G3

Mariana Adrian Roland Natalie

Executive summary

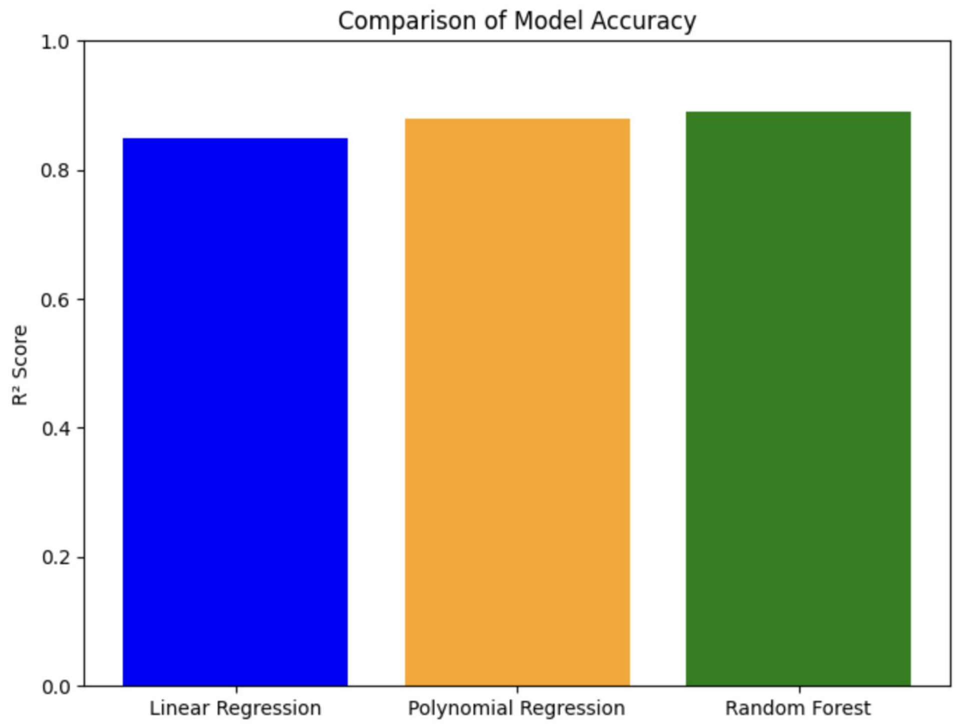
- Accuracy with training data (sales.csv)
- Best model:Polynomial

- R^2 Prediction: 0.876

Quick recap of alternatives considered:

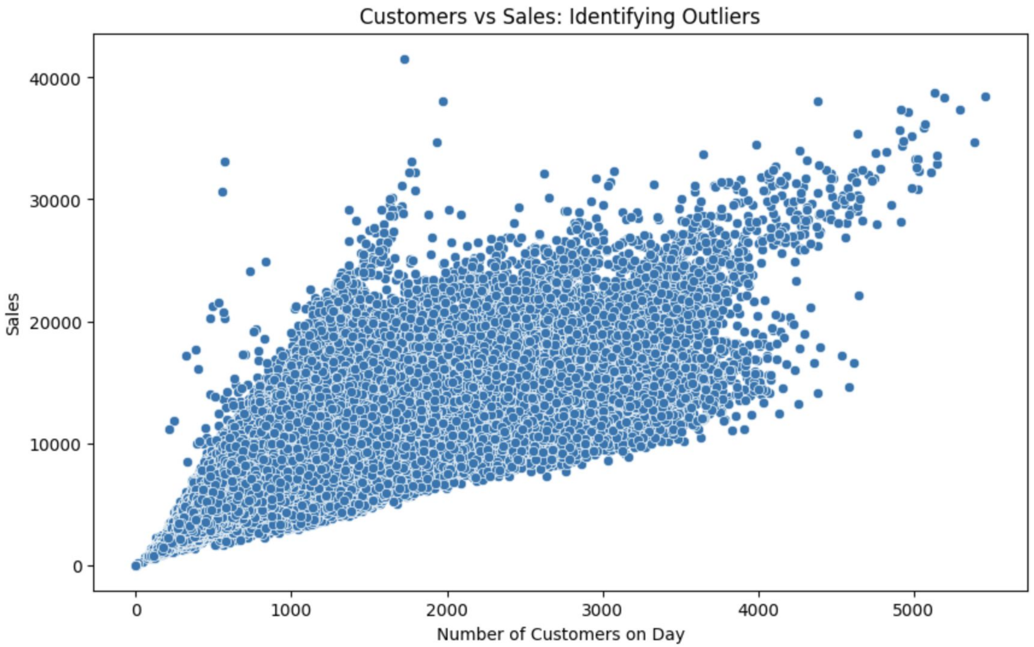
- Random Forest was accurate but too slow

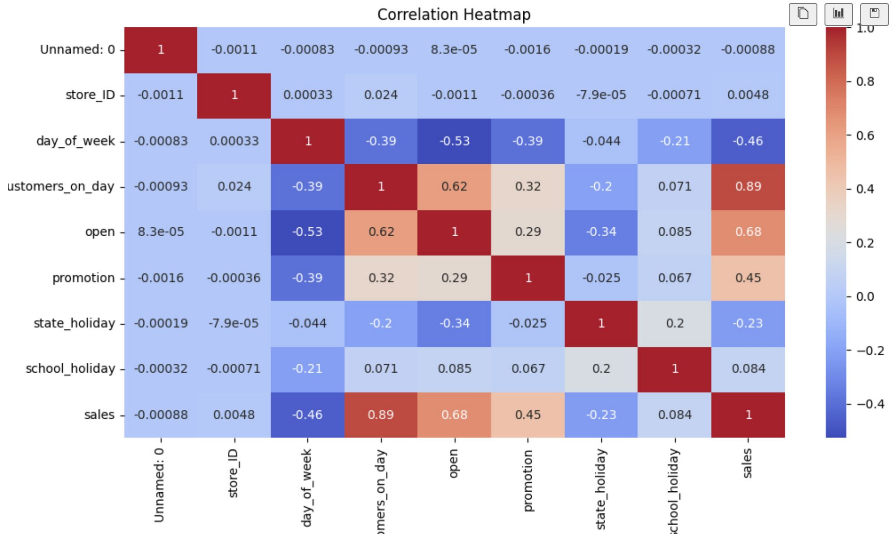
  Linear Regression Not as accurate
- Speed/ Accuracy tradeoff



Methods (preprocessing)

- Loaded the data set
- Examined the shape and the types of data
- Checked for null values
- Deleted irrelevant columns
- Converted dates from into integers
- Looked for Correlation between features to show relationship between number of people and sales and to show outliers
- Created a Correlation Heatmap





Methods (models)

- We tried 3 different models:
  - Linear Regression
  - Polynomial regression (ensemble of polynomial pre-processing combined with linear regression)
  - Random Forest

LINEAR REGRESSION MODEL

Model Used: Linear Regression

- **Library:** `scikit-learn`
- **Goal:** Predict daily **sales** using features like number of customers, promotions, holidays, and store information.
- **Approach:**

  1. **Train-Test Split**

     80% training data, 20% testing data

     Ensures model is evaluated on unseen data

  2. **Feature Scaling**
  3. **Model Training**

LINEAR REGRESSION model

Model Used: Linear Regression

🔍 Evaluation Metrics (Regression):

Metric	Meaning	Train	Test
MAE (Mean Absolute Error)	Average absolute difference between predicted and actual sales	986.76	986.88
MSE (Mean Squared Error)	Penalizes large errors	2,164,500	2,185,334
R^2 (Coefficient of Determination)	How much variance in sales is explained by the model	**0.85**	**0.85**

- $R^2 = 0.85$ → the model explains **85% of the sales variability**, which indicates a **strong predictive performance**.
- Similar results between training and test sets → **no overfitting** (good generalization).
- MAE ≈ 987 → on average, the model's predictions are within about **€987** of the true sales.

POLYNOMIAL REGRESSION model

Model Used: Polynomial Regression (degree = 4)

- **Library:** `scikit-learn`
- **Goal:** Improve sales prediction by capturing **non-linear relationships** between features (e.g., promotions, holidays, customer counts).

🔍 Evaluation Metrics (Regression)

Metric	Meaning	Train	Test
MAE	Average absolute difference between predicted and actual sales	897.21	896.42
MSE	Penalizes large errors	1,819,521	1,832,625
R^2	Variance in sales explained by model	**0.88**	**0.88**

R^2 = 0.88 → The model explains **88% of the sales variability**, showing a better fit than Linear Regression (0.85).

MAE ≈ 896 → Average error dropped by ~90€, meaning the model is more accurate.

Train/Test R^2 are equal → The model generalizes well (no overfitting).

Degree = 4 achieved a good balance between accuracy and computation time.

RANDOMFOREST model

Model Used: Randomforest

1. `TimeSeriesSplit`
    - Used because sales data has a time component (avoids “seeing the future”).
    - Splits the dataset chronologically into folds for cross-validation.
2. `RandomForestRegressor`
    - Trains multiple trees (ensemble) to learn complex relationships.
3. `cross_val_score`
    - Evaluates model performance using **R^2** across folds for reliability.

20 MINUTES

Metric	Fold 1	Fold 2	Mean
R^2 (Explained Variance)	0.93	0.95	**0.94**

Takeaways

- Recap / conclusions
- Challenges were making sure the data was clean and usable, making sure the number of columns were the same for training and test data
- Key learnings :-If you scaled the data during training you had to to the inverse at the end