Forecasting Retail Sales

Studying the effects of past sales on future sales

Satish Gurram

2023-12-12

Introduction

About Me

Hello! My name is Satish, a recent graduate with a Master's degree in Analytics and Systems, with a strong passion for data analytics. My journey in data analytics began during my undergraduate studies where I developed a strong foundation in statistical analysis, data visualization, and programming languages such as R and Python. I am eager to apply my knowledge and skills in a professional setting and contribute to data-driven decision-making processes.

Throughout my academic career, I have completed several projects that showcase my ability to analyze complex datasets, derive meaningful insights, and present findings in a clear and concise manner. In an effort to build a portfolio of projects that showcase my analytical and technical skills, here is a project that aims at forecasting retail sales based on past sales.

Project Overview

In this project, I performed an in-depth analysis of retail sales data, which is an anonymized real-world data downloaded from Kaggle. The primary goal of this analysis was to predict future sales based on the last 3 years of sales across multiple departments for a few different stores. The analysis also includes analysis on the effect of markdowns during holidays on the overall sales.

Key tasks performed in this project include:

- **Data Cleaning**: Handling missing values, outliers, and ensuring data quality.
- **Exploratory Data Analysis (EDA)**: Using statistical methods and visualizations to understand the data.
- **Modeling**: Applying appropriate statistical or machine learning models to achieve the project objectives.
- **Interpretation**: Drawing actionable insights from the analysis and presenting them in an easy-to-understand format.

The detailed analysis and results are presented in the following sections.

Data Analysis

Initialize the Analysis

I prefer to keep track of all the packages that are used for the analysis and so the first thing in setting up this data analysis is loading the necessary packages. I will be using the following packages:

- ***dplyr***: For data manipulation

- ***ggplot2***: For data visualization
- ***lubridate***: For working with dates and times
- ***forecast***: For time series forecasting
- ***caret***: For machine learning and predictive modeling
- ***ggpubr***: For publication-ready plots

Load Data

In this section, we will load the datasets required for the analysis. The datasets include information about stores, additional features related to store activities, and historical sales data. These datasets provide a comprehensive view of the store operations, environmental factors, and sales performance.

Here is a brief overview of the three input datasets:

1. Stores

The **Stores** dataset contains anonymized information about 45 stores, including their type and size. This dataset helps us understand the structural differences between the stores which might influence sales performance.

2. Features

The **Features** dataset includes additional data related to the store, department, and regional activity for given dates. Key columns in this dataset are:

- **Store**: The store number
- **Date**: The week
- **Temperature**: Average temperature in the region
- **Fuel_Price**: Cost of fuel in the region
- **MarkDown1-5**: Anonymized data related to promotional markdowns, available only after Nov 2011 and not for all stores all the time (missing values marked as NA)
- **CPI**: The consumer price index
- **Unemployment**: The unemployment rate
- **IsHoliday**: Indicator whether the week is a special holiday week

This dataset provides crucial information on external factors that can affect store performance.

3. Sales

The **Sales** dataset contains historical sales data from 2010-02-05 to 2012-11-01. Key columns in this dataset are:

- **Store**: The store number
- **Dept**: The department number
- **Date**: The week
- **Weekly_Sales**: Sales for the given department in the given store
- **IsHoliday**: Indicator whether the week is a special holiday week

This dataset is essential for analyzing sales trends and identifying patterns related to holidays and other external factors.

```
# Load the data
dWD <- paste0(getwd(),"/data/")
stores_df <- read.csv(file.path(dWD, "stores data-set.csv"))
features_df <- read.csv(file.path(dWD, "Features data set.csv"))
sales_df <- read.csv(file.path(dWD, "sales data-set.csv"))

# Display the first few rows of each dataframe
head(stores_df)
```

```
##   Store Type   Size
## 1     1    A 151315
## 2     2    A 202307
## 3     3    B  37392
## 4     4    A 205863
## 5     5    B  34875
## 6     6    A 202505
```

```
head(features_df)
```

```
##   Store       Date Temperature Fuel_Price MarkDown1 MarkDown2 MarkDown3
## 1     1 05/02/2010       42.31      2.572        NA        NA        NA
## 2     1 12/02/2010       38.51      2.548        NA        NA        NA
## 3     1 19/02/2010       39.93      2.514        NA        NA        NA
## 4     1 26/02/2010       46.63      2.561        NA        NA        NA
## 5     1 05/03/2010       46.50      2.625        NA        NA        NA
## 6     1 12/03/2010       57.79      2.667        NA        NA        NA
##   MarkDown4 MarkDown5      CPI Unemployment IsHoliday
## 1        NA        NA 211.0964        8.106     FALSE
## 2        NA        NA 211.2422        8.106      TRUE
## 3        NA        NA 211.2891        8.106     FALSE
## 4        NA        NA 211.3196        8.106     FALSE
## 5        NA        NA 211.3501        8.106     FALSE
## 6        NA        NA 211.3806        8.106     FALSE
```

```
head(sales_df)
```

```
##   Store Dept       Date Weekly_Sales IsHoliday
## 1     1    1 05/02/2010     24924.50     FALSE
## 2     1    1 12/02/2010     46039.49      TRUE
## 3     1    1 19/02/2010     41595.55     FALSE
## 4     1    1 26/02/2010     19403.54     FALSE
## 5     1    1 05/03/2010     21827.90     FALSE
## 6     1    1 12/03/2010     21043.39     FALSE
```

Data Cleaning and Pre-processing

In this section, we will clean and pre-process the data to ensure it is ready for analysis. Data cleaning involves converting data types, handling missing values, and merging datasets to create a comprehensive dataset for analysis.

We will convert the date columns in the sales and features datasets to the Date type to facilitate date-based operations. The MarkDown columns in the features dataset contain missing values (NA). For the purpose of this analysis, we will replace these missing values with 0. Next, we will merge the sales, features, and stores datasets to create a single, unified dataframe. This merged dataframe will allow us to perform comprehensive analysis combining sales data, store attributes, and additional features.

```
# Convert date columns to Date type
sales_df$Date <- as.Date(sales_df$Date, format = "%d/%m/%Y")
features_df$Date <- as.Date(features_df$Date, format = "%d/%m/%Y")

# Handle missing values in markdown columns
features_df <- features_df %>%
  mutate(across(starts_with("MarkDown"), ~ ifelse(is.na(.), 0, .)))

# To filter sales by dates
# sales_df2 <- rbind(sales_df %>% filter(between(Date, as.Date('2010-09-01'), as.Date('2
010-12-31'))),
#                    sales_df %>% filter(between(Date, as.Date('2011-09-01'), as.Date('2
011-12-31'))),
#                    sales_df %>% filter(between(Date, as.Date('2012-09-01'), as.Date('2
012-12-31'))))

# Merge dataframes
merged_df <- sales_df %>%
  left_join(features_df, by = c("Store", "Date")) %>%
  left_join(stores_df, by = "Store")

head(merged_df)
```

```
##   Store Dept       Date Weekly_Sales IsHoliday.x Temperature Fuel_Price
## 1     1    1 2010-02-05     24924.50       FALSE       42.31      2.572
## 2     1    1 2010-02-12     46039.49        TRUE       38.51      2.548
## 3     1    1 2010-02-19     41595.55       FALSE       39.93      2.514
## 4     1    1 2010-02-26     19403.54       FALSE       46.63      2.561
## 5     1    1 2010-03-05     21827.90       FALSE       46.50      2.625
## 6     1    1 2010-03-12     21043.39       FALSE       57.79      2.667
##   MarkDown1 MarkDown2 MarkDown3 MarkDown4 MarkDown5      CPI Unemployment
## 1         0         0         0         0         0 211.0964        8.106
## 2         0         0         0         0         0 211.2422        8.106
## 3         0         0         0         0         0 211.2891        8.106
## 4         0         0         0         0         0 211.3196        8.106
## 5         0         0         0         0         0 211.3501        8.106
## 6         0         0         0         0         0 211.3806        8.106
##   IsHoliday.y Type   Size
## 1       FALSE    A 151315
## 2        TRUE    A 151315
## 3       FALSE    A 151315
## 4       FALSE    A 151315
## 5       FALSE    A 151315
## 6       FALSE    A 151315
```

Feature Engineering

In this section, we will perform feature engineering to create new variables that can provide additional insights and improve the predictive power of our models. Feature engineering involves creating new features from the existing data to highlight important patterns and relationships.

We will create new binary features to indicate whether a given week corresponds to major holidays such as the Super Bowl, Labor Day, Thanksgiving, and Christmas. Additionally, we will create a feature to indicate whether a week is a major holiday week, encompassing any of these holidays.

```
# Create new features for holiday weeks
merged_df <- merged_df %>%
  mutate(Week = week(Date),
         Year = year(Date),
         IsSuperBowl = ifelse(Week == 6, 1, 0),
         IsLaborDay = ifelse(Week == 36, 1, 0),
         IsThanksgiving = ifelse(Week == 47, 1, 0),
         IsChristmas = ifelse(Week == 52, 1, 0),
         IsMajorHoliday = ifelse(IsSuperBowl + IsLaborDay + IsThanksgiving + IsChristmas
> 0, 1, 0))

head(merged_df)
```

```
##   Store Dept       Date Weekly_Sales IsHoliday.x Temperature Fuel_Price
## 1     1    1 2010-02-05     24924.50       FALSE       42.31      2.572
## 2     1    1 2010-02-12     46039.49        TRUE       38.51      2.548
## 3     1    1 2010-02-19     41595.55       FALSE       39.93      2.514
## 4     1    1 2010-02-26     19403.54       FALSE       46.63      2.561
## 5     1    1 2010-03-05     21827.90       FALSE       46.50      2.625
## 6     1    1 2010-03-12     21043.39       FALSE       57.79      2.667
##   MarkDown1 MarkDown2 MarkDown3 MarkDown4 MarkDown5      CPI Unemployment
## 1         0         0         0         0         0 211.0964        8.106
## 2         0         0         0         0         0 211.2422        8.106
## 3         0         0         0         0         0 211.2891        8.106
## 4         0         0         0         0         0 211.3196        8.106
## 5         0         0         0         0         0 211.3501        8.106
## 6         0         0         0         0         0 211.3806        8.106
##   IsHoliday.y Type   Size Week Year IsSuperBowl IsLaborDay IsThanksgiving
## 1       FALSE    A 151315    6 2010           1          0              0
## 2        TRUE    A 151315    7 2010           0          0              0
## 3       FALSE    A 151315    8 2010           0          0              0
## 4       FALSE    A 151315    9 2010           0          0              0
## 5       FALSE    A 151315   10 2010           0          0              0
## 6       FALSE    A 151315   11 2010           0          0              0
##   IsChristmas IsMajorHoliday
## 1           0              1
## 2           0              0
## 3           0              0
## 4           0              0
## 5           0              0
## 6           0              0
```

Predicting Department-Wide Sales

In this section, we will use historical sales data to predict future department-wide sales. This involves aggregating sales data, splitting the data into training and testing sets, training a time series model, and making forecasts.

The following steps are necessary for this analysis:

- ***Aggregate Sale Data***: We will aggregate the weekly sales data by store, department, year, and week to get a comprehensive view of sales trends over time.
- ***Split Data into Training and Testing Sets***: To evaluate our model's performance, we will split the aggregated sales data into training (80%) and testing (20%) sets. The training set will be used to train the model, while the testing set will be used to validate its accuracy.
- ***Train a Time Series Model***: Using the training data, we will train a time series model to capture the underlying patterns and trends in the sales data. We will use the auto.arima function to automatically select the best ARIMA (Auto-Regressive Integrated Moving Average) model, which is best suited to predict future values based on past values.
- ***Forecast Sales for the Next Year***: Once the model is trained, we will use it to forecast sales for the next year. The forecasted sales will be visualized to provide insights into expected future sales trends.

```
# Aggregate weekly sales data by Store and Dept
agg_sales <- merged_df %>%
  group_by(Store, Dept, Year, Week) %>%
  summarize(Weekly_Sales = sum(Weekly_Sales))
```

```
## `summarise()` has grouped output by 'Store', 'Dept', 'Year'. You can override
## using the `.groups` argument.
```

```
# Split data into training and testing sets
set.seed(123)
trainIndex <- createDataPartition(agg_sales$Weekly_Sales, p = 0.8, list = FALSE, times =
1)
train_data <- agg_sales[trainIndex,]
test_data <- agg_sales[-trainIndex,]

# Train a time series model
ts_train <- ts(train_data$Weekly_Sales, frequency = 52)
fit <- auto.arima(ts_train)

# Forecast sales for the next year
forecasted_sales <- forecast(fit, h = 52)
autoplot(forecasted_sales) + scale_y_log10() + theme_bw()
```

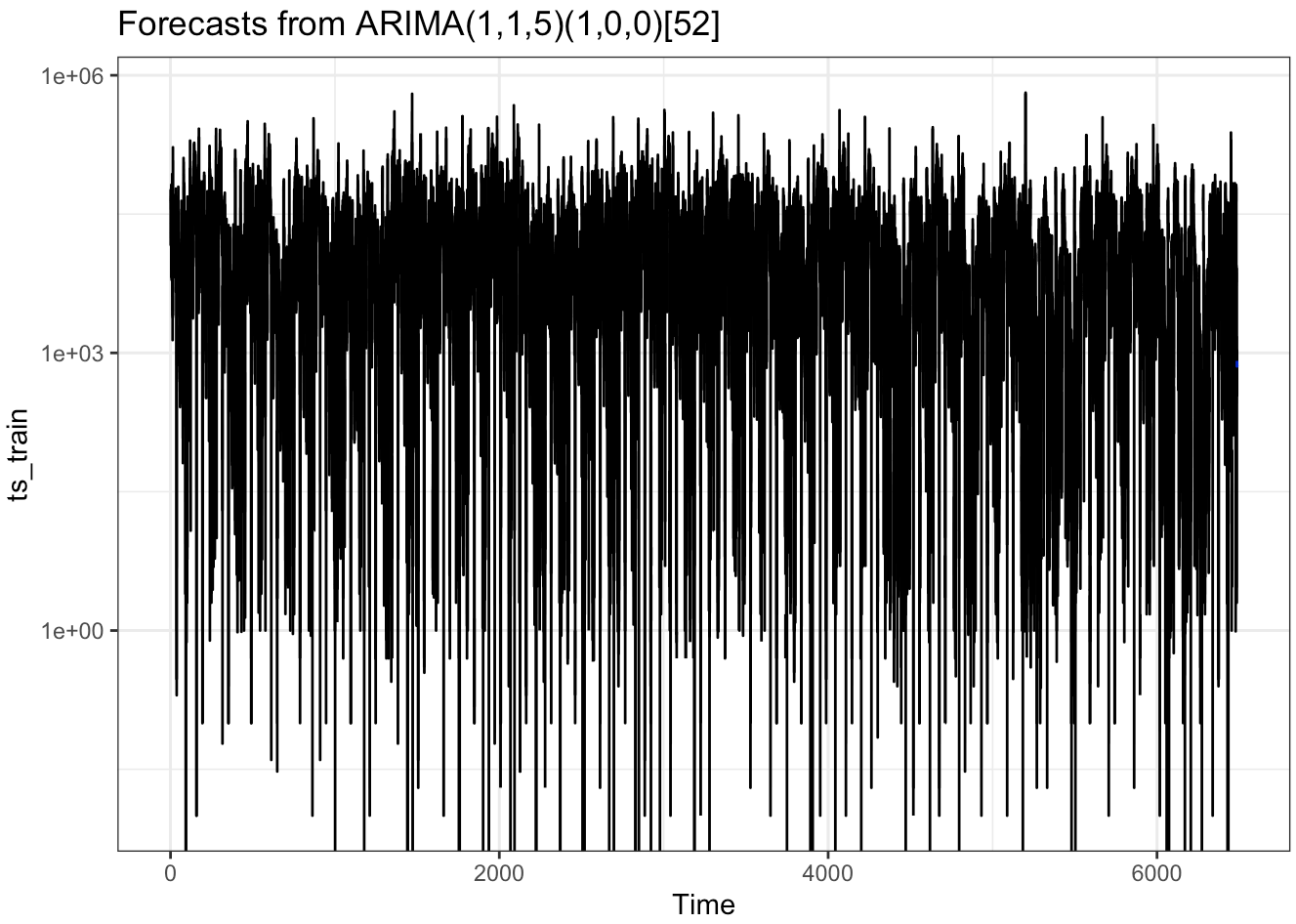
```
## Warning in transformation$transform(x): NaNs produced
```

```
## Warning in scale_y_log10(): log-10 transformation introduced infinite values.
```

```
## Warning in transformation$transform(x): NaNs produced
```

```
## Warning in scale_y_log10(): log-10 transformation introduced infinite values.
```

```
## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning
## -Inf
## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning
## -Inf
```



Modeling the effects of markdowns on holiday weeks

In this section, we will analyze the impact of promotional markdowns on sales during holiday weeks. By building a linear model, we aim to quantify how markdowns influence sales performance, particularly during major holidays.

We will create a linear regression model to examine the relationship between markdowns and weekly sales. The model will include the five markdown variables (MarkDown1, MarkDown2, MarkDown3, MarkDown4, and MarkDown5) and a binary indicator for major holiday weeks (IsMajorHoliday). This will help us understand the combined and individual effects of markdowns and holidays on sales.

```
# Linear model to analyze the effect of markdowns on sales during holiday weeks
markdown_effect_model <- lm(Weekly_Sales ~ MarkDown1 + MarkDown2 + MarkDown3 + MarkDown4
+ MarkDown5 + IsMajorHoliday, data = merged_df)
summary(markdown_effect_model)
```

```
##
## Call:
## lm(formula = Weekly_Sales ~ MarkDown1 + MarkDown2 + MarkDown3 +
##     MarkDown4 + MarkDown5 + IsMajorHoliday, data = merged_df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -36999 -13775  -8274   4234 677916
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     1.518e+04  4.035e+01 376.315  < 2e-16 ***
## MarkDown1       1.241e-01  1.124e-02  11.046  < 2e-16 ***
## MarkDown2       3.676e-02  7.180e-03   5.120 3.06e-07 ***
## MarkDown3       1.413e-01  6.575e-03  21.485  < 2e-16 ***
## MarkDown4      -8.781e-03  1.657e-02  -0.530    0.596
## MarkDown5       1.880e-01  9.191e-03  20.456  < 2e-16 ***
## IsMajorHoliday  1.066e+03  1.454e+02   7.331 2.29e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22650 on 421563 degrees of freedom
## Multiple R-squared:  0.005025,   Adjusted R-squared:  0.005011
## F-statistic: 354.8 on 6 and 421563 DF,  p-value: < 2.2e-16
```

After fitting the linear model, we will examine the summary output to interpret the coefficients and determine the statistical significance of each variable. The summary will provide insights into:

- The direction and magnitude of the effect of each markdown variable on weekly sales.
- The significance of the holiday indicator, showing how sales are impacted during major holidays.
- Overall model performance, including R-squared and p-values for each predictor.

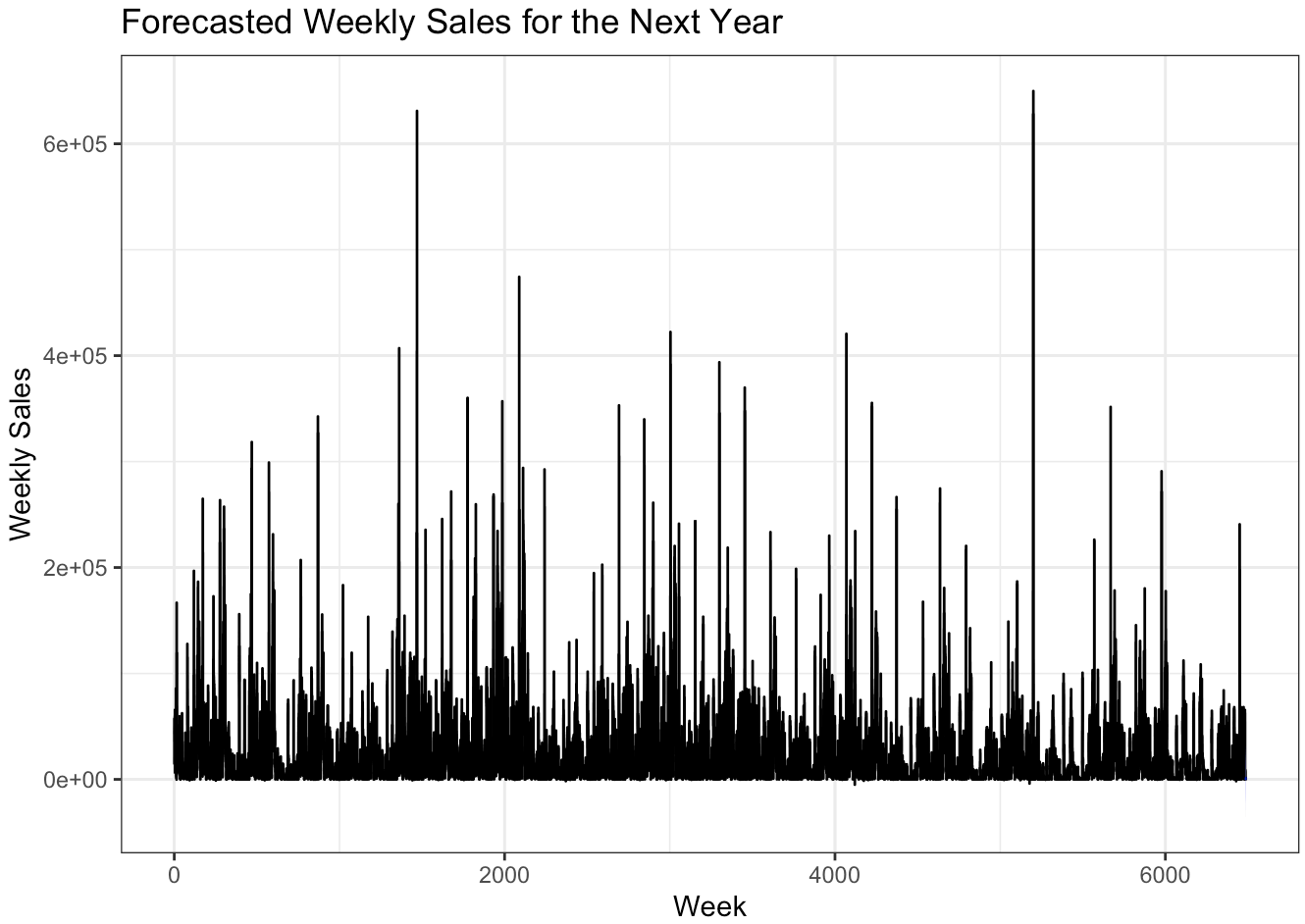
Visualizations and Recommendations

In this section, we present visualizations to provide insights into the sales forecast for the next year and the effects of markdowns on weekly sales during holiday weeks. Based on these visualizations, we offer recommendations for optimizing sales strategies.

Sales forecast for the next year

The following plot shows the forecasted weekly sales for the next year. This visualization helps in understanding the expected sales trends and planning accordingly.

```
autoplot(forecasted_sales) +
  ggtitle("Forecasted Weekly Sales for the Next Year") +
  xlab("Week") +
  ylab("Weekly Sales") +
  theme_bw()
```



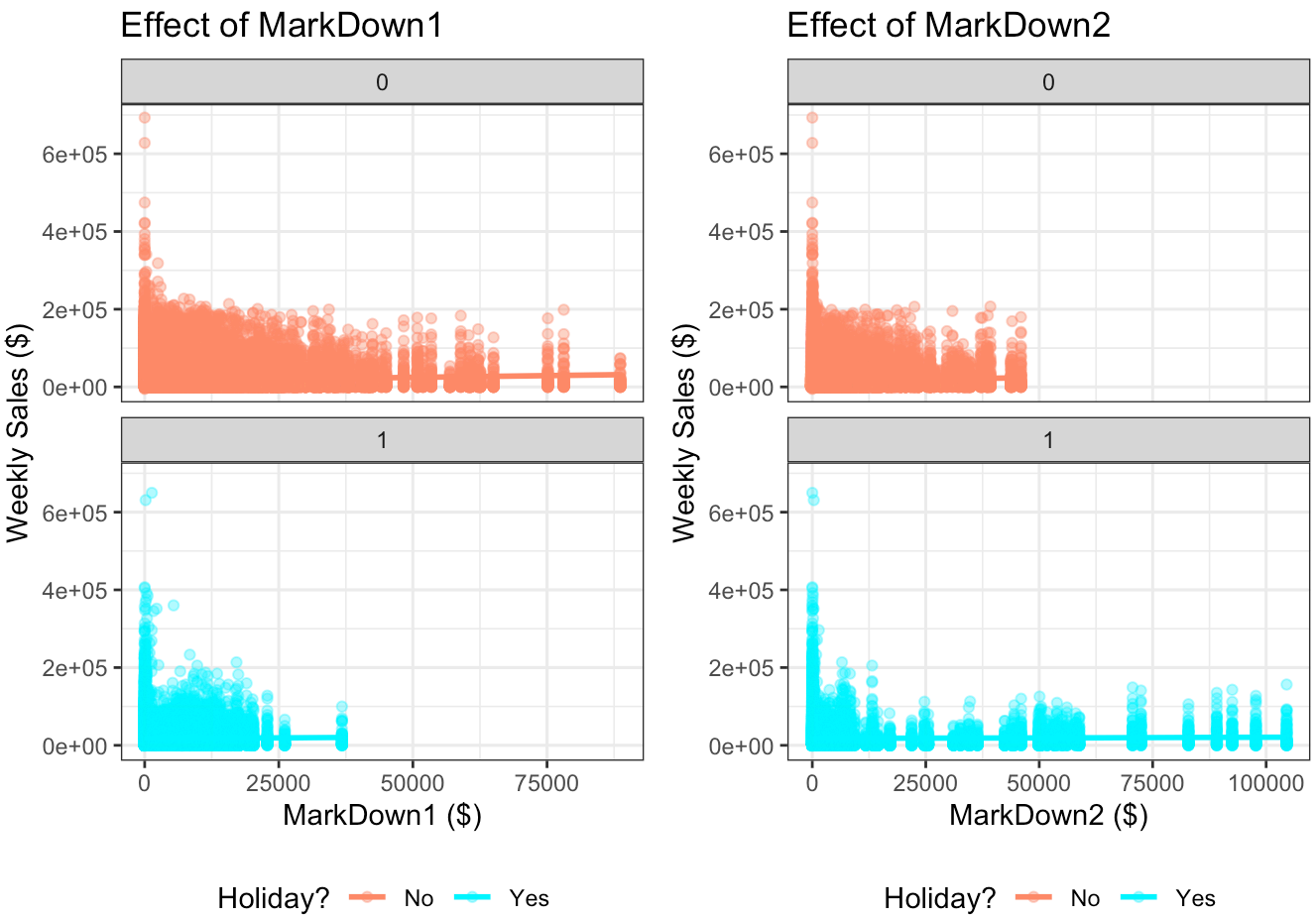
Effect of Markdowns on Holiday Weeks

The following series of plots illustrate the effects of different markdowns on weekly sales during holiday weeks. These visualizations help in assessing the impact of promotional markdowns and can guide the implementation of effective marketing strategies during key holiday periods.

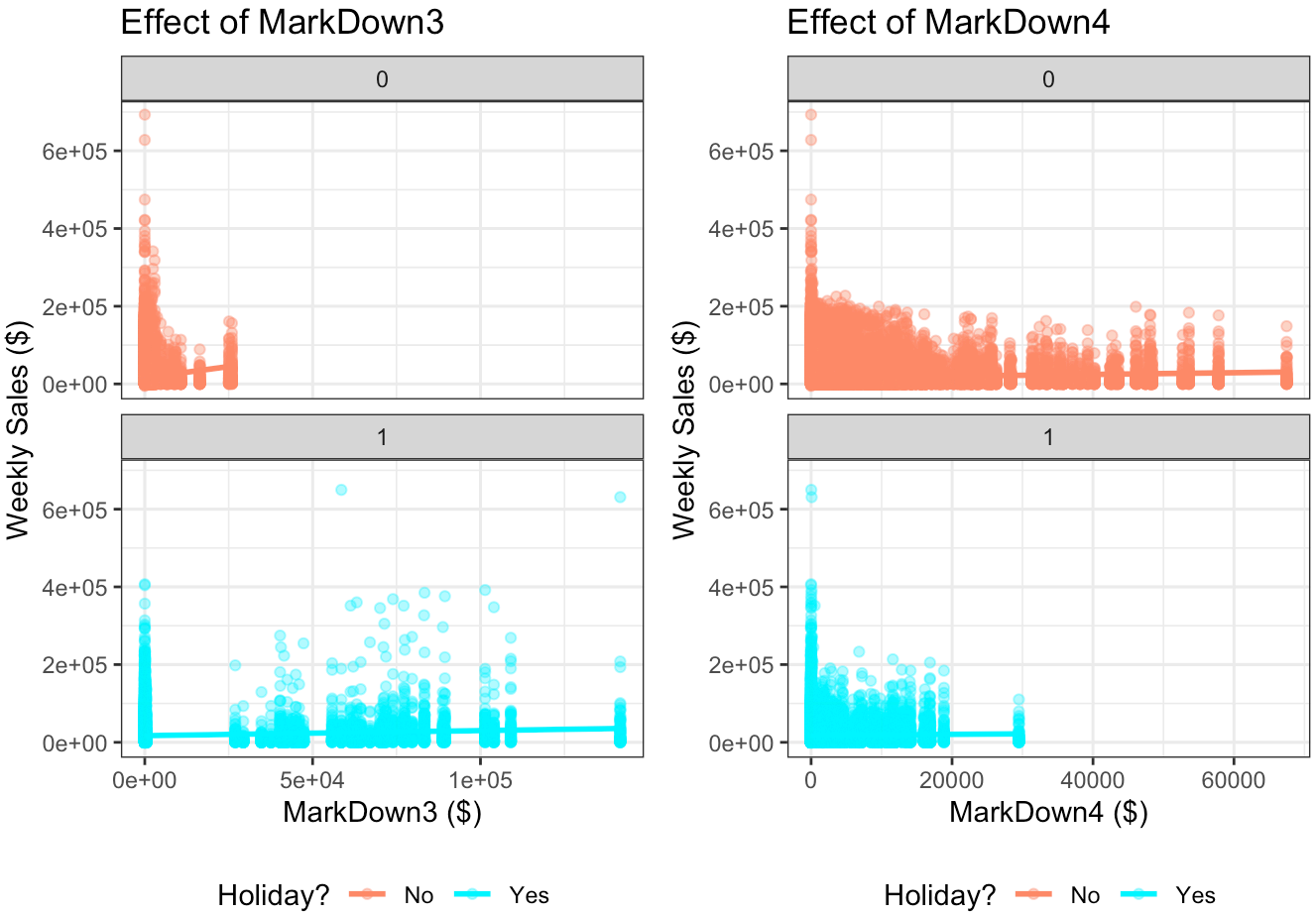
```
g1 <- ggplot(merged_df, aes(x = MarkDown1, y = Weekly_Sales, color = factor(IsMajorHolid
ay))) +
  geom_point(alpha = 0.4) +
  scale_color_manual(values = c("salmon1","turquoise1"), labels = c("No","Yes")) +
  geom_smooth(method = "lm", se = FALSE) +
  ggtitle("Effect of MarkDown1") +
  labs(x = "MarkDown1 ($)", y = "Weekly Sales ($)",
       color = "Holiday?") +
  theme_bw() + facet_wrap(~IsMajorHoliday, ncol = 1) +
  theme(legend.position = "bottom")

g2 <- ggplot(merged_df, aes(x = MarkDown2, y = Weekly_Sales, color = factor(IsMajorHolid
ay))) +
  geom_point(alpha = 0.4) +
  scale_color_manual(values = c("salmon1","turquoise1"), labels = c("No","Yes")) +
  geom_smooth(method = "lm", se = FALSE) +
  ggtitle("Effect of MarkDown2") +
  labs(x = "MarkDown2 ($)", y = "Weekly Sales ($)",
       color = "Holiday?") +
  theme_bw() + facet_wrap(~IsMajorHoliday, ncol = 1) +
  theme(legend.position = "bottom")

g3 <- ggplot(merged_df, aes(x = MarkDown3, y = Weekly_Sales, color = factor(IsMajorHolid
ay))) +
  geom_point(alpha = 0.4) +
  scale_color_manual(values = c("salmon1","turquoise1"), labels = c("No","Yes")) +
  geom_smooth(method = "lm", se = FALSE) +
  ggtitle("Effect of MarkDown3") +
  labs(x = "MarkDown3 ($)", y = "Weekly Sales ($)",
       color = "Holiday?") +
  theme_bw() + facet_wrap(~IsMajorHoliday, ncol = 1) +
  theme(legend.position = "bottom")

g4 <- ggplot(merged_df, aes(x = MarkDown4, y = Weekly_Sales, color = factor(IsMajorHolid
ay))) +
  geom_point(alpha = 0.4) +
  scale_color_manual(values = c("salmon1","turquoise1"), labels = c("No","Yes")) +
  geom_smooth(method = "lm", se = FALSE) +
  ggtitle("Effect of MarkDown4") +
  labs(x = "MarkDown4 ($)", y = "Weekly Sales ($)",
       color = "Holiday?") +
  theme_bw() + facet_wrap(~IsMajorHoliday, ncol = 1) +
  theme(legend.position = "bottom")

g5 <- ggplot(merged_df, aes(x = MarkDown5, y = Weekly_Sales, color = factor(IsMajorHolid
ay))) +
  geom_point(alpha = 0.4) +
  scale_color_manual(values = c("salmon1","turquoise1"), labels = c("No","Yes")) +
  geom_smooth(method = "lm", se = FALSE) +
  ggtitle("Effect of MarkDown5") +
  labs(x = "MarkDown5 ($)", y = "Weekly Sales ($)",
       color = "Holiday?") +
```

```
  theme_bw() + facet_wrap(~IsMajorHoliday, ncol = 1) +
  theme(legend.position = "bottom")

gg <- ggarrange(g1, g2, g3, g4, g5, ncol = 2);

gg
```

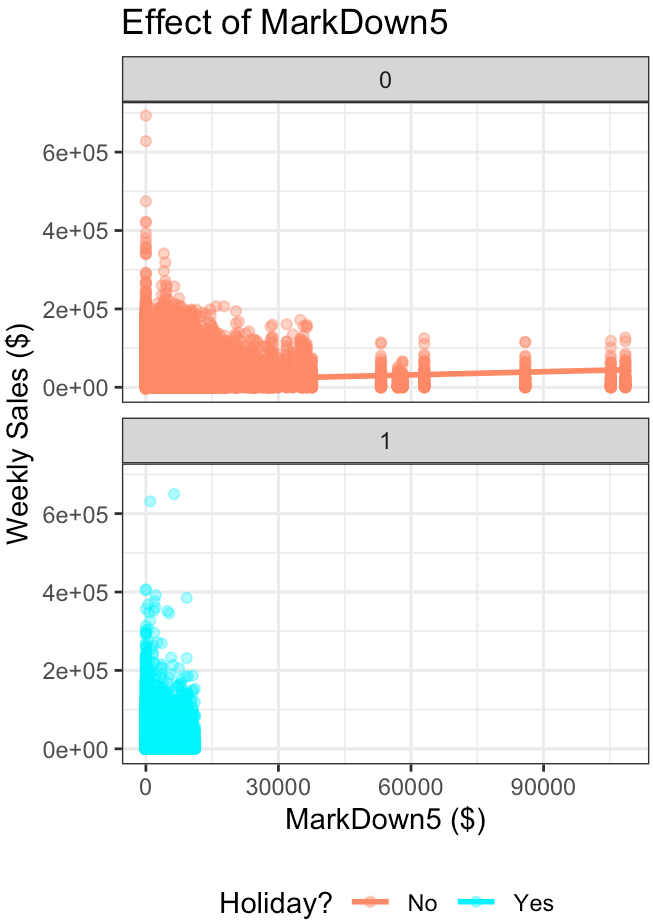
```
## $`1`
```




```
##
## $`2`
```


Effect of MarkDown3
0
1
6e+05
4e+05
2e+05
0e+00
Weekly Sales ($)
0e+00
5e+04
1e+05
MarkDown3 ($)
Holiday?
No
Yes
Effect of MarkDown4
0
1
6e+05
4e+05
2e+05
0e+00
Weekly Sales ($)
0
20000
40000
60000
MarkDown4 ($)
Holiday?
No
Yes

```
##
## $`3`
```




```
##
## attr(,"class")
## [1] "list"      "ggarrange"
```

Recommendations:

Based on the above analysis and visualizations, we recommend the following strategies/actions:

- **Increase Markdowns During Major Holidays**: The analysis shows a significant impact of markdowns on sales during major holiday weeks. Increasing markdowns during these periods can significantly boost sales. The analysis also showed that not all holiday weeks are the same, and certain markdown strategies are better suited for different holiday weeks.
- **Focus on High-Impact Stores**: Stores with larger sizes and higher historical sales should be prioritized for promotional activities to maximize the business impact. The smaller stores, despite the promotions could not match up to the bigger stores.
- **Monitor and Adjust Fuel Prices**: A decent correlation was observed between the fuel prices and sales. Given this association, it would be beneficial to monitor the fuel prices and adjust the promotions accordingly to optimize better sales.
- **Prepare for Seasonal Demand**: The sales forecast indicates peaks during specific weeks of the year. A finer deep-dive into these sales trends can identify department-wide trends for these seasonal trends. A careful planning of preparing the inventory and staffing for these peaks can have the following benefits:

- ***Better Inventory Management***: Ensuring optimal stock levels, reduced holding costs, better supplier relations and minimized wastage of perishable goods.
- ***Improved Operational Efficiency***: Preparing for peaks ensures adequate workforce to handle customer traffic, improved employee morale by reducing burnouts, and better resource allocation.
- ***Enhanced Financial Performance***: Meeting customer demand during peak periods maximizes sales opportunities, better planinng of employee resources ensures cost efficiency, and a consistent availability of products lead to revenue stability.
- ***Brand Loyalty***: Adequate stock and staff ensures higher customer satisfaction and leads to enhanced brand loyalty and customer retention.
- ***Competitive Advantage***: The ability to analyze sales trends to understand customer behavior patterns ensure a competitive advantage through better market responsiveness, thereby enhancing the brand value
- ***Risk Management***: Preparing for peaks and also closely monitoring external factors like fuel prices, temperatures, staffing shortages, can help mitigate risks associated with supply chain disruptions, and acute demand spikes (e.g., panic buying).