Retail Sales Analytics for Predictive Insights & Business Recommendations

TABLE OF CONTENTS

BUSINESS PROBLEM STATEMENT

Retail sales analytics deliver predictive insights and actionable business recommendations to enhance sales forecasting, markdown optimization, and operational efficiency.

BUSINESS IMPACT OF RETAIL DATA ANALYTICS:

Financial Impact: Accurate sales forecasting minimizes stockouts, reducing lost sales, while optimizing inventory levels lowers holding costs and markdown losses.

Customer Experience: Data-driven insights ensure product availability, enhancing customer satisfaction, loyalty, and repeat purchases.

Operational Efficiency: Improved demand forecasting streamlines inventory management, reducing waste, storage costs, and supply chain inefficiencies.

Strategic Decision-Making: Actionable analytics enable dynamic pricing, personalized marketing, and efficient resource allocation, maximizing profitability.

PROPOSED SOLUTION TO THE PROBLEM

To enhance sales forecasting, we propose leveraging an Extra Trees Regressor, a robust machine learning model that captures complex patterns efficiently. By incorporating historical sales, promotions, and store-specific features, the model will deliver more precise predictions.Advanced Machine Learning Model: Implement a tuned Extra Trees Regressor for improved forecasting accuracy.Feature Engineering: Utilize historical sales, promotional impact, and store-specific attributes for enhanced prediction.Hyperparameter Optimization: Fine-tune model parameters to minimize error and maximize predictive power.

EXPECTED VALUE ADDITIONS AND BENEFITS

This solution will lead to better demand forecasting, reducing stock shortages and excess inventory. Additionally, it will enable data-driven promotional strategies and efficient resource management, ultimately boosting profitability.Higher Forecast Accuracy: Reduces uncertainty in inventory planning and stock management.Optimized Promotions: Helps in scheduling discounts and offers based on data-driven insights.Better Resource Allocation: Supports efficient staffing and logistics decisions for peak sales periods.Increased Revenue: Enhances decision-making to drive higher profitability through improved demand prediction.

EXPLORATORY DATA ANALYSIS (EDA)

DATASET DESCRIPTION AND DOMAIN

A dataset is created after merging all three datasets.

```
        date  store  department     sales  IsHoliday_x  Temperature  \
0 2010-02-05      1           1  24924.50         False        42.31
1 2010-02-12      1           1  46039.49          True        38.51
2 2010-02-19      1           1  41595.55         False        39.93
3 2010-02-26      1           1  19403.54         False        46.63
4 2010-03-05      1           1  21827.90         False        46.50

   Fuel_Price         CPI  Unemployment  IsHoliday_y Type    Size
0       2.572  211.096358         8.106        False    A  151315
1       2.548  211.242170         8.106         True    A  151315
2       2.514  211.289143         8.106        False    A  151315
3       2.561  211.319643         8.106        False    A  151315
4       2.625  211.350143         8.106        False    A  151315
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 421570 entries, 0 to 421569
Data columns (total 12 columns):
 #   Column        Non-Null Count   Dtype
---  ------        --------------   -----
 0   date          421570 non-null  datetime64[ns]
 1   store         421570 non-null  int64
 2   department    421570 non-null  int64
 3   sales         421570 non-null  float64
 4   IsHoliday_x   421570 non-null  bool
 5   Temperature   421570 non-null  float64
 6   Fuel_Price    421570 non-null  float64
 7   CPI           421570 non-null  float64
 8   Unemployment  421570 non-null  float64
 9   IsHoliday_y   421570 non-null  bool
 10  Type          421570 non-null  object
 11  Size          421570 non-null  int64
dtypes: bool(2), datetime64[ns](1), float64(5), int64(3), object(1)
memory usage: 33.0+ MB
```

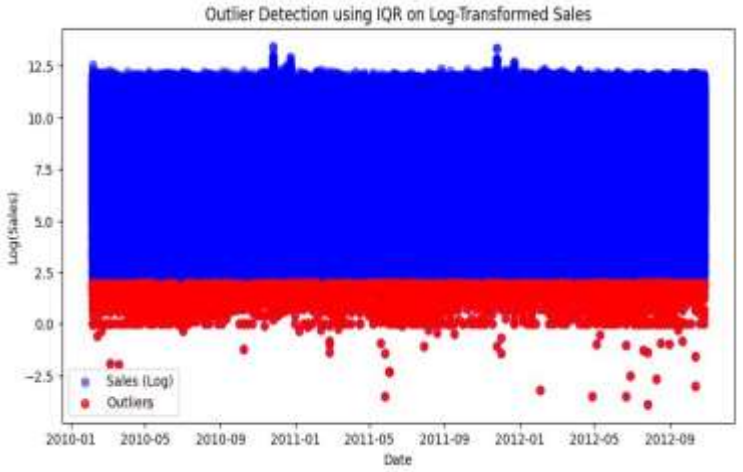
```
                                date          store     department  \
count                         421570  421570.000000  421570.000000
mean   2011-06-18 08:30:31.963375104      22.200546      44.260317
min              2010-02-05 00:00:00       1.000000       1.000000
25%              2010-10-08 00:00:00      11.000000      18.000000
50%              2011-06-17 00:00:00      22.000000      37.000000
75%              2012-02-24 00:00:00      33.000000      74.000000
max              2012-10-26 00:00:00      45.000000      99.000000
std                              NaN      12.785297      30.492054

               sales    Temperature     Fuel_Price            CPI  \
count  421570.000000  421570.000000  421570.000000  421570.000000
mean    15981.258123      60.090059       3.361027     171.201947
min     -4988.940000      -2.060000       2.472000     126.064000
25%      2079.650000      46.680000       2.933000     132.022667
50%      7612.030000      62.090000       3.452000     182.318780
75%     20205.852500      74.280000       3.738000     212.416993
max    693099.360000     100.140000       4.468000     227.232807
std     22711.183519      18.447931       0.458515      39.159276

        Unemployment           Size
count  421570.000000  421570.000000
mean        7.960289  136727.915739
min         3.879000   34875.000000
25%         6.891000   93638.000000
50%         7.866000  140167.000000
75%         8.572000  202505.000000
max        14.313000  219622.000000
std         1.863296   60980.583328

Final combined dataset exported successfully to 'final_combined_dataset.csv'.
```

HANDLING MISSING VALUES

```
Missing values per column before cleaning:
Store                0
Dept                 0
Date                 0
Weekly_Sales         0
IsHoliday_x          0
Temperature          0
Fuel_Price           0
MarkDown1       270889
MarkDown2       310322
MarkDown3       284479
MarkDown4       286603
MarkDown5       270138
CPI                  0
Unemployment         0
IsHoliday_y          0
Type                 0
Size                 0
dtype: int64
```

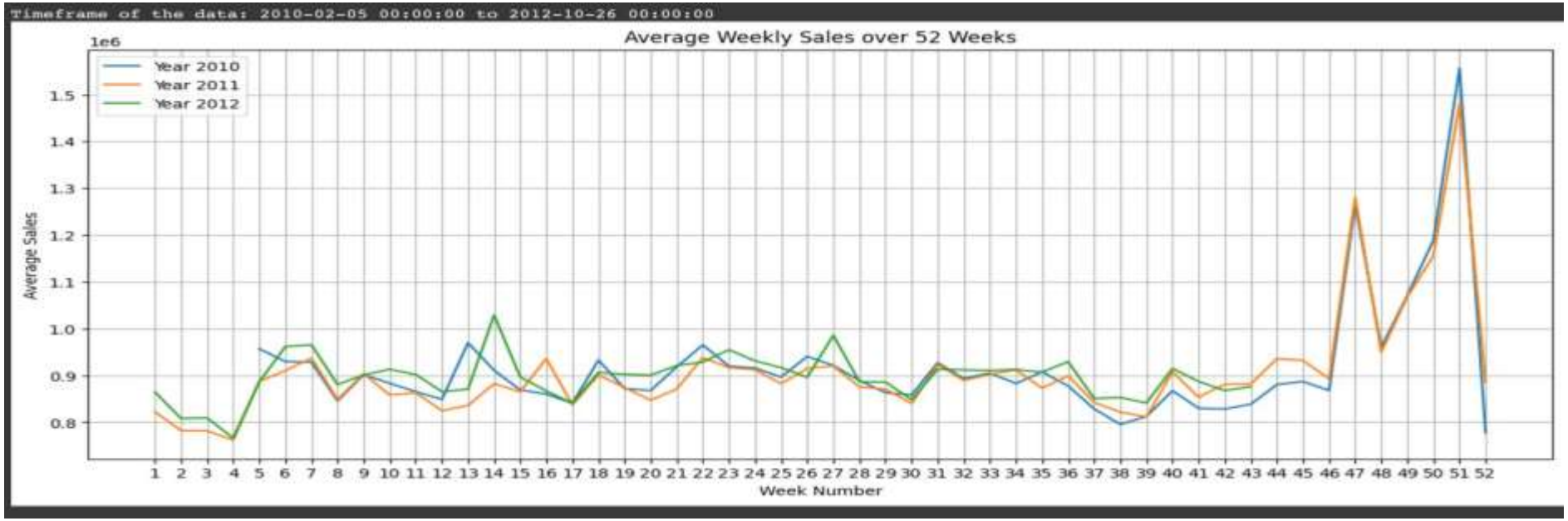
- Defined a 25% missing value threshold for each column and dropped the columns as necessary ie ['MarkDown1', 'MarkDown2', 'MarkDown3', 'MarkDown4', 'MarkDown5']
- Used forward fill for missing categorical data.
- For numerical data the missing data is replaced with median.

OUTLIER DETECTION

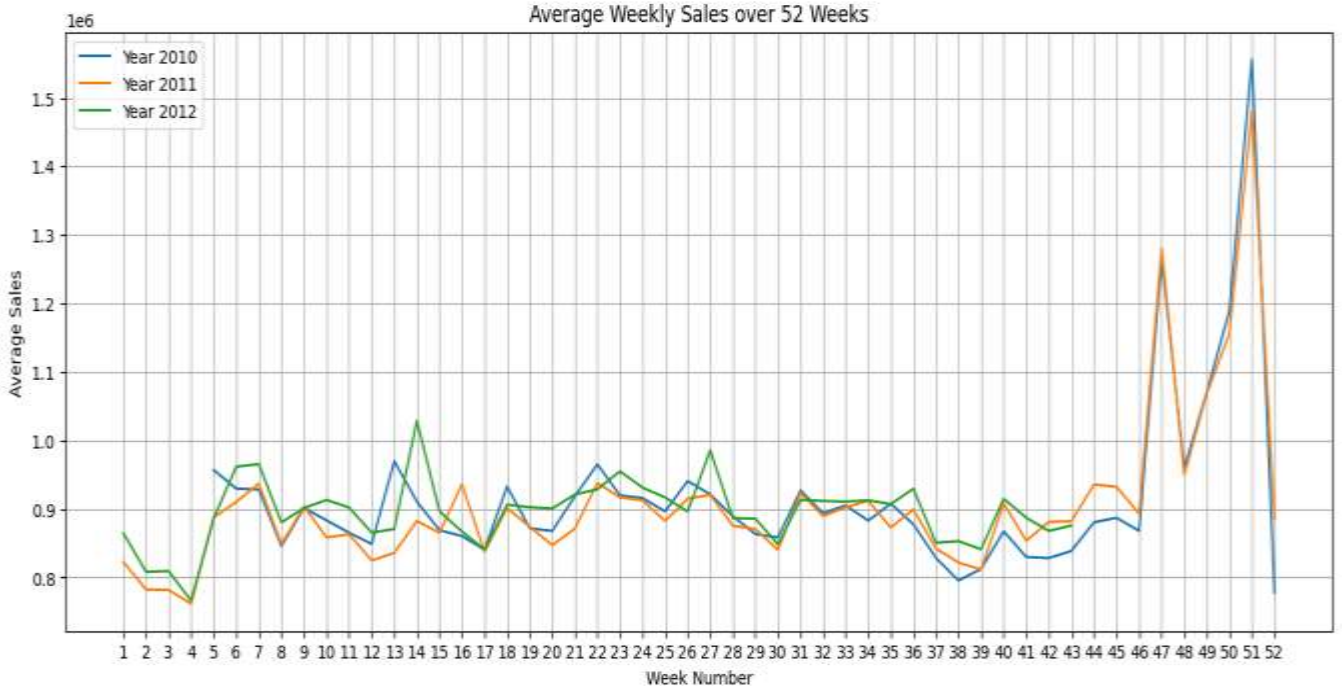


- Outliers were detected using IQR method
- Sales were log transformed to handle extreme values
- A 2.5 threshold was used both upper and lower bounds

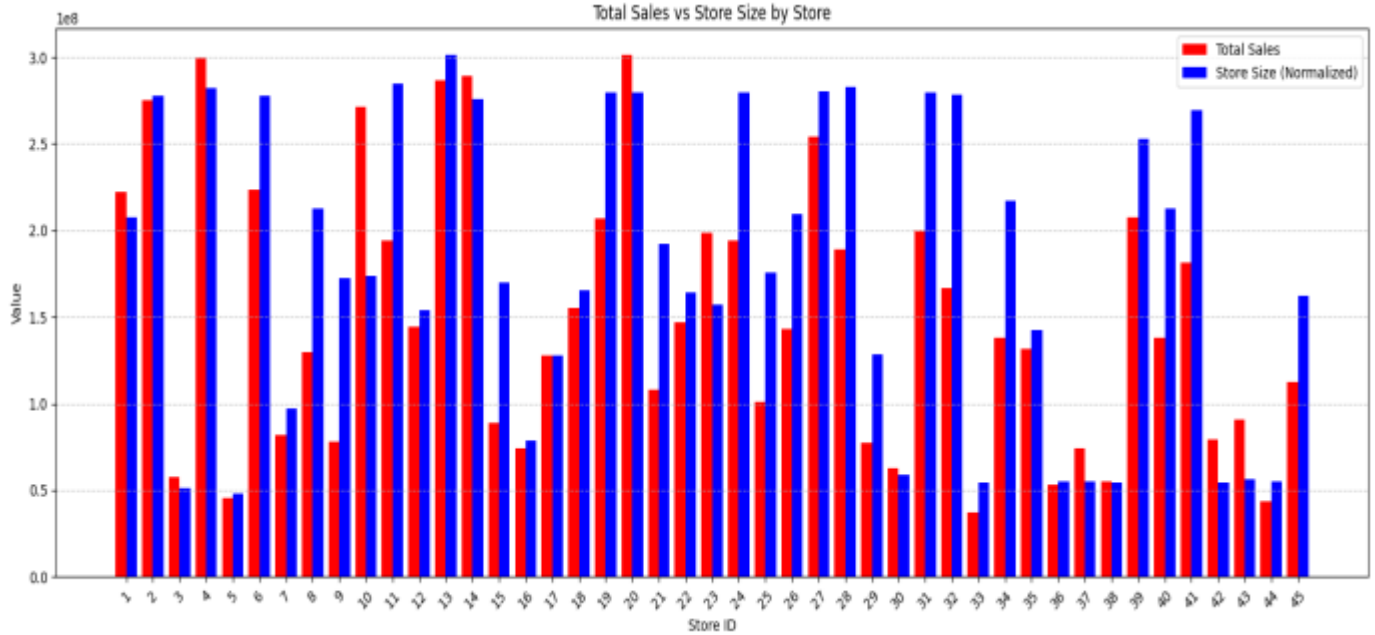
Time Series Analysis



SALES TRENDS

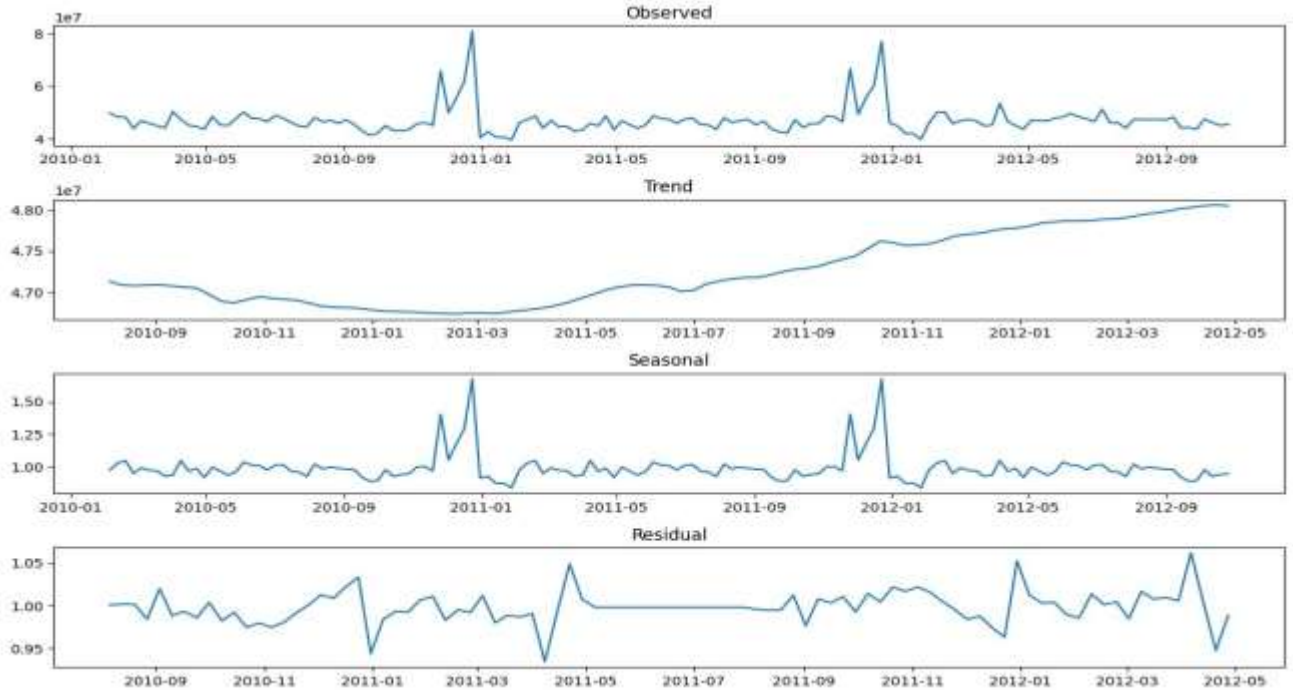


- Sales trends exhibit a recurring seasonal pattern, with increases or decreases occurring during the same set of weeks each year.



- Total sales generally scale with store size, with larger stores achieving higher sales. While a few exceptions exist, this pattern holds across most stores.

TIME SERIES DECOMPOSITION



Insights:

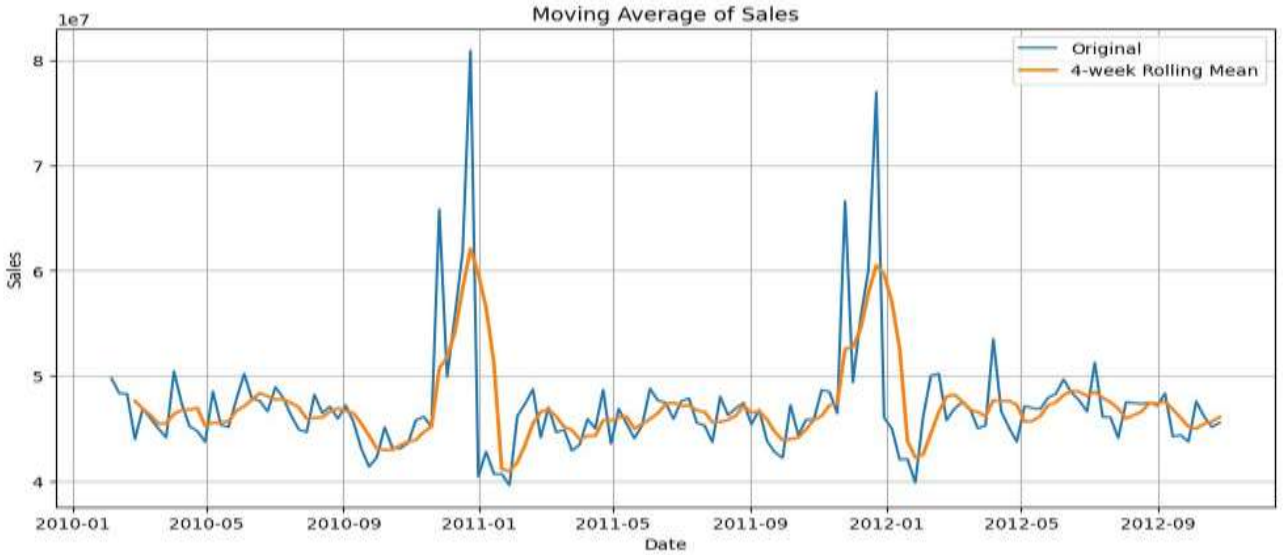
•The Observed graph shows the overall sales pattern over time, with noticeable spikes during specific periods.

•The Trend component reveals a steady upward trajectory, indicating overall growth in sales.

•The Seasonal component highlights recurring spikes, suggesting a yearly seasonal sales pattern.

•The Residual component captures random fluctuations that are not explained by the trend or seasonality.

MOVING AVERAGE OF SALES



Insights:

•The moving average (4-week rolling mean) smooths short-term fluctuations, making it easier to observe long-term trends.

•The peaks align with the spikes seen in the raw sales data, reinforcing the seasonal nature of sales.

REMOVING MULITCOLLINEARITY USING VIF

	Feature	VIF
0	store	4.363462
1	department	3.076239
2	Temperature	12.900796
3	Fuel_Price	32.950522
4	CPI	16.063119
5	Unemployment	17.015658
6	Size	6.775908
7	sales_log	18.707034

- Variance Inflation Factor (VIF) analysis was conducted to detect multicollinearity among numerical features. Features with VIF > 10 were removed to ensure a more reliable regression model.

- By removing the high-VIF features, the final model will retain only the most independent predictors, leading to better generalization.

Train – Test split

```
Final feature set:
['store', 'department', 'Temperature', 'Size']
Training set shape: (224863, 4)
Test set shape: (96371, 4)
```

- To ensure a clean dataset, all negative sales values were removed. The target variable was transformed using a logarithmic scale (sales_log) to normalize the distribution. Standardization was applied to numerical features for better model performance.The dataset was then split into training and testing sets using a 70:30 ratio, ensuring sufficient data for model learning while retaining a robust test set for evaluation.

MODEL EVALUATION FUNCTION

```python
def evaluate_model(y_true, y_pred):
    # Calculate metrics
    mse = mean_squared_error(y_true, y_pred)
    rmse = np.sqrt(mse)
    r2 = r2_score(y_true, y_pred)

    # Calculate MAPE
    mape = np.mean(np.abs((np.expm1(y_true) - np.expm1(y_pred)) / np.expm1(y_true)))

    # Calculate SMAPE
    smape = np.mean(2 * np.abs(np.expm1(y_true) - np.expm1(y_pred)) /
                    (np.abs(np.expm1(y_true)) + np.abs(np.expm1(y_pred))))

    # Print results
    print(f"Mean Squared Error (MSE): {mse}")
    print(f"Root Mean Squared Error (RMSE): {rmse}")
    print(f"R-squared (R2): {r2}")
    print(f"Mean Absolute Percentage Error (MAPE): {mape:.2f}")
    print(f"Symmetric Mean Absolute Percentage Error (SMAPE): {smape:.2f}")

    return mse, rmse, r2, mape, smape
```

EVALUATION USING LINEAR MODELS

LINEAR REGRESSION

```
Mean Squared Error (MSE): 1.0281303066115242
Root Mean Squared Error (RMSE): 1.013967606292984
R-squared (R2): 0.02540423070126141
Mean Absolute Percentage Error (MAPE): 1.21
Symmetric Mean Absolute Percentage Error (SMAPE): 0.75
```

RIDGE MODEL

```
Mean Squared Error (MSE): 1.028130303779406
Root Mean Squared Error (RMSE): 1.0139676048964317
R-squared (R2): 0.02540423338591158
Mean Absolute Percentage Error (MAPE): 1.21
Symmetric Mean Absolute Percentage Error (SMAPE): 0.75
```

LASSO MODEL

```
Mean Squared Error (MSE): 1.028130303779406
Root Mean Squared Error (RMSE): 1.0139676048964317
R-squared (R2): 0.02540423338591158
Mean Absolute Percentage Error (MAPE): 1.21
Symmetric Mean Absolute Percentage Error (SMAPE): 0.75
```

- Three linear models-Linear Regression, Ridge Regression, and Lasso Regression-were implemented to predict sales. However, the evaluation metrics (MSE, RMSE, R7, MAPE, and SMAPE) remained nearly identical across all models, indicating that linear models are not well-suited for this refined dataset.

- The Residuals vs. Predicted Sales plot further supports this conclusion. The residuals appear randomly scattered without a clear pattern, but their spread suggests that the model fails to capture complex relationships within the data.

- Given these findings, we proceed to explore more advanced machine learning models for improved performance.

PERFORMANCE EVALUATION OF ADVANCED MODEL

- To overcome the limitations of linear models, we implemented Decision Tree Regressor, Random Forest Regressor, Gradient Boosting Regressor, and Extra Trees Regressor. These models were selected for their ability to capture complex patterns and improve predictive accuracy.

- Each model was evaluated using Mean Squared Error (MSE), R-squared (R^2), Mean Absolute Percentage Error (MAPE), and Symmetric Mean Absolute Percentage Error (SMAPE). The results showed a significant improvement in predictive performance compared to traditional linear models, demonstrating the effectiveness of tree-based methods in handling the refined dataset. Furthermore, a residual analysis was performed to validate model consistency.

- The residuals for these advanced models exhibited a more random distribution compared to the linear regression models, indicating better generalization and reduced systematic bias. Among all, Random Forest Regressor outperformed the rest, striking an optimal balance between variance and bias, making it the most suitable choice for our dataset.

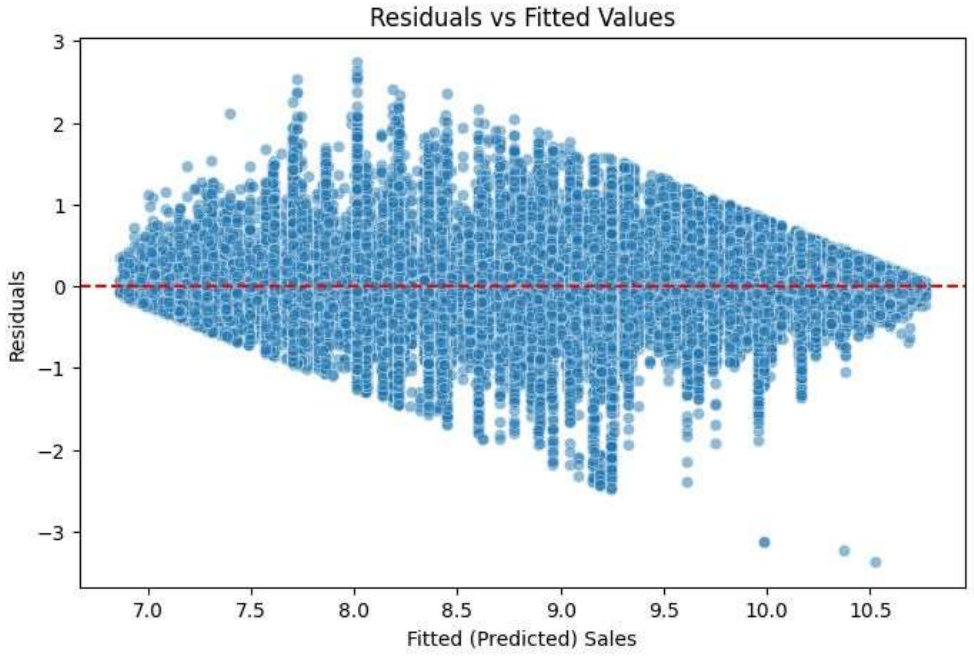
➢ **Key insights from the evaluation:**

- Random Forest performed the best, achieving the highest accuracy and lowest error.
- Extra Trees Regressor showed stable performance, but slightly underperformed compared to Random Forest.
- Decision Tree showed competitive results, but had a higher tendency to overfit.
- Gradient Boosting underperformed, possibly due to hyperparameter sensitivity.
- Residual analysis confirmed Random Forest's consistency, with a more random error distribution.

➢ **Conclusion:**

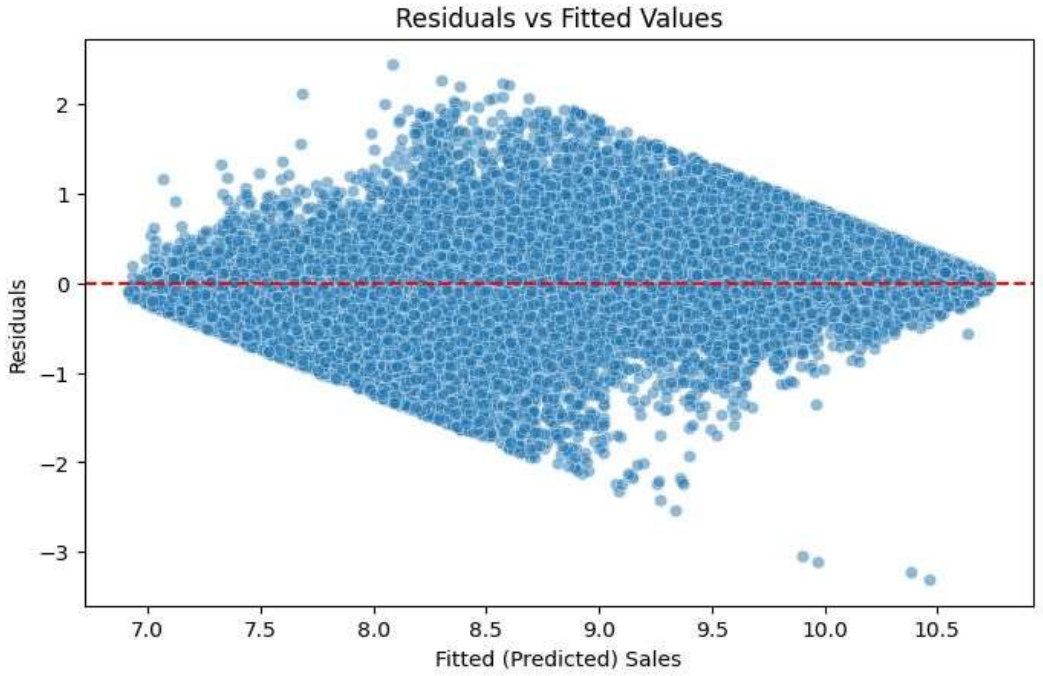
- Among all models, Random Forest was identified as the most optimal, striking the best balance between accuracy, stability, and generalization. This model will be used for further analysis and future sales predictions.Additionally, the insights gained from model evaluations and residual analysis will help refine feature selection and improve predictive performance in future iterations.

DECISION TREE MODEL



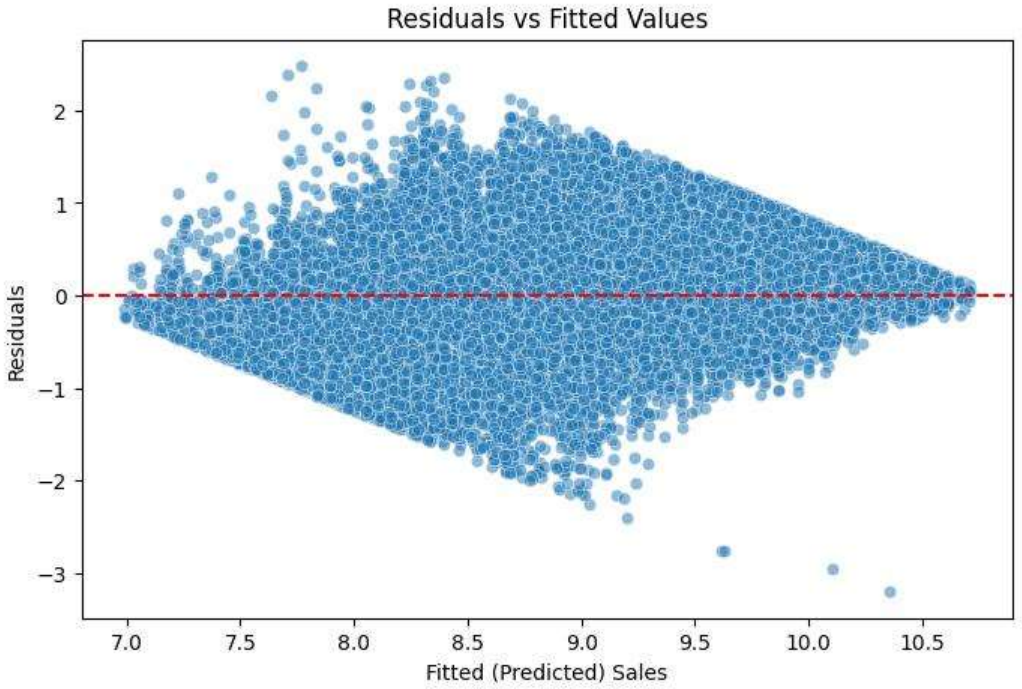

```
Mean Squared Error (MSE): 0.14418021822255692
Root Mean Squared Error (RMSE): 0.3797107033289382
R-squared (R2): 0.8633272166060494
Mean Absolute Percentage Error (MAPE): 0.29
Symmetric Mean Absolute Percentage Error (SMAPE): 0.26
```

RANDOM FOREST




```
Mean Squared Error (MSE): 0.14418021822255692
Root Mean Squared Error (RMSE): 0.3797107033289382
R-squared (R2): 0.8633272166060494
Mean Absolute Percentage Error (MAPE): 0.29
Symmetric Mean Absolute Percentage Error (SMAPE): 0.26
```

ExtraTreesRegressor




```
Mean Squared Error (MSE): 0.21224567065349445
Root Mean Squared Error (RMSE): 0.46070128136732424
R-squared (R2): 0.7988059185293249
Mean Absolute Percentage Error (MAPE): 0.39
Symmetric Mean Absolute Percentage Error (SMAPE): 0.34
```

FUTURE WORK

1.Hyperparameter Tuning & Optimization – Further fine-tuning of the Random Forest model using GridSearchCV or Bayesian Optimization to enhance predictive accuracy.

2.Ensemble Learning Approaches – Exploring stacking and boosting techniques by combining models like XGBoost, Random Forest, and Linear Regression to improve performance.

3.Advanced Feature Engineering – Identifying new relevant features, handling seasonality, and incorporating external factors like holidays and promotions.

4.Time Series Forecasting – Utilizing ARIMA and Prophet models to forecast long-term sales trends, seasonality effects, and demand fluctuations.

5.Real-world Deployment & Business Impact – Integrating the model into a live system with automated updates and dashboard visualization for business insights and decision-making.

CONCLUSION

This project began with a comprehensive Exploratory Data Analysis (EDA), where we processed raw sales data, handled missing values, removed multicollinearity, and performed various visualizations to gain initial insights. We then built and evaluated multiple machine learning models, starting with linear regression-based approaches, which proved insufficient for this dataset. Advanced models like Decision Trees, Random Forest, and Gradient Boosting were implemented, with Random Forest emerging as the most effective model.With further improvements, this predictive framework can be instrumental in forecasting future sales trends, optimizing inventory management, and aiding strategic business decisions. Moving forward, integrating time series forecasting and deploying the model in a real-world environment will enhance its impact, providing valuable insights for business growth and development.