BAN 673: Time Series Analytics

PROJECT REPORT

Time Series Analysis: Hobby & Game Stores Retail Sales

TABLE OF CONTENTS

Executive Summary

This project uses the live data of monthly retail sales of hobby and game stores. We have collected this data from Federal Reserve Economic Data and the U.S Census Bureau. It is a monthly data which is presented in terms of its sales in million dollars. After visualizing data, we identified that it has an additive seasonality with a trending pattern. For every year from 1995 till 2018, there was a pattern of increase and decrease from January through September, and then the data is significantly increasing from October through December. However, there seems to be a COVID impact on retail sales from 2018 till 2021. Yet, the previous trend seems to be repeating. In this project, we use the historical data to predict monthly retail sales across all stores for next 2 fiscal years Regression-based models, advanced exponential smoothing models, and autoregressive integrated moving average models (ARIMA) were utilized for this project. Additional variations of the regression and advanced exponential smoothing models were also constructed to ensure better forecasts. Model evaluation was based on the MAPE and RMSE accuracy measures. We identify the Auto-ARIMA as the best model and can be used to forecast future Hobby and Game Stores retail sales.

Introduction

The primary goal of this project is to efficiently forecast retail sales for future years with the help of different types of Time Series forecasting models. The dataset used in this project is monthly retail sales trade data of various hobby and game stores across the United States of America obtained from the Economic Research datastore of the Federal Reserve Bank of St. Louis. The data is collected and can be found on the United States Census website. The United States Code, Title 13, authorizes this survey and provides for voluntary responses. This documentation ensures that we achieve consistent results and provide the most accurate forecast data about U.S. game stores' retail economic activity.

8 Steps of Forecasting Process

The 8 steps necessary for Forecasting Analysis of data are as mentioned below:

Step-1: Define Goal

The goal of this project is a predictive analysis using time series forecasting. We will predict the future retail sales for which the actual time series data is not available. The forecasting will be executed for the future 2 years i.e, Jan 2022 - Dec 2023. This means we will be forecasting the future retail sales values for each month for 2 years. To identify the best forecasting models, we will use the various accuracy measures of each model and decide the best possible forecasting model. Data visualization will be used to scope time series components, accuracy measures, correlogram, etc to get an in-depth look into the time-series dataset. For this project, we will be utilizing R software for simple and advanced forecasting methods.

Step-2: Get Data

We collected the 26 years of historical retail sales data from 1st January 1995 till 1st December 2021. The collected dataset has a monthly temporal frequency with each data point as 1st day of January. Further investigating the retail sales historical data, we can see that the dataset is highly seasonal with some hint of trending pattern. For each year, it can be observed that from January till September (Quarter 1 - Quarter 3), the retail sales are relatively low with some minor ups and downs. However, from October till December (4th Quarter), the retail sales have a significant increase in sales. This is because famous festivals and holidays like Halloween, Thanksgiving, Christmas, and New Year’s Eve and game stores are mostly targeted shops in this quarter of the year. Also, the sudden change in cyclical behavior of data from 2018 onwards is due to the COVID impact.

Step-3: Explore and Visualize Series

Exploring the time series components using the stl*()* function, the following are the season, trend, and level components of the game store's retail sales. From this plot, it can be inferred that this dataset has additive seasonality with some presence of a quadratic trend. As shown below, there is an upward and downward change in the series from one period to the next. Therefore, it can be concluded that this dataset is seasonal along with a trending pattern.

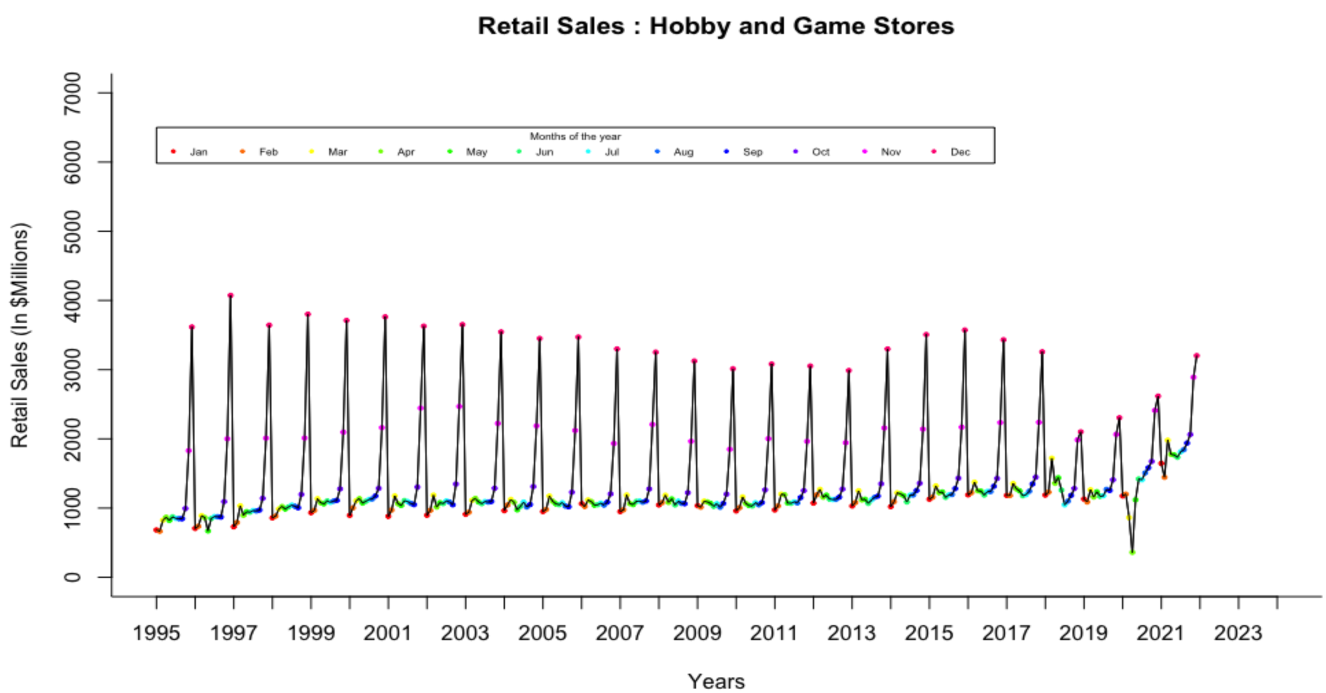


Figure 1: Retail Sales: Game Stores Monthly Data Visualization

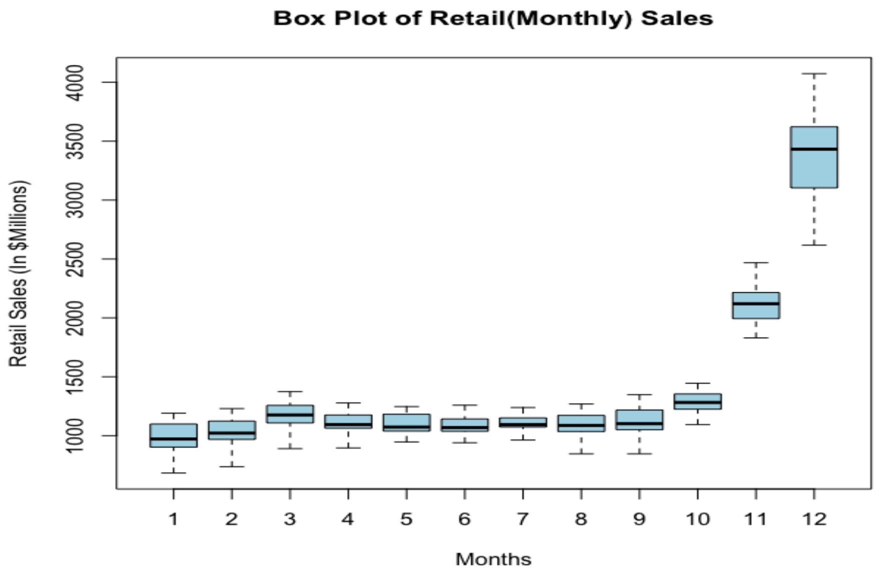


Figure 2: Box plot of monthly Retail Sales

The boxplot here, gives a pictorial representation of the monthly data. It helps us understand that the retail sales are low in the 10 months of the year, but in the November and December, it grows exponentially.

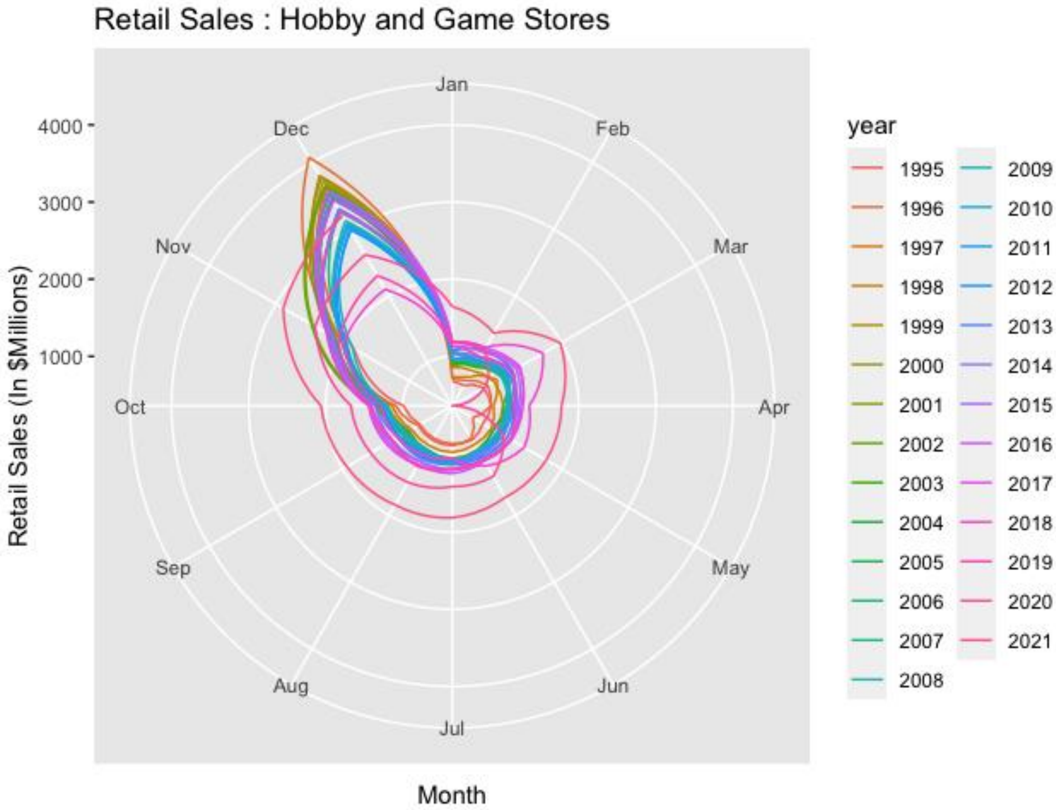


Figure 3: Polar seasonal plot of Retail Sales Data

Just like the box plot, the above polar seasonal plot shows how the retail sales show substantial increase in December month. This plot also gives us a pictorial representation of the whole data.

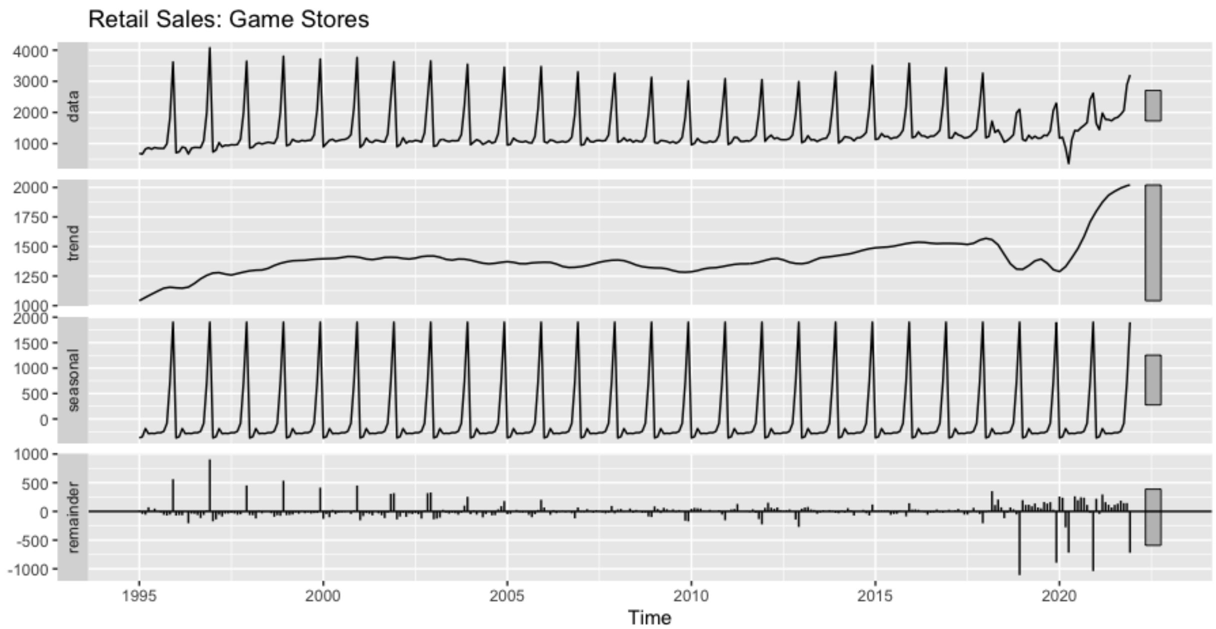


Figure 4: Seasonal, Trend, and Level components

Using the Autocorrelation chart, we visualize the autocorrelation coefficient between time series data and lagged versions of the same time series. To achieve this functionality, we are using the *Acf ()* function in R for 12 lags and plot a correlogram for various lags.

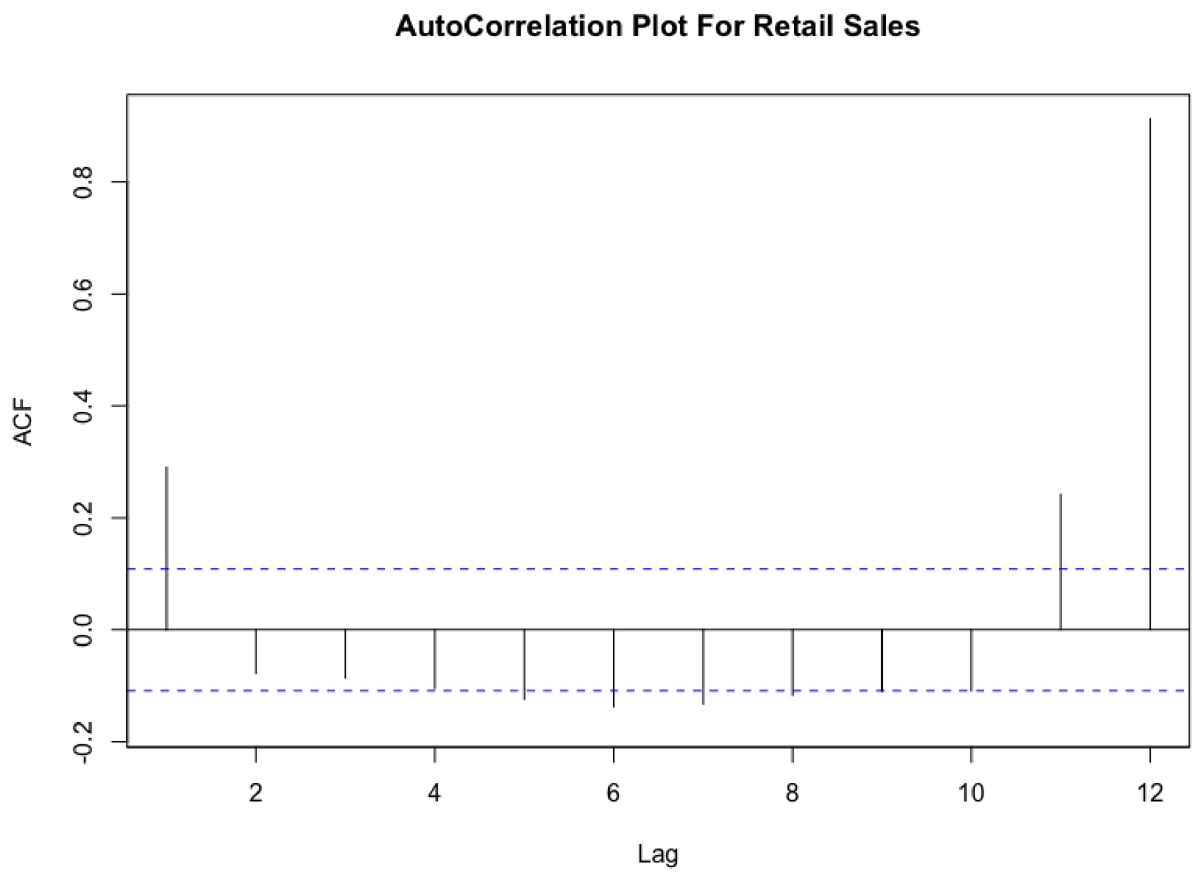
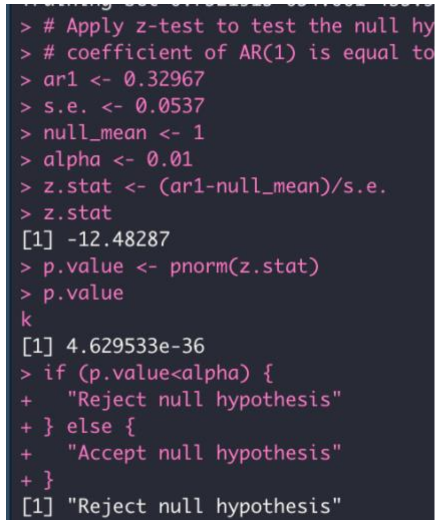


Figure 5: Autocorrelation Chart for the entire dataset

From the autocorrelation chart, it can be observed that lag 12 has a significant autocorrelation coefficient, this is the evidence of seasonal components in time series. Also, lag 1 has a significant autocorrelation coefficient (above horizontal threshold), hence this dataset also contains a trending pattern.

To evaluate the dataset predictability, we perform hypothesis testing. Using the Autoregressive model AR (1), we test the null hypothesis that the slope coefficient *beta* is equal to 1. So if the null hypothesis is rejected i.e., p-value < 0.05, then the series is not a random walk, else it is a random walk.



```
> # Apply z-test to test the null hy
> # coefficient of AR(1) is equal to
> ar1 <- 0.32967
> s.e. <- 0.0537
> null_mean <- 1
> alpha <- 0.01
> z.stat <- (ar1-null_mean)/s.e.
> z.stat
[1] -12.48287
> p.value <- pnorm(z.stat)
> p.value
k
[1] 4.629533e-36
> if (p.value<alpha) {
+   "Reject null hypothesis"
+ } else {
+   "Accept null hypothesis"
+ }
[1] "Reject null hypothesis"
```

Figure 6: Hypothesis Testing using AR(1) Model in R

As shown in Figure 4, we reject the null hypothesis, therefore, the game store retail sales dataset is not a random walk and can be used for predictions.

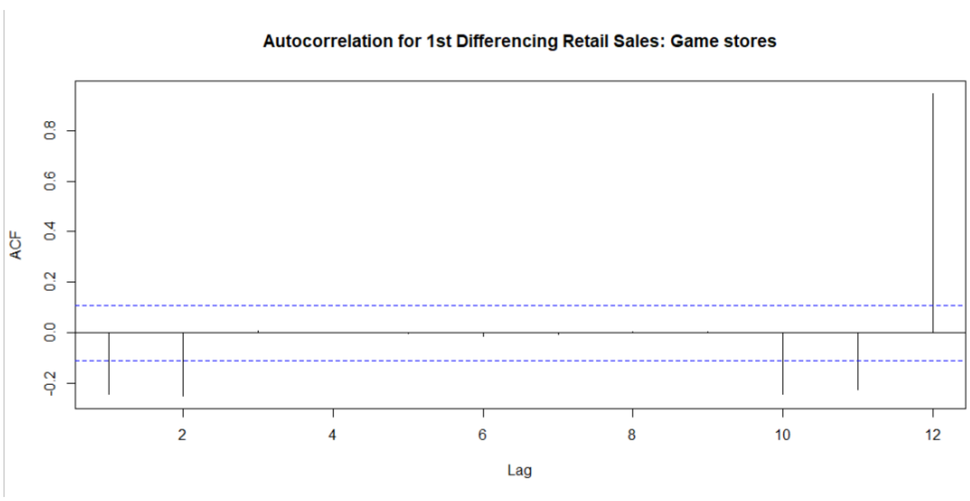


Figure 7: Autocorrelation Chart for 1st differencing

The second approach, considering the 1st differencing autocorrelation chart of game stores retail sales, for lag 1, lag 2, lag 10, lag 11, and lag 12, autocorrelation coefficients have significance. This indicates that correlation coefficients at lag 1, lag2, lag 10, lag 11, and lag 12 are not within horizontal thresholds, hence it can be inferred that retail sales data is not a random walk, can be predicted. Since lag 12 has a much more significant value, it's another evidence of the seasonality in historical data.

Step-4: Data Preprocessing

The dataset is monthly retail sales collected on the 1st day of January for every year. Hence, data is already considered aggregate. Also, there seem to be no outliers in the dataset. The only extreme value observed is the sales value of December for every year. But this is an expected result.

Step-5: Partition time series

We are partitioning the entire dataset into training and validation datasets to develop different time series, forecasting models. Training partition will be used to develop models and the validation partition will be used to evaluate the performance of each model. Here, we are partitioning the dataset into 74% training partition (data from January 1995 till December 2014) and 26% validation partition (data from January 2015 till December 2021). The reason for using 7 years of validation partition is to overcome the COVID impact on retail sales.

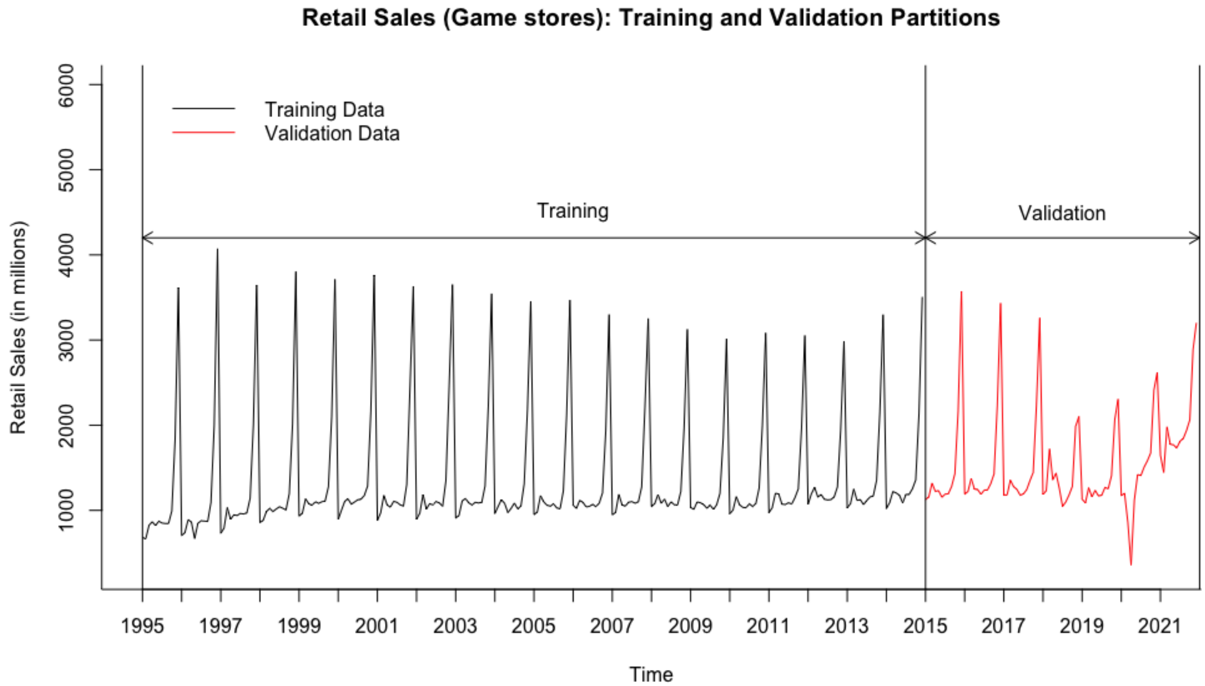


Figure 8: Training and Validation dataset partitions

Step-6: Forecasting Methods

Now that we know that the retail sales historical dataset is predictable, we develop various forecasting models. We will utilize the training partition to develop models and test its performance on validation data. Furthermore, we calculate the accuracy performance measure and decide which forecasting model to utilize for predictions.

MODEL 1: Regression Models

To develop regression models for the time series dataset, we utilized the training and validation partitions of the dataset. We developed five regression models with combinations of a linear trend, quadratic trend, and seasonality. All five models are developed on a training partition and tested on validation partitions. It is being observed that two models 1) Linear Trend with Seasonality &

2) Quadratic Trend with Seasonality seem to be statistically significant in terms of p-value and relatively great fit according to the R-Square coefficient of determination. Below are the model summaries of the two models in training partition,

Regression Model with Linear Trend and Seasonality

```
Call:
tslm(formula = train.ts ~ trend + season)

Residuals:
    Min      1Q  Median      3Q     Max 
-512.16  -35.01    6.33   50.91  682.02 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  862.2562    32.1804  26.794  < 2e-16 ***
trend          0.5634     0.1210   4.655 5.52e-06 ***
season2       43.8366    41.0322   1.068 0.286500    
season3      193.6232    41.0328   4.719 4.15e-06 ***
season4      137.1597    41.0337   3.343 0.000971 ***
season5      104.9463    41.0349   2.557 0.011195 *  
season6      109.3329    41.0365   2.664 0.008269 ** 
season7      135.6195    41.0385   3.305 0.001105 ** 
season8      119.7560    41.0408   2.918 0.003877 ** 
season9      137.1426    41.0435   3.341 0.000975 ***
season10     310.9792    41.0465   7.576 8.95e-13 ***
season11    1153.0658    41.0499  28.089  < 2e-16 ***
season12    2515.2023    41.0537  61.266  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 129.8 on 227 degrees of freedom
Multiple R-squared:  0.9685,	Adjusted R-squared:  0.9668 
F-statistic: 581.5 on 12 and 227 DF,  p-value: < 2.2e-16
```

Figure 9: Training model summary: regression model with linear trend & seasonality

Model Equation:

$$\boldsymbol{Y_t = 862.2562 + 0.5634\, t + 43.8366\, D_2 + 193.6232\, D_3 + 137.1597\, D_4 + 104.9463\, D_5 + 109.3329\, D_6 + 135.6195\, D_7 + 119.7560\, D_8 + 137.1426\, D_9 + 310.9792\, D_{10} + 1153.0658\, D_{11} + 2515.2023\, D_{12}}$$

As shown in the summary, the R-square coefficient of determination is 0.9685 (Approx 0.97), ie, 97% variation of retail sales is explained by the variation of time index. Therefore, the model seems to be a very good fit for the forecast. Considering the p-value, it's significantly less than 0.01 (1%), hence with a confidence of 99%, we can say that it’s a very good fit. Also, Intercept, trend, and season (except season 2) coefficients are statistically significant (p-value < 0.01). Therefore, overall, we can say that the regression model with linear trend and seasonality is a statistically very good fit and can be applied for forecasting retail sales.

Regression Model with Quadratic Trend and Seasonality

```
Call:
tslm(formula = train.ts ~ trend + I(trend^2) + season)

Residuals:
    Min      1Q  Median      3Q     Max
-480.51  -44.17   -7.43   56.27  715.08

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.909e+02  3.623e+01  21.832  < 2e-16 ***
trend        2.337e+00  4.711e-01   4.960 1.39e-06 ***
I(trend^2)  -7.358e-03  1.893e-03  -3.887 0.000134 ***
season2      4.376e+01  3.981e+01   1.099 0.272854
season3      1.935e+02  3.981e+01   4.860 2.20e-06 ***
season4      1.370e+02  3.982e+01   3.440 0.000692 ***
season5      1.047e+02  3.982e+01   2.631 0.009110 **
season6      1.091e+02  3.982e+01   2.740 0.006629 **
season7      1.354e+02  3.982e+01   3.400 0.000796 ***
season8      1.196e+02  3.982e+01   3.002 0.002983 **
season9      1.370e+02  3.982e+01   3.439 0.000695 ***
season10     3.108e+02  3.983e+01   7.805 2.20e-13 ***
season11     1.153e+03  3.983e+01  28.947  < 2e-16 ***
season12     2.515e+03  3.983e+01  63.141  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 125.9 on 226 degrees of freedom
Multiple R-squared:  0.9705,    Adjusted R-squared:  0.9688
F-statistic: 571.3 on 13 and 226 DF,  p-value: < 2.2e-16
```

Figure 10: Training model summary: regression model with quadratic trend & seasonality

Model Equation:

yt = 790.9 + 2.337 t – 0.007358 t^2 + 43.76 D2 + 193.5 D3 + 137 D4 + 104.7 D5 + 109.1 D6 + 135.4 D7 + 119.6 D8 + 137 D9 + 310 D10 + 1153 D11 + 2515 D12

As shown in the summary, the R-square coefficient of determination is 0.9705 (Approx 0.97), ie, 97% variation of retail sales is explained by the variation of time index. Therefore, the model seems to be the best fit for the forecast. Considering the p-value, it’s significantly less than 0.01 (1%), hence with a confidence of 99%, we can say that it’s the best fit. Also, Intercept, trend, trend^2, and season (except season 2) coefficients are statistically significant (p-value < 0.01).

Therefore, overall, we can say that the regression model with quadratic trend and seasonality is a statistically best fit and can be applied for forecasting revenues.

Below are the RMSE and MAPE accuracy measures of all the five models in training and validation partitions.

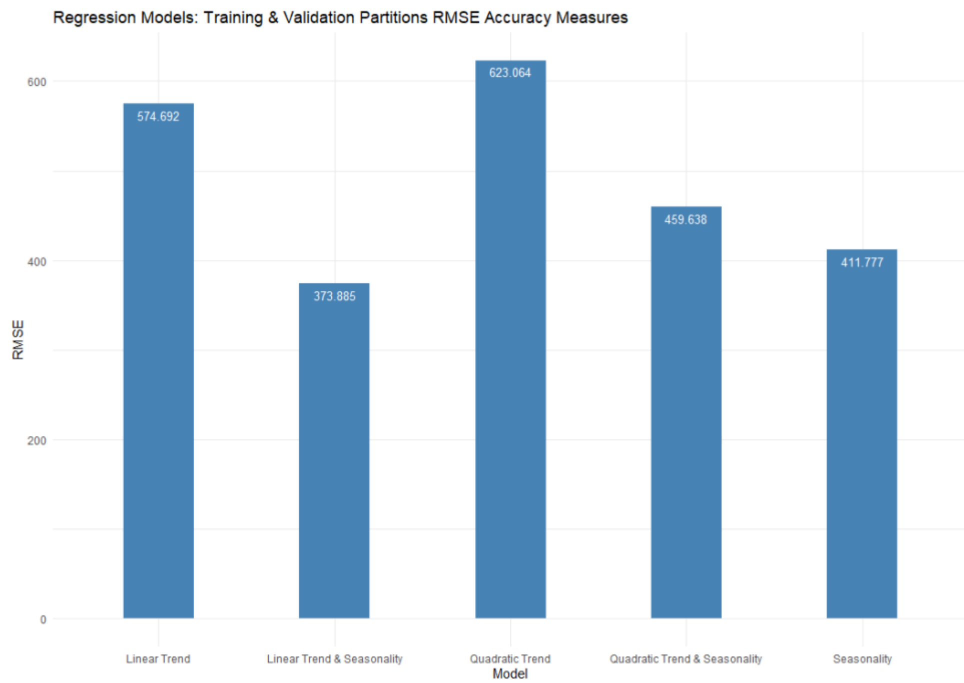


Figure 11: Training RMSE accuracy measures

Figure 12: Training MAPE accuracy measures

As per RMSE and MAPE accuracy measures for training partitions, the Regression model with linear trend and Seasonality seems to be performing best since it contains the lowest RMSE and MAPE values. However, considering the statistical significance of the quadratic trend & seasonality model over the linear trend & seasonality model, we are considering both these models for entire data forecasting.

For the entire dataset, below is the regression with linear trend and seasonality model summary,

```
Call:
tslm(formula = retailsales.ts ~ trend + season)

Residuals:
     Min       1Q   Median       3Q      Max
-1331.20   -69.13   -15.52    40.97   902.56

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  850.9615    44.9212  18.943  < 2e-16 ***
trend          0.9915     0.1248   7.942 3.66e-14 ***
season2       27.6011    57.1625   0.483  0.62954
season3      188.1281    57.1629   3.291  0.00111 **
season4       91.2106    57.1636   1.596  0.11159
season5       99.3302    57.1646   1.738  0.08327 .
season6       95.2646    57.1658   1.666  0.09663 .
season7      112.8657    57.1673   1.974  0.04923 *
season8      112.8742    57.1690   1.974  0.04922 *
season9      143.7716    57.1711   2.515  0.01242 *
season10     302.6689    57.1734   5.294 2.27e-07 ***
season11    1120.7515    57.1760  19.602  < 2e-16 ***
season12    2295.6859    57.1789  40.149  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 210 on 311 degrees of freedom
Multiple R-squared:  0.9096,    Adjusted R-squared:  0.9061
F-statistic: 260.7 on 12 and 311 DF,  p-value: < 2.2e-16
```

Figure 13: Model summary: regression model with linear trend & seasonality

Model Equation:

$$\boldsymbol{Y_t = 850.9615 + 0.9915\, t + 27.6011\, D_2 + 188.1281\, D_3 + 91.2106\, D_4 + 99.3302\, D_5 + 95.2646\, D_6 + 112.8657\, D_7 + 112.8742\, D_8 + 143.7716\, D_9 + 302.6689\, D_{10} + 1120.7515\, D_{11} + 2295.6859\, D_{12}}$$

As shown in the summary, the R-square coefficient of determination is 0.9096 (Approx 0.91), ie, 91% variation of retail sales is explained by the variation of time index. Therefore, the model seems to be a good fit for the forecast in the entire dataset. Considering the p-value, it’s significantly less than 0.01 (1%), hence with a confidence of 99%, we can say that it’s the best fit. Therefore, overall, we can say that the regression model with linear trend and seasonality is a statistically good fit for the entire dataset. Below is the forecast for the future 24 months with a 95% confidence interval using this model,

	Point Forecast	Lo 95	Hi 95
Jan 2022	1173.204	750.3472	1596.062
Feb 2022	1201.797	778.9398	1624.654
Mar 2022	1363.316	940.4584	1786.173
Apr 2022	1267.390	844.5324	1690.247
May 2022	1276.501	853.6435	1699.358
Jun 2022	1273.427	850.5695	1696.284
Jul 2022	1292.019	869.1621	1714.876
Aug 2022	1293.019	870.1621	1715.876
Sep 2022	1324.908	902.0509	1747.765
Oct 2022	1484.797	1061.9398	1907.654
Nov 2022	2303.871	1881.0139	2726.728
Dec 2022	3479.797	3056.9398	3902.654
Jan 2023	1185.103	761.9476	1608.258
Feb 2023	1213.695	790.5402	1636.850
Mar 2023	1375.214	952.0587	1798.369
Apr 2023	1279.288	856.1328	1702.443
May 2023	1288.399	865.2439	1711.554
Jun 2023	1285.325	862.1698	1708.480
Jul 2023	1303.917	880.7624	1727.072
Aug 2023	1304.917	881.7624	1728.072
Sep 2023	1336.806	913.6513	1759.961
Oct 2023	1496.695	1073.5402	1919.850
Nov 2023	2315.769	1892.6143	2738.924
Dec 2023	3491.695	3068.5402	3914.850

Figure 14: Forecast 24 months, regression with linear trend and seasonality

Similarly, for the entire dataset, below is the regression with quadratic trend and seasonality model summary,

```
Call:
tslm(formula = retailsales.ts ~ trend + I(trend^2) + season)

Residuals:
     Min       1Q   Median       3Q      Max
-1351.04   -61.05    -6.76    49.05   872.97

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 9.011e+02  5.184e+01  17.383  < 2e-16 ***
trend       6.844e-02  4.984e-01   0.137  0.89085
I(trend^2)  2.840e-03  1.485e-03   1.913  0.05671 .
season2     2.763e+01  5.692e+01   0.485  0.62773
season3     1.882e+02  5.692e+01   3.306  0.00106 **
season4     9.128e+01  5.692e+01   1.604  0.10982
season5     9.941e+01  5.692e+01   1.746  0.08173 .
season6     9.535e+01  5.692e+01   1.675  0.09493 .
season7     1.130e+02  5.692e+01   1.984  0.04811 *
season8     1.130e+02  5.693e+01   1.984  0.04811 *
season9     1.438e+02  5.693e+01   2.527  0.01201 *
season10    3.027e+02  5.693e+01   5.317 2.02e-07 ***
season11    1.121e+03  5.693e+01  19.686  < 2e-16 ***
season12    2.296e+03  5.694e+01  40.320  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 209.1 on 310 degrees of freedom
Multiple R-squared:  0.9106,    Adjusted R-squared:  0.9069
F-statistic:   243 on 13 and 310 DF,  p-value: < 2.2e-16
```

Figure 15: Model summary: regression model with quadratic trend & seasonality

Model Equation:

$$Y_t = 901.1 + 0.06844\, t + 0.002840\, t^2 + 27.63\, D_2 + 188.2\, D_3 + 91.28\, D_4 + 99.41\, D_5 + 95.35\, D_6 + 113\, D_7 + 113\, D_8 + 143.8\, D_9 + 302.7\, D_{10} + 1121\, D_{11} + 2296\, D_{12}$$

As shown in the summary, the R-square coefficient of determination is 0.9106 (Approx 0.91), ie, 91% variation of retail sales is explained by the variation of time index. Therefore, the model seems to be a good fit for the forecast in the entire dataset. Considering the p-value, it’s significantly less than 0.01 (1%), hence with a confidence of 99%, we can say that it’s the best fit. Therefore, overall, we can say that the regression model with quadratic trend and seasonality is a statistically good fit for the entire dataset. Below is the forecast for the future 24 months with a 95% confidence interval using this model,

	Point Forecast	Lo 95	Hi 95
Jan 2022	1223.306	799.0964	1647.516
Feb 2022	1252.853	828.5229	1677.183
Mar 2022	1415.326	990.8731	1839.778
Apr 2022	1320.354	895.7766	1744.932
May 2022	1330.420	905.7149	1755.124
Jun 2022	1328.300	903.4658	1753.134
Jul 2022	1347.847	922.8812	1772.812
Aug 2022	1349.801	924.7017	1774.900
Sep 2022	1382.644	957.4089	1807.880
Oct 2022	1543.487	1118.1139	1968.861
Nov 2022	2363.516	1938.0018	2789.030
Dec 2022	3540.396	3114.7393	3966.053
Jan 2023	1246.690	820.5900	1672.790
Feb 2023	1276.305	850.0474	1702.563
Mar 2023	1438.846	1012.4282	1865.264
Apr 2023	1343.943	917.3620	1770.523
May 2023	1354.076	927.3303	1780.822
Jun 2023	1352.025	925.1109	1778.938
Jul 2023	1371.640	944.5556	1798.724
Aug 2023	1373.662	946.4053	1800.919
Sep 2023	1406.574	979.1413	1834.006
Oct 2023	1567.485	1139.8748	1995.095
Nov 2023	2387.582	1959.7909	2815.372
Dec 2023	3564.530	3136.5564	3992.503

Figure 16: Forecast 24 months, regression with quadratic trend and seasonality

Below is the RMSE and MAPE accuracy measures of the two models for the entire dataset,

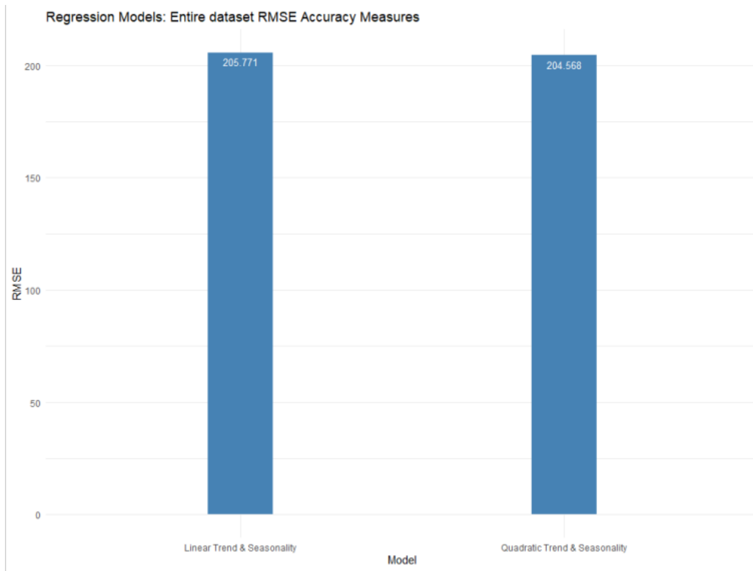


Figure 17: RMSE Accuracy measures of two models: Entire dataset

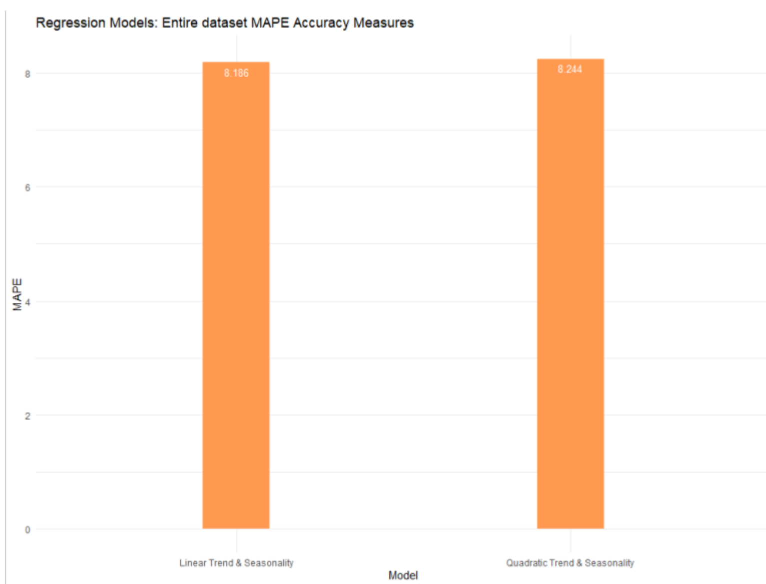


Figure 18: MAPE Accuracy measures of two models: Entire dataset

Comparing both the graphs, it can be observed that both RMSE and MAPE values seem to be very close to each other for both models. Therefore, we will be using a Regression **Model with Quadratic Trend and Seasonality** as **Model 1** for the future 24 month's forecast of retail sales to handle future non-linear changes in data patterns. Below is the forecast plot for the future 24 months,

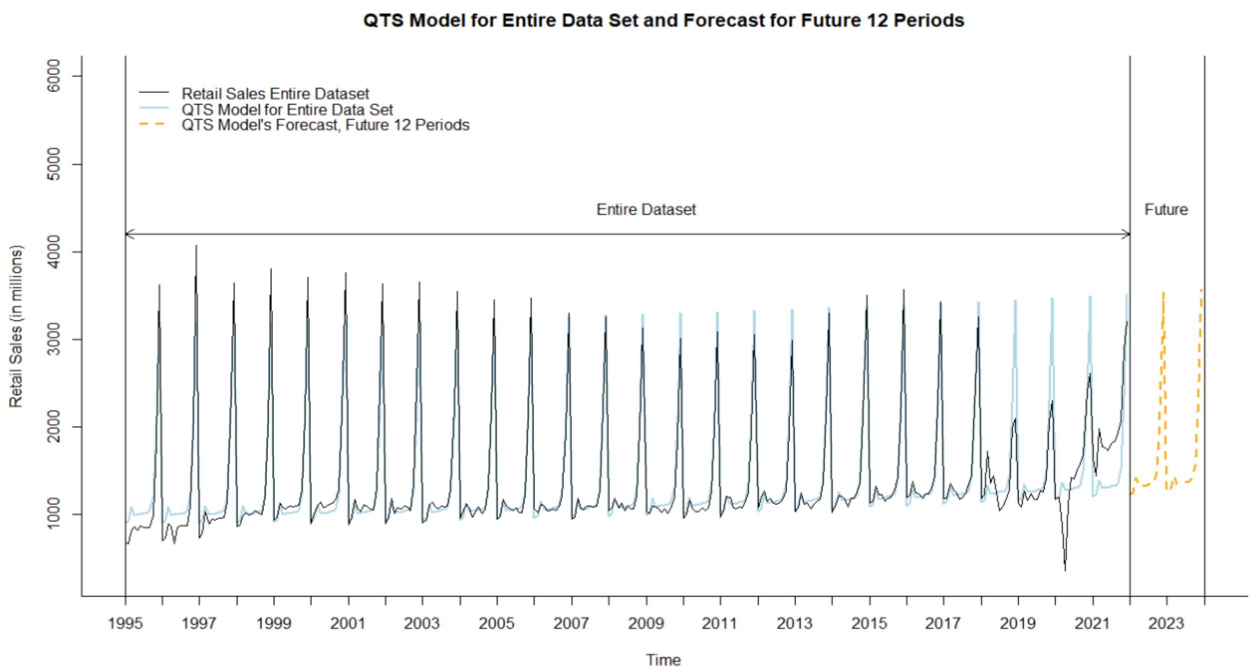


Figure 19: Future 24 month's forecast: Regression with quadratic trend and seasonality

MODEL 2: Holt-Winters Model (ZZZ automated selection of error, trend, and seasonality)

In this model, we will be developing an advanced exponential smoothing method i.e, Holt-Winters model using *ets()* function. For training partition, below is the HW model summary with the automated selection of the model options and automated selection of the smoothing parameters,

```
ETS(M,A,A)

Call:
 ets(y = train.ts, model = "ZZZ")

  Smoothing parameters:
    alpha = 0.0787
    beta  = 1e-04
    gamma = 0.3914

  Initial states:
    l = 1107.4677
    b = 2.1822
    s = 2123.091 737.1111 -101.7431 -282.9801 -296.7709 -282.2248
           -307.3437 -304.2141 -270.9329 -209.8216 -359.3809 -444.7901

  sigma:  0.0522

     AIC     AICc      BIC
3330.808 3333.565 3389.979
```

Figure 20: Holt-Winters ‘ZZZ’ model summary: Training Partition

This HW model has the (M, A, A) options, i.e., multiplicative error, additive trend, and additive seasonality. The optimal value for exponential smoothing constant (alpha) is 0.0787, smoothing constant for trend estimate (beta = 0.0001), and smoothing constant for seasonality estimate (gamma) is 0.3914. The alpha value of this model indicates that the model’s level component tends to be more global, the trend component tends to be more global, while additive seasonality is locally adjusted as gamma is not close to zero. The latter is also indicating that, according to this model, the seasonality changes over time.

Below is the RMSE and MAPE accuracy measures comparing the Holt-Winters model and Regression model with quadratic trend and seasonality in training partition,

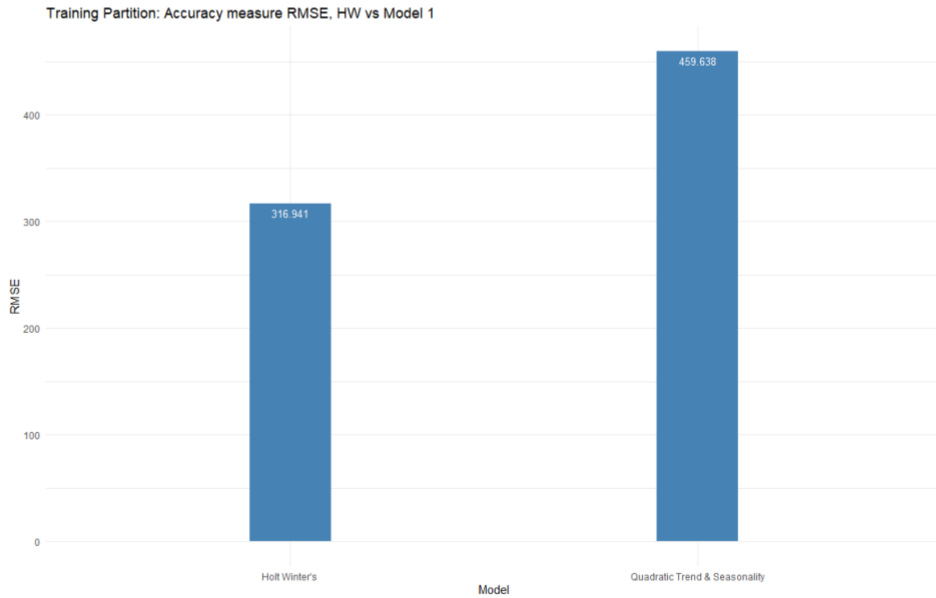


Figure 21: RMSE Accuracy measures of HW vs Model 1: Training Partition

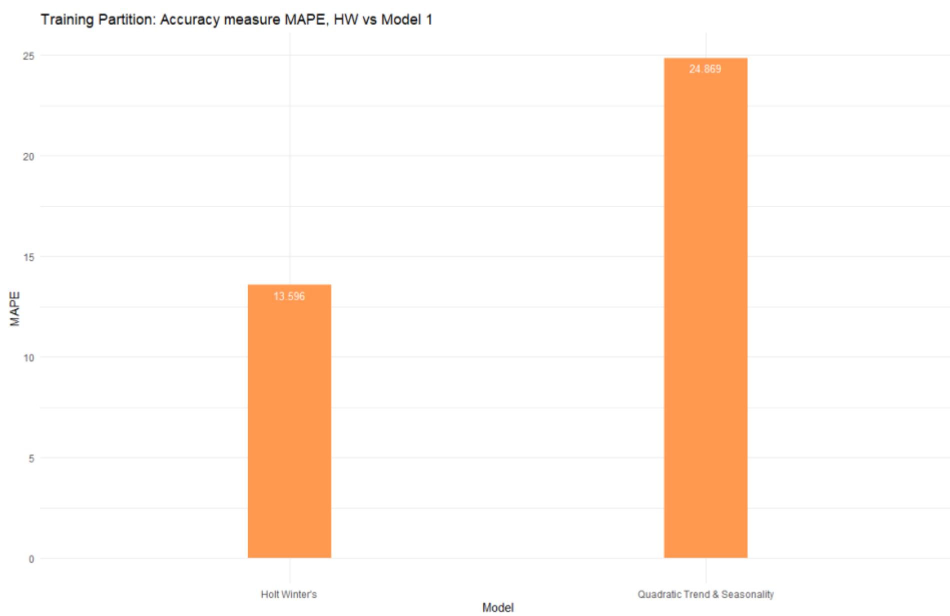


Figure 22: MAPE Accuracy measures of HW vs Model 1: Training Partition

Comparing RMSE and MAPE accuracy measures of both models for training partition, it can be observed that the Holt-Winters model is performing better than the Regression model with quadratic trend and seasonality because the HW model has lower RMSE and MAPE values.

The below plot depicts the validation partition forecast using (M, A, A) HW model,

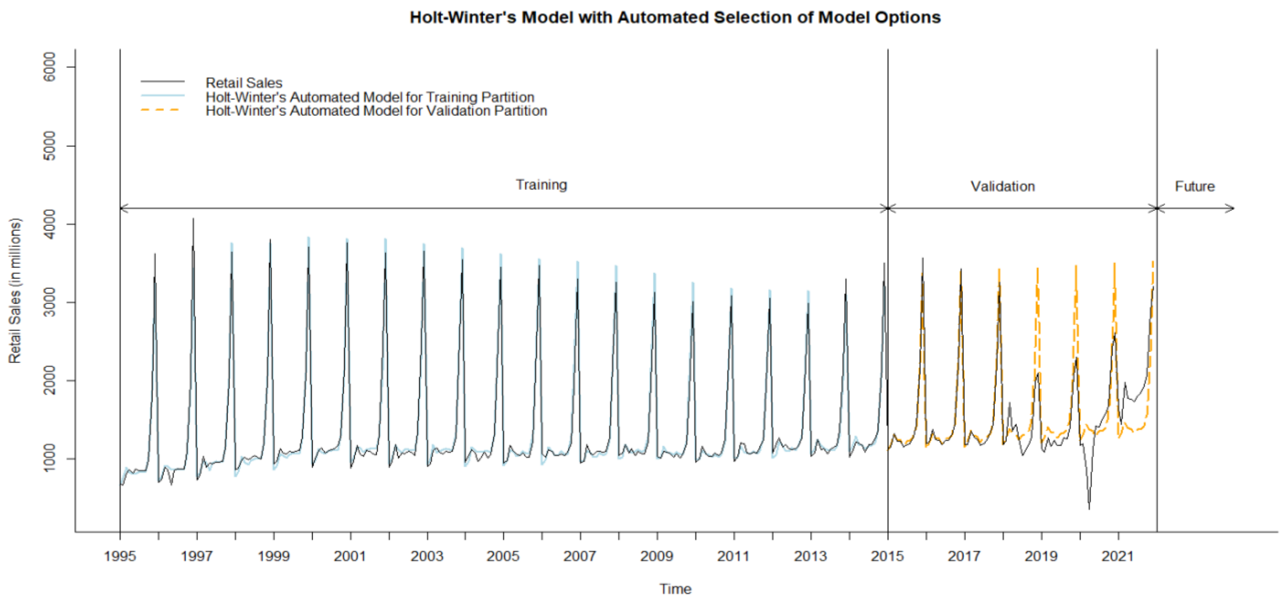


Figure 23: Validation Partition forecast for HW (M, A, A) model

Using the entire dataset below is the HW model summary with the automated selection of the model options and automated selection of the smoothing parameters,

```
ETS(M,N,M)

Call:
 ets(y = retailsales.ts, model = "ZZZ")

  Smoothing parameters:
    alpha = 0.2373
    gamma = 0.2658

  Initial states:
    l = 1201.9627
    s = 3.0296 1.5285 0.8963 0.7546 0.7289 0.7474
           0.7742 0.7151 0.7804 0.7557 0.6601 0.6291

  sigma:  0.0954

     AIC     AICc      BIC
5001.194 5002.752 5057.905
```

Figure 24: Holt-Winters ‘ZZZ’ model summary: Entire Dataset

This HW model has the (M, N, M) options, i.e., multiplicative error, no trend, and multiplicative seasonality. The optimal value for exponential smoothing constant (alpha) is 0.2373, no smoothing constant for trend estimate (beta), and smoothing constant for seasonality estimate (gamma) is 0.26. The alpha value of this model indicates that the model’s level component tends to be more local, while multiplicative seasonality is locally adjusted as gamma is not close to zero. The latter is also indicating that, according to this model, the seasonality changes over time.

Below is the RMSE and MAPE accuracy measures comparing the Holt-Winters model and Regression model with quadratic trend and seasonality for the entire dataset,

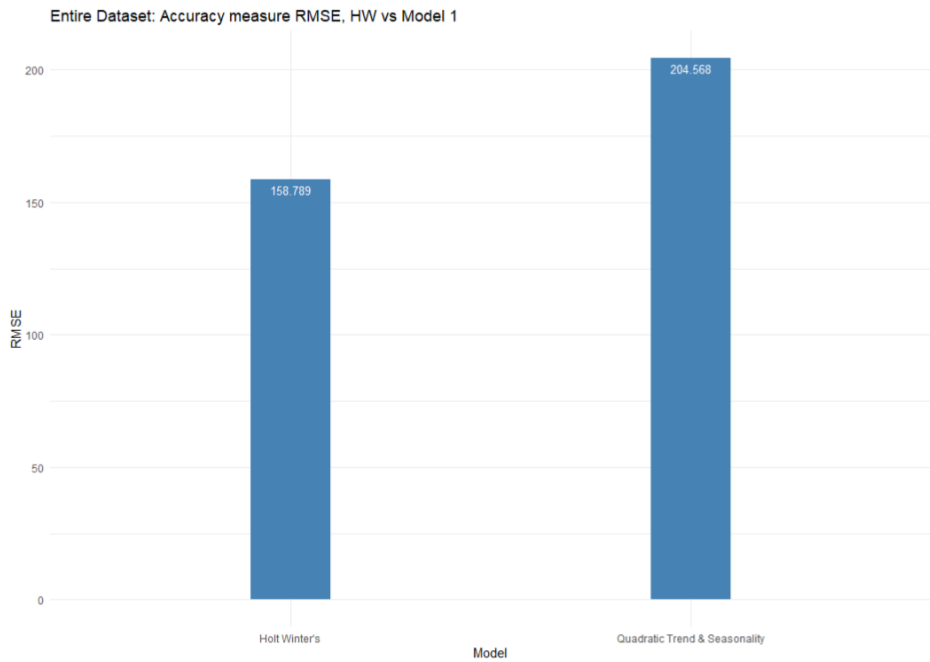


Figure 25: RMSE Accuracy measures of HW vs Model 1: Entire Dataset

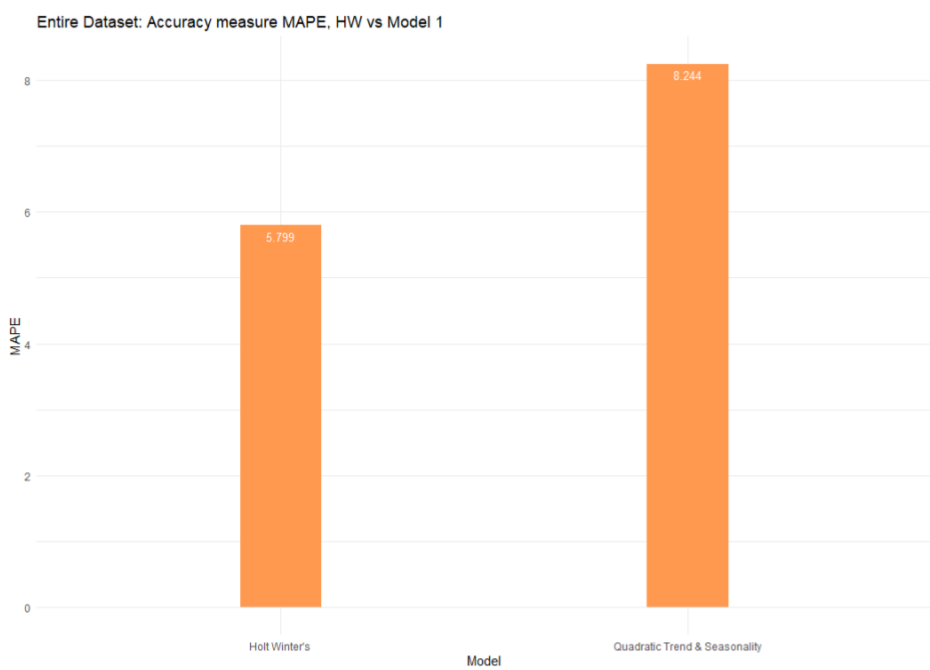


Figure 26: MAPE Accuracy measures of HW vs Model 1: Entire Dataset

Comparing RMSE and MAPE accuracy measures of both models for the entire dataset, it can be observed that Holt-Winters (M, N, M) model is performing much better than the Regression model with quadratic trend and seasonality because the HW model has lower RMSE and MAPE values. Therefore, we can use Holt-Winters (M, N, M) model for forecasting future 24 months retail sales. Below is the forecast and plot for the same,

	Point Forecast	Lo 95	Hi 95
Jan 2022	1733.865	1409.744	2057.985
Feb 2022	1641.550	1326.091	1957.009
Mar 2022	1869.265	1500.513	2238.017
Apr 2022	1521.149	1213.505	1828.794
May 2022	1782.920	1413.675	2152.166
Jun 2022	1792.235	1412.551	2171.919
Jul 2022	1745.454	1367.571	2123.337
Aug 2022	1770.978	1379.510	2162.445
Sep 2022	1813.753	1404.738	2222.768
Oct 2022	1930.586	1486.775	2374.398
Nov 2022	2829.754	2167.078	3492.430
Dec 2022	3500.935	2666.300	4335.570
Jan 2023	1734.859	1289.836	2179.883
Feb 2023	1642.492	1214.805	2070.178
Mar 2023	1870.338	1376.185	2364.490
Apr 2023	1522.022	1114.167	1929.876
May 2023	1783.943	1299.278	2268.608
Jun 2023	1793.263	1299.493	2287.033
Jul 2023	1746.456	1259.254	2233.658
Aug 2023	1771.993	1271.334	2272.653
Sep 2023	1814.794	1295.632	2333.955
Oct 2023	1931.694	1372.349	2491.039
Nov 2023	2831.377	2001.749	3661.006
Dec 2023	3502.943	2464.586	4541.300

Figure 27: Future 24 months retail sales forecast for HW (M, N, M) model

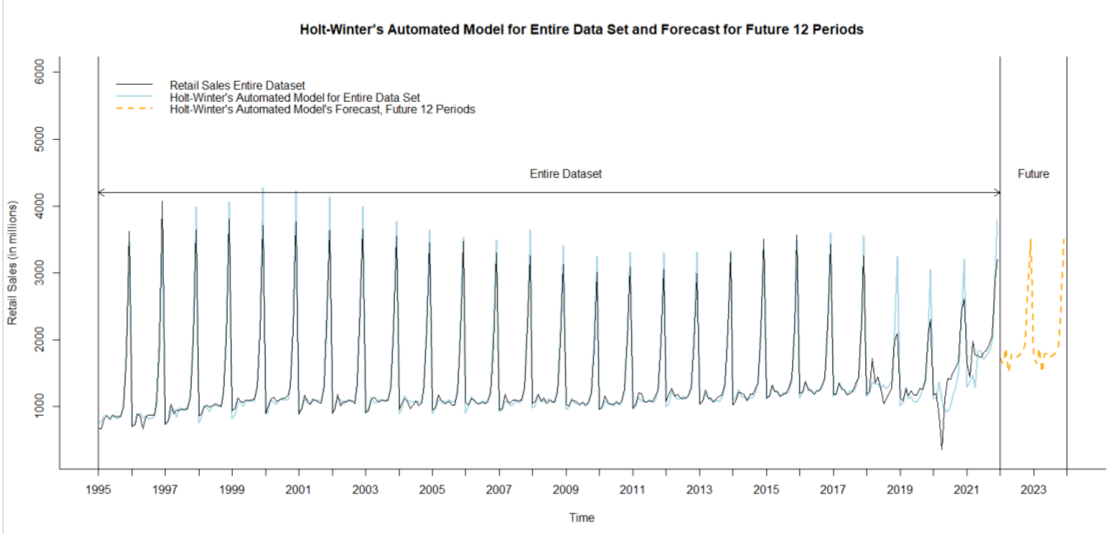


Figure 28: Future 24 month's forecast: Holt Winter's model

MODEL 3: Two-Level Forecast Model (Regression model with Quadratic trend and seasonality And Autoregression for Residuals)

Combining methods can be done via two-level (or multilevel) methods, where the first method uses the original time series to generate forecasts of future values, and the second method uses the forecast errors from the first layer to generate forecasts of future forecast errors, thereby "correcting" the first level forecasts. In this model, we use the **Regression Model with Quadratic Trend and Seasonality** as the first level model. And **AR (12) model** for residuals from the first level forecast. This approach captures autocorrelation by constructing a second-level forecasting model for the residuals, as follows:

1. Generate a k-step-ahead forecast of the series (F_{t+k}), using a forecasting method.

2. Generate k-step-ahead forecast of the forecast error (e_{t+k}), using an AR (or other) model.

3 Improve the initial k-step-ahead forecast of the series by adjusting it according to its forecasted error: Improved $(F_{t+k})^* = (F_{t+k}) + (e_{t+k})$

This three-step process means that we fit a low-order AR model to the series of residuals (or forecast errors) that is then used to forecast future residuals. By fitting the series of residuals, rather than the raw series, we avoid the need for initial data transformations (because the residual series is not expected to contain any trends or cyclical behavior besides autocorrelation). To fit an AR model to the series of residuals, we first examine the autocorrelations of the residual series. We then choose the order of the AR model according to the lags in which autocorrelation appears.

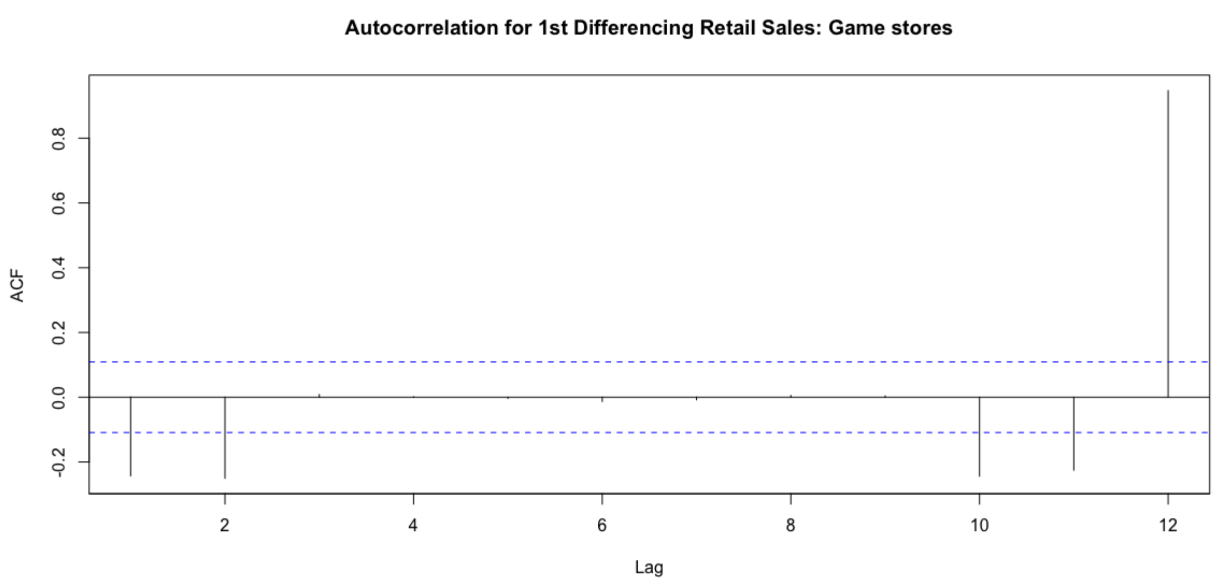


Figure 29: Autocorrelation for 1st Differencing AR(1)

As seen in the figure above, lag 12 of the 1st differencing has the highest value of significant autocorrelation, hence we choose the AR (12) model as the second-level forecast model.

```
Series: train.quadratic.seasonality$residuals
ARIMA(12,0,0) with non-zero mean

Coefficients:
         ar1     ar2     ar3     ar4     ar5     ar6      ar7     ar8      ar9     ar10
      0.1286  0.0233  0.0524  0.0191  0.0409  0.0463  -0.0552  0.0523  -0.0552  -0.0429
s.e.  0.0451  0.0458  0.0456  0.0456  0.0455  0.0440   0.0453  0.0454   0.0455   0.0454
        ar11    ar12     mean
      0.0259  0.6916   5.7936
s.e.  0.0455  0.0447  50.2067

sigma^2 = 6718:  log likelihood = -1395.86
AIC=2819.72   AICc=2821.59   BIC=2868.45
```

Figure 30: AR (12) model summary: Training Dataset

Using the above coefficients of the AR (12) model, we generate a forecast for residuals to add to the forecast values obtained from the Regression model using Quadratic trend and Seasonality.

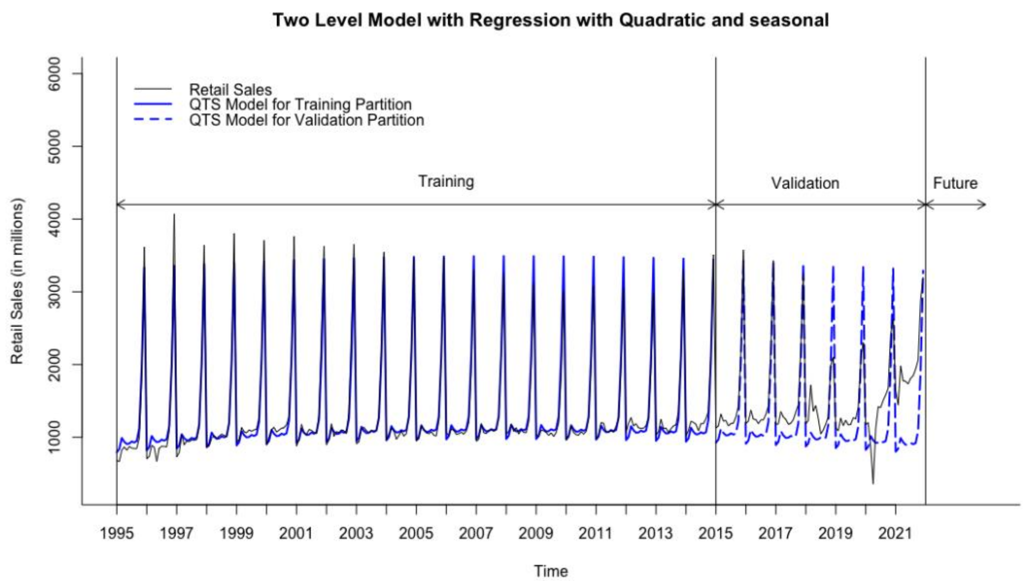


Figure 31: Two-level model fitted onto the training dataset
and forecasted for Validation period.

Once we get a good model fit, we run the same model for the entire dataset with residuals from the Regression model of Quadratic trend + Seasonality.

```
Series: quadratic.trend.seasonality.for$residuals
ARIMA(12,0,0) with non-zero mean

Coefficients:
         ar1     ar2     ar3     ar4     ar5     ar6      ar7     ar8      ar9     ar10
      0.2540  0.0953  0.0835  0.0906  0.0524  0.0145  -0.0280  0.0220  -0.1182  -0.0480
s.e.  0.0443  0.0463  0.0465  0.0463  0.0466  0.0457   0.0466  0.0465   0.0462   0.0466
         ar11    ar12      mean
      -0.1105  0.6275   45.6184
s.e.   0.0465  0.0449  104.9825

sigma^2 = 22207:  log likelihood = -2077.94
AIC=4183.87   AICc=4185.23   BIC=4236.81
```

Figure 32: AR(12) model summary: Entire Dataset

We get 12 coefficients different from the original model on the training dataset. We get below auto-correlation between residuals of residuals for the entire dataset.

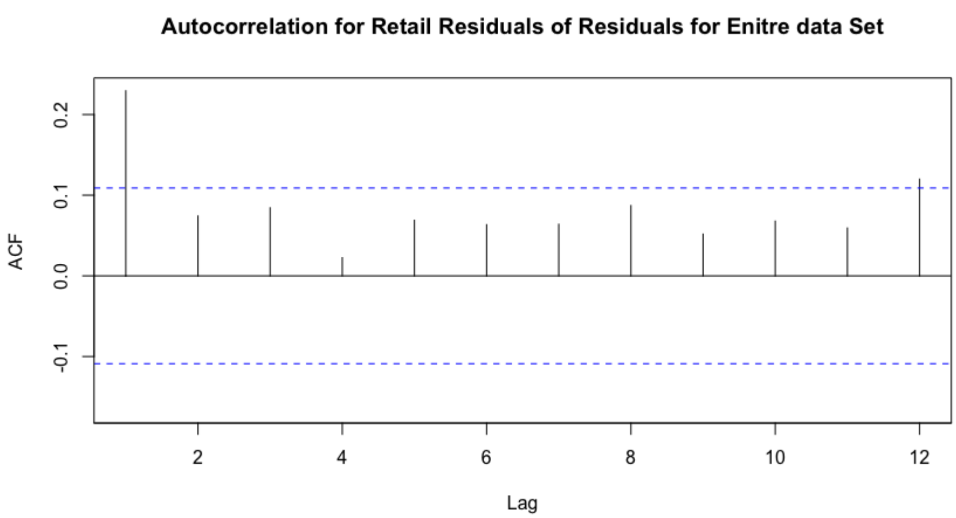


Figure 33: Autocorrelation for Revenue Residuals of Residuals

We observe a significant drop in the autocorrelation values amongst all different lags. And less correlation in lag 1. After applying this two-level forecast model for the entire dataset, we get the following forecast results.

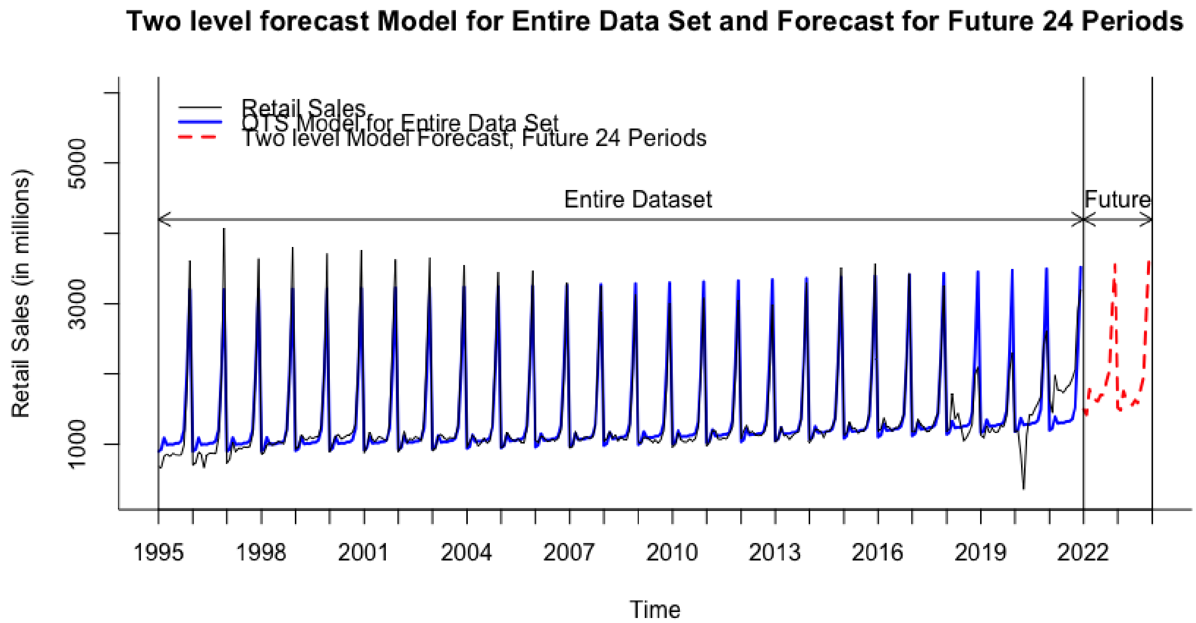


Figure 34: Two-level forecast model for entire Retail Sales dataset

MODEL 4: AUTO REGRESSIVE INTEGRATED MOVING AVERAGE(ARIMA):

Auto-Regressive Integrated **M**oving **A**verage is a model that is capable of presenting every time series component like trend, seasonality, and level as the approach can include up to 6 parameters. Non-seasonal ARIMA includes three parts Auto-Regressive, Integrated, and Moving Average. It only considers level and trend but no seasonality.

AUTOREGRESSIVE(AR):

Auto-Regressive model is a type of model where it models the autocorrelation directly in a regression model using past observations as predictors. The term autocorrelation indicates that it is a regression of the variable against itself. Auto-Regressive models can be built of any order depending on the autocorrelation in the data. Below are the equations and representations of various orders of AR models. A pure Auto-Regressive (AR only) model is one where Yt depends only on its lags. That is, Yt is a function of the 'lags of Yt'. It is represented as AR (p,0,0) where p is the order of the model. p represents the lag order. AR Model Equation of Order p:

$$Y_t = \beta_0 + \beta_1 * Y_{t-1} + \beta_2 * Y_{t-2} ...... + \beta_p * Y_{t-p} + \varepsilon_t$$

Below is the summary of AR,

```
> summary(Arima(retailsales.ts,c(2,0,0))) #AR(2)
Series: retailsales.ts
ARIMA(2,0,0) with non-zero mean

Coefficients:
         ar1      ar2       mean
      0.3458  -0.1775  1394.7669
s.e.  0.0550   0.0555    43.0165

sigma^2 = 418487:  log likelihood = -2555.3
AIC=5118.6   AICc=5118.72   BIC=5133.72

Training set error measures:
                    ME     RMSE      MAE       MPE     MAPE     MASE         ACF1
Training set 0.0192902 643.9036 443.3862 -14.12355 31.25796 4.499041 -0.001269666
```

Figure 35: AR (2) model summary and accuracy measures

From the above model summary, we can interpret that ar1 0.3458, ar2 -0.1775 are coefficients with mean as 1394.7669. Where Yt-1 and Yt-2 are preceding time period values. Below is the formula for the AR model.

$$Y_t = 1394.7669 + 0.3458 * Y_{t-1} - 0.1775 * Y_{t-2}$$

MOVING AVERAGE(MA):

A pure Moving Average (MA only) model is one where Yt depends only on the lagged forecast errors. It works by analyzing the errors from the lagged observations. The Moving Average of order q is represented as ARIMA (0, 0, q). Below is the equation for MA of order q

$$Y_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} \ldots + \theta_q \varepsilon_{t-q}$$

Where c= constant mean of MA model

ε_t is error term (other coefficients are selected in a way to minimize this error) ε_{t-1}, ε_{t-2},... ε_{t-q} represents error terms of lagged time periods

$\theta_1, \theta_2, .. \theta_q$ represents coefficients of variables to be estimated

The following image shows the summary of MA.

```
> summary(Arima(retailsales.ts,c(0,0,2))) # MA(2)
Series: retailsales.ts
ARIMA(0,0,2) with non-zero mean

Coefficients:
         ma1      ma2       mean
      0.3480  -0.0469  1395.1093
s.e.  0.0576   0.0596    46.5911

sigma^2 = 419838:  log likelihood = -2555.82
AIC=5119.64   AICc=5119.76   BIC=5134.76

Training set error measures:
                    ME     RMSE     MAE       MPE     MAPE     MASE        ACF1
Training set 0.4877908 644.9426 434.907 -14.10615 30.31035 4.413002 0.002789754
```

Figure 36: Moving average MA (2) model summary and training accuracy measures

The formula for MA is as below:

$$Y_t = 1395.1093 + 0.3480\varepsilon_{t-1} - 0.0469 * \varepsilon_{t-2}$$

Here, the ma1 value is 0.3480 and the ma2 value is -0.0469. The mean value is 1395.1093.

INTEGRATED (I):

Term **I** ("Integrated") represents the differencing operation in ARIMA. It is the difference between values at lagged periods (d). It represents the differencing of raw observations to allow for the time series to become stationary (i.e., data values are replaced by the difference between the data values and the previous values). Differencing will help in stabilizing the mean and will remove the trend from the data.

Typically, auto regressive and Moving Average models work best with the data that has no trend or/and seasonality. So, to remove the trend from the data and to stabilize the data around mean or to make it stationary we introduce differencing into the picture, which can be achieved using ARIMA (0, d,0) where d is the level or order of differencing.

Below is the representation of how different levels of differencing happened with the value of d.

$d = 0$: no differencing (series does not have a trend), yt

$d = 1$: difference the series once which can remove linear trend,

$d = 2$: difference the series twice, each time of lag-1 (first difference of the first

difference).

ARIMA (p, d, q):

ARIMA (p, d, q) model is used to forecast data with level and trend components – non-seasonal ARIMA model. Non-Seasonal Arima model does not include seasonality which is why it does not work best for a data that has seasonality. Seasonal patterns need to be removed from data which then makes the data more stationary.

SEASONAL ARIMA MODEL:

In order to overcome the shortcomings of ARIMA model, more parameters like P, D, Q were introduced which together made the Seasonal ARIMA model. Seasonal ARIMA model is represented as **ARIMA (p, d, q)(P,D,Q)[m].** The meanings of its parameters are given below:

p: order of autoregressive model AR (p) (number of autocorrelation lags included)

d: order of differencing in AR model (indicates how many rounds of lag-1 differencing are performed to remove certain trend)

q: order of moving average MA(q) (number of residuals' autocorrelation lags included)

P: order of autoregressive seasonal model AR (P) (number of autocorrelation lags included)

D: order of differencing in AR seasonal model (indicates how many rounds of lag-1 differencing the are performed to remove certain trend)

Q: order of moving average MA(Q) (number of residuals' autocorrelation lags included)

m: number of seasons

Seasonality m is identified by the type of time series data used.

Below image represents the summary of seasonal ARIMA model (for training data).

```
> summary(train.arima.seas)
Series: train.ts
ARIMA(2,1,2)(1,1,2)[12]

Coefficients:
          ar1     ar2     ma1      ma2      sar1    sma1     sma2
      -0.7456  0.2457  0.0664  -0.9111  -0.9054  0.5039  -0.3077
s.e.   0.0792  0.0753  0.0422   0.0402   0.1574  0.1795   0.1060

sigma^2 = 6332:  log likelihood = -1314.15
AIC=2644.3   AICc=2644.96   BIC=2671.7

Training set error measures:
                    ME     RMSE      MAE       MPE     MAPE      MASE       ACF1
Training set 0.3113578 76.18746 50.72159 0.1562264 3.708965 0.8132575 0.00602314
```

Figure 37: ARIMA (2, 1, 2) (1,1,2) [12] model summary and training accuracy measures

As we can interpret from the above summary, the training set error measures provide an RMSE of 76.18746 and MAPE of 3.708965.

Below is the image of a plot that represents the forecasted data of the validation period.

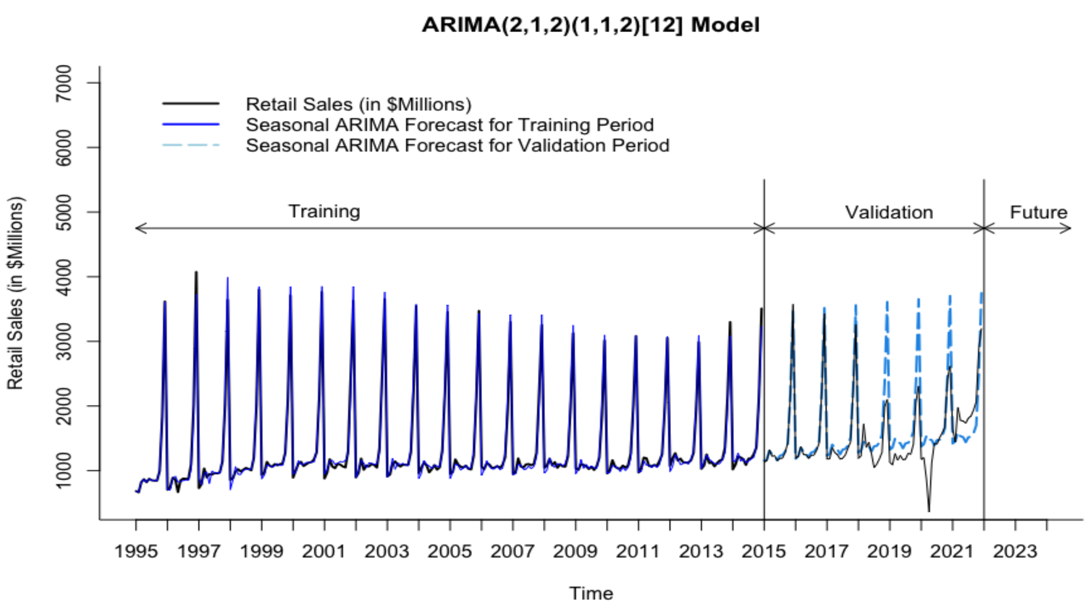


Figure 38: ARIMA(2,1,2)(1,1,2)[12] Model for training data

The black line graph represents original retail sales. The dashed graph in light blue colour represents the forecasted data. There is a difference between the forecasted data and original data between the period of 2019 to mid 2021. This was the time when COVID 19 had hit the world. So, this graph from the validation period will help us correctly identify the forecast for the future period of 24 months from 2022 to 2023 end.

Below is the summary of seasonal ARIMA model for the entire data.

```
> summary(arima.seas)
Series: retailsales.ts
ARIMA(2,1,2)(1,1,2)[12]

Coefficients:
          ar1     ar2      ma1      ma2     sar1    sma1     sma2
      -0.1656  0.0538  -0.2888  -0.3760  -0.8462  0.4884  -0.2574
s.e.   0.3009  0.1461   0.2938   0.2504   0.1820  0.1933   0.0972

sigma^2 = 18587:  log likelihood = -1967.5
AIC=3951   AICc=3951.48   BIC=3980.92

Training set error measures:
                   ME     RMSE      MAE       MPE     MAPE     MASE         ACF1
Training set 5.273417 132.0581 72.99802 0.1106165 5.596186 0.740711 -0.004415422
```

Figure 39: Summary of ARIMA(2,1,2)(1,1,2)[12] Model for entire dataset

We can interpret from the above summary, the training set error measures provide an RMSE of 132.0581 and MAPE of 5.596186. The model indicates that we have the first difference, first order seasonal difference, third order auto regressive model, no auto regressive model for seasonality, non-seasonal second order MA for error lags and seasonal second order MA for error lags. Model equation can be represented as below,

$$y_t - y_{t-1} = -0.1656\,(y_{t-1} - y_{t-2}) + 0.0538(y_{t-2} - y_{t-3}) - 0.2888\varepsilon_{t-1} - 0.3760\,\varepsilon_{t-2} - 0.8462(y_{t-1} - y_{t-13})$$

$$-0.4884\,\rho_{t-1} + 0.2574\,\rho_{t-2}$$

From the model equation, we can see that it is first order differenced as we have yt- yt-1 on the left side of the equation. - 0.1656(ar1), 0.0538(ar2) are the coefficients of the second order auto regressive model, -1.1567(ma1) and 0.9889(ma2) are the coefficients of the second order moving average for error lags. yt-1 -yt-2, yt-2 -yt-3, yt-3 -yt-4 represents elements of the first order difference. εt-1, εt-2 are error terms of second order auto regressive model. -0.4884(sma1), - 0.2574(sma2) are the coefficients of seasonal second order moving average for error lags. ρt-1, ρt-2 are error terms of the second order seasonal auto regressive model.

The ARIMA (2, 1, 2) (1,1,2) [12] has a log likelihood of -1967.5, BIC as 3980.92 , AICc as 3951.48 and AIC as 3951. These metrics can be used to compare with other models with same differencing orders.

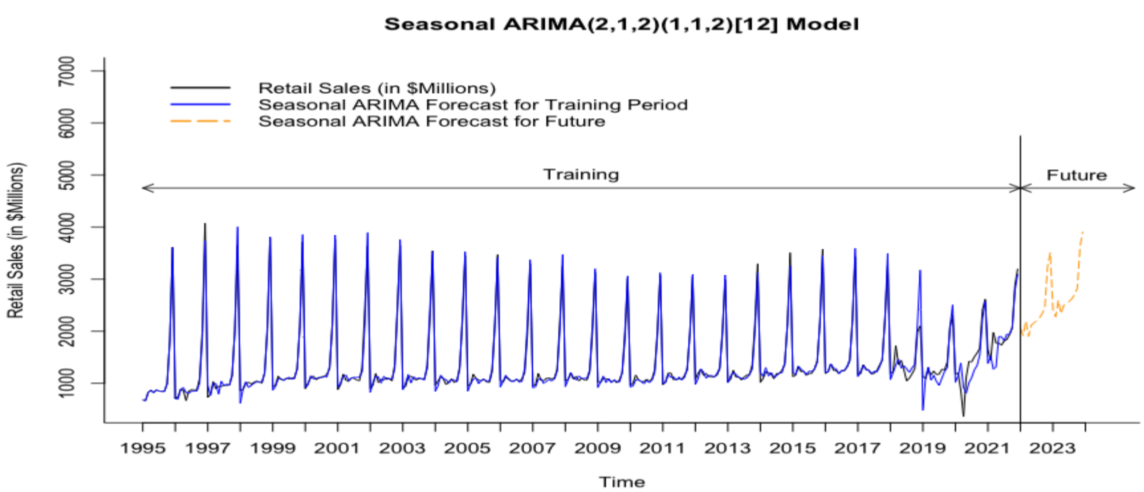


Figure 40: Visualization of future 24 months using Seasonal ARIMA (2, 1, 2) (1, 1, 2) [12] model

We can interpret from the above graph that the blue orange colored dashed line represents the future forecast. The graph below shows the 80-95% confidence interval of the Seasonal ARIMA model for future 24 months, particularly from January 2022 to December 2023.

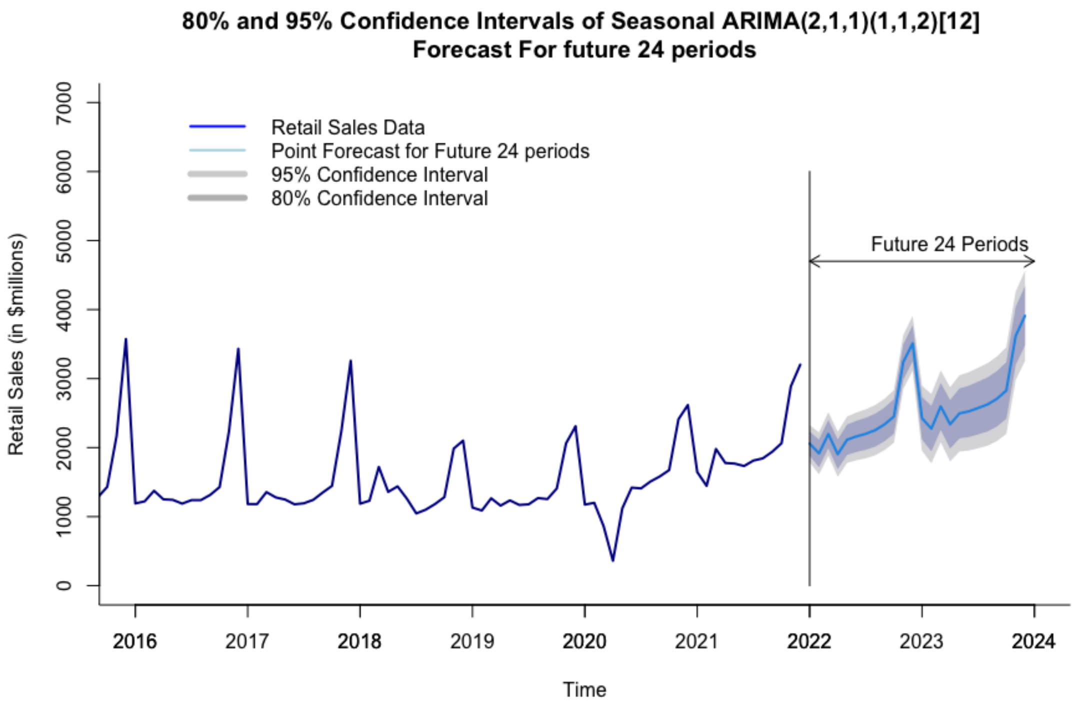


Figure 41: Confidence Intervals for future 24 months using Seasonal ARIMA(2,1,1)(1,1,2)[12]

```
> # Arima
> round(accuracy(arima.seas.pred$fitted, retailsales.ts), 3)
             ME    RMSE    MAE   MPE  MAPE   ACF1 Theil's U
Test set 5.273 132.058 72.998 0.111 5.596 -0.004     0.287
```

Figure 42: Accuracy of Auto ARIMA for entire dataset

MODEL 5: Auto ARIMA Model:

In order to determine optimal values for components in the ARIMA model which will be a hectic task so an automated model Auto ARIMA is introduced. This model selects the parameter values based on many conditions such as AIC, AICc, BIC, accuracy, and log likelihood values. A model with less complexity or less AIC or BIC values with higher log likelihood is given preference as a best model.

Below figure represents the summary of Auto ARIMA model for training data.

```
> summary(train.auto.arima)
Series: train.ts
ARIMA(0,1,2)(2,1,1)[12]

Coefficients:
          ma1      ma2     sar1     sar2    sma1
      -0.6743  -0.2154  -0.6499  -0.2158  0.2151
s.e.   0.0695   0.0699   0.3308   0.1455  0.3310

sigma^2 = 6221:  log likelihood = -1313.13
AIC=2638.26   AICc=2638.64   BIC=2658.81

Training set error measures:
                    ME     RMSE      MAE       MPE     MAPE      MASE         ACF1
Training set 0.6770442 75.85546 50.31429 0.1786246 3.692535 0.8067271 -0.00754258
```

Figure 43: Summary for training data of Auto ARIMA model

From the above summary, we can interpret that it has -0.6743(ma1) and -0.2154(ma2) as the coefficients of the second order moving average for error lags. ε_{t-1}, ε_{t-2} are the error terms of the second-order autoregressive model. (y_{t-1} -y_{t-13}) and (y_{t-1} -y_{t-14}) are the terms for sar1(-0.6499) and sar2(-0.2158). - 0.2151(sma1) is the coefficient of the seasonal moving average for error lags. ρ_{t-1} is the error term of the seasonal autoregressive model.

The ARIMA (0, 1, 2) (2, 1, 1) [12] has a log-likelihood of -1313.13, BIC as 2658.81, AICc as 2638.64 and AIC as 2638.26. These metrics can be used to compare with other models.

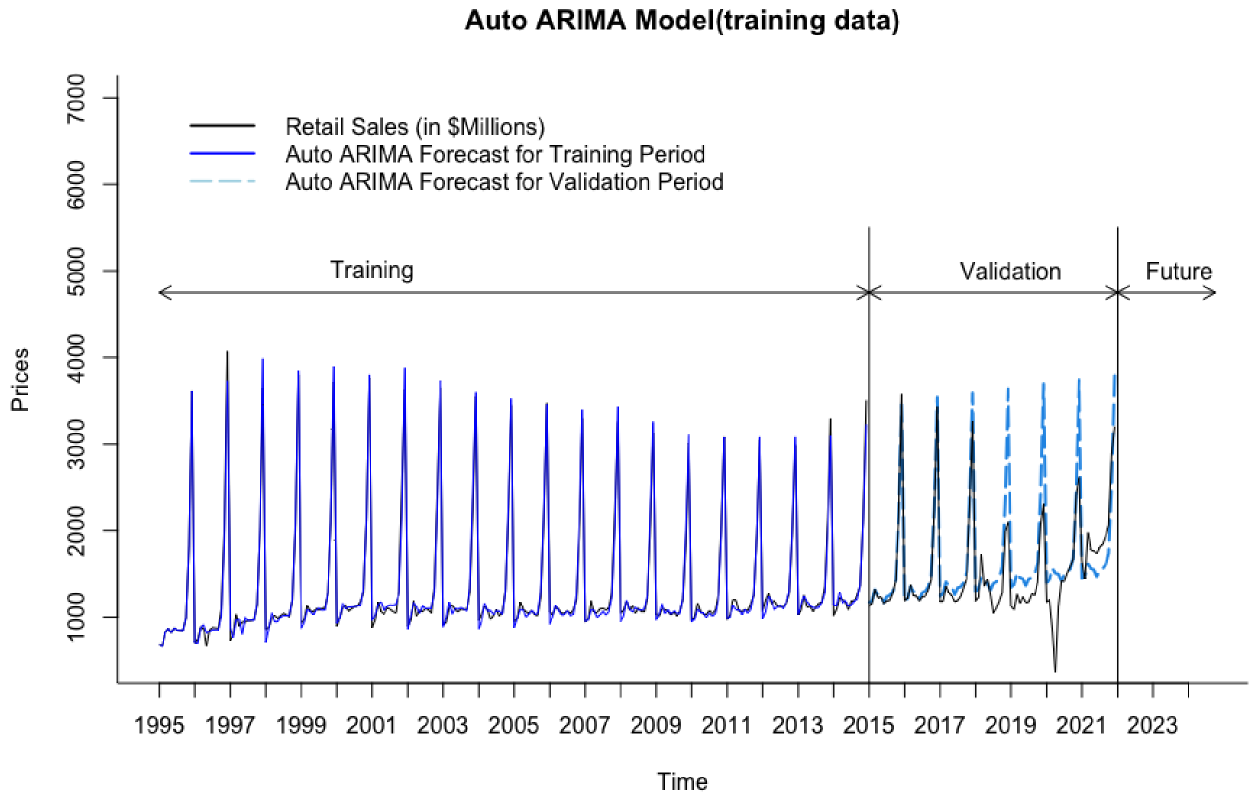


Figure 44: Auto ARIMA Model for training data

The above figure represents the graph for the training dataset of Retail sales. The light blue dashed line represents the forecast performed on validation periods based on the training data. We can see that the forecasted value is a bit different from the original values in the period 2019 to 2022.

Auto ARIMA for the entire dataset:

Here, we have applied the Arima (1, 0, 2) (1, 1, 2) model which was chosen by the auto-arima function on the entire data set of Retail Sales.

The following figure represents the summary of Auto ARIMA for the entire dataset.

```
> summary(auto.arima.full)
Series: retailsales.ts
ARIMA(1,0,2)(1,1,2)[12]

Coefficients:
         ar1      ma1      ma2      sar1    sma1     sma2
      0.9486  -0.4040  -0.2096   -0.8555  0.4980  -0.2470
s.e.  0.0342   0.0712   0.0650    0.1458  0.1594   0.0899

sigma^2 = 18331:  log likelihood = -1972.24
AIC=3958.48   AICc=3958.85   BIC=3984.69

Training set error measures:
                   ME     RMSE      MAE       MPE     MAPE      MASE        ACF1
Training set 8.788337 131.5791 72.50279 0.3636998 5.547737 0.7356859 -0.01207101
```

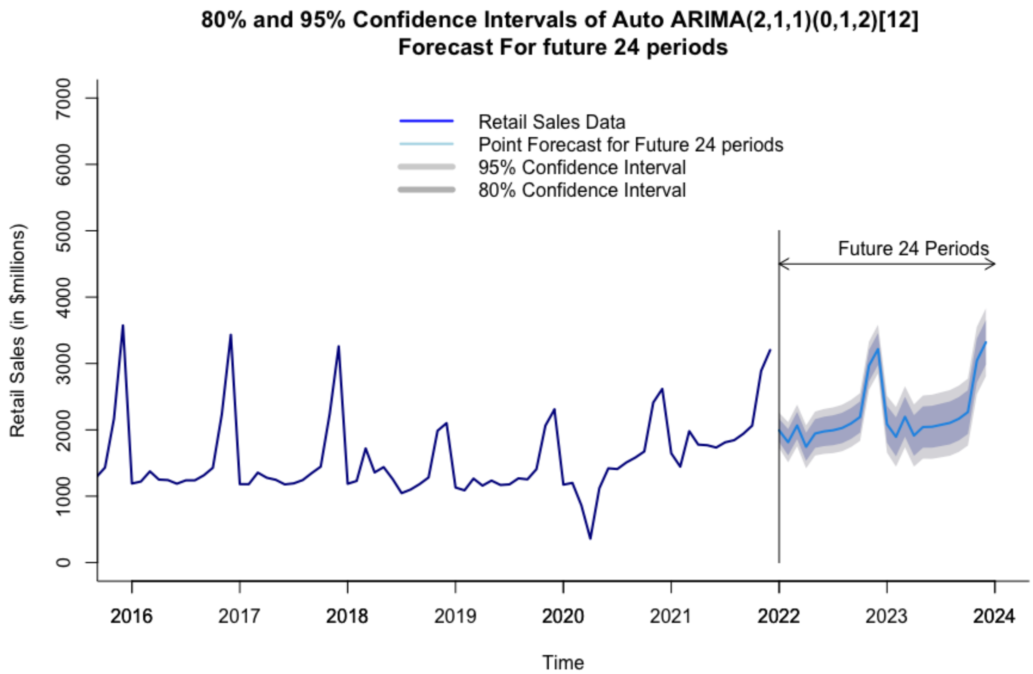
Figure 45: Summary of Auto ARIMA (1, 0, 2) (1, 1, 2) [12] model for the entire dataset

From the above summary we can infer that the model is a seasonal ARIMA model. It has first order 1 autoregressive model AR (1), no differencing to remove linear trend, order 2 moving average for error lags, order 1 autoregressive model for seasonality, order 1 differencing to remove linear trend for seasonality and order 2 moving average MA (2) for error lags for seasonality. Model equation can be represented as below,

$$y_t - y_{t-1} = - 0.9486 (y_{t-1} - y_{t-2}) - 0.4040\, \varepsilon_{t-1} - 0.2096\, \varepsilon_{t-2} - 0.8555(y_{t-1} - y_{t-13}) - 0.4980\, \rho_{t-1} - 0.2470\, \rho_{t-2}$$

The ARIMA (1, 0, 2) (1, 1, 2) [12] has a log-likelihood of -1972.24, BIC as 3984.69, AICc as 3958.85 and AIC as 3958.48. These metrics can be used to compare with other models with the same differencing orders.

The following image represents Confidence Intervals for the forthcoming 24 months of the Retail Sales dataset from 2022 January to 2023 December.



s

Figure 46: Confidence Intervals for future 24 months using Auto ARIMA (1, 0, 2) (1, 1, 2) [12]

The image below shows the forecast from January 2022 to December 2023, based on the historical data of Retail Sales dataset.

```
> auto.arima.full.pred$mean
          Jan      Feb      Mar      Apr      May      Jun      Jul      Aug      Sep
2022 1991.380 1813.633 2065.937 1744.268 1944.192 1976.350 1993.342 2027.413 2095.929
2023 2084.923 1894.475 2196.431 1914.125 2041.744 2046.096 2075.077 2104.452 2167.269
          Oct      Nov      Dec
2022 2192.766 2967.938 3216.791
2023 2267.721 3043.822 3319.139
```

Figure 47: Forecast for future 24 months using Auto ARIMA model

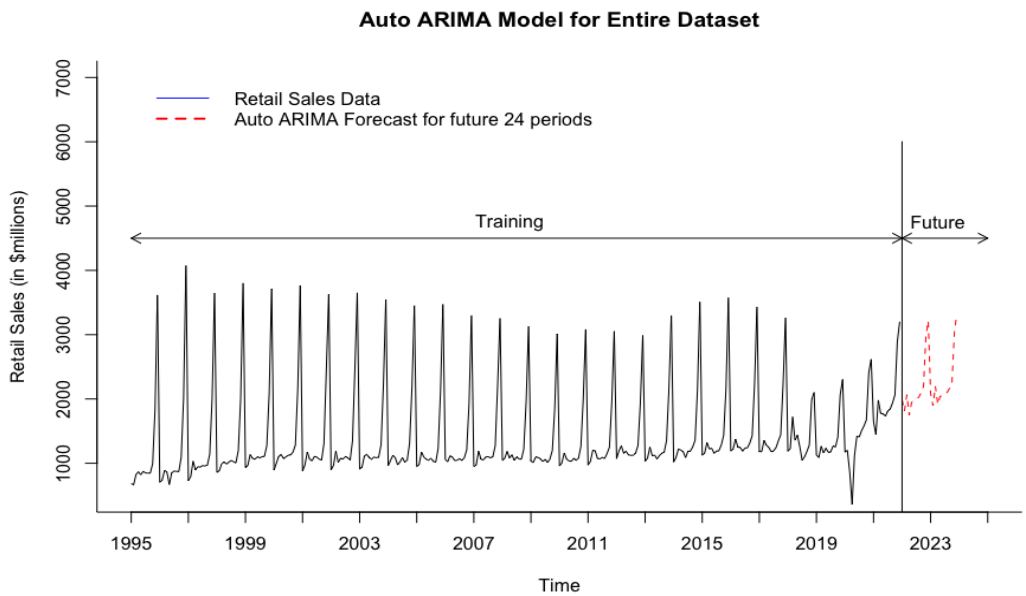


Figure 48: Visualization of future 24 months using Auto ARIMA (1, 0, 2) (1, 1, 2) [12] model

The above image shows the future predictions with the use of the Auto ARIMA model. The forecast is performed from January 2022 to December 2023.

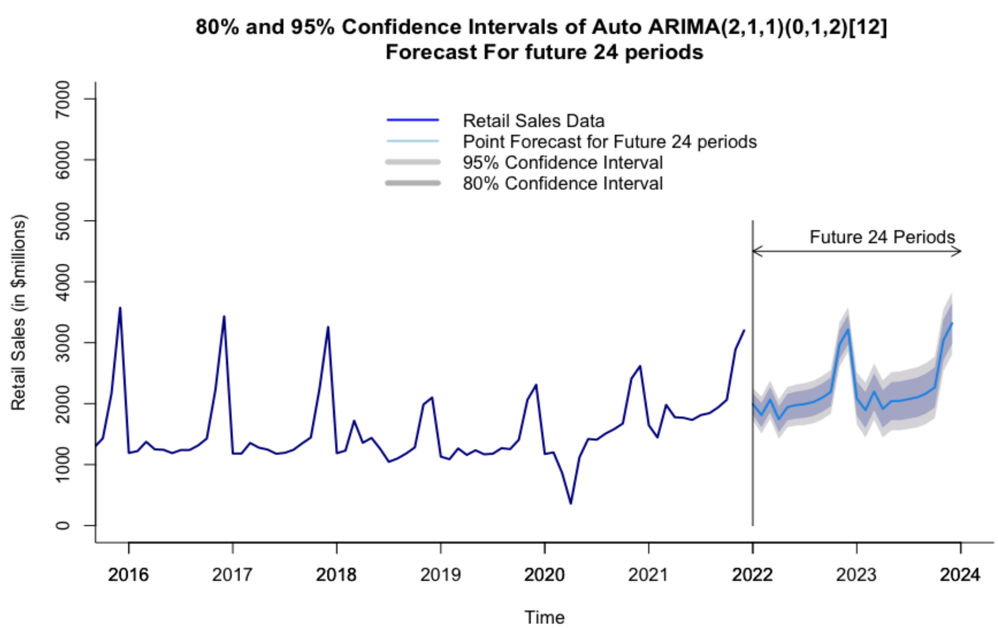


Figure 49: Confidence Intervals for future 24 months using Auto ARIMA (1, 0, 2) (1, 1, 2) [12]

```
> round(accuracy(auto.arima.full.pred$fitted, retailsales.ts), 3)
            ME    RMSE    MAE   MPE  MAPE   ACF1 Theil's U
Test set 8.788 131.579 72.503 0.364 5.548 -0.012     0.284
```

Figure 50: Accuracy for Auto ARIMA (1, 0, 2) (1, 1, 2) [12] model

Step-7: Evaluate and Compare Performance

Forecast Method	ME	RMSE	MAE	MPE	MAPE	ACF1	THEIL'S U
Two Level	**2.29**	**101.43**	**68.14**	**-1.18**	**5.18**	**0.16**	**0.21**
Auto ARIMA	**8.79**	**131.58**	**72.50**	**0.36**	**5.55**	**-0.01**	**0.28**
Seasonal ARIMA (2, 1, 1) (1, 1, 2) [12]	**5.27**	**132.06**	**73.00**	**0.11**	**5.60**	**0.00**	**0.29**
Holt-Winters Model	**-9.35**	**158.79**	**84.54**	**-0.31**	**5.80**	**0.18**	**0.30**
Snaïve	**33.28**	**185.10**	**98.55**	**1.52**	**7.20**	**0.60**	**0.35**
Regression QTS	**0.00**	**204.57**	**114.36**	**2.01**	**8.24**	**0.36**	**0.39**
Regression LTS	**0.00**	**205.77**	**115.57**	**-1.97**	**8.19**	**0.37**	**0.38**
Naïve	**7.80**	**809.37**	**414.52**	**-11.50**	**30.98**	**-0.24**	**1.00**

Table 1: A comparison for all the accuracy measures of all forecasting methods used above.

Let's compare RMSE and MAPE accuracy measures to ensure we have a good fit model to forecast the 'Retail Sales' time series.

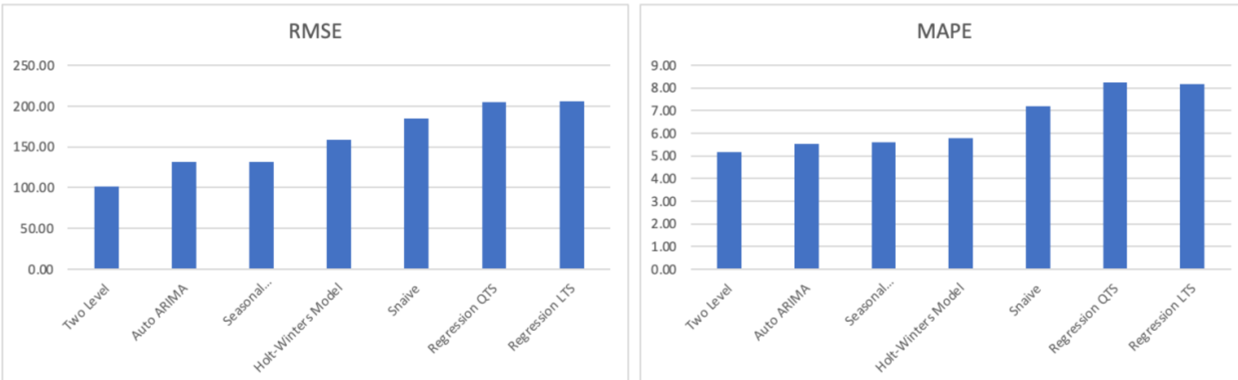


Figure 51: RMSE and MAPE comparison for all models

We can observe from the above models that the two-level forecast (Holt-Winter's Automatic Model with optimal parameters + AR (12) model for residuals) has lesser MAPE and RMSE values among all the models. Although two-level forecast (Regression Model with Quadratic Trend and seasonality + AR (12) model for residuals) is best in terms of accuracy one should notice that AR (12) model is a complex model with 12 variables and an ensemble model will increase cost and computational time in real-time. If complexity and computational time are not an issue, we can choose the two-level forecast (Holt-Winter's Automatic Model with optimal parameters + AR (12) model for residuals) as the best model for forecasting into the future. Else Arima (3, 1, 2) (0, 1, 2) model can be chosen which closely follows two-level forecasts in terms of forecasting accuracy and uses less computational complexity and time.

Step-8: Implement Forecast System

As seen from the comparison table (for the entire data set) which compares the performance of all the models to choose the best one, it is evident that the Auto ARIMA model gives the best prediction. This is because of the low values of MAPE (5.55) and RMSE (131.58). This is the recommended model to implement forecasting of Retail Sales dataset.

The following image gives a visualization of confidence intervals which can be referred by companies to perform forecast based analysis.

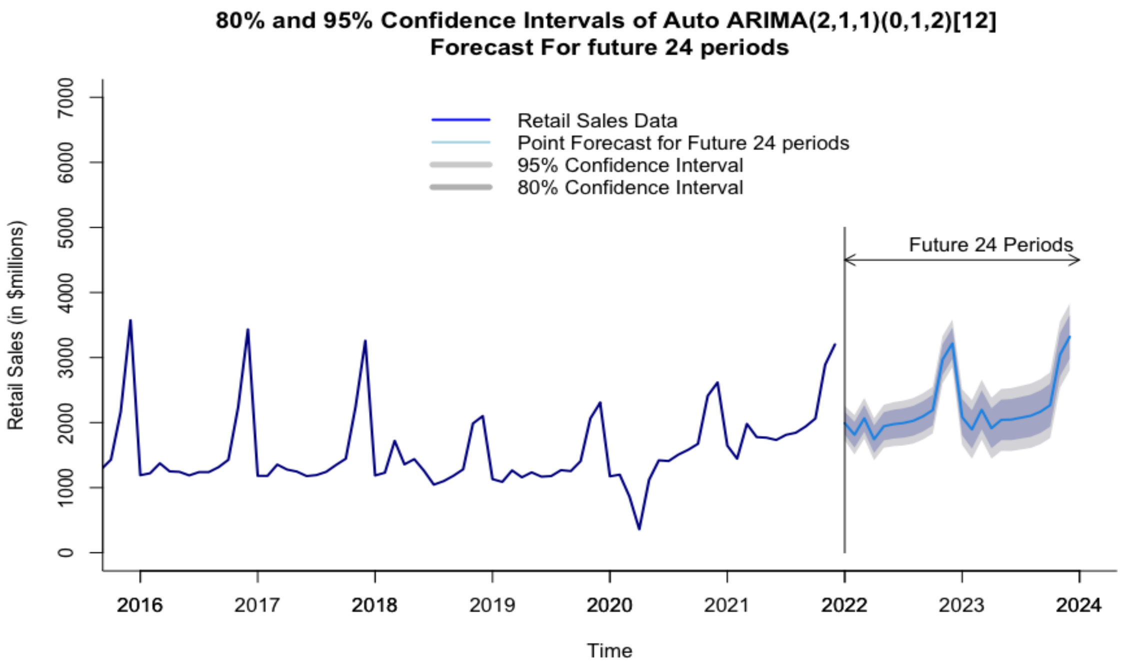


Figure 52: Confidence Intervals for future 24 months using Auto ARIMA (1, 0, 2) (1, 1, 2) [12]

After choosing the best forecasting model, one should make sure that new data is added from time to time and the forecast is performed at timely intervals. A reevaluation must be done at regular intervals as the new data gets added to the historical data.

As this data is monthly data, a reevaluation must be done quarterly to ensure the best growth of the company.

Conclusion

After the analysis of all the accuracy measures, especially RMSE and MAPE, and considering the computational complexity and time, we recommend using the Auto-ARIMA method for forecasting the Retail Sales. It incorporated the trend, seasonality, and the residuals of the time series in the most efficient manner. For better accuracy, at the cost of complexity, the user can utilize a Two-Level Forecast for Regression with quadratic trend and seasonality and AR (12) for residuals.

Appendix

Training Data

```
> train.ts
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1995	683	665	823	867	821	873	851	846	846	993	1830	3616
1996	705	737	890	862	669	849	877	874	870	1094	2002	4073
1997	730	791	1032	896	947	940	963	958	971	1141	2009	3643
1998	858	883	987	1022	986	1016	1042	1025	1005	1197	2011	3801
1999	933	966	1135	1076	1061	1101	1083	1101	1111	1277	2096	3711
2000	893	1002	1106	1136	1072	1098	1124	1132	1175	1285	2161	3763
2001	880	971	1176	1063	1038	1107	1094	1065	1050	1303	2445	3628
2002	896	966	1186	1014	1078	1065	1103	1087	1048	1347	2468	3651
2003	910	940	1111	1139	1088	1064	1094	1087	1095	1286	2223	3545
2004	962	1045	1125	1082	974	1024	1086	1016	1051	1311	2187	3451
2005	950	980	1173	1098	1061	1049	1077	1028	1019	1229	2120	3471
2006	1063	1022	1117	1094	1040	1048	1070	1043	1086	1205	1933	3299
2007	948	974	1188	1064	1050	1096	1102	1084	1102	1275	2208	3253
2008	1045	1078	1185	1082	1132	1045	1100	1067	1063	1221	1964	3125
2009	1034	1012	1099	1089	1068	1025	1062	1011	1067	1200	1850	3013
2010	958	1005	1161	1065	1036	1030	1077	1046	1080	1262	2003	3082
2011	972	1030	1201	1192	1073	1069	1089	1074	1151	1252	1964	3052
2012	1072	1192	1270	1156	1186	1130	1123	1126	1158	1275	1945	2987
2013	1029	1079	1250	1117	1127	1067	1118	1156	1172	1351	2156	3298
2014	1020	1091	1221	1204	1178	1088	1186	1189	1254	1358	2140	3507

Validation Data

```
> valid.ts
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2015	1127	1157	1320	1222	1232	1154	1189	1196	1281	1431	2168	3573
2016	1191	1221	1374	1250	1243	1188	1239	1239	1315	1428	2234	3432
2017	1182	1181	1355	1278	1247	1178	1193	1243	1348	1445	2239	3258
2018	1187	1230	1720	1358	1438	1259	1046	1100	1182	1282	1986	2101
2019	1131	1088	1265	1159	1235	1169	1178	1269	1254	1407	2064	2305
2020	1174	1200	862	360	1120	1420	1409	1508	1582	1674	2412	2617
2021	1646	1445	1980	1777	1768	1733	1812	1844	1939	2063	2889	3202

Autocorrelation:

Autocorrelation represents the correlation between a random variable (time series data) itself and the same variable lagged one or more periods. The coefficient of autocorrelation($\boldsymbol{r_k}$) is calculated as below:

$$\mathbf{r_k} = \frac{\sum_{t=k+1}^{n} (\mathbf{Y_t} - \overline{\mathbf{Y}})(\mathbf{Y_{t-k}} - \overline{\mathbf{Y}})}{\sum_{t=1}^{n} (\mathbf{Y_t} - \overline{\mathbf{Y}})^2}$$

Here,

rk = autocorrelation coefficient for a lag of k periods (k = 1, 2, 3, …, 12, …)

Y = mean of the values of the series

Yt= observation in time period t

Yt-k = observation k time periods earlier or at time period t-k

RMSE:

RMSE is Root Mean Square Error. It is a standard way to measure the error of a model in predicting quantitative data. Formally it is defined as follows:

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}}$$

MAPE:

Mean absolute percentage error is abbreviated as MAPE. Formally it is defined as follows:

$$MAPE = \frac{100}{v} \sum_{t=1}^{v} \left| \frac{e_t}{y_t} \right|$$

References

1. Retail Sales Data - St. Louis Fred Economic Data: https://fred.stlouisfed.org/series
2. United States Census Bureau - Monthly Retail Trade: https://www.census.gov/retail/mrts/about_the_surveys.html
3. Show Me the Data: 8 Awesome Time Series Sources: https://opendatascience.com/show-me-the-data-8-awesome-time-series-sources/
4. Research Gate: https://www.researchgate.net/
5. Plotting ts objects: https://cran.r-project.org/web/packages/ggfortify/vignettes/plot_ts.html
6. ARIMA Model – Complete Guide to Time Series Forecasting in Python: https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/
7. Autoregressive Integrated Moving Average (ARIMA): https://www.investopedia.com/terms/a/autoregressive-integrated-moving-average-arima.asp
8. R Color Styles: https://bootstrappers.umassmed.edu/bootstrappers-courses/pastCourses/rCourse_2016-04/Additional_Resources/Rcolorstyle.html#terrain.colors
9. Introduction to Time Series and Plotting techniques in R: https://rpubs.com/RLodge/timeseriesintro
10. R Plot pch Symbols: Different point shapes in R: https://www.r-bloggers.com/2021/06/r-plot-pch-symbols-different-point-shapes-in-r/
11. Custom dygraphs time series example: https://www.r-graph-gallery.com/318-custom-dygraphs-time-series-example.html
12. Dplyr package - Documentation: https://www.rdocumentation.org/packages/dplyr/versions/0.7.8
13. ARIMA Model - Complete Guide to Time Series Forecasting: https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/
14. INVESTOPEDIA (advanced technical concepts): https://www.investopedia.com/terms/a/autoregressive-integrated-moving-average-arima.asp