MONTHLY WHOLESALE TRADE FORECASTING

FOR

MOTOR VEHICLE AND SUPPLIES

Executive Summary

For this project, we use Monthly Wholesale Trade Forecasting for Motor Vehicle and Motor Vehicle Parts and Supplies over a period of around 30 years. The analysis aims to identify patterns of monthly wholesale trade and understand how these patterns vary over time, as well as to develop accurate models for forecasting future trade sales.

Regression-based models, Holt-Winter's Exponential smoothing Model with Automatic Selection, and autoregressive integrated moving average models (ARIMA) were utilized for this project.

The analysis is based on monthly wholesale trade data for over 30 years. From visualization, the data is highly auto correlated, as the autocorrelation coefficient in all 12 lags are significant. Positive autocorrelation coefficients are significantly greater for all the lag coefficients.

Model evaluation was based on the RMSE and MAPE accuracy metrics.

The best and optimal results were obtained when the advanced exponential smoothing model was used with the least error values followed by two level model. Model evaluation was based on the RMSE and MAPE accuracy metrics. It can be concluded that the best wholesale trade pattern can be obtained from HW advanced exponential smoothing model with Automatic Selection of Parameters.

Introduction

The Motor Vehicle and Motor Vehicle Parts and Supplies industry is a crucial contributor to the U.S. economy, and accurate forecasting of monthly wholesale trade in this sector is essential for businesses operating within it. Time series analysis is a powerful tool for understanding this data, as it enables us to model the behavior of the time series over time and make predictions about future wholesale trades.

The primary goal of this project is to perform a time series analysis of Monthly Wholesale Trade Forecasting for Motor Vehicle and Motor Vehicle Parts and Supplies, covering around 30 years of historical data. The analysis aims to identify patterns and seasonality in the data and develop accurate forecasting models to predict future wholesale trade activity. The insights gained through this analysis will be invaluable to businesses in staying ahead of evolving trends and changes in consumer preferences, while making informed plans.

By leveraging sophisticated time series analysis techniques, this project will provide a deep understanding of wholesale trade behavior within the Motor Vehicle and Motor Vehicle Parts and Supplies industry. With this knowledge, businesses can make data-driven decisions that are grounded in a comprehensive understanding of the past and a reliable vision of the future. Ultimately, this project will help organizations in the industry to stay competitive and thrive in a rapidly changing economic landscape.

Eight Steps of Forecasting

Step 1: Define Goal

The objective of this project is to create accurate numeric forecasts for the wholesale trade of motor vehicle and supplies industry for the upcoming four fiscal quarters. The aim is to develop a predictive model that considers both the trend and seasonal components of historical data and accurately forecasts the desired quarters.

The primary goal is to identify the model with the highest level of accuracy. The resulting forecasts will be used to monitor the wholesale trade of motor vehicles and supplies in terms of volume and value. The forecasting models will be developed using the R language.

Step 2: Get data.

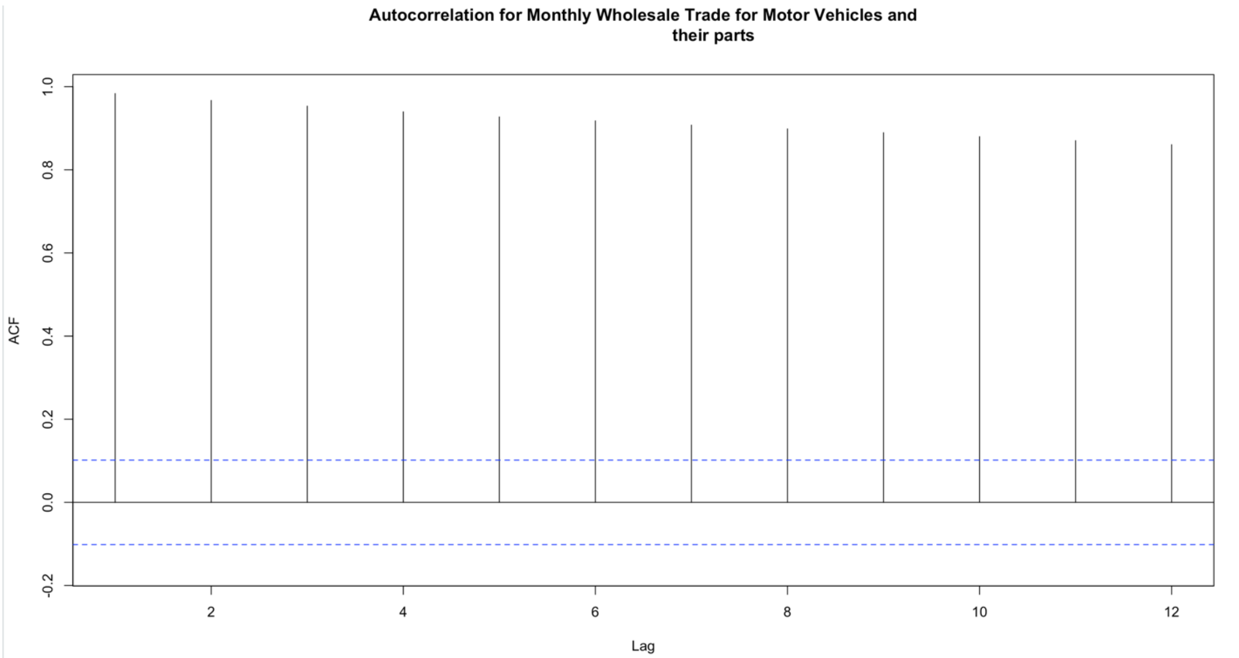
The data used in this project was collected from a small motor vehicle industry over a period of 30 years (1992-2022). The data includes monthly wholesale trade of motor vehicles and parts and supplies.

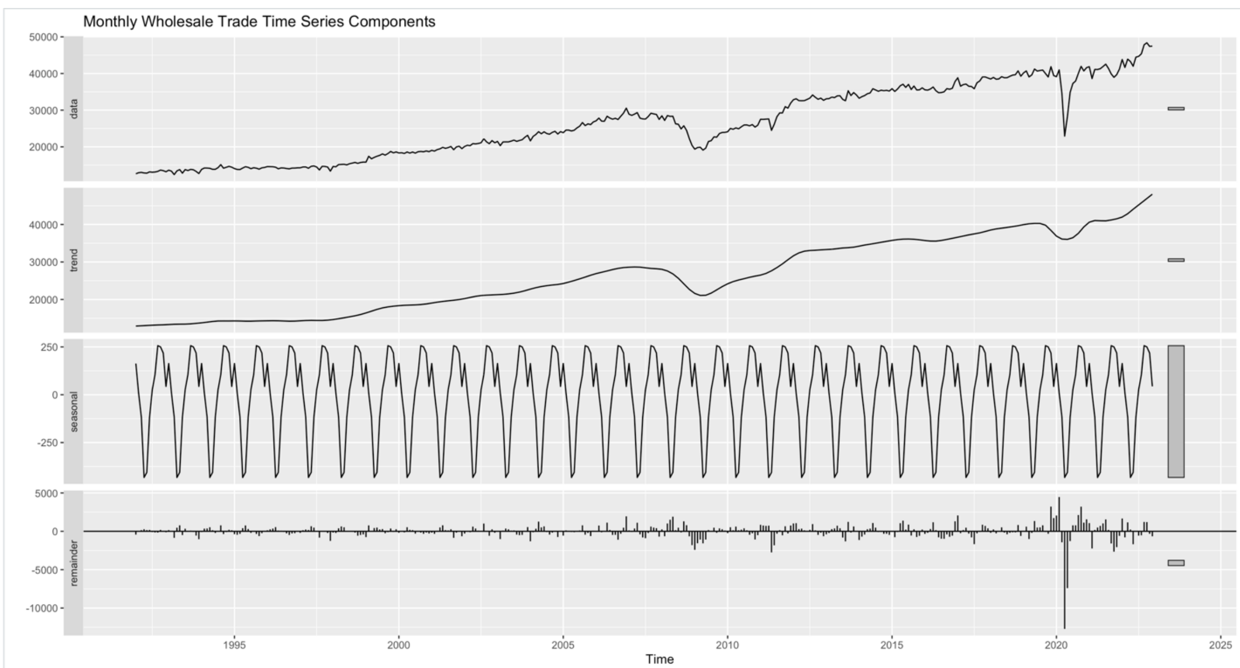
We got the data set from U.S. Census Bureau – Business and Industry Time Series Datasets. Please refer the link provided below:

Link: https://www.census.gov/econ/currentdata/datasets/

Step 3: Explore and Visualize Series

From the Correlogram using acf function, (acf function identifies and plots multiple autocorrelation coefficients for various lags of a time series data) strong positive autocorrelation coefficients are significantly greater for all the lags.





The stl function showed an upward trend and seasonality in the wholesale trade data. The trend component indicates an overall increase in the data over time, while the seasonality component suggests regular patterns that repeat at fixed intervals. Analyzing these components can aid in developing effective forecasting models.

Step 4: Data Preprocessing

The data was clean and there are no potential issues or challenges observed. The data collected is free from outliers, missing values, data entry errors, unequal spacing, irrelevant periods etc. Hence there was no cleaning required as it is already preprocessed.

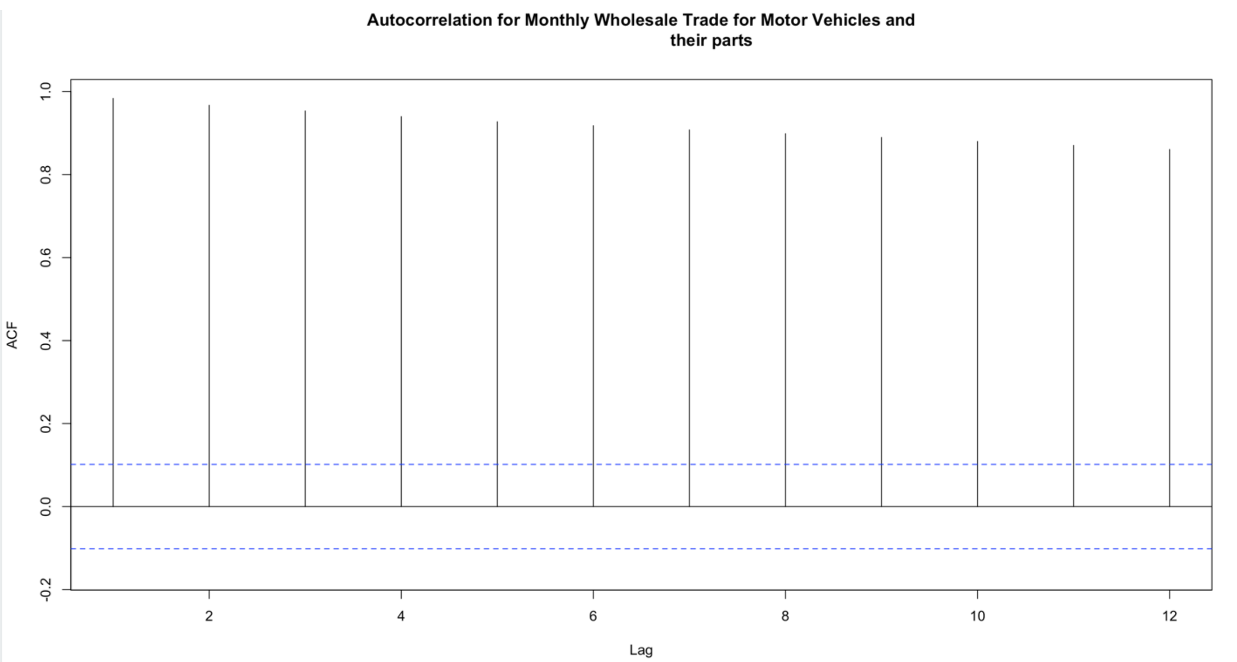
Step 5: Partition Series

We divided our dataset into two subsets: one for training and one for validation. The training set consists of 336 records and covers the period from January 1992 to December 2019. The validation set contains 36 records and covers the period from January 2020 to December 2022.

nTrain	336
nValid	36
trade.ts	Time-Series [1:372] from 1992 to 2023: 12652 12942 13032 12876 12821 13162 13070 13130 13300 13661 ...
train.ts	Time-Series [1:336] from 1992 to 2020: 12652 12942 13032 12876 12821 13162 13070 13130 13300 13661 ...
valid.ts	Time-Series [1:36] from 2020 to 2023: 39128 40984 34562 22919 28219 34862 37279 37942 40014 41953 ...

Step 6 & 7: Apply Forecasting & Comparing Performance

Apply Forecasting and Comparing Performances:



The Acf() function is employed to detect autocorrelation and plot the autocorrelation values for various lags. In this case, all the lag values for the autocorrelation are found to be statistically significant, indicating that the time series data is predictable, and there is no presence of a random walk. Hence, incorporating this data into the forecasting process is recommended to achieve accurate predictions.

Advanced Exponential Smoothing

The method used for analysing time series data is called advanced exponential smoothing, which specifically involves the Holt-Winters model. This model is effective because it takes into account both the trend and seasonal patterns when making predictions. Before applying the model to the entire dataset, it was first assessed using the training and validation sections.

The Holt-Winters Model is employed in an automated fashion, with suitable training and validation portions. The c(Z, Z, Z) parameter in the code is responsible for the automatic selection of error, trend, and seasonality components. As default values for alpha (error) and beta (trend) are not specified in the code, the model will calculate and provide us with optimized values.

Holt-Winter's exponential smoothing for partitioned data:

```
> # Create Holt-Winter's (HW) exponential smoothing for partitioned data.
> # Use ets() function with model = "ZZZ", i.e., automatic selection of
> # error, trend, and seasonality options.
> # Use optimal alpha, beta, & gamma to fit HW over the training period.
> hw.ZZZ <- ets(train.ts, model = "ZZZ")
> hw.ZZZ
ETS(M,A,N)

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

  Smoothing parameters:
    alpha = 0.7634
    beta  = 1e-04

  Initial states:
    l = 12747.4235
    b = 78.6805

  sigma:  0.0293

     AIC     AICc      BIC
6343.470 6343.652 6362.556
>
```

The Holt-Winter's model, with automated parameter selection for training and validation data, effectively fits the data by accurately capturing both the trend and seasonal patterns.
The model demonstrates a strong ability to capture and represent the trend and seasonal components of the data, showcasing a good fit.

The following table represents the values of the smoothers of the Holt-Winters Model.

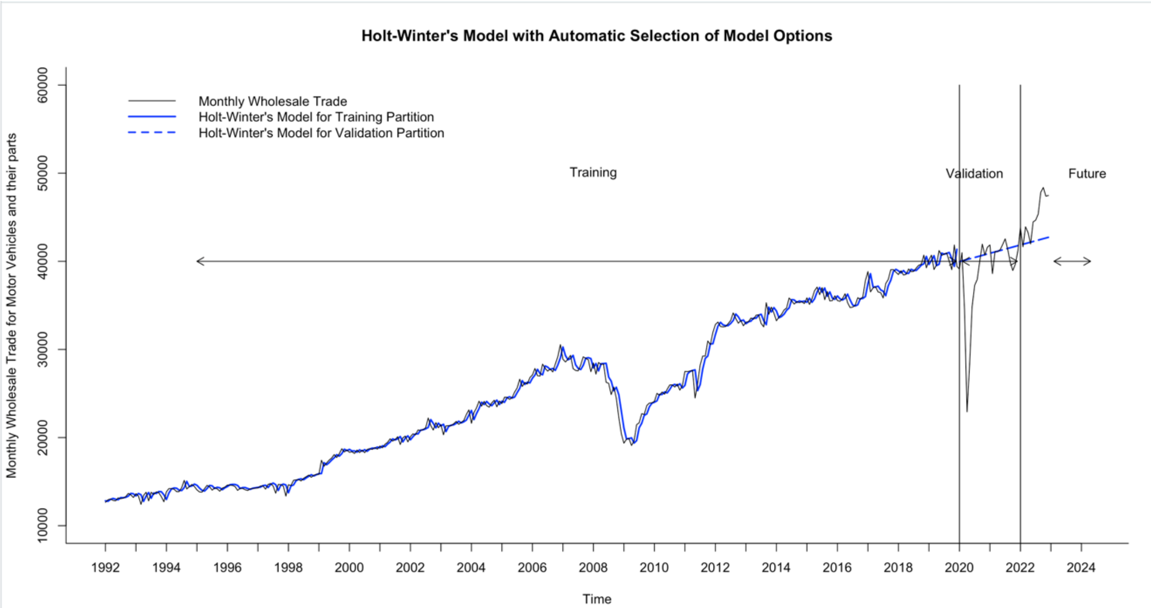
alpha	beta
0.7634	1e-04

The AIC, AICc, and BIC values for the model are 6343.470, 6343.652, and 6362.556, respectively.

The predictions of the validation data are as follows:

```
> hw.ZZZ.pred <- forecast(hw.ZZZ, h = nValid, level = 0)
> hw.ZZZ.pred
         Point Forecast     Lo 0     Hi 0
Jan 2020       39965.37 39965.37 39965.37
Feb 2020       40044.15 40044.15 40044.15
Mar 2020       40122.93 40122.93 40122.93
Apr 2020       40201.71 40201.71 40201.71
May 2020       40280.49 40280.49 40280.49
Jun 2020       40359.27 40359.27 40359.27
Jul 2020       40438.05 40438.05 40438.05
Aug 2020       40516.83 40516.83 40516.83
Sep 2020       40595.61 40595.61 40595.61
Oct 2020       40674.38 40674.38 40674.38
Nov 2020       40753.16 40753.16 40753.16
Dec 2020       40831.94 40831.94 40831.94
Jan 2021       40910.72 40910.72 40910.72
Feb 2021       40989.50 40989.50 40989.50
Mar 2021       41068.28 41068.28 41068.28
Apr 2021       41147.06 41147.06 41147.06
May 2021       41225.84 41225.84 41225.84
Jun 2021       41304.62 41304.62 41304.62
Jul 2021       41383.40 41383.40 41383.40
Aug 2021       41462.18 41462.18 41462.18
Sep 2021       41540.96 41540.96 41540.96
Oct 2021       41619.74 41619.74 41619.74
Nov 2021       41698.52 41698.52 41698.52
Dec 2021       41777.30 41777.30 41777.30
Jan 2022       41856.07 41856.07 41856.07
Feb 2022       41934.85 41934.85 41934.85
Mar 2022       42013.63 42013.63 42013.63
Apr 2022       42092.41 42092.41 42092.41
May 2022       42171.19 42171.19 42171.19
Jun 2022       42249.97 42249.97 42249.97
Jul 2022       42328.75 42328.75 42328.75
Aug 2022       42407.53 42407.53 42407.53
Sep 2022       42486.31 42486.31 42486.31
Oct 2022       42565.09 42565.09 42565.09
Nov 2022       42643.87 42643.87 42643.87
Dec 2022       42722.65 42722.65 42722.65
> 
```

The plot of the Holt-Winter's model with automatic parameter selection for the partitioned data set showcases a robust fit, accurately capturing the trend patterns.



Holt-Winter’s exponential smoothing for entire data:

```
> HW.ZZZ <- ets(trade.ts, model = "ZZZ")
> HW.ZZZ
ETS(M,A,N)

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

  Smoothing parameters:
    alpha = 0.7851
    beta  = 1e-04

  Initial states:
    l = 12745.9063
    b = 78.3624

  sigma:  0.0393

     AIC     AICc      BIC
7319.157 7319.321 7338.752
>
```

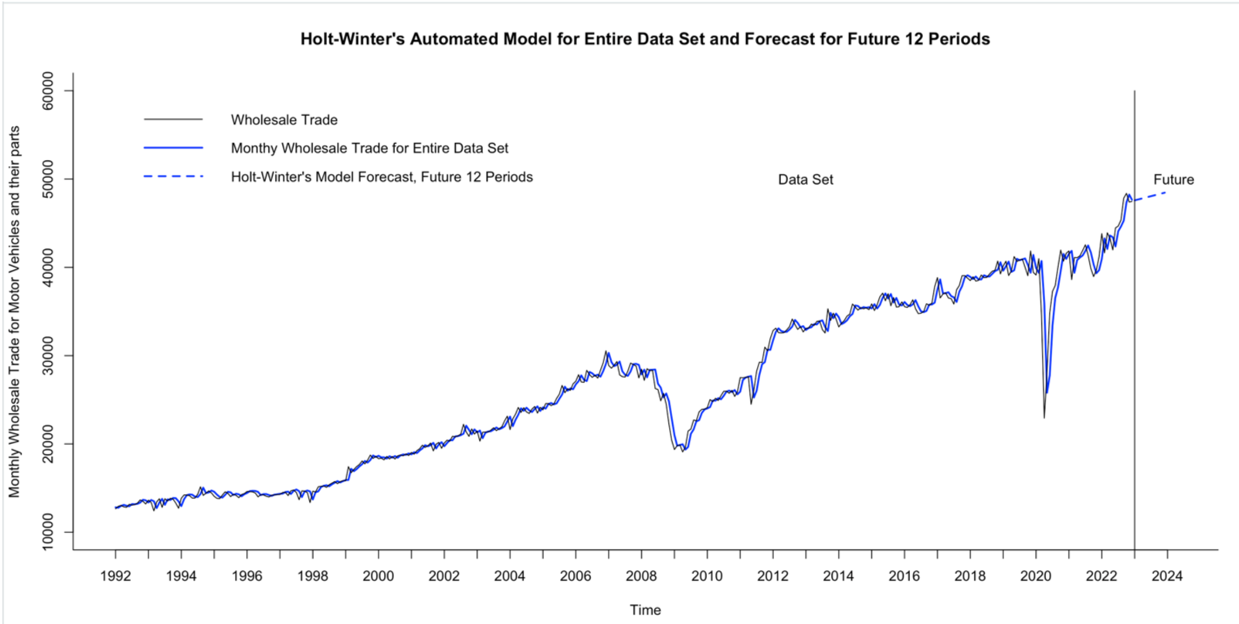
The ETS(M,A,N) model with automated parameter selection effectively fits the Monthly Wholesale Trade Forecasting for Motor Vehicle and Motor Vehicle Parts and Supplies data by accurately capturing both the trend and seasonal patterns. The model's smoothing parameters are alpha = 0.7851 and beta = 1e-04, and its initial states are l = 12745.9063 and b = 78.3624. The estimated sigma value for the model is 0. 0393.The AIC, AICc, and BIC values for the model are 7319.157, 7319.321, and 7338.752, respectively.

Overall, the ETS(M,A,N) model demonstrates a strong ability to capture and represent the trend and seasonal components of the data, showcasing a good fit. The automated parameter selection process helps to ensure that the model is optimized for training and validation data, leading to more accurate predictions.

The predictions of the data are as follows:

```
> # Use forecast() function to make predictions using this HW model for
> # 12 month into the future.
> HW.ZZZ.pred <- forecast(HW.ZZZ, h = 12 , level = 0)
> HW.ZZZ.pred
         Point Forecast     Lo 0     Hi 0
Jan 2023       47587.85 47587.85 47587.85
Feb 2023       47666.94 47666.94 47666.94
Mar 2023       47746.02 47746.02 47746.02
Apr 2023       47825.10 47825.10 47825.10
May 2023       47904.19 47904.19 47904.19
Jun 2023       47983.27 47983.27 47983.27
Jul 2023       48062.35 48062.35 48062.35
Aug 2023       48141.44 48141.44 48141.44
Sep 2023       48220.52 48220.52 48220.52
Oct 2023       48299.60 48299.60 48299.60
Nov 2023       48378.69 48378.69 48378.69
Dec 2023       48457.77 48457.77 48457.77
> 
```

The plot of the Holt-Winter's model with Automatic selection of parameters for Entire data set shows that the model is fitting well into data, capturing the trend components.



Autoregressive Integrated Moving Average Models

The Autoregressive Integrated Moving Average (ARIMA) model is a flexible model that can be used for forecasting data with level, trend, and seasonal components. Since our data consists of all three, this model is appropriate to use for analysis. We generated an optimal ARIMA model with automatic selection of (p,d,q)(P,D,Q) parameters using the auto.arima() function.

auto.arima() function in R is used to automatically identify ARIMA model and its respective *(p, d, q)(P, D, Q)* parameters.

Automated ARIMA for Training & Validation Data:

```
> # Use auto.arima() function to fit ARIMA model.
> # Use summary() to show auto ARIMA model and its parameters.
> train.auto.arima <- auto.arima(train.ts)
> summary(train.auto.arima)
Series: train.ts
ARIMA(0,1,1) with drift

Coefficients:
          ma1    drift
      -0.2275  81.1323
s.e.   0.0528  30.4965

sigma^2 = 524272:  log likelihood = -2680.3
AIC=5366.61   AICc=5366.68   BIC=5378.05

Training set error measures:
                    ME     RMSE      MAE         MPE     MAPE      MASE          ACF1
Training set 0.1769038 720.8266 507.7581 -0.09162563 2.112059 0.2978016 -0.001059336
>
```

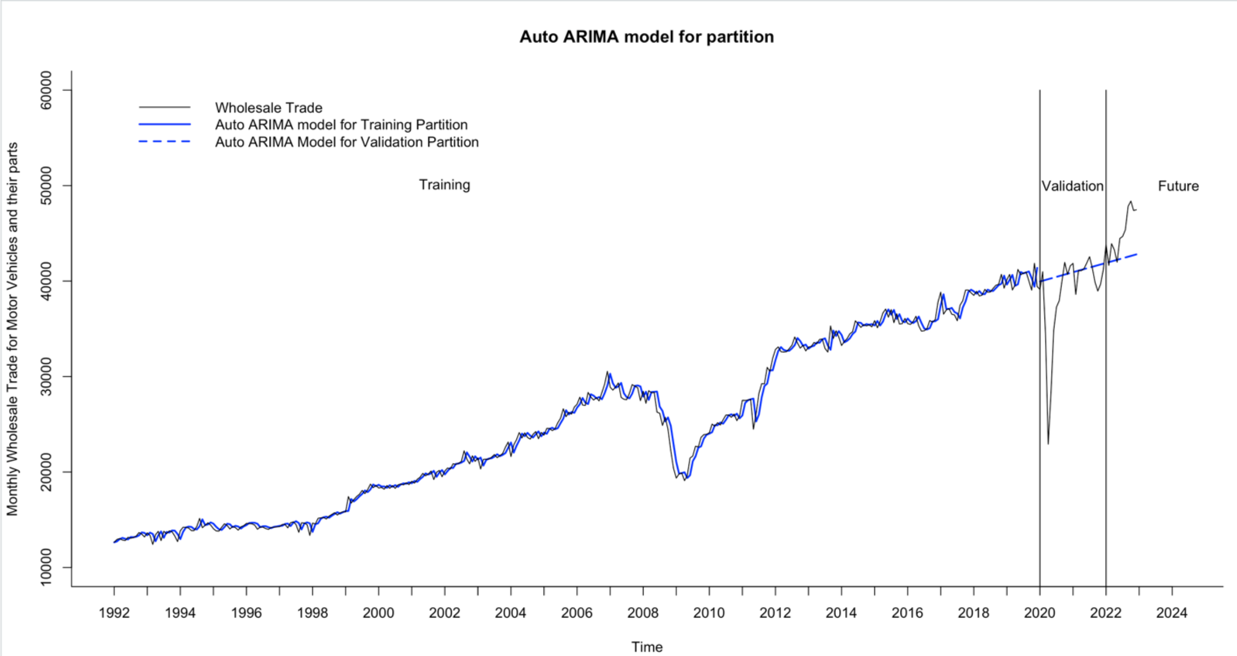
The model has two estimated coefficients: a moving average coefficient (ma1) of -0.2275 and a drift coefficient of 81.1323. The variance of the error term (sigma^2) is 524272, and the log likelihood is -2680.3. The model selection criteria AIC, AICc, and BIC are 5366.61, 5366.68, and 5378.05, respectively.

MODEL EQUATION: $y_t - y_{t-1} = 81.1323 - 0.2275\ e_{t-1}$

The forecast of the Auto Arima model on the validation period:

```
> train.auto.arima.pred <- forecast(train.auto.arima, h = nValid, level = 0)
> train.auto.arima.pred
         Point Forecast     Lo 0     Hi 0
Jan 2020       39955.07 39955.07 39955.07
Feb 2020       40036.21 40036.21 40036.21
Mar 2020       40117.34 40117.34 40117.34
Apr 2020       40198.47 40198.47 40198.47
May 2020       40279.60 40279.60 40279.60
Jun 2020       40360.74 40360.74 40360.74
Jul 2020       40441.87 40441.87 40441.87
Aug 2020       40523.00 40523.00 40523.00
Sep 2020       40604.13 40604.13 40604.13
Oct 2020       40685.27 40685.27 40685.27
Nov 2020       40766.40 40766.40 40766.40
Dec 2020       40847.53 40847.53 40847.53
Jan 2021       40928.66 40928.66 40928.66
Feb 2021       41009.79 41009.79 41009.79
Mar 2021       41090.93 41090.93 41090.93
Apr 2021       41172.06 41172.06 41172.06
May 2021       41253.19 41253.19 41253.19
Jun 2021       41334.32 41334.32 41334.32
Jul 2021       41415.46 41415.46 41415.46
Aug 2021       41496.59 41496.59 41496.59
Sep 2021       41577.72 41577.72 41577.72
Oct 2021       41658.85 41658.85 41658.85
Nov 2021       41739.99 41739.99 41739.99
Dec 2021       41821.12 41821.12 41821.12
Jan 2022       41902.25 41902.25 41902.25
Feb 2022       41983.38 41983.38 41983.38
Mar 2022       42064.51 42064.51 42064.51
Apr 2022       42145.65 42145.65 42145.65
May 2022       42226.78 42226.78 42226.78
Jun 2022       42307.91 42307.91 42307.91
Jul 2022       42389.04 42389.04 42389.04
Aug 2022       42470.18 42470.18 42470.18
Sep 2022       42551.31 42551.31 42551.31
Oct 2022       42632.44 42632.44 42632.44
Nov 2022       42713.57 42713.57 42713.57
Dec 2022       42794.70 42794.70 42794.70
> 
```

The plot of the Auto ARIMA for training and validation data sets shows that the model is fitting well into data, capturing the trend components.



Automated ARIMA for Entire Data Set

```
> # Use auto.arima() function to fit ARIMA model for entire data set.
> # use summary() to show auto ARIMA model and its parameters for entire data set.
> auto.arima <- auto.arima(trade.ts)
> summary(auto.arima)
Series: trade.ts
ARIMA(0,1,0) with drift

Coefficients:
        drift
      93.8464
s.e.  60.8481

sigma^2 = 1377337:  log likelihood = -3148.09
AIC=6300.18   AICc=6300.21   BIC=6308.01

Training set error measures:
                     ME     RMSE      MAE        MPE     MAPE      MASE        ACF1
Training set 0.03375846 1170.441 666.6576 -0.1434417 2.539263 0.3331252 0.007150448
> 
```

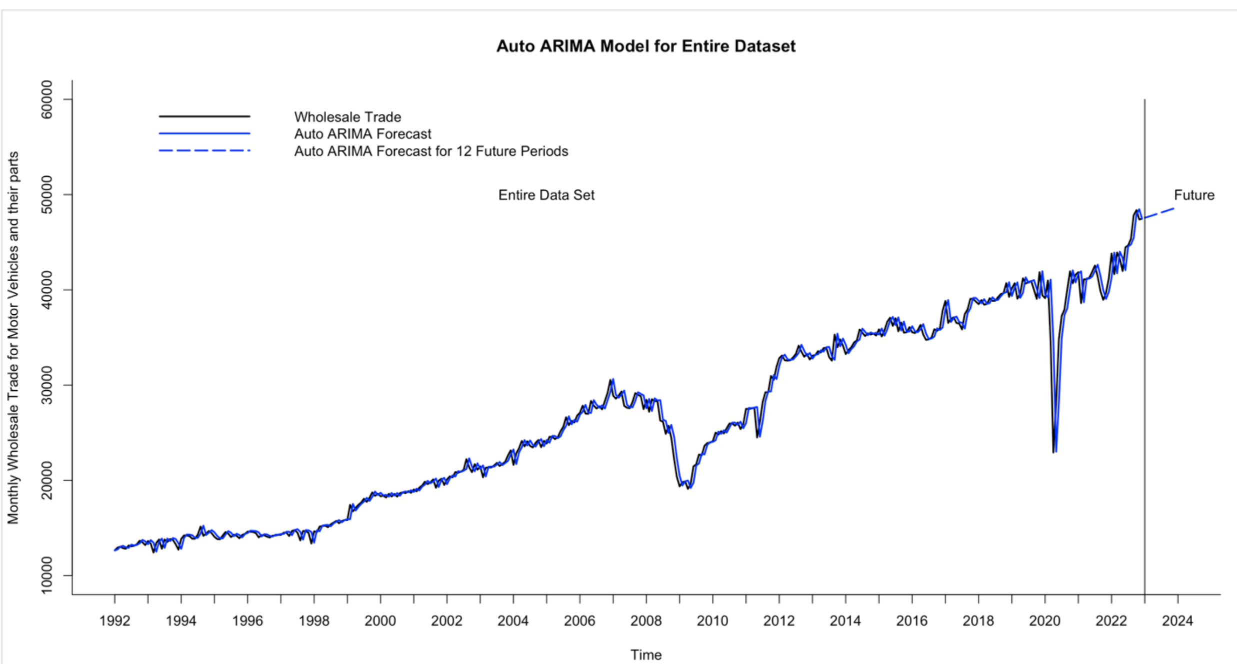
The estimated coefficient for the drift term is 93.8464, and the standard error of this coefficient is 60.8481.

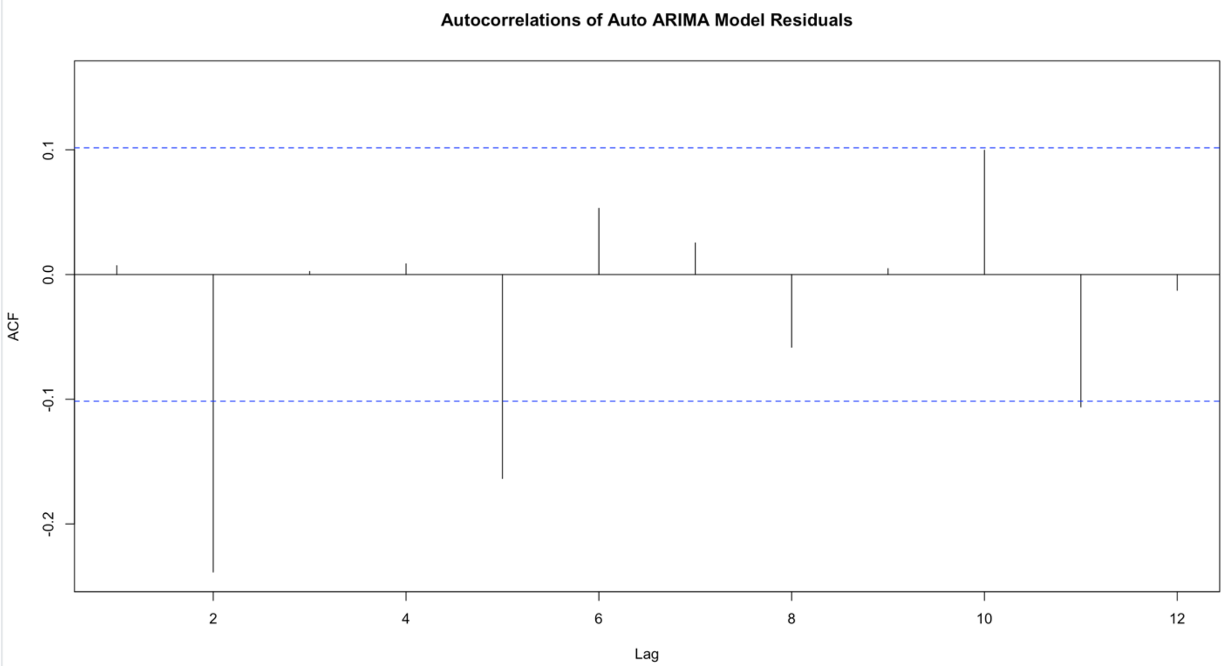
The variance of the error term (sigma^2) is 1377337, and the log likelihood is -3148.09. The model selection criteria AIC, AICc, and BIC are 6300.18, 6300.21, and 6308.01, respectively.

The forecast of the Auto Arima model on the future period:

```
> auto.arima.pred <- forecast(auto.arima, h = 12, level = 0)
> auto.arima.pred
         Point Forecast     Lo 0     Hi 0
Jan 2023       47562.85 47562.85 47562.85
Feb 2023       47656.69 47656.69 47656.69
Mar 2023       47750.54 47750.54 47750.54
Apr 2023       47844.39 47844.39 47844.39
May 2023       47938.23 47938.23 47938.23
Jun 2023       48032.08 48032.08 48032.08
Jul 2023       48125.92 48125.92 48125.92
Aug 2023       48219.77 48219.77 48219.77
Sep 2023       48313.62 48313.62 48313.62
Oct 2023       48407.46 48407.46 48407.46
Nov 2023       48501.31 48501.31 48501.31
Dec 2023       48595.16 48595.16 48595.16
> 
```

The plot of the Auto ARIMA for entire data sets shows that the model is fitting well into data, capturing the trend components.





The x-axis shows the time lag from 0 to 12 and y-axis shoes correlation coefficients.
All lags except lag 2, lag 5 and lag 11 are insignificant since they lie in between the blue dotted lines. The lag 2, lag 5 and lag 11 are statistically significant with negative correlation. Lag 2 has strong negative correlation compared to other lags.

Regression Based Models

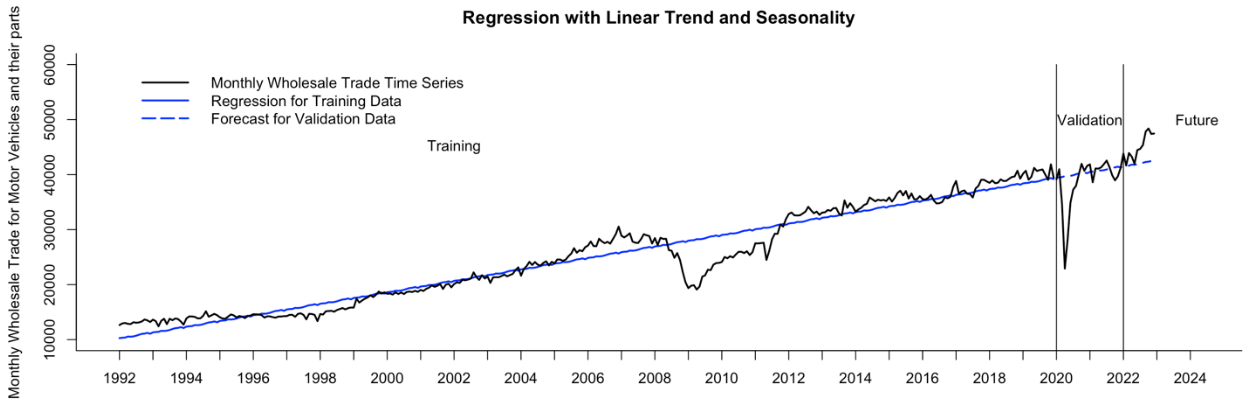
Regression-based models were employed for time series analysis, with the specific model type chosen based on the characteristics observed in the time series plot. This approach was favored due to its simplicity, effectiveness in considering both trend and seasonality, and the potential for enhancing the model with autoregressive components. Before applying the model to the entire data set, it underwent evaluation using separate training and validation partitions.

Regression Model with Linear Trend and Seasonality:

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

Residuals:
    Min      1Q  Median      3Q     Max 
-9171.6  -742.7   350.2  1473.3  4894.7 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 10188.740    468.479  21.749   <2e-16 ***
trend          86.824      1.255  69.177   <2e-16 ***
season2       -19.146    596.019  -0.032    0.974    
season3       -75.149    596.023  -0.126    0.900    
season4        24.348    596.029   0.041    0.967    
season5       -87.047    596.039  -0.146    0.884    
season6      -117.765    596.051  -0.198    0.844    
season7       -79.732    596.065  -0.134    0.894    
season8         6.765    596.082   0.011    0.991    
season9        72.334    596.102   0.121    0.903    
season10       37.010    596.125   0.062    0.951    
season11       84.042    596.150   0.141    0.888    
season12     -158.818    596.177  -0.266    0.790    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2230 on 323 degrees of freedom
Multiple R-squared:  0.9369,	Adjusted R-squared:  0.9345 
F-statistic: 399.3 on 12 and 323 DF,  p-value: < 2.2e-16
```

The model above represents a regression model with linear trend and seasonality for training and partition data sets. The model is statistically significant since the F-Statistic p-value is very low (<2.2e16), much lower than an alpha of 5%. The R-Square of the model is 93.69%. The adjusted R-square of the model is 93.45%. This model consists of 1 trend predictor and 11 seasonal predictors which are dummy variables. These predictors are all significant since their p-values are lower than an alpha of 5%. This indicates that the model is a good fit for the data and is statistically significant. Therefore, it may be applied for forecasting.

Model Equation: = $y_t = 10188.740 + 86.824\,t - 19.146\,D_2 - 75.149\,D_3 + \ldots\ldots\ldots - 158.818 D_{12} + e$



From the plot we can see that the respective models fit well. They are taking trend into consideration.

Forecasting Entire Dataset

Forecasting on the Entire Dataset In order to create forecasts, a model must be constructed using the entire dataset. Regression with linear trend and seasonality is utilized along with variations of the models. The variations include the use of an autoregressive model. The use of these multilevel models will, in theory, result in more accurate forecasts.

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

Residuals:
     Min       1Q   Median       3Q      Max 
-16336.9   -815.1    376.5   1505.3   5846.1 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 10286.939    499.957  20.576   <2e-16 ***
trend          86.876      1.209  71.856   <2e-16 ***
season2      -140.682    635.720  -0.221    0.825    
season3      -252.719    635.724  -0.398    0.691    
season4      -568.756    635.729  -0.895    0.372    
season5      -541.697    635.738  -0.852    0.395    
season6      -264.056    635.748  -0.415    0.678    
season7      -132.287    635.761  -0.208    0.835    
season8       -57.098    635.775  -0.090    0.928    
season9        91.671    635.793   0.144    0.885    
season10      101.925    635.812   0.160    0.873    
season11       86.855    635.834   0.137    0.891    
season12      -60.375    635.858  -0.095    0.924    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2503 on 359 degrees of freedom
Multiple R-squared:  0.9351,	Adjusted R-squared:  0.933 
F-statistic: 431.3 on 12 and 359 DF,  p-value: < 2.2e-16
```

The model above represents a regression model with linear trend and seasonality for the entire data set. The model is statistically significant since the F-Statistic p-value is very low (<2.2e16), much lower than an alpha of 5%. The R-Square of the model is 93.51%. The adjusted R-square of the model is 93.3%. This model consists of 1 trend predictor and 11 seasonal predictors which are dummy variables. These predictors are all significant since their p-values are lower than an alpha of 5%.

Model Equation: = $y_t = 10286.939 + 86.876\,t - 140.682\,D_2 - 252.719\,D_3 + \ldots\ldots\ldots - 60.375 D_{12} + e$

Autoregressive Models for Regression Residuals

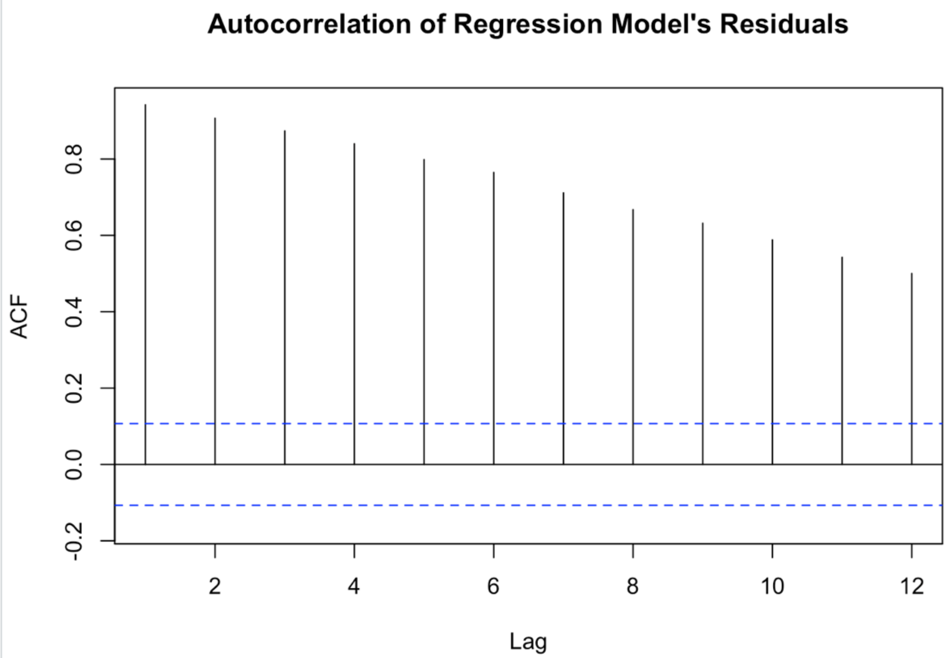
Prior to using an autoregressive model, a correlogram must be created which shows the autocorrelation between the residuals of the respective regression models.

```
Series: train.lin.season$residuals
ARIMA(1,0,0) with non-zero mean

Coefficients:
         ar1      mean
      0.9426  100.6479
s.e.  0.0176  656.5471

sigma^2 = 526562:  log likelihood = -2690.11
AIC=5386.22   AICc=5386.29   BIC=5397.67

Training set error measures:
                    ME     RMSE      MAE       MPE     MAPE      MASE
Training set -9.938501 723.4834 517.8379 -59.67117 152.2646 0.3752399
                    ACF1
Training set -0.1744465
```

The Model Equation is $y_t = a + b_1 y_{t-1} + y_t$

We get the following model equation $y_t = 100.649 + 0.9426\, y_{t-1}$

The coefficient of the *ar1* (Y_{t-1}) variable, β1 = 0.9426, and standard error of estimate, s.e. = 0.0176. We will use these two parameters for hypothesis testing about the value of the AR(1) regression coefficient.



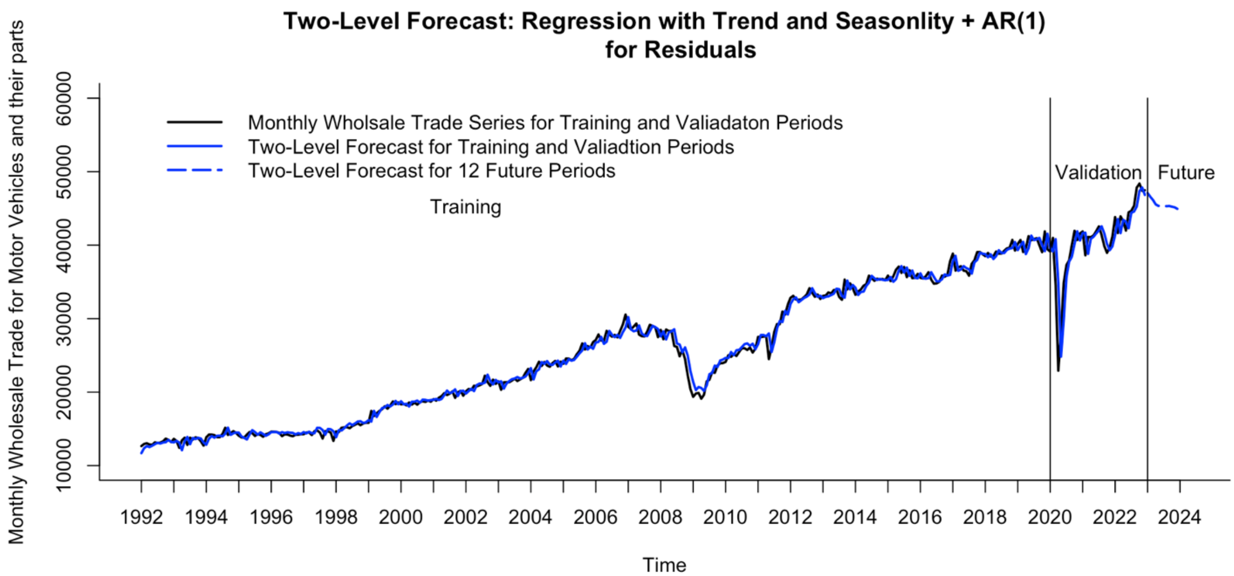
Most of the lags for Autocorrelation for Regression Model's residuals are statistically significant with strong positive correlation in decreasing pattern. and hence the data set is predictable, it is a good idea to add it to our forecast an AR model for residuals.

Two-level model for linear trend and seasonality model and AR(1) model for residuals

The table describes the wholesale trade data and forecasts in the validation partition (Valid. regression model's forecast in the validation period (Reg.Forecast), AR(1) model's forecast of the regression residuals in the validation period (AR(1)Forecast), and combined forecast (Combined.Forecast) as a sum of the regression and AR(1) models' forecasts.

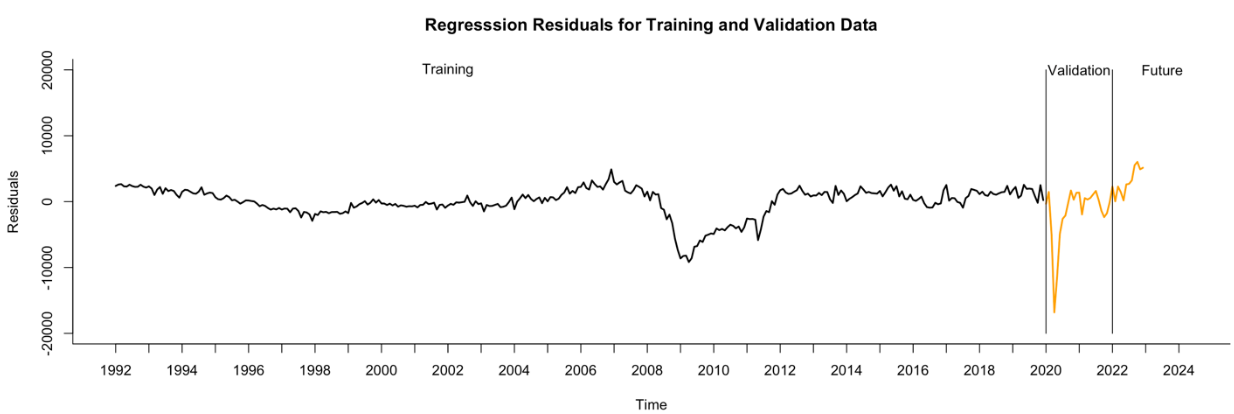
	Monthly Wholesale Trade	Reg.Forecast	AR(1)Forecast	Combined.Forecast
1	39128	39448.54	218.896	39667.44
2	40984	39516.22	212.106	39728.33
3	34562	39547.04	205.706	39752.75
4	22919	39733.36	199.673	39933.04
5	28219	39708.79	193.987	39902.78
6	34862	39764.90	188.628	39953.53
7	37279	39889.76	183.576	40073.33
8	37942	40063.08	178.814	40241.89
9	40014	40215.47	174.325	40389.80
10	41953	40266.97	170.095	40437.07
11	40697	40400.83	166.107	40566.93
12	41593	40244.79	162.348	40407.14
13	41859	40490.43	158.805	40649.24
14	38610	40558.11	155.466	40713.58
15	41117	40588.93	152.318	40741.25
16	41092	40775.25	149.351	40924.61
17	41303	40750.68	146.554	40897.24
18	41899	40806.79	143.918	40950.71
19	42553	40931.65	141.434	41073.08
20	41374	41104.97	139.092	41244.06
21	39874	41257.36	136.884	41394.25
22	38954	41308.86	134.803	41443.67
23	39674	41442.72	132.842	41575.56
24	41200	41286.68	130.993	41417.68
25	43817	41532.33	129.251	41661.58
26	41647	41600.00	127.608	41727.61
27	43919	41630.83	126.060	41756.89
28	43266	41817.15	124.601	41941.75
29	41975	41792.58	123.226	41915.80
30	44465	41848.68	121.929	41970.61
31	44676	41973.54	120.707	42094.25
32	45363	42146.86	119.555	42266.42
33	47829	42299.25	118.470	42417.72
34	48379	42350.75	117.446	42468.20
35	47393	42484.61	116.482	42601.09
36	47469	42328.58	115.572	42444.15

The plot of the Two-level model for linear trend and seasonality model and AR(1) model for residuals for training and validation data sets shows that the model is fitting well into data, capturing the trend components.

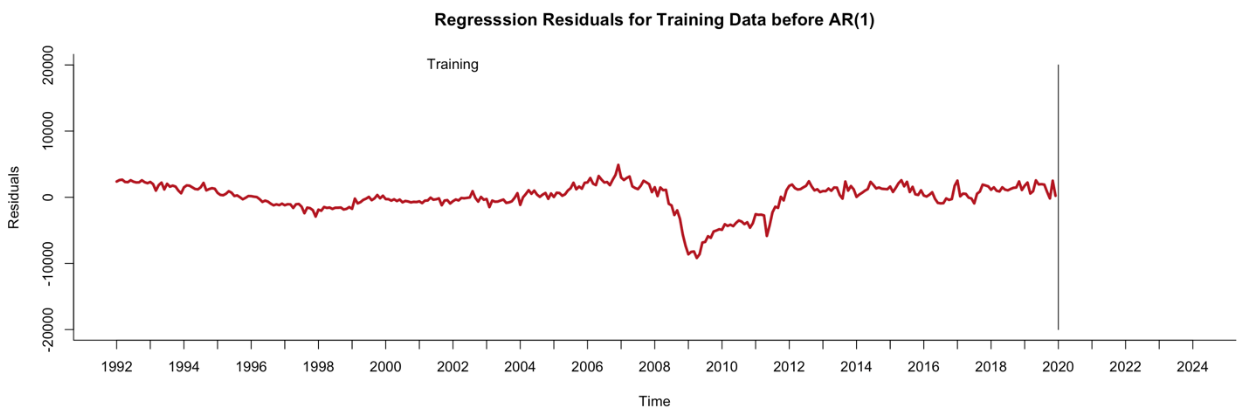


Regression Residuals Plots

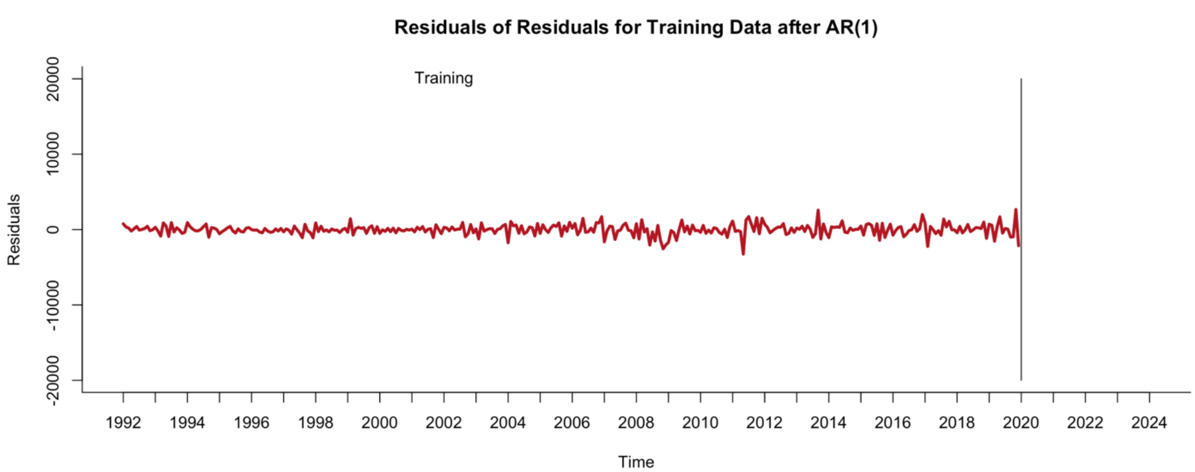
For Training and Validation Data:



For Training Data before AR(1) :



For Training Data After AR(1):



Step 8: Implement Forecast

Models	RMSE	MAPE
Holt-Winter's Model with Automatic Selection of Parameters	720.943	2.114
Auto ARIMA	1170.441	2.539
Two-level model (linear trend and seasonality model + AR (1) model for residuals)	753.358	2.335
Seasonal Naive	1175.768	2.563

```
> round(accuracy(hw.ZZZ.pred$fitted, trade.ts), 3)
             ME     RMSE      MAE     MPE  MAPE  ACF1 Theil's U
Test set 2.941 720.943 508.089 -0.082 2.114 0.008     0.967
> round(accuracy(auto.arima.pred$fitted, trade.ts), 3)
             ME      RMSE      MAE     MPE  MAPE  ACF1 Theil's U
Test set 0.034 1170.441 666.658 -0.143 2.539 0.007     0.993
> round(accuracy(train.lin.season$fitted + residual.ar1$fitted, trade.ts), 3)
              ME     RMSE      MAE     MPE  MAPE   ACF1 Theil's U
Test set -30.92 753.358 550.173 -0.261 2.335 -0.048     1.024
> round(accuracy((naive(trade.ts))$fitted, trade.ts), 3)
              ME      RMSE      MAE     MPE  MAPE  ACF1 Theil's U
Test set 93.846 1175.768 676.412 0.268 2.563 0.007         1
>
```

From the above table, MAPE and RMSE values of Holt-Winter's Model with Automatic Selection of Parameters are low when compared with other models. Hence, we can say that Holt-Winters mode is the best fit for the data.

Conclusion

At the end of this Time series Project to Predict the monthly wholesale trade for motor vehicles and their parts and supplies, we found out that Holt-Winter's Model with Automatic Selection of Parameters is the best model for the dataset.

Overall, our findings suggest that the Holt-Winters model with automatic selection of parameters is a statistically significant and useful tool for forecasting the monthly wholesale trade for motor vehicles and their parts and supplies.

Appendix

Figure 1: Training Partition

```
> train.ts
       Jan   Feb   Mar   Apr   May   Jun   Jul   Aug   Sep   Oct   Nov   Dec
1992 12652 12942 13032 12876 12821 13162 13070 13130 13300 13661 13504 13202
1993 13633 13368 12421 13436 13793 12817 13796 13534 13842 13749 13293 12716
1994 13851 14220 14209 14142 13870 13871 14316 15148 14170 14421 14684 14450
1995 14070 13834 13808 14214 14578 14408 14038 14299 14175 13915 14268 14383
1996 14616 14599 14575 14446 14009 14251 14224 14082 13996 14197 14207 14271
1997 14279 14506 14511 14156 14690 14786 14501 13689 14685 14659 14493 13362
1998 14654 14559 15151 15204 15235 15082 15386 15563 15741 15499 15728 15839
1999 15825 17431 16742 17117 17436 17675 18073 17740 18167 18735 18366 18601
2000 18332 18388 18202 18587 18300 18577 18291 18661 18743 18658 18869 18691
2001 19038 18841 19243 19455 19858 19603 19788 20101 19214 19974 20173 19509
2002 20104 20426 20311 20873 20800 20917 21109 22215 21364 20858 21701 21115
2003 21472 20319 21332 21352 21349 21564 21830 21506 21730 21943 22612 23147
2004 21623 22839 23372 24135 23568 24103 23620 23454 23948 24240 23489 24150
2005 23862 24571 24561 24335 24531 25162 25637 26628 25796 26260 26071 26804
2006 27103 27842 27015 26982 28337 27854 27539 27781 27453 28331 29196 30553
2007 28890 28580 28916 29333 27822 27618 27577 28249 29162 28980 28830 27477
2008 28460 27197 28520 28304 28333 26276 26157 24880 25735 24488 22265 20399
2009 19366 19804 19892 19101 19645 21462 21683 22712 22628 23632 23920 23950
2010 24117 25020 24797 25169 24924 25471 25964 25972 25735 26070 25382 25909
2011 27511 27491 27544 27608 24492 26135 28232 29256 29237 30963 30546 31928
2012 32837 33108 32600 32569 32603 32925 33285 34147 33513 32975 33279 32687
2013 33095 33150 33564 33426 33888 33927 32940 32565 35304 33981 34801 34141
2014 33252 33683 33977 34478 34696 35842 35505 35149 35436 35309 35403 35210
2015 35858 35105 35765 36658 37075 36215 37019 35645 36578 35503 35544 36087
2016 35547 35461 35766 36320 35307 34754 34801 35008 35870 35700 35961 37810
2017 38842 36534 36973 37116 36528 36452 35841 37469 37949 39055 39061 38767
2018 38503 38931 38443 38525 39129 38836 38858 39239 39559 39664 40709 39259
2019 40159 40697 39067 39609 41221 40664 40825 40936 39995 39047 41860 39429
```

Figure 2: Validation Partition

```
> valid.ts
       Jan   Feb   Mar   Apr   May   Jun   Jul   Aug   Sep   Oct   Nov   Dec
2020 39128 40984 34562 22919 28219 34862 37279 37942 40014 41953 40697 41593
2021 41859 38610 41117 41092 41303 41899 42553 41374 39874 38954 39674 41200
2022 43817 41647 43919 43266 41975 44465 44676 45363 47829 48379 47393 47469
>
```

Figure 3: HW ZZZ Training/Validation

```
> hw.ZZZ <- ets(train.ts, model = "ZZZ")
> hw.ZZZ
ETS(M,A,N)

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

  Smoothing parameters:
    alpha = 0.7634
    beta  = 1e-04

  Initial states:
    l = 12747.4235
    b = 78.6805

  sigma:  0.0293

     AIC     AICc      BIC
6343.470 6343.652 6362.556
>
```

Figure 4: Auto ARIMA

```
> train.auto.arima <- auto.arima(train.ts)
> summary(train.auto.arima)
Series: train.ts
ARIMA(0,1,1) with drift

Coefficients:
          ma1    drift
      -0.2275  81.1323
s.e.   0.0528  30.4965

sigma^2 = 524272:  log likelihood = -2680.3
AIC=5366.61   AICc=5366.68   BIC=5378.05

Training set error measures:
                    ME     RMSE      MAE         MPE     MAPE      MASE          ACF1
Training set 0.1769038 720.8266 507.7581 -0.09162563 2.112059 0.2978016 -0.001059336
>
```

Figure 5: Regression model with Linear Trend and Seasonality

```
> summary(train.lin.season)

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

Residuals:
    Min      1Q  Median      3Q     Max 
-9171.6  -742.7   350.2  1473.3  4894.7 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 10188.740    468.479  21.749   <2e-16 ***
trend          86.824      1.255  69.177   <2e-16 ***
season2       -19.146    596.019  -0.032    0.974    
season3       -75.149    596.023  -0.126    0.900    
season4        24.348    596.029   0.041    0.967    
season5       -87.047    596.039  -0.146    0.884    
season6      -117.765    596.051  -0.198    0.844    
season7       -79.732    596.065  -0.134    0.894    
season8         6.765    596.082   0.011    0.991    
season9        72.334    596.102   0.121    0.903    
season10       37.010    596.125   0.062    0.951    
season11       84.042    596.150   0.141    0.888    
season12     -158.818    596.177  -0.266    0.790    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2230 on 323 degrees of freedom
Multiple R-squared:  0.9369,	Adjusted R-squared:  0.9345 
F-statistic: 399.3 on 12 and 323 DF,  p-value: < 2.2e-16

> 
```

Figure 6: AR(1)

```
> summary(res.ar1)
Series: train.lin.season$residuals 
ARIMA(1,0,0) with non-zero mean 

Coefficients:
         ar1      mean
      0.9426  100.6479
s.e.  0.0176  656.5471

sigma^2 = 526562:  log likelihood = -2690.11
AIC=5386.22   AICc=5386.29   BIC=5397.67

Training set error measures:
                    ME     RMSE      MAE       MPE     MAPE      MASE       ACF1
Training set -9.938501 723.4834 517.8379 -59.67117 152.2646 0.3752399 -0.1744465
```

Figure 7: Two-level Forecasting (Combining AR(1) and Regression model with Linear Trend and Seasonality

```
> summary(lin.season)

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

Residuals:
     Min       1Q   Median       3Q      Max 
-16336.9   -815.1    376.5   1505.3   5846.1 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 10286.939    499.957  20.576   <2e-16 ***
trend          86.876      1.209  71.856   <2e-16 ***
season2      -140.682    635.720  -0.221    0.825    
season3      -252.719    635.724  -0.398    0.691    
season4      -568.756    635.729  -0.895    0.372    
season5      -541.697    635.738  -0.852    0.395    
season6      -264.056    635.748  -0.415    0.678    
season7      -132.287    635.761  -0.208    0.835    
season8       -57.098    635.775  -0.090    0.928    
season9        91.671    635.793   0.144    0.885    
season10      101.925    635.812   0.160    0.873    
season11       86.855    635.834   0.137    0.891    
season12      -60.375    635.858  -0.095    0.924    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2503 on 359 degrees of freedom
Multiple R-squared:  0.9351,	Adjusted R-squared:  0.933 
F-statistic: 431.3 on 12 and 359 DF,  p-value: < 2.2e-16

> 
```

Team Members:

Name	Net id	Email id
Sahithi Priya Kandanelly	it7005	skandanelly@horizon.csueastbay.edu
Abhishek Malla	vb7520	amalla3@horizon.csueastbay.edu
Amruth Sai Reddy Bandi	ge5649	abandi2@horizon.csueastbay.edu
Venkatesh Vuyyala	kx8516	vvuyyala@horizon.csueastbay.edu
Kousalya Kunamneni	qi7338	kkunamneni@horizon.csueastbay.edu