2242-ECON-5337-001

Business & Economic Forecasting

Spring 2024

Professor: Dr. Christopher Candreva

PROJECT REPORT

TOPIC:

UBER DATA ANALYSIS

Group:

- Yuvarekha Mahendran- 1002175347
- Sushmitha Ugle Ashok- 1002163441

Introduction:

With over 118 million users, 5 million drivers, and 6.3 billion trips with 17.4 million trips completed per day - Uber is the company behind the data for moving people and making deliveries hassle-free. How are drivers assigned to riders' cost-efficiently, and how is dynamic pricing leveraged to balance supply and demand? Thanks to the large volumes of data Uber collects and the fantastic team that handles Uber Data Analysis using R tools and frameworks. If you're curious to learn more about how data analysis is done at Uber to ensure positive experiences for riders while making the ride profitable for the company - Get your hands dirty working with the Uber dataset to gain in-depth insights.

Sometimes it's easy to give up on someone else's driving. This is less stress; more mental space and one uses that time to do other things. Yes, that's one of the ideas that grew and later became the idea behind **Uber and Lyft**.

Both companies offer passenger boarding services that allow users to rent cars with drivers through websites or mobile apps. Whether traveling a short distance or traveling from one city to another, these services have helped people in many ways and have made their lives very difficult. **Uber** is an international company located in 69 countries and around 900 cities around the world. Lyft, on the other hand, operates in approximately 644 cities in the US and 12 cities in Canada alone. However, in the US, it is the second-largest passenger company with a market share of 31%. From booking a taxi to paying a bill, both services have similar features. But there are some exceptions when the two passenger services reach the neck. The same goes for prices, especially **Uber's** "**surge**" and "Prime Time" in uber. There are certain limitations that depend on where service providers are classified.

UBER INDUSTRY OUTLOOK CURRENTLY:

Uber kickstarted the evolution of the taxi market in the early 2010s, when it launched an app which easily connected drivers with riders. In California where the app launched, ordering a cab was a nightmare, so much so that co-founder Garrett Camp had established his own fleet of black cabs to pick him up from bars and clubs.

Camp recognized the value of this service, which was less temperamental than waiting on the sidewalk for a cab or waiting more than an hour for someone to pick them up. Uber Cab was founded in 2009 and Travis Kalanick joined shortly afterwards. Kalanick become the CEO shortly afterwards in December 2010, a position he held until 2017.

During Kalanick's tenure as CEO, Uber became the brand most people associate with ride hailing. It took the US by storm and quickly expanded into Europe, Asia, and South America. Kalanick also oversaw Uber launch Eats, Freight, and its autonomous vehicle unit, as he attempted to expand Uber's reach to all parts of the transportation market.

Kalanick's tenure brought lots of growth, but it also brought controversy. Uber's entry into countries and states often came before agreements had been made with those in power, which led to penalties and bans in some areas of the world. Allegations of sexual harassment and bullying in the workplace led to Kalanick resigning in 2017 and being replaced by Expedia CEO Dara Khosrow Shahi.

Since 2017, Uber has taken its foot off the gas, selling its stake in India, and leaving several European countries. It also sold its autonomous vehicle unit to Aurora, in return for a significant stake in the startup. It has cleaned up its image, although several countries are currently looking into whether Uber drivers should be considered employees. The UK was one of the first countries to recognize them as such.

The coronavirus pandemic hit Uber's ride-hailing business hard, but by Q4 2020 it had reached pre-pandemic revenue figures. Uber Eats became the main business during this time, with over 200 percent increase in revenue year-on-year. In 2022, ride hailing surpassed delivery as the main driver of revenue and operating profit, and this continued into 2023.

We have collected key statistics on Uber. Read on below to find out more.

Uber Key Statistics

- Uber generated $37.2 billion revenue in 2023, an 16% increase on the previous year.
- It made $19.6 billion revenue from ride-hailing, and $12.1 billion from delivery. The rest came from freight services.
- 137 million people use Uber or Uber Eats once a month, an 11% increase year-on-year.
- Uber drivers completed 9.44 billion trips in 2023, almost two billion more than in 2022.

VARIABLES USED IN DATASET:

The training data set serves as the basis for the forecasting models, which are utilized to foretell the testing data set. The dataset's Open, High, Low, and Close columns represent the Opening, Highest, Lowest, and Closing prices of Uber price value to the US dollar on a given day. Change % refers to the change in price value over the years, whereas Volume refers to the volume of uber value incurred on a specific day. In the dataset we have chosen it has 7 variables namely, The dataset we have chosen has 7 variables namely,

Date Date of Listing (YYYY/MM/DD).

Price Price Value.

Close/Last Price at market closing.

Volume Number of stocks sold in a day.

Open Price at market opening.

High Highest price recorded for the day.

Low Lowest price recorded for the day.

The price of uber data throughout that period is the main determinant. Over the years, we have seen a consistent seasonal change in the price of uber, and the trend has been nearly constant. We used the ARIMA and dynamic regression model approach for forecasting and obtained the model's lowest AICc score. The Forecast varies amongst models, and we select the best model based on values with the lowest RMSE. Based on the testing data set's least AICC and RMSE values, we forecast the model.

METHODOLOGY:

- With over 118 million users, 5 million drivers, and 6.3 billion trips with 17.4 million trips completed per day - Uber is the company behind the data for moving people and making deliveries hassle-free.
- How are drivers assigned to rides cost-efficiently, and how prices are changing over the years.

- Thanks to the large volumes of data Uber collects and the fantastic team that handles Uber Data Analysis using R tools and frameworks.
- I am curious to learn more about how data analysis is done at Uber to ensure positive experiences for riders while making the ride profitable for the company - Get your hands dirty working with the Uber dataset to gain in-depth insights.
- With the help of visualization, uber can avail the benefit of understanding complex data and gaining insights that would help them to craft decisions.
- In this data analysis, we analyze Uber data from **May 2019 to April 2024**.

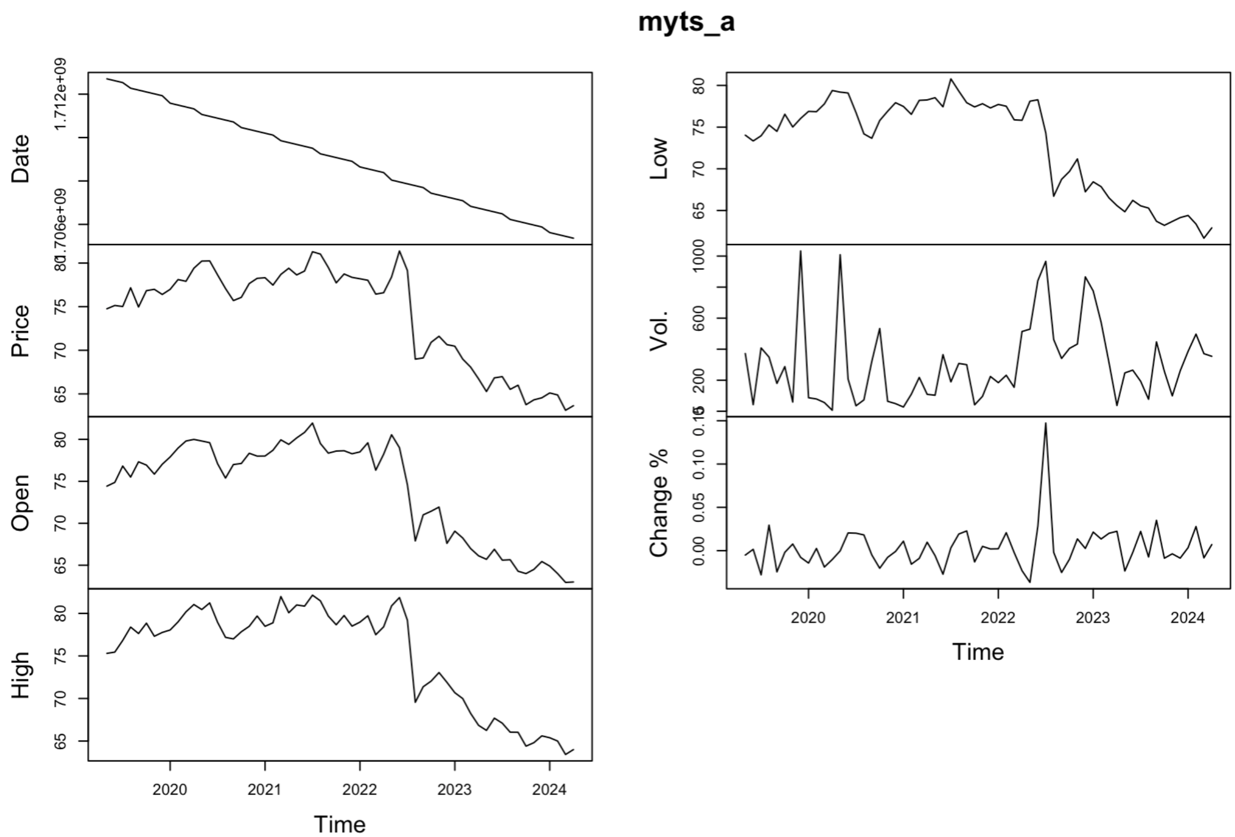
With the help of four methodologies, namely Holt's, Time Series Regression,ARIMA forecasting models, and Dynamic Regression Models, we have predicted the price of Uber for the upcoming year.

We must first comprehend the data, which entails extracting or forecasting the trend. We use the time series model on the data and examine the data's stationarity as well.

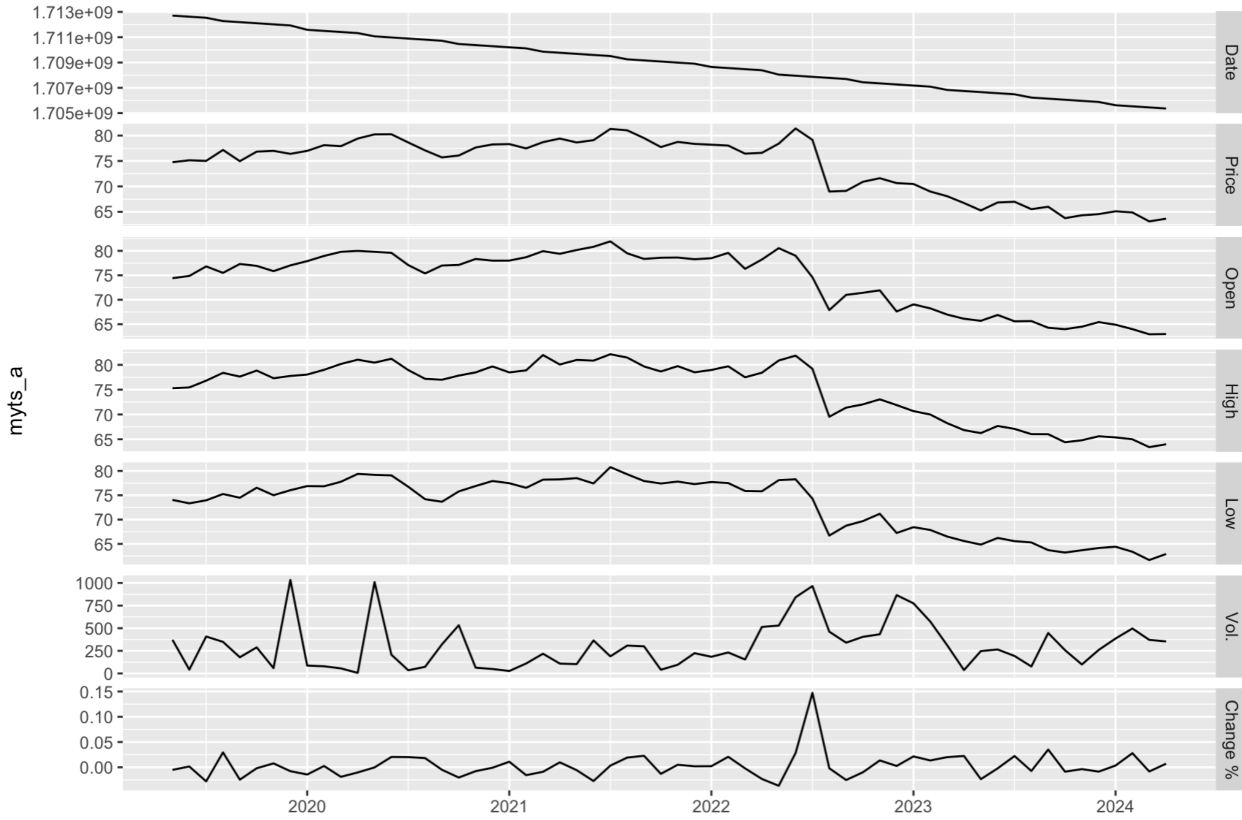
Initial Data Analysis:

Plotting Raw Data:

Annual Data:

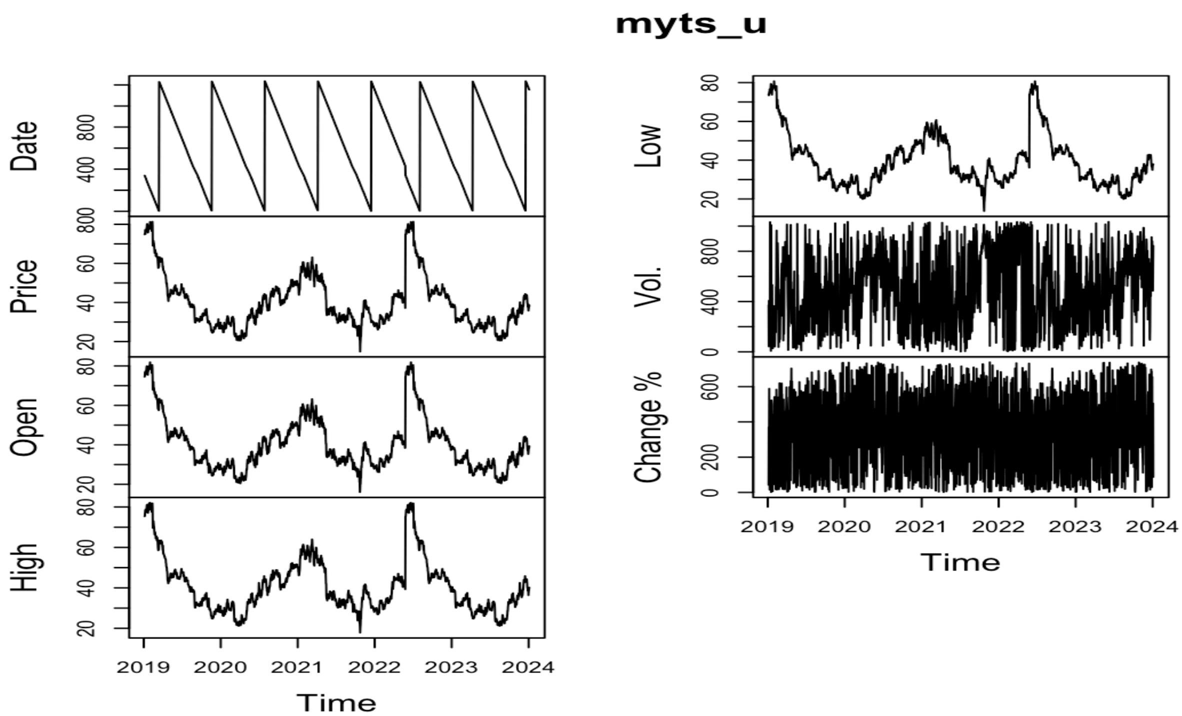


myts_a
Date
1.712e+09
1.706e+09
Price
Open
High
Low
Vol.
Change %
Time
2020
2021
2022
2023
2024

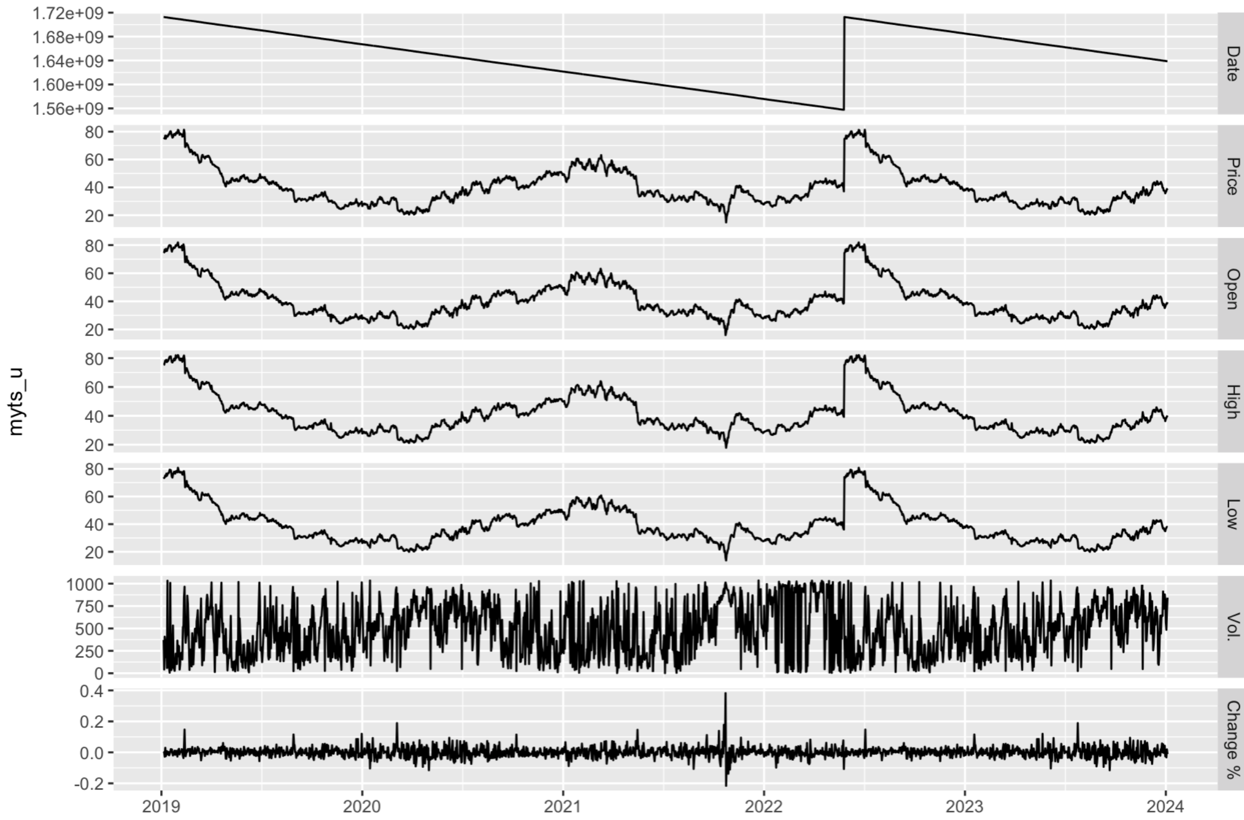


myts_a
Date
Price
Open
High
Low
Vol.
Change %
1.713e+09
1.711e+09
1.709e+09
1.707e+09
1.705e+09
80
75
70
65
1000
750
500
250
0
0.15
0.10
0.05
0.00
2020
2021
2022
2023
2024

Daily Data:



myts_u
Date
Price
Open
High
Low
Vol.
Change %
Time
Time
2019
2020
2021
2022
2023
2024



Exponential Smoothening

The simplest of the exponential smoothing methods is naturally called **simple exponential smoothing** (SES). This method is suitable for forecasting data with no clear trend or seasonal pattern. Exponential smoothing is a forecasting method for univariate time series data. This method produces forecasts that are weighted averages of past observations where the weights of older observations exponentially decrease. Forms of exponential smoothing extend the analysis to model data with trends and seasonal components.

Statisticians began developing exponential smoothing back in the 1950s. Since then, it has enjoyed a very successful presence among analysts as a quick way to generate accurate forecasts in diverse fields, particularly in industry. It's also used in signal processing to smooth signals by filtering high-frequency noise.

In this post, I show you how to use various exponential smoothing methods, including those that can model trends and seasonality. These methods include simple, double, and triple (Holt-Winters) exponential smoothing.

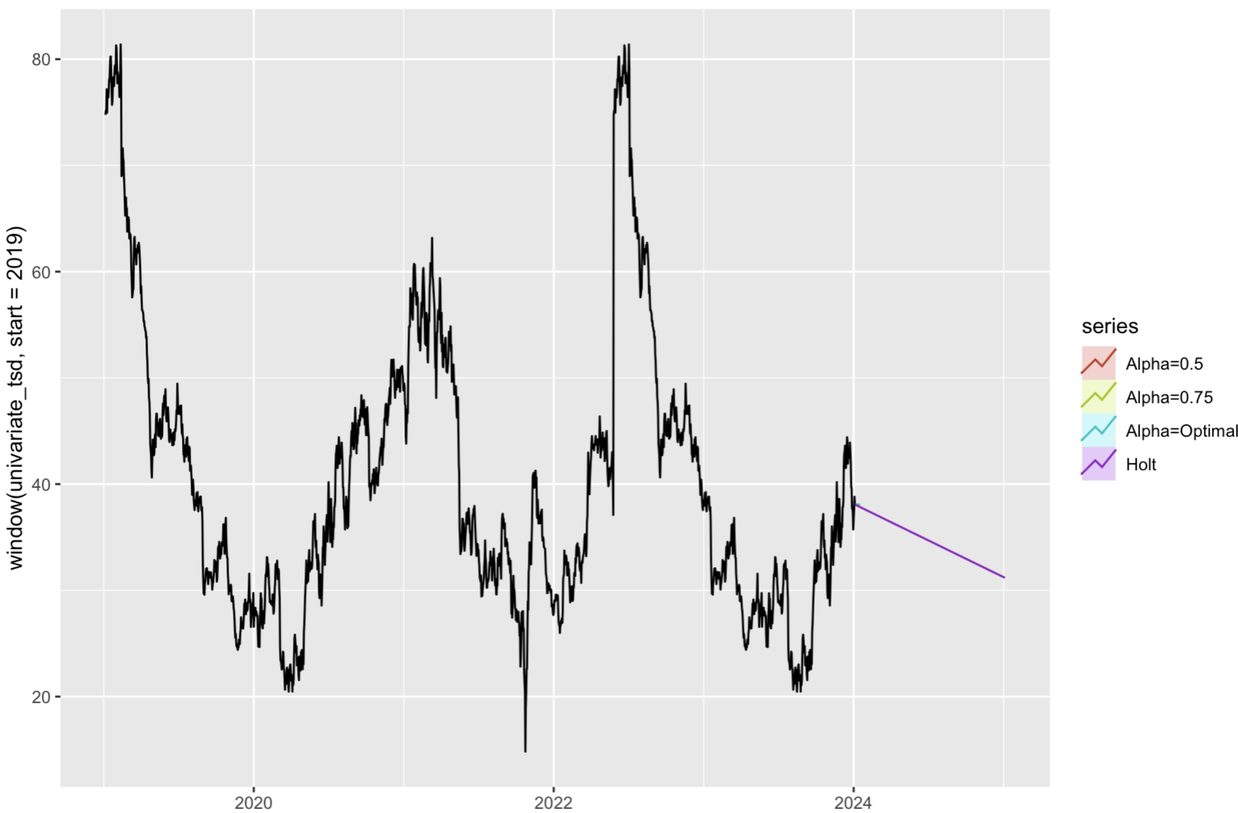
There is an Increasing trend with small cycles till year 2022 and decreasing trend from 2023.

Daily Data:

Daily Data:

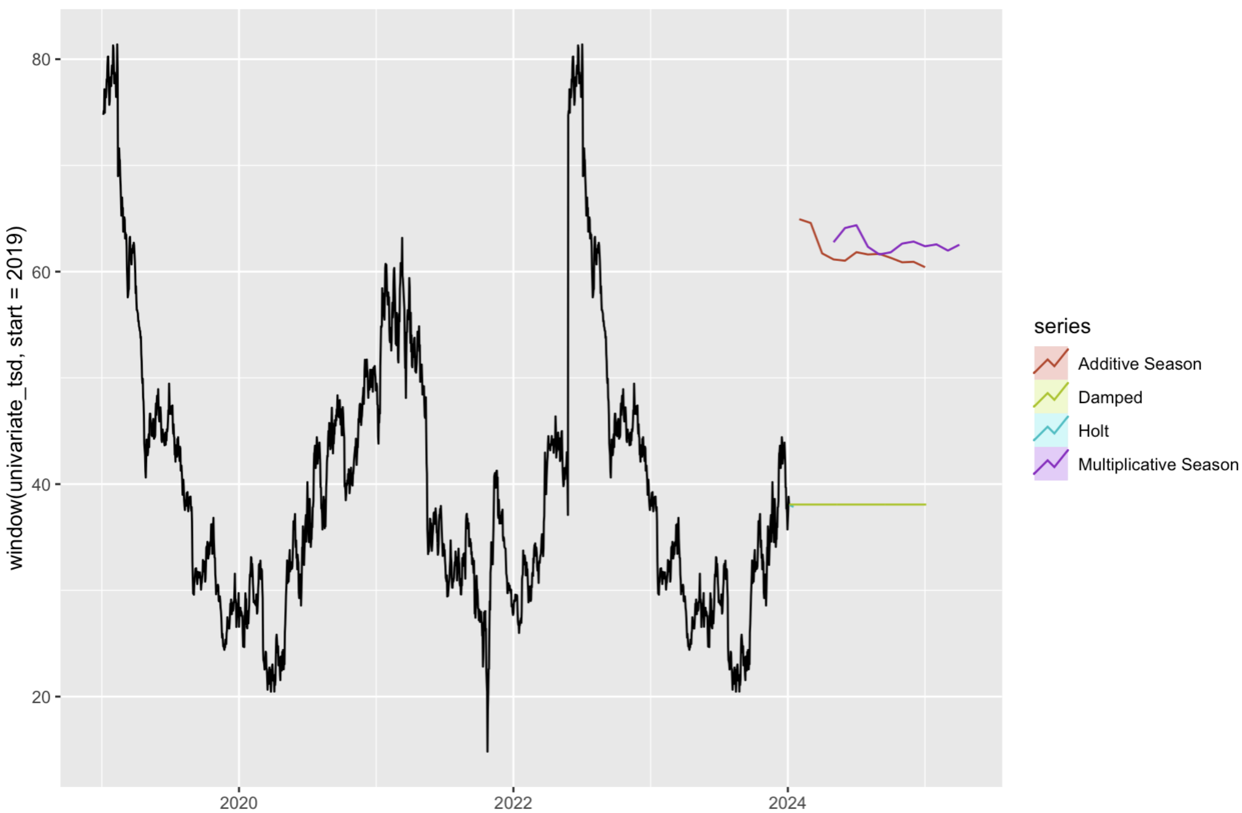
Plot for all the alpha values -0.5,0.75 and R picked value and Holt but holt model

Plot looks visible and kind of the same pattern .



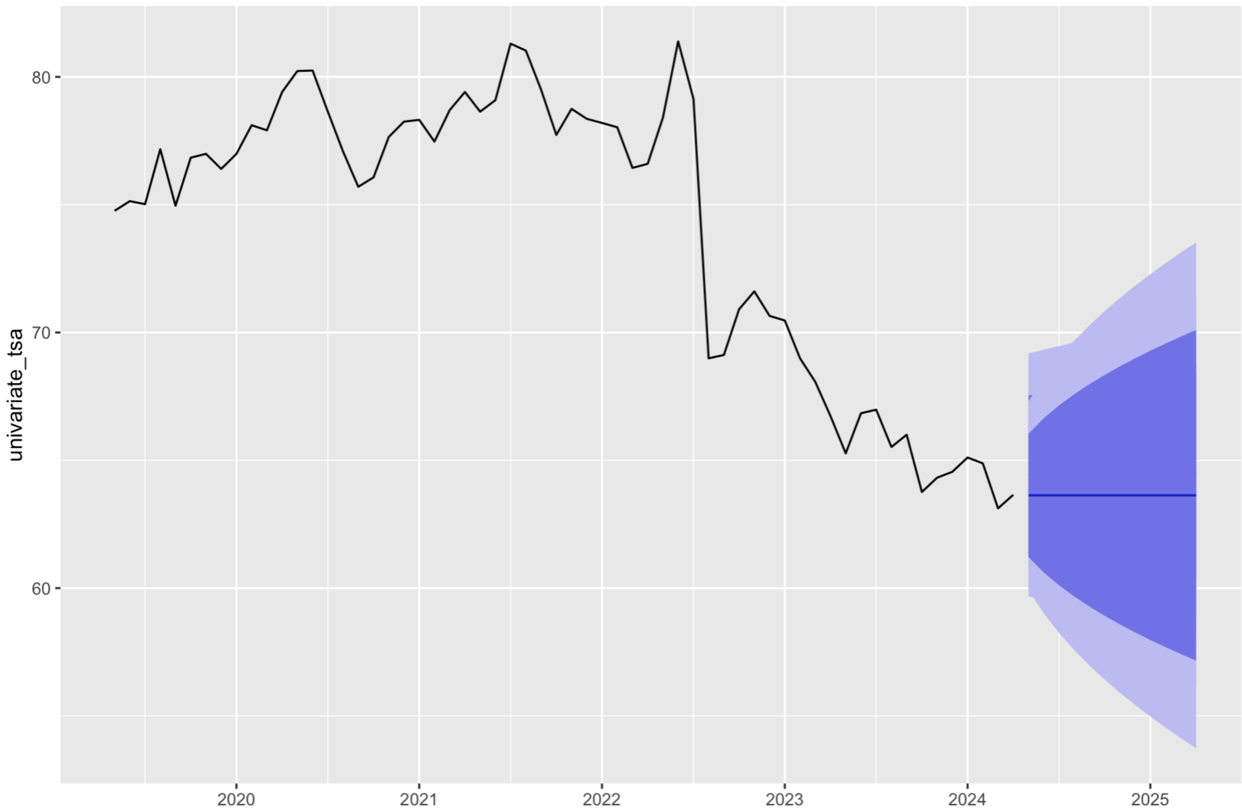
Daily Data:

For all possible models (A,M,D)



Plot looks slightly accurate for the damped layer.

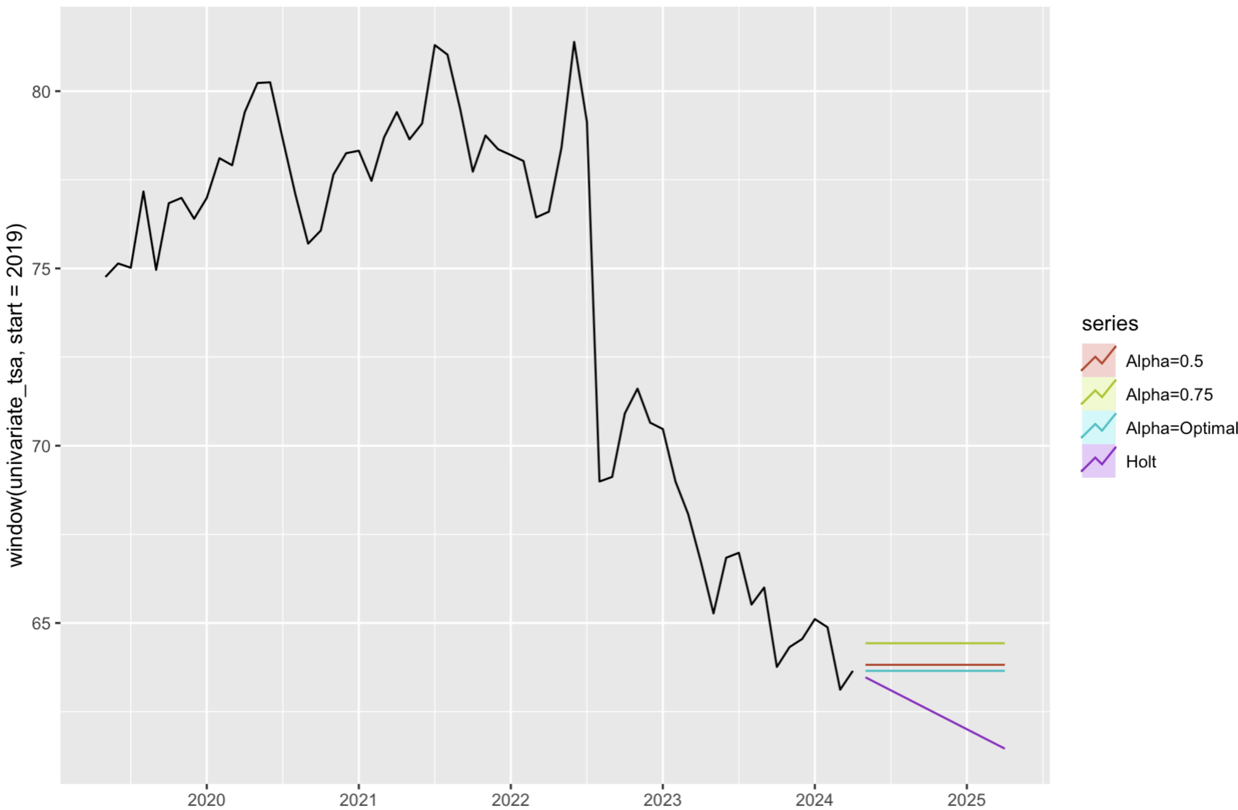
Annual Data:



Plot looks slightly off.

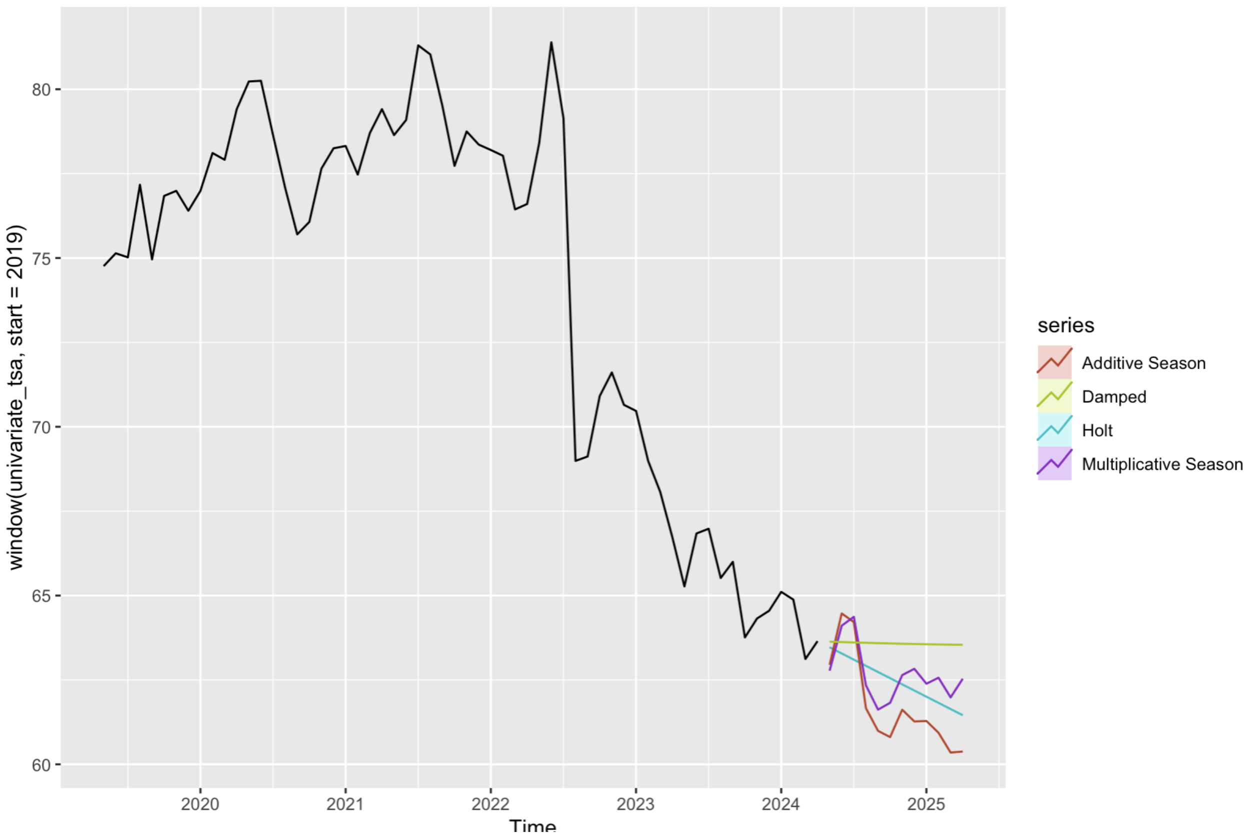
Annual Data:

For Alpha values 0.5, 0.75 and R alpha picked values and holt model



Plot looks slightly off for all models

Annual data–For all possible models:



In the above graph, as it can be observed, the price of uber has exhibited a consistent variance over time, and the trend has been essentially accurate with annual frequency and with the daily data the plot is kind of implausible due to the high frequency in some cases ,we can see a constant pattern in (multiplicative) damped model working in case of daily data .

Training and Testing Splitting:

We have divided the data into a training set and testing set into 80:20 ratio.

- Training data set = 80%
- Testing data set = remaining 20%

FORECASTING MODELS:

1. **Holt's Forecasting Model:**

Holt-Winters is a model of time series behavior. Forecasting always requires a model, and Holt-Winters is a way to model three aspects of the time series: a typical value (average), a slope (trend) over time, and a cyclical repeating pattern (seasonality).

Time series anomaly detection is a complicated problem with plenty of practical methods. It's easy to get lost in all of the topics it encompasses. A key element of anomaly detection is forecasting—taking what you know about a time series, either based on a model or its history, and making decisions about values that arrive later.

You know how to do this already. Imagine someone asked you to forecast the prices for a certain stock, or the local temperature over the next few days. You could draw out your prediction, and chances are it's a pretty good one. Your brain works amazingly well for problems like this, and our challenge is to try to get computers to do the same.

Especially when monitoring systems, this approach doesn't work well, if at all! Real systems rarely fit mathematical models. There's an alternative. You can do something a lot simpler with exponential smoothing.A step up from simple exponential smoothing, Holt's exponential smoothing method is capable of taking into account a trend component. Holt’s method is often referred to as double exponential smoothing.Holt’s method extends simple exponential smoothing by assuming that the time series has both a level and a trend. A forecast with Holt’s method can therefore be defined as:

$$F_{t+k} = L_t + kT_t$$

Where:

L_t is the level estimate for time t, k is the number of forecasts into the future, and T_t is the trend at time t.

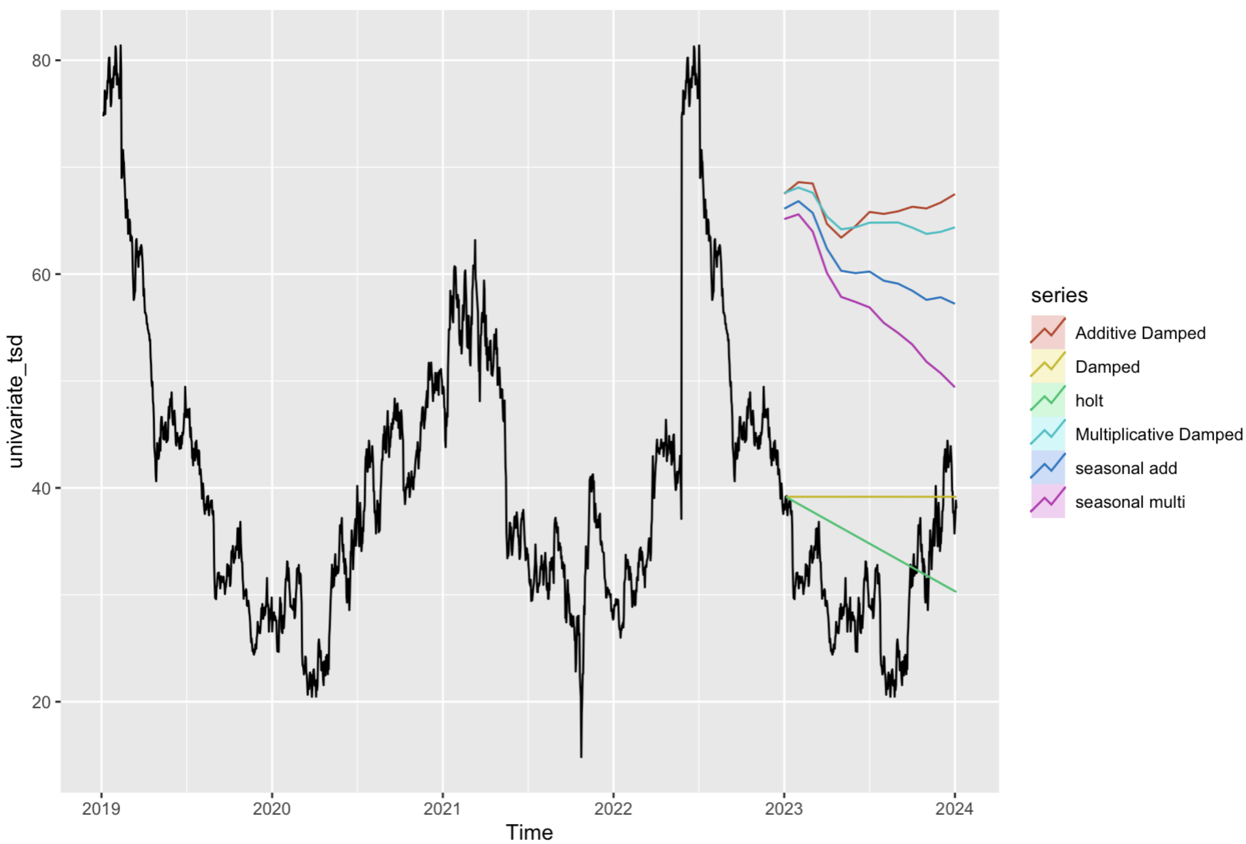
As we can see, it is literally just a simple extenuation of original *SES* method, just with the inclusion of the trend, *T*, component.However, it is important to note that there are two types of time series, each with their own slightly different forecasting equation:

Additive: In an additive time series, the time series is the sum of its components.

Multiplicative: In a multiplicative time series, the time series is the product of its components.

Exponential smoothing is used in Holt's forecasting, also known as the double exponential smoothing approach, depending on the input values alpha and beta.

Daily Data:



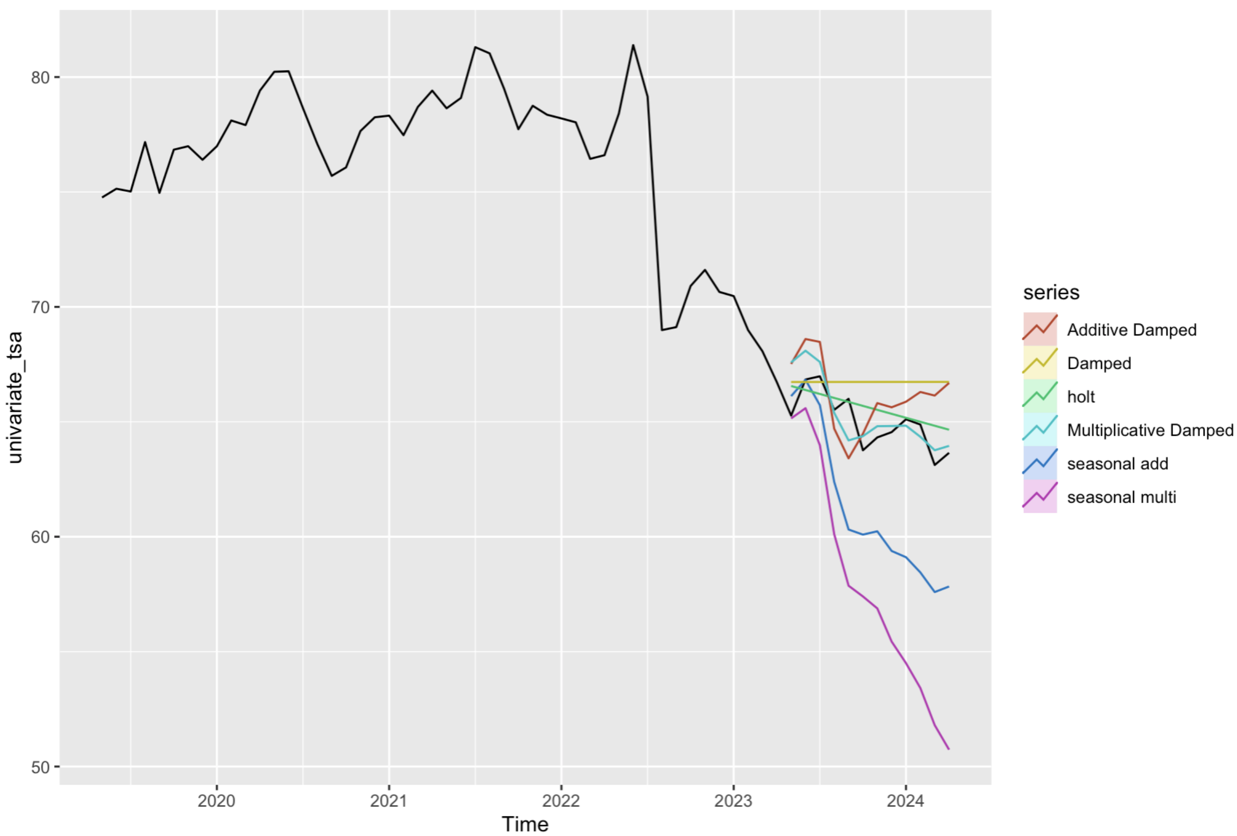
We choose a basic Holt model with damped for daily data due to low RMSE value.

	Holt's Forecasting Model
ME	-0.02620114
MAE	0.9268319
RMSE	1.580192

MPE	-0.1136293
MAPE	2.322778

If it is under 10%, the mean accurate prediction error 9.2% is a reasonably good fit. The model can't foresee the drop in uber prices, which is the issue.

Annual Data:



We choose basic Holt Multiplicative with a damped model as well due to low RMSE value.

	Holt's Forecasting Model
ME	-0.3126899
MAE	1.1639221
RMSE	1.741059
MPE	-0.45777365
MAPE	11.544292s

2. **<u>ETS:</u>**

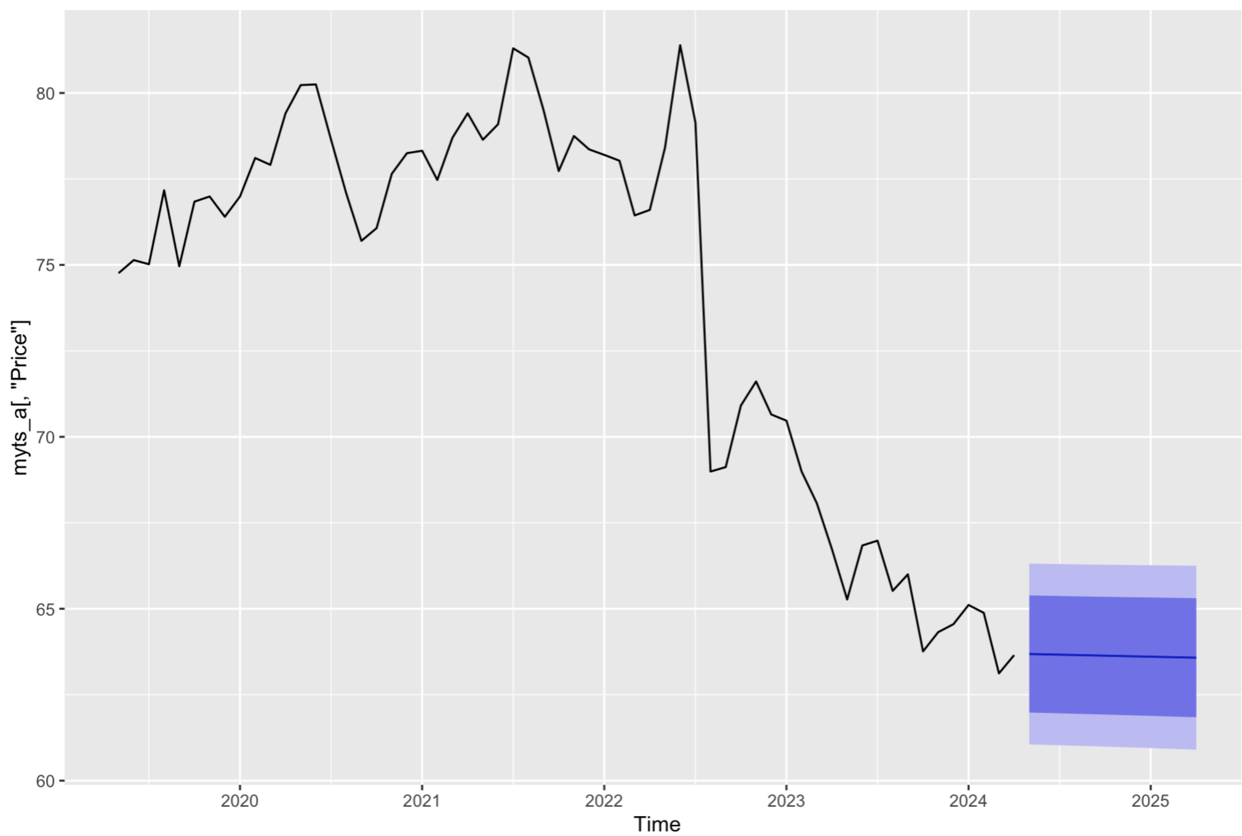
These forecasts are identical to the forecasts from Holt's linear method, and also to those from model ETS(A,A,N). Thus, the point forecasts obtained from the method and from the two models that underlie the method are identical (assuming that the same parameter values are used).

ETS point forecasts are equal to the medians of the forecast distributions. For models with only additive components, the forecast distributions are normal, so the medians and means are equal. For ETS models with multiplicative errors, or with multiplicative seasonality, the point forecasts will not be equal to the means of the forecast distributions.

<u>For Annual Data: ExAnte Forecasting</u>

	ETS Forecasting Model
ME	-1.105597e-15
MAE	0.9656448
RMSE	1.2192466
MPE	-0.027764631
MAPE	1.295166

<u>Forecast for Ex-Ante forecasting:</u>



For Daily Data: ExAnte Forecasting

	ETS Forecasting Model
ME	2.336031e-17
MAE	0.7683055
RMSE	1.008312
MPE	-0.07673757
MAPE	2.040446

Forecast for Ex-Ante forecasting:

3. Time Series Regression Models:

Simple linear regression:

In the simplest case, the regression model allows for a linear relationship between the forecast variable y and a single predictor variable x:$y_t=\beta_0+\beta_1x_t+\varepsilon_t$.The coefficients β_0 and β_1 denote the intercept and the slope of the line respectively. The intercept β_0 represents the predicted value of y when x=0. The slope β_1 represents the average predicted change in y resulting from a one unit increase in x. Notice that the observations do not lie on the straight line but are scattered around it. We can think of each observation y_t as consisting of the systematic or explained part of the model, $\beta_0+\beta_1x_t$, and the random "error", ε_t. The "error" term does not imply a mistake, but a deviation from the underlying straight line model. It captures anything that may affect yt other than xt.

Stepwise regression:

If there are a large number of predictors, it is not possible to fit all possible models. For example, 40 predictors leads to 240> 1 trillion possible models! Consequently, a strategy is required to limit the number of models to be explored.

An approach that works quite well is *backwards stepwise regression*:

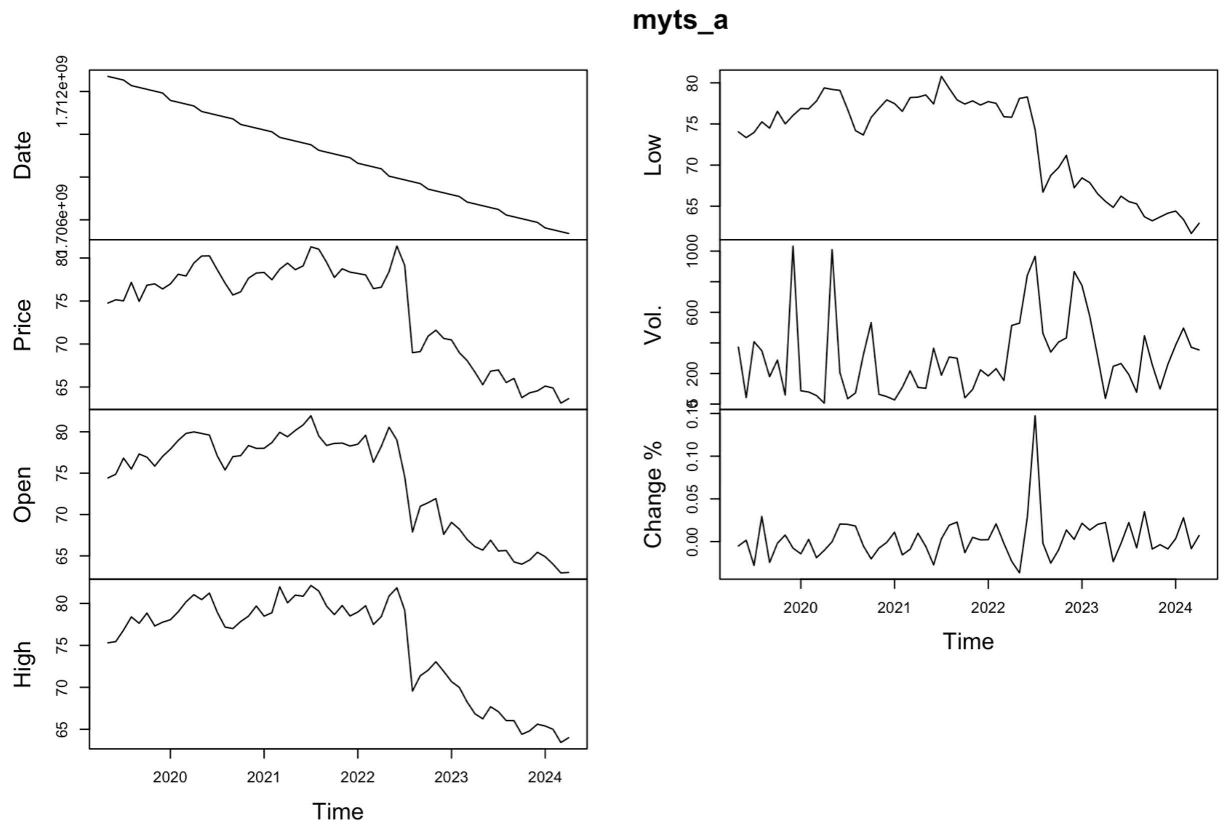
- Start with the model containing all potential predictors.
- Remove one predictor at a time. Keep the model if it improves the measure of predictive accuracy.
- Iterate until no further improvement.

If the number of potential predictors is too large, then the backwards stepwise regression will not work and *forward stepwise regression* can be used instead. This procedure starts with a model that includes only the intercept. Predictors are added one at a time, and the one that most improves the measure of predictive accuracy is retained in the model. The procedure is repeated until no further improvement can be achieved.

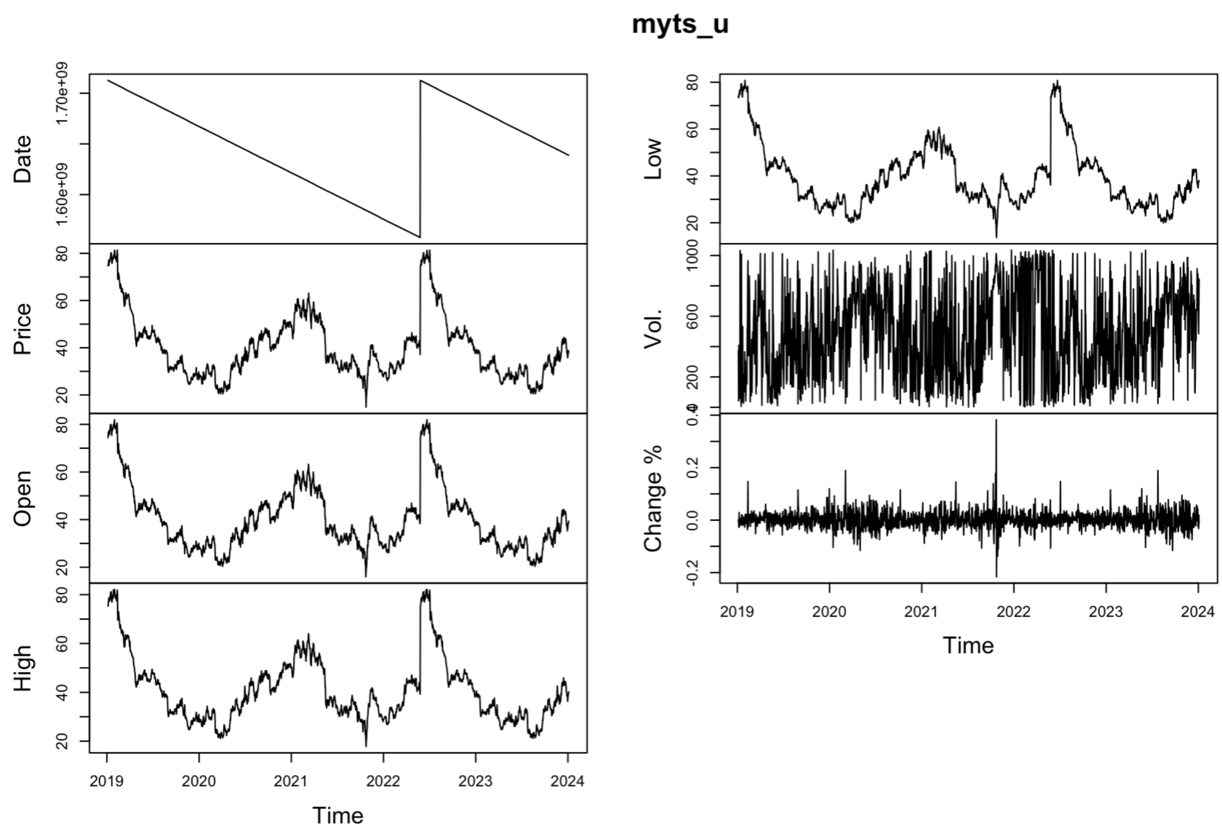
Alternatively for either the backward or forward direction, a starting model can be one that includes a subset of potential predictors. In this case, an extra step needs to be included. For the backwards procedure we should also consider adding a predictor with each step, and for the forward procedure we should also consider dropping a predictor with each step. These are referred to as *hybrid* procedures.

Initial Plot of Time Series Regression :

Annual data



Daily Data:



Backward Stepwise regression:

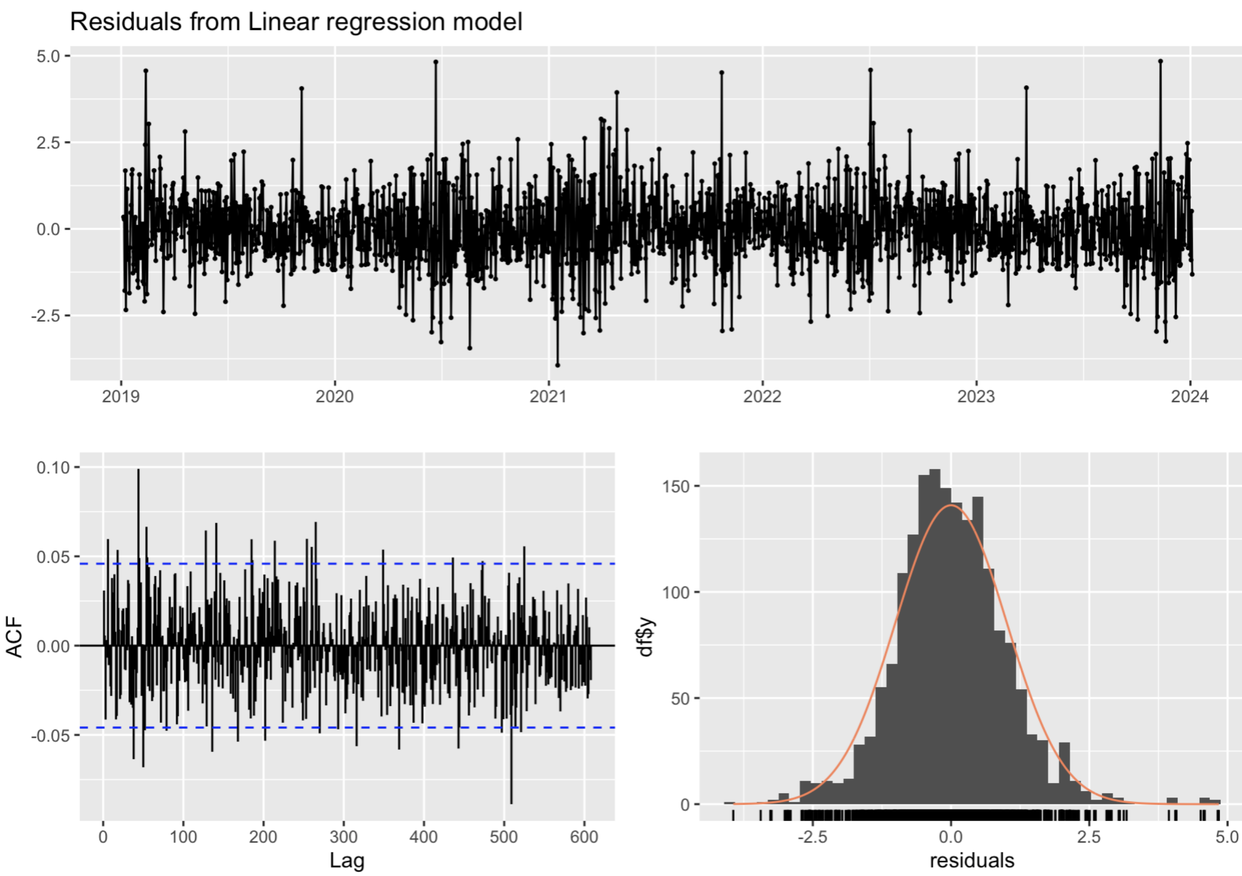
For Daily Data:

Checking the residuals for the best fitting Model.

bsr12 = tslm(Price ~ Date +Open+trend,myts_u)

CV(bsr12) # 40.247162 -best till now

Residuals for the best model:



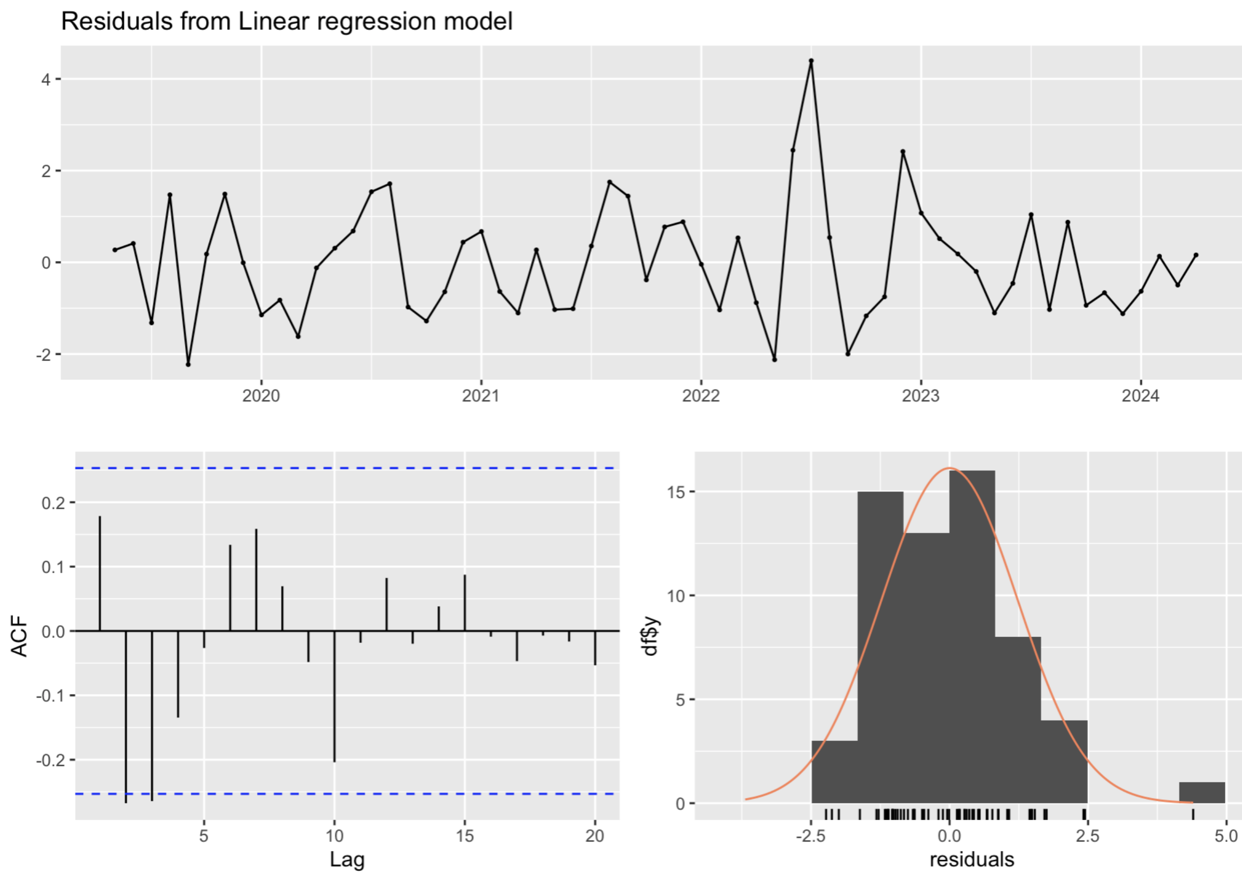
Looks like White Noise - p-value= 0.005747 >0.001 and 0.005747< 0.05

<u>For Annual Data:</u>

bsr120 = tslm(Price ~ Date +Open+trend,myts_a)

CV(bsr120) #34.8990123 -best till now

Residuals for best model:



Looks like white noise as p value -0.3758

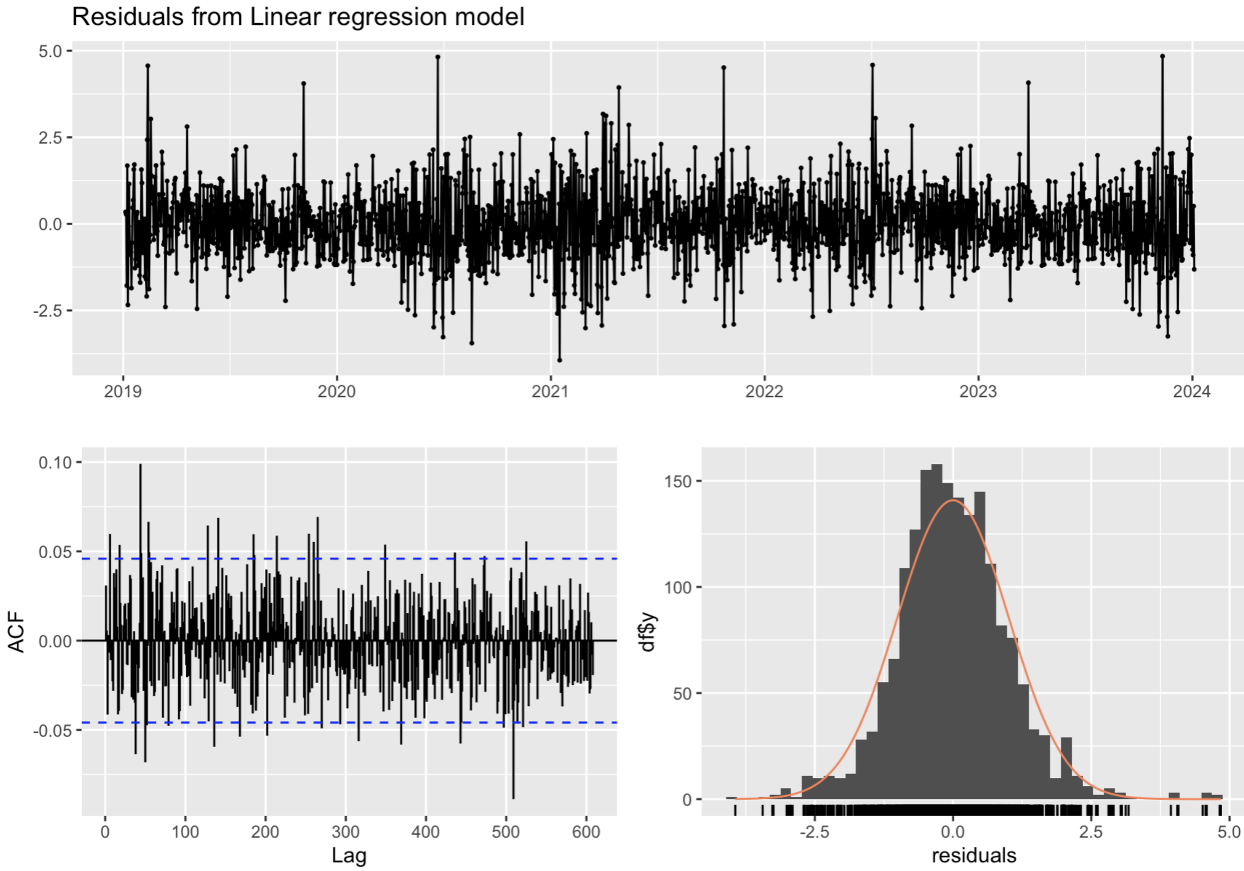
<u>Forward Stepwise Regression:</u>

For Daily Data: Checking the residuals for best fitting model:

sr4 = tslm(Price~trend+Date+Open,myts_u)

CV(sr4)#40.247162-Best till now

Residuals for the best regression model.

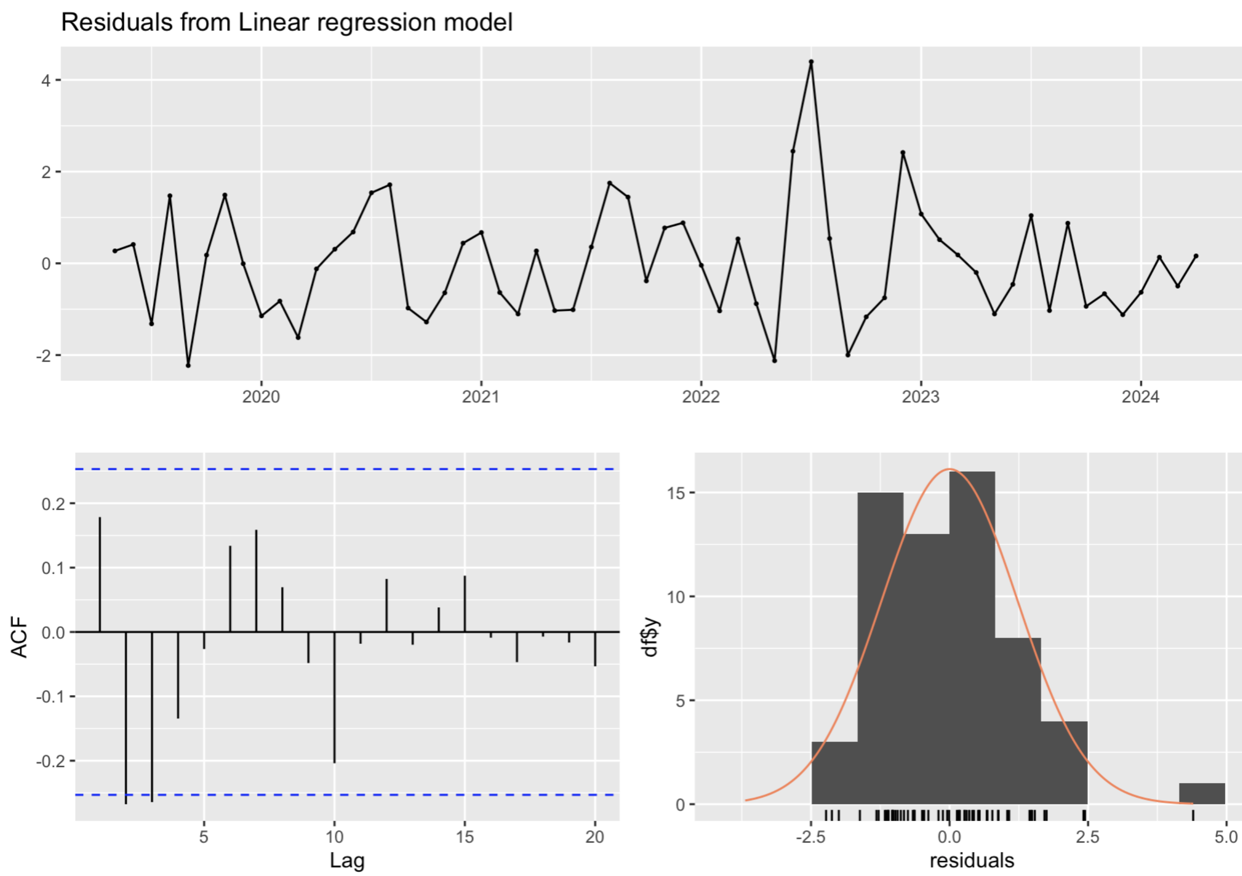


<u>For Annual Data:</u>

sr44 = tslm(Price~trend+Date+Open,myts_a)

CV(sr44)#34.8990123-Best till now

Residuals for the best model:



Looks like White Noise-p-value = 0.3758

4. ARIMA:

ARIMA is primarily known as the "**A**uto**R**egressive **I**ntegrated **M**oving **A**verage" model. Before stepping into the insights of ARIMA, we have to understand the important properties of time series, namely **stationarity** where the joint distribution of our time series depends upon the distance between the two respective periods, and **weakly dependence** where the correlation between two periods like in period t and t+h goes to 0 as h goes to infinity.

In short, ARIMA works for only nonseasonal patterns. With ets(), estimation is done using **Maximum Likelihood Estimation** to estimate the maximum probability of obtaining the dataset. To follow modeling procedures, we

- Plot the data to observe the variance.
- Apply Logs or BoxCox if necessary to smooth out the variance.
- Do differencing if the time series isn't stationary.
- Analyze ACF/PACF from the plot to get appropriate ARIMA(p,d,0) where ACF is sinusoidal or ARIMA(0,d,q) where PACF is sinusoidal.
- Pick the best fit model using AICc.
- Check for residuals
- Do forecasting in case of white noise residuals.

General equation of ARIMA:

$$y_t = c + \varphi_1 y_{t-1} + \cdots + \varphi_p y_{t-p} + \theta_1 e_{t-1} + \cdots + \theta_q e_{t-q} + e_t.$$

• Predictors include both lagged values of yt and lagged errors.

• Conditions on coefficients ensure stationarity.

• Conditions on coefficients ensure invertibility.

ARIMA usually:

• Combines ARMA model with differencing.

• (1 − B) dyt follows an ARMA model.

ARIMA(p,d,q) model:

AR: p = order of the autoregressive part.

I: d = degree of first differencing involved.

MA: q = order of the moving average part.

Results:

When considering for frequency=12(annual data):

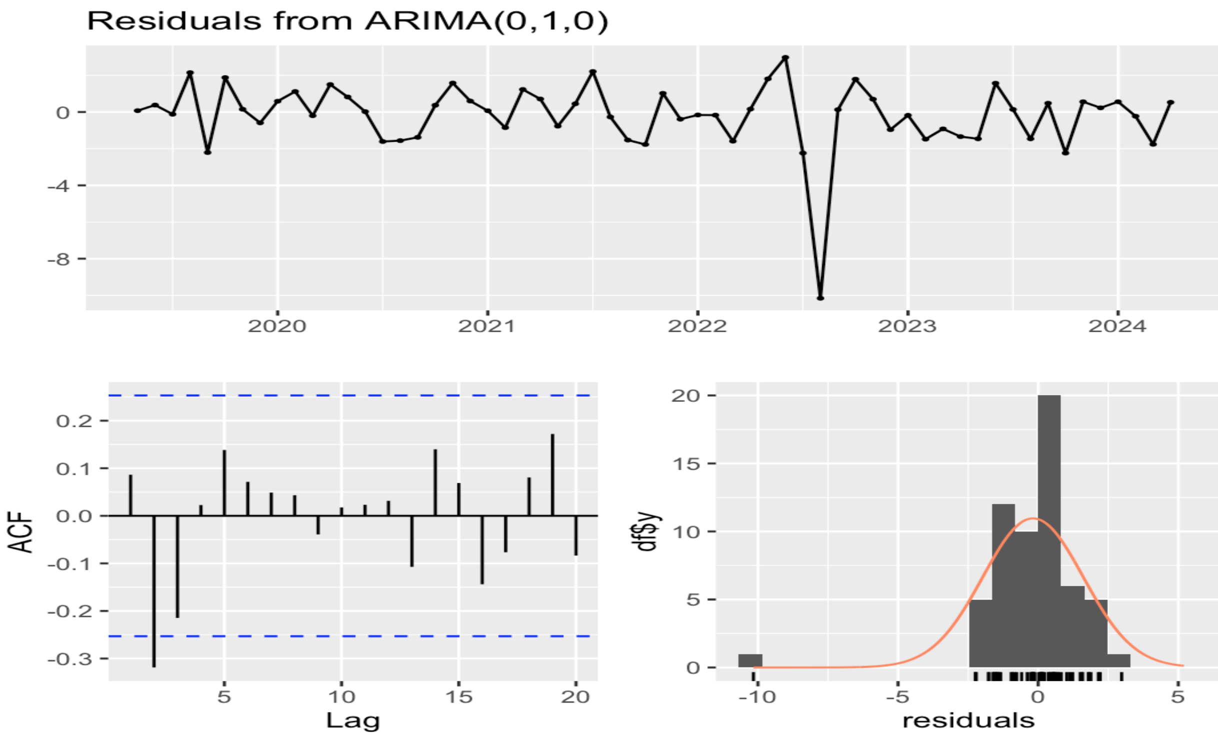
fit_4 = Arima(univariate_ts1,order = c(0,1,0))

summary(fit_4)#239.07(best)

Checking for presence of white noise:

checkresiduals(fit_4)

#There is a presence of white noise and since p-value is greater than 0.05, null hypothesis is not rejected.



Auto.arima() gives:

auto1 = auto.arima(univariate_ts1)

summary(auto1)

#Series: univariate_ts1

#ARIMA(0,1,0)

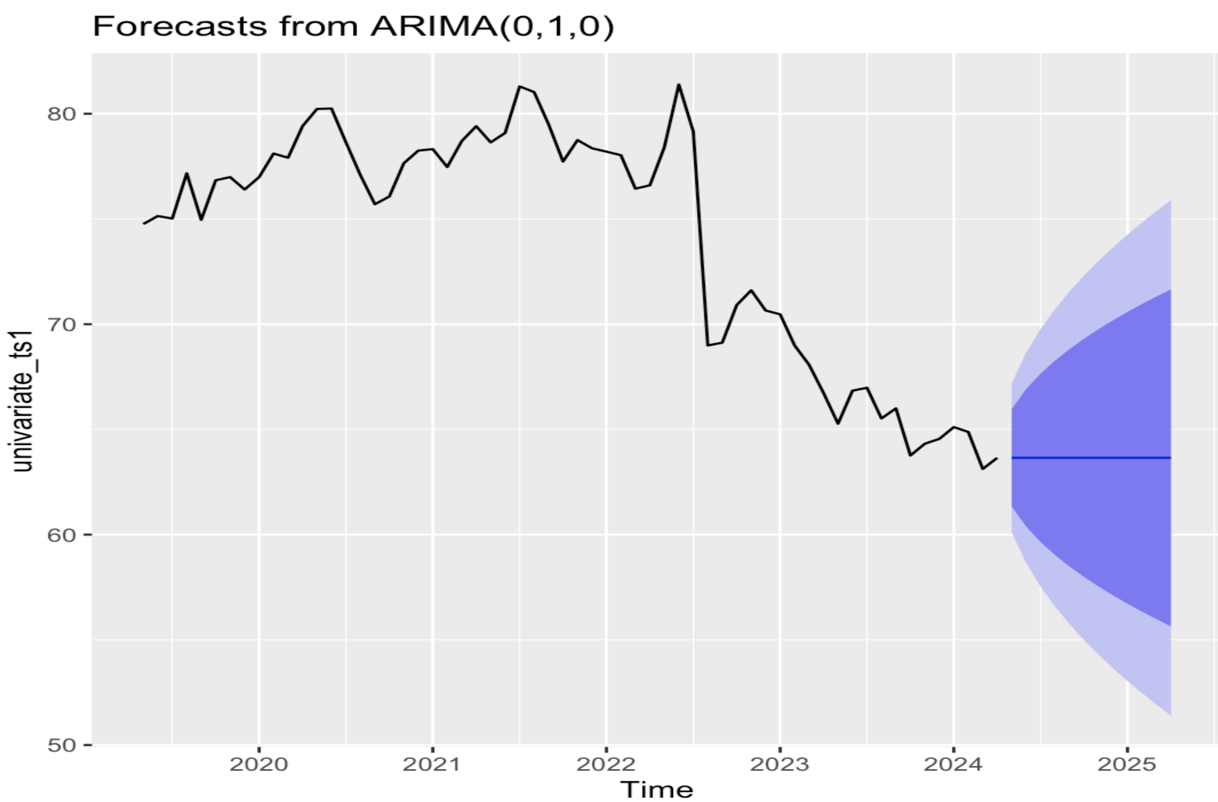
Forecasting:

fauto1 = forecast(auto1,h=12)

autoplot(fauto1)

summary(fauto1)

#This seems to have a forecast that is not very accurate for annual data.



	ARIMA Forecasting Model
ME	-0.1839
MAE	1.1360
RMSE	1.7881
MPE	-0.2963
MAPE	1.5516

<u>When considering for frequency=365(daily data)</u>:

fit4 = Arima(univariate_ts,order = c(0,1,0))

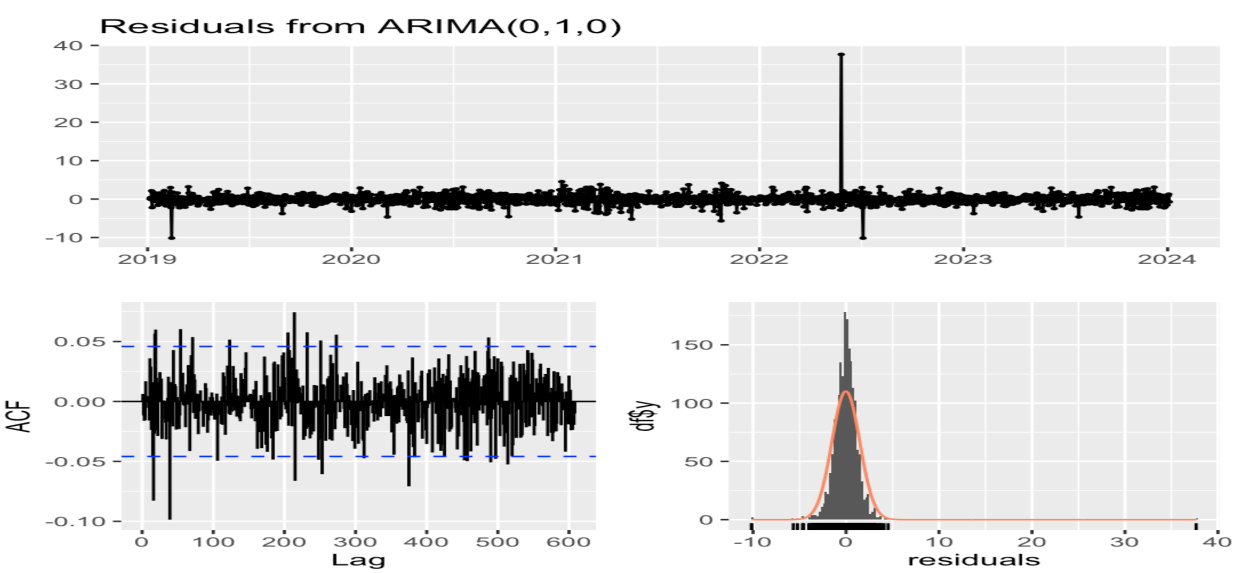
summary(fit4)#6648.73-we consider fit4 as best model till now let's check for closer models.

<u>Checking for presence of white noise</u>:

checkresiduals(fit4)

```
> checkresiduals(fit4)

        Ljung-Box test

data:  Residuals from ARIMA(0,1,0)
Q* = 363.82, df = 365, p-value = 0.5076

Model df: 0.   Total lags used: 365
```

#We check residuals in which we can partially accept that there is no presence of white noise and and since p-value is greater than 0.05, null hypothesis is not rejected.



Auto.arima gives:

auto = auto.arima(univariate_ts)

summary(auto)

#Series: univariate_ts

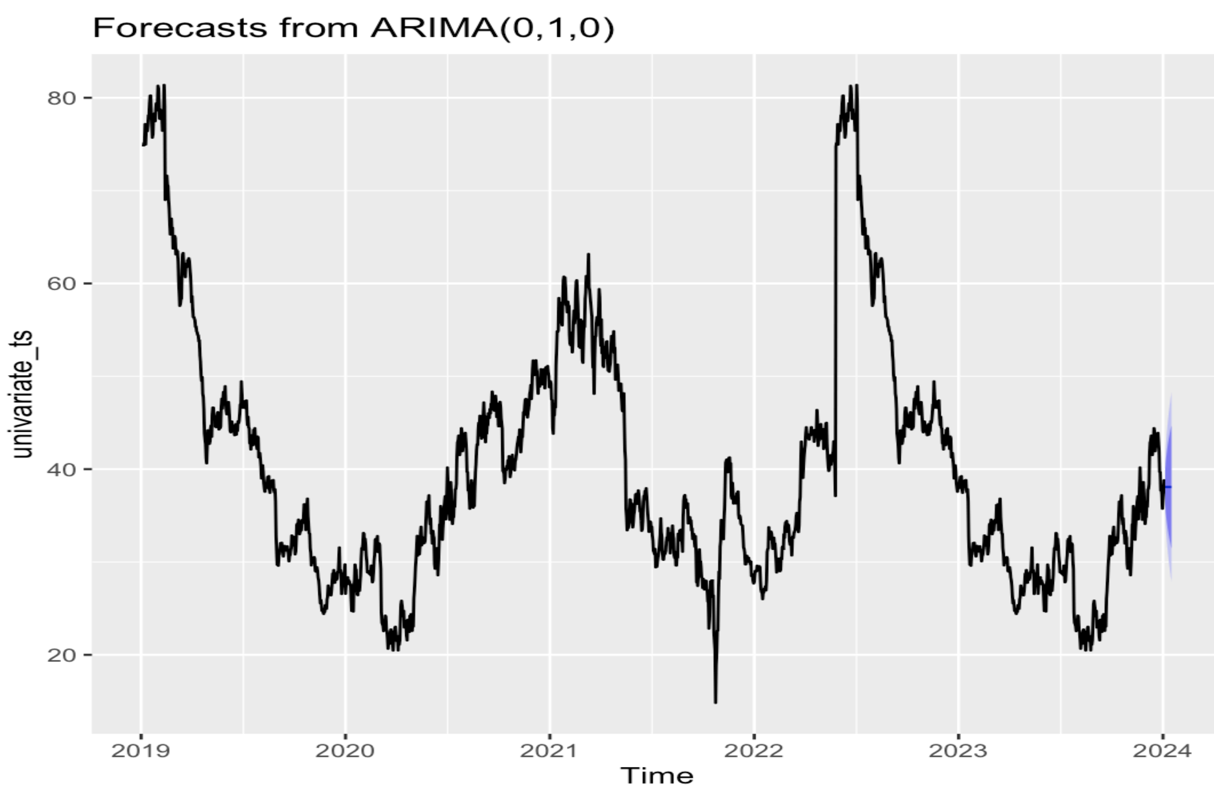
#ARIMA(0,1,0)

Forecasting:

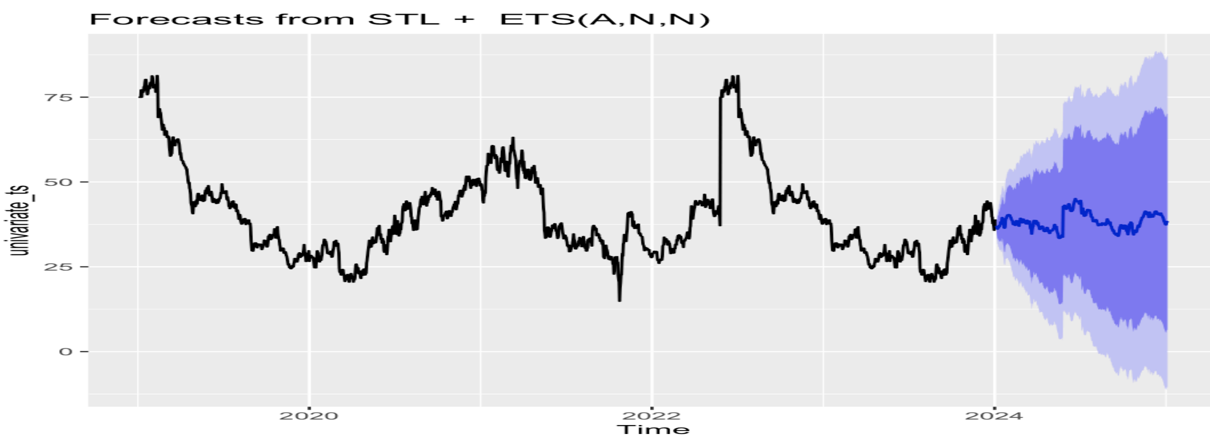
fauto = forecast(univariate_ts,h=365)

autoplot(fauto)

#The forecast seems to be reasonable.



#When we tried to forecast fauto, at first few runs we found that this model seems to be reasonable. But after doing multiple runs, we were able to notice the combination of STL (Seasonal, trend decomposition) + ETS(A,N,N) that made us interpret this change might be due to changes with the plot.



	ARIMA Forecasting Model
ME	-0.0198
MAE	0.8283
RMSE	1.3029
MPE	-0.0857
MAPE	2.2305

5. DYNAMIC REGRESSION MODEL:

Dynamic regression model will alter the assumption by allowing the error to continue with autocorrelation of a particular form where

$y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + e_t$

e_t = white noise

During estimation, we get problem with n^2 t as it causes issues with statistical tests associated with the model, p-values for coefficients that could be small, and AIC of fitted models could be misleading and we need to make sure to check for nearby ARIMA models to improve the fit. So, minimizing using e2t would be an optimal solution as it avoids these issues by applying differencing to make all variables in the model appear stationary. While selecting predictors, we use the least AICc to select the best model.

To forecast a regression model with ARIMA errors, we need to forecast the regression part of the model and the ARIMA part of the model and then combine the results. To forecast predictors, we can do Ex Ante or Scenario Based forecasting. Useful ways that could help us to include predictors are by

- Deterministic trend: $y_t = \beta_0 + \beta_1 t + n_t$ where n_t is the ARMA process.
- Stochastic trend: $y_t = \beta_0 + \beta_1 t + n_t$ where n_t is the ARIMA process with $d \geq 1$. We apply differencing on both sides until n_t is stationary: $y0t = \beta_1 + n0$ where $n0t$ is the ARMA process. This trend has wider forecasting intervals as errors are non-stationary.

Dynamic regression models deal with seasonality. In case of lengthy seasonal patterns, we use Fourier terms with alterations of sin, and cos waves that mimic seasonal patterns for large K factors.. In the case of fewer terms, seasonal data can have a quality approximation. We choose K by minimizing AICc.

Lagged explanatory variables: This model will include present and past values of predictor: x_t , x_{t-1}, x_{t-2},.... $y_t = a + v_0 x_t + v_1 x_{t-1} + \cdots + v_k x_{t-k} + n_t$ where n_t is an ARIMA process. We can choose the appropriate number of lags using AICc.

Important findings:

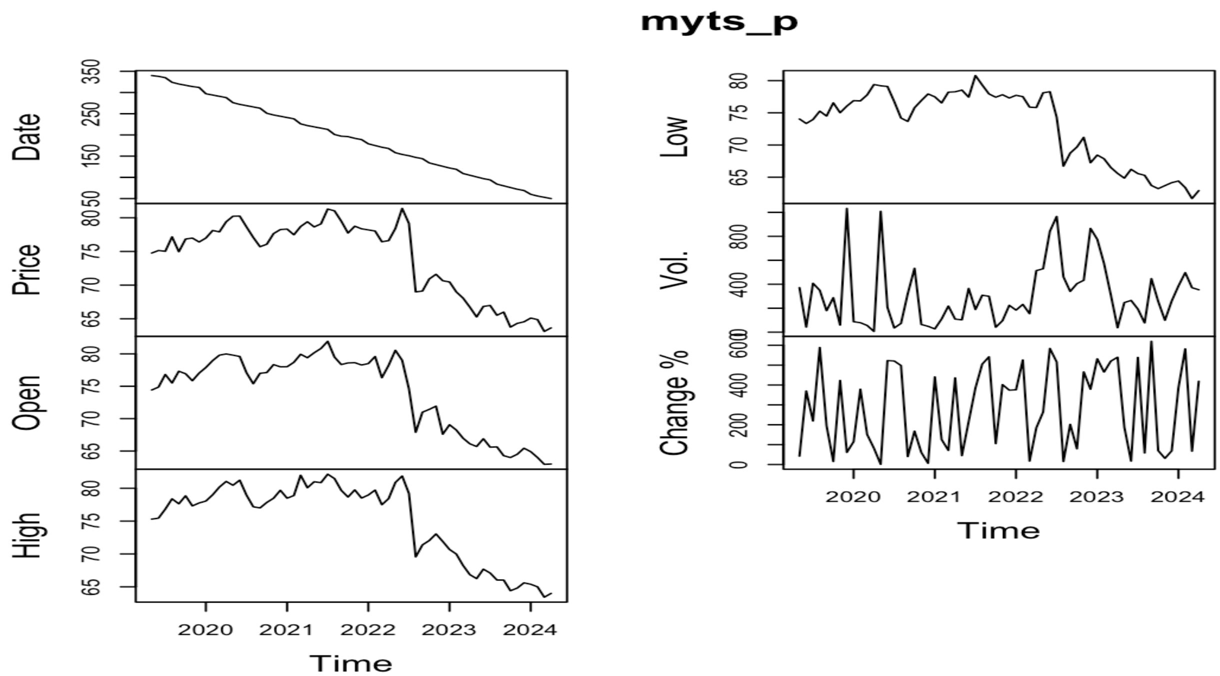
When considering for frequency=12 for annual data:

Train/test split:

```
summary(myts_p[,"Price"])

length(myts_p[,"Price"])#60

60*0.8=48

traindataAR11 = head(myts_p[,"Price"],48)

summary(traindataAR11)

testdataAR11 = tail(myts_p[,"Price"],12)

summary(testdataAR11)
```

myts_p =ts(UBER_Historical_Data, start=c(2019, 5), end=c(2024, 4), frequency=12)#annual data

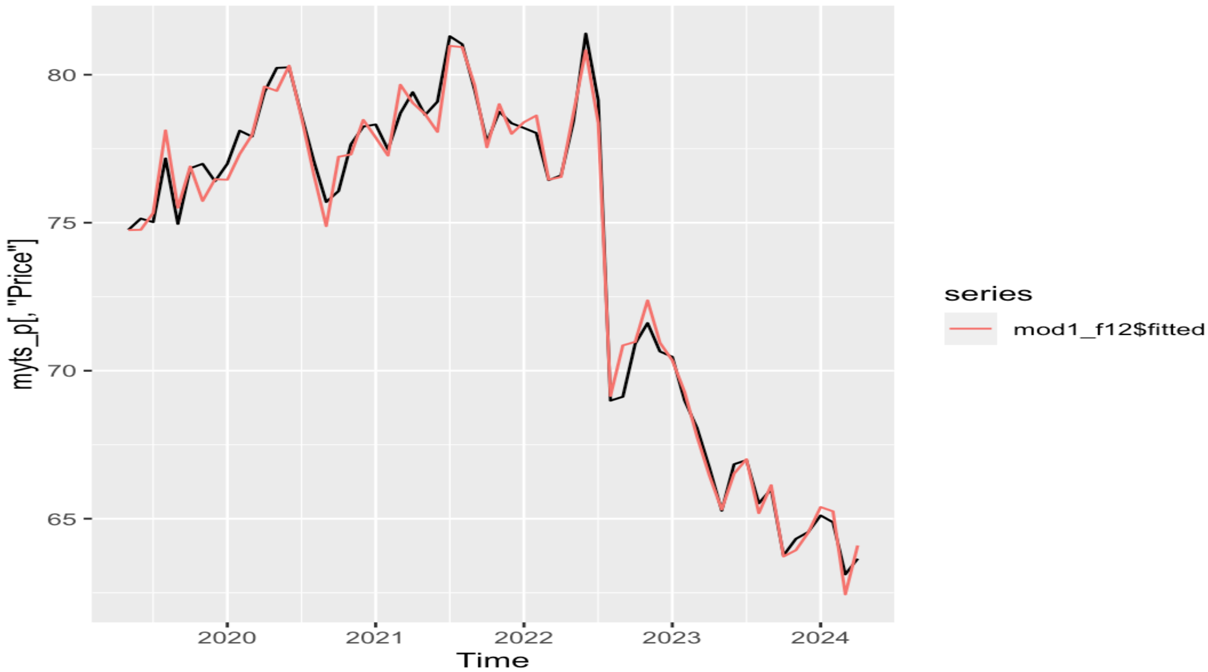
myts_p



After checking the presence of unit root for “price” variable, we apply one differencing and no seasonal differencing.

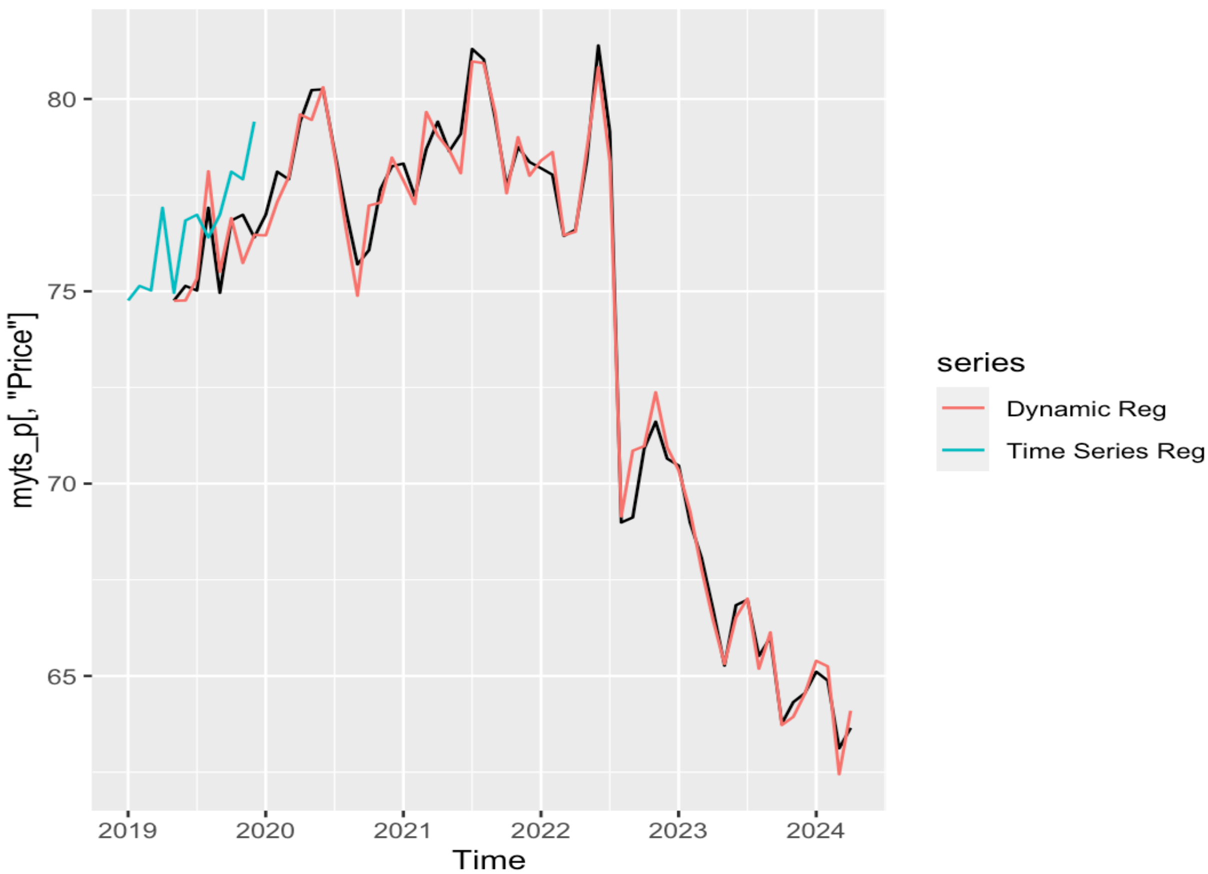
<u>Checking for models</u>:

#The maximum likelihood of the fit seems reasonable with slight changes.



<u>Comparing both dynamic regression model and time series model</u>:

#There is a smooth drift for dynamic regression models whereas in time series regression, we can notice an initiation of upward trend.



<u>Checking for the presence of white noise</u>:

checkresiduals(mod1_f12)

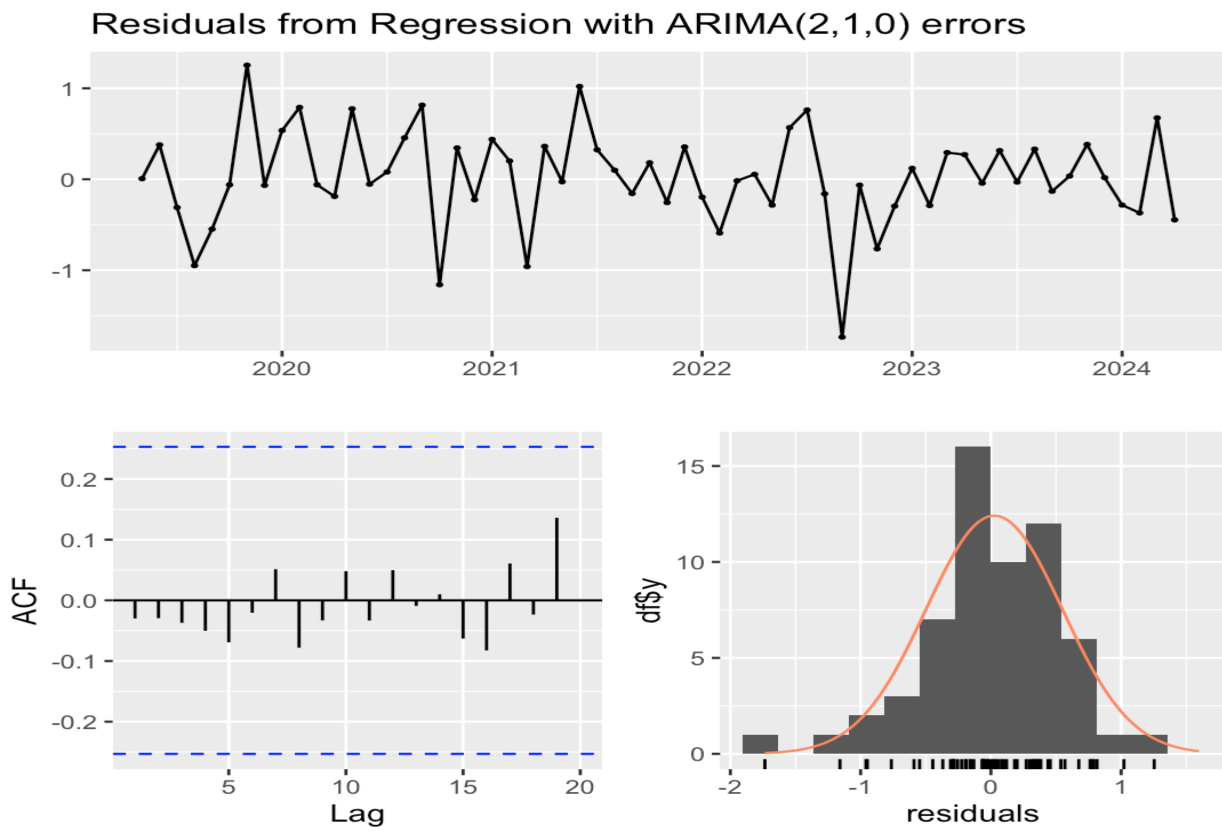
Ljung-Box test

data: Residuals from Regression with ARIMA(2,1,0) errors

Q* = 1.869, df = 10, p-value = 0.9972

Model df: 2. Total lags used: 12

#We can notice that there is a presence of white noise. Since p-value is more than 0.05, we can say that null hypothesis can be rejected.



<u>Ex ante forecast</u>:

priceETS=ets(myts_p[,"Price"])

summary(priceETS)#It has picked up M,N,N model.

#ETS(M,N,N)

Then, after applying Ex-Ante forecast we can conclude that ETS model has picked M,N,N model with alpha=0.9999 as smoothing parameters.

We do dynamic forecasting,

#We can notice the drift and forecast is showing changes that doesn't show better results.

<u>We compared this with plain arima model</u>:

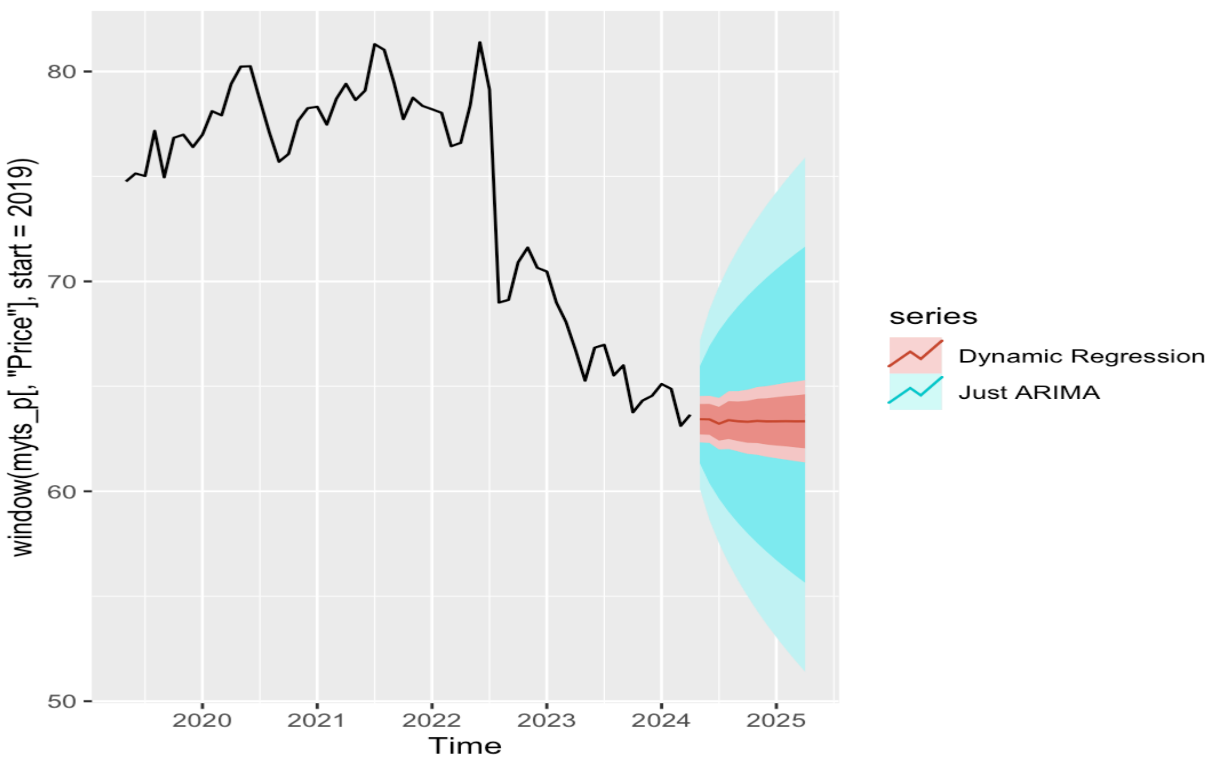
#comparing this to a plain ARIMA model

justArima = auto.arima(myts_p[,"Price"],d=1,D=0)

summary(justArima)#It picked (0,1,0) model

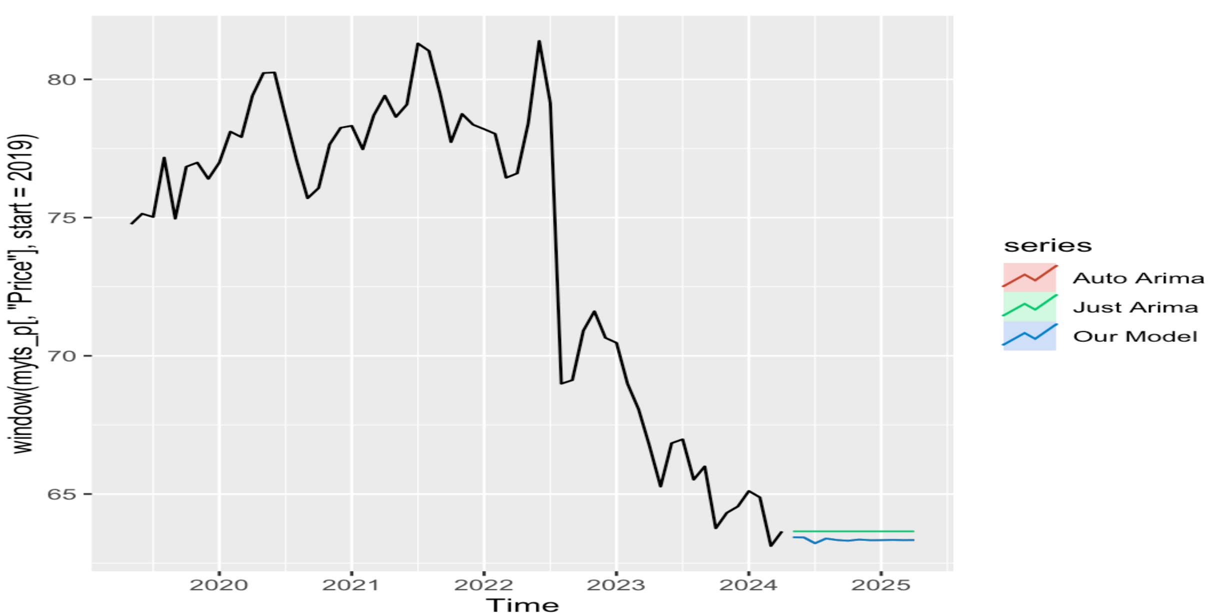
Plotting:

#We can see smooth drift and the forecast seems to be reasonable.



Forecasting Just ARIMA, AUTO ARIMA and our model:

#It cannot forecast ARIMA model but our model is slightly showing changes.



	Dynamic Regression Forecasting Model
ME	0.0254
MAE	0.3826
RMSE	0.5203
MPE	0.0269
MAPE	0.5152

<u>When considering for frequency=365 in daily data</u>:

#train/test split for daily data:

summary(myts_u[,"Price"])

length(myts_u[,"Price"])#1825

1825*0.8=1460

traindataAR12 = head(myts_u[,"Price"],1460)

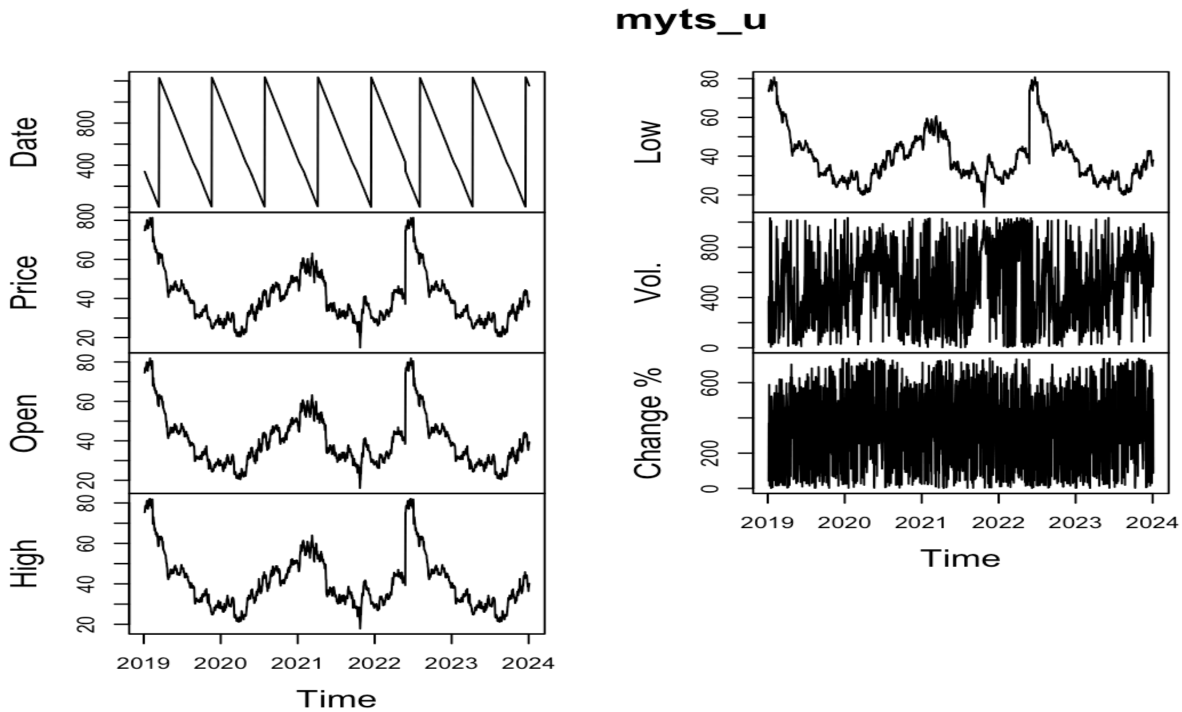
summary(traindataAR12)

testdataAR12 = tail(myts_u[,"Price"],365)

summary(testdataAR12)

myts_u =ts(UBER_Historical_Data, start=c(2019, 5), end=c(2024, 4), frequency=365)#daily data

myts_u

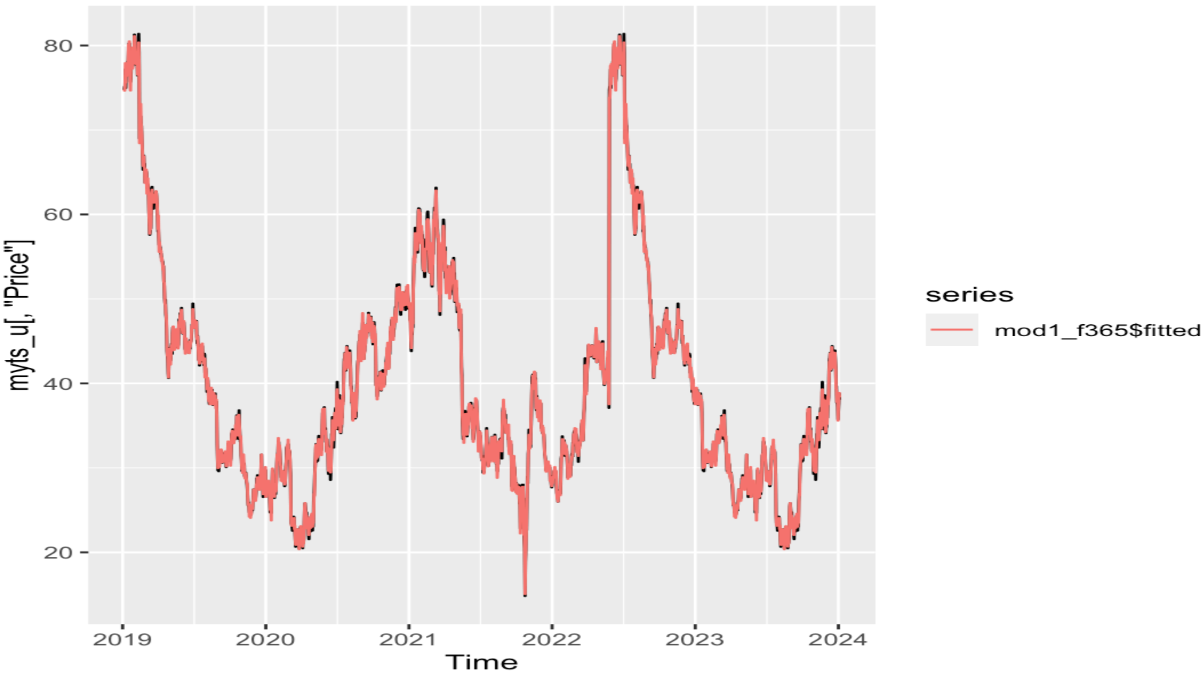


<u>Checking for the presence of unit root</u>:

After checking the presence of unit root for “price” variable, we apply one differencing and no seasonal differencing.

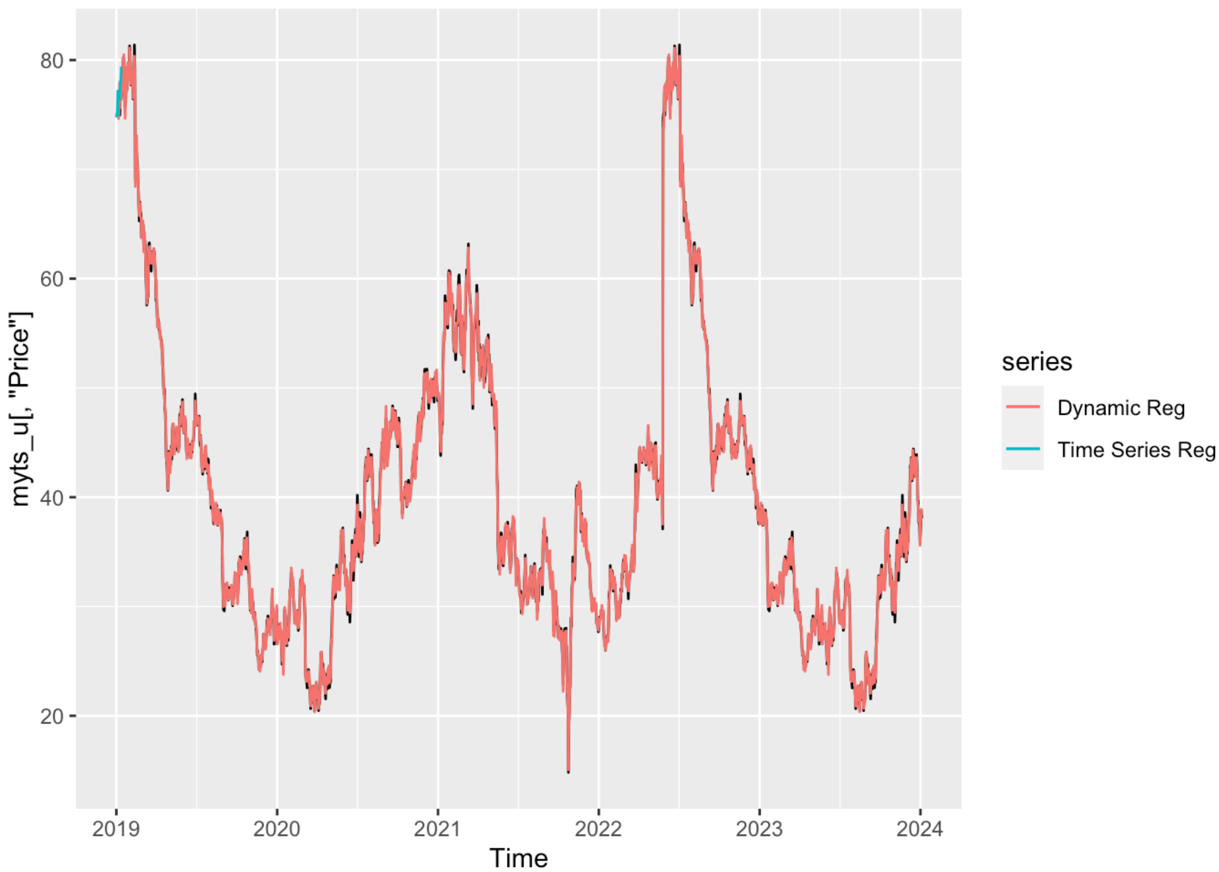
<u>Checking for models</u>:

#This seems to be fitted well.



<u>Comparing both dynamic regression model and time series model</u>:

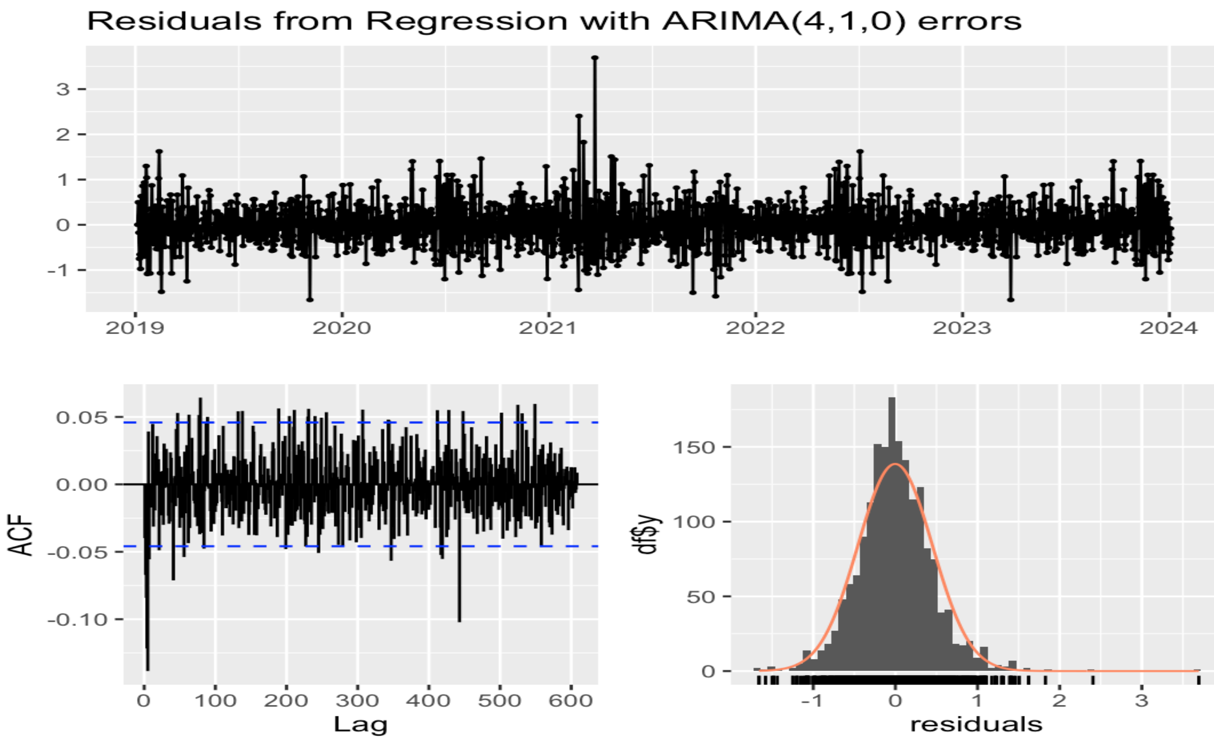
#There is a cyclicity in dynamic regression model whereas in time series regression, there is a slight rise with the initiation of upward trend after 2019.



Checking for white noise residuals:

checkresiduals(mod1_f365)

#Since p-value is less than 0.05, null hypothesis will be rejected with partial presence of white noise.



<u>Ex Ante forecasting</u>:

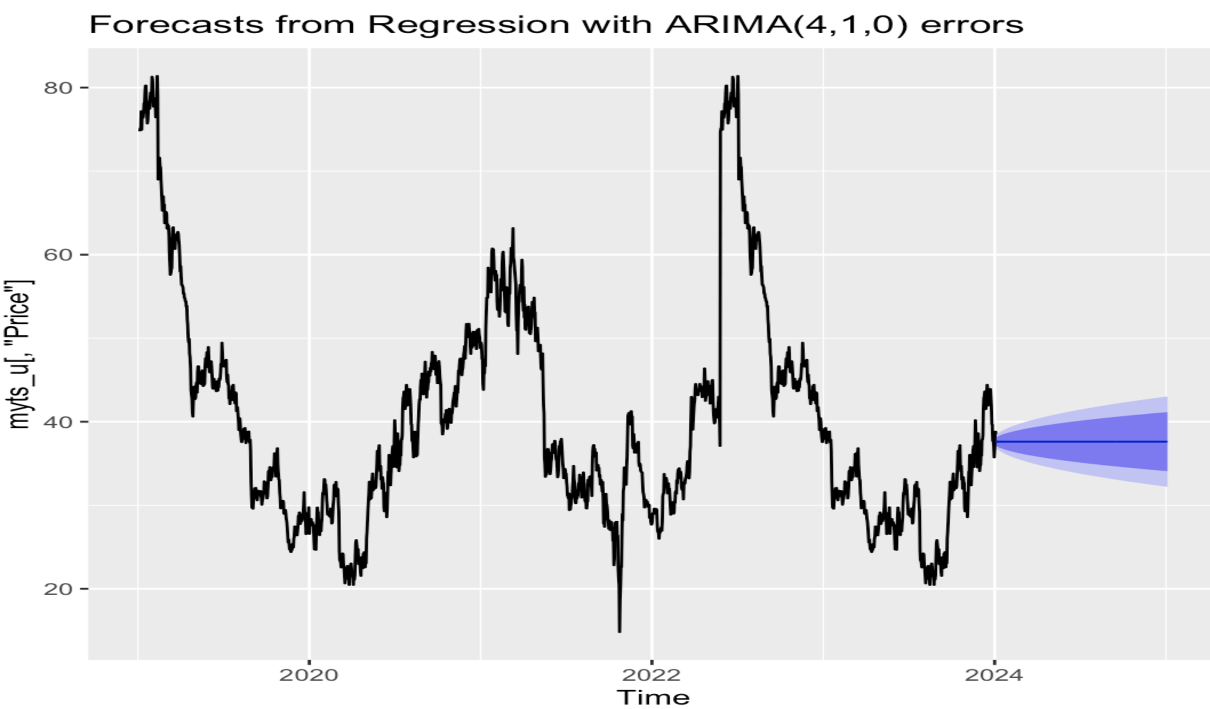
price1ETS=ets(myts_u[,"Price"])

summary(price1ETS)

#It can't handle data with frequency greater than 24.

<u>Dynamic forecasting</u>:

#We can notice the forecast to be slightly reasonable.



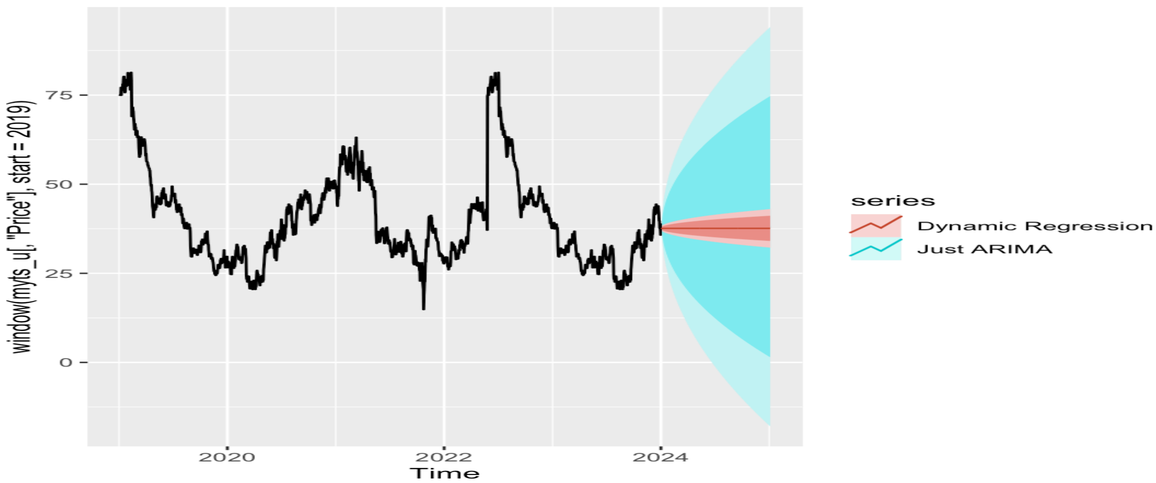
<u>Comparing this to a plain ARIMA model</u>:

justArima1 = auto.arima(myts_u[,"Price"],d=1,D=0)

summary(justArima1)#It picked (0,1,0) model

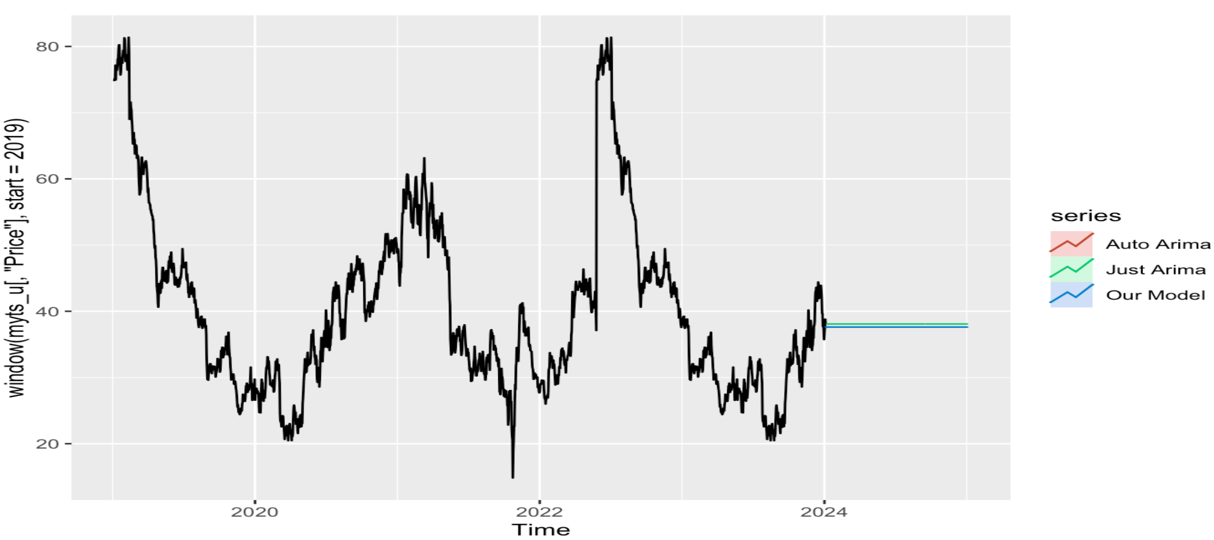
#AICc=6648.73

#We can notice that the forecast seems to be reasonable



<u>Forecasting Just ARIMA, AUTO ARIMA and our model</u>:

#We can notice that the model is slightly off with just ARIMA and our model in closer areas where we can't see AUTO ARIMA forecast.



	Dynamic Regression Forecasting Model
ME	-0.0027
MAE	0.3388
RMSE	0.4535
MPE	-0.0179
MAPE	0.8886

CONCLUSION:

Using the historical data from the dataset, we have predicted the price for the following upcoming year (May 2024 to May 2025).We used four alternative strategies to achieve this forecasting, and we discovered that the **Dynamic Regression** forecasting model outperformed the others.

Following accuracy testing using the above-mentioned test and training sets, the RMSE values for each model.

Daily Data:

MODEL	Accuracy test (RMSE)
Holt's Method	1.580192
ETS(Ex-Ante Forecasting)	1.008312
ARIMA	1.3029
Dynamic Regression	0.4535

Annual Data:

MODEL	Accuracy test (RMSE)
Holt's Method	1.741059
ETS (Ex-Ante Forecasting)	1.2192466
ARIMA	1.7881
Dynamic Regression	0.5203

CITATIONS:

https://data-flair.training/blogs/r-data-science-project-uber-data-analysis/

https://www.kaggle.com/code/prakharrathi25/uber-data-analysis-in-r

https://jespublication.com/upload/2021-V12I759.pdf