Project Report

Title: "*Store Sales Data Analysis*"

DSC-425

Time Series Analysis and Forecasting

Group Members

Fauzia Khan

Maheen Adeeb

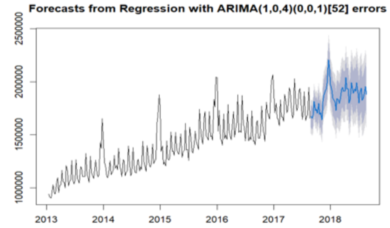
Sumera Fatima

Sai Chaitanya

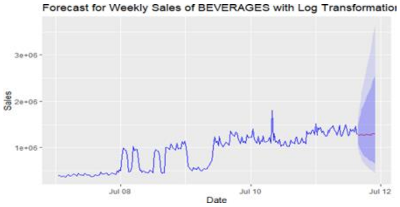
1. Non-technical Summary

This store sales dataset is taken from a grocery store in Ecuador spanning from 2013 to 2017. The primary goal was to forecast future sales using time series analysis techniques. We explored the impact of promotions on sales and identified top-performing stores and product categories. This analysis is vital for stakeholders to make informed decisions regarding inventory management and promotional strategies.

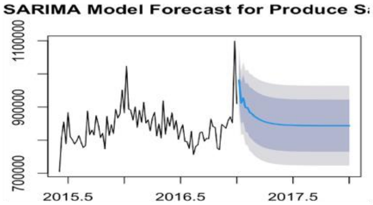
Accurate forecasting helps retailers ensure they have the right products at the right time, pleasing customers. If they underestimate demand, popular items can quickly sell out, leading to lost revenue and upset customers. For grocery stores, accurate forecasting reduces food waste from overstocking and improves customer satisfaction.



The top three product family of highest sales are groceries, beverages, and produce. We constructed different models to predict the unit sales of these products. For groceries (which consists of the highest sales among all the other product family), predicts a seasonal increase in sales for 2018. We can also expect higher sales at the beginning of 2018, similar to the patterns observed at the end of each year starting from 2013. However, the model shows that at the end of 2017, there is a high chance that sales will remain steady or experience slight fluctuations until 2018, possibly due to consumer purchasing power. After 2018, the forecast indicates that sales will continue to increase and follow an upward trend. The confidence bands suggest a high probability that the sales series will either remain steady or follow an upward trend for every month after 2018.



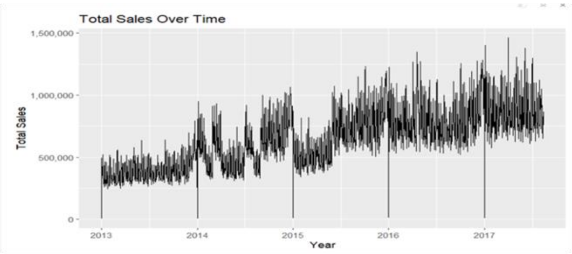
In addition to this, if we take a look at the forecasted beverage sales series, we can see that it has significantly captured a steady trend starting approximately from July 2010. This suggests that the sales series might remain steady, but the confidence bands indicate that there is an 80% chance that beverage sales will increase. However, there is always a possibility that sales might drop due to external factors and causes. In contrast, if we look at the produce sales than it is not very well giving very useful predictions as mostly are shows that the sales will decrease over time.



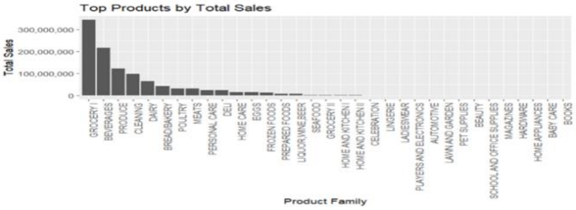
Furthermore, our analysis also shows that groceries, beverages, and produce sales are interconnected over time. This means that when grocery sales go up, beverages sales tend to go up as well. On the other hand, when one goes up, the other tends to go down. The same patterns can be seen between grocery and produce sales, and beverages and produce sales. These insights are important for businesses because they help in understanding how different products interact. This understanding can improve how businesses manage their inventory, plan their marketing strategies, and predict future sales trends effectively.

2. Technical Summary

a. Exploratory Data Analysis (EDA)



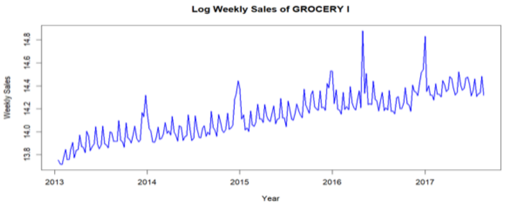
Total sales reveal a clear upward trend, indicating consistent growth in sales volume over this period. Noticeable seasonal variations suggest regular fluctuations likely due to periodic events, holidays, or promotional periods. From 2015 onward, there is an increase in the amplitude of these fluctuations, indicating growing variability in sales. Despite these fluctuations and increasing variability, the overall trend remains positive, highlighting robust growth in total sales throughout the observed period.



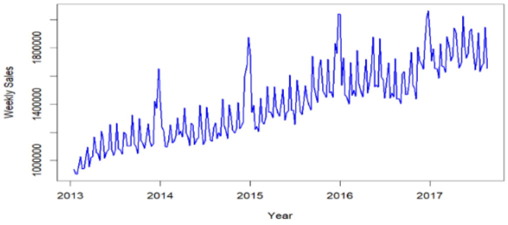
Grocery 1, Beverages, and Produce are the top performing product groups, each contributing significantly to overall sales.

b. Model Fitting

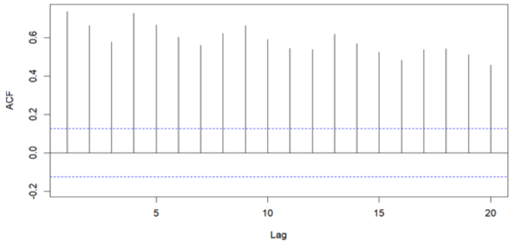
❖ Grocery Analysis:



The original weekly grocery series has some high peaks from 2016 and 2017 and in 2017. We tried to remove the outliers from the weekly grocery sales. It is evident that after removing these extreme data points, the series tends to be clearer in visualizing seasonal patterns and regular fluctuations over the years and across the months.

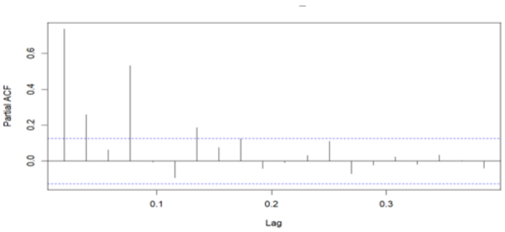


Overall, we see an upward trend, trend, although there is a small downward trend from 2016 to 2017. After this period, the series resumed its upward trend during 2017. The series is not multiplicative hence transformation is not needed.



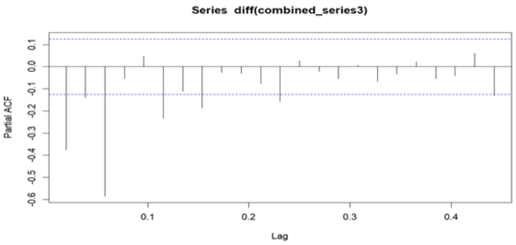
The ACF plot of this series shows significant moving average (MA) behavior, along with evident monthly seasonality. However, the series seems to be non-stationary which requires further deep analysis. However, we observe distinct peaks at regular intervals, indicating a strong cyclical pattern.

The PACF plot of the series reveals significant autoregressive (AR) behavior. The initial lags showing significant partial autocorrelations a lag 1, 4, and so on.

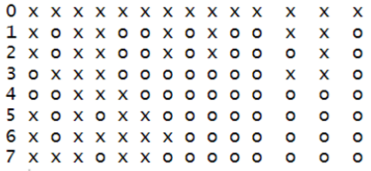


ADF AND KPSS TESTS Analysis: The Augmented Dickey-Fuller (ADF) test result for the grocery sales series indicates a Dickey-Fuller statistic of -5.4021 with a p-value of 0.01, suggesting that the series is stationary since the test rejects the null hypothesis of a unit root.

Conversely, the KPSS (Kwiatkowski-Phillips-Schmidt-Shin) test results show a KPSS statistic of 4.4142 with a p-value of 0.01, indicating that the series is non-stationary since the test rejects the null hypothesis of level stationarity.



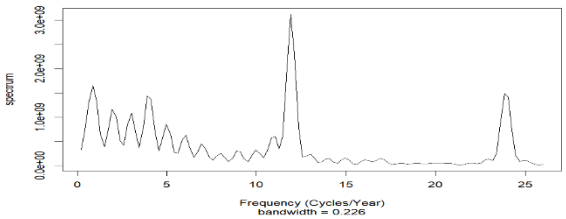
This discrepancy in ADF and KPSS made us dive deeper, possibly involving differencing the series. As we can visualize these lags are reverted which is the sign of over-differencing which can affect the model accuracy. This is why we concluded that the series is stationary, and we extended our analysis with EACF plot.



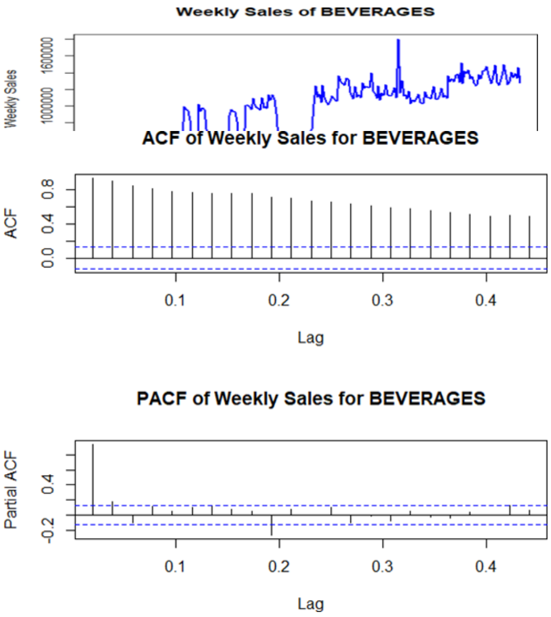
```
0 x x x x x x x x x x x   x   x   x
1 x o x x o o x o x o o   x   x   o
2 x o x x o o x o x o o   o   x   o
3 o x x x o o o o o o o   x   x   o
4 o o x x x o o o o o o   o   o   o
5 x o x o x x o o o o o   o   o   o
6 x o x x x x x o o o o   o   o   o
7 x x x o x x o o o o o   o   o   o
```

By looking at the EACF, it shows that the series is in fact stationary but since we have performed unit root test above the series is however, stationary. In addition, from EACF plot we choose MA 4 and AR 1 order for ARMA model.

As we can see from the spectral density plot, there are high peaks up to around 5 cycles per year, indicating significant seasonal components. And another highest peak is at 12 cycles per year, suggesting strong seasonality at this frequency.



❖ Beverages Analysis



The Overall Trend of Beverages shows an increasing trend from 2013 to 2018. While there are some spikes from 2014 to 2015, the weekly sales showed massive increase in sales from 2015.

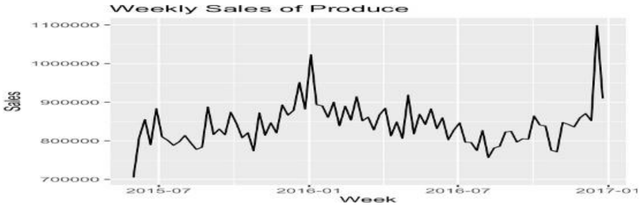
ACF plot revealed the value of ACF going down to a constant rate for each lag from lag 0 equal to 1 to lag 0.5 approximately equal to 0.5. This unveils that there is a progressive reduction in correlation An with the rise of the lag. PACF plot most of the lags were in the confidence interval represented by the blue lines but several peaks at the top are above the blue lines. These spikes were not perceived as very high, indicating that it could be suitable to use a simpler of ARIMA like model.

The EACF plot suggests that both AR and MA components may be present in the series. Based on the test results, this research

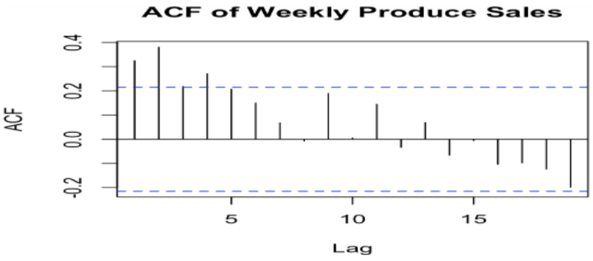
```
>    AR/MA
    0 1 2 3 4 5 6 7 8 9 10 11 12 13
1.   x x x x x x x x x x x  x  x  x
2.   x x x o x o o o x x o  x  x  x
3.   x o x o o o o o x o o  o  o  o
4.   x x o o o o o o x o o  o  o  o
5.   x x o o o o o o x o o  o  o  o
6.   x x o x o o o o x o o  o  o  o
7.   x x x o o o o o x o o  o  o 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

Augmented Dickey-Fuller Test

data:  window_ts
Dickey-Fuller = -
3.4757, Lag order = 6,
p-value = 0.04577
alternative
hypothesis: stationary
```

confirmed that the time series is stationery and p-value was 0.04577, thereby enabling to proceed to ARIMA modeling.

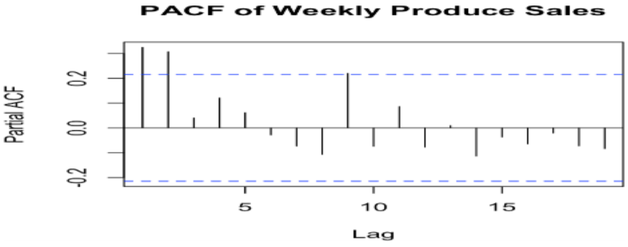
❖ Produce Analysis



This is the weekly sales graph of produce we have removed the data from 2013-2015 mid because the sales in the dataset has 0 and null values so that's why we have removed the portion to maintain a clean and stationary data.

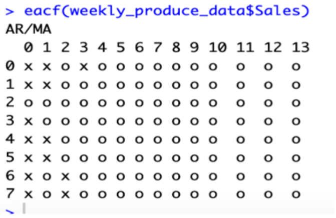


The ACF plot shows significant autocorrelation at lag 1, gradually decreasing over subsequent lags. The first few lags show significant positive autocorrelation, indicating persistence or trend in the data.



The PACF plot shows significant autocorrelation at lag 1, with less significant values at higher lags. The significant spikes at the initial lags suggest that the time series can be modeled with a lower order AR term.

The EACF table provides a clear pattern to identify the AR and MA orders. From the table, we see significant values at (0,0), (0,1), (1,0), and (1,1) suggesting potential orders for AR and MA components.



```
> eacf(weekly_produce_data$Sales)
AR/MA
  0 1 2 3 4 5 6 7 8 9 10 11 12 13
0 x x o x o o o o o o o  o  o  o
1 x x o o o o o o o o o  o  o  o
2 o o o o o o o o o o o  o  o  o
3 x o o o o o o o o o o  o  o  o
4 x x o o o o o o o o o  o  o  o
5 x x o o o o o o o o o  o  o  o
6 x o x o o o o o o o o  o  o  o
7 x o x o o o o o o o o  o  o  o
>
```

Dickey-Fuller Test: The p-value is 0.9191, much higher than 0.05, indicating the null hypothesis of a unit root cannot be rejected. This says the series is non-stationary.

KPSS Test: The p-value is 0.1, higher than 0.05, indicating the null hypothesis of level stationarity cannot be rejected, suggesting the series is stationary.

c. Residual Analysis and Model Diagnostics

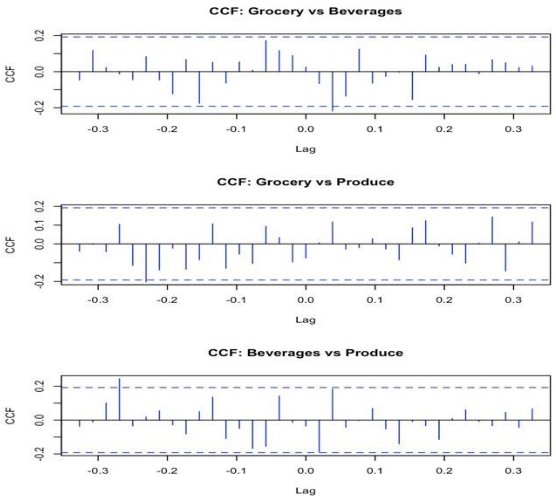
To find the best fitted model for all the three products sales series, we performed different models some of which are shown below:

Product Sales	Fitted Models	sigma^2	AIC	BIC	MPE
Grocery	Regression with ARIMA(1,0,5)(0,0,1)[52] errors	6.911e+3	6185.39	6193.68	40%
	Regression with ARIMA(2,0,5)(0,0,1)[52] errors	6.779e+0	6183.39	6287.81	32%
	Regression with ARIMA(1,0,4)(0,0,1)[52] errors	6.88e+03	6186.76	6282.21	42%

Product Sales	Fitted Models	Sigma^2	AIC	BIC
Produce	Regression with ARIMA(3,0,0)(52)	2.314e+09	2031.88	2043.97
	Regression with log-transformed ARIMA(1,0,4)(52)	0.003079	-233.72	-219.3

Sales	Fitted Models	sigma^2	AIC	BIC	MPE
Beverages	ARIMA(1,1,3)(0,0,1)[52] errors without intervention dummy	1.405e+10	6270.60	6291.46	8.04%
	ARIMA(3,1,1)(0,0,1)[52] errors with intervention dummy	1.387e+10	6268.33	6292.66	8.03%
	ARIMA(0,1,0) on log-transformed data without dummy	0.019	-266.98	-263.51	0.60%
	ARIMA(0,1,0)(0,0,1)[52] errors on log-transformed data with intervention dummy	0.01878	-267.14	-256.71	0.59%

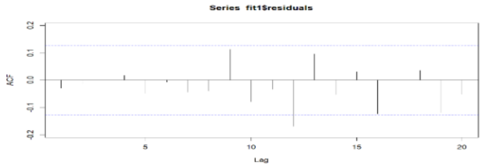
❖ CCF

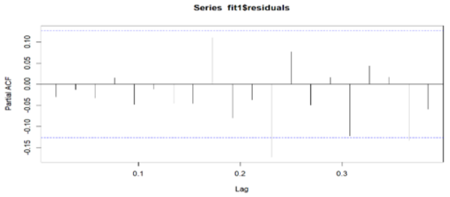


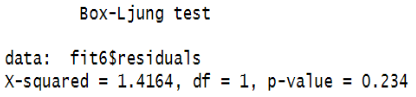
We also tried looking at the Cross correlation for all three products. The CCF analysis reveals the relationships between grocery, beverages, and produce sales over time. Significant positive correlations between grocery and beverages at specific lags indicate that increases in grocery sales often correspond with increases in beverages sales at those time shifts, while negative correlations suggest an inverse relationship at other lags. Similarly, the grocery and produce CCF plot shows positive correlations at some lags and negative correlations at others, indicating varying relationships over time. The beverages and produce CCF plot also display both positive and negative correlations at different lags, suggesting that changes in beverages sales can have a direct or inverse relationship with produce sales depending on the time shift. These insights help businesses understand the interplay between product categories, aiding in better inventory management, marketing strategies, and sales forecasting.

❖ Residual Analysis of Grocery:

Also, we looked at the residuals of the fitted model. The residuals are mostly white noise, except for one lag visible in the ACF and PACF plots. There is no significant autocorrelation in the residuals, and the final model provides good predictions by capturing the complex seasonality.



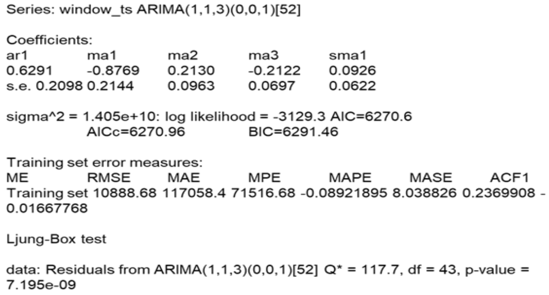




```
        Box-Ljung test

data:  fit6$residuals
X-squared = 1.4164, df = 1, p-value = 0.234
```

❖ Residual Analysis of Beverages:

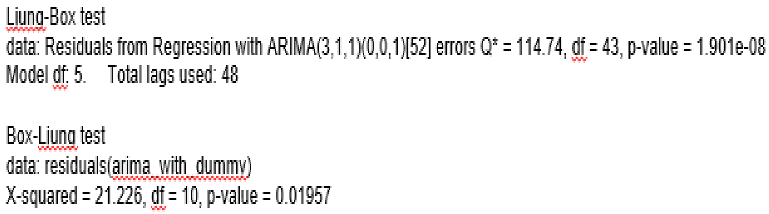
Analysis without dummy



```
Series: window_ts ARIMA(1,1,3)(0,0,1)[52]

Coefficients:
ar1          ma1          ma2          ma3          sma1
0.6291       -0.8769      0.2130       -0.2122      0.0926
s.e. 0.2098 0.2144        0.0963       0.0697       0.0622

sigma^2 = 1.405e+10: log likelihood = -3129.3 AIC=6270.6
             AICc=6270.96              BIC=6291.46

Training set error measures:
ME           RMSE         MAE          MPE          MAPE         MASE         ACF1
Training set 10888.68 117058.4 71516.68 -0.08921895 8.038826 0.2369908 -
0.01667768

Ljung-Box test

data: Residuals from ARIMA(1,1,3)(0,0,1)[52] Q* = 117.7, df = 43, p-value =
7.195e-09
```

The residuals' Ljung-Box test shows significant autocorrelation (p-value = 0.01025), indicating the model may not fully capture the underlying patterns.

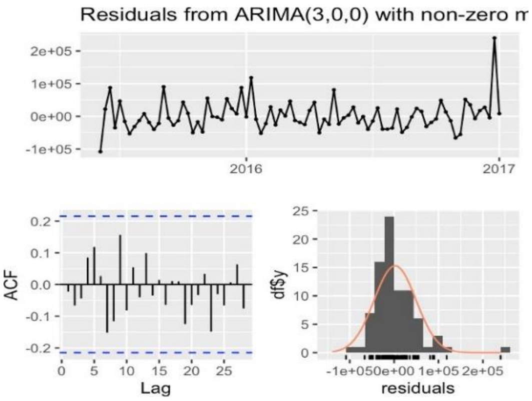
Analysis with dummy:



```
Ljung-Box test
data: Residuals from Regression with ARIMA(3,1,1)(0,0,1)[52] errors Q* = 114.74, df = 43, p-value = 1.901e-08
Model df: 5.    Total lags used: 48

Box-Ljung test
data: residuals(arima_with_dummy)
X-squared = 21.226, df = 10, p-value = 0.01957
```

ARIMA(3,1,1)(0,0,1)[52] model with an intervention dummy. The model captures the level shifts more effectively, as indicated by improved residual diagnostics and a lower AIC value. The intervention dummy helps in modeling the shifts observed in 2015 and 2016.

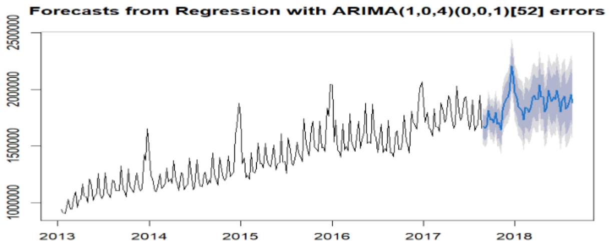
❖ Residual Analysis of Produce:



The top plot displays the residuals over time, indicating the model's fit to the data with occasional large deviations. The bottom left plot is the autocorrelation function (ACF) of the residuals, showing mostly insignificant autocorrelations, suggesting the residuals are uncorrelated. The bottom right plot is a histogram of the residuals, with an overlaid normal distribution curve, indicating the residuals are approximately normally distributed. These diagnostics suggest the ARIMA model is a good fit, with residuals. So, we have done so much of analysis we are getting the same results for the Arima and Sarima so my final module is the Arima.

d. Forecast Analysis

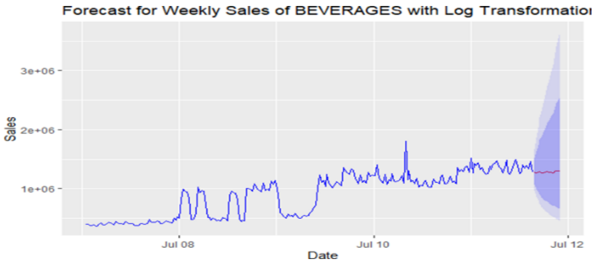
❖ Grocery Sales:



The forecasted plot of the harmonic regression with Arima model highlights both the overall trend and the seasonal fluctuations. This model with Fourier terms predicts the seasonal patterns very well by capturing long-term upward trends and regular fluctuations. However, this model can be used for making future predictions for improved grocery sales.

❖ Beverages Sales:

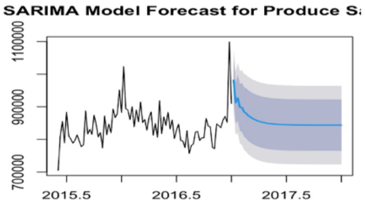
Weekly sales of Beverages with Log Transformation and with Intervention Dummy

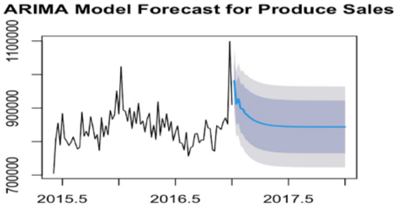


We fitted multiple ARIMA models to capture the underlying patterns in the sales data. This included models with and without log transformations and intervention dummy variables. The intervention dummy was introduced to account for the structural shifts observed in the data, significantly improving model accuracy. Residual diagnostics, including the Ljung-Box test, were conducted for each model to ensure the residuals behaved like white noise.

The final model comparison showed that the ARIMA model applied to the log-transformed data with an intervention dummy performed the best. This model provided the lowest AIC and effectively captured both trends and structural shifts. Forecasts for the next 15 weeks were generated, displaying confidence intervals to indicate the uncertainty of the predictions. This comprehensive approach underscored the importance of addressing structural changes in time series data and conducting thorough residual diagnostics to ensure robust and reliable forecasts.

❖ Produce Sales:





Both plots provide forecasts for produce sales with their respective confidence intervals. The SARIMA model likely incorporates seasonality, while the ARIMA model provides a non-seasonal forecast. Both models show a decline in forecast accuracy over time, as indicated by the widening confidence intervals.

e. Analysis of The Results and Discussion

In our analysis, we applied harmonic regression with ARIMA to understand the yearly seasonality of grocery sales, finding that the ARIMA model with Fourier terms effectively captures the data's variability, achieving a Mean Percentage Error (MPE) of 42%. This indicates the model's robustness for making future predictions after removing outliers. For the beverage's sales series, the ARIMA model with log transformation and intervention dummy provided the best fit, capturing both trends and level shifts accurately, making it highly suitable for forecasting and decision-making. When analyzing the produce sales series, we encountered significant null and zero values, which led to ARIMA and SARIMA models yielding similar forecasts. Despite various attempts to refine the model, the similarity in forecasts suggests the model's robustness, though the data quality remains a limiting factor.

3. Appendix

Individual Report

I) Fauzia Khan

Initial Dataset:

From the beginning of the project, I took part in finding an appropriate dataset that aligned with my and group members interests. After considering a couple of datasets, our group decided to work on an e-commerce dataset, which we later replaced. Initially, I took the initiative to create exploratory graphs like bar graphs and scatter plots, and I also assessed the normality of the data through QQ plots to analyze the distribution. Additionally, I constructed the main weekly sales for the top-selling product in that dataset. After receiving valuable feedback, our group switched to a different sales dataset because the initial one lacked sufficient data for comprehensive analysis.

Store Sales Forecast Dataset:

In this store sales dataset I was responsible for creating the main model of harmonic regression of grocery sales. I started off by looking at ACF, PACF, EACF, and Box plots to KPSS and ADF tests to analyze and construct different ARIMA models for forecasting. This required a deep understanding of how to handle non-stationary series and how to deal with series that are not white noise. I also tried different transformation techniques such as taking logs, differencing, and log returns. During this project I also tried looking at the differenced series for the same grocery series, but it was over-differencing. Although, I also tried looking at the log transformed on this grocery series and analyzed its ACF, PACF, EACF etc because I visualized a little bit of transformation, but it didn't really affect the series. Before coming up to this final harmonic regression with ARIMA model I also tried to explore different combinations of ARIMA model to best look at these sales seasonal pattern and also analyzed their residuals and performed backtesting. This was very challenging, as it requires significant time to try different models, especially with seasonality, to achieve residuals that are white noise. I also did forecasting after constructing each of the model before exploring the multiple seasonality which leads me to harmonic regression with ARIMA model to be the best fitted model for . In addition, I analyzed the auto ARIMA models to compare and evaluate their performance with the manaully constructed models. This entire process of building an harmonic regression model has greatly helped me understand the concepts of multi-seasonality, stationarity, Fourier terms and forecasting.

In addition to the project's technical aspects, I took the initiative to arrange online meetings for our group to discuss how to analyze the dataset effectively. These meetings were crucial for coordinating our efforts and ensuring everyone was on the same page. On behalf of my group, I presented the initial series in class, where we received valuable feedback from the professor. This feedback provided us with a clearer direction on how to start our analysis.

Throughout the project, I contributed to completing every milestone. During the creation of the final presentation, I was responsible for generating slides related to weekly grocery sales. I created all the relevant slides showing ACF, PACF, EACF, model building, spectrum analysis, and residual analysis. Also, I included my overall analysis of the harmonic regression model for predictions in the final project report and the conclusion of the results. I made sure to align the document with proper adjustments for clarity and readability.

Despite the limited time frame, I aimed to include as much detail as possible in the final report. However, it was challenging to cover everything I did during reaching the final model for this project. This project was a comprehensive learning experience, and I ensured that my contributions were well-documented and presented clearly to highlight the rigorous analysis and modeling efforts involved.

Take Aways and Reflection: This project has significantly enriched my grasp and application of time series techniques and concepts. I've developed the ability to analyze and construct models for stationary data, employing various transformations and effectively implementing different ARIMA models. One of the most important aspects of this learning experience was diagnosing model performance through residual analysis and ensuring that residuals exhibit white noise properties, which are essential for precise forecasting. This step involved thorough checks using techniques such as the autocorrelation function (ACF) and partial autocorrelation function (PACF) plots.

In addition, this project expanded my knowledge of understanding the evolution of variables over time, particularly in the context of grocery sales. I learned how to handle and interpret various seasonal patterns within the data, which is crucial for accurate forecasting in retail environments. Specifically, I enjoyed learning how to assess multiple seasonality through harmonic regression, a technique that allowed us to capture complex seasonal variations effectively.

Furthermore, this project provided hands-on experience in the practical application of theoretical concepts. I gained insights into the challenges of real-world data analysis, such as dealing with missing values and outliers, and learned how to apply various statistical methods to address these issues.

Moreover, this course has expanded my understanding of forecasting methodologies using time series data, which holds substantial long-term benefits. The practical application of class examples, discussions, and regular practice sessions was particularly effective in solidifying complex concepts which was helpful in project. The thing that really amazed me is understanding the patterns and trends in data, identifying seasonality and trends, handling non-stationarity through differencing or transformations like log and Box-Cox, and selecting appropriate models like ARIMA (Autoregressive Integrated Moving Average) or its variants such as SARIMA (Seasonal ARIMA) for forecasting. These techniques are crucial in various fields including finance, economics, epidemiology, etc and can be beneficial for my career path.

ii) Maheen Adeeb

Contribution to Team’s Efforts

In our time series analysis project focusing on store sales data, I concentrated on the "BEVERAGES" product family while other team members worked on the "GROCERIES" and "PRODUCE" product families. My contributions to the team's efforts include cleaning and preparing the dataset, ensuring proper date formatting and filtering relevant records, and aggregating the sales data to weekly totals for coherent analysis. During the exploratory data analysis (EDA) phase, I visualized the time series data to identify trends and potential structural changes, generating ACF and PACF plots to understand the autocorrelation structure and performing the Augmented Dickey-Fuller test to check for stationarity. For model building, I fitted multiple ARIMA models, including an intervention dummy variable to account for structural shifts observed around 2015 and 2016. Residual diagnostics and the Ljung-Box test were conducted to ensure the residuals were white noise. Forecasts were generated for the next 15 weeks with confidence intervals to indicate uncertainty, and model performance was compared based on AIC, BIC, and residual diagnostics. Additionally, I initially worked on a different dataset focusing on e-commerce sales, performing EDA by visualizing time series for different products, generating ACF and PACF plots, and conducting initial time series modeling. However, this dataset did not have enough data to proceed with a reliable analysis. Despite the limited data, this effort contributed to our understanding of the challenges involved in time series analysis for different types of datasets. Overall, these contributions ensured a thorough and robust analysis of the BEVERAGES product family, addressing key aspects of time series analysis such as trend identification, model fitting, and forecasting.

1. **Data Preprocessing and Cleaning**:
    - I was responsible for extracting and preparing the dataset specific to the "BEVERAGES" product family. This involved filtering the data to include only relevant records and removing any entries with zero sales.
    - Ensured that the date column was properly formatted and consistent, facilitating smooth analysis and accurate time series conversion.
    - Aggregated sales data to weekly totals to align with our analysis period.
2. **Exploratory Data Analysis (EDA)**:
    - Conducted a detailed exploratory analysis of the "BEVERAGES" sales data to identify trends, seasonal patterns, and any anomalies.
    - Visualized the time series data to highlight key trends and potential structural changes, which guided our modeling approach.
3. **Model Building and Selection**:
    - Fitted multiple ARIMA models to the original, log-transformed, and intervention-adjusted "BEVERAGES" sales data.
    - Applied log transformations to stabilize the variance of the sales data and examined the impact on model performance.
    - Introduced intervention dummy variables to account for structural changes observed in the sales data, improving the model's ability to capture these shifts.
    - Used auto.arima to select the best-fitting models based on AIC criteria, ensuring that our models were both accurate and parsimonious.
4. **Residual Analysis and Diagnostics**:
    - Performed residual diagnostics for each model to ensure that the residuals were white noise, indicating a good fit.
    - Conducted the Ljung-Box test to check for autocorrelation in the residuals and iterated on model selection to minimize this issue.
5. **Forecasting**:
    - Generated forecasts for the next 15 weeks for the "BEVERAGES" product family using the selected ARIMA models.
    - Visualized the forecasted values along with historical sales data, providing a clear comparison and highlighting the model's performance.
    - Displayed confidence intervals to communicate the uncertainty in our forecasts.
6. **Presentation Preparation**:
    - Developed comprehensive presentation slides to summarize our findings for the "BEVERAGES" product family.

- Included detailed explanations of each analysis step, model selection process, and the resulting forecasts.
- Collaborated with team members to ensure consistency and coherence in the overall presentation.

Takeaways from the Project

This project has significantly enhanced my understanding of time series analysis and its application in real-world scenarios. Here are two key takeaways:

7. **Significance of Structural Change Handling**:
    - One of the major insights I gained is the importance of accounting for structural changes within time series data. The "BEVERAGES" sales data exhibited distinct level shifts, which we addressed by incorporating intervention dummy variables.
    - This approach not only improved the model's accuracy but also highlighted the necessity of understanding and modeling such shifts to produce reliable forecasts. This experience has taught me that structural changes are common in business data, and addressing them appropriately is crucial for accurate modeling.
8. **Importance of Comprehensive Model Diagnostics**:
    - Another critical lesson learned is the value of thorough residual diagnostics. Simply selecting a model with good fit metrics is not enough; it's essential to ensure that the model's residuals are free of autocorrelation.
    - The Ljung-Box test was particularly useful in identifying residual autocorrelation, prompting further refinement of our models. This process underscored the need for rigorous evaluation of model assumptions and diagnostics to ensure the robustness of our forecasts.

Overall, this project has deepened my understanding of time series analysis, emphasized the importance of handling structural changes and conducted comprehensive model diagnostics. These skills will be invaluable in future data analysis endeavors across various domains.

iii) Sai Chaitanya Balla

About my variable: The variable which i have selected was produce which is very complicated because we the produce dataset has many null and 0 values so due to that we are getting nonstationary, so we have removed the data form 2013 –2015 mid because we don't have any sales on the dataset which is very strange due to that we have experienced very vast time on the variable.

In this project, I played a crucial role in analyzing and forecasting produce sales data using ARIMA and SARIMA models. The project encompassed various stages, from initial data preparation and exploratory data analysis to model fitting and generating forecasts. Here’s a comprehensive summary of the key activities and findings:

1. Data Preparation and Initial Analysis:

- **Data Loading and Filtering**: The dataset was filtered to include only the "produce" category, focusing on sales data from mid-2015 to mid-2017.
- **Weekly Aggregation**:
- Sales data were aggregated on a weekly basis to smooth out daily fluctuations and provide a clearer trend.
- **Handling Missing Values**: The dataset was checked for missing values, which were handled appropriately to ensure the integrity of the analysis.
- **Log Transformation and Differencing**: Log transformation and differencing were applied to the sales data to stabilize variance and achieve stationarity, essential for accurate time series modeling.

2. Stationarity Testing:

- **ADF Test**: The Augmented Dickey-Fuller (ADF) test was conducted, which initially indicated non-stationarity in the time series.
- **KPSS Test**: The KPSS test further confirmed the need for transformation. After log transformation and differencing, the series met the stationarity criteria, paving the way for effective model fitting.

3. Model Fitting:

- **ARIMA Model**: The ARIMA model was fitted to the transformed data. Model selection criteria, including AIC and BIC, were used to determine the optimal model parameters.
- **SARIMA Model**: A Seasonal ARIMA (SARIMA) model was also fitted to capture any seasonal patterns in the data. This model considered seasonal variations and was evaluated similarly using AIC and BIC.
- **Residual Diagnostics**: Both models underwent residual diagnostics to ensure no significant patterns were left unexplained, confirming that the residuals behaved like white noise.

4. Forecasting:

- **Forecast Generation**: Forecasts for the next 52 weeks were generated using both the ARIMA and SARIMA models. These forecasts included 80% and 95% confidence intervals to reflect the uncertainty in predictions.
- **Visualization**: The forecasts were visualized, showing the expected sales trends along with the confidence intervals.

5. Model Comparison:

- **Performance Metrics**: The performance of both models was compared using metrics such as Mean Error (ME), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE). Both models exhibited similar performance, indicating robust modeling.

- **Seasonal Component Impact**: The comparison showed that adding a seasonal component did not significantly alter the model's fit, suggesting that a simpler ARIMA model could be sufficient for this data.

6. *Key Findings and Insights:*

- **Stabilization Trend**: Both models predicted that produce sales would initially decline slightly but then stabilize around a mean value.
- **Uncertainty**: The forecasts indicated increasing uncertainty over longer periods, as reflected by the widening confidence intervals.
- **Model Robustness**: The similarity in forecasts from both ARIMA and SARIMA models highlighted the robustness of the chosen modeling approach.

Conclusion:

This project successfully demonstrated the application of ARIMA and SARIMA models for sales forecasting. The process highlighted the importance of data transformation for achieving stationarity and the effectiveness of ARIMA models in capturing sales trends. The insights gained from the forecasts are valuable for inventory management and strategic planning, emphasizing the need for continuous model evaluation and optimization.

Overall, the project provided a thorough understanding of time series analysis and its practical applications, reinforcing the critical role of meticulous data preparation, rigorous model fitting, and comprehensive validation in achieving accurate forecasts. This experience has equipped me with enhanced skills in time series forecasting and a deeper appreciation for collaborative project execution.

iv) Sumera Fatima Khatoon

Introduction and Group Formation

I played a crucial role in the formation of our group and participated actively in the primary discussions. My contributions included:

- Setting objectives

- Defining the scope of work

- Assigning responsibilities

- Formulating the strategic plan for project implementation

As the writer of the project report's introduction section, I presented:

- An overview of the dataset used in the study

- Justification of its usefulness

Goals of the study

The first section of our paper introduced the research topic and provided context by emphasizing the significance of sales forecasting for grocery stores and our application of machine learning in this context.

Exploratory Data Analysis (EDA)

I assumed overall responsibility for the analysis of the Sales Data and specifically focused on the Top Products data. Key activities included:

- Examining regular movements in the data

- Presenting findings graphically

- Synthesizing the results

In the top products analysis, I identified the most popular product categories and their sales trends.

ARIMA Model Implementation

I centered my work around the implementation of the ARIMA model on the total sales series. This involved:

- Data cleaning for time series analysis

- Handling missing observations and necessary transformations

- Using the ADF test for stationarity testing

- Differencing to achieve stationarity

I identified the best ARIMA model based on AIC and BIC and estimated the model on the total sales series. Using the selected ARIMA model, I generated sales predictions for the upcoming period and illustrated the prediction outcomes.

Report Preparation and Model Evaluation

For the third milestone, I significantly contributed to the preparation and submission of the report by:

- Synthesizing evidence

- Documenting all analyses conducted

- Verifying the correctness of outcomes

The next steps involve:

- Comparing the final three models on our sales data

- Evaluating model performance using RMSE and/or MAPE measures

- Adjusting model parameters for optimal prediction accuracy

- Identifying the most suitable model for our data

Additionally, I conducted a correlation analysis of three variables: grocers, beverages, and produce.

Final Report and Presentation

In the final report, I contributed to both the non-technical and technical parts, focusing on:

- EDA

- Correlation analysis of three variables

- Conclusion, discussing implications and future research areas

I also played a key role in preparing our PowerPoint presentation, from designing the slide layout to synthesizing the introduction, main findings, and comparisons of the three variables, and rehearsing for the concluding presentation.

Learning Outcomes and Project Summary

Throughout this project, I learned various concepts related to time series analysis and machine learning algorithms, including:

- Data cleaning and normalization

- Stationarity testing, ACF/PACF analysis, and differencing

- Applying ARIMA models for forecasting

- Model selection and evaluation

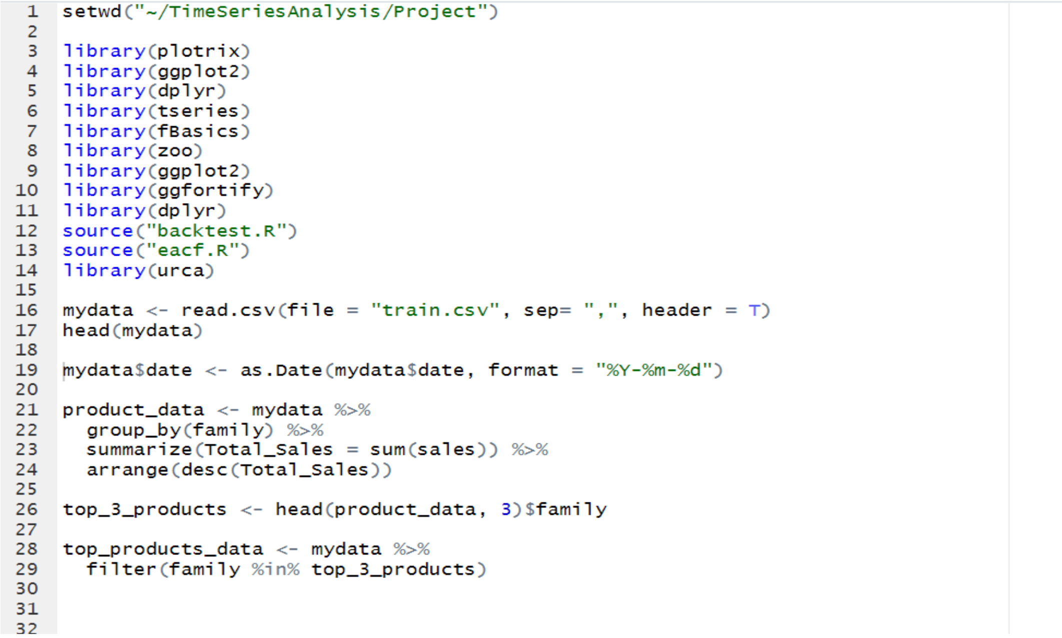
The importance of group cooperation and proper coordination was also evident in the project's success. In summary, I actively participated in the initial meetings, exploratory analysis, and ARIMA model application. I gained substantial knowledge and am ready to continue contributing to model comparison, report writing, and presentation completion. This project has been enjoyable and has allowed me to develop valuable skills for future projects and real-world applications.

Conclusion

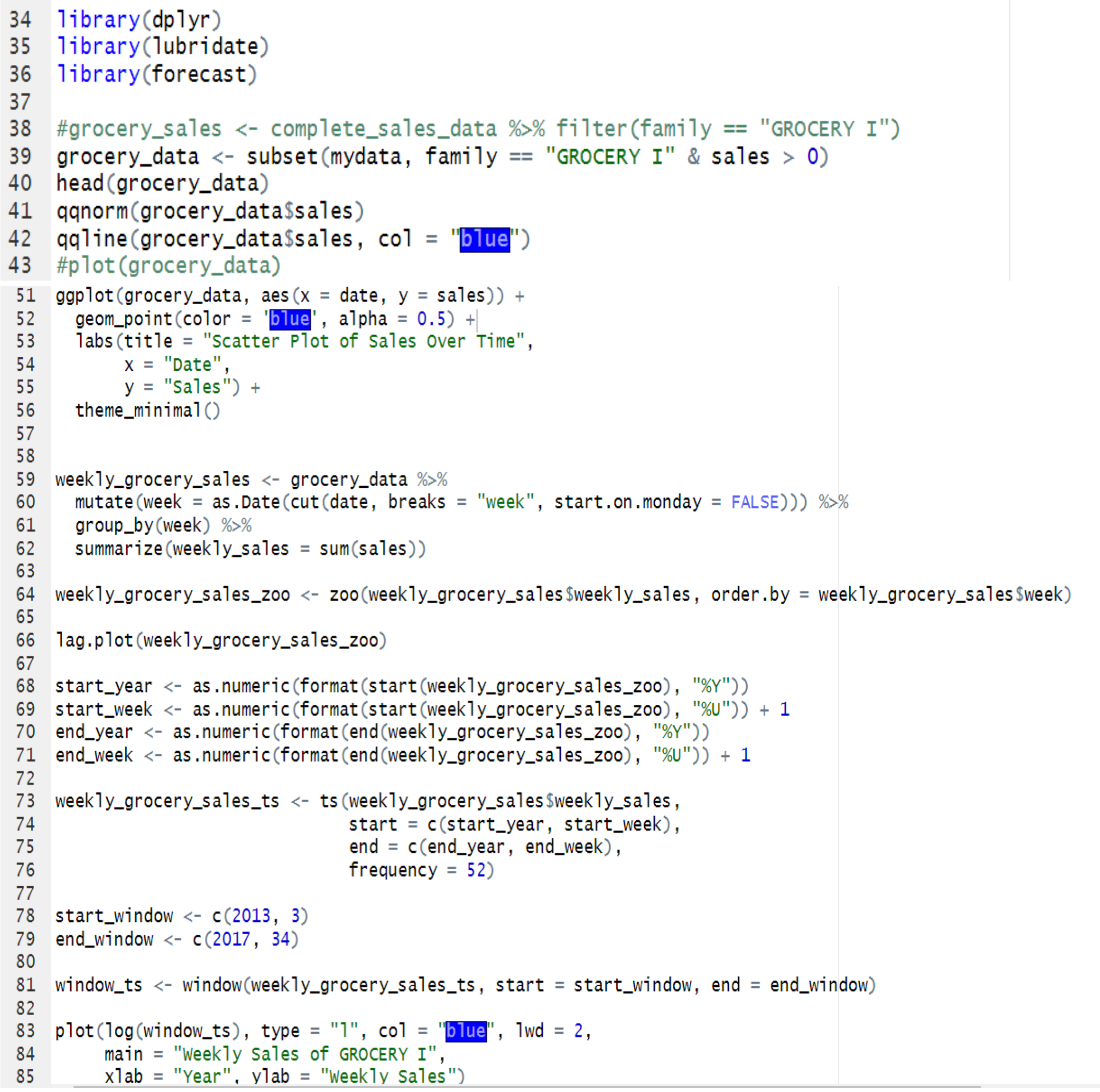
Applying these methods has enhanced our ability to improve sales forecast accuracy. Understanding seasonal variations, accounting for external influences, and ensuring data consistency have helped us make more informed decisions. These lessons have enriched my understanding of time series analysis and equipped me with skills valuable for future projects.

Project Code

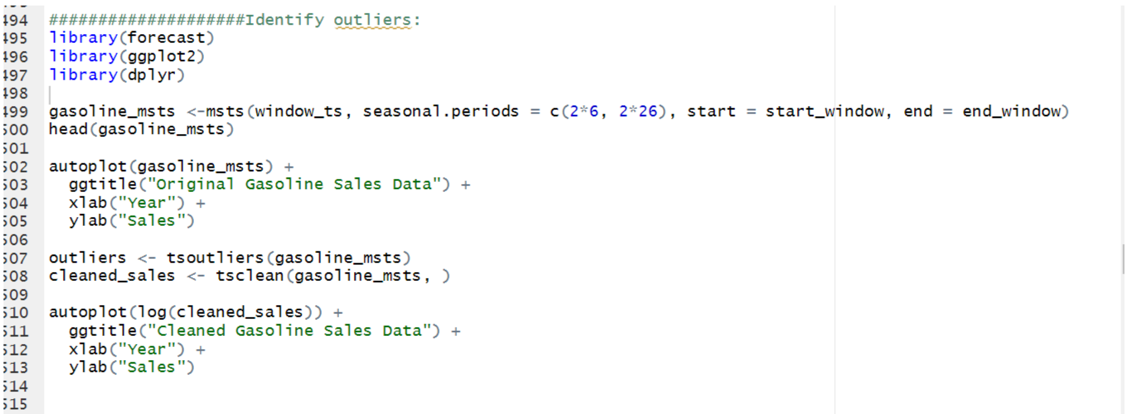
Grocery Series:



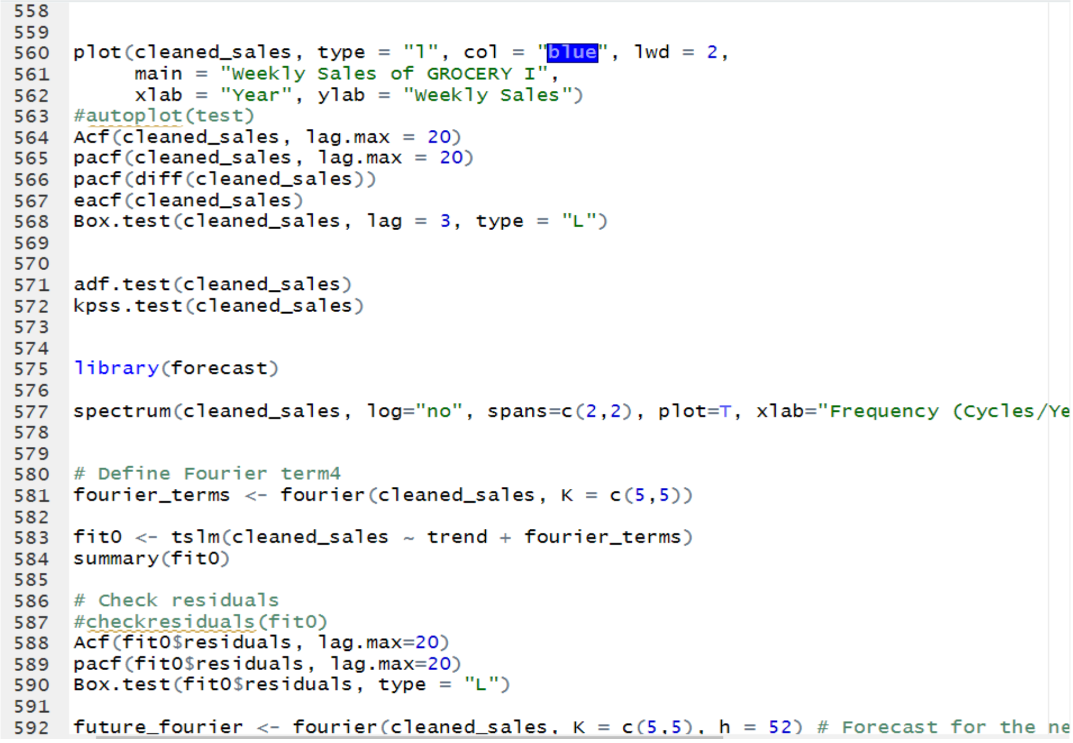
```
setwd("~/TimeSeriesAnalysis/Project")

library(plotrix)
library(ggplot2)
library(dplyr)
library(tseries)
library(fBasics)
library(zoo)
library(ggplot2)
library(ggfortify)
library(dplyr)
source("backtest.R")
source("eacf.R")
library(urca)

mydata <- read.csv(file = "train.csv", sep= ",", header = T)
head(mydata)

mydata$date <- as.Date(mydata$date, format = "%Y-%m-%d")

product_data <- mydata %>%
  group_by(family) %>%
  summarize(Total_Sales = sum(sales)) %>%
  arrange(desc(Total_Sales))

top_3_products <- head(product_data, 3)$family

top_products_data <- mydata %>%
  filter(family %in% top_3_products)
```



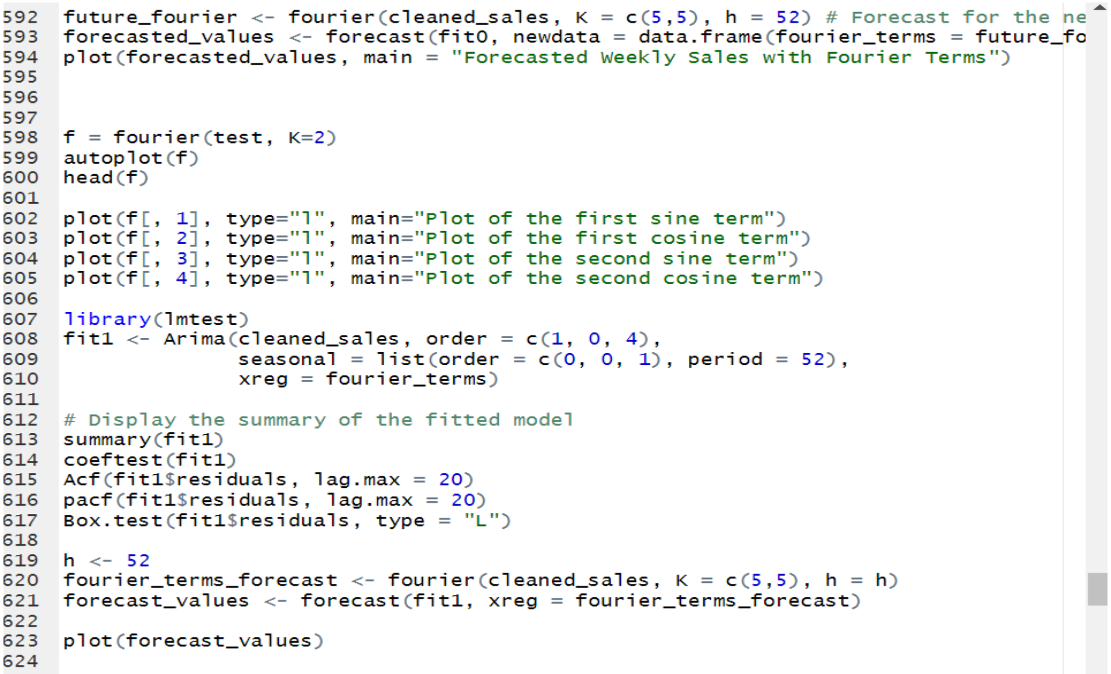
```r
library(dplyr)
library(lubridate)
library(forecast)

#grocery_sales <- complete_sales_data %>% filter(family == "GROCERY I")
grocery_data <- subset(mydata, family == "GROCERY I" & sales > 0)
head(grocery_data)
qqnorm(grocery_data$sales)
qqline(grocery_data$sales, col = "blue")
#plot(grocery_data)
ggplot(grocery_data, aes(x = date, y = sales)) +
  geom_point(color = 'blue', alpha = 0.5) +
  labs(title = "Scatter Plot of Sales Over Time",
       x = "Date",
       y = "Sales") +
  theme_minimal()


weekly_grocery_sales <- grocery_data %>%
  mutate(week = as.Date(cut(date, breaks = "week", start.on.monday = FALSE))) %>%
  group_by(week) %>%
  summarize(weekly_sales = sum(sales))

weekly_grocery_sales_zoo <- zoo(weekly_grocery_sales$weekly_sales, order.by = weekly_grocery_sales$week)

lag.plot(weekly_grocery_sales_zoo)

start_year <- as.numeric(format(start(weekly_grocery_sales_zoo), "%Y"))
start_week <- as.numeric(format(start(weekly_grocery_sales_zoo), "%U")) + 1
end_year <- as.numeric(format(end(weekly_grocery_sales_zoo), "%Y"))
end_week <- as.numeric(format(end(weekly_grocery_sales_zoo), "%U")) + 1

weekly_grocery_sales_ts <- ts(weekly_grocery_sales$weekly_sales,
                              start = c(start_year, start_week),
                              end = c(end_year, end_week),
                              frequency = 52)

start_window <- c(2013, 3)
end_window <- c(2017, 34)

window_ts <- window(weekly_grocery_sales_ts, start = start_window, end = end_window)

plot(log(window_ts), type = "l", col = "blue", lwd = 2,
     main = "Weekly Sales of GROCERY I",
     xlab = "Year", ylab = "Weekly Sales")
```



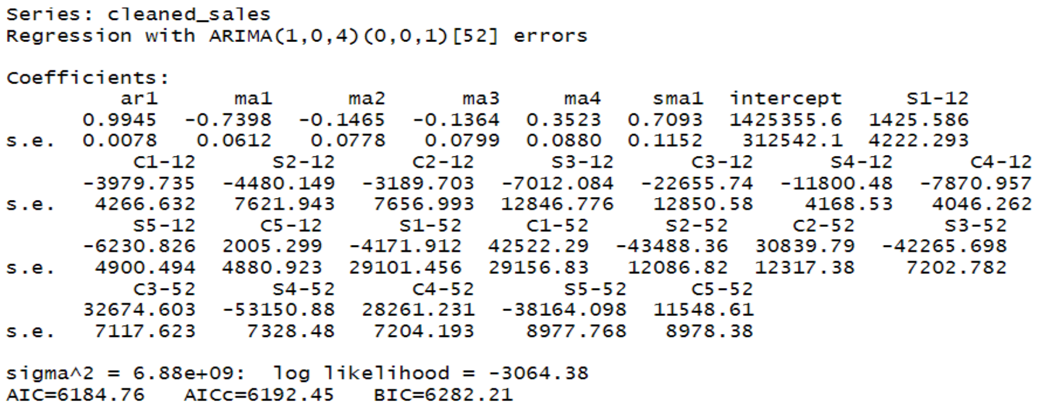
```r
####################Identify outliers:
library(forecast)
library(ggplot2)
library(dplyr)

gasoline_msts <-msts(window_ts, seasonal.periods = c(2*6, 2*26), start = start_window, end = end_window)
head(gasoline_msts)

autoplot(gasoline_msts) +
  ggtitle("Original Gasoline Sales Data") +
  xlab("Year") +
  ylab("Sales")

outliers <- tsoutliers(gasoline_msts)
cleaned_sales <- tsclean(gasoline_msts, )

autoplot(log(cleaned_sales)) +
  ggtitle("Cleaned Gasoline Sales Data") +
  xlab("Year") +
  ylab("Sales")

```



```
plot(cleaned_sales, type = "l", col = "blue", lwd = 2,
     main = "Weekly Sales of GROCERY I",
     xlab = "Year", ylab = "Weekly Sales")
#autoplot(test)
Acf(cleaned_sales, lag.max = 20)
pacf(cleaned_sales, lag.max = 20)
pacf(diff(cleaned_sales))
eacf(cleaned_sales)
Box.test(cleaned_sales, lag = 3, type = "L")


adf.test(cleaned_sales)
kpss.test(cleaned_sales)


library(forecast)

spectrum(cleaned_sales, log="no", spans=c(2,2), plot=T, xlab="Frequency (Cycles/Ye


# Define Fourier term4
fourier_terms <- fourier(cleaned_sales, K = c(5,5))

fit0 <- tslm(cleaned_sales ~ trend + fourier_terms)
summary(fit0)

# Check residuals
#checkresiduals(fit0)
Acf(fit0$residuals, lag.max=20)
pacf(fit0$residuals, lag.max=20)
Box.test(fit0$residuals, type = "L")

future_fourier <- fourier(cleaned_sales, K = c(5,5), h = 52) # Forecast for the ne
```

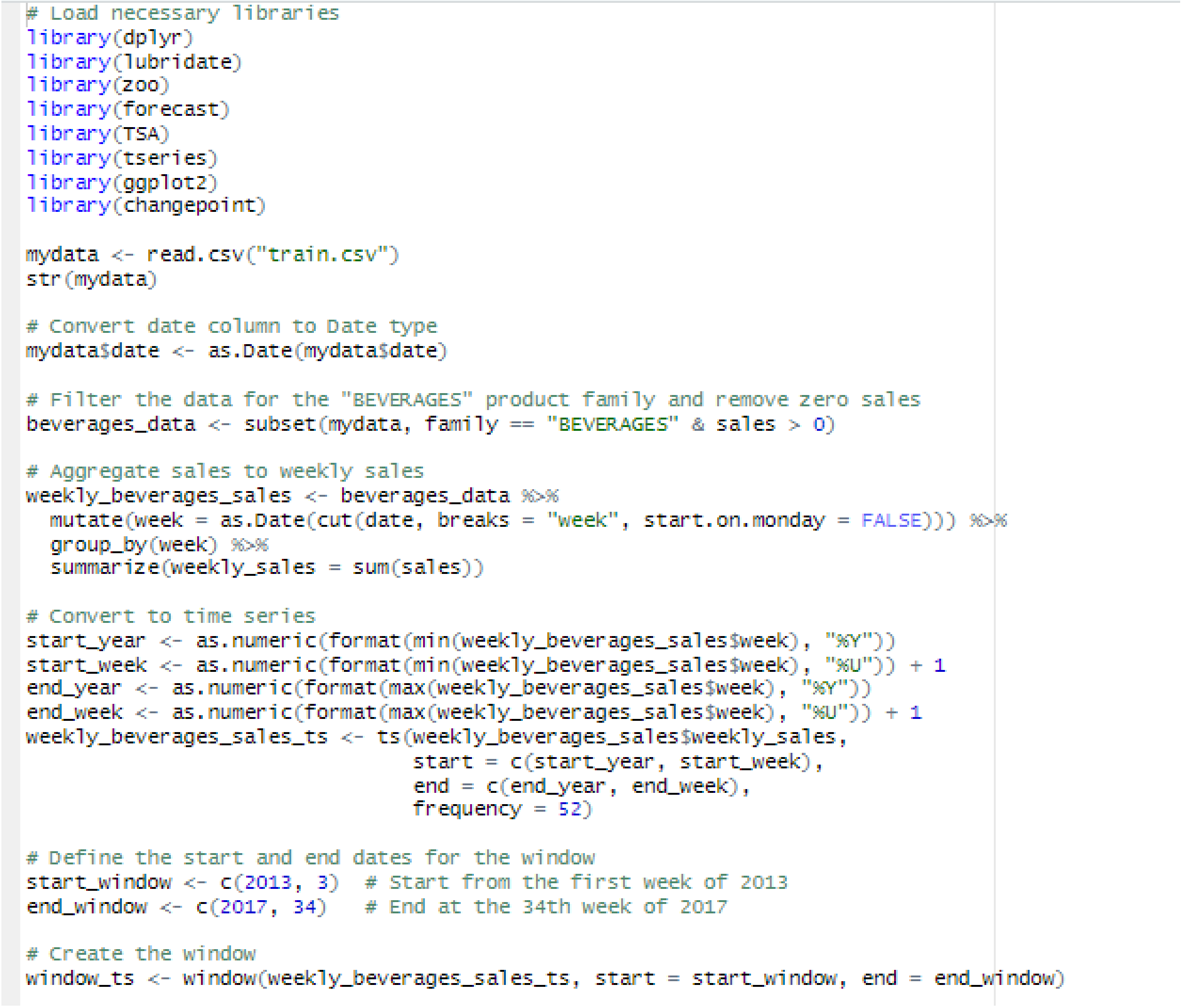


```
future_fourier <- fourier(cleaned_sales, K = c(5,5), h = 52) # Forecast for the ne
forecasted_values <- forecast(fit0, newdata = data.frame(fourier_terms = future_fo
plot(forecasted_values, main = "Forecasted Weekly Sales with Fourier Terms")



f = fourier(test, K=2)
autoplot(f)
head(f)

plot(f[, 1], type="l", main="Plot of the first sine term")
plot(f[, 2], type="l", main="Plot of the first cosine term")
plot(f[, 3], type="l", main="Plot of the second sine term")
plot(f[, 4], type="l", main="Plot of the second cosine term")

library(lmtest)
fit1 <- Arima(cleaned_sales, order = c(1, 0, 4),
              seasonal = list(order = c(0, 0, 1), period = 52),
              xreg = fourier_terms)

# Display the summary of the fitted model
summary(fit1)
coeftest(fit1)
Acf(fit1$residuals, lag.max = 20)
pacf(fit1$residuals, lag.max = 20)
Box.test(fit1$residuals, type = "L")

h <- 52
fourier_terms_forecast <- fourier(cleaned_sales, K = c(5,5), h = h)
forecast_values <- forecast(fit1, xreg = fourier_terms_forecast)

plot(forecast_values)

```

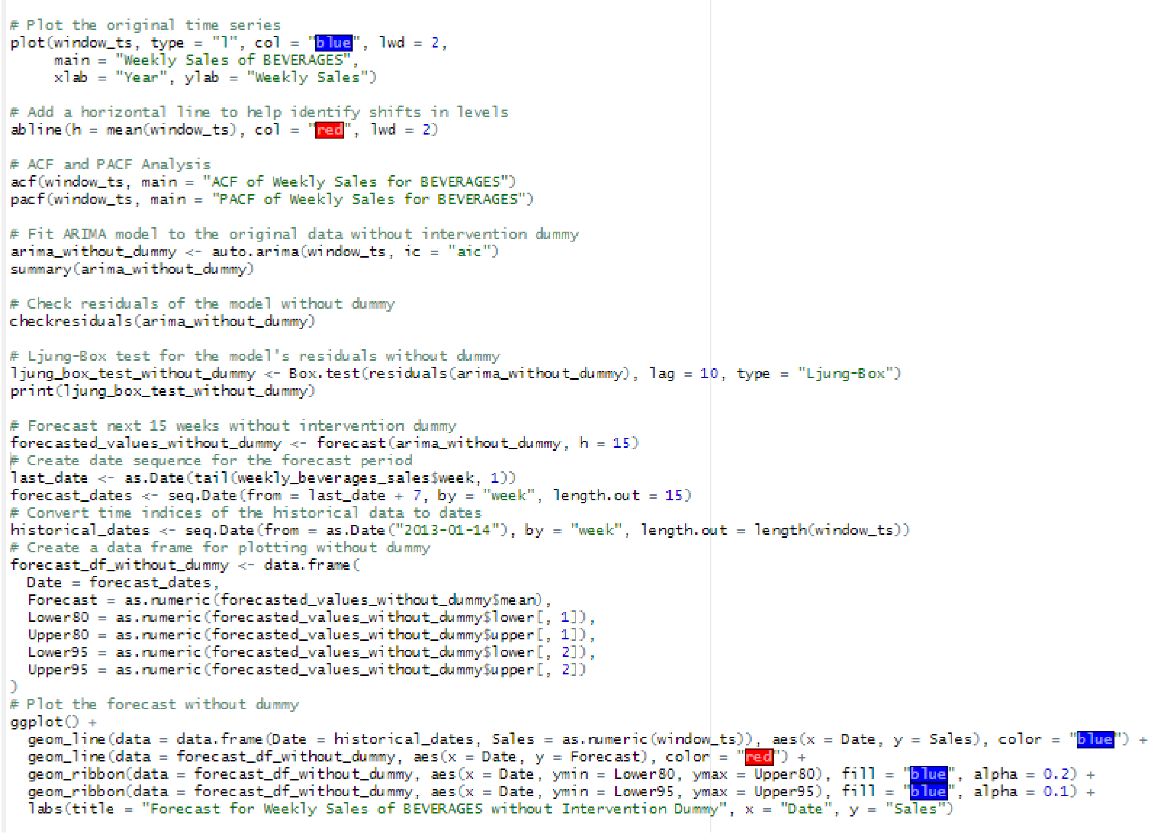


```
Series: cleaned_sales
Regression with ARIMA(1,0,4)(0,0,1)[52] errors

Coefficients:
         ar1      ma1      ma2      ma3     ma4    sma1  intercept     S1-12
      0.9945  -0.7398  -0.1465  -0.1364  0.3523  0.7093  1425355.6  1425.586
s.e.  0.0078   0.0612   0.0778   0.0799  0.0880  0.1152   312542.1  4222.293
          C1-12      S2-12      C2-12      S3-12      C3-12      S4-12      C4-12
      -3979.735  -4480.149  -3189.703  -7012.084  -22655.74  -11800.48  -7870.957
s.e.   4266.632   7621.943   7656.993  12846.776   12850.58    4168.53   4046.262
          S5-12     C5-12      S1-52     C1-52      S2-52     C2-52       S3-52
      -6230.826  2005.299  -4171.912  42522.29  -43488.36  30839.79  -42265.698
s.e.   4900.494  4880.923  29101.456  29156.83   12086.82  12317.38    7202.782
          C3-52      S4-52      C4-52       S5-52     C5-52
      32674.603  -53150.88  28261.231  -38164.098  11548.61
s.e.   7117.623    7328.48   7204.193    8977.768   8978.38

sigma^2 = 6.88e+09:  log likelihood = -3064.38
AIC=6184.76   AICc=6192.45   BIC=6282.21
```

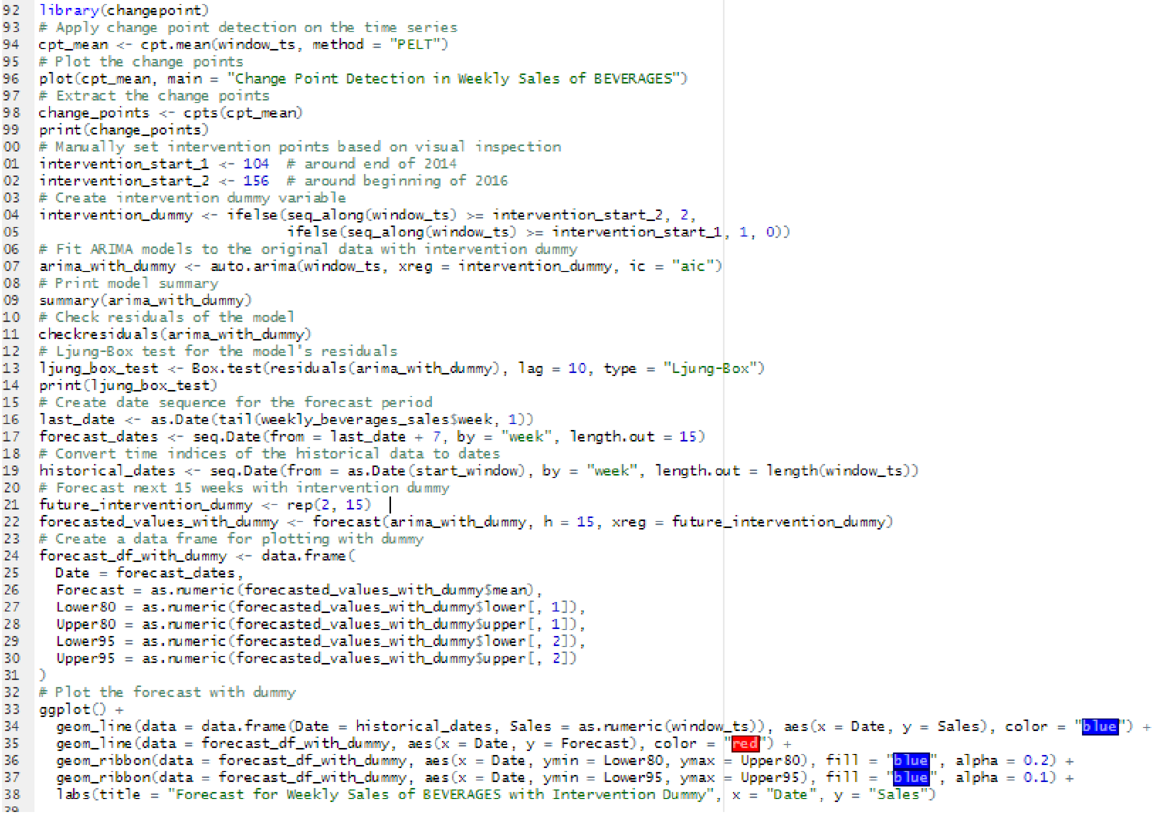
Beverages series:



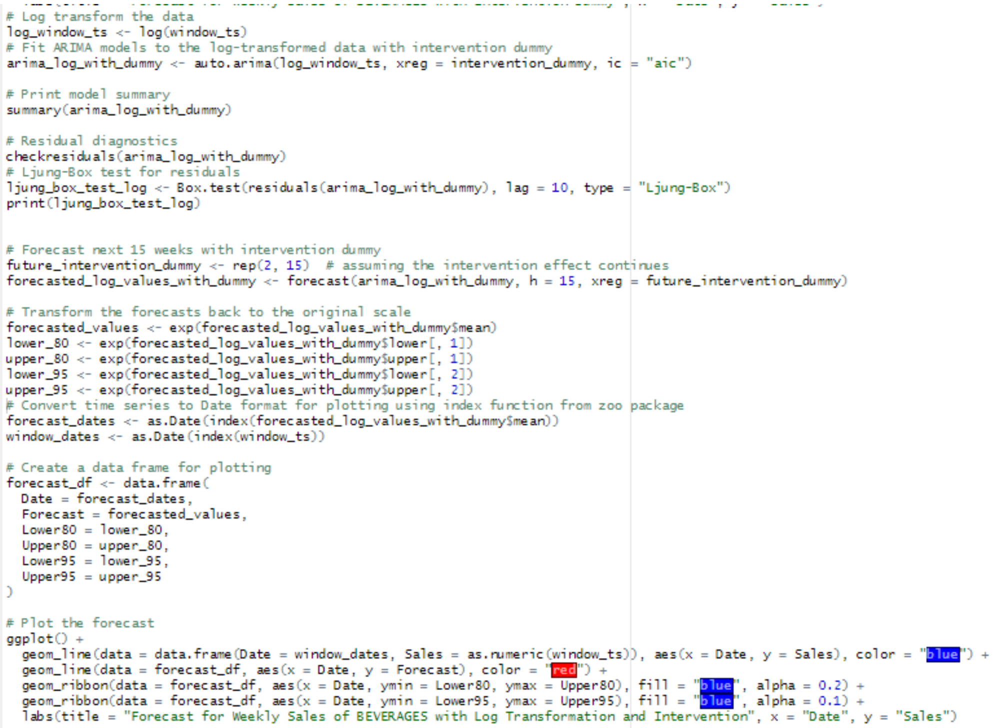
```
# Load necessary libraries
library(dplyr)
library(lubridate)
library(zoo)
library(forecast)
library(TSA)
library(tseries)
library(ggplot2)
library(changepoint)

mydata <- read.csv("train.csv")
str(mydata)

# Convert date column to Date type
mydata$date <- as.Date(mydata$date)

# Filter the data for the "BEVERAGES" product family and remove zero sales
beverages_data <- subset(mydata, family == "BEVERAGES" & sales > 0)

# Aggregate sales to weekly sales
weekly_beverages_sales <- beverages_data %>%
  mutate(week = as.Date(cut(date, breaks = "week", start.on.monday = FALSE))) %>%
  group_by(week) %>%
  summarize(weekly_sales = sum(sales))

# Convert to time series
start_year <- as.numeric(format(min(weekly_beverages_sales$week), "%Y"))
start_week <- as.numeric(format(min(weekly_beverages_sales$week), "%U")) + 1
end_year <- as.numeric(format(max(weekly_beverages_sales$week), "%Y"))
end_week <- as.numeric(format(max(weekly_beverages_sales$week), "%U")) + 1
weekly_beverages_sales_ts <- ts(weekly_beverages_sales$weekly_sales,
                                start = c(start_year, start_week),
                                end = c(end_year, end_week),
                                frequency = 52)

# Define the start and end dates for the window
start_window <- c(2013, 3)  # Start from the first week of 2013
end_window <- c(2017, 34)   # End at the 34th week of 2017

# Create the window
window_ts <- window(weekly_beverages_sales_ts, start = start_window, end = end_window)
```



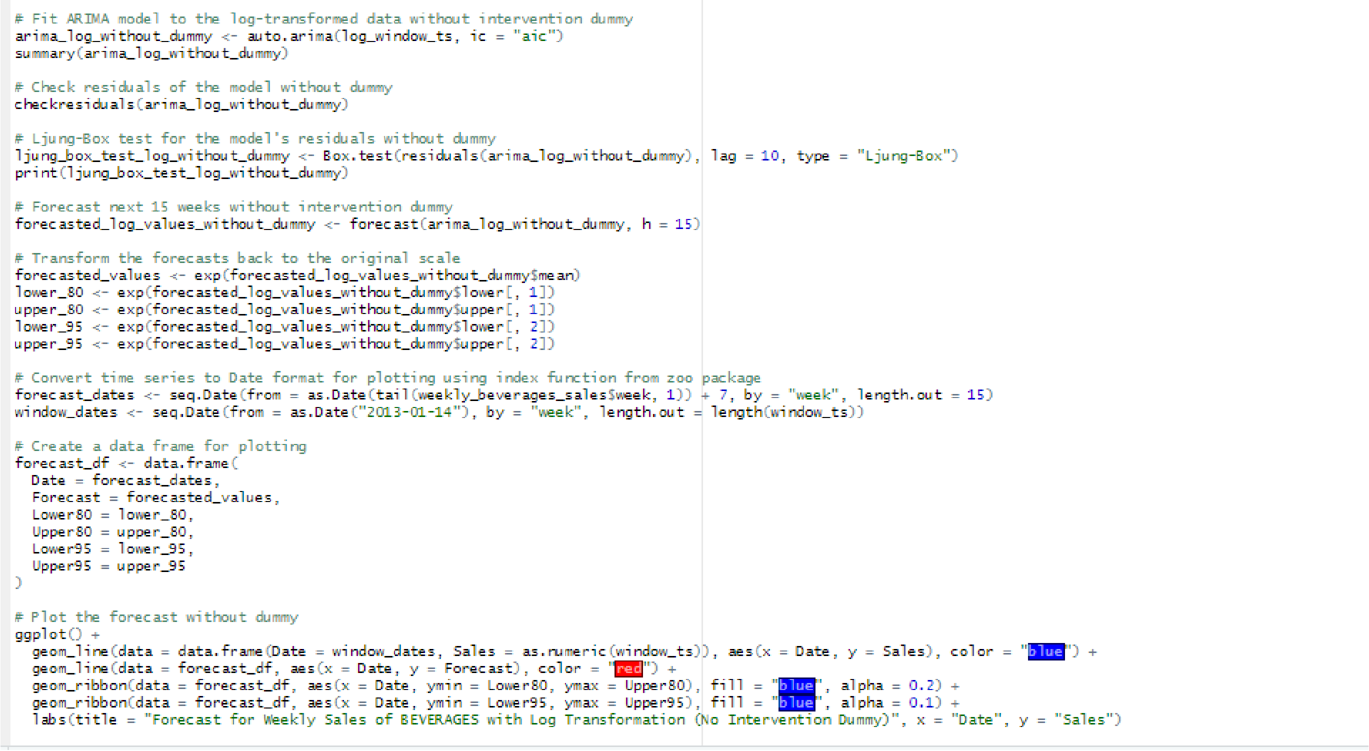
```r
# Plot the original time series
plot(window_ts, type = "l", col = "blue", lwd = 2,
     main = "Weekly Sales of BEVERAGES",
     xlab = "Year", ylab = "Weekly Sales")

# Add a horizontal line to help identify shifts in levels
abline(h = mean(window_ts), col = "red", lwd = 2)

# ACF and PACF Analysis
acf(window_ts, main = "ACF of Weekly Sales for BEVERAGES")
pacf(window_ts, main = "PACF of Weekly Sales for BEVERAGES")

# Fit ARIMA model to the original data without intervention dummy
arima_without_dummy <- auto.arima(window_ts, ic = "aic")
summary(arima_without_dummy)

# Check residuals of the model without dummy
checkresiduals(arima_without_dummy)

# Ljung-Box test for the model's residuals without dummy
ljung_box_test_without_dummy <- Box.test(residuals(arima_without_dummy), lag = 10, type = "Ljung-Box")
print(ljung_box_test_without_dummy)

# Forecast next 15 weeks without intervention dummy
forecasted_values_without_dummy <- forecast(arima_without_dummy, h = 15)
# Create date sequence for the forecast period
last_date <- as.Date(tail(weekly_beverages_sales$week, 1))
forecast_dates <- seq.Date(from = last_date + 7, by = "week", length.out = 15)
# Convert time indices of the historical data to dates
historical_dates <- seq.Date(from = as.Date("2013-01-14"), by = "week", length.out = length(window_ts))
# Create a data frame for plotting without dummy
forecast_df_without_dummy <- data.frame(
  Date = forecast_dates,
  Forecast = as.numeric(forecasted_values_without_dummy$mean),
  Lower80 = as.numeric(forecasted_values_without_dummy$lower[, 1]),
  Upper80 = as.numeric(forecasted_values_without_dummy$upper[, 1]),
  Lower95 = as.numeric(forecasted_values_without_dummy$lower[, 2]),
  Upper95 = as.numeric(forecasted_values_without_dummy$upper[, 2])
)
# Plot the forecast without dummy
ggplot() +
  geom_line(data = data.frame(Date = historical_dates, Sales = as.numeric(window_ts)), aes(x = Date, y = Sales), color = "blue") +
  geom_line(data = forecast_df_without_dummy, aes(x = Date, y = Forecast), color = "red") +
  geom_ribbon(data = forecast_df_without_dummy, aes(x = Date, ymin = Lower80, ymax = Upper80), fill = "blue", alpha = 0.2) +
  geom_ribbon(data = forecast_df_without_dummy, aes(x = Date, ymin = Lower95, ymax = Upper95), fill = "blue", alpha = 0.1) +
  labs(title = "Forecast for Weekly Sales of BEVERAGES without Intervention Dummy", x = "Date", y = "Sales")
```



```r
library(changepoint)
# Apply change point detection on the time series
cpt_mean <- cpt.mean(window_ts, method = "PELT")
# Plot the change points
plot(cpt_mean, main = "Change Point Detection in Weekly Sales of BEVERAGES")
# Extract the change points
change_points <- cpts(cpt_mean)
print(change_points)
# Manually set intervention points based on visual inspection
intervention_start_1 <- 104  # around end of 2014
intervention_start_2 <- 156  # around beginning of 2016
# Create intervention dummy variable
intervention_dummy <- ifelse(seq_along(window_ts) >= intervention_start_2, 2,
                             ifelse(seq_along(window_ts) >= intervention_start_1, 1, 0))
# Fit ARIMA models to the original data with intervention dummy
arima_with_dummy <- auto.arima(window_ts, xreg = intervention_dummy, ic = "aic")
# Print model summary
summary(arima_with_dummy)
# Check residuals of the model
checkresiduals(arima_with_dummy)
# Ljung-Box test for the model's residuals
ljung_box_test <- Box.test(residuals(arima_with_dummy), lag = 10, type = "Ljung-Box")
print(ljung_box_test)
# Create date sequence for the forecast period
last_date <- as.Date(tail(weekly_beverages_sales$week, 1))
forecast_dates <- seq.Date(from = last_date + 7, by = "week", length.out = 15)
# Convert time indices of the historical data to dates
historical_dates <- seq.Date(from = as.Date(start_window), by = "week", length.out = length(window_ts))
# Forecast next 15 weeks with intervention dummy
future_intervention_dummy <- rep(2, 15)
forecasted_values_with_dummy <- forecast(arima_with_dummy, h = 15, xreg = future_intervention_dummy)
# Create a data frame for plotting with dummy
forecast_df_with_dummy <- data.frame(
  Date = forecast_dates,
  Forecast = as.numeric(forecasted_values_with_dummy$mean),
  Lower80 = as.numeric(forecasted_values_with_dummy$lower[, 1]),
  Upper80 = as.numeric(forecasted_values_with_dummy$upper[, 1]),
  Lower95 = as.numeric(forecasted_values_with_dummy$lower[, 2]),
  Upper95 = as.numeric(forecasted_values_with_dummy$upper[, 2])
)
# Plot the forecast with dummy
ggplot() +
  geom_line(data = data.frame(Date = historical_dates, Sales = as.numeric(window_ts)), aes(x = Date, y = Sales), color = "blue") +
  geom_line(data = forecast_df_with_dummy, aes(x = Date, y = Forecast), color = "red") +
  geom_ribbon(data = forecast_df_with_dummy, aes(x = Date, ymin = Lower80, ymax = Upper80), fill = "blue", alpha = 0.2) +
  geom_ribbon(data = forecast_df_with_dummy, aes(x = Date, ymin = Lower95, ymax = Upper95), fill = "blue", alpha = 0.1) +
  labs(title = "Forecast for Weekly Sales of BEVERAGES with Intervention Dummy", x = "Date", y = "Sales")
```



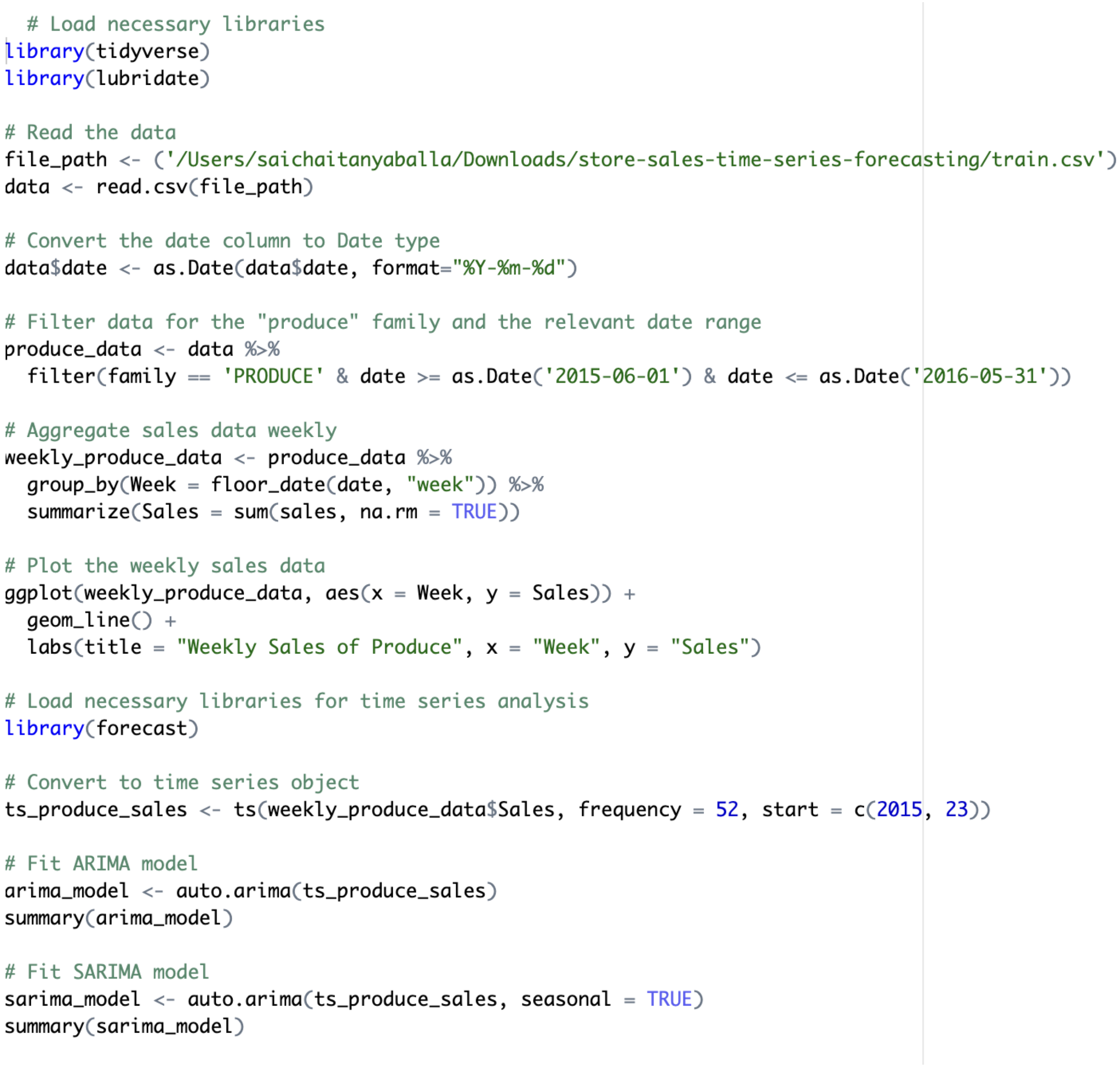
```r
# Log transform the data
log_window_ts <- log(window_ts)
# Fit ARIMA models to the log-transformed data with intervention dummy
arima_log_with_dummy <- auto.arima(log_window_ts, xreg = intervention_dummy, ic = "aic")

# Print model summary
summary(arima_log_with_dummy)

# Residual diagnostics
checkresiduals(arima_log_with_dummy)
# Ljung-Box test for residuals
ljung_box_test_log <- Box.test(residuals(arima_log_with_dummy), lag = 10, type = "Ljung-Box")
print(ljung_box_test_log)


# Forecast next 15 weeks with intervention dummy
future_intervention_dummy <- rep(2, 15)  # assuming the intervention effect continues
forecasted_log_values_with_dummy <- forecast(arima_log_with_dummy, h = 15, xreg = future_intervention_dummy)

# Transform the forecasts back to the original scale
forecasted_values <- exp(forecasted_log_values_with_dummy$mean)
lower_80 <- exp(forecasted_log_values_with_dummy$lower[, 1])
upper_80 <- exp(forecasted_log_values_with_dummy$upper[, 1])
lower_95 <- exp(forecasted_log_values_with_dummy$lower[, 2])
upper_95 <- exp(forecasted_log_values_with_dummy$upper[, 2])
# Convert time series to Date format for plotting using index function from zoo package
forecast_dates <- as.Date(index(forecasted_log_values_with_dummy$mean))
window_dates <- as.Date(index(window_ts))

# Create a data frame for plotting
forecast_df <- data.frame(
  Date = forecast_dates,
  Forecast = forecasted_values,
  Lower80 = lower_80,
  Upper80 = upper_80,
  Lower95 = lower_95,
  Upper95 = upper_95
)

# Plot the forecast
ggplot() +
  geom_line(data = data.frame(Date = window_dates, Sales = as.numeric(window_ts)), aes(x = Date, y = Sales), color = "blue") +
  geom_line(data = forecast_df, aes(x = Date, y = Forecast), color = "red") +
  geom_ribbon(data = forecast_df, aes(x = Date, ymin = Lower80, ymax = Upper80), fill = "blue", alpha = 0.2) +
  geom_ribbon(data = forecast_df, aes(x = Date, ymin = Lower95, ymax = Upper95), fill = "blue", alpha = 0.1) +
  labs(title = "Forecast for Weekly Sales of BEVERAGES with Log Transformation and Intervention", x = "Date", y = "Sales")
```



```
# Fit ARIMA model to the log-transformed data without intervention dummy
arima_log_without_dummy <- auto.arima(log_window_ts, ic = "aic")
summary(arima_log_without_dummy)

# Check residuals of the model without dummy
checkresiduals(arima_log_without_dummy)

# Ljung-Box test for the model's residuals without dummy
ljung_box_test_log_without_dummy <- Box.test(residuals(arima_log_without_dummy), lag = 10, type = "Ljung-Box")
print(ljung_box_test_log_without_dummy)

# Forecast next 15 weeks without intervention dummy
forecasted_log_values_without_dummy <- forecast(arima_log_without_dummy, h = 15)

# Transform the forecasts back to the original scale
forecasted_values <- exp(forecasted_log_values_without_dummy$mean)
lower_80 <- exp(forecasted_log_values_without_dummy$lower[, 1])
upper_80 <- exp(forecasted_log_values_without_dummy$upper[, 1])
lower_95 <- exp(forecasted_log_values_without_dummy$lower[, 2])
upper_95 <- exp(forecasted_log_values_without_dummy$upper[, 2])

# Convert time series to Date format for plotting using index function from zoo package
forecast_dates <- seq.Date(from = as.Date(tail(weekly_beverages_sales$week, 1)) + 7, by = "week", length.out = 15)
window_dates <- seq.Date(from = as.Date("2013-01-14"), by = "week", length.out = length(window_ts))

# Create a data frame for plotting
forecast_df <- data.frame(
  Date = forecast_dates,
  Forecast = forecasted_values,
  Lower80 = lower_80,
  Upper80 = upper_80,
  Lower95 = lower_95,
  Upper95 = upper_95
)

# Plot the forecast without dummy
ggplot() +
  geom_line(data = data.frame(Date = window_dates, Sales = as.numeric(window_ts)), aes(x = Date, y = Sales), color = "blue") +
  geom_line(data = forecast_df, aes(x = Date, y = Forecast), color = "red") +
  geom_ribbon(data = forecast_df, aes(x = Date, ymin = Lower80, ymax = Upper80), fill = "blue", alpha = 0.2) +
  geom_ribbon(data = forecast_df, aes(x = Date, ymin = Lower95, ymax = Upper95), fill = "blue", alpha = 0.1) +
  labs(title = "Forecast for Weekly Sales of BEVERAGES with Log Transformation (No Intervention Dummy)", x = "Date", y = "Sales")
```

Produce Series

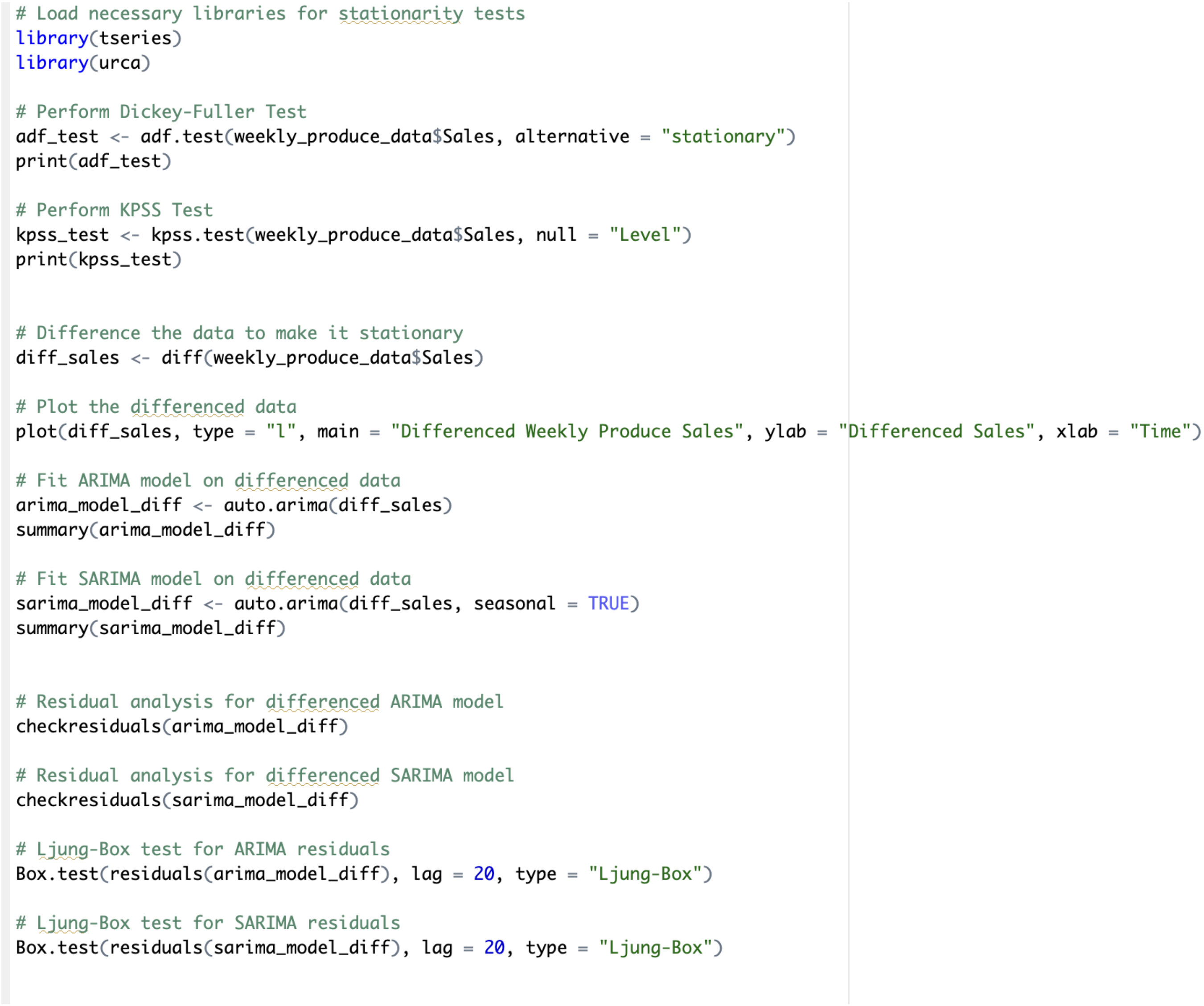
```r
####. ACF PACF AND EACF
# Load necessary libraries for time series analysis
library(forecast)
library(TSA)

# Plot ACF
acf(weekly_produce_data$Sales, main="ACF of Weekly Produce Sales")

# Plot PACF
pacf(weekly_produce_data$Sales, main="PACF of Weekly Produce Sales")

# EACF plot (EACF requires the TSA package)
eacf(weekly_produce_data$Sales)


## DICKEY KPSS

# Load necessary libraries for stationarity tests
library(tseries)
library(urca)

# Perform Dickey-Fuller Test
adf_test <- adf.test(weekly_produce_data$Sales, alternative = "stationary")
print(adf_test)

# Perform KPSS Test
kpss_test <- kpss.test(weekly_produce_data$Sales, null = "Level")
print(kpss_test)


# Difference the data to make it stationary
diff_sales <- diff(weekly_produce_data$Sales)

# Plot the differenced data
plot(diff_sales, type = "l", main = "Differenced Weekly Produce Sales", ylab = "Differenced Sales", xlab = "Time")

# Fit ARIMA model on differenced data
arima_model_diff <- auto.arima(diff_sales)
summary(arima_model_diff)
```



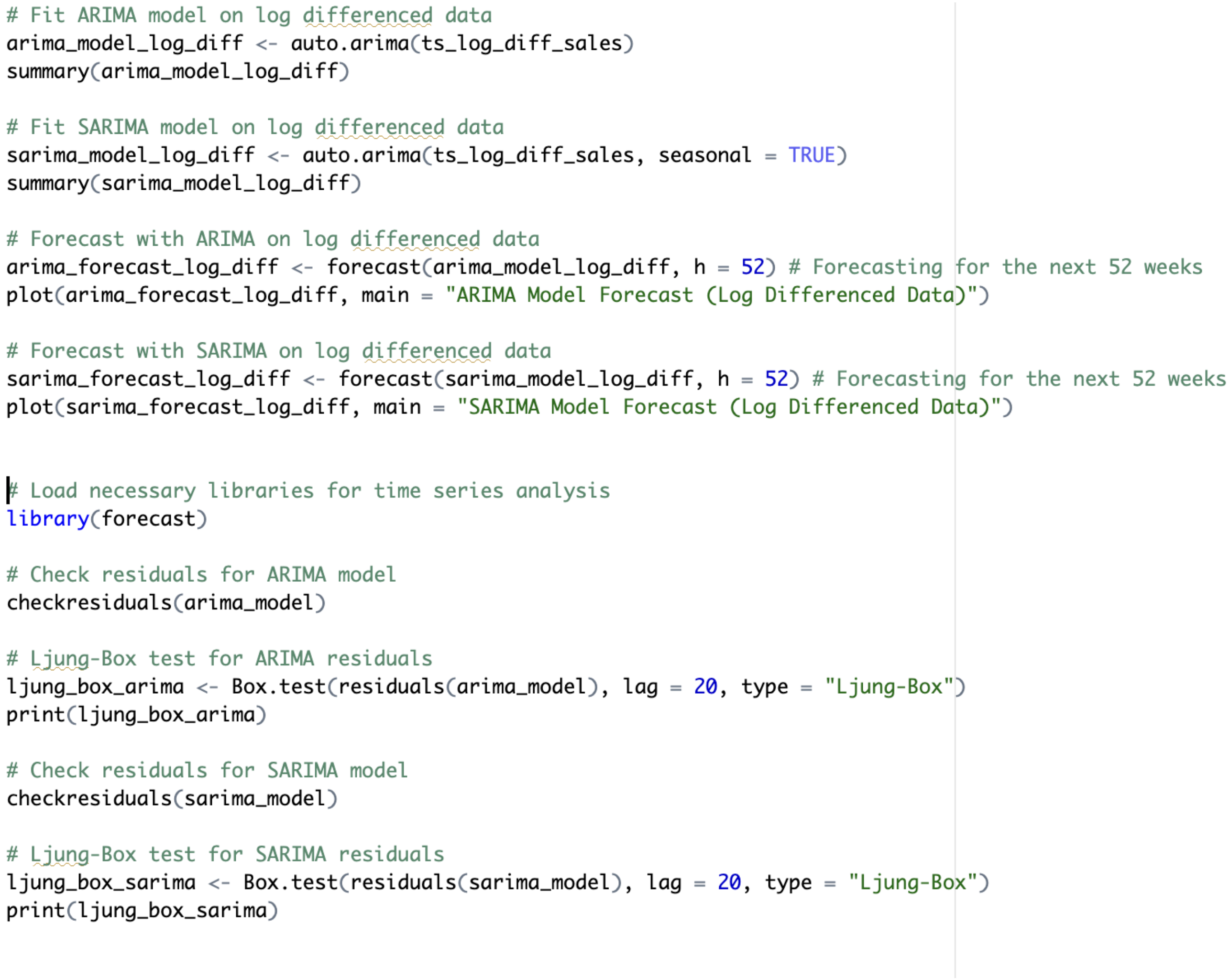
```r
  # Load necessary libraries
library(tidyverse)
library(lubridate)

# Read the data
file_path <- ('/Users/saichaitanyaballa/Downloads/store-sales-time-series-forecasting/train.csv')
data <- read.csv(file_path)

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

# Filter data for the "produce" family and the relevant date range
produce_data <- data %>%
  filter(family == 'PRODUCE' & date >= as.Date('2015-06-01') & date <= as.Date('2016-05-31'))

# Aggregate sales data weekly
weekly_produce_data <- produce_data %>%
  group_by(Week = floor_date(date, "week")) %>%
  summarize(Sales = sum(sales, na.rm = TRUE))

# Plot the weekly sales data
ggplot(weekly_produce_data, aes(x = Week, y = Sales)) +
  geom_line() +
  labs(title = "Weekly Sales of Produce", x = "Week", y = "Sales")

# Load necessary libraries for time series analysis
library(forecast)

# Convert to time series object
ts_produce_sales <- ts(weekly_produce_data$Sales, frequency = 52, start = c(2015, 23))

# Fit ARIMA model
arima_model <- auto.arima(ts_produce_sales)
summary(arima_model)

# Fit SARIMA model
sarima_model <- auto.arima(ts_produce_sales, seasonal = TRUE)
summary(sarima_model)
```

by applying the log differbcig to the model

```
# Load necessary libraries
library(tidyverse)
library(lubridate)
library(forecast)
library(tseries)
library(urca)

file_path <- ('/Users/saichaitanyaballa/Downloads/store-sales-time-series-forecasting/train.csv')
data <- read.csv(file_path)

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

# Filter data for the "produce" family and the relevant date range
produce_data <- data %>%
  filter(family == 'PRODUCE' & date >= as.Date('2015-06-01') & date <= as.Date('2016-12-31'))

# Aggregate sales data weekly
weekly_produce_data <- produce_data %>%
  group_by(Week = floor_date(date, "week")) %>%
  summarize(Sales = sum(sales, na.rm = TRUE))

# Log transformation and differencing
log_diff_sales <- diff(log(weekly_produce_data$Sales))

# Plot the log differenced data
plot(log_diff_sales, type = "l", main = "Log Differenced Weekly Produce Sales", ylab = "Log Differenced Sales", xlab

# Convert to time series object
ts_log_diff_sales <- ts(log_diff_sales, frequency = 52, start = c(2015, 23))

# Fit ARIMA model on log differenced data
arima_model_log_diff <- auto.arima(ts_log_diff_sales)
summary(arima_model_log_diff)
```



```r
# Load necessary libraries for stationarity tests
library(tseries)
library(urca)

# Perform Dickey-Fuller Test
adf_test <- adf.test(weekly_produce_data$Sales, alternative = "stationary")
print(adf_test)

# Perform KPSS Test
kpss_test <- kpss.test(weekly_produce_data$Sales, null = "Level")
print(kpss_test)


# Difference the data to make it stationary
diff_sales <- diff(weekly_produce_data$Sales)

# Plot the differenced data
plot(diff_sales, type = "l", main = "Differenced Weekly Produce Sales", ylab = "Differenced Sales", xlab = "Time")

# Fit ARIMA model on differenced data
arima_model_diff <- auto.arima(diff_sales)
summary(arima_model_diff)

# Fit SARIMA model on differenced data
sarima_model_diff <- auto.arima(diff_sales, seasonal = TRUE)
summary(sarima_model_diff)


# Residual analysis for differenced ARIMA model
checkresiduals(arima_model_diff)

# Residual analysis for differenced SARIMA model
checkresiduals(sarima_model_diff)

# Ljung-Box test for ARIMA residuals
Box.test(residuals(arima_model_diff), lag = 20, type = "Ljung-Box")

# Ljung-Box test for SARIMA residuals
Box.test(residuals(sarima_model_diff), lag = 20, type = "Ljung-Box")
```



```r
# Fit ARIMA model on log differenced data
arima_model_log_diff <- auto.arima(ts_log_diff_sales)
summary(arima_model_log_diff)

# Fit SARIMA model on log differenced data
sarima_model_log_diff <- auto.arima(ts_log_diff_sales, seasonal = TRUE)
summary(sarima_model_log_diff)

# Forecast with ARIMA on log differenced data
arima_forecast_log_diff <- forecast(arima_model_log_diff, h = 52) # Forecasting for the next 52 weeks
plot(arima_forecast_log_diff, main = "ARIMA Model Forecast (Log Differenced Data)")

# Forecast with SARIMA on log differenced data
sarima_forecast_log_diff <- forecast(sarima_model_log_diff, h = 52) # Forecasting for the next 52 weeks
plot(sarima_forecast_log_diff, main = "SARIMA Model Forecast (Log Differenced Data)")


# Load necessary libraries for time series analysis
library(forecast)

# Check residuals for ARIMA model
checkresiduals(arima_model)

# Ljung-Box test for ARIMA residuals
ljung_box_arima <- Box.test(residuals(arima_model), lag = 20, type = "Ljung-Box")
print(ljung_box_arima)

# Check residuals for SARIMA model
checkresiduals(sarima_model)

# Ljung-Box test for SARIMA residuals
ljung_box_sarima <- Box.test(residuals(sarima_model), lag = 20, type = "Ljung-Box")
print(ljung_box_sarima)
```