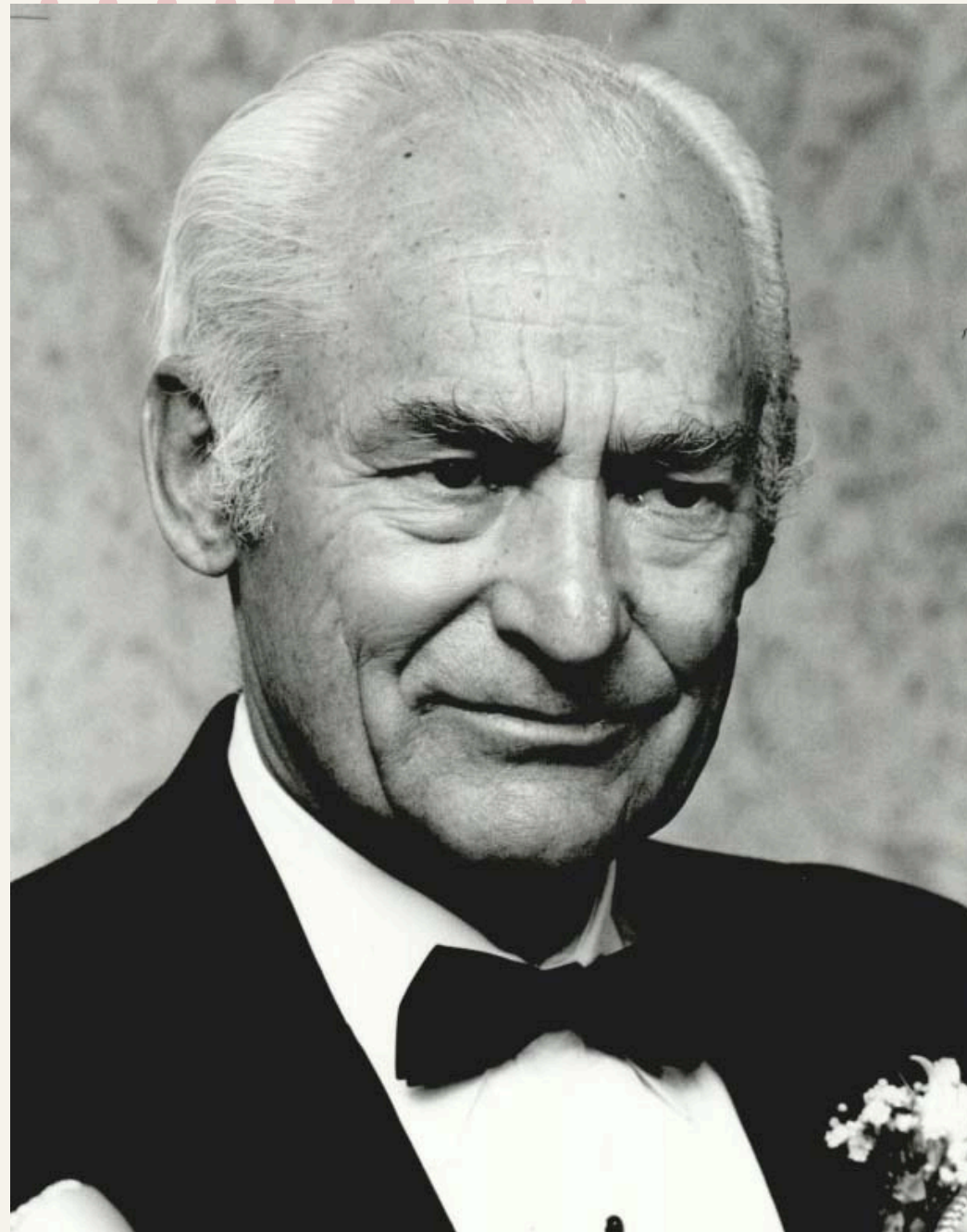


# **MACHINE LEARNING PROJECT**

## **RETAIL SALES PREDICTION**

**By : Vaibhavi Gaonkar**

**Date: 17-06-2024**



---



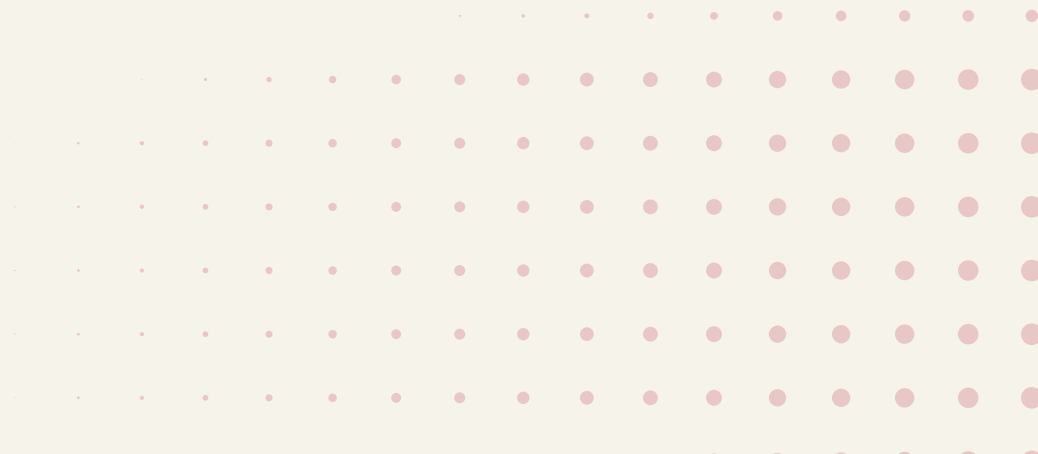
# **SAM WALTON**

**Founder, Walmart**





# CONTENT

- Introduction
  - Project Overview
  - Data Exploration
  - Data Preprocessing
- 
- Data Visualization
  - Feature Engineering
  - Feature Selection
  - Outliers in Sales Data
- 
- Data Transformation
  - Model Building
  - Evaluating Models
  - Conclusion
- 

# IMPORTANCE OF SALES FORECASTING IN RETAIL



Accurate sales forecasting is crucial for retail success, balancing inventory management with customer needs.

## Benefits

Estimate revenue, make data-driven decisions, allocate resources effectively.

## Challenges

Fluctuating preferences, seasonality, unforeseen events make accurate forecasting difficult.

## Solution

Machine learning analyzes sales data to enhance retail strategies.



---

# THE XYZ RETAIL CHALLENGE

## ● Problem Statement

XYZ, a leading drugstore chain, struggles with inconsistent sales forecasts across its 3,000 stores.

## ● Project Objective

Develop a machine learning model to accurately predict sales, improving decision-making and efficiency.

## ● Project Scope

Utilize historical sales data from 1,115 stores, considering factors like promotions, holidays, and store locations.

---



---

# **DATA EXPLORATION: A GLIMPSE INTO THE DATASET**

## **Data Sources:**

- **Salesdata.csv: Over 1 million rows of Historical sales data with .**
  - **Store.csv: Detailed information about each of the 1,115 stores.**
- 
-

# SALES DATASET

Store	DayOfWeek	Date	Sales	Customers	Open	Promo	StateHoliday	SchoolHoliday
1	5	31-07-2015	5263	555	1	1	0	1
2	5	31-07-2015	6064	625	1	1	0	1

1. 1017209 rows, 9 columns

2. Columns:

- Store: Unique identifier ranging from 1 to 1,115
- DayOfWeek: Indicates sales day (1 = Monday to 7 = Sunday)
- Date: Specific sales date, covering a period from January 1st, 2013, to July 31st, 2015.
- Sales: Total revenue generated per store on a given day, the variable for prediction.
- Customers: Number of store visitors on that day.
- Open: Store open (1) or closed (0) on that day.
- Promo: Presence of store promotional offers on that day.
- State Holiday: Type of state holiday ('a' = public, 'b' = Easter, 'c' = Christmas, 0 = none)
- School Holiday: Indicates public school closure on that day (0 = none, 1 = holiday).

3. No missing values



# STORE DATASET

Store	StoreType	Assortment	CompetitionDistance	CompetitionOpenSinceMonth	CompetitionOpenSinceYear	Promo2	Promo2SinceWeek	Promo2SinceYear	PromoInterval
1	c	a	1270	9	2008	0			
2	a	a	570	11	2007	1	13	2010	Jan, Apr, Jul, Oct
3	a	a	14130	12	2006	1	14	2011	Jan, Apr, Jul, Oct

1. 1,115 rows, 10 columns

2. Columns:

- **Store:** Unique identifier ranging from 1 to 1,115
- **StoreType:** Differentiates store models ('a', 'b', 'c', 'd') based on layout and design.
- **Assortment:** Describes product variety ('a' = Basic, 'b' = Extra, 'c' = Extended) offered at each store.
- **CompetitionDistance:** Distance in meters to the nearest competitor store
- **CompetitionOpen Since[Month/Year]:** Approximate opening month and year of the nearest competitor store
- **Promo2:** Indicates if the store participates in a continuing promotion (Promo2), with values '0' for no participation and '1' for participation.
- **Promo2 Since[Year/Week]:** Year and week when the store began participating in Promo2.
- **PromoInterval:** Specifies the months of each new Promo2 round, such as "Feb, May, Aug, Nov".

3. Missing values are present



# DATA PREPROCESSING

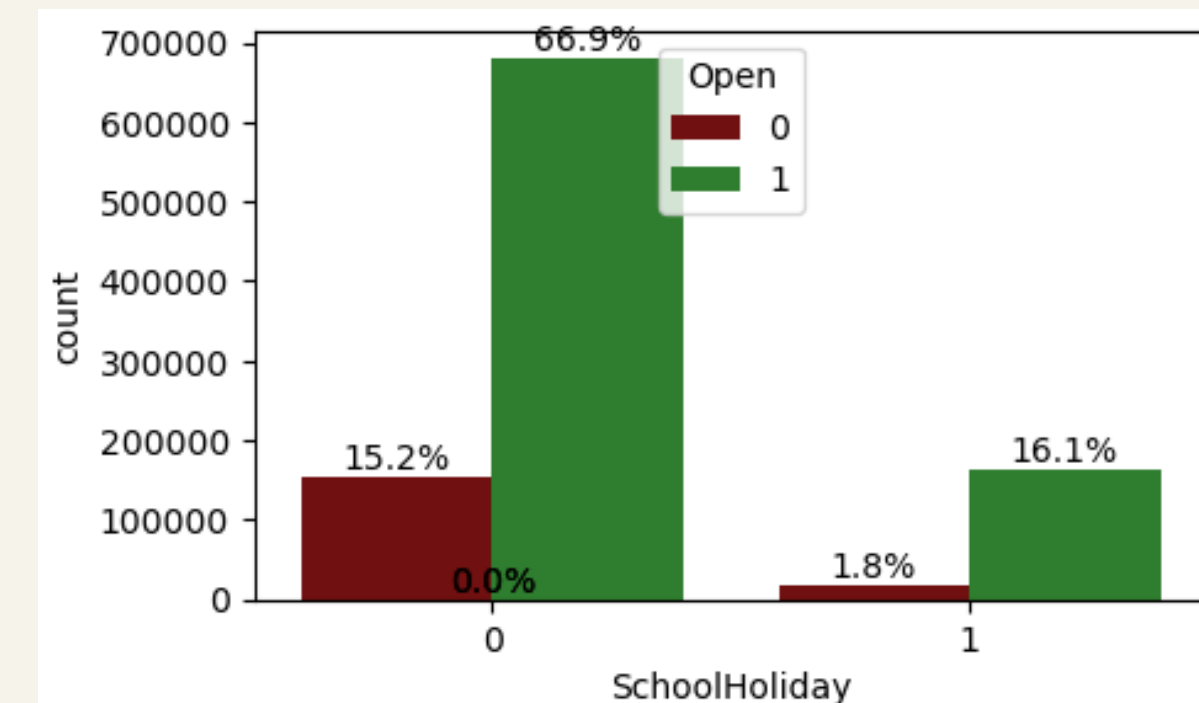
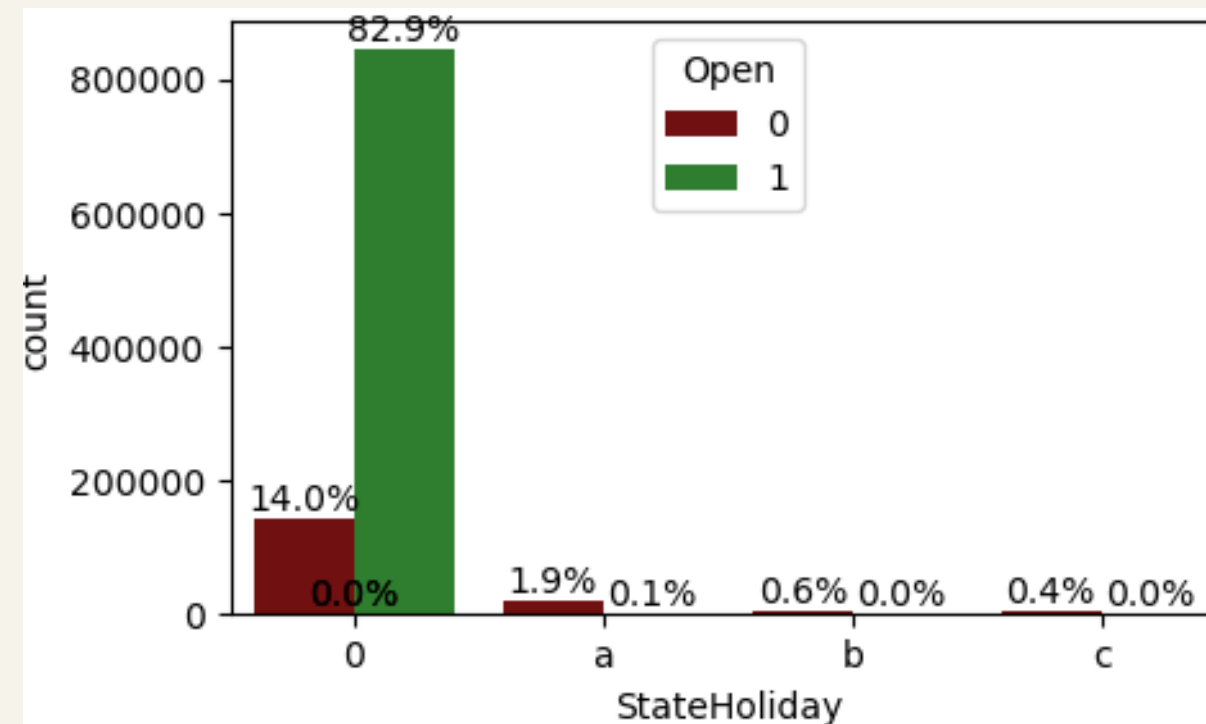
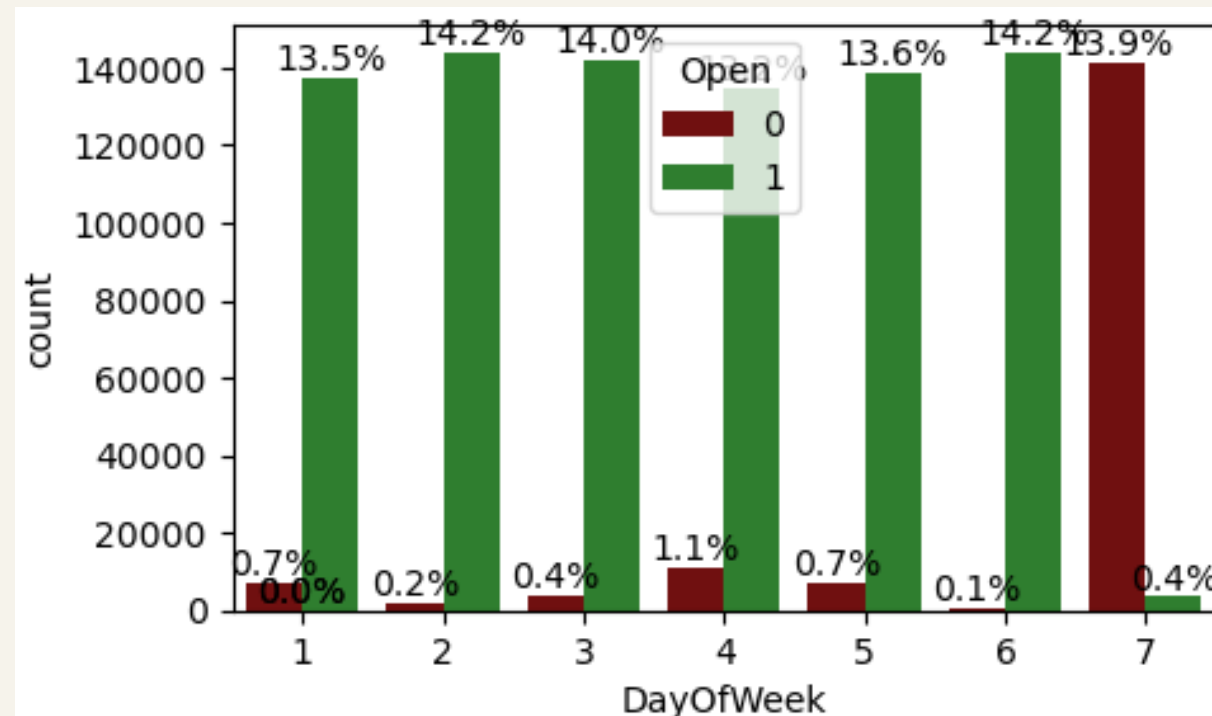
**1. Merging datasets**

**2. Handled data errors**

**3. Handling missing values**

# DATA VISUALIZATION

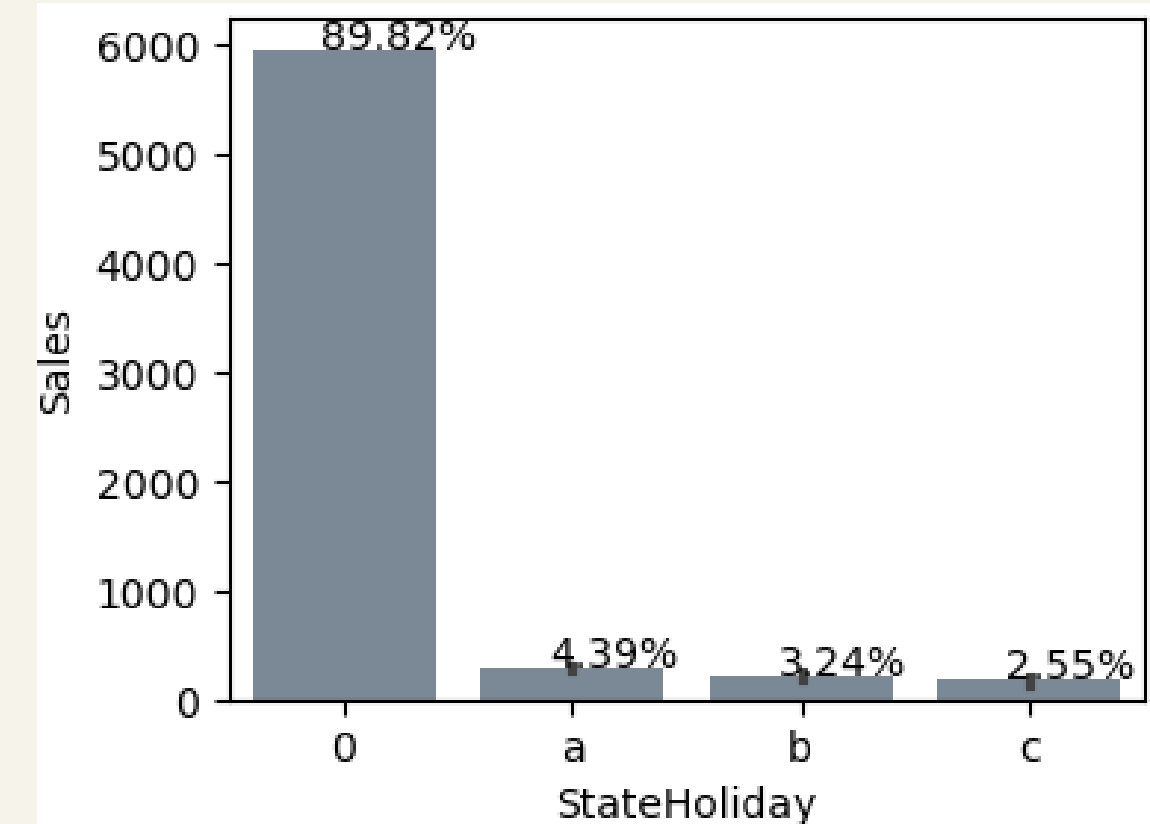
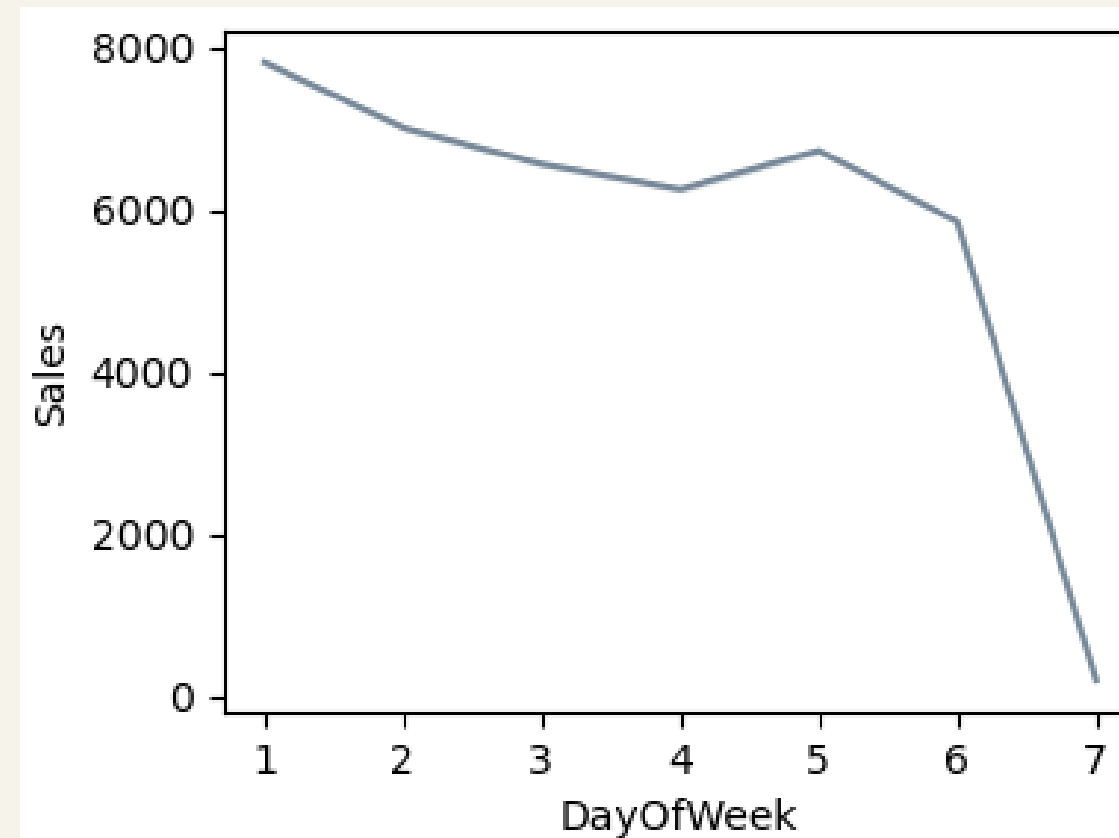
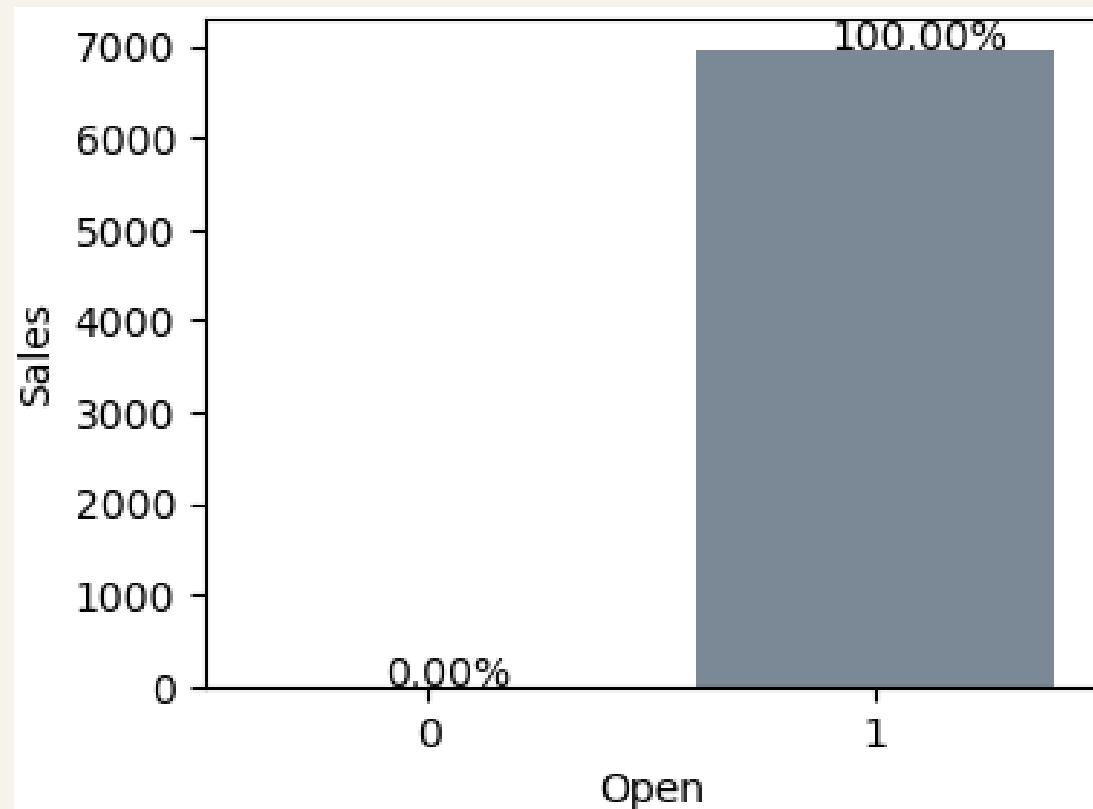
## 1 Store Openings



The graphs show more store closures on Sundays, frequent closures on state holidays, and minimal impact of school holidays on store openings.

# DATA VISUALIZATION

## 2 Sales and Store Closures

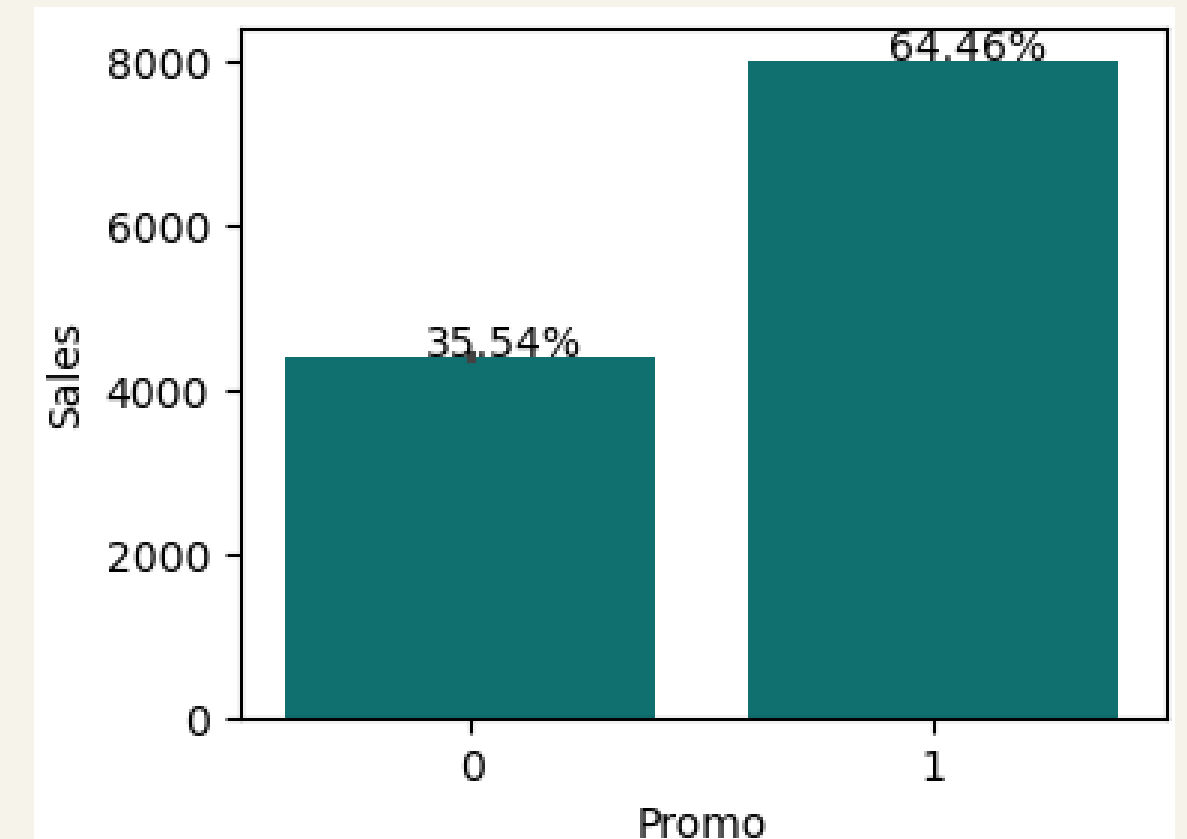
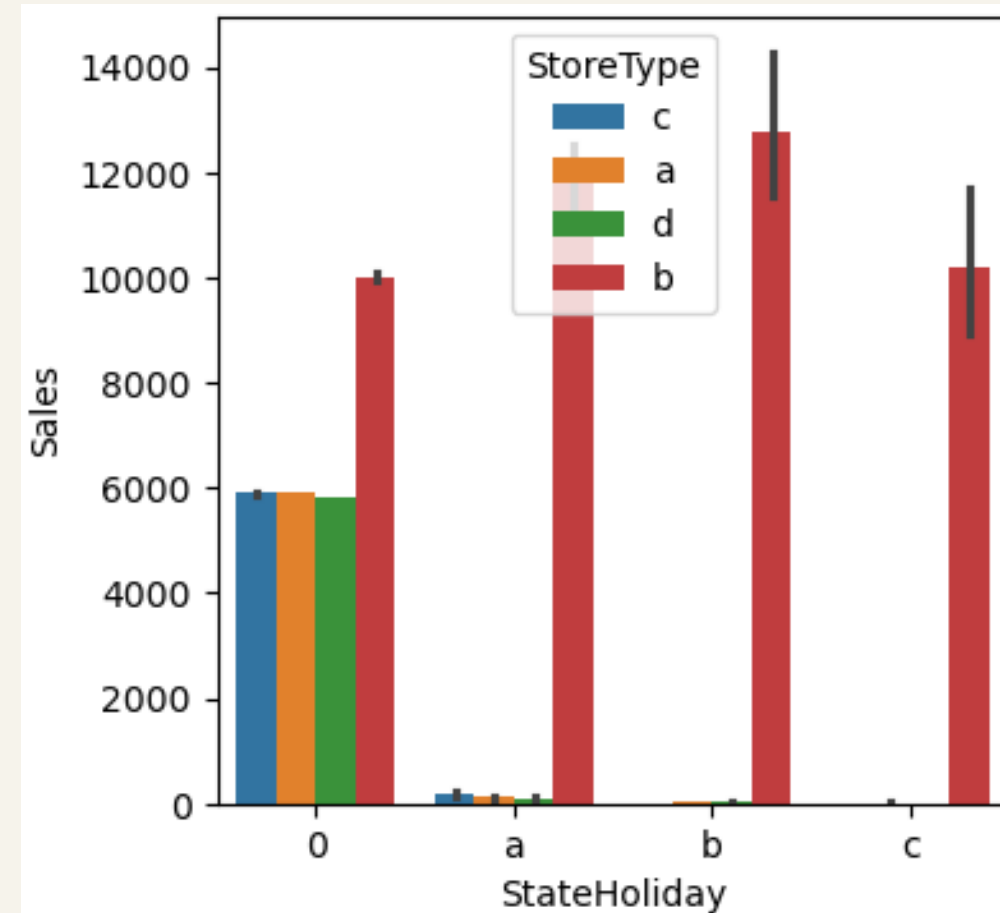
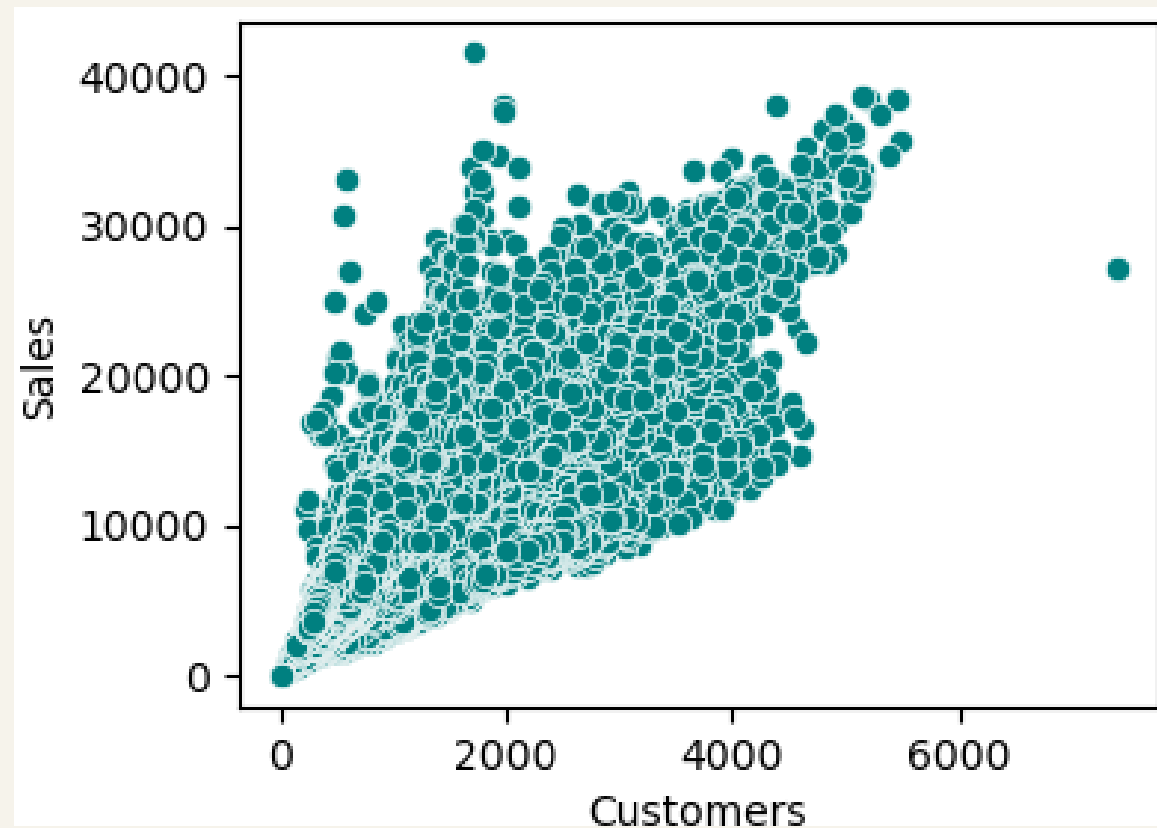


- The first graph shows a strong correlation between store closures and zero sales.
- The second graph indicates low Sunday sales due to frequent closures.
- The last graph shows state holidays negatively impact sales



# DATA VISUALIZATION

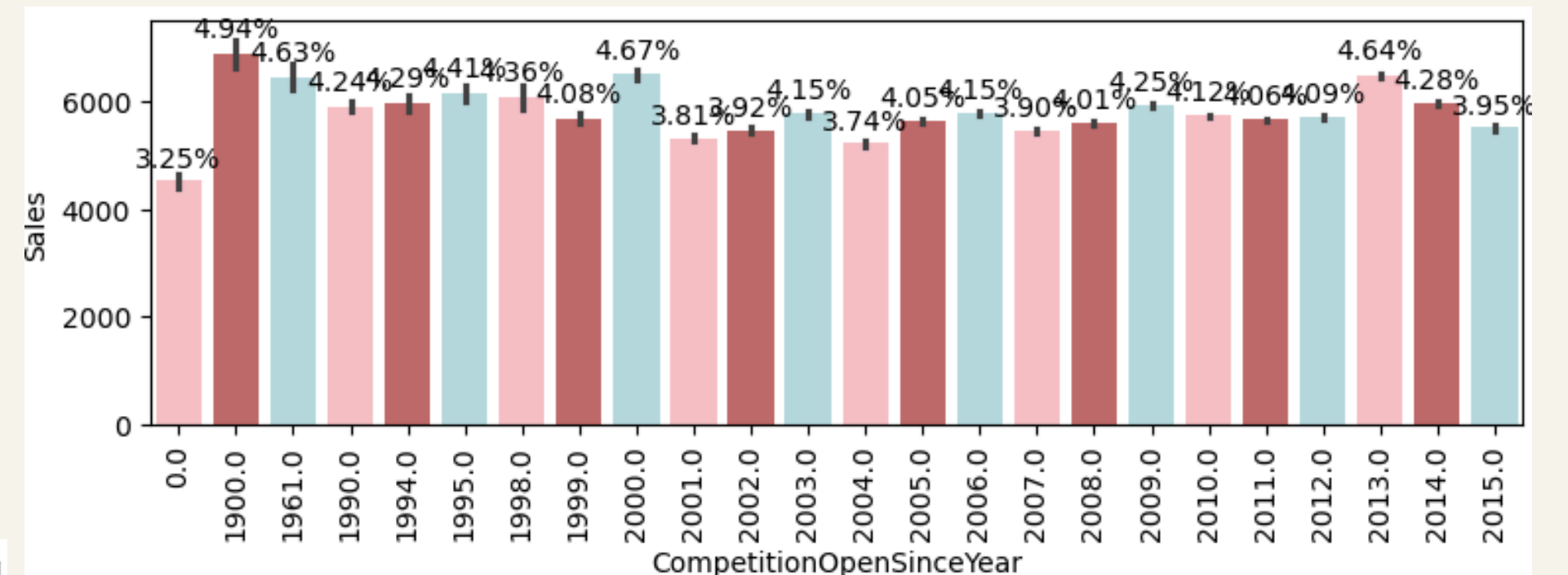
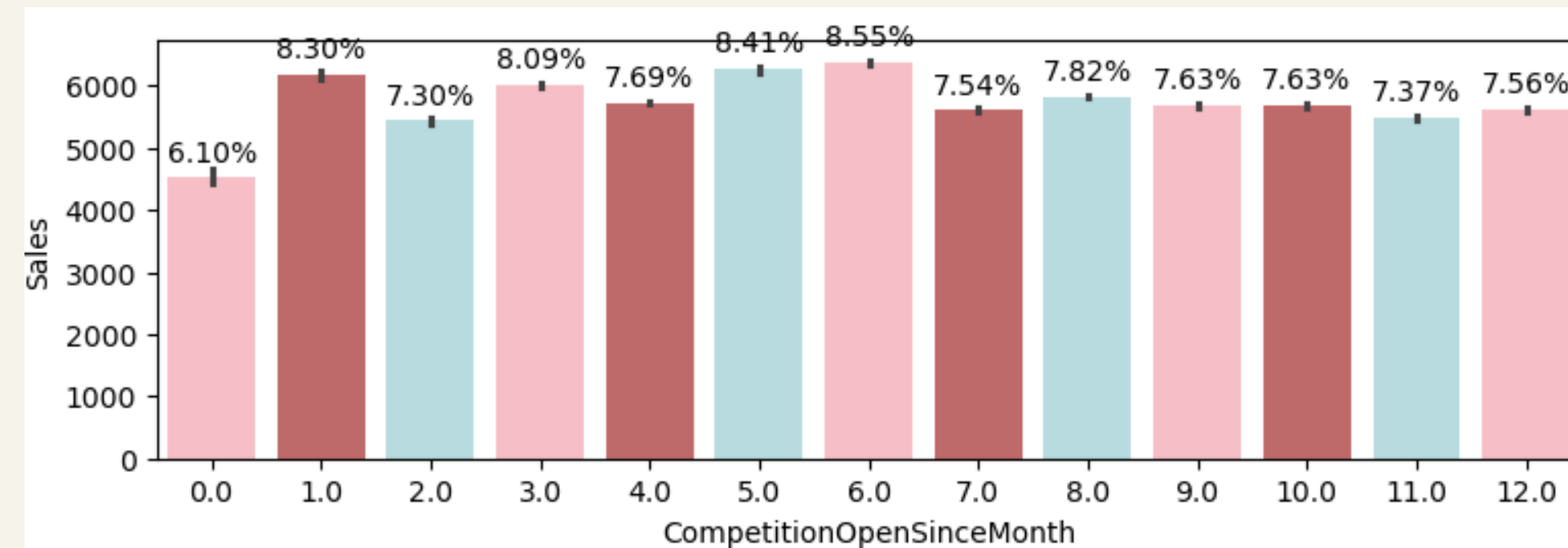
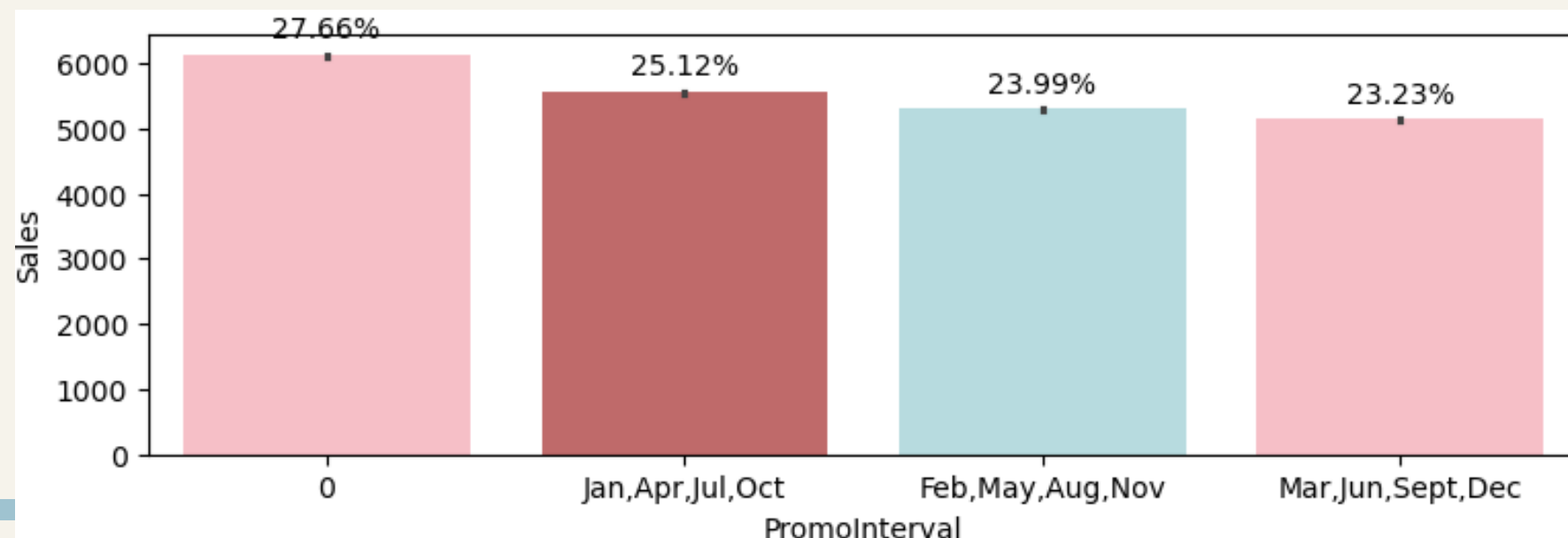
## 3 Sales and Other Factors



- The first plot shows a strong positive correlation between sales and customer counts.
- The second plot indicates Type 'b' stores are less affected by state holidays.
- The last plot shows promotions double store sales.

## 4 Sales and Other Factors

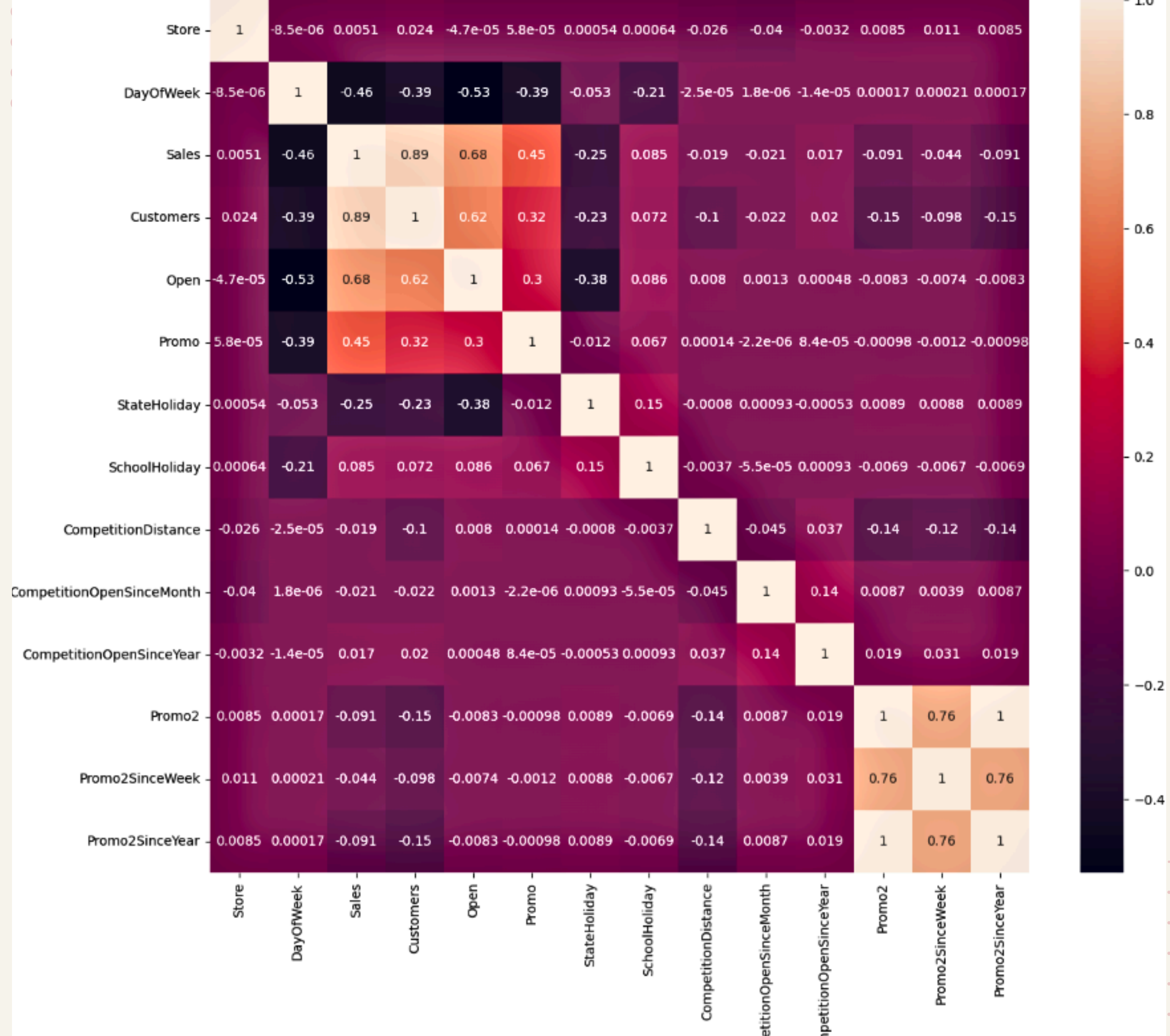
The bar plots for sales versus 'CompetitionOpenSinceMonth' and 'CompetitionOpenSinceYear' reveal that sales have remained relatively stable across different months and years since the competitor store opened. This indicates that the opening month and year of the competitor store did not significantly affect sales.



The distribution of sales is relatively similar across different promo intervals, with sales volume comparable to periods with no promo interval. Therefore, we can drop this column before building the model.



## 4 Correlation Matrix

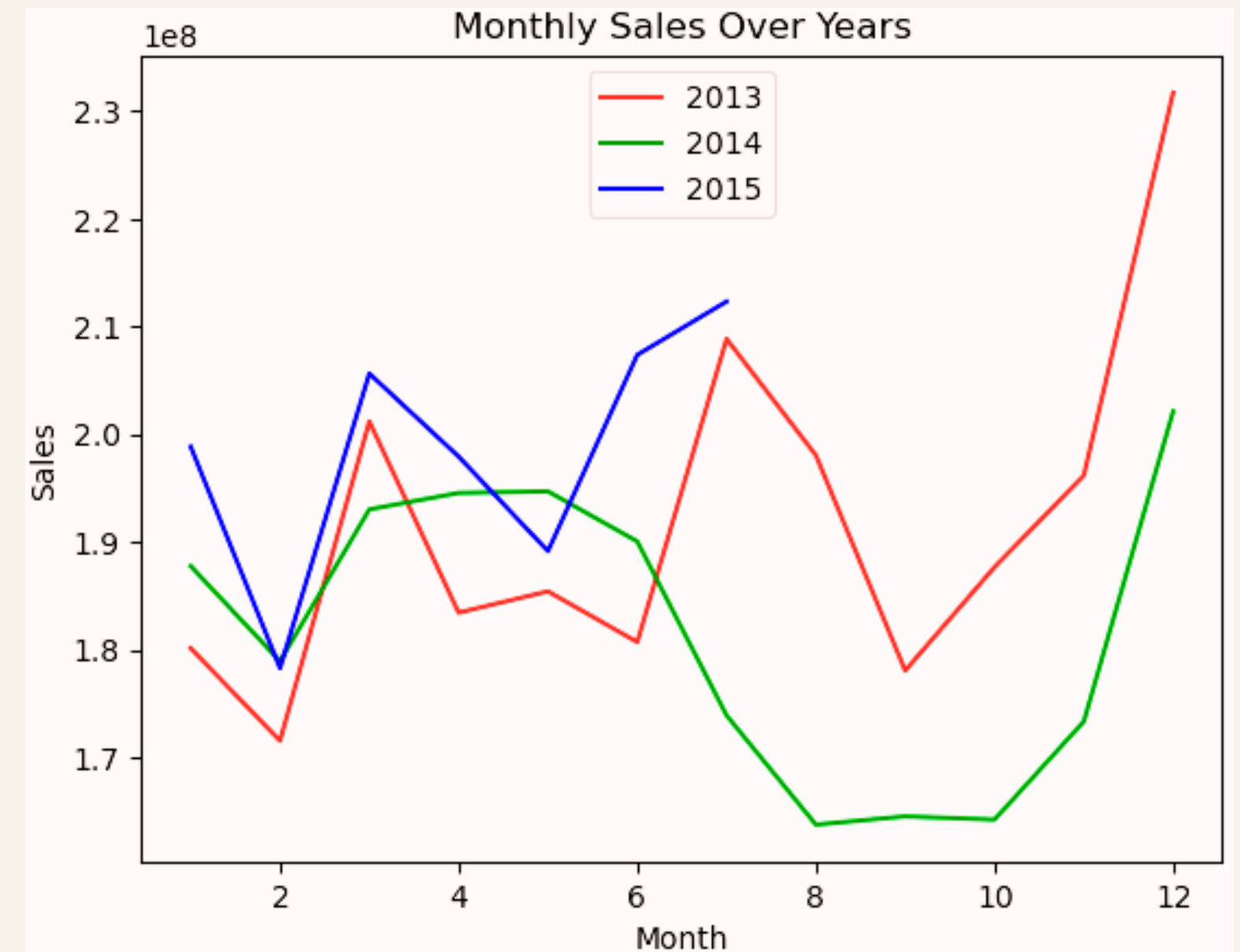




# FEATURE ENGINEERING

Feature Engineering is the process of creating new features or transforming existing features to improve the performance of a machine-learning model.

- **Extracted temporal features from 'Date' column: Day, Month, Year, WeekOfYear.**
- **Monthly sales trend from 2013 to mid-2015 reveals:**
  - **Seasonal patterns with significant sales peaks in July and December, underscoring the impact of seasonality on forecasting.**
  - **Periods of reduced sales in August-October and February, suggesting potential store closures during these times.**







# FEATURE SELECTION

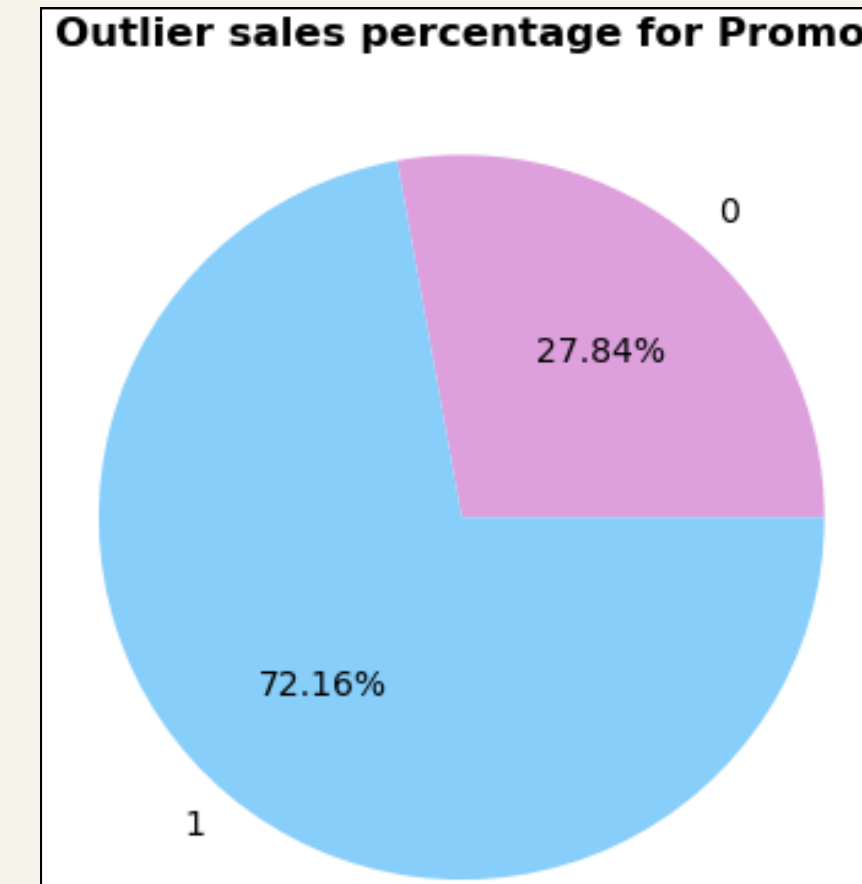
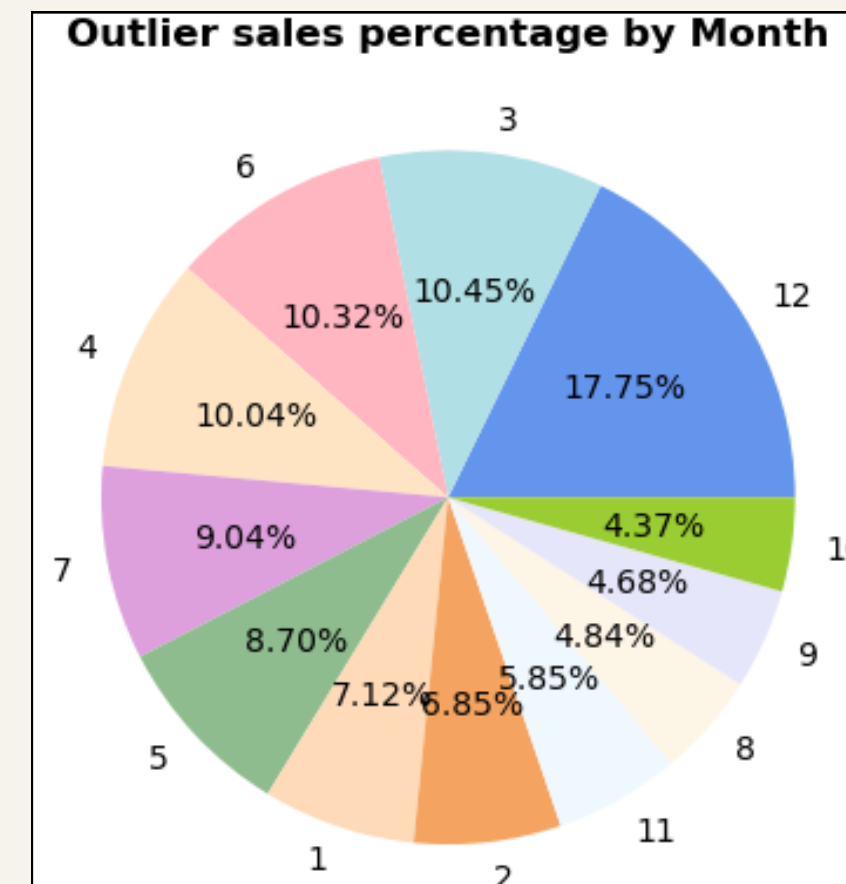
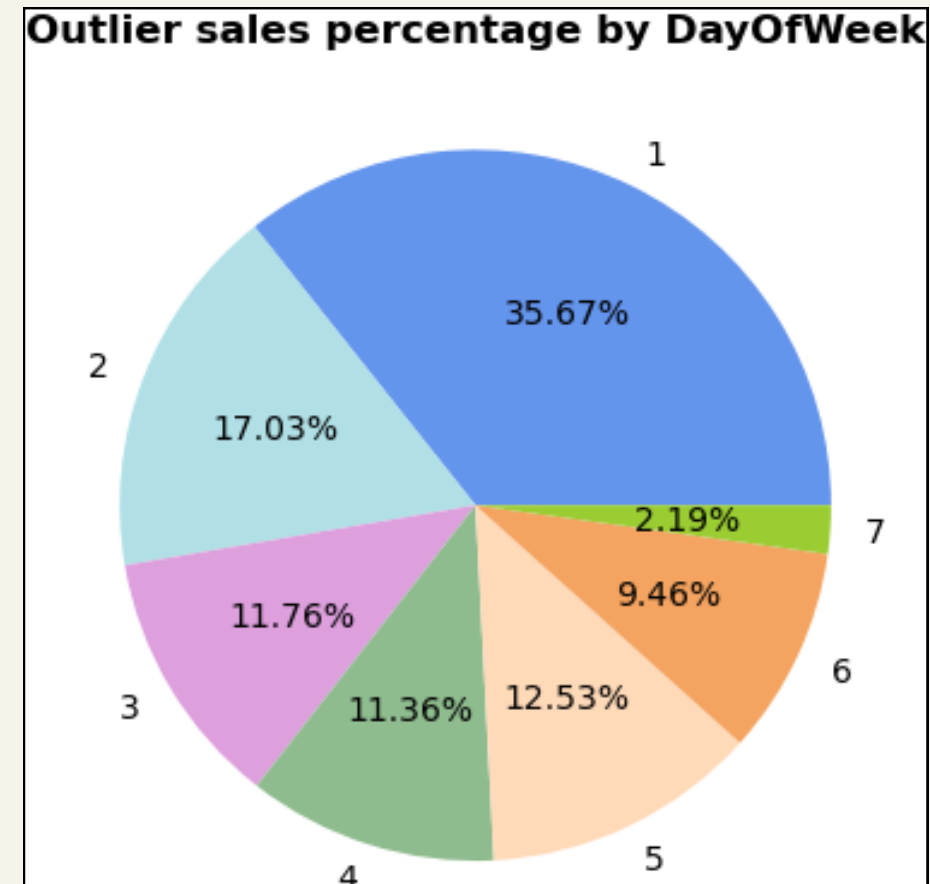
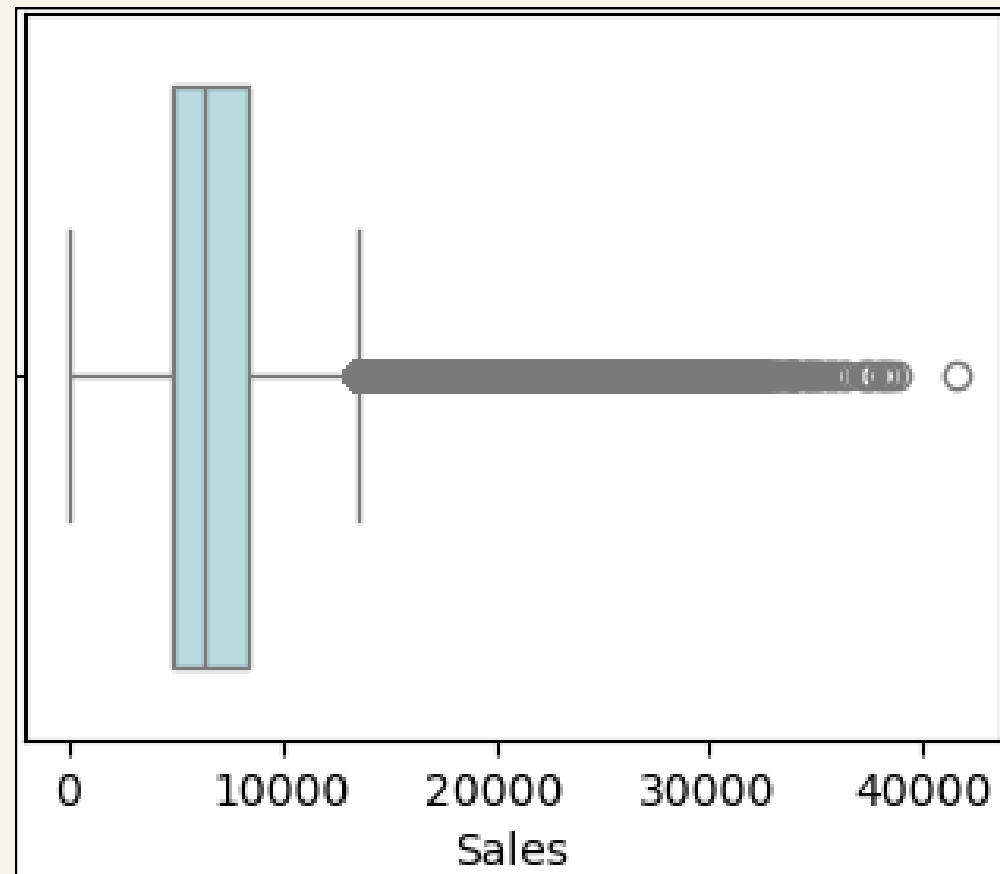
---

Feature Selection involves reducing the input variables in the model by utilising only relevant data and removing any unnecessary noise from the dataset.

- StateHoliday
  - Open
  - Promo2 Since[Year/Week]
  - Competition Open Since[Month/Year]
  - PromoInterval
  - Sales
- 
-

# OUTLIERS IN SALES DATA

Outlier detection: Identified through boxplots



- 1 Mondays see 35% of outlier sales, attributed to Sunday closures
- 2 Sales outliers peak in December (Christmas), March (end of school), and June (back-to-school).
- 3 Promotions coincide with peak sales periods

Outliers are valid and crucial from business perspective.

# DATA TRANSFORMATION

## 1. Feature Scaling

- CompetitionDistance, Customers

## 2. Sampling

- sample 100,000 rows

## 3. Data Splitting

- Separate recent 6 weeks for testing, the rest for training.

## 4. Encoding

- DayOfWeek, StoreType, Assortment

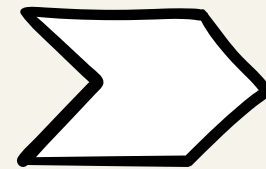
# PREPARED DATA

	Store	DayOfWeek	Date	Sales	Customers	Promo	StateHoliday	SchoolHoliday	StoreType	Assortment	CompetitionDistance	Promo2	Year	Month	Day	WeekOfYear
54810	423	2	2013-01-01	9643	2.463205	0	1	1	b	a	-0.534831	0	2013	1	1	1
94791	769	2	2013-01-01	5035	1.209447	0	1	1	b	b	-0.589928	1	2013	1	1	1
62174	335	2	2013-01-01	2401	-0.699854	0	1	1	b	a	-0.686028	1	2013	1	1	1
20643	948	2	2013-01-01	4491	0.688502	0	1	1	b	b	-0.514330	0	2013	1	1	1
9991	562	2	2013-01-01	8498	2.273770	0	1	1	b	c	-0.542519	0	2013	1	1	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
4255	15	5	2015-07-31	9191	0.008033	1	0	1	d	c	-0.170933	1	2015	7	31	31
96830	83	5	2015-07-31	3866	-1.086201	1	0	1	a	a	-0.350319	0	2015	7	31	31
21746	526	5	2015-07-31	11006	1.673063	1	0	1	a	a	-0.682184	1	2015	7	31	31
56263	464	5	2015-07-31	11967	1.266776	1	0	1	c	a	-0.630931	0	2015	7	31	31
84217	815	5	2015-07-31	8186	-0.084192	1	0	0	a	a	-0.621961	1	2015	7	31	31
100000 rows × 16 columns																

# MODEL BUILDING

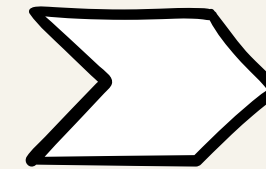
## Model Tracking

Created a DataFrame (Models\_df) to track model performance metrics (MAE, MSE, RMSE, MAPE,  $R^2$ , Adjusted  $R^2$ , Accuracy).



## Evaluation Function

Developed evaluate\_model function to streamline model training and evaluation.



## Model Training

Trained

- Linear Regression
- Decision Tree (tuned)
- Random Forest (tuned)
- XGBoost

# EVALUATING MODELS

## LINEAR REGRESSION

A simple yet powerful algorithm that finds a linear relationship between input features and the target variable (sales). It aims to minimize the difference between predicted and actual sales values by fitting a straight line.

	Model_Name	MAE_Train	MSE_Train	RMSE_Train	MAPE_Train	R2_Train	Adj_R2_Train	MAE_Test	MSE_Test	RMSE_Test	MAPE_Test	R2_Test	Adj_R2_Test	Accuracy
0	Linear Regression	934.25	1627678.815	1275.805	14.599	0.835	0.835	946.123	1732034.676	1316.068	14.332	0.827	0.826	85.668

- **Performance:**
  - **Training:** Achieves good R-squared (0.835), indicating strong correlation, but MAE is relatively high (934.25).
  - **Testing:** Maintains accuracy (R-squared 0.827), suggesting good generalization.
- **Strengths:** Interpretability, often provides a good starting point, computationally efficient.
- **Weaknesses:** Might not capture complex relationships, could be less accurate for highly nonlinear data.

# DECISION TREE

Decision trees are a fundamental machine learning algorithm that works by creating a tree-like structure to represent a series of decisions leading to a prediction. They are popular for their interpretability and ease of understanding.

Model_Name	MAE_Train	MSE_Train	RMSE_Train	MAPE_Train	R2_Train	Adj_R2_Train	MAE_Test	MSE_Test	RMSE_Test	MAPE_Test	R2_Test	Adj_R2_Test	Accuracy
Decision Tree	0.000	0.000	0.000	0.000	1.000	1.000	804.853	1335415.204	1155.602	11.565	0.867	0.866	88.435
Decision Tree Tuned	396.882	337649.952	581.077	5.869	0.966	0.966	733.728	1108031.605	1052.631	10.438	0.889	0.888	89.562

- **Untuned Decision Tree:**
  - **Problem:** Severe overfitting to training data (R2 score: 1.000), but poor generalization (R2 score: 0.867 on testing).
  - Untuned decision trees often memorize the training data and don't perform well on unseen data.
- **Decision Tree Tuned (Using Grid Search):**
  - Significant improvement in testing accuracy (R2 score: 0.889), demonstrating the power of tuning.
  - **Best Parameters:** max\_depth: 20, min\_samples\_leaf: 4, min\_samples\_split: 10.
  - Tuning controls complexity and prevents overfitting, leading to a more reliable model.



# RANDOM FOREST

An ensemble method that combines multiple decision trees. It averages predictions from individual trees, reducing overfitting and generally improving accuracy.

	Model_Name	MAE_Train	MSE_Train	RMSE_Train	MAPE_Train	R2_Train	Adj_R2_Train	MAE_Test	MSE_Test	RMSE_Test	MAPE_Test	R2_Test	Adj_R2_Test	Accuracy
0	Random Forest	5.786	86985.587	294.933	3.067	0.991	0.991	603.775	758068.124	870.671	8.493	0.924	0.924	91.507
1	Random Forest Tuned	273.502	144784.153	380.505	4.226	0.985	0.985	766.014	1192280.230	1091.916	11.371	0.881	0.880	88.629

- **Untuned Random Forest:**
  - Very high R2 scores on both training data (0.991) and testing(0.924).
  - High accuracy, robust to overfitting, good at handling complex data.
- **Random Forest Tuned (Using RandomizedSearchCV):**
  - Tuning didn't improve the test performance in this case.
  - Resulted in lower R2 scores(0.881) compared to the untuned version.
  - This suggests that the default parameters for Random Forest were already quite optimal for this specific dataset.

# XGBOOST

A powerful gradient boosting algorithm that builds an ensemble of decision trees iteratively. It's known for its high accuracy and robustness.

Model_Name	MAE_Train	MSE_Train	RMSE_Train	MAPE_Train	R2_Train	Adj_R2_Train	MAE_Test	MSE_Test	RMSE_Test	MAPE_Test	R2_Test	Adj_R2_Test	Accuracy
XGBoost	486.293	434515.081	659.178	7.472	0.956	0.956	584.093	651491.772	807.15	8.926	0.935	0.935	91.074

- **Performance:**
  - **Training:** Good R2 score (0.956), suggesting good fit.
  - **Testing:** High accuracy (R2 score 0.935), suggesting good generalization.
- **Strengths:** Very high accuracy, can handle complex datasets, often outperforms other algorithms.
- **Weaknesses:** Can be more computationally expensive, may be less interpretable than linear regression or decision trees.

# EVALUATING MODELS

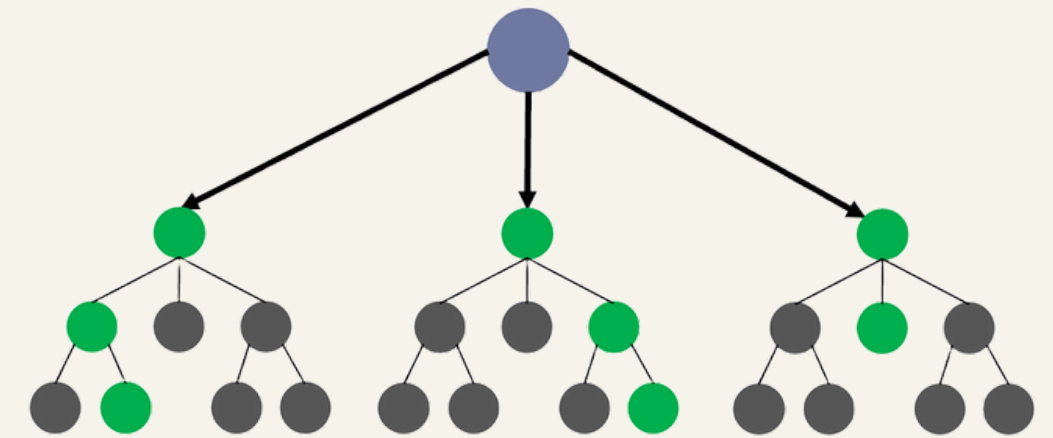
	R2_Train	Adj_R2_Train	R2_Test	Adj_R2_Test	Accuracy
Model_Name					
Linear Regression	0.835	0.835	0.827	0.826	85.668
Decision Tree	1.000	1.000	0.867	0.866	88.435
Decision Tree Tuned	0.966	0.966	0.889	0.888	89.562
Random Forest	0.991	0.991	0.924	0.924	91.507
Random Forest Tuned	0.978	0.978	0.881	0.880	88.657
XGBoost	0.956	0.956	0.935	0.935	91.074

## Accuracy Improvements

- Decision Tree (+3.23% over Linear Regression)
- Decision Tree Tuned (+1.27% over Decision Tree)
- Random Forest (+2.17% over Decision Tree Tuned)
- Random Forest Tuned (-3.37% over Random Forest)
- XGBoost (-0.48% over Random Forest)

# MODEL SELECTION

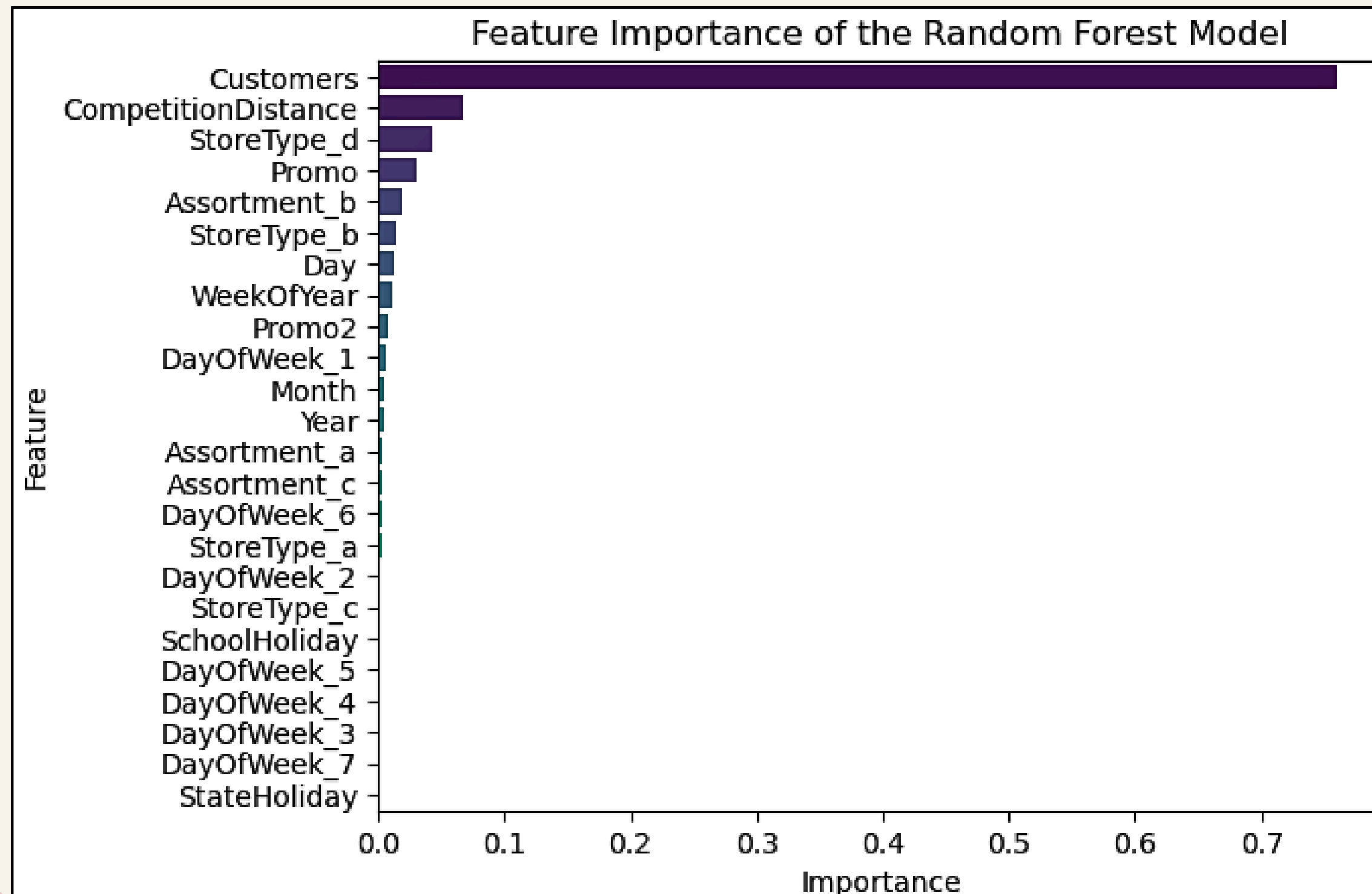
## Random Forest



### Reasons for Selection:

- Strong Accuracy and Generalization
- Clear Feature Importance Insights
- Computational Efficiency
- Consistent Performance

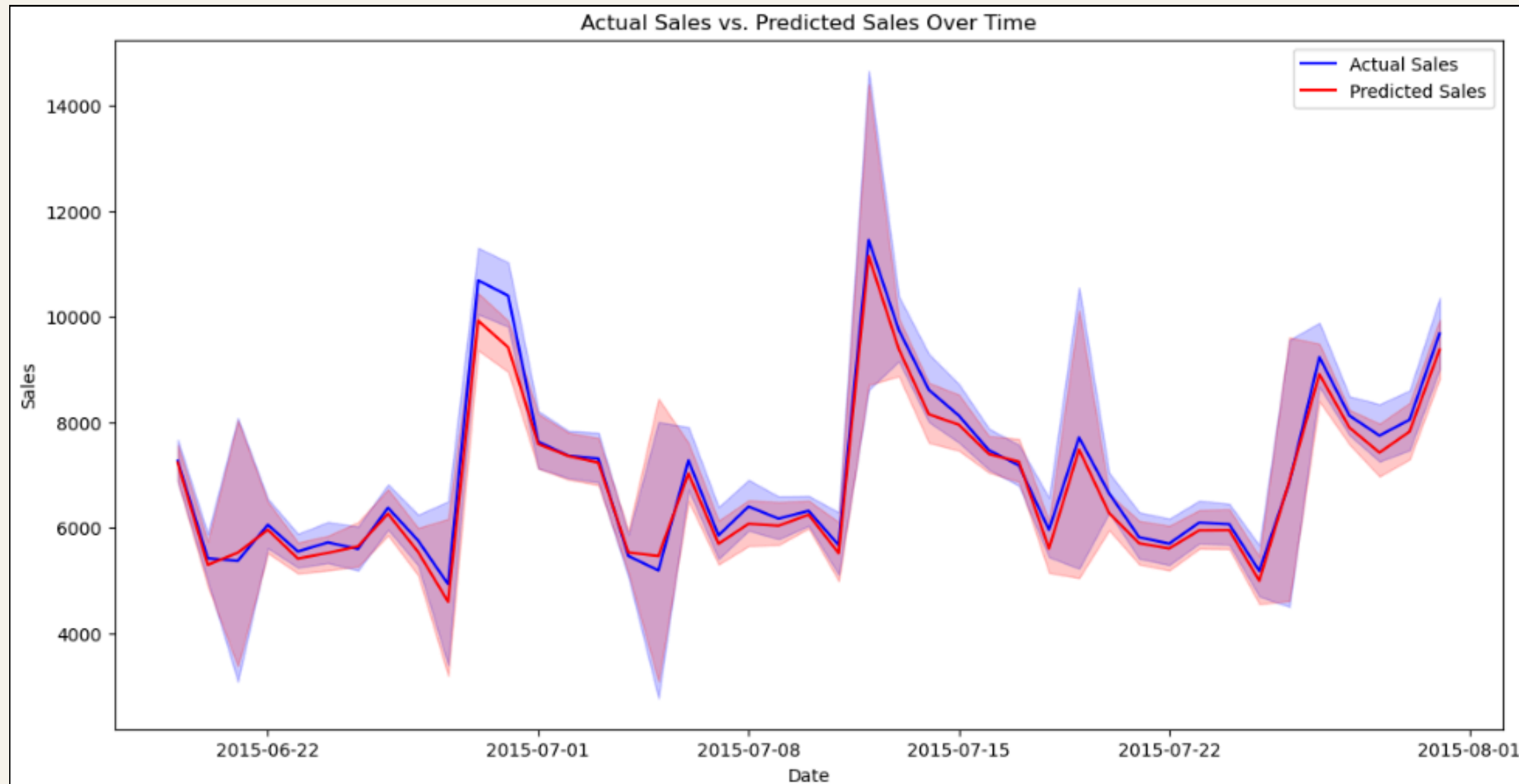
# FEATURE IMPORTANCE



# ACTUAL VS PREDICTED SALES

Store	Date	Sales_x	Pred Sales
1	2015-06-23	3762	3511.22
	2015-07-02	5558	5521.06
	2015-07-07	3650	3939.29
	2015-07-09	3897	3834.35
	2015-07-16	4427	4819.39
...	...	...	...
1114	2015-07-31	27508	22671.63
1115	2015-06-24	5463	4891.80
	2015-07-08	5900	5007.38
	2015-07-17	7874	6748.50
	2015-07-28	8093	7074.99

# PREDICTED VS ACTUAL SALES



## Trend alignment

Model captures overall sales trends.

## Variability

Model reflects spikes and drops in sales, with some variation.

## Alignment

Predictions closely match actual sales, with deviations during peaks.)



# CONCLUSION

Our machine learning model, powered by Random Forest, provided the best performance for sales prediction, achieving strong accuracy and generalization, has successfully captured the key drivers of sales for XYZ Retail, offering valuable insights to boost their business.

- The model highlights the paramount importance of customers. Invest in strategies to attract new customers and build loyalty among existing ones
- "CompetitionDistance" is a significant factor, underscoring the need for strategic store locations and competitive pricing.
- Capitalize on the strengths of "StoreType\_d" by understanding its unique customer base and optimizing product offerings, promotions, and marketing strategies tailored to that segment.
- "Promo" demonstrates a notable influence, confirming that well-planned and executed promotions can boost sales.
- XYZ Retail should proactively adjust inventory levels to meet increased demand during June (school openings), December (Christmas), and March (summer vacations).

This project demonstrates how machine learning, along with a deep understanding of data, can provide practical insights for retail success. By continually monitoring the model, incorporating new data, and iteratively refining strategies based on these insights, XYZ Retail can confidently navigate the dynamic retail landscape and achieve lasting growth.

The background features three vertical stripes on the left: a wide pink stripe, a medium blue stripe, and a narrow beige stripe. The right side of the image is a light beige background with two rectangular areas of small, light pink dots in the top right and bottom right corners.

**THANK YOU**