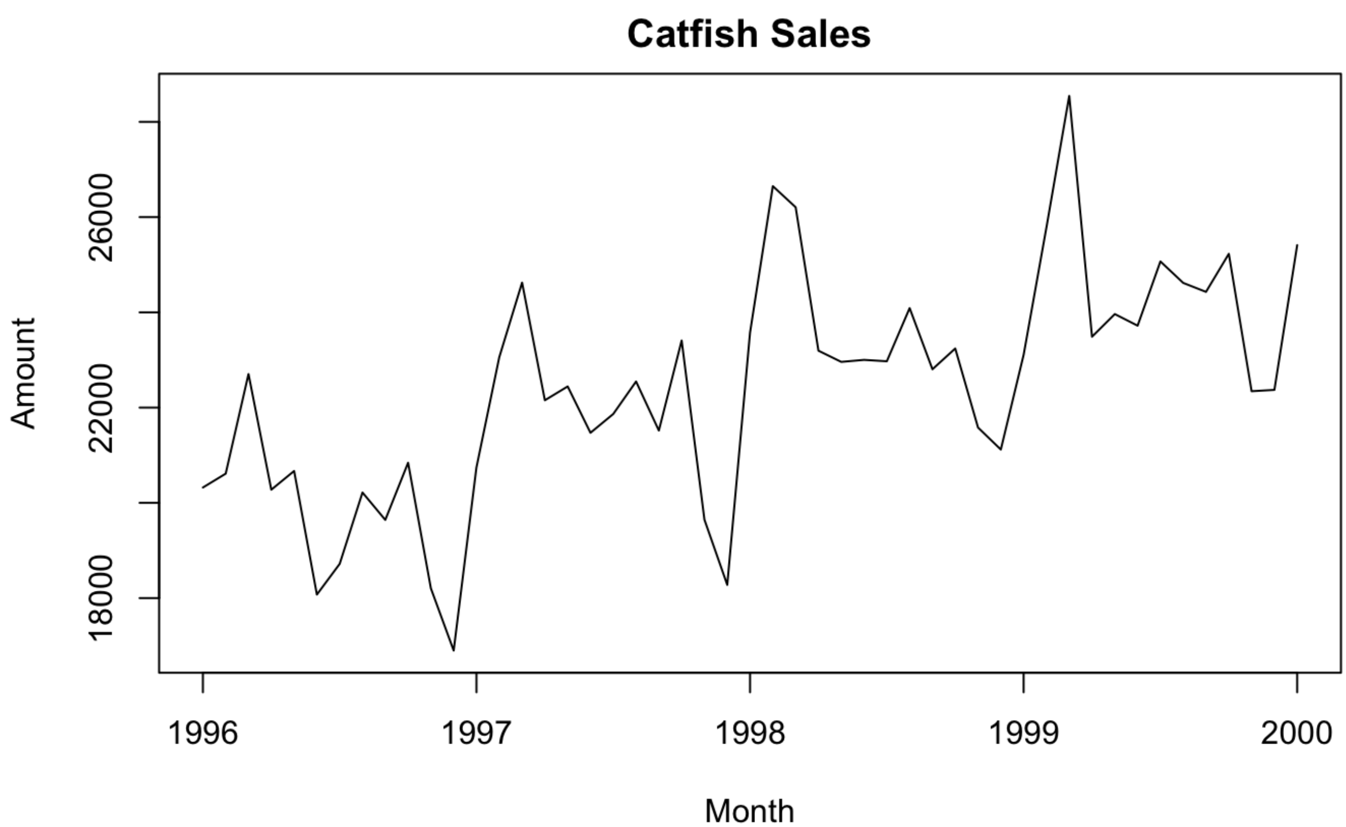



ACF of Catfish Sales Data
ACF
0.8
0.6
0.4
0.2
0.0
-0.2
0
2
4
6
8
Lag


Partial ACF
0.8
0.6
0.4
0.2
0.0
-0.4
0
2
4
6
8
Lag

	X	Year	Month	Value	Date
	<int>	<int>	<chr>	<dbl>	<chr>
121	10	1996	Jan	20322	1996-01-01
122	37	1996	Feb	20613	1996-02-01
123	64	1996	Mar	22704	1996-03-01
124	91	1996	Apr	20276	1996-04-01
125	118	1996	May	20669	1996-05-01
126	145	1996	Jun	18074	1996-06-01

6 rows



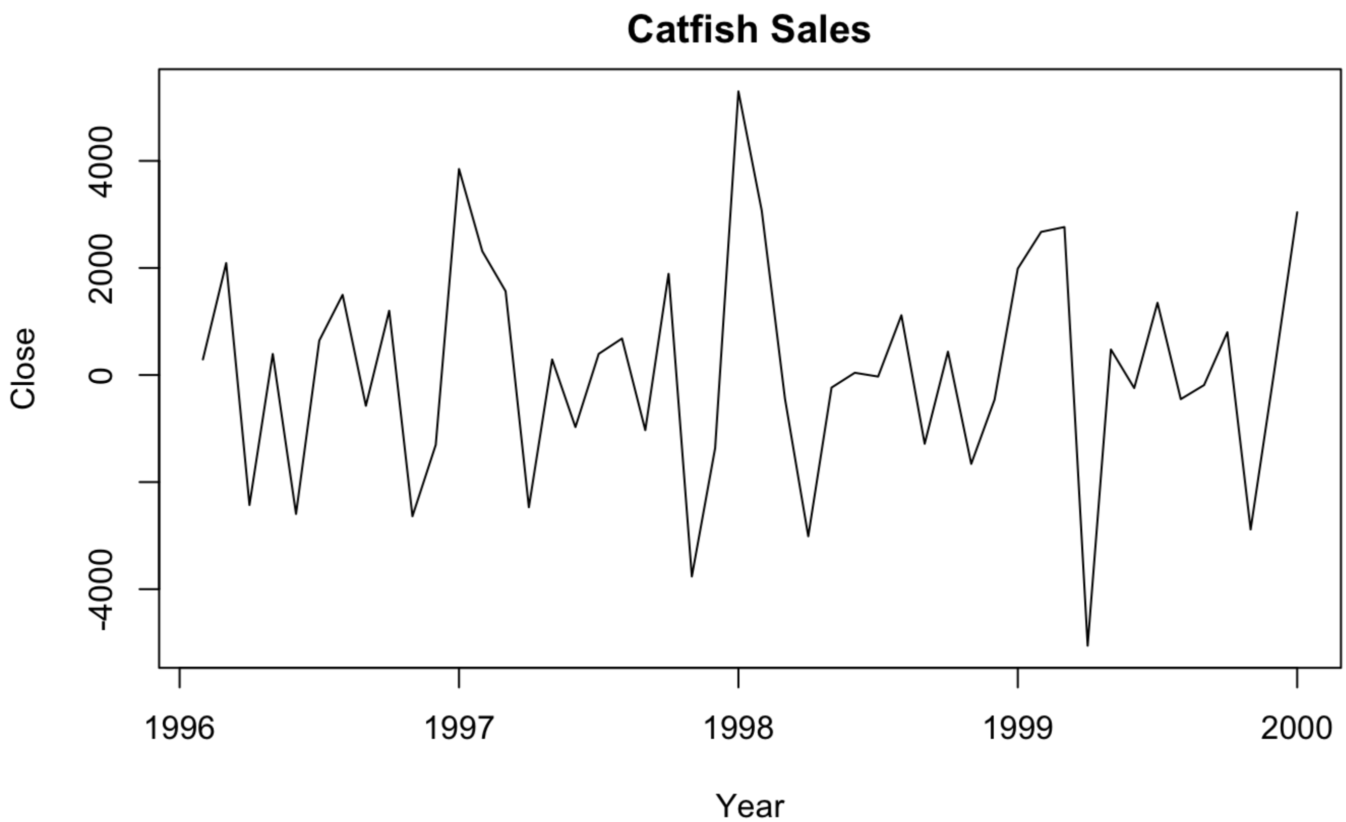
Hide

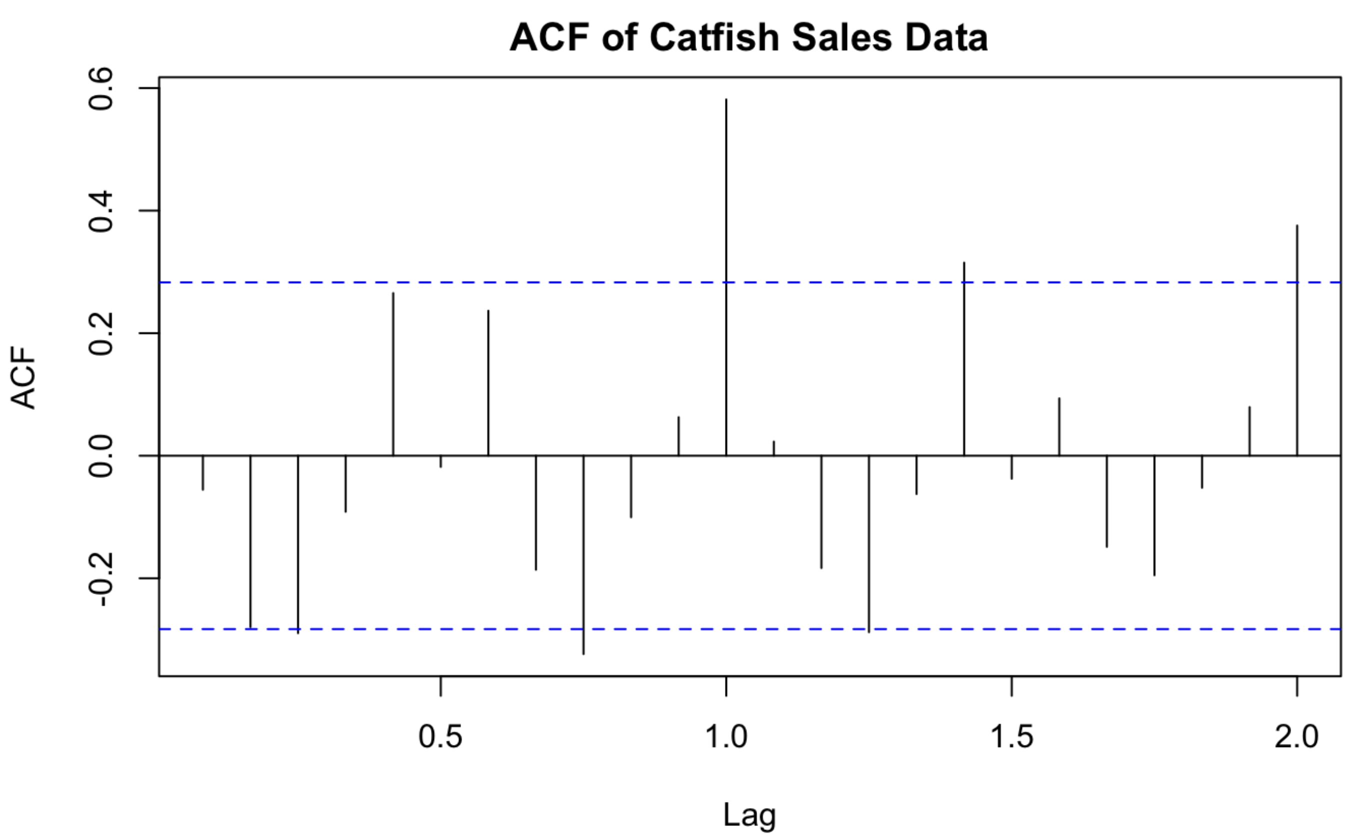
```
adf.test(ts_s_data)
```

```
	Augmented Dickey-Fuller Test

data:  ts_s_data
Dickey-Fuller = -3.7765, Lag order = 3, p-value = 0.02792
alternative hypothesis: stationary
```

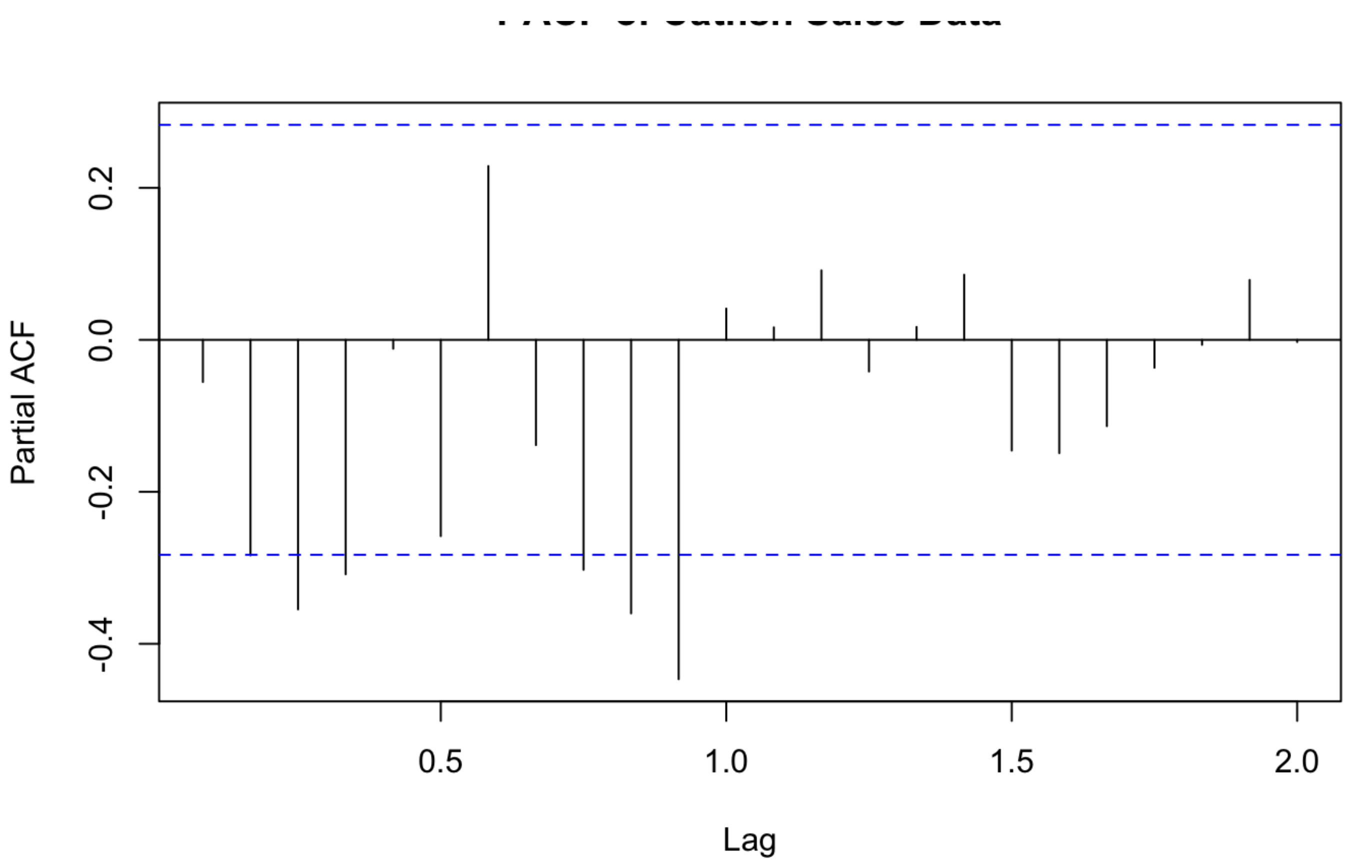
As the p-value is 0.02792 < 0.05, we reject H0, the data is stationary.

Since, we are unable to directly capture the seasonality in the data, we try to modify the data by taking difference of log of data.





Based on ACF we can see that it is following seasonal MA as there is lag at every 12 months. We also see regular MA(3) or None.



Based on PACF we do not any seasonal AR. We do see regualr AR(3), regular AR(4) and None

Hide

```
eacf(diff(ts_diff_s_data))
```

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

We try the following Models based on EACF: 1. ARIMA(3,1,3)x(0,0,1)12 2. ARIMA(4,1,3)x(0,0,1)12 3. ARIMA(0,1,0)x(0,0,1)12

Hide

```
s_model1
```

```
Call:
arima(x = ts_diff_s_data, order = c(3, 1, 3), seasonal = list(order = c(0, 0, 
    1), period = 12))

Coefficients:
          ar1     ar2      ar3      ma1      ma2     ma3    sma1
      -0.3913  0.1908  -0.1789  -1.1014  -0.7893  0.8956  0.7363
s.e.   0.2789  0.2090   0.1725   0.2592   0.4580  0.2537  0.2515

sigma^2 estimated as 1514094:  log likelihood = -410.32,  aic = 834.65
```

Hide

```
AIC(s_model1)
```

```
[1] 836.647
```

Hide

```
BIC(s_model1)
```

```
[1] 851.4481
```

Hide

```
s_model2
```

```
Call:
arima(x = ts_diff_s_data, order = c(4, 1, 3), seasonal = list(order = c(0, 0, 
    1), period = 12))

Coefficients:
          ar1     ar2      ar3      ar4      ma1      ma2     ma3    sma1
      -0.3958  0.1837  -0.2060  -0.0545  -1.1132  -0.7670  0.8843  0.7088
s.e.   0.2972  0.2158   0.1927   0.1734   0.2944   0.5104  0.2858  0.2460

sigma^2 estimated as 1523060:  log likelihood = -410.27,  aic = 836.55
```

Hide

```
AIC(s_model2)
```

```
[1] 838.5497
```

Hide

```
BIC(s_model2)
```

```
[1] 855.2011
```

Hide

```
s_model3 <- arima(ts_diff_s_data, order= c(0,1,0), seasonal=list(order=c(0,0,1), period=
12))
s_model3
```

```


Call:
arima(x = ts_diff_s_data, order = c(0, 1, 0), seasonal = list(order = c(0, 0,
    1), period = 12))

Coefficients:
        sma1
      0.8118
s.e.  0.4463

sigma^2 estimated as 4805098:  log likelihood = -433.86,  aic = 869.73
```

Hide

```
AIC(s_model3)
```

```
[1] 871.7255
```

Hide

```
BIC(s_model3)
```

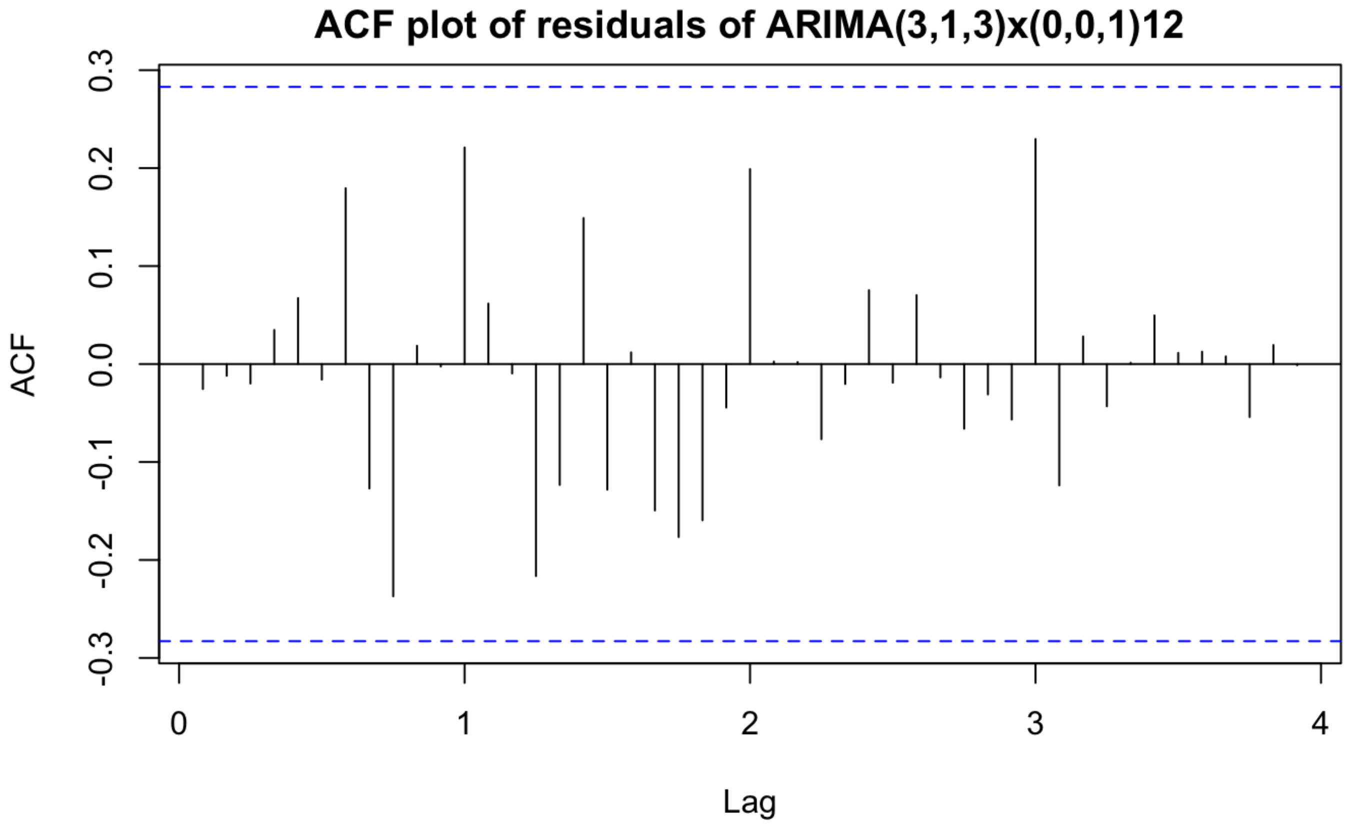
```
[1] 875.4258
```

####We go with Seasonal Model_1 as it has least AIC and BIC values.

Residual Analysis

Hide

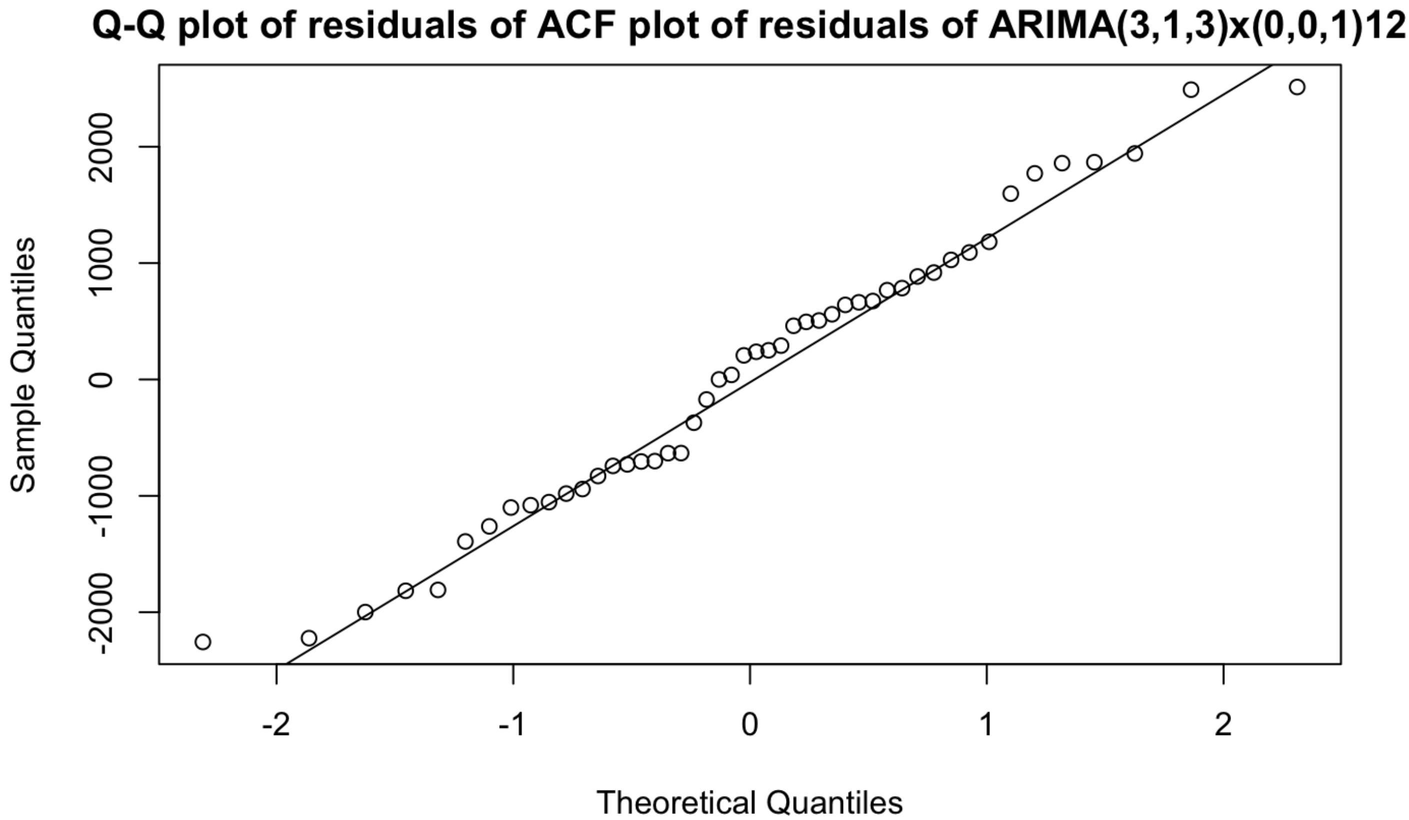
```
s_model <- arima(ts_diff_s_data, order= c(3,1,3), seasonal=list(order=c(0,0,1), period=
12))
acf(residuals(s_model), lag.max = 100, main = "ACF plot of residuals of ARIMA(3,1,3)x(0,
0,1)12")
```



We se no significant lags in ACF of residuals

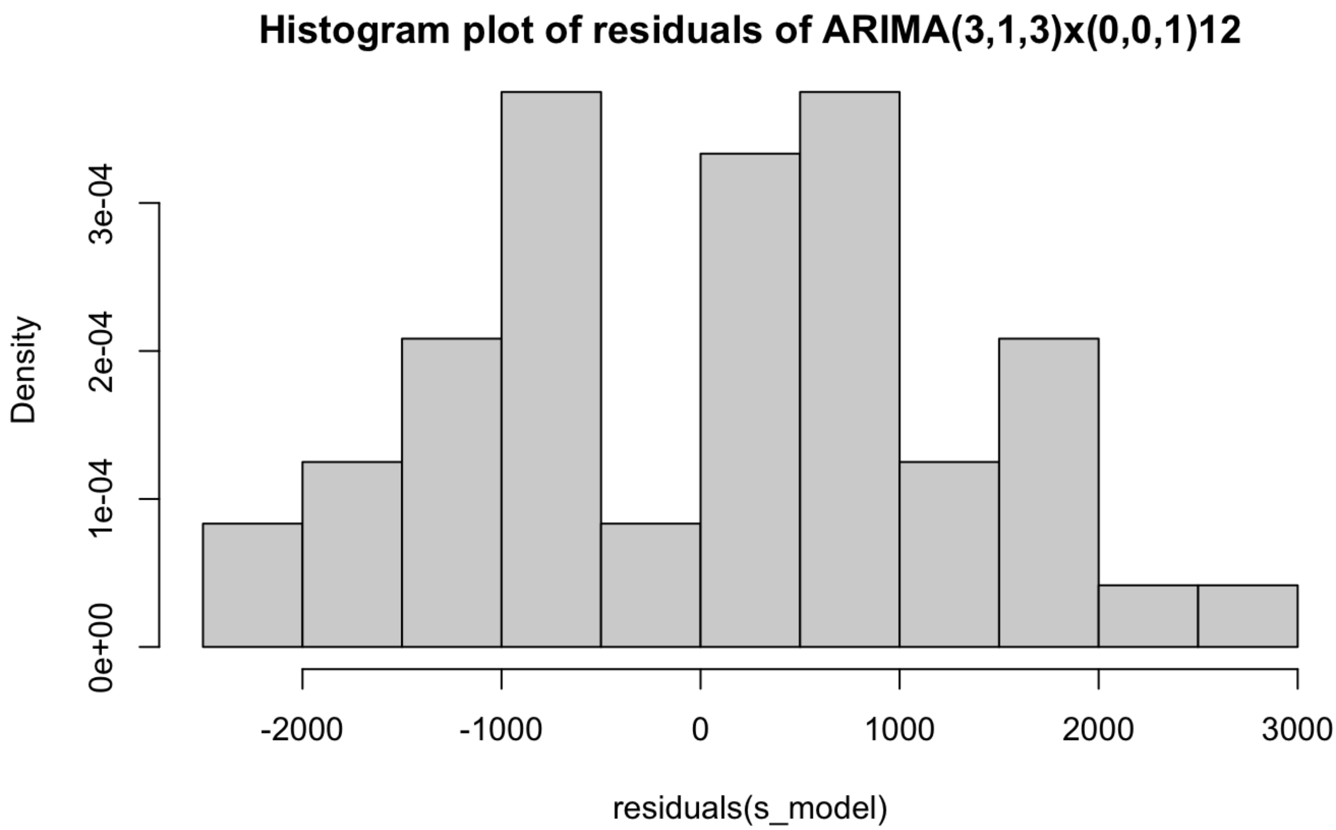
Hide

```
qqnorm(residuals(s_model), main = "Q-Q plot of residuals of ACF plot of residuals of ARI
MA(3,1,3)x(0,0,1)12"); qqline(residuals(s_model))
```



Hide

```
hist(residuals(s_model), freq = FALSE, main = "Histogram plot of residuals of ARIMA(3,1,
3)x(0,0,1)12")
```



Hide

```
shapiro.test(residuals(s_model))
```

```
	Shapiro-Wilk normality test

data:  residuals(s_model)
W = 0.97704, p-value = 0.462
```

From the Shapiro-Wilk test, the p-value of 0.462 > 0.05, shows that the residual is normal.

Hide

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

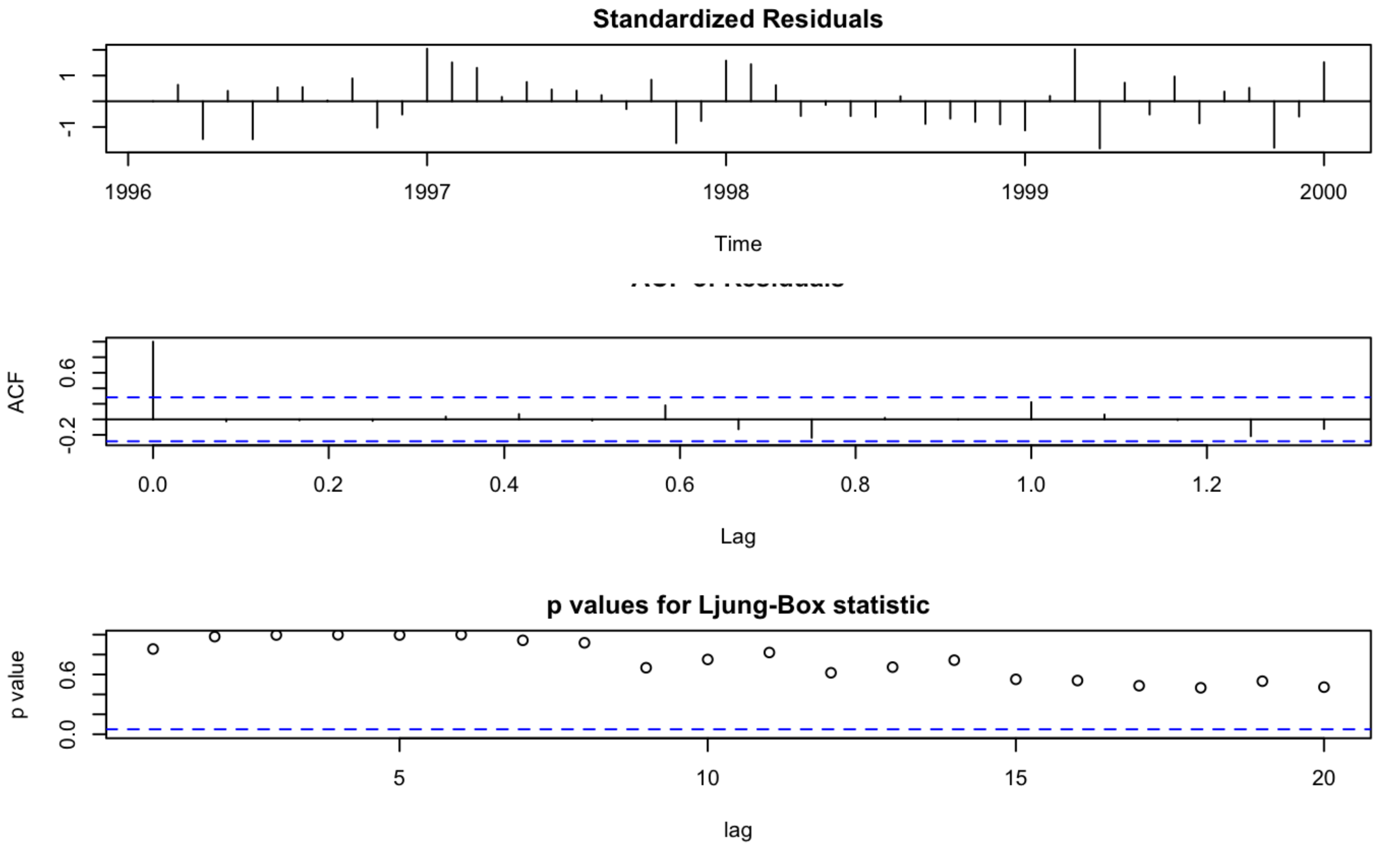
```
	Box-Ljung test

data:  residuals(s_model)
X-squared = 6.7306, df = 10, p-value = 0.7506
```

The Box-Ljung test, having p-value 0.5455 > 0.05, shows that the residuals are independent and identically distributed.

Diagnostic plot of ARIMA(3,1,3)x(0,0,1)12

Hide

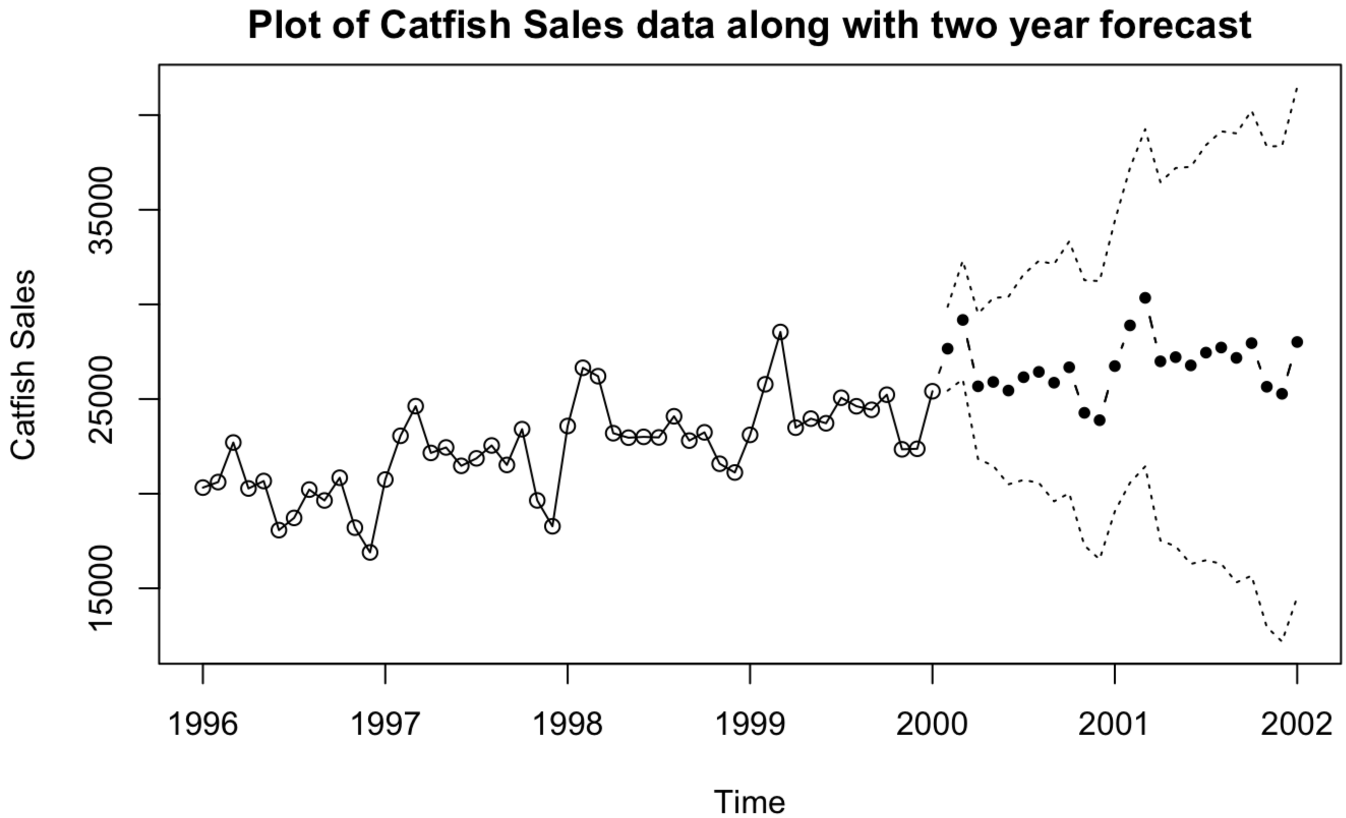
```
tsdiag(s_model, gof.lag = 20)
```



Forecast

Hide

```
s_model <- arima(ts_s_data, order= c(0,1,0), seasonal=list(order=c(1,0,1), period= 12))
plot(s_model, n1=c(1996,1), n.ahead=24,ylab='Catfish Sales',pch=20, main = "Plot of Catf
ish Sales data along with two year forecast")
```



Conclusion

We can see that SARIMA(3,1,3)x(0,0,1)[12] is a great fit to the data, and is able to forecast the Catfish Sales by capturing the seasonality trends.

References

1. Cryer, J. D., & Chan, K. S. (2008). Time series analysis: with applications in R (Vol. 2). New York: Springer.
2. Katesari, H. S., & Zarodi, S. (2016). Effects of coverage choice by predictive modeling on frequency of accidents. Caspian Journal of Applied Sciences Research, 5(3), 28-33.
3. Safari-Katesari, H., Samadi, S. Y., & Zaroudi, S. (2020). Modelling count data via copulas. Statistics, 54(6), 1329-1355.
4. Shumway, R. H., Stoffer, D. S., & Stoffer, D. S. (2000). Time series analysis and its applications (Vol. 3). New York: springer.
5. Safari-Katesari, H., & Zaroudi, S. (2020). Count copula regression model using generalized beta distribution of the second kind. Statistics, 21, 1-12.
6. Safari-Katesari, H., & Zaroudi, S. (2021). Analysing the impact of dependency on conditional survival functions using copulas. Statistics in Transition New Series, 22(1).
7. Safari Katesari, H., (2021) Bayesian dynamic factor analysis and copula-based models for mixed data, PhD dissertation, Southern Illinois University Carbondale