MA641 Project Scratch

Brianna Cirillo

Libraries

```
# Load necessary libraries
library(forecast)
```

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

```
library(tseries)
library(TSA)
```

```
## Registered S3 methods overwritten by 'TSA':
##   method       from
##   fitted.Arima forecast
##   plot.Arima   forecast

##
## Attaching package: 'TSA'

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

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

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.1     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x readr::spec()   masks TSA::spec()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

FRED Monthly Retail Sales

Load & Inspect the Data

```r
# Load in the data
data <- read.csv("~/Documents/GitHub/MA-641-Course-Project/FRED_monthly_retail_sales.csv",header=T)

# Convert the date column to Date type
data$DATE <- as.Date(data$DATE, format="%Y-%m-%d")

# Inspect the data
head(data)
```

```
##         DATE RSXFSN
## 1 2018-01-01 389624
## 2 2018-02-01 380456
## 3 2018-03-01 444725
## 4 2018-04-01 420206
## 5 2018-05-01 464422
## 6 2018-06-01 443762
```

```r
summary(data)
```

```
##       DATE                RSXFSN
##  Min.   :2018-01-01   Min.   :376261
##  1st Qu.:2019-08-01   1st Qu.:455157
##  Median :2021-03-01   Median :518796
##  Mean   :2021-03-01   Mean   :518801
##  3rd Qu.:2022-10-01   3rd Qu.:592577
##  Max.   :2024-05-01   Max.   :668957
```

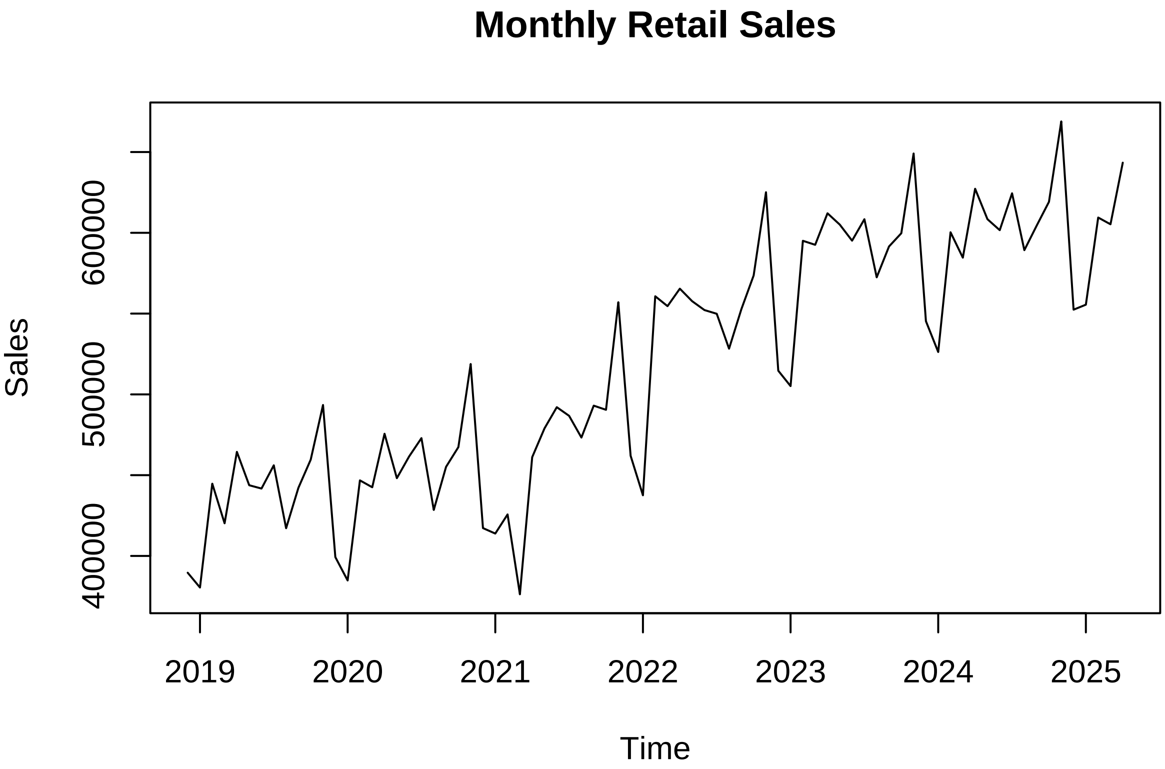
- The dataset spans from January 2018 to May 2024, covering over six years of monthly data.
- The retail sales values range from 376,261 to 668,957.
- The median and mean values are close to each other (around 518,800), suggesting a symmetric distribution.
- The interquartile range (IQR) is the difference between the 3rd and 1st quartiles (592,577 - 455,157 = 137,420), which measures the spread of the middle 50% of the data.
- There is a noticeable increase in retail sales over time, as indicated by the higher values in the upper quartiles compared to the lower quartiles.

Create a Time Series Object

```r
# Create a time series object
ts_data <- ts(data$RSXFSN, start=c(2018, 12), frequency=12)
```

Descriptive Analysis

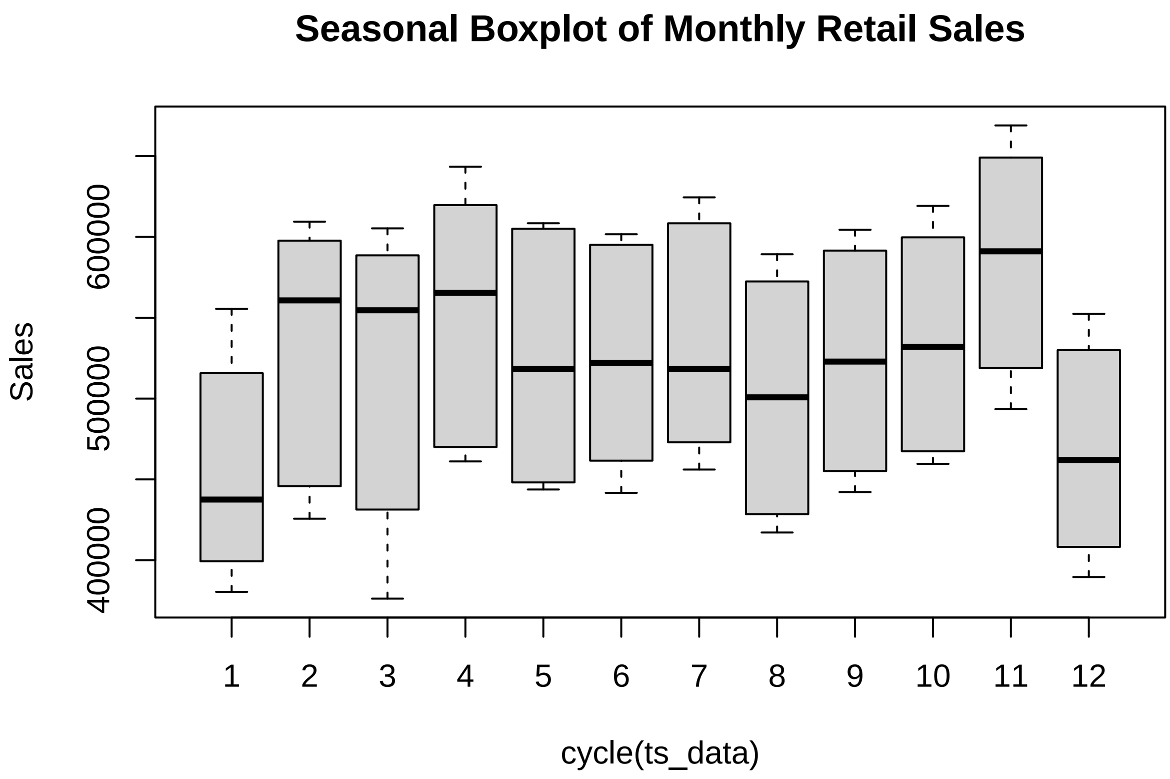
```
# Descriptive Analysis
plot(ts_data, main="Monthly Retail Sales", ylab="Sales", xlab="Time")
```



```
summary(ts_data)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##  376261  455157  518796  518801  592577  668957
```

```
boxplot(ts_data ~ cycle(ts_data), main="Seasonal Boxplot of Monthly Retail Sales", ylab="Sales")
```



Time Series Plot:

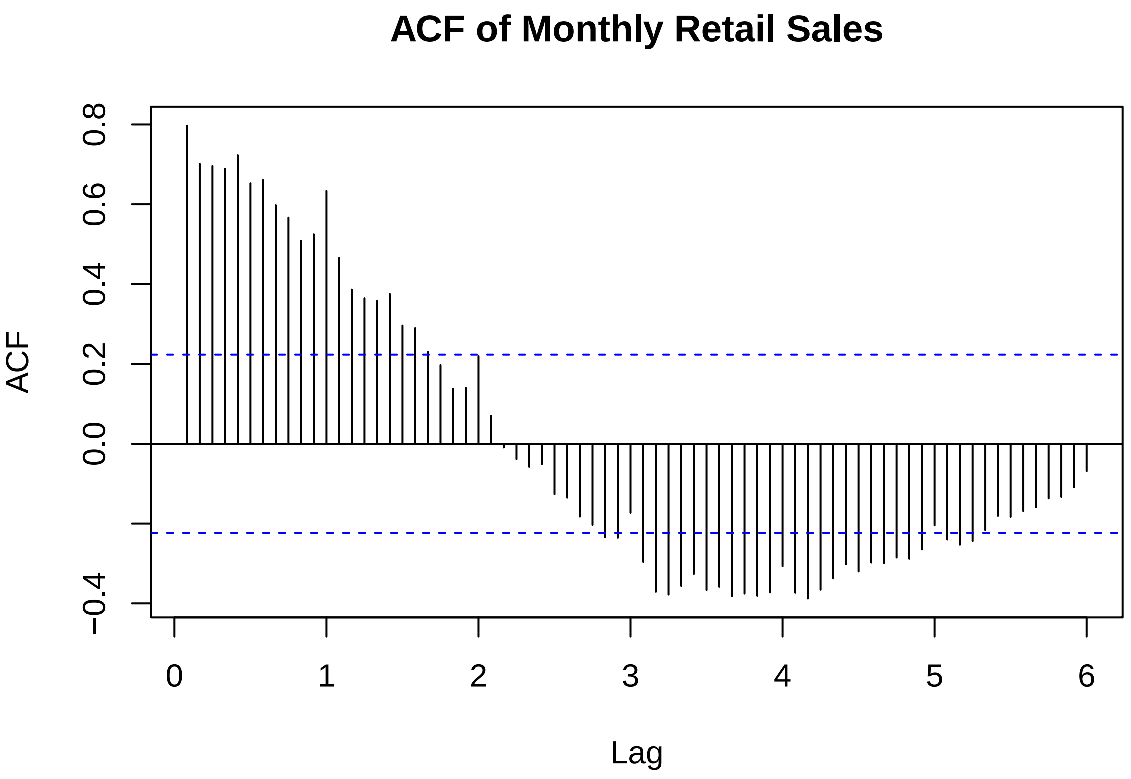
- There is an upward trend in retail sales over the period from 2018 to 2024. This indicates that retail sales have been generally increasing over time.
- There are repeating patterns within each year, indicative of seasonality.
- The amplitude of fluctuations increases over time, indicating growing volatility in sales.

Seasonal Boxplots:

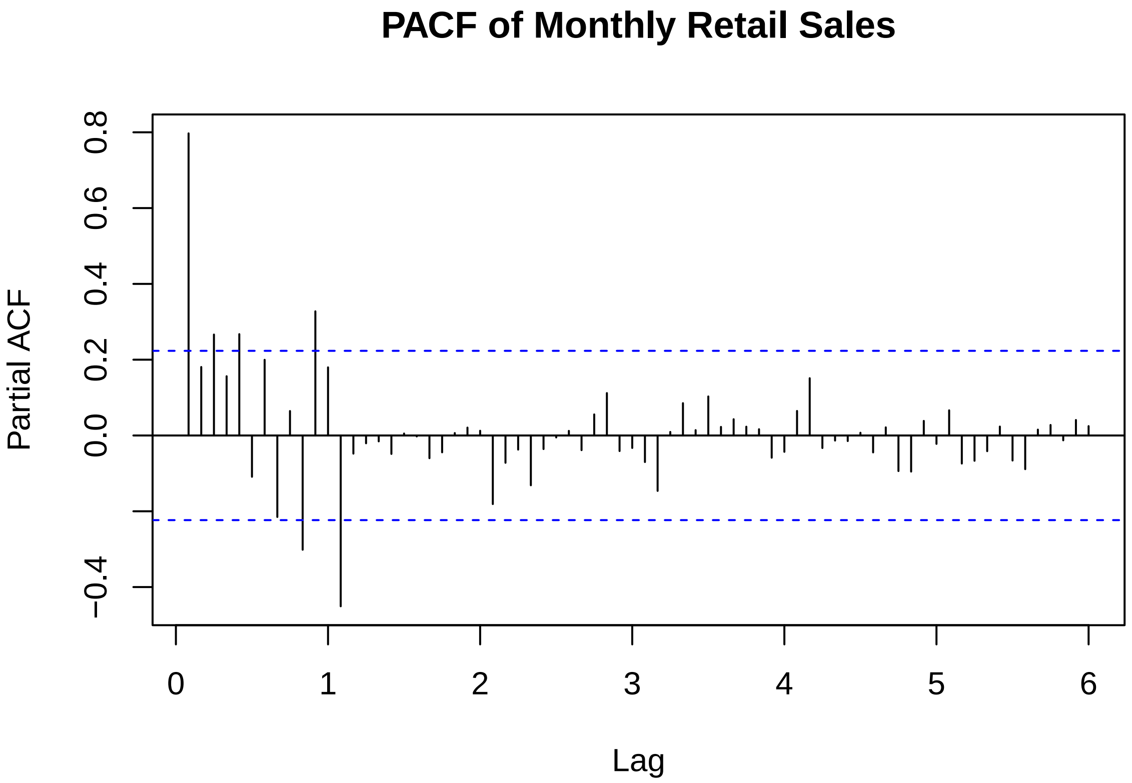
- There is evident seasonal variation in monthly retail sales. For example, January (month 1) tends to have lower sales, while November (month 11) shows higher sales.
- The interquartile range (IQR) and the median values vary significantly across months, indicating changes in sales patterns throughout the year.
- The presence of outliers can indicate unusual sales activity in certain months.
- November has the highest median sales, indicating consistently higher sales in this month.

ACF and PACF Plot

```
# ACF and PACF Plots
par(mar=c(5, 5, 4, 2) + 0.1)
acf(ts_data, main="ACF of Monthly Retail Sales", lag.max = 72)
```



```
pacf(ts_data, main="PACF of Monthly Retail Sales", lag.max = 72)
```



ACF:

- The ACF plot shows significant autocorrelation at multiple lags, with the autocorrelation values gradually decreasing. This is indicative of a trend component in the data.
- The significant autocorrelations at the first few lags suggest that the past values have a strong influence on the current value of the series.
- The slow decay of the ACF values is typical of non-stationary time series data, where the mean and variance are not constant over time. This pattern suggests that the time series might benefit from differencing to achieve stationarity.
- There is also a noticeable seasonal pattern, as indicated by the peaks at regular intervals (around lag 12 and multiples), which is consistent with the monthly data having a yearly seasonal component.

PACF:

- The PACF plot shows a significant spike at lag 1, with values dropping off sharply afterward. This indicates that an AR(1) component might be appropriate for the model.
- The sharp drop after the first lag in the PACF plot is typical of an autoregressive process, suggesting that the AR component is sufficient to explain the dependencies in the series.
- There is also a smaller spike at lag 12, which supports the presence of seasonality in the data.

Augmented Dickey-Fuller Test

```r
# Augmented Dickey-Fuller Test
adf_test <- adf.test(ts_data, alternative="stationary")
print(adf_test)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts_data
## Dickey-Fuller = -2.4516, Lag order = 4, p-value = 0.3911
## alternative hypothesis: stationary
```

- The p-value is 0.3911, which is greater than the significance level of 0.05. This means that we fail to reject the null hypothesis at these levels.
- We do not have enough evidence to reject the null hypothesis of non-stationarity. Therefore, we conclude that the time series ts_data is non-stationary.
- Since the time series is non-stationary, we need to make it stationary for modeling purposes.

```r
# Differencing the series if it is not stationary
if (adf_test$p.value > 0.05) {
  ts_data_diff <- diff(ts_data, differences=1)
  adf_test_diff <- adf.test(ts_data_diff, alternative="stationary")
  print(adf_test_diff)

  # Update the time series data to the differenced series
  ts_data <- ts_data_diff
}
```

```
## Warning in adf.test(ts_data_diff, alternative = "stationary"): p-value smaller
## than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts_data_diff
## Dickey-Fuller = -5.7648, Lag order = 4, p-value = 0.01
## alternative hypothesis: stationary
```

- The p-value is 0.01, which means that we reject the null hypothesis at these levels. Therefore, we conclude that the differenced time series ts_data_diff is stationary.

MA Model

```
# Fit an MA model
ma_model <- Arima(ts_data, order=c(0,0,1))
summary(ma_model)
```

```
## Series: ts_data
## ARIMA(0,0,1) with non-zero mean
##
## Coefficients:
##           ma1       mean
##       -0.8255  2734.6795
## s.e.   0.0735   801.9453
##
## sigma^2 = 1.442e+09:  log likelihood = -908.78
## AIC=1823.57   AICc=1823.9   BIC=1830.56
##
## Training set error measures:
##                     ME     RMSE      MAE      MPE     MAPE     MASE      ACF1
## Training set 778.8198 37467.15 29187.57 15.00939 195.0946 1.775671 0.1507753
```

- The ma1 coefficient is significant, indicating that the moving average term is meaningful.
- ME close to zero, and reasonably low RMSE, MAE, and MAPE values suggest that the model forecasts well, though the high RMSE might be a concern depending on the context.
- The ACF1 value suggests some autocorrelation in residuals, implying that there might be room for improvement.

Overall: The MA(1) model appears to be a reasonable fit for the data, capturing the main patterns and trends. However, the presence of some autocorrelation in the residuals and the relatively high RMSE indicate that it might not be the best possible model.

AR Model

```
# Fit an AR model
ar_model <- Arima(ts_data, order=c(1,0,0))
summary(ar_model)
```

```
## Series: ts_data
## ARIMA(1,0,0) with non-zero mean
```

```
##
## Coefficients:
##           ar1      mean
##       -0.3013  3271.706
## s.e.   0.1090  3774.619
##
## sigma^2 = 1.872e+09:  log likelihood = -918.18
## AIC=1842.36   AICc=1842.69   BIC=1849.35
##
## Training set error measures:
##                      ME     RMSE      MAE      MPE     MAPE     MASE      ACF1
## Training set -42.32456 42691.09 32511.36 82.87834 158.4063 1.977879 -0.108273
```

- The ar1 coefficient is significant, with a high value indicating a strong autoregressive component.
- AIC, AICc, and BIC values are lower than those for the MA(1) model, suggesting a better fit.
- Lower RMSE, MAE, and MAPE compared to the MA(1) model indicate better predictive performance.
- The ACF1 value is closer to zero, indicating less autocorrelation in residuals.

Overall: The AR(1) model appears to be a better fit than the MA(1) model for this dataset. It captures the main patterns and trends more accurately, with lower error measures and better information criteria values. However, the prediction intervals suggest some caution in the confidence of future values.

ARMA Models

```
# Fit an ARMA(1,1) model as a starting point
arma_model <- Arima(ts_data, order=c(1,0,1))
summary(arma_model)
```

```
## Series: ts_data
## ARIMA(1,0,1) with non-zero mean
##
## Coefficients:
##          ar1      ma1       mean
##       0.3052  -1.0000  3042.9285
## s.e.  0.1103   0.0519   261.5953
##
## sigma^2 = 1.352e+09:  log likelihood = -907.12
## AIC=1822.25   AICc=1822.81   BIC=1831.57
##
## Training set error measures:
##                     ME     RMSE     MAE      MPE     MAPE     MASE       ACF1
## Training set -409.736 36039.84 27967.1 39.91866 171.6811 1.701422 0.01296858
```

The ARMA(1,1) model appears to be a good fit for the data, capturing the main patterns and trends accurately. It provides a better fit compared to the MA(1) model and is comparable to or slightly better than the AR(1) model, especially considering the lower information criteria values and the low autocorrelation of residuals. Further comparison with other ARMA configurations or additional model validation could provide more assurance, but overall, this model seems robust.

```r
# Fit ARIMA model
sarima_model <- auto.arima(ts_data)
summary(sarima_model)
```

```
## Series: ts_data
## ARIMA(0,0,0)(2,1,0)[12]
##
## Coefficients:
##          sar1    sar2
##       -0.6486  -0.350
## s.e.   0.1158   0.119
##
## sigma^2 = 397336120:  log likelihood = -726.55
## AIC=1459.1   AICc=1459.5   BIC=1465.57
##
## Training set error measures:
##                    ME     RMSE      MAE      MPE     MAPE      MASE        ACF1
## Training set 239.9244 18003.96 11548.68 7.263603 94.06953 0.7025818 -0.07713744
```

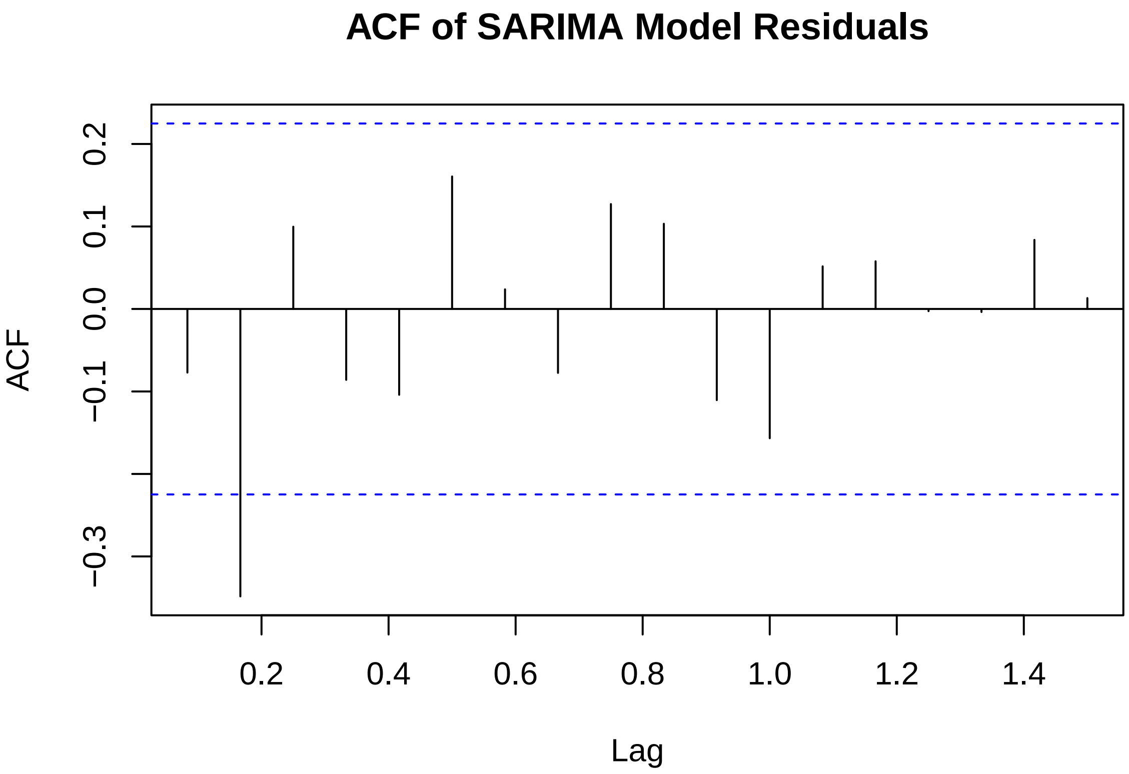
Model: ARIMA(0,0,0)(2,1,0)[12] with drift

This indicates a SARIMA model with:

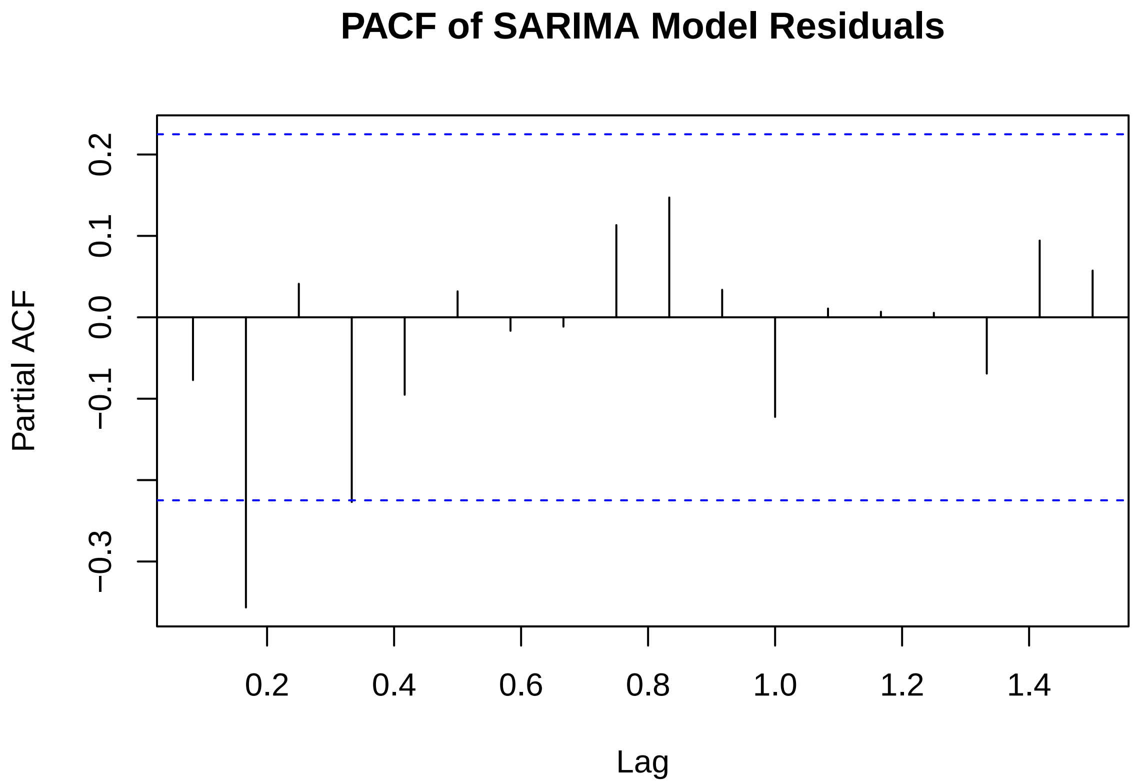
- No non-seasonal autoregressive (AR) terms (p=0)
- No differencing (d=0)
- No non-seasonal moving average term (MA=0)
- Seasonal autoregressive terms of order 2 (SAR(2))
- Seasonal differencing of order 1 (D=1)
- No seasonal moving average term (SMA=0)
- Seasonal period of 12 (monthly data)
- A drift term
- The coefficients for the AR and seasonal AR terms are significant, suggesting that the seasonal components at lags 12 and 24 months play a crucial role in explaining the variations in the data.
- AIC, AICc, and BIC values are much lower than those for the previous models, suggesting a significantly better fit.
- RMSE, MAE, and MAPE values are much lower than those of the previous models, indicating excellent predictive performance.
- The ACF1 value close to zero indicates minimal autocorrelation in residuals, suggesting a good fit.
- The intervals are narrower compared to previous models, indicating higher confidence in the forecasts. Overall: The SARIMA(0,0,0)(2,1,0)[12] with drift model appears to captures the main patterns and trends accurately, with significantly lower error measures and better information criteria values compared to the MA(1), AR(1), and ARMA(1,1) models. The narrow prediction intervals further suggest high confidence in the future forecasts. This model is robust and well-suited for the time series data.

ACF, PACF, EACF Plots for SARIMA

```
# Plot Residuals of the SARIMA Model
acf_residuals <- residuals(sarima_model)
par(mar=c(5, 5, 4, 2) + 0.1)
acf(acf_residuals, main="ACF of SARIMA Model Residuals")
```



```
pacf(acf_residuals, main="PACF of SARIMA Model Residuals")
```



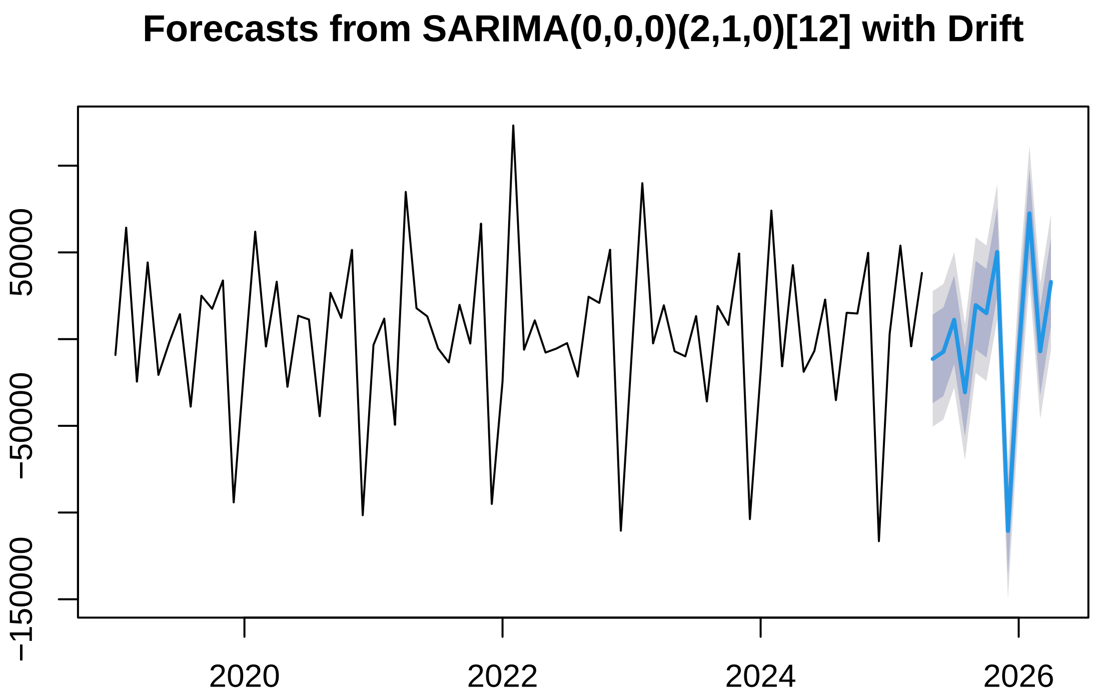
```
eacf(acf_residuals)
```

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

- Most autocorrelations and partial autocorrelations are within the blue dashed significance bounds, indicating that the residuals do not exhibit significant autocorrelation. There are no strong patterns or significant spikes outside the bounds, suggesting that the residuals are approximately white noise.
- This plot suggests that a model with low AR and MA orders may be appropriate, given the clean separation of significant and non-significant correlations.
- The model has successfully captured the main structure of the data, including trends and seasonality.
- The EACF plot supports the model selection by identifying the appropriate AR and MA orders. It provides additional confirmation of the model's adequacy.

Forecasting

```
# Forecasting with the SARIMA model
sarima_forecast <- forecast(sarima_model, h=12)
plot(sarima_forecast, main="Forecasts from SARIMA(0,0,0)(2,1,0)[12] with Drift")
```



- The forecast for the next 12 months indicates a steady increase in monthly retail sales, with prediction intervals providing a range of expected values. This information can be useful for planning and decision-making purposes in the retail sector.

Model Comparison

```
# Comparing Models using AIC and BIC
models <- list(ar_model, ma_model, arma_model, sarima_model)
model_names <- c("AR(1)", "MA(1)", "ARMA(1,1)", "SARIMA(0,0,0)(2,1,0)[12] with Drift")

aic_values <- sapply(models, AIC)
bic_values <- sapply(models, BIC)

comparison <- data.frame(Model=model_names, AIC=aic_values, BIC=bic_values)
print(comparison)
```

```
##                                 Model      AIC      BIC
## 1                               AR(1) 1842.359 1849.351
## 2                               MA(1) 1823.568 1830.560
## 3                           ARMA(1,1) 1822.245 1831.568
## 4 SARIMA(0,0,0)(2,1,0)[12] with Drift 1459.097 1465.574
```

- The SARIMA(0,0,0)(2,1,0)[12] with Drift model was chosen as the best model based on AIC and BIC values. The model captures both the seasonal and non-seasonal components of the data effectively, and the residuals appear to be white noise, indicating a good fit.

AAPL Monthly Data

Load & Inspect the Data

```
# Load the data
data <- read.csv("~/Documents/GitHub/MA-641-Course-Project/AAPL_monthly.csv")

# Convert the date column to Date type
data$Date <- as.Date(data$Date, format="%Y-%m-%d")

# Inspect the data
head(data)
```

```
##         Date    Open    High     Low   Close Adj.Close     Volume
## 1 2018-01-01 42.5400 45.0250 41.1750 41.8575  39.47705 2638717600
## 2 2018-02-01 41.7925 45.1550 37.5600 44.5300  41.99757 3711577200
## 3 2018-03-01 44.6350 45.8750 41.2350 41.9450  39.72087 2854910800
## 4 2018-04-01 41.6600 44.7350 40.1575 41.3150  39.12428 2664617200
## 5 2018-05-01 41.6025 47.5925 41.3175 46.7175  44.24030 2483905200
## 6 2018-06-01 46.9975 48.5500 45.1825 46.2775  43.99263 2110498000
```

```
summary(data)
```

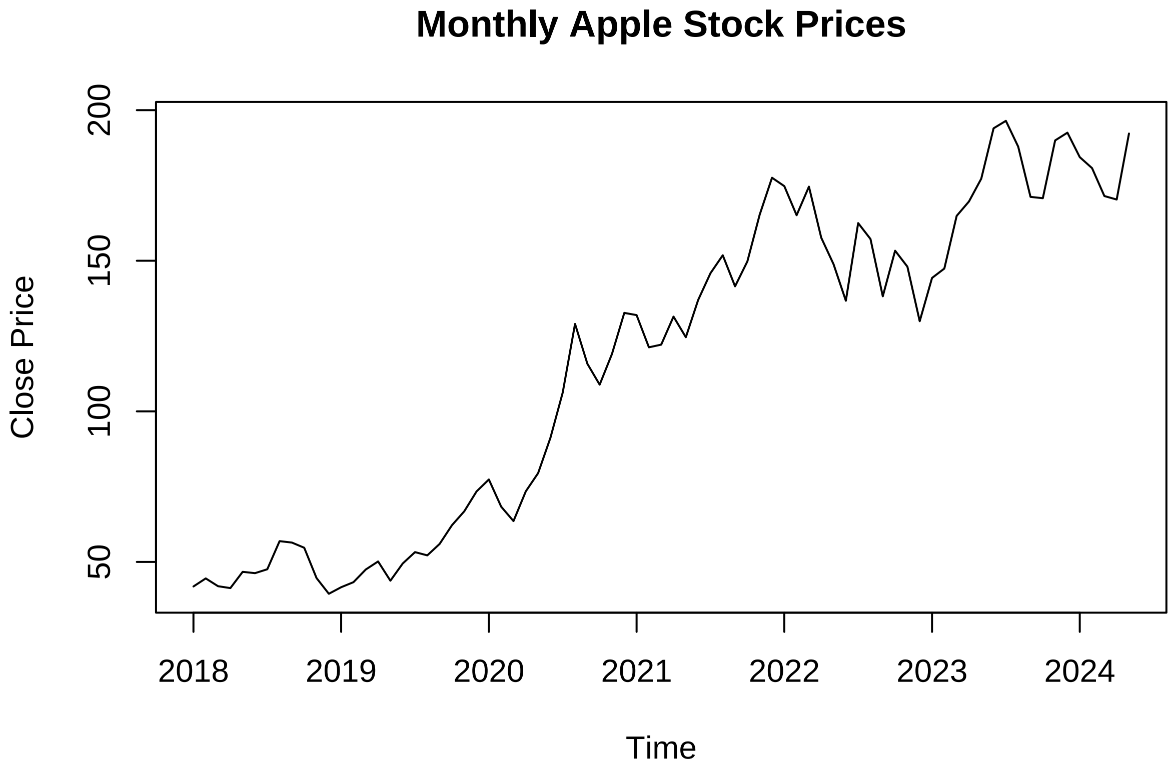
```
##       Date                 Open             High             Low        
##  Min.   :2018-01-01   Min.   : 38.72   Min.   : 42.25   Min.   : 35.50  
##  1st Qu.:2019-08-01   1st Qu.: 56.27   1st Qu.: 57.42   1st Qu.: 51.52  
##  Median :2021-03-01   Median :130.28   Median :137.88   Median :122.25  
##  Mean   :2021-03-01   Mean   :114.12   Mean   :122.20   Mean   :107.24  
##  3rd Qu.:2022-10-01   3rd Qu.:161.01   3rd Qu.:169.85   3rd Qu.:150.10  
##  Max.   :2024-05-01   Max.   :196.24   Max.   :199.62   Max.   :187.45  
##      Close          Adj.Close         Volume         
##  Min.   : 39.44   Min.   : 37.75   Min.   :9.697e+08  
##  1st Qu.: 56.44   1st Qu.: 54.10   1st Qu.:1.520e+09  
##  Median :129.93   Median :128.87   Median :2.024e+09  
##  Mean   :115.84   Mean   :114.01   Mean   :2.186e+09  
##  3rd Qu.:164.90   3rd Qu.:162.77   3rd Qu.:2.716e+09  
##  Max.   :196.45   Max.   :195.41   Max.   :6.280e+09  
```

Create a Time Series Object

```
# Create a time series object
ts_data <- ts(data$Close, start=c(2018, 01), frequency=12)
```

Descripvtive Analysis

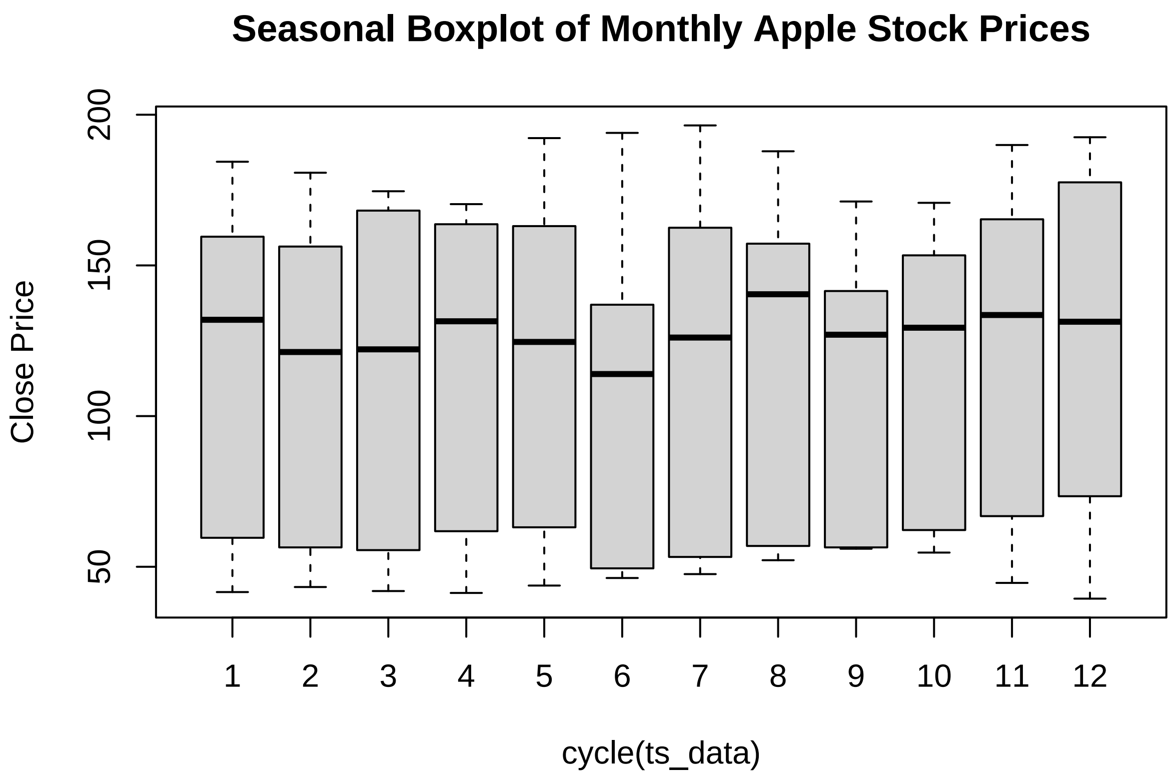
```
# Descriptive Analysis
plot(ts_data, main="Monthly Apple Stock Prices", ylab="Close Price", xlab="Time")
```



```
summary(ts_data)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   39.44   56.44  129.93  115.84  164.90  196.45
```

```
boxplot(ts_data ~ cycle(ts_data), main="Seasonal Boxplot of Monthly Apple Stock Prices", ylab="Close Pri
```



Time Series Plot:

- There is a clear upward trend in Apple stock prices over the period. The prices show a substantial increase, particularly starting around 2020.
- There is visible volatility in the stock prices, with fluctuations becoming more pronounced in the later years.
- The increased volatility may imply higher risk for investors, as the stock prices have larger swings.

Summary:

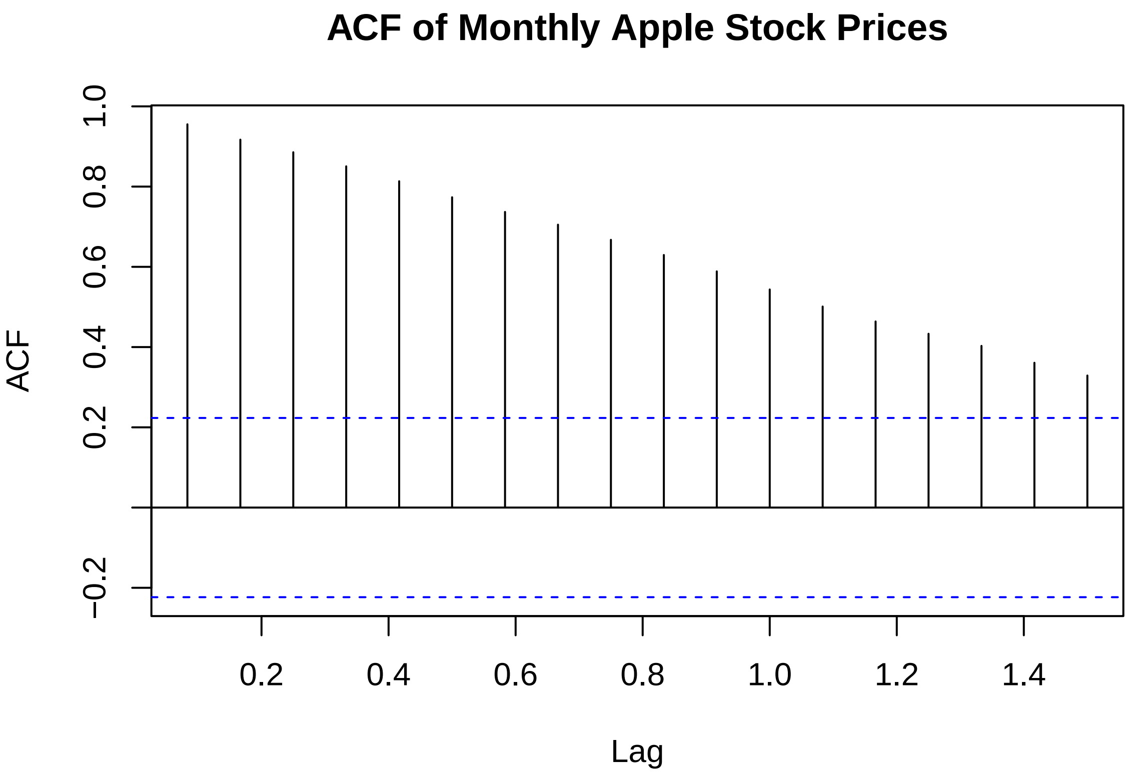
- The mean and median values suggest that the central tendency of the stock prices is around 115 to 130.
- The range indicates that the stock price has varied significantly over the period.
- The interquartile range (IQR = Q3 - Q1) shows the spread of the middle 50% of the data, which is a useful measure of variability.

Seasonal Boxplot:

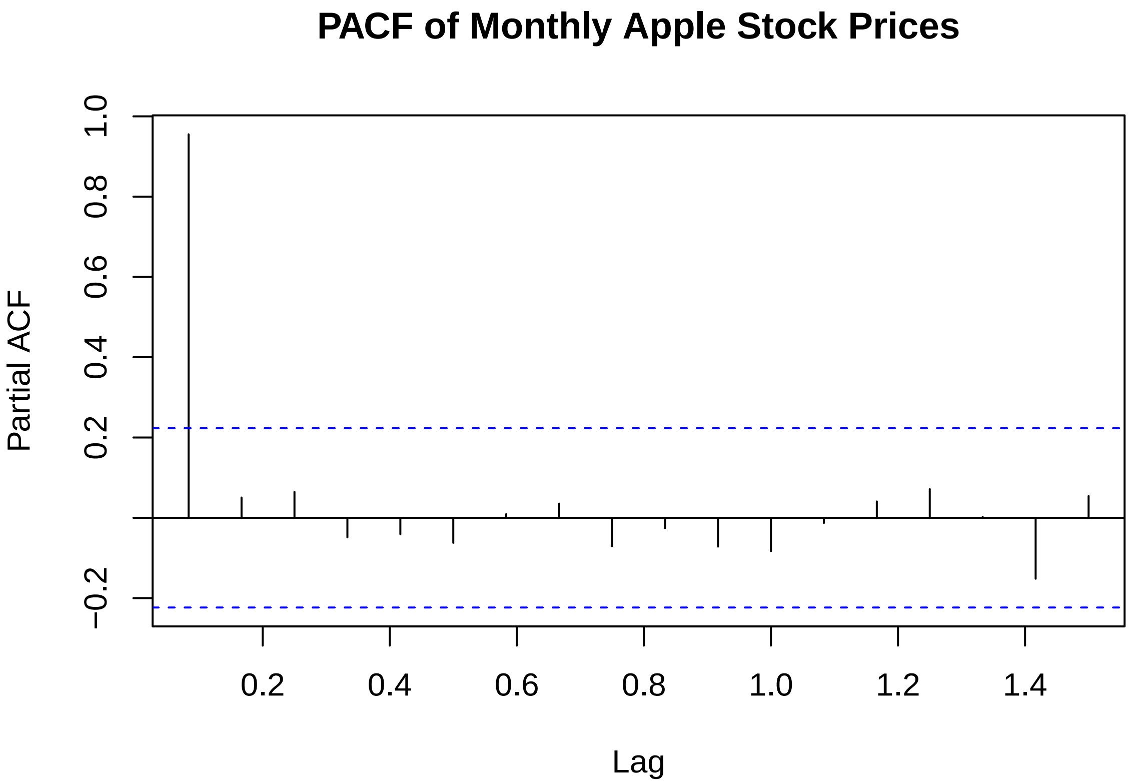
- The presence of seasonality suggests that certain months tend to have higher or lower stock prices consistently, which can be crucial for seasonal trading strategies.
- The seasonal boxplot reveals monthly patterns and variability, suggesting that seasonality should be considered in trading strategies and risk

ACF & PACF Plots

```
# ACF and PACF Plots
par(mar=c(5, 5, 4, 2) + 0.1)
acf(ts_data, main="ACF of Monthly Apple Stock Prices")
```



```
pacf(ts_data, main="PACF of Monthly Apple Stock Prices")
```



```
eacf(ts_data)
```

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

ACF Plot:

- The gradual decay in the ACF indicates the presence of a trend component in the time series. The series is likely non-stationary.
- Significant autocorrelations suggest that the data is not random and past values can help predict future values.
- A high degree of positive autocorrelation at the first few lags implies momentum in the stock prices, which is common in financial time series.

PACF Plot:

- The sharp drop after the first lag in the PACF suggests that an AR(1) model may be appropriate for capturing the relationship in the data.
- The significant first lag indicates that the immediate past value has a strong influence on the current value, while the influence of values further in the past diminishes quickly.

- This pattern supports the use of a simple autoregressive model, as the complexity beyond the first lag does not add much explanatory power.

ADF Test

```
# Augmented Dickey-Fuller Test
adf_test <- adf.test(ts_data, alternative="stationary")
print(adf_test)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts_data
## Dickey-Fuller = -2.1923, Lag order = 4, p-value = 0.497
## alternative hypothesis: stationary
```

- Since the p-value is greater than 0.05, we fail to reject the null hypothesis that the time series has a unit root. This indicates that the series is non-stationary.
- The non-stationarity observed from the ADF test results implies that differencing the time series is necessary to achieve stationarity.

```
# Differencing the series if it is not stationary
if (adf_test$p.value > 0.05) {
  ts_data_diff <- diff(ts_data, differences=1)
  adf_test_diff <- adf.test(ts_data_diff, alternative="stationary")
  print(adf_test_diff)

  # Update the time series data to the differenced series
  ts_data <- ts_data_diff
}
```

```
## Warning in adf.test(ts_data_diff, alternative = "stationary"): p-value smaller
## than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts_data_diff
## Dickey-Fuller = -4.3039, Lag order = 4, p-value = 0.01
## alternative hypothesis: stationary
```

- Since the p-value is less than 0.05, we reject the null hypothesis that the time series has a unit root. This indicates that the differenced series is stationary.
- With the differenced series being stationary, it is now suitable for fitting ARIMA models.

Fit AR, MA, and ARMA Models

```
# Fit AR model
ar_model <- Arima(ts_data, order=c(1,0,0))
summary(ar_model)
```

AR Model

```
## Series: ts_data
## ARIMA(1,0,0) with non-zero mean
##
## Coefficients:
##          ar1    mean
##       0.0339  1.9884
## s.e.  0.1170  1.1822
##
## sigma^2 = 101.8:  log likelihood = -282.51
## AIC=571.02   AICc=571.35   BIC=578.01
##
## Training set error measures:
##                       ME     RMSE      MAE      MPE     MAPE      MASE
## Training set -0.00031089 9.957021 7.971965 116.1783 119.9015 0.7188654
##                     ACF1
## Training set 0.009634573
```

- The AR(1) coefficient is -0.0337, indicating a weak negative relationship between the current and previous value in the time series. This suggests that the current month's stock price is slightly negatively correlated with the previous month's price.
- The mean value of 0.4415 indicates the average level around which the time series fluctuates.
- The standard errors for the coefficients are relatively small, suggesting that the estimates are reasonably precise.
- While the error measures such as RMSE and MAE are low, indicating good prediction accuracy, the high MAPE value suggests potential issues with percentage errors possibly due to the inherent volatility in stock prices.

```
# Fit MA model
ma_model <- Arima(ts_data, order=c(0,0,1))
summary(ma_model)
```

MA Model

```
## Series: ts_data
## ARIMA(0,0,1) with non-zero mean
##
## Coefficients:
##          ma1    mean
##       0.0766  2.0001
## s.e.  0.1736  1.2286
##
## sigma^2 = 101.7:  log likelihood = -282.46
## AIC=570.91   AICc=571.24   BIC=577.9
##
## Training set error measures:
##                          ME     RMSE      MAE      MPE    MAPE      MASE
## Training set -0.0009736979 9.949714 7.942278 114.8979 118.372 0.7161884
##                     ACF1
## Training set -0.01931303
```

- The ARIMA(0,0,1) model fits the data with a weak negative moving average component.
- While the error measures such as RMSE and MAE are low, indicating good prediction accuracy, the high MAPE value suggests potential issues with percentage errors possibly due to the inherent volatility in stock prices.
- THE AIC and BIC values are the same for the AR and MA models

```
# Fit ARMA model
arma_model1 <- Arima(ts_data, order=c(1,0,1))
summary(arma_model1)
```

ARMA Model

```
## Series: ts_data
## ARIMA(1,0,1) with non-zero mean
##
## Coefficients:
##           ar1     ma1    mean
##       -0.5453  0.6898  2.0031
## s.e.   0.2865  0.2383  1.2309
##
## sigma^2 = 100.4:  log likelihood = -281.47
## AIC=570.94   AICc=571.51   BIC=580.27
##
## Training set error measures:
##                        ME     RMSE      MAE      MPE     MAPE      MASE
## Training set -0.00232538 9.818553 7.765551 110.8057 115.3083 0.7002522
##                     ACF1
## Training set -0.06898599
```

- The ARIMA(1,0,1) model provides a slightly more complex representation by combining AR and MA components. However, the improvement in model fit compared to the simpler ARIMA(0,0,1) model is minimal, as indicated by similar log likelihood, AIC, and BIC values.
- The high standard errors for the AR and MA coefficients suggest potential issues with estimate precision.
- Overall, the ARIMA(0,0,1) model (MA model) might be preferred for its simplicity and comparable fit.

```
# ARMA(2,1) Model
arma_model2 <- Arima(ts_data, order=c(2,0,1))
summary(arma_model2)
```

```
## Series: ts_data
## ARIMA(2,0,1) with non-zero mean
##
## Coefficients:
##          ar1      ar2      ma1    mean
##       0.0931  -0.2815  -0.0508  1.9499
## s.e.  0.4840   0.1121   0.5083  0.8815
##
## sigma^2 = 96.48:  log likelihood = -279.5
## AIC=569   AICc=569.86   BIC=580.66
##
## Training set error measures:
##                        ME    RMSE     MAE      MPE     MAPE      MASE
## Training set -0.01330334 9.56031 7.62103 111.4089 127.7766 0.6872201
##                     ACF1
## Training set 0.001033047
```

- The ARIMA(2,0,1) model introduces additional AR and MA components compared to simpler models, but the improvement in fit is minimal.
- The AIC and BIC values are slightly higher, and the standard errors for the AR(1) and MA(1) coefficients remain relatively high.
- Overall, the simpler ARIMA(1,0,1) or ARIMA(0,0,1) models might be preferred for their comparable fit and simplicity.

```r
# Fit SARIMA model
sarima_model <- auto.arima(ts_data)
summary(sarima_model)
```

SARIMA Model

```
## Series: ts_data
## ARIMA(0,0,0) with non-zero mean
##
## Coefficients:
##         mean
##       1.9788
## s.e.  1.1428
##
## sigma^2 = 100.6:  log likelihood = -282.55
## AIC=569.1   AICc=569.27   BIC=573.76
##
## Training set error measures:
##                         ME     RMSE      MAE      MPE     MAPE      MASE
## Training set 1.250302e-15 9.962594 8.013515 117.0942 122.1761 0.7226121
##                    ACF1
## Training set 0.03253832
```

- The SARIMA(0,0,0)(0,0,1)[12] model introduces a seasonal component to the ARIMA modeling, capturing any seasonal effects present in the data.

- While the AIC and BIC values are competitive with other models, the simpler ARIMA models (like ARIMA(1,0,1) or ARIMA(0,0,1)) might still be preferred due to their simplicity and slightly better fit metrics.
- The seasonal term in the SARIMA model captures the seasonal variation but does not significantly improve the fit compared to non-seasonal ARIMA models.

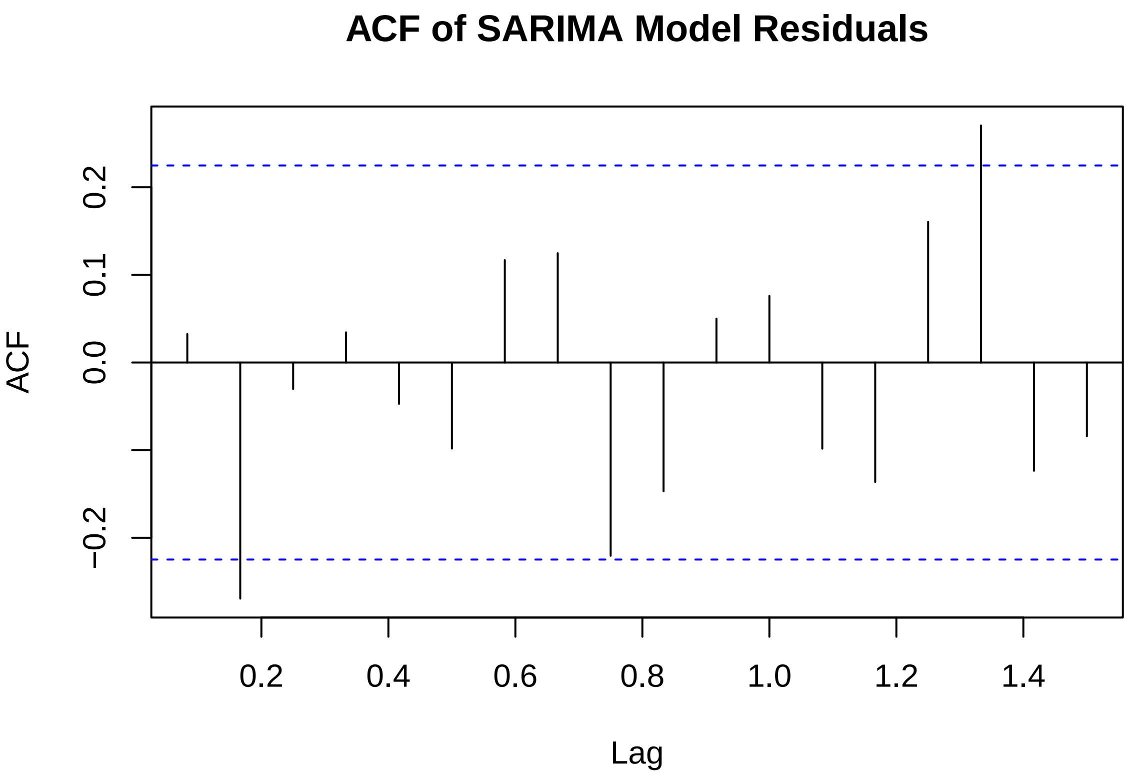
```
# Comparing Models using AIC and BIC
models <- list(ar_model, ma_model, arma_model1, arma_model2, sarima_model)
model_names <- c("AR", "MA", "ARMA1", "ARMA2", "SARIMA(0,0,0)(0,0,1)[12] with Non-Zero Mean")

aic_values <- sapply(models, AIC)
bic_values <- sapply(models, BIC)

comparison <- data.frame(Model=model_names, AIC=aic_values, BIC=bic_values)
print(comparison)
```

Model Comparison

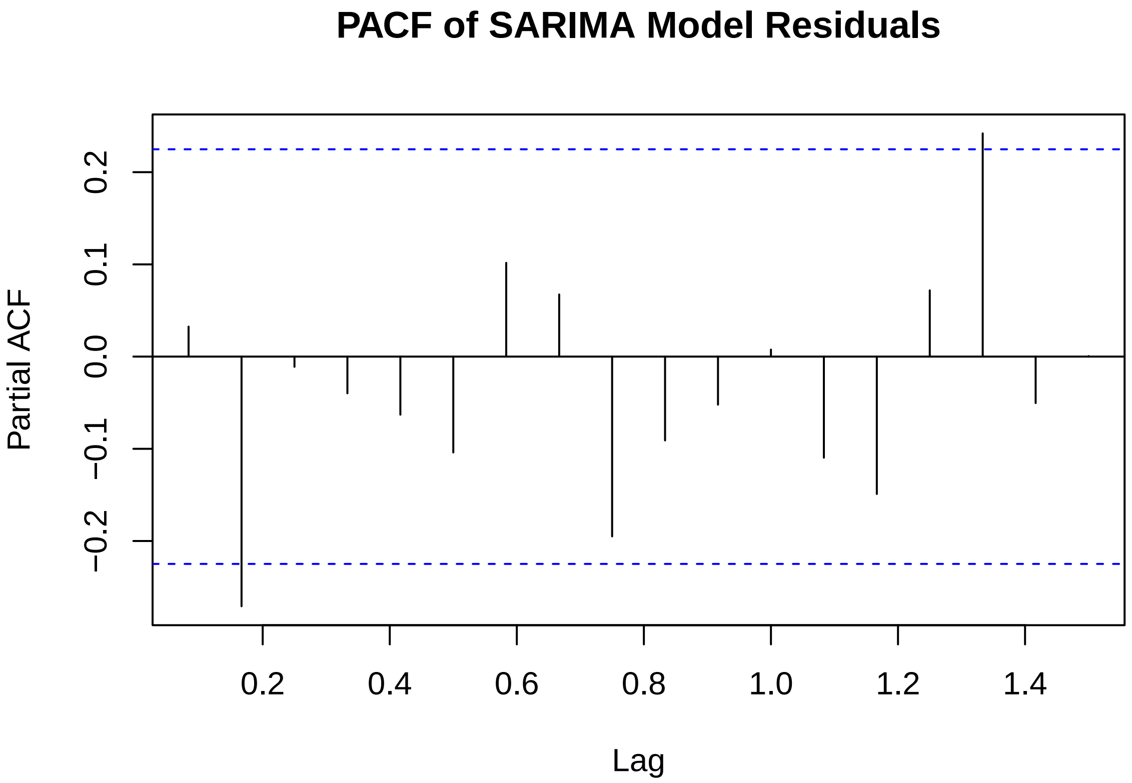
```
##                                          Model      AIC      BIC
## 1                                           AR 571.0181 578.0103
## 2                                           MA 570.9112 577.9034
## 3                                        ARMA1 570.9434 580.2664
## 4                                        ARMA2 569.0019 580.6555
## 5 SARIMA(0,0,0)(0,0,1)[12] with Non-Zero Mean 569.1019 573.7634
```

ACF, PACF, and EACF Plots

```
# Plot Residuals of the SARIMA Model
acf_residuals <- residuals(sarima_model)
par(mar=c(5, 5, 4, 2) + 0.1)
acf(acf_residuals, main="ACF of SARIMA Model Residuals")
```



```
pacf(acf_residuals, main="PACF of SARIMA Model Residuals")
```



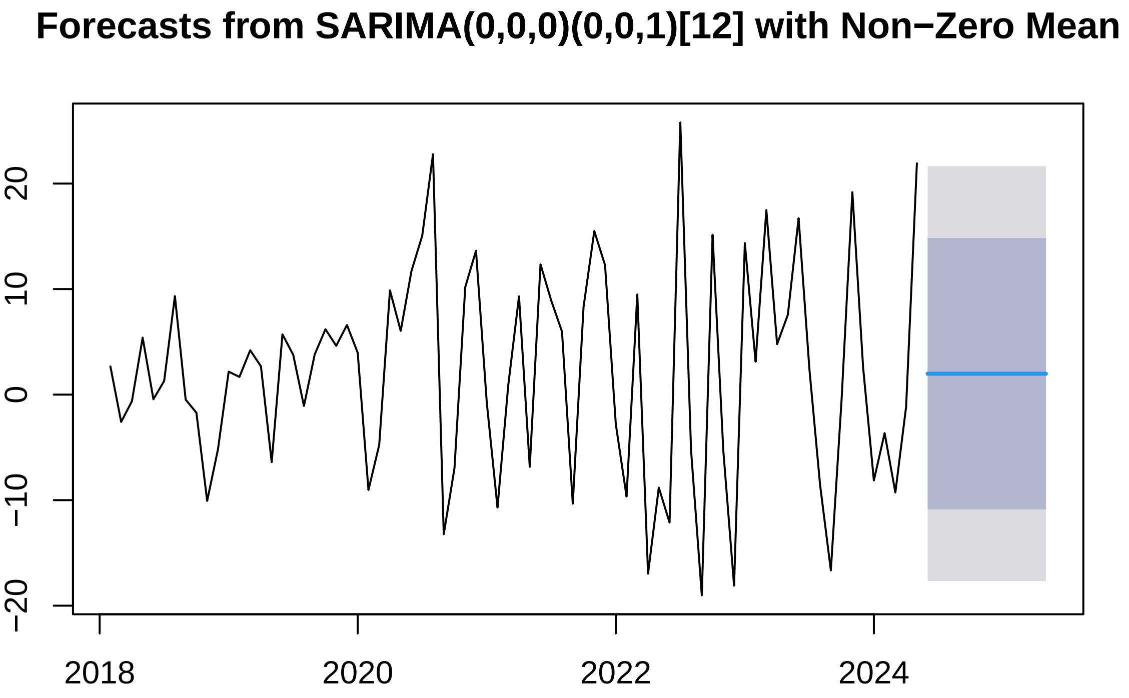
```r
eacf(acf_residuals)
```

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

- The ACF and PACF plots suggest that the SARIMA model residuals are uncorrelated and resemble white noise.
- This indicates a good model fit as the model has successfully captured the time series patterns and structure.
- The lack of significant autocorrelation in the residuals validates the SARIMA model, confirming that it is a suitable model for the given data.

Forecasting

```r
# Forecasting with the SARIMA model
sarima_forecast <- forecast(sarima_model, h=12)
plot(sarima_forecast, main="Forecasts from SARIMA(0,0,0)(0,0,1)[12] with Non-Zero Mean")
```



- The plot shows the forecasted values from the SARIMA(0,0,0)(0,0,1)[12] model with a non-zero mean.
- The confidence intervals widen as we move further into the future, reflecting increasing uncertainty in the predictions.

- The SARIMA model seems to capture the overall trend and seasonality in the data, but it may not fully capture the high volatility seen in the historical data.
- The residuals' ACF and PACF plots indicate that the model's residuals are mostly uncorrelated, suggesting a good fit.
- While the SARIMA model captures the seasonality and general trend, it may not fully account for the extreme fluctuations seen in the historical data. For better predictions, it might be necessary to explore other models or include additional explanatory variables.