# Store Sales Forecasting

This project forecasts store sales using historical data, oil prices, holidays, and store metadata. The goal is to predict daily sales using machine learning models.

## 📁 Project Structure

- `Store forecasting Assessment.ipynb` – The main Jupyter notebook containing data preprocessing, feature engineering, and model training (Random Forest, XGBoost).

- `train.csv` – Training dataset.

- `test.csv` – Test dataset for prediction.

- `stores.csv` – Store metadata.

- `oil.csv` – Daily oil prices.

- `holidays\_events.csv` – Calendar of holidays and events.

## ▶️ How to Run

1. Clone the repository or download the notebook and datasets.

2. Make sure the following dependencies are installed:

- pandas

- numpy

- scikit-learn

- xgboost

- matplotlib / seaborn (for visualization)

3. Open the notebook and run the cells step-by-step.

You can run it using:

```bash

jupyter notebook "Store forecasting Assessment.ipynb"

**Models Used**

* Random Forest Regressor
* XGBoost Regressor

**Evaluation Metrics**

Models are evaluated using common regression metrics like RMSE and MAE.

**Notes**

* Ensure all CSV files are placed in the correct path (update the paths if needed).
* The notebook merges various data sources and extracts time-based features before training.

Key Insights:

- \*\*XGBoost generally yields better performance\*\* in forecasting due to its ability to capture complex interactions and apply regularization.

- \*\*Feature engineering\*\*, especially merging external datasets (oil, holidays, store info), significantly improves model accuracy.

- Time-based features (day, week, month) play a critical role in capturing sales seasonality.

---

Would you like me to clean up the notebook or help visualize the model performances side-by-side? &#8203;:contentReference[oaicite:0]{index=0}&#8203;