STORE SALES- TIME SERIES FORECASTING

Fall 2023

Group Members

Susheel Kumar-21569

Mustafa Qutbuddin-22055

Usama Hanif-21706

Sana Masood-21589

Warda Zainab-22147

Contents

Executive Summary

The project is based on data set "Store Sales – Time Series Forecasting" which comprises of data for numerous product families for a retail stores "Favorita" located in Ecuador. The objective of the project was to predict sales through variables such as promotions, holidays, event, income proxy and day of the week. Linear Regression Model was used to establish the relationship between sales (dependent variable) and the independent variables. Decision tree analysis was used a complementary choice. The results showed the predictive ability of linear regression to predict a few models' sales. However, the model's predictions were limited due to data limitation.

Dataset Description

Variable Name	Description	Type
Store_Nbr	Unique identifier for stores	Identifier
Family	Category of the product sold	Categorical
Date	Data for transaction	Date
On Promotion	Number of products on promotion	Numeric
Sales	Number of products purchased	Numeric
ID	Unique identifier for sale	Identifier
Holiday type	Working day or not	Categorical
Transferred	Was the holiday date changed	Binary
Oil Prices	The price for the oil on the day	Numeric
Events	Any special events like Black Friday	Factor

Assumptions

The primary model we chose to use for our sales forecasting was linear regression. Hence, here are the few assumptions we made before training the data using our model:

- Linearity: Y (dependent variable) can be expressed as a linear combination of all x variables (independent variables).
- No Multicollinearity: Each x variable, which is the part of the dataset doesn't explain or is explained by another x variable.

Problem Statement

It is essential to accurately predict and optimize sales in the retail industry for a business to flourish. The challenge in this dataset was to clean the data of different variables, engineer features, and test the impact on sales using Linear Regression model.

Detailed Methodology

Hypotheses

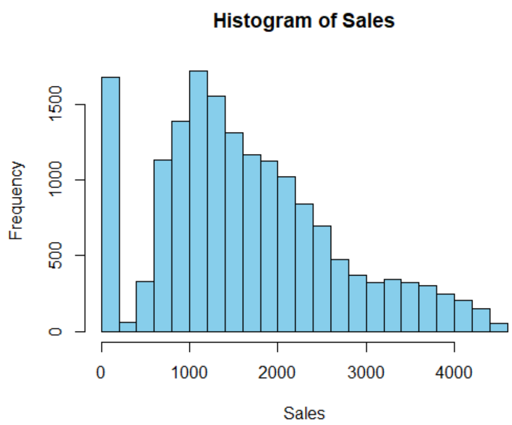
After examining the Store Sales – Time Series Forecasting dataset, we came up with a few hypotheses. Hypotheses are regarding the impact of the independent variables like promotion, event, salary payout, and oil prices on the sales (i.e. dependent variable). Through exploratory data analysis, we will test these hypotheses. The hypotheses are as follows:

1. Promotion: The higher the number of items on promotion in a store, the higher the store sales will be.
2. Salary payout: Sales will be higher after the salary payout days of the month. Salaries are paid out on the 1st and 15th day of the month.
3. Week of the Day: Weekend days (i.e. Saturday and Sunday) are expected to yield higher sales compared to weekdays.

EDA

Descriptive Statistics:

Target Variables – sales

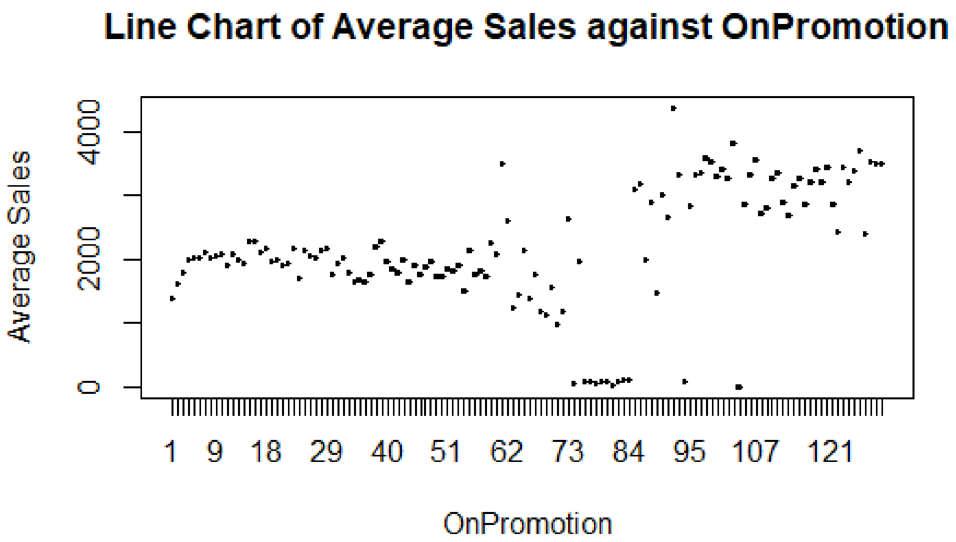



```
     sales
Min.   :      0.0
1st Qu.:      0.0
Median :     11.0
Mean   :    357.8
3rd Qu.:    195.8
Max.   :124717.0
```

As observed in the histogram and descriptive statistics, the sales of the stores were positively skewed. Mean of the data was much greater than the median. The minimum value is 0 while the maximum value of 124717 was recorded. It can be reasonably inferred that the data has outliers.

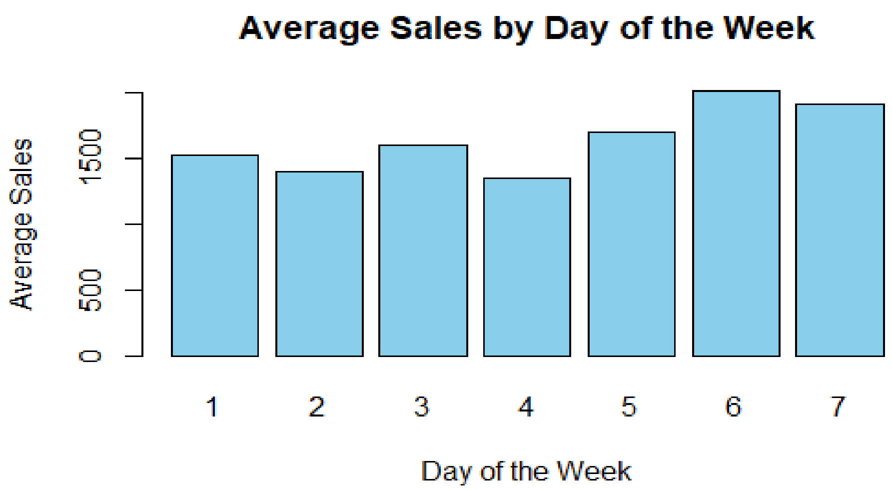
Independent Variables

- Promotions



The line chart above shows that sales is positively correlated with the number of products on promotion in the stores.

- Day of the week



The bar chart above shows that, on average, sales were higher on 6th and 7th day of the week (i.e. Saturday and Sunday).

The day of the week starts from Monday.

Data Cleaning and Transformations

Data Auditing and Transformation

Three csv files were imported from the dataset named: Holiday_Events, Train, and Oil Prices. These files were renamed for easier understanding. The data structure and summary were checked for appropriate transformation to be decided on. The data types were updated based on the nature of the data housed. Due to the limitation of time for the project, the top 2 stores and top 5 families were identified for further analysis.

The training data set was split into store and family level data sets. The data types of the variables were checked and changed into appropriate data types. The date was rectified in the proper format from the data frame of "Income". In the train data set, store-nbr, family, type, transferred and date were converted into categorical variables. The names of the columns were changed. In the data frame "Income", the column was named as 'date' and 'Oil_Prices'.

Data Cleaning

Columns Removal

It was broken up into different portions. Some of the columns which were not useful, and those columns were removed from the data frame. The extra columns named description, locale_name and locale were removed from the data frame "Holiday". Similarly, extra columns used for feature engineering in excel removed like Date, Month, and Year.

Based on data auditing, there were extreme values in the Sales tab. To remove these, rows with highest 1.5% and lowest 1.5% sales were counted except 0s (part of the trend).

Outliers

To replace these outliers. K means clustering technique was used to assign clusters to all the sale transactions. The data was divided into clusters, for example, the rows from 1-5 were assigned to cluster 1 and rows 6-10 to cluster 2. This helped identify the closest values to the outlier values based on distance. The mean was calculated of other values to replace the outlier values.

Oil Price – Proxy for Income

Oil price is an important variable in predicting sales. The use in this dataset was different as it was used as an income proxy because most of the population is involved in the oil industry. The dataset mentioned the date when oil prices changed so a transformation was used to map the oil prices till the date it was changed again to match the time in the main data frame.

Holiday Variable

The Holiday variable (Created in Feature Engineering) only covered dates from the "Holiday" data frame and the rest of the dates in the "Train" data frame were covered by replacing the N/As with 0 as those were working days.

Before merging the columns from each data frame, date formats were changed to a common format to avoid problems in merging the data frames.

Feature Engineering

Data Selection

The top 2 stores from the data set were selected out of which the top 5 families were further selected to reduce the data size based on the timelines of the project. A combination of stores and families is implemented to create 10 models since the effectiveness of a global model was in question

Holiday & Events

The "Holiday type" column started with 6 categories: Holiday, Additional, Bridge, Working day, transferred and Event. We created two factor variables: Holiday and Event.

Holiday column showed whether there was a working day on the date or not and compared it against the transferred column to make sure the date of the holdiay was not changed. No working day was represented by 1 and existed in case of Holiday, Additional, and Bridge type when the transferred condition was False. These covered local and regional holidays and extras like weekends or special days.

Event variable was separately created to record special days like Black Friday. If a particular date was an event, then the event variable recorded a value of 1 and if not then a 0.

Salary Payout

The dataset provided an additional piece of information that the public sector employees were paid on the 1st and 15th of every month. The data about the proportion of public sector employment in the country was missing. To test the impact, two factor variables were created ("Salary_3days" and "Salary_7days") where one recorded the dates within 3 days after each payment date while the other variable covered the 7 days after each payment date. The results were then tested to confirm which variable was significant.

Last 3 Sales average

The model wanted to test the impact of autocorrelation in predicting future sales. To test the effectiveness in predicting sales, the average of the last 3 sales was calculated. This was done in Excel by using the Average formula.

Day of the Week

To test the hypothesis, a variable was created to create a 7 level factor for each day in the week. This was built to incorporate the variation in sales due to the day of the week.

Model Building

Supervised and Unsupervised learning techniques were used. Linear Regression and Decision Tree Models were run. Correlation was run on R on numeric variables as it was not possible to run on factor variables.

Supervised Machine Learning Techniques

Linear Regression Model

The dataset had a time series data for stores with different variables like holidays and oil prices (as an income proxy) and family level data for what products were sold. Due to the time constraint, we focused on 5 top performing product families from the top 2 stores. Instead of using a time series trend-based model like Hierarchical Time Series forecasting, our group decided to build a Linear Regression Model to incorporate the effect of various factors from the dataset and to test their efficacy in predicting sales.

Using different factors like “Sales_3trans_avg”, “Oil Prices”, “Holiday”, “Event”, “Date”, “Day of the Week”, and “Onpromotion”, we built a Linear Regression model to predict sales for 10 categories (from 2 stores).

Unsupervised Machine Learning

Decision Tree

Features such as date, onpromotion, Holiday, Event, Oil_Prices, and Sales_3trans_avg are used to create decisions in the decision tree.

During the training phase, the tree learns how to split the data based on these features to minimize prediction errors. This allows the decision tree to follow the learned structure to predict values for each data point on the new data (testing set).

This allows the decision tree to make predictions by applying a set of conditions based on the input features until it reaches the terminal node, where the final prediction is received. This allows the decision tree to capture the complex relationships in the dataset and make predictions for the unseen data

Model Results

Metrics and KPIs: MAPE and MSE

10 Linear Regression Models MSE & MAPE

In this model, 2 stores were selected out of which were further modelled based on family. These were used to predict the sales.

The results of MAPE are extracted in percentage due to the limitation of the data available. MSE shows variation to some extent, supporting that the model can be used to predict in some models.

Data Set	MAPE	MSE
Beverages_46	1.00%	2298810
Cleaning_46	1.00%	616808.5
Dairy_46	1.00%	127006.1

Meats_46	1.00%	50273.86
Produce_46	1.00%	1315593
Cleaning_44	1.00%	360776.5
Dairy_44	1.00%	725642.8
Meats_44	1.00%	26237.64
Poultry_44	1.00%	58301.82
Bread_Bakery_44	1.00%	78641.74

10 Decision Tree Models MSE & MAPE

Data Set	MAPE	MSE
Beverages_46	0.996%	2168921
Cleaning_46	0.996%	542619.2
Dairy_46	0.996%	147190.8
Meats_46	0.996%	50390.62
Produce_46	0.996%	1213572
Cleaning_44	0.996%	378773.6
Dairy_44	0.996%	654824.8
Meats_44	0.996%	26428.96
Poultry_44	0.996%	54516.3
Bread_Bakery_44	0.996%	121691.3

Final Model

Linear regression was chosen as the final model. Decision tree was abandoned as it was a complementary choice. Number of considerations were being incorporated which made it intricate to use decision tree model and rely on its predictions.

Recommendations / Scope for Further Work

- The model is struggling with predictions due to the quality of data and their predictive capability as a result. This is reflected by the abnormally low MAPE levels across all the models. This can be worked on by introducing new variables and changes to existing variables to improve the model's performance.

- The current model relies on a simple Linear Regression model, and stepwise regression as the main model for prediction. This can be improved by using more sophisticated models which have relatively better performance.
- The project can be expanded by hierarchal time series forecasting or by collecting additional data like the locality to holidays to segregate the holidays across regions to create a better variable.

Changes in existing variables

Sales

- It had a lot of zeros (second highest in terms of frequency) which reduces the predictive capability of the model.
- This variable reflected sales in terms of quantities instead of value which reduces the quality of data since it would make it unreliable.

OnPromotion

- This variable lacked granularity as it only mentioned products on promotion for each family which makes it hard to predict if it resulted in higher sales.

Oil_Prices

- This variable was mentioned as a proxy for income, but it is a poor proxy since the changes in oil prices will not result in change in income and sales as a result.

New variables

- The existing variables can be improved by improving the Sales, Onpromotion, and Oil_Prices variable either through reduction in missing values or more granularity.
- New variables like income growth/GDP growth can be useful in predicting sales as some categories will have positive relationship while some will have negative relationship with these variables.

Conclusion

After taking a deep dive in this dataset, Linear regression proved to be better than Decision tree. the regression was successful in predicting sales for a few models. This was due the data limitation. Moreover, as Sir had suggested to use time series model but we had to use another approach. We implemented linear regression as variable impact can be shown in it. Whereas with time series analysis only trend could have been shown. However, the model can be expanded using the hierarchical time series by gathering more data.

Contribution Statement

NAME	ERP ID	CONTRIBUTIONS
Susheel Kumar	21569	1. Dataset Selection 2. Worked on cleaning, auditing, and building the model. 3. Report
Mustafa Qutbuddin	22055	1. Data set selection 2. Worked on excel (data cleaning and feature engineering). 3. Worked on R to build the model
Usama Hanif	21706	1. Dataset Selection 2. Worked on R to build the model. 3. Report
Sana Masood	21589	1. Dataset Selection 2. Worked on excel. 3. Report
Warda Zainab	22147	1. Dataset Selection 2. Worked on R 3. Report

Appendices

Summary

- **Train**

```
       id               date              store_nbr        family
 Min.   :      0   Length:3000888     Min.   : 1.0   Length:3000888
 1st Qu.: 750222   Class :character   1st Qu.:14.0   Class :character
 Median :1500444   Mode  :character   Median :27.5   Mode  :character
 Mean   :1500444                      Mean   :27.5
 3rd Qu.:2250665                      3rd Qu.:41.0
 Max.   :3000887                      Max.   :54.0
     sales            onpromotion
 Min.   :      0.0   Min.   :  0.000
 1st Qu.:      0.0   1st Qu.:  0.000
 Median :     11.0   Median :  0.000
 Mean   :    357.8   Mean   :  2.603
 3rd Qu.:    195.8   3rd Qu.:  0.000
 Max.   :124717.0   Max.   :741.000
```

- **Income**

```
     date             dcoilwtico
 Length:1218        Min.   : 26.19
 Class :character   1st Qu.: 46.41
 Mode  :character   Median : 53.19
                    Mean   : 67.71
                    3rd Qu.: 95.66
                    Max.   :110.62
                    NA's   :43
```

- **HolidayFE**

```
     date               type              locale           locale_name
 Length:350         Length:350         Length:350         Length:350
 Class :character   Class :character   Class :character   Class :character
 Mode  :character   Mode  :character   Mode  :character   Mode  :character


 description        transferred        Holiday             Event
 Length:350         Mode :logical   Min.   :0.0000   Min.   :0.00
 Class :character   FALSE:338       1st Qu.:1.0000   1st Qu.:0.00
 Mode  :character   TRUE :12        Median :1.0000   Median :0.00
                                    Mean   :0.7914   Mean   :0.16
                                    3rd Qu.:1.0000   3rd Qu.:0.00
                                    Max.   :1.0000   Max.   :1.00
```

K-Means Clustering

- Code on R
  #Creating a K-Means Cluster and assign cluster no to each data point
  sales_clust <- kmeans(Train[, c(5,6)], 150, iter.max = 10)
  Train$sales_clust <- sales_clust$cluster
  print(sales_clust)

- Output

```
K-means clustering with 150 clusters of sizes 35212, 10976, 9164, 15550, 10087, 10898, 3329, 2007, 4005, 50546, 8139, 15667, 9238, 1695, 14731, 1531, 3474, 17641,
6791, 479, 718, 33761, 2590, 9236, 12164, 25196, 2643, 2076, 9341, 14497, 13994, 14271, 5890, 8360, 4852, 8950, 3197, 9321, 1276, 1361, 14121, 15174, 7801, 15038,
8277, 15047, 1623, 4519, 63888, 3735, 1140806, 17370, 10711, 7066, 604, 15140, 9033, 11373, 11656, 3462, 9042, 5731, 4091, 6595, 3073, 13393, 1717, 1344, 8633, 14
800, 9053, 10197, 7417, 823, 813, 27962, 9475, 16519, 1091, 5194, 18714, 9446, 9937, 14034, 11567, 28708, 1667, 10087, 18213, 15261, 15272, 6591, 2762, 6724, 346
4, 221420, 3869, 3531, 6413, 7500, 5, 1689, 5164, 9631, 1365, 3638, 7118, 14173, 12214, 4135, 5923, 1984, 34, 13731, 5968, 9232, 7732, 6144, 2183, 11118, 58482, 9
580, 14269, 5839, 19564, 9372, 6744, 87565, 13465, 4621, 17327, 2451, 3132, 10064, 131957, 6351, 11245, 11548, 791, 9815, 694, 7074, 6240, 7768, 10859, 15978, 146
95, 6231, 825, 9720

Cluster means:
         sales  onpromotion
1   3.189010e+01 4.664603e-01
2   4.065582e+02 2.228225e+00
3   9.623581e+02 5.649716e+00
4   2.390648e+02 1.662637e+00
5   6.837395e+02 2.863587e+00
6   4.398721e+02 2.457974e+00
7   2.451330e+01 6.157705e+00
```

10 Linear Regression Models

```
Model for Beverages_46 :

Call:
lm(formula = sales ~ date + onpromotion + Holiday + Event + Oil_Prices +
    Sales_3trans_avg + Day_of_the_week, data = train_data)

Coefficients:
     (Intercept)              date       onpromotion          Holiday1            Event1
      22223.3009           -1.1382           -7.1054         -329.0435        -1525.8117
      Oil_Prices  Sales_3trans_avg  Day_of_the_week2  Day_of_the_week3  Day_of_the_week4
         -9.4484           -0.6313          350.3893          581.4585          736.9176
Day_of_the_week5  Day_of_the_week6  Day_of_the_week7
        632.2852         -445.3588         -825.4446

Model for Cleaning_46 :

Call:
lm(formula = sales ~ date + onpromotion + Holiday + Event + Oil_Prices +
    Sales_3trans_avg + Day_of_the_week, data = train_data)

Coefficients:
     (Intercept)              date       onpromotion          Holiday1            Event1
      1416.72755           0.04626           1.48641         571.11706        -202.62959
      Oil_Prices  Sales_3trans_avg  Day_of_the_week2  Day_of_the_week3  Day_of_the_week4
         0.53416          -0.19945        -138.45519        -159.41088        -329.91778
Day_of_the_week5  Day_of_the_week6  Day_of_the_week7
        57.42333        1005.04721        1018.95291
```

```
Model for Dairy_46 :

Call:
lm(formula = Sales ~ date + onpromotion + Holiday + Event + Oil_Prices +
    Sales_3trans_avg + Day_of_the_week, data = train_data)

Coefficients:
     (Intercept)              date       onpromotion          Holiday1            Event1
      -2.642e+03         2.431e-01         4.487e+00         3.160e+02        -4.645e+01
      Oil_Prices  Sales_3trans_avg  Day_of_the_week2  Day_of_the_week3  Day_of_the_week4
       1.201e+00        -2.473e-02        -2.629e+02        -2.851e+02        -3.997e+02
Day_of_the_week5  Day_of_the_week6  Day_of_the_week7
      -2.052e+02         6.270e+02         1.011e+03
```

```
Model for Meats_46 :

Call:
lm(formula = Sales ~ date + onpromotion + Holiday + Event + Oil_Prices +
    Sales_3trans_avg + Day_of_the_week, data = train_data)

Coefficients:
     (Intercept)              date       onpromotion          Holiday1            Event1
      2244.42860          -0.08131           3.53025          97.67880         -52.10045
      Oil_Prices  Sales_3trans_avg  Day_of_the_week2  Day_of_the_week3  Day_of_the_week4
         1.29465           0.01816        -119.38844         -37.06058        -155.06682
Day_of_the_week5  Day_of_the_week6  Day_of_the_week7
       576.00584         275.97723         329.02385
```

```
Model for Produce_46 :

Call:
lm(formula = Sales ~ date + onpromotion + Holiday + Event + Oil_Prices +
    Sales_3trans_avg + Day_of_the_week, data = train_data)

Coefficients:
     (Intercept)              date       onpromotion          Holiday1            Event1
      -1.851e+04         1.223e+00         8.531e+00        -7.818e+02         5.217e+02
      Oil_Prices  Sales_3trans_avg  Day_of_the_week2  Day_of_the_week3  Day_of_the_week4
      -1.918e+00         8.822e-02        -2.975e+02        -7.869e+02        -4.236e+02
Day_of_the_week5  Day_of_the_week6  Day_of_the_week7
      -2.480e+02         2.502e+02        -1.132e+03
```

```
Model for Cleaning_44 :

Call:
lm(formula = Sales ~ date + onpromotion + Holiday + Event + Oil_Prices +
    Sales_3trans_avg + Day_of_the_week, data = train_data)

Coefficients:
     (Intercept)              date       onpromotion          Holiday1            Event1
       4101.2483           -0.1301            0.7985           77.9359          -69.3348
      Oil_Prices  Sales_3trans_avg  Day_of_the_week2  Day_of_the_week3  Day_of_the_week4
          1.7293            0.2838         -237.0709          143.8039         -422.0845
Day_of_the_week5  Day_of_the_week6  Day_of_the_week7
        -88.0831          590.4611          520.3086
```

```
Model for Poultry_44 :

Call:
lm(formula = Sales ~ date + onpromotion + Holiday + Event + Oil_Prices +
    Sales_3trans_avg + Day_of_the_week, data = train_data)

Coefficients:
     (Intercept)              date       onpromotion          Holiday1            Event1
       739.01237          -0.01691           6.72721          49.86116          73.40399
      Oil_Prices  Sales_3trans_avg  Day_of_the_week2  Day_of_the_week3  Day_of_the_week4
         1.41999           0.32978         -92.94850         -42.64214         -99.81462
Day_of_the_week5  Day_of_the_week6  Day_of_the_week7
       225.52872         339.55995         490.34000
```

```
Model for Bread_Bakery_44 :

Call:
lm(formula = Sales ~ date + onpromotion + Holiday + Event + Oil_Prices +
    Sales_3trans_avg + Day_of_the_week, data = train_data)

Coefficients:
     (Intercept)              date       onpromotion          Holiday1            Event1
      -8.779e+03         5.842e-01         8.484e+00        -9.779e+01        -4.052e+01
      Oil_Prices  Sales_3trans_avg  Day_of_the_week2  Day_of_the_week3  Day_of_the_week4
       3.802e+00         6.148e-02        -1.518e+02         4.146e+01        -2.481e+02
Day_of_the_week5  Day_of_the_week6  Day_of_the_week7
       5.029e+01         5.457e+02         7.354e+02
```

```
Model for Dairy_44 :

Call:
lm(formula = Sales ~ date + onpromotion + Holiday + Event + Oil_Prices +
    Sales_3trans_avg + Day_of_the_week, data = train_data)

Coefficients:
     (Intercept)              date       onpromotion          Holiday1            Event1
       7582.2909           -0.3277            2.3933          376.0117           30.3906
      Oil_Prices  Sales_3trans_avg  Day_of_the_week2  Day_of_the_week3  Day_of_the_week4
         -3.7340            0.3300         -311.4665           -3.3764         -523.0780
Day_of_the_week5  Day_of_the_week6  Day_of_the_week7
         51.8236          733.1246          330.2690


Model for Meats_44 :

Call:
lm(formula = Sales ~ date + onpromotion + Holiday + Event + Oil_Prices +
    Sales_3trans_avg + Day_of_the_week, data = train_data)

Coefficients:
     (Intercept)              date       onpromotion          Holiday1            Event1
        165.1732            0.0086            1.1246           10.9716          -55.5244
      Oil_Prices  Sales_3trans_avg  Day_of_the_week2  Day_of_the_week3  Day_of_the_week4
          1.4747            0.1943          -27.6193          -12.5064          -47.2423
Day_of_the_week5  Day_of_the_week6  Day_of_the_week7
        590.1651          238.4737          250.4683
```

Correlation of Sales with Independent Variables

```
> # Check correlation with target variable
> cor(train_data[, "onpromotion"], train_data$Sales)
[1] 0.2996999
> cor(train_data[, "Holiday"], train_data$Sales)

> cor(train_data[, "Event"], train_data$Sales)

> cor(train_data[, "Oil_Prices"], train_data$Sales)
[1] -0.2575046
> cor(train_data[, "Sales_3trans_avg"], train_data$Sales)
[1] 0.4319045
```

Decision Tree Model

```
Model for Beverages_46 :
node), split, n, deviance, yval
      * denotes terminal node

 1) root 1400 3.999e+09 1626.00
   2) Day_of_the_week: 6,7 401 6.641e+08  583.10
     4) onpromotion < 2.5 134 4.581e+08 1488.00
       8) Sales_3trans_avg < 2217.84 76 2.716e+08 2043.00 *
       9) Sales_3trans_avg > 2217.84 58 1.323e+08  760.00 *
     5) onpromotion > 2.5 267 4.129e+07  129.00 *
   3) Day_of_the_week: 1,2,3,4,5 999 2.723e+09 2045.00
     6) onpromotion < 29.5 774 1.839e+09 2246.00
      12) Oil_Prices < 52.52 194 5.662e+08 2702.00
        24) Sales_3trans_avg < 1419.31 105 1.499e+08 3186.00 *
        25) Sales_3trans_avg > 1419.31 89 3.627e+08 2131.00 *
      13) Oil_Prices > 52.52 580 1.219e+09 2094.00
        26) onpromotion < 15.5 541 1.056e+09 2181.00 *
        27) onpromotion > 15.5 39 1.015e+08  880.90
          54) Day_of_the_week: 1,2,5 24 4.887e+04   68.08 *
          55) Day_of_the_week: 3,4 15 6.022e+07 2181.00 *
     7) onpromotion > 29.5 225 7.448e+08 1353.00
      14) Oil_Prices < 45.33 81 2.645e+08 2580.00
        28) Sales_3trans_avg < 1291.86 25 3.671e+06 3664.00 *
        29) Sales_3trans_avg > 1291.86 56 2.183e+08 2095.00 *
      15) Oil_Prices > 45.33 144 2.898e+08  663.20
        30) Sales_3trans_avg < 1508.42 74 2.319e+08 1177.00 *
        31) Sales_3trans_avg > 1508.42 70 1.770e+07  119.90 *
```

```
Model for Cleaning_46 :
node), split, n, deviance, yval
      * denotes terminal node

 1) root 1400 1.132e+09 2197.0
   2) Day_of_the_week: 1,2,3,4,5 999 2.909e+08 1865.0
     4) Sales_3trans_avg < 1026.33 308 1.258e+08 2162.0
       8) onpromotion < 0.5 42 2.746e+07 1611.0 *
       9) onpromotion > 0.5 266 8.356e+07 2249.0 *
     5) Sales_3trans_avg > 1026.33 691 1.258e+08 1733.0
      10) Day_of_the_week: 1,2,3,4 509 6.737e+07 1649.0 *
      11) Day_of_the_week: 5 182 4.467e+07 1968.0 *
   3) Day_of_the_week: 6,7 401 4.574e+08 3024.0
     6) Sales_3trans_avg < 1164.31 282 2.131e+08 3152.0
      12) Sales_3trans_avg < 992.99 146 9.610e+07 2878.0 *
      13) Sales_3trans_avg > 992.99 136 9.433e+07 3446.0 *
     7) Sales_3trans_avg > 1164.31 119 2.286e+08 2720.0
      14) Oil_Prices < 74.37 49 1.494e+08 2086.0
        28) Oil_Prices < 53.6 34 8.740e+07 2430.0
          56) Sales_3trans_avg < 1205.68 11 2.779e+06 3637.0 *
          57) Sales_3trans_avg > 1205.68 23 6.095e+07 1853.0
           114) Sales_3trans_avg < 1467.32 13 3.397e+07  957.9
             228) Sales_3trans_avg < 1247.77 6 2.242e+07 1975.0 *
             229) Sales_3trans_avg > 1247.77 7 1.141e+04   86.0 *
           115) Sales_3trans_avg > 1467.32 10 3.017e+06 3017.0 *
        29) Oil_Prices > 53.6 15 4.890e+07 1307.0 *
      15) Oil_Prices > 74.37 70 4.574e+07 3163.0 *
```

```
Model for Dairy_46 :
node), split, n, deviance, yval
      * denotes terminal node

 1) root 1400 523000000 1566
   2) Day_of_the_week: 1,2,3,4,5 999 116600000 1266
     4) Day_of_the_week: 2,3,4,5 799  75890000 1208
       8) onpromotion < 14.5 612  51020000 1155 *
       9) onpromotion > 14.5 187  17470000 1383 *
     5) Day_of_the_week: 1 200  27360000 1497
      10) Sales_3trans_avg < 874.384 5   3576000 2811 *
      11) Sales_3trans_avg > 874.384 195  14930000 1463 *
   3) Day_of_the_week: 6,7 401  94080000 2311
     6) Sales_3trans_avg < 1006.9 36  12770000 2966 *
     7) Sales_3trans_avg > 1006.9 365  64350000 2246
      14) Day_of_the_week: 6 191  20050000 2082 *
      15) Day_of_the_week: 7 174  33520000 2427
        30) Sales_3trans_avg < 1593.49 27   1680000 2008 *
        31) Sales_3trans_avg > 1593.49 147  26230000 2503 *
```

```
Model for Meats_46 :
node), split, n, deviance, yval
      * denotes terminal node

 1) root 1400 192800000 1151.0
   2) Day_of_the_week: 1,2,3,4 799  38930000  928.2
     4) Day_of_the_week: 2,4 400  13760000  869.4 *
     5) Day_of_the_week: 1,3 399  22410000  987.1
      10) Oil_Prices < 51.8 124   3491000  880.3 *
      11) Oil_Prices > 51.8 275  16870000 1035.0 *
   3) Day_of_the_week: 5,6,7 601  61790000 1446.0
     6) Day_of_the_week: 6,7 401  20940000 1313.0
      12) Oil_Prices < 52.73 125   3688000 1199.0 *
      13) Oil_Prices > 52.73 276  14920000 1364.0 *
     7) Day_of_the_week: 5 200  19390000 1714.0 *
```

```
Model for Produce_46 :
node), split, n, deviance, yval
      * denotes terminal node

  1) root 1400 2.936e+09 1711.0
    2) onpromotion < 0.5 704 1.338e+09 1118.0
      4) Oil_Prices < 51.58 141 1.997e+08 2185.0
        8) Sales_3trans_avg < 2355.43 125 1.224e+08 2428.0
         16) Day_of_the_week: 2,4,5,7 80 6.073e+07 2037.0 *
         17) Day_of_the_week: 1,3,6 45 2.762e+07 3124.0 *
        9) Sales_3trans_avg > 2355.43 16 1.218e+07  286.0 *
      5) Oil_Prices > 51.58 563 9.374e+08  851.0 *
    3) onpromotion > 0.5 696 1.100e+09 2311.0
      6) Sales_3trans_avg < 2544.37 656 9.024e+08 2418.0
       12) onpromotion < 175.5 531 7.365e+08 2247.0
         24) Day_of_the_week: 3,7 97 3.097e+08 1370.0
           48) Sales_3trans_avg < 1676.52 43 6.822e+07  666.7
             96) onpromotion < 91 35 9.352e+06  108.3 *
             97) onpromotion > 91 8 2.198e+05 3109.0 *
           49) Sales_3trans_avg > 1676.52 54 2.033e+08 1929.0
             98) Sales_3trans_avg < 2223.78 25 5.955e+07 3092.0 *
             99) Sales_3trans_avg > 2223.78 29 8.088e+07  927.4 *
         25) Day_of_the_week: 1,2,4,5,6 434 3.354e+08 2443.0
           50) Sales_3trans_avg < 1831.1 307 8.705e+07 2215.0 *
           51) Sales_3trans_avg > 1831.1 127 1.940e+08 2994.0
            102) Holiday: 0 118 1.278e+08 3167.0 *
            103) Holiday: 1 9 1.586e+07  715.3 *
       13) onpromotion > 175.5 125 8.469e+07 3143.0
         26) Sales_3trans_avg < 1975.64 120 4.473e+07 3242.0 *
         27) Sales_3trans_avg > 1975.64 5 1.050e+07  764.4 *
      7) Sales_3trans_avg > 2544.37 40 6.808e+07  565.3 *
```

```
Model for Cleaning_44 :
node), split, n, deviance, yval
      * denotes terminal node

 1) root 1400 646400000 2528
   2) Day_of_the_week: 1,2,3,4,5 999 205000000 2293
     4) Day_of_the_week: 2,4 400  64770000 2115 *
     5) Day_of_the_week: 1,3,5 599 119100000 2411
      10) Sales_3trans_avg < 1494.14 451  76460000 2335
        20) Sales_3trans_avg < 551.235 8   8830000 1414 *
        21) Sales_3trans_avg > 551.235 443  60730000 2351 *
      11) Sales_3trans_avg > 1494.14 148  31940000 2645 *
   3) Day_of_the_week: 6,7 401 247500000 3116
     6) onpromotion < 47.5 382 197500000 3149
      12) Sales_3trans_avg < 2294.29 302 160300000 3057 *
      13) Sales_3trans_avg > 2294.29 80  24830000 3499 *
     7) onpromotion > 47.5 19  41020000 2443 *
```

```
Model for Dairy_44 :
node), split, n, deviance, yval
      * denotes terminal node

 1) root 1400 999600000 2579.0
   2) Day_of_the_week: 1,2,3,4,5 999 198500000 2324.0
     4) Day_of_the_week: 2,4 400  56280000 2081.0
       8) Sales_3trans_avg < 2418.77 367  37860000 2034.0 *
       9) Sales_3trans_avg > 2418.77 33   8268000 2613.0 *
     5) Day_of_the_week: 1,3,5 599 103000000 2486.0
      10) Sales_3trans_avg < 1245.26 172  27540000 2179.0 *
      11) Sales_3trans_avg > 1245.26 427  52760000 2609.0 *
   3) Day_of_the_week: 6,7 401 573300000 3216.0
     6) Sales_3trans_avg < 3380.18 380 432300000 3303.0
      12) Sales_3trans_avg < 1178.47 16  36080000 1619.0 *
      13) Sales_3trans_avg > 1178.47 364 348800000 3378.0 *
     7) Sales_3trans_avg > 3380.18 21  85690000 1637.0
      14) onpromotion < 21 11  45620000 2297.0
        28) onpromotion < 5.5 6  21650000 1401.0 *
        29) onpromotion > 5.5 5  13370000 3372.0 *
      15) onpromotion > 21 10  30010000  910.4 *
```

```
Model for Meats_44 :
node), split, n, deviance, yval
      * denotes terminal node

 1) root 1400 137500000  975.0
   2) Day_of_the_week: 1,2,3,4 799  14730000  754.5
     4) Sales_3trans_avg < 2123.55 607   6823000  711.9 *
     5) Sales_3trans_avg > 2123.55 192   3318000  889.2 *
   3) Day_of_the_week: 5,6,7 601  32300000 1268.0
     6) Day_of_the_week: 6,7 401  13250000 1184.0 *
     7) Day_of_the_week: 5 200  10550000 1436.0
      14) Sales_3trans_avg < 2097.1 118   5289000 1346.0 *
      15) Sales_3trans_avg > 2097.1 82   2915000 1566.0 *
```

```
Model for Poultry_44 :
node), split, n, deviance, yval
      * denotes terminal node

 1) root 1400 266000000 1367.0
   2) Day_of_the_week: 1,2,3,4 799  41740000 1075.0
     4) Sales_3trans_avg < 1842.94 517  17200000  976.1
       8) Sales_3trans_avg < 1416.13 64   6273000  747.6 *
       9) Sales_3trans_avg > 1416.13 453   7117000 1008.0 *
     5) Sales_3trans_avg > 1842.94 282  10310000 1255.0
      10) Sales_3trans_avg < 2251.98 230   4087000 1199.0 *
      11) Sales_3trans_avg > 2251.98 52   2308000 1503.0 *
   3) Day_of_the_week: 5,6,7 601  64820000 1756.0
     6) Sales_3trans_avg < 2583.14 372  45870000 1657.0
      12) Oil_Prices < 99.525 363  39700000 1676.0 *
      13) Oil_Prices > 99.525 9    909900  901.8 *
     7) Sales_3trans_avg > 2583.14 229   9301000 1918.0 *
```

```
Model for Bread_Bakery_44 :
node), split, n, deviance, yval
      * denotes terminal node

 1) root 1400 317500000 1458
   2) Day_of_the_week: 1,2,3,4,5 999  87020000 1251
     4) onpromotion < 2.5 701  42810000 1153
       8) Day_of_the_week: 2,4 286  13440000 1036 *
       9) Day_of_the_week: 1,3,5 415  22850000 1233
        18) Oil_Prices < 88.96 262  11550000 1306 *
        19) Oil_Prices > 88.96 153   7506000 1108 *
     5) onpromotion > 2.5 298  21510000 1482
      10) Day_of_the_week: 2,4 114   5215000 1321 *
      11) Day_of_the_week: 1,3,5 184  11530000 1582 *
   3) Day_of_the_week: 6,7 401  80640000 1974
     6) onpromotion < 2.5 285  45530000 1850
      12) Sales_3trans_avg < 1775.39 178  23360000 1745 *
      13) Sales_3trans_avg > 1775.39 107  16950000 2025 *
     7) onpromotion > 2.5 116  19920000 2280
      14) Sales_3trans_avg < 2314.6 88  12690000 2181 *
      15) Sales_3trans_avg > 2314.6 28   3665000 2590 *
```