# Trend analysis

## Introduction

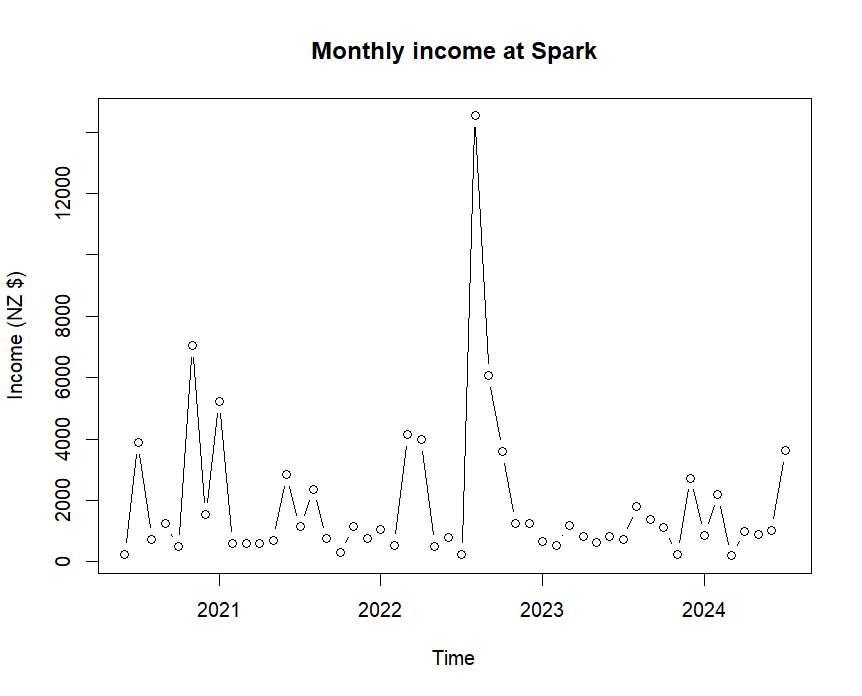
The objective of doing a trend analysis on the data provided by Spark & Flare was to assist in understanding the patterns in the events and income of the data. This can be used in prediction modeling and making decisions for future investments.

The trend analysis used historical income data, breaking it into training and validation sets to assist in forecasting. Several time series models were used to assist in the trend analysis: Linear, exponential, and quadratic models and ARIMA and SARIMA models.

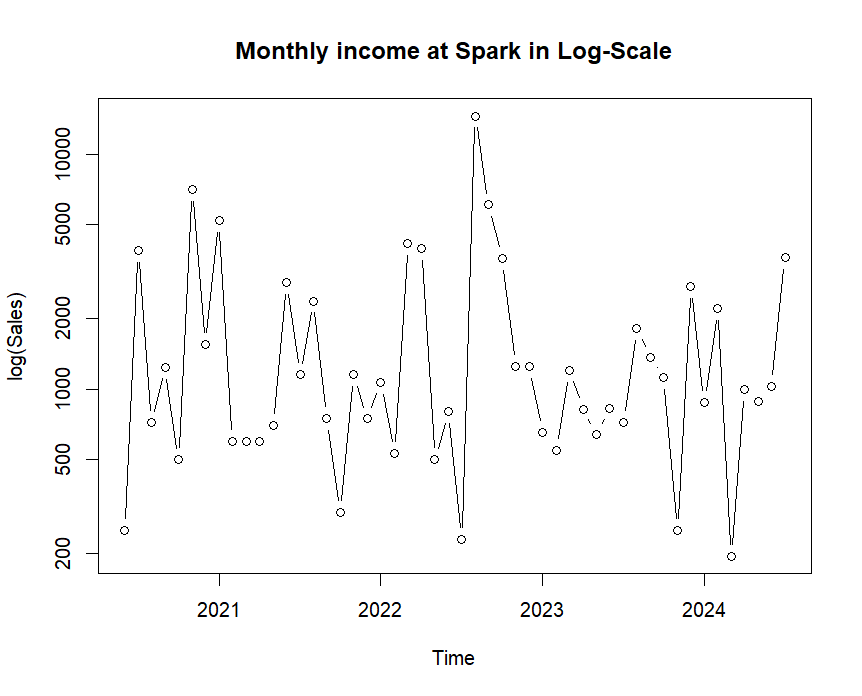
## Visualizing the pattern

A time series was plotted to visualize and see if there was any trend. There is a pattern in the data, which can be seen better when plotted with a log scale.

*Figure 1 Time series of Income at Spark & Flare*



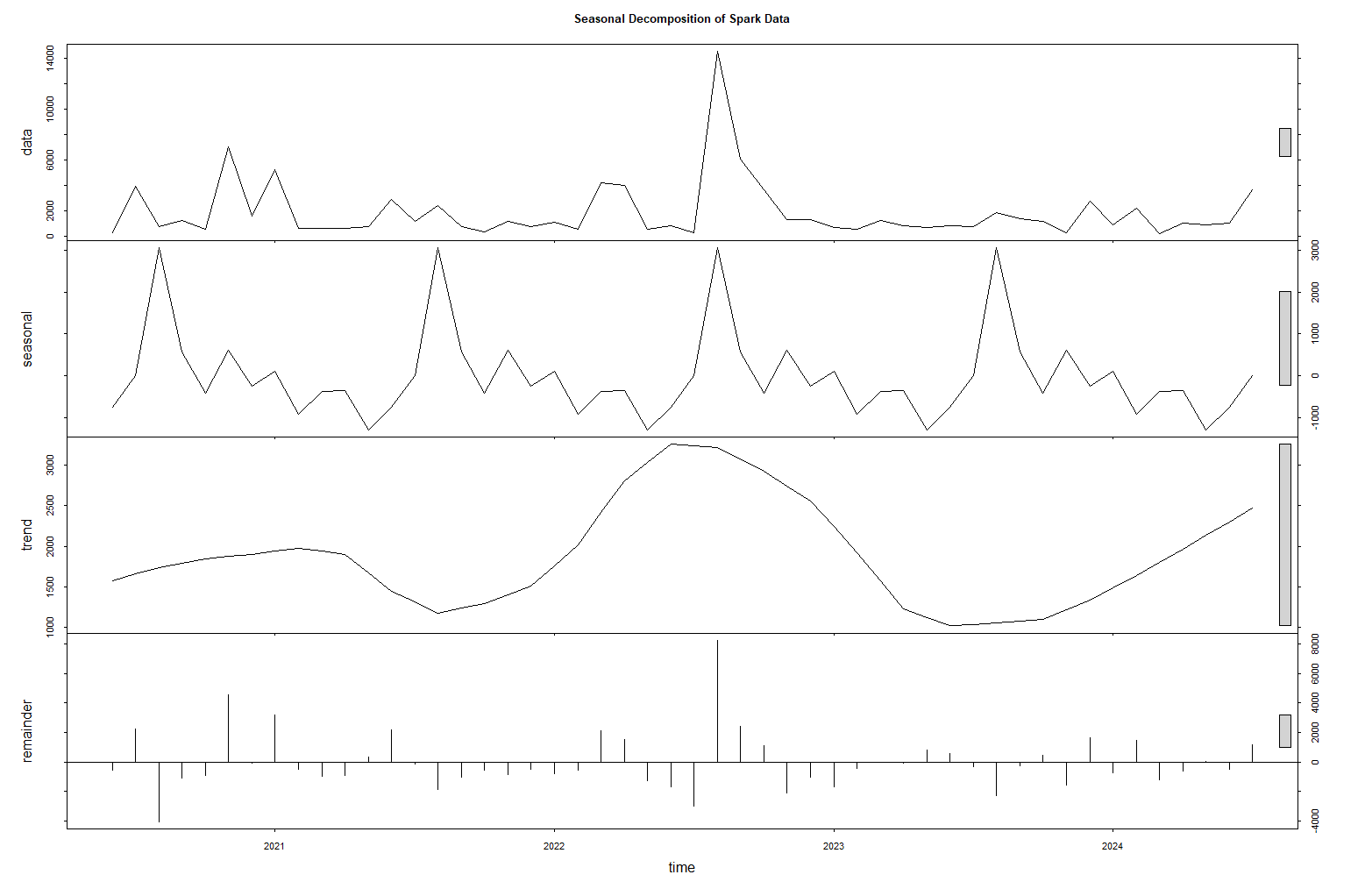
*Figure 2 Log-scale Time series of Income at Spark & Flare*



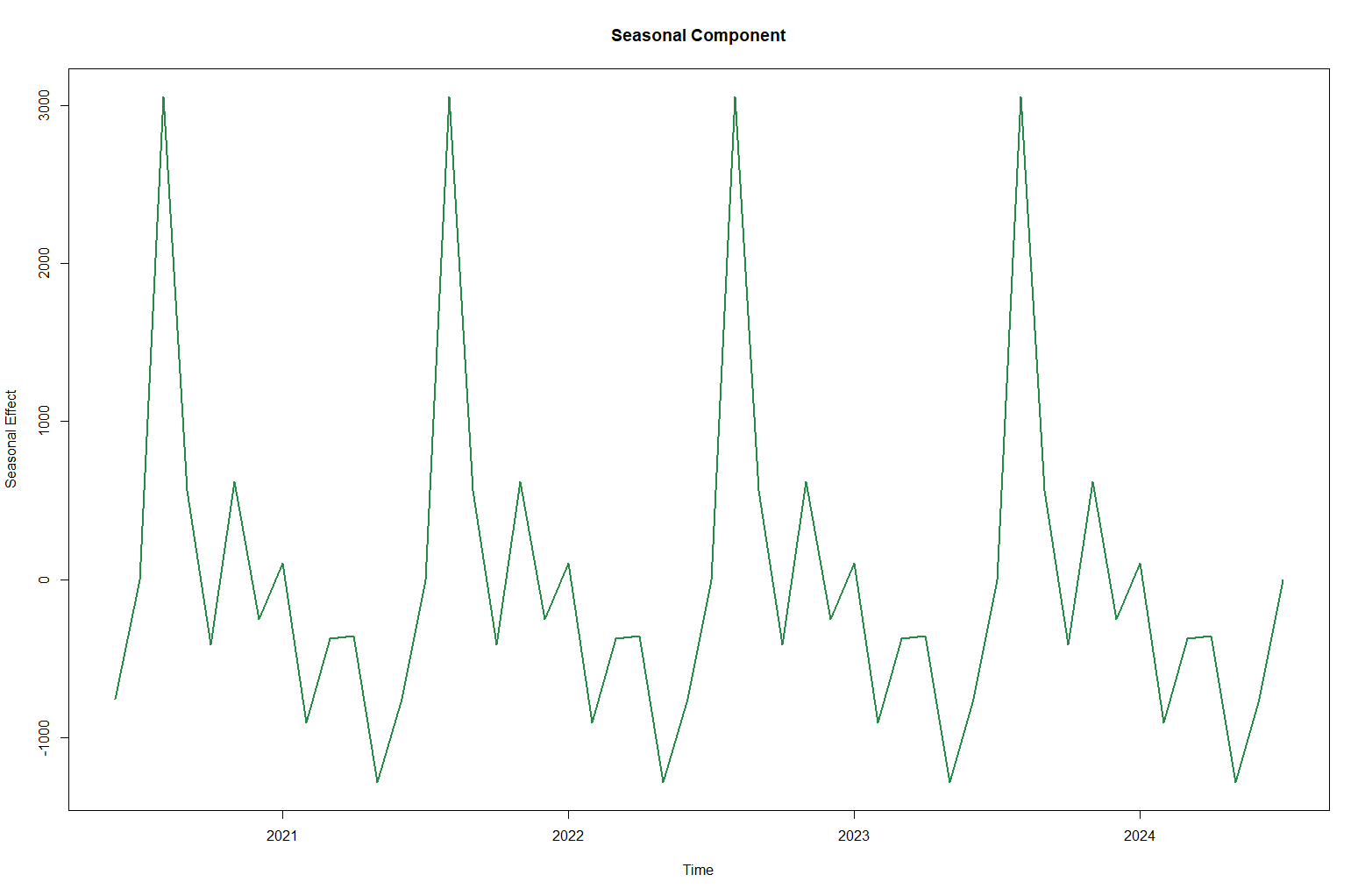
## Identifying the pattern

An STL (Seasonal-Trend decomposition using Loess) test was run to identify the pattern in the data. An STL helps break down the data into stationary components that make determining the pattern easier. It breaks it down into Trend, Seasonality, and remainder. The seasonal component captures any pattern that reoccurs every year. The Trend component shows where the data is increasing or decreasing over time and the remainder component represents irregularities or noise in the time series after the seasonal and trend components are removed.

*Figure 3 Seasonal Decomposition of Spark & Flare Income*

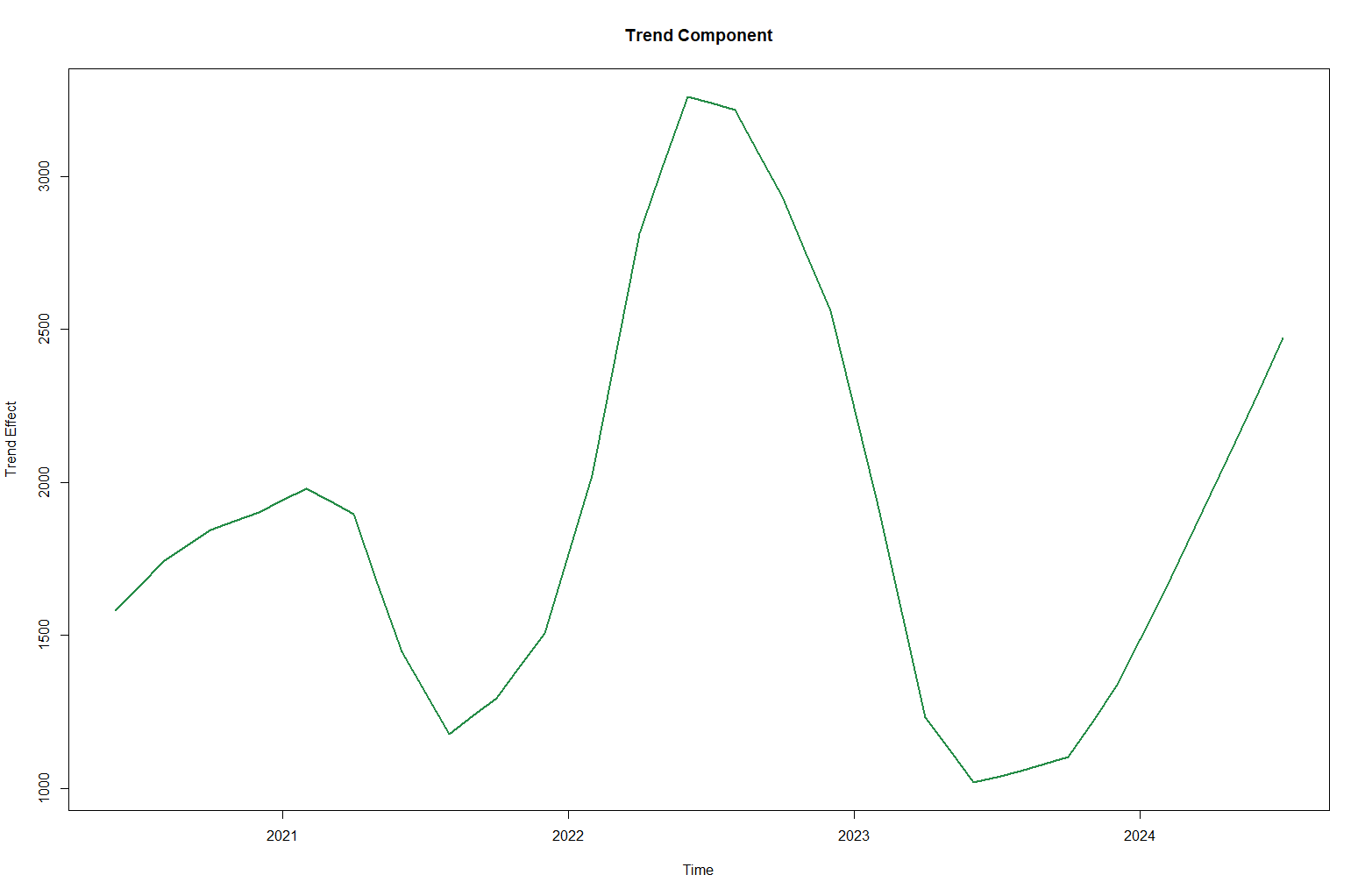


*Figure 4 Seasonal Component of Spark & Flare*



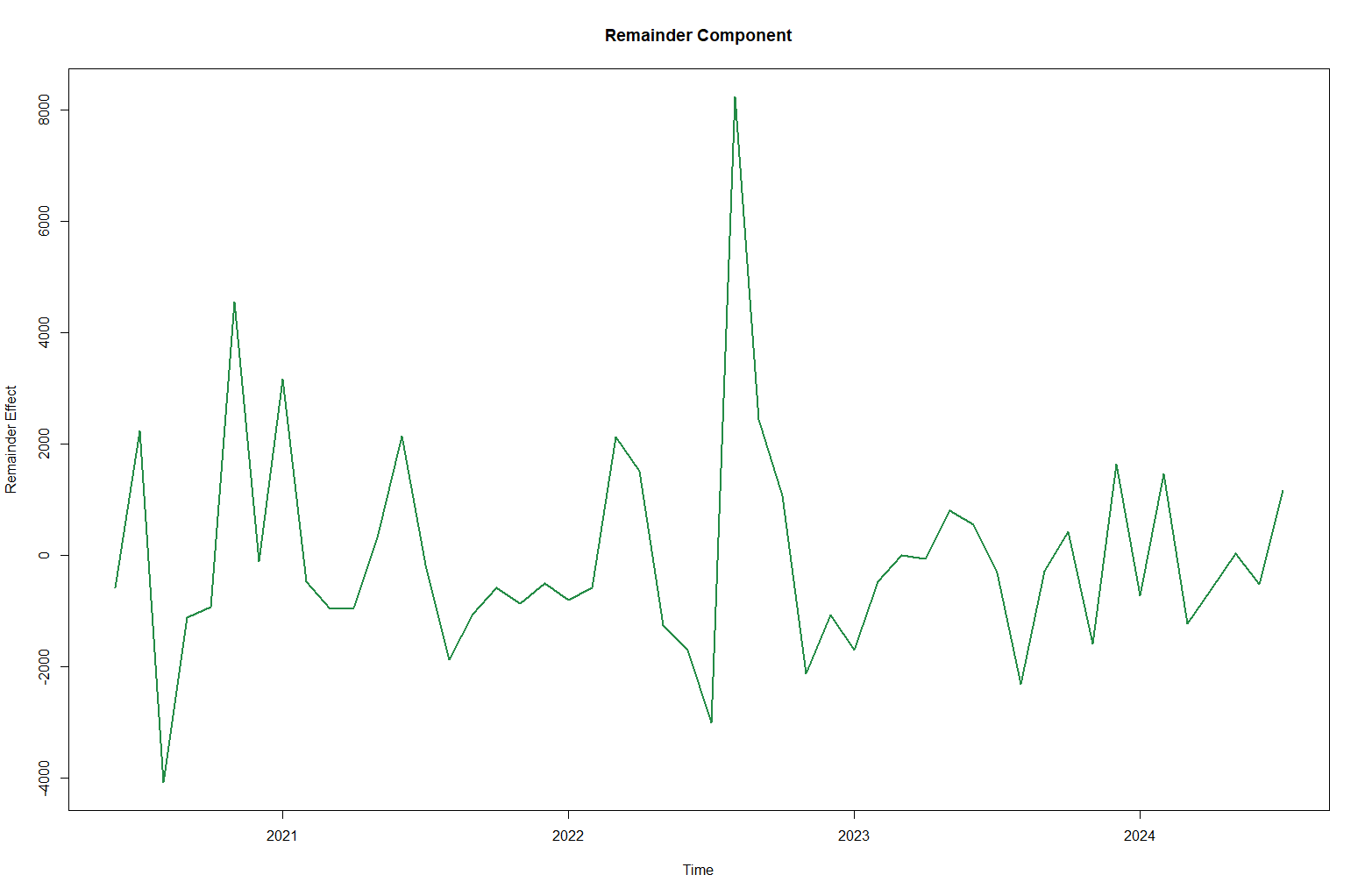
The seasonal component shows a clear seasonal trend in the data. There is a large peak at the end of the year (the data starts at month 6 in 2020). There is also a large dip just before this peak. This is consistent with what was found in the exploratory analysis. The exploratory analysis shows that there was a peak in Nov-Jan and the lowest income month is October in general.

*Figure 5 Trend component of Spark & Flare Income*



The trend component shows that the data dips and increases. There is a peak at the end of 2022/start of 2023, with a decrease in income on either side of it. It is however on the rise at the end of the data.

*Figure 6 Remainder Component of Spark & Flare Income*

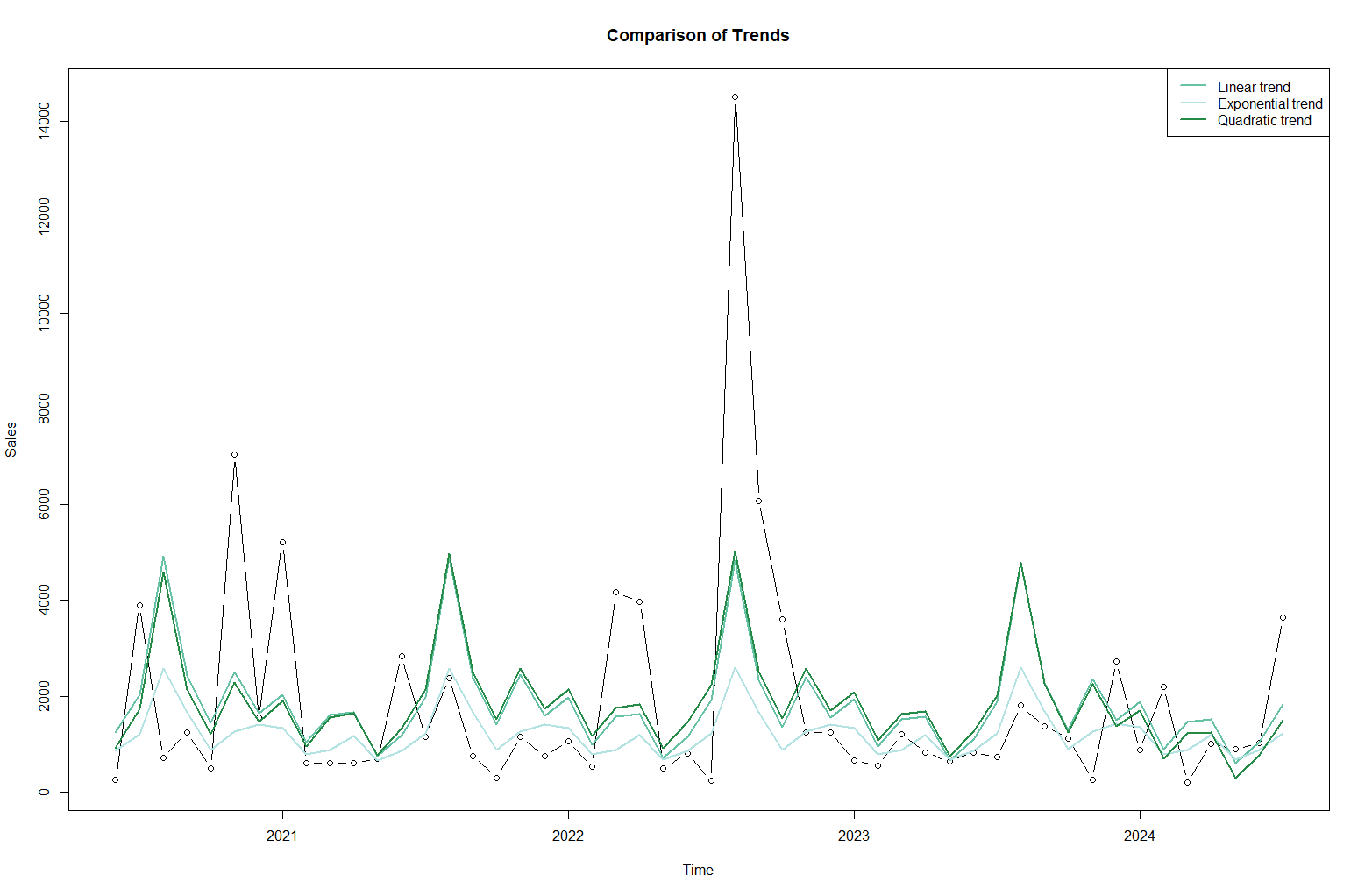


The remaining data shows some random noise but variability in the data is natural.

## Trend analysis, Linear, exponential, and Quadratic trend models

A linear, exponential, and Quadratic trend model was fit to the data.

*Figure 7 Comparison of Linear, Quadratic, and Exponential trends*



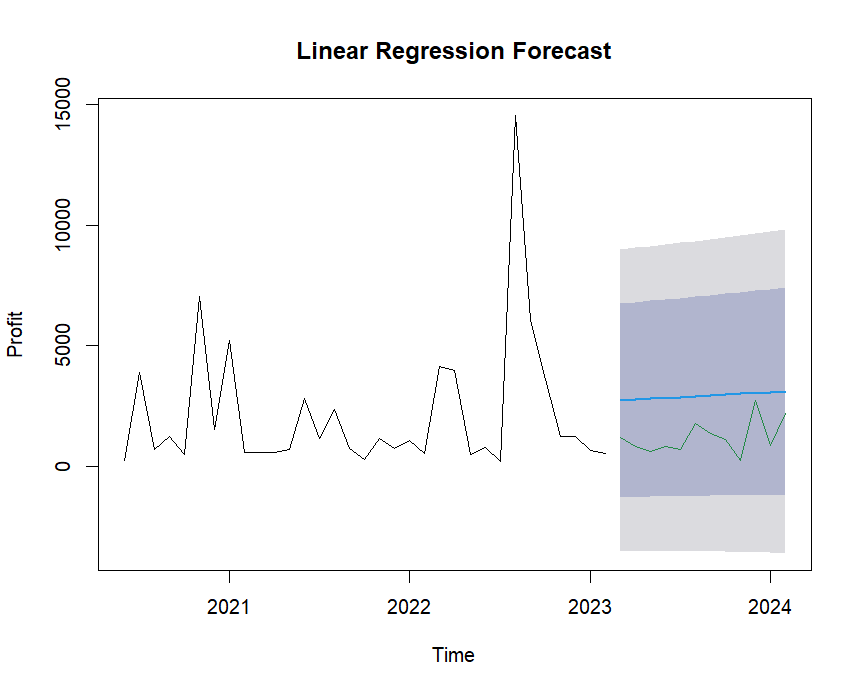
None of these models fit perfectly with the data. This shows there is a lot of variation and extreme values in the data. Based on the above plot we can see that the Quadratic model fits the best but still misses significant peaks and dips. This is likely due to the complexity of the data. Further analysis using advanced time series models should be done.

## Forecasting-Linear, exponential, and quadratic models

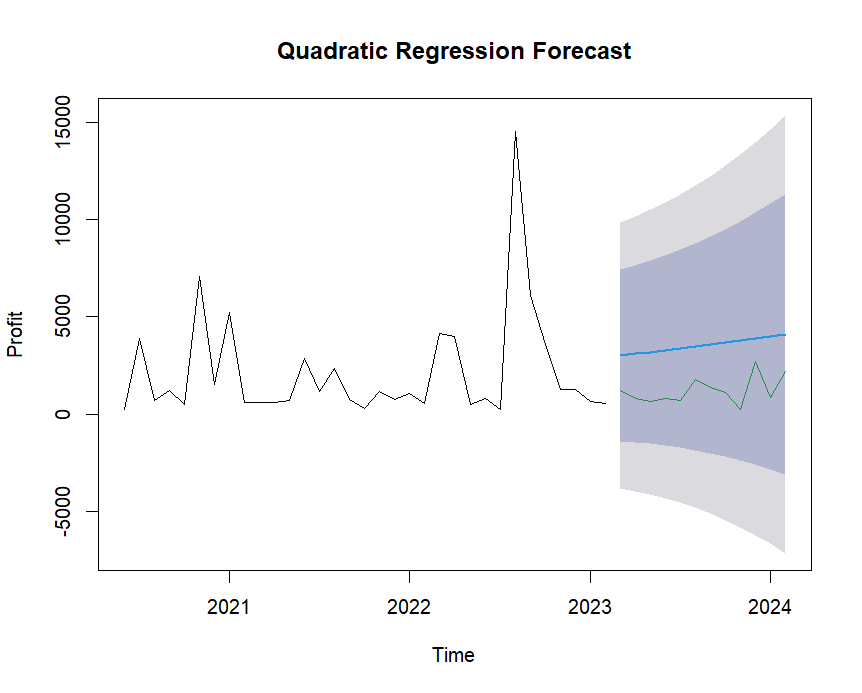
The data was then partitioned into two parts a training set and a validation set. The Validation set is the last 12 months of data, and the training set is the rest. The models were then trained on the training set and used to forecast predictions for the validation time frame.

When looking at the accuracy of the models based on the validation set the exponential model fits the best, and linear the worst. There is still room for improvement however as none of the models capture the seasonality peaks and dips of the data.

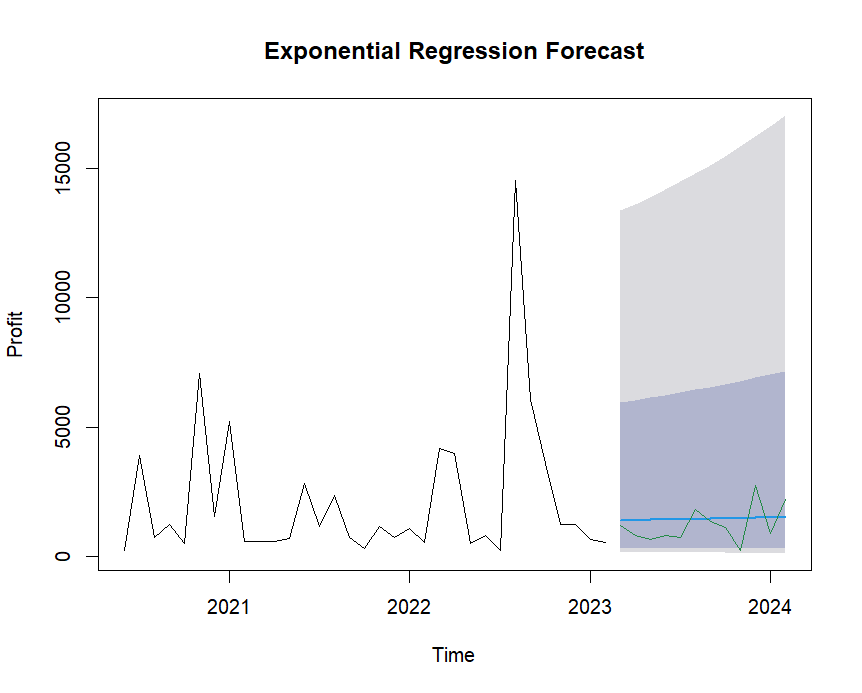
*Figure 8 Linear regression forecast*



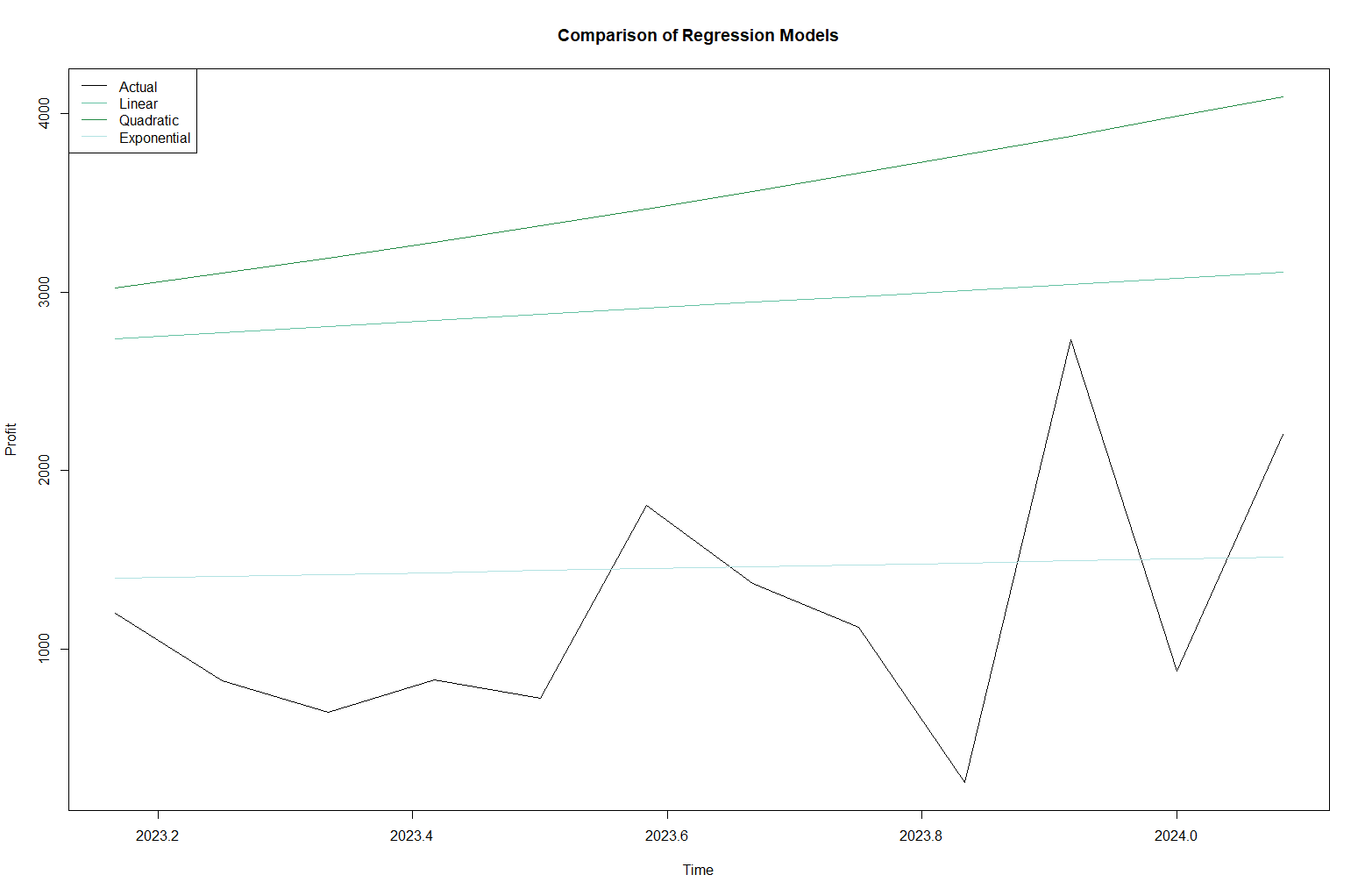
*Figure 9 Quadratic regression forecast*



*Figure 10 Exponential regression forecast*



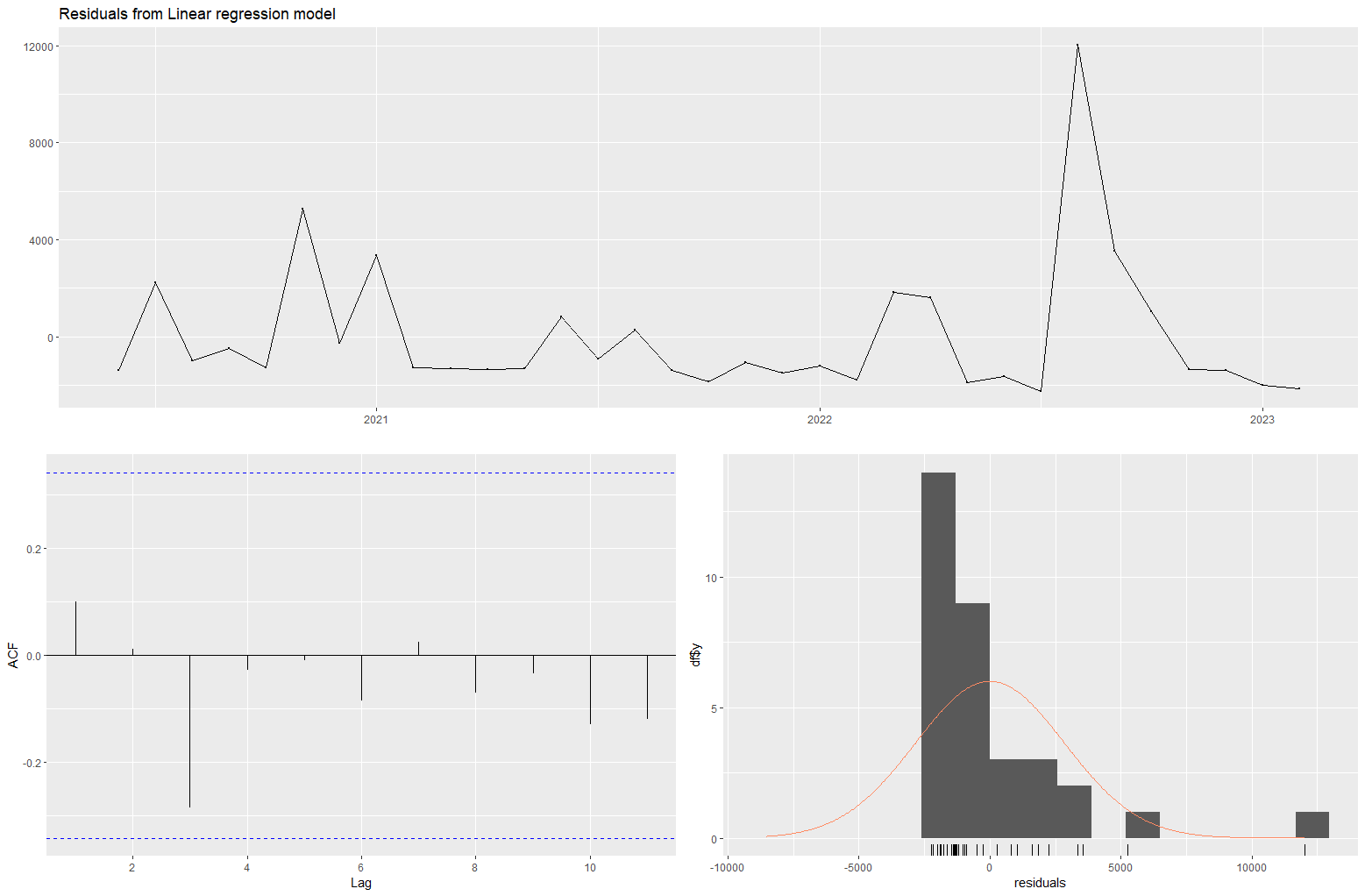
*Figure 11 Comparison of Regression models*



The residuals were then explored for the three models to confirm the results gathered above. A Breusch-Godfrey test was also run on the residuals to determine if the errors were independent over time or not.

## Linear model residual diagnostics

*Figure 12 Linear regression model residual diagnostics*

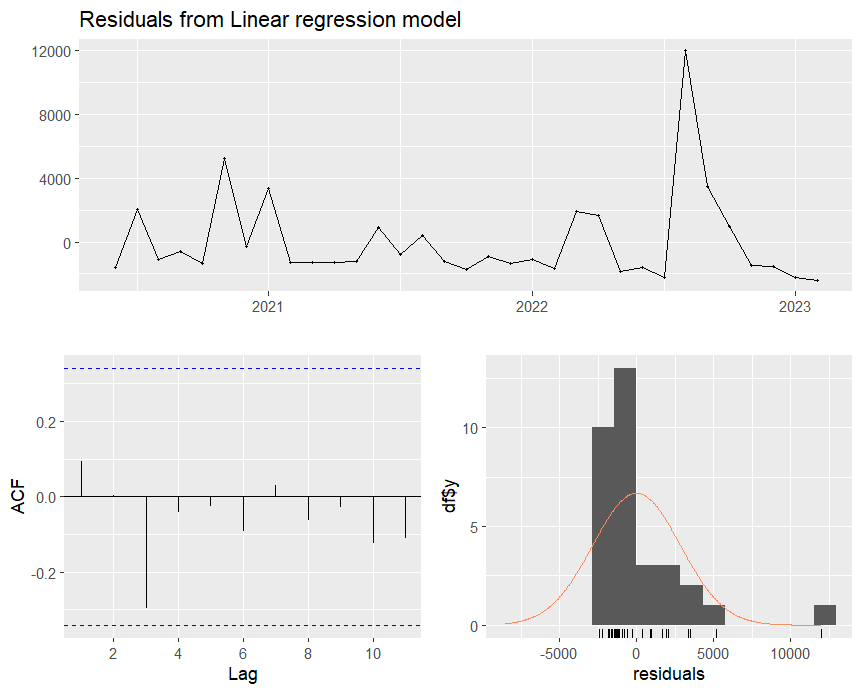


The top time series plot shows that the model doesn’t fully capture aspects of the data during the training period. If the model captured all the aspects of the data, the residuals should appear randomly and have no clear pattern. The ACF plot (bottom left) shows that the data is uncorrelated as all the data lies within the blue lines. There is a notable spike at lag 1 and 3 indicating that there could be a slight time-dependent structure to the data. The Histogram (bottom right) shows that data is slightly skewed to the right indicating the data is not normally distributed which can be an issue for linear models.

The Breusch-Godfrey test also shows that the residuals do not exhibit a significant correlation with a p-value >0.05. However, given the earlier analysis conducted shows a poor fit and high errors the result does not necessarily imply this model is well suited for forecasting this data.

## Quadratic model residual diagnosis

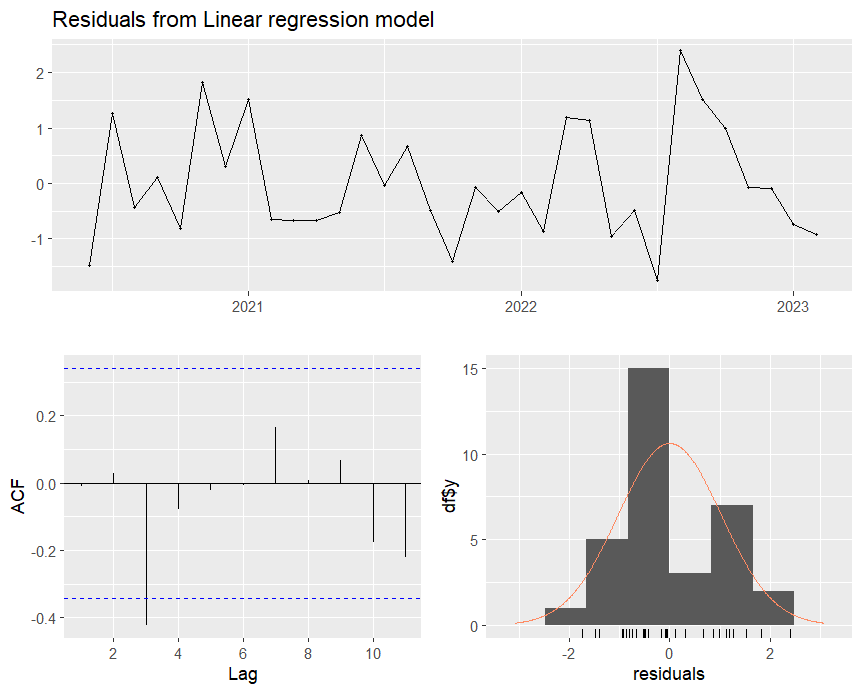
*Figure 13 Quadratic regression model residual diagnostics*



Similar to the Linear model the model doesn’t fully capture the aspects of the data. The ACF shows that the data is uncorrelated but there is still a spike at lag 1 and 3. The spikes are just as significant as the linear suggesting the model also doesn’t fit the data well. The data is still skewed to the right.

## Exponential model residual diagnosis

*Figure 14 Exponential regression model residual diagnostics*



This model, when looking at residuals, is the best fit compared to linear and quadratic models. It still doesn’t fit the data 100%, the time series plot is flatter than the other models but still has several peaks. The ACF plot has a point at lag 3 that goes past the blue line, highly suggesting a seasonal component that won’t be captured by the exponential model. This model has more normality than the other two shown by the histogram, which shows a better fit than the other two models.

Overall, the Exponential model fits the best compared to the Linear and Quadratic models based on the residuals. It still, however, doesn’t capture all the data and further analysis using more complex models like ARIMA and SARIMA models.

## ARIMA model

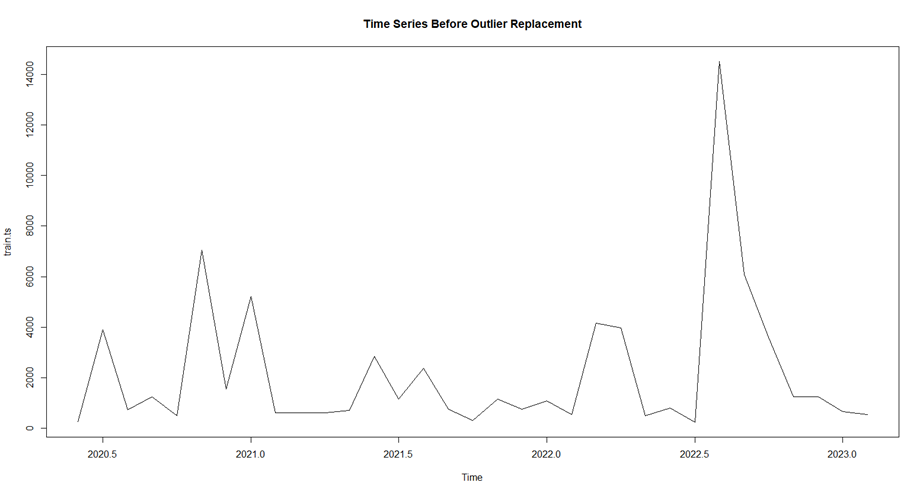
The auto ARIMA function in R studio was used to automatically assign the best Arima model to our data. Initially, this function determined the ARIMA (0,0,0) to be the best fit. This model is a model with a non-zero mean, which means the data gets modeled as a series of random fluctuations around a constant mean. The model was fit to both the training and validation set of data. In the below table, we can see the evaluation values to see how well it fits before outlier removal and after-outlier removal.

*Table 1 ARIMA Evaluation values, BOR(before outlier removal), AOR( After outlier removal)*

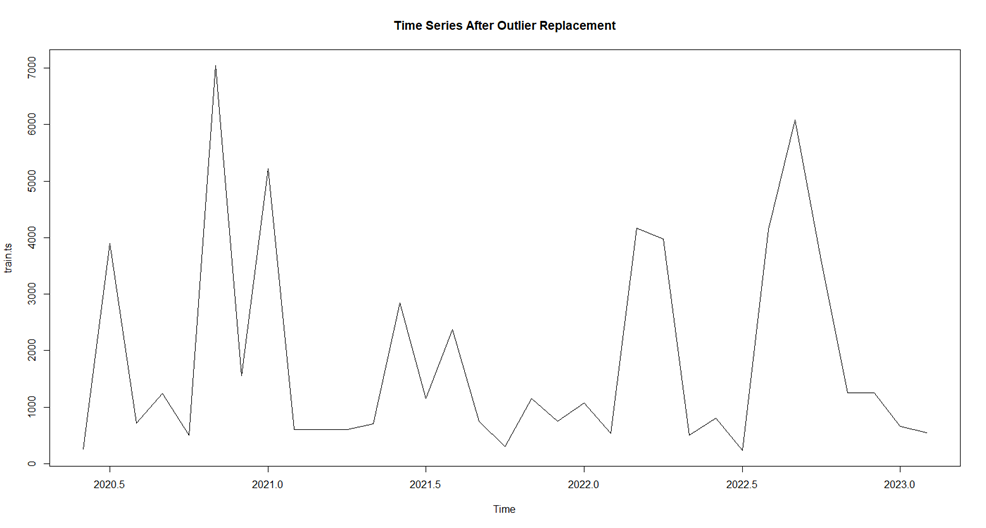
|  | ME | RMSE | MAE | MPE | MAPE | MASE | ACF1 | Theil's U |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ARIMA Training set BOR | -1.329832e-12 | 2816.288 | 1944.316 | -167.13 | 196.69 | 0.808 | 0.111 | NA |
| ARIMA Test set BOR | -9.506078e+02 | 1169.582 | 1051.021 | -158.61 | 162.34 | 0.436 | -0.321 | 0.452 |
| ARIMA Training set AOR | -6.821210e-13 | 1822.237 | 1505.573 | -129.25 | 161.39 | 0.787 | 0.0586 | NA |
| ARIMA test set AOR | -6.358573e+02 | 931.977 | 841.187 | -121.00 | 129.04 | 0.440 | -0.322 | 0.463 |

We can see the RMSE and MAE are lower in the validation sets vs the training sets indicating the model fits somewhat well. The MAPE is still significantly high indicating there are significant errors, but it does lower after the outliers are removed indicating the outliers had a significant impact on the model fit which can be seen in the below figures.

*Figure 15 Time series of ARIMA Model before outlier replacement.*

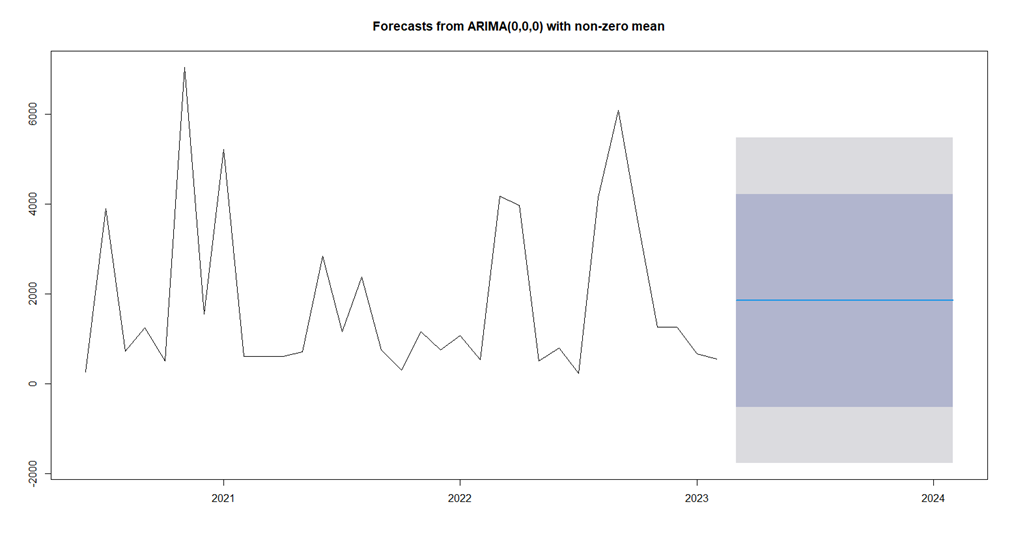
**

*Figure 16 Time series of ARIMA Model After outlier replacement.*



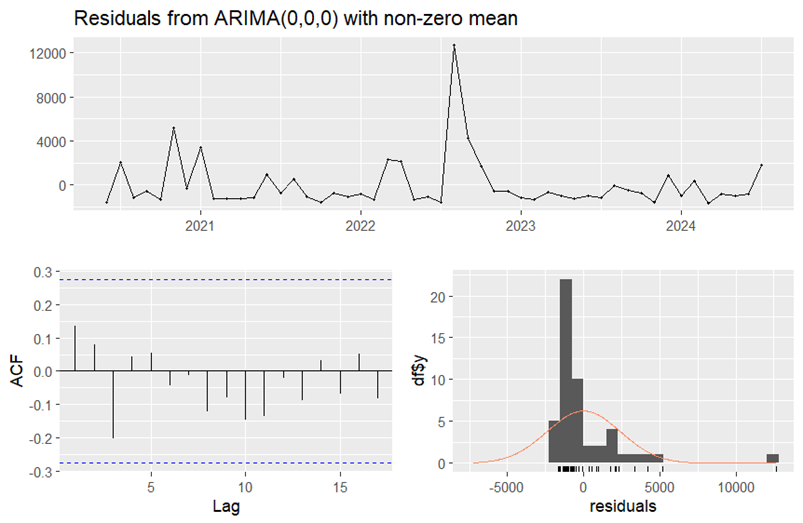
You can see however attempting to do a forecast there are still errors

*Figure 17 Forecast ARIMA (0,0,0)*

**

## ARIMA modal residual diagnosis

In the time series plot the data seems to have some large spikes indicating that some predictions may have significant errors. The ACF plot shows that there is most lags are within the blue line, indicating that they are uncorrelated. This suggests that most of the correlation has been accounted for in this model. The histogram shows that the data is still skewed to the right and not necessarily normally distributed.



## ARIMA Model Stationary tests

An augmented dickey test was conducted to determine if the data is Stationary or not, this is to assist in producing a more accurate model as many models assume the underlying data is stationary. The ADF test had a result of a P-value of 0.1725, which means we can reject the null hypothesis which is that the data is stationary.

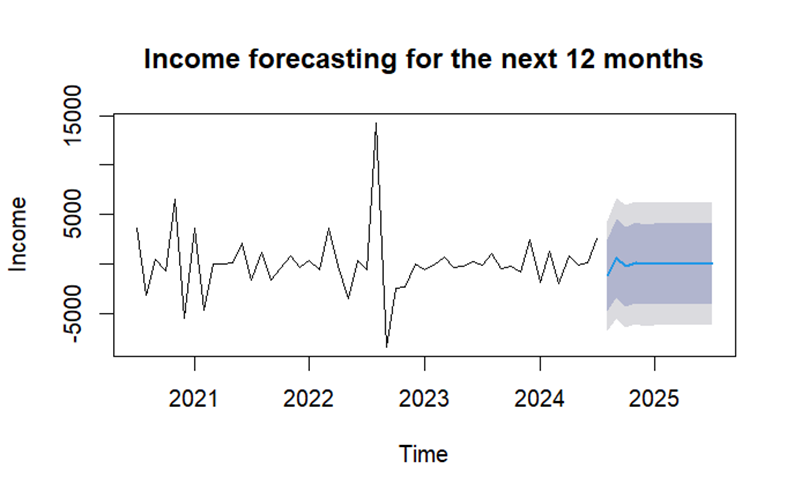
After first-order differencing was done on the data to make it stationary the p-value dropped to 0.01 indicating we can now accept the null hypothesis. Other transformations were made to make the data stationary. A Log transformation and box-cox transformation had a p-value of less than 0.05 meaning we can accept the null hypothesis. A Square root transformation was also done but the p-value was above 0.05 thus we can reject the null hypothesis. The models were refitted with the transformations that made the data stationary.

## ARIMA after first order differencing

The Auto Arima function determined that the best fit would be ARIMA (1,0,0) with zero mean. This is different from before the data was transformed. This model has better MAE and MAPE than the original ARIMA model, which suggests it has a better predictive accuracy than before the first-order transformation was done. However, the RMSE and Theil’s U metrics became slightly worse which means that while most metrics improved the overall error and model efficiency didn’t.

|  | ME | RMSE | MAE | MPE | MAPE | MASE | ACF1 | Theil's U |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Training set | 3.091344 | 3376.139 | 2066.563 | -Inf | Inf | 0.6110774 | -0.048986 | NA |
| Test set | 1210.93511 | 1389.406 | 1210.935 | 99.7858 | 99.7858 | 0.3580703 | -0.321238 | 1.144644 |

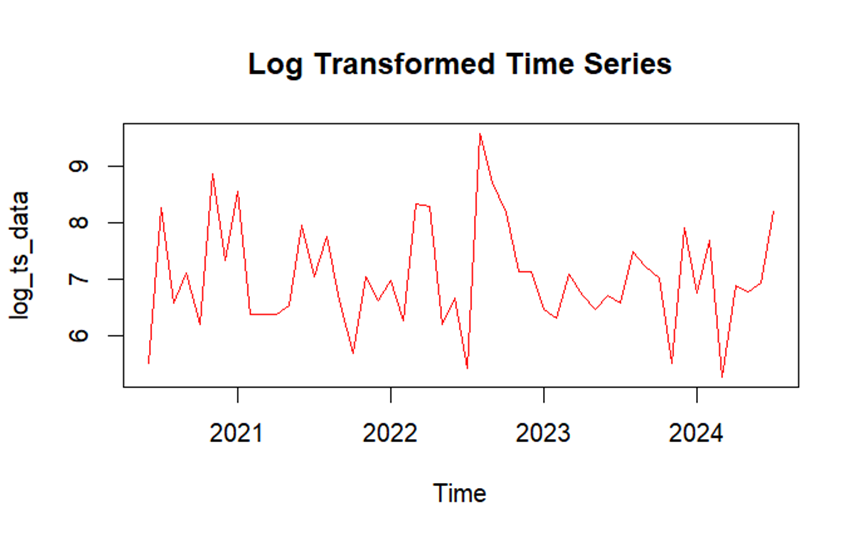
*Table 2 ARIMA First-order transformation evaluation metricsFigure 19 ARIMA first order differencing transformation prediction modeling*



## ARIMA Log transformation

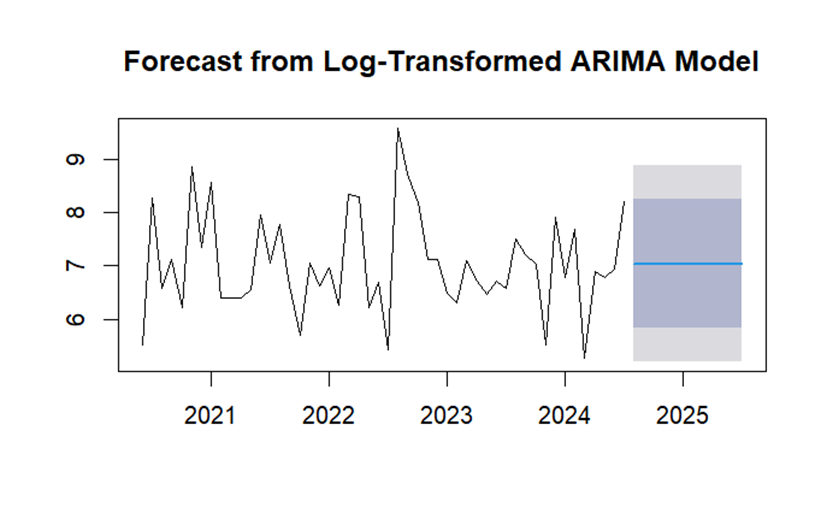
The auto Arima function determined the ARIMA(0,0,0) with zero mean was still the best fit. This transformation did improve the model’s fit on the training data it still struggles with the validation data, which can be seen by the high MAPE and RMSE values. The model may be overfitting the training data but isn’t capturing the underlying patterns in the validation data effectively.

*Figure 20 Log transformed time series*



*Table 3 ARIMA Log transformation evaluation metrics*

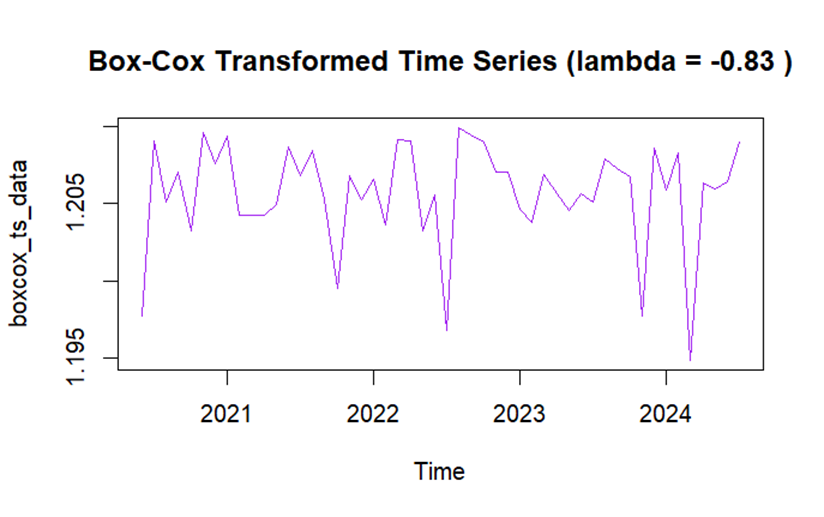
|  | ME | RMSE | MAE | MPE | MAPE | MASE | ACF1 | Theil's U |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Training set | -6.351835e-15 | 1.0132 | 0.8283252 | -1.965556 | 11.63409 | 0.7056556 | -0.008508 | NA |
| Test set | 1.206539e+03 | 1385.642 | 1206.5395 | 99.150041 | 99.15004 | 1027.8587 | -0.321858 | 1.1412 |



## ARIMA Box-cox transformation

The auto Arima model fitted an ARIMA (0,0,0) model to the data. The Box-coz transformation improved the model’s fit, with the RMSE and MAE being lower than the earlier models. The MAPE still indicated some percentage error as it is still high on the validation set. The negative MPE suggests a tendency for the model to underestimate the predictions but is it still more reliable than previous models.

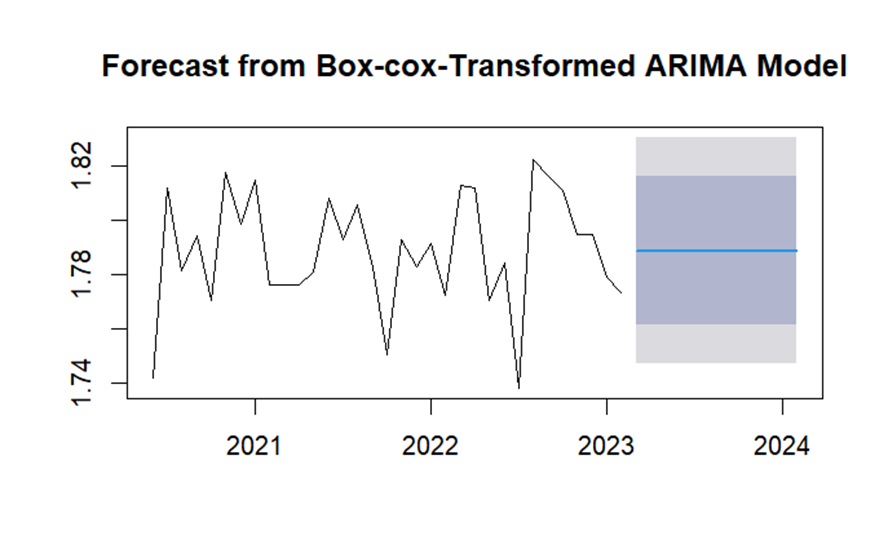
*Figure 21 Box-Cox transformed Time series.*

**

*Table 4 Box-Coz transformation evaluation metrics*

|  | ME | RMSE | MAE | MPE | MAPE | MASE | ACF1 | Theil's U |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Training set | 2.085878e-16 | 0.0210 | 0.01716 | -0.0139 | 0.9617 | 0.7092258 | -0.08681723 | NA |
| Test set | 257.9162 | 728.5521 | 523.9792 | -14.2042 | 53.18741 |  | -0.3218578 | 0.7229215 |

*Figure 22 Box-Cox transformation forecast*

**

## SARIMA Model

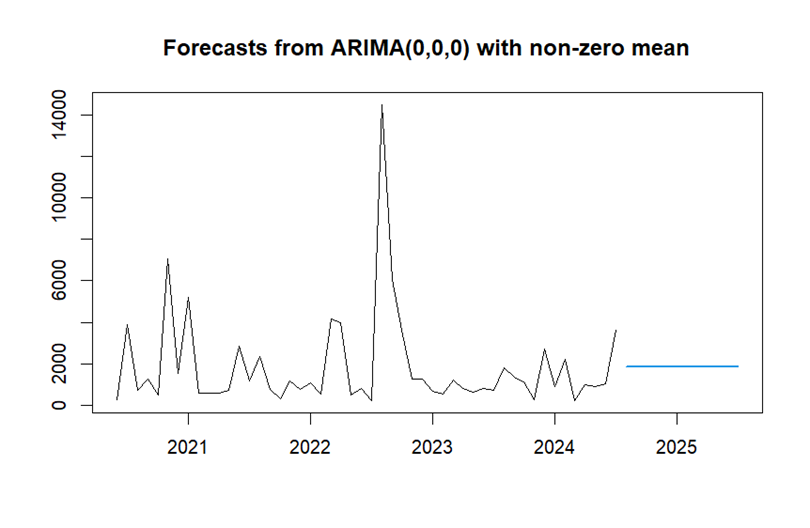
The SARIMA auto ARIMA function also initially chose the ARIMA (0,0,0) model, this indicates the function thinks that the model still fits well even when considering seasonality.

*Table 2 SARIMA Evaluation metrics*

|  | ME | RMSE | MAE | MPE | MAPE | MASE | ACF1 | Theil's U |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| SARIMA Training set | -6.821210e-13 | 1822.2375 | 1505.573 | -129.25 | 161.3912 | 0.7879 | 0.0586 | NA |
| SARIMA Test set | -6.358573e+02 | 931.9774 | 841.1873 | -121.00 | 129.0390 | 0.440 | -0.3218 | 0.463 |

There is some improvement based on the RMSE and MAE metrics compared to the ARIMA Model. It isn’t large however but still somewhat significant. It also doesn’t seem to predict values well.

*Figure 18 SARIMA Model forecast*

**

## Exponential smoothing

An Exponential smoothing model was also then fitted to compare against ARIMA and SARIMA Models. This is a broadly accurate model when forecasting short term.

*Table 3 Exponential smoothing evaluation metrics*

|  | ME | RMSE | MAE | MPE | MAPE | MASE | ACF1 |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ETS | -7.273071e-14 | 2381.987 | 1527.332 | -135.8148 | 162.13 | 0.674099 | 0.1361322 |

When comparing the metrics with ARIMA and SARIMA models the exponential smoothing model has a higher RMSE, MAE and MAPE, making it a less suitable model for this data set. The higher AC1 also indicates the model struggles to identify and capture underlining patterns in the data.

## Conclusion

The best fitting model is the Box-cox transformation based on the error metrics. It has the lowest validation metrics and has the best residual behavior, which indicates that it captures the underlying patterns in the time series data. The data needed to be transformed to improve handling of the non-stationarity and variance of the data.