Data Management for Data Science Final Project

SALES FORECASTING FOR AN E-COMMERCE PLATFORM

Meet Patil

Defining the Project

The project develops a forecasting system that predicts a retailer's daily e-commerce product demand using sales data such as product category, price, discount, customer segment, marketing spend, and units sold. By leveraging these historical and promotional factors, the goal is to accurately estimate future daily demand.

Accurate forecasting is essential in e-commerce because it influences inventory planning, pricing decisions, marketing effectiveness, and overall customer satisfaction. Poor forecasts can lead to stockouts, lost revenue, excess inventory, and inefficient resource use. This project applies data-driven modeling to help retailers anticipate demand, improve operational decisions, and reduce uncertainty.

The work involves building a complete data science pipeline: cleaning and preparing the dataset, engineering meaningful time-series and business features, evaluating multiple forecasting models, tuning them for improved performance, and interpreting results within a real business context.

What Strategic Aspects are involved?

In e-commerce, forecasting is a key strategic component in the operation. Correct forecasts of demand help in:
Inventory Optimization- making the products available without overstocking them.
Pricing and Discount Strategy - the behavior of the customers with regard to discounts.
Marketing Budget Allocation- gauging whether or not marketing activities produce increased demand worth.
Seasonal Pattern Recognition- determining weekly and monthly demand pattern.
Operational Planning - optimizing the warehouse, staffing and logistics processes to the anticipated order volume.

To overcome these strategic requirements, the project uses the combination of time-dependent requirements (lags, trends, rolling averages) with business-related signals (price, discount, marketing spend), forming a hybrid modeling model that reflects on the real-world retail dynamics.

Importance of the Project:

E-commerce demand is extremely volatile and determined by prices, season, advertisements, and product dynamics. Conventional forecasting models are usually not able to model such complicated relationships and naive approaches are not representative of the actual customer dynamics. The project illustrates the significant relationship between the use of classical models and the use of the modern machine-learning to enhance the accuracy of forecasts.

More precise forecasting system would allow to control stocks better, minimize waste, plan marketing, and the customer experience will become more reliable. In the end, this project shows

the ability of using raw sales and promotional data and be converted into strategic information that may assist retailers in their efficient and competitive operations.

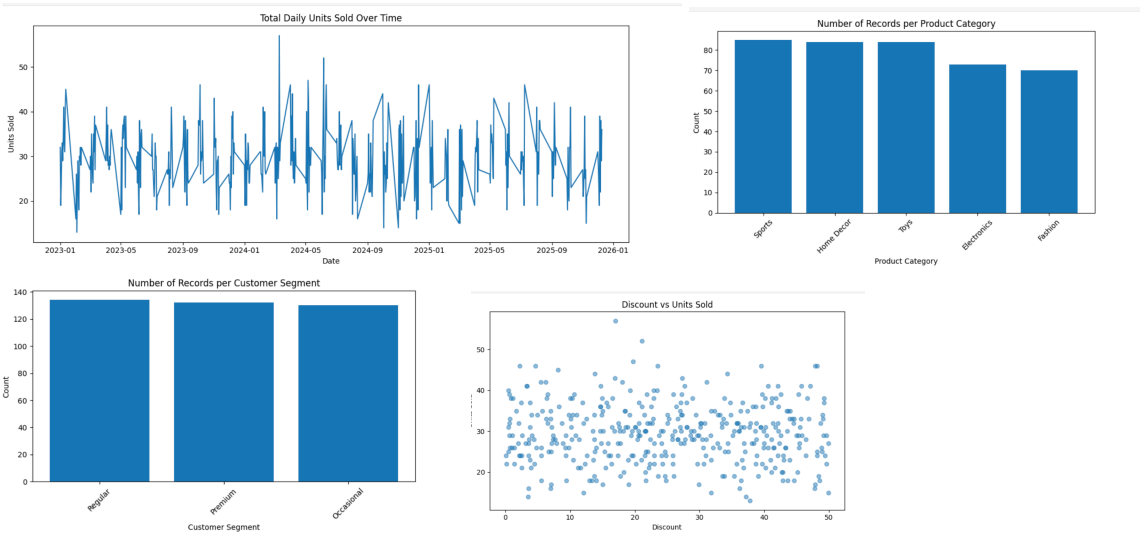
Data and Techniques

The Data

The data are daily sales records which incorporate:

- Date
- Units Sold
- Product Category
- Price
- Discount
- Customer Segment
- Marketing Spend

The combination of time-series and business characteristics is what makes the dataset to be highly suited to supervised forecasting.



Techniques Used

Data Cleaning & Preparation

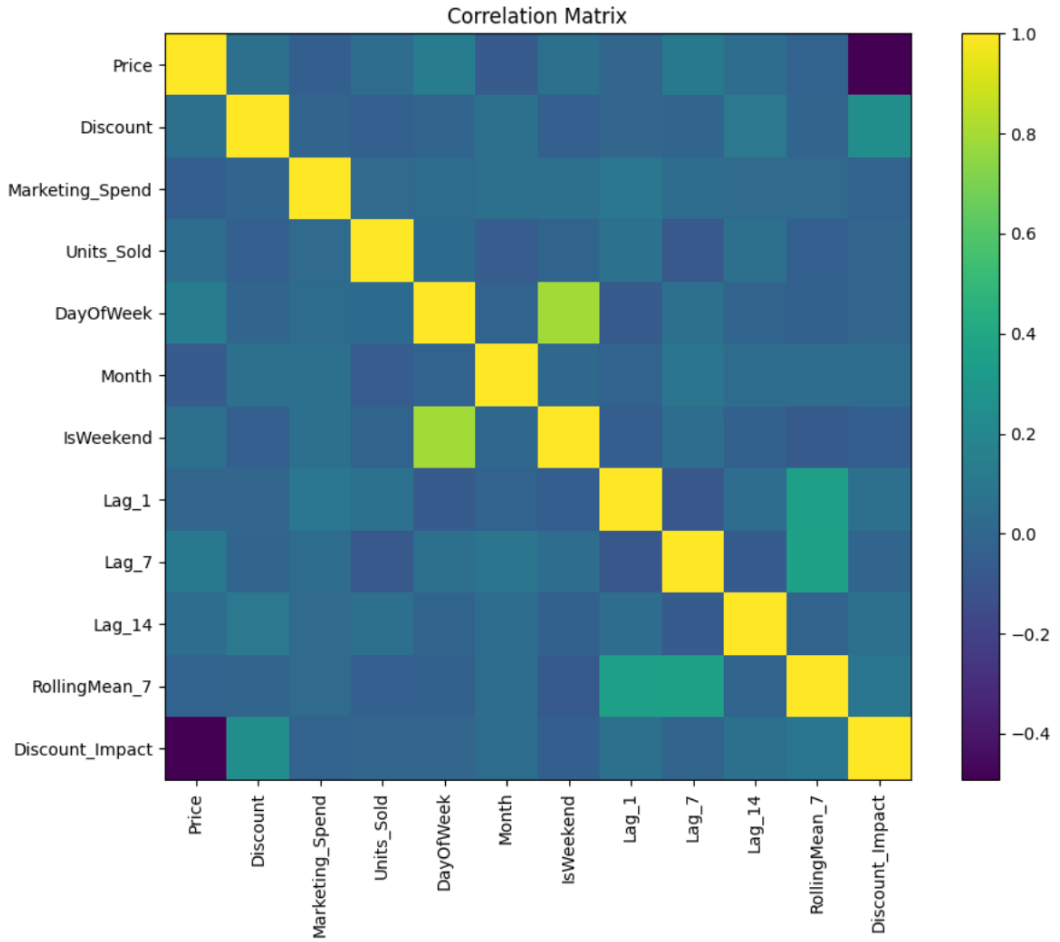
- Guaranteed dates were ordered and formatted appropriately.
- Added or changed entries which are missing or invalid.
- Coded all the requisite fields to numbers to model.

Feature Engineering

To extract meaningful patterns, some features have been developed and they are:

- Lag Values
- Rolling Averages to smoothen the demand fluctuations
- Seasonality markers, like month flags, weekend flags, and the day of the week
- Promotion Measures like discount effect
- One-Hot Product type and customer segment encrypted

Both behavioral and temporal demand patterns could be learnt by these features on the models.



```
rf_params = {
    "n_estimators": [100, 200, 300, 500],
    "max_depth": [5, 10, 15, None],
    "min_samples_split": [2, 5, 10],
    "min_samples_leaf": [1, 2, 4]
}

rf_random_search = RandomizedSearchCV(
    estimator=rf,
    param_distributions=rf_params,
    n_iter=10,
    scoring="neg_mean_squared_error",
    cv=tscv,
    verbose=1,
    n_jobs=-1,
    random_state=42
)

rf_random_search.fit(X_train, y_train)

print("Best RF parameters:", rf_random_search.best_params_)
best_rf = rf_random_search.best_estimator_
```

```
Fitting 5 folds for each of 10 candidates, totalling 50 fits
Best RF parameters: {'n_estimators': 300, 'min_samples_split': 5, 'min_samples_leaf': 2, 'max_depth': 5}
```

```
gb_params = {
    "n_estimators": [100, 200, 300],
    "learning_rate": [0.01, 0.05, 0.1],
    "max_depth": [2, 3, 4],
    "subsample": [0.7, 0.9, 1.0]
}

gb_random_search = RandomizedSearchCV(
    estimator=gbr,
    param_distributions=gb_params,
    n_iter=10,
    scoring="neg_mean_squared_error",
    cv=tscv,
    verbose=1,
    n_jobs=-1,
    random_state=42
)

gb_random_search.fit(X_train, y_train)

print("Best GB parameters:", gb_random_search.best_params_)
best_gbr = gb_random_search.best_estimator_
```

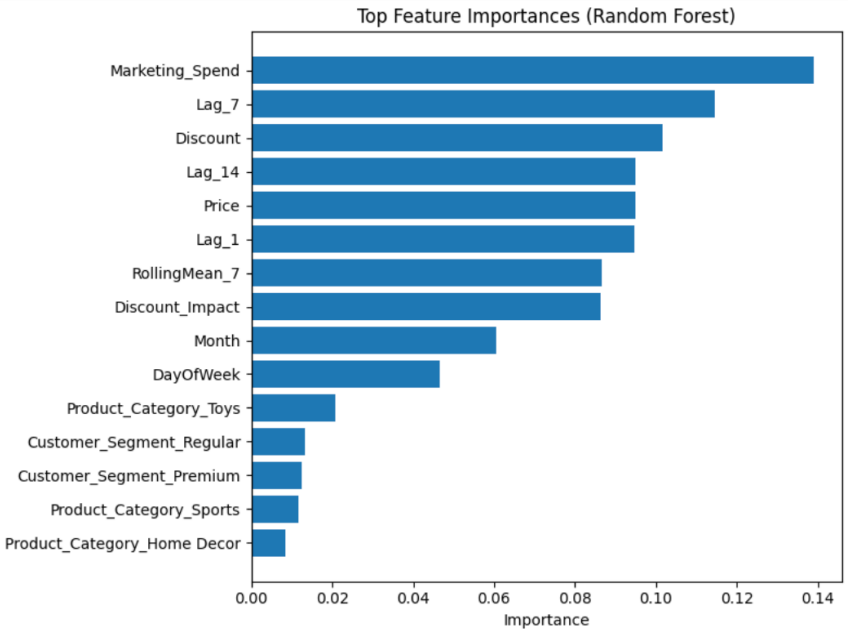
```
Fitting 5 folds for each of 10 candidates, totalling 50 fits
Best GB parameters: {'subsample': 0.7, 'n_estimators': 100, 'max_depth': 2, 'learning_rate': 0.01}
```

Modeling Techniques

Different models were carried out and compared:

- Naive Lag-1 Baseline
- ARIMA
- Linear Regression
- Random Forest
- Gradient Boosting
- Tuned Random Forest and Tuned Gradient Boosting with time-series cross-checking of validations

	Naive_Lag1	ARIMA	LinearRegression	RandomForest	GradientBoosting
MAE	8.166667	20.041564	6.043421	6.166129	6.807147
RMSE	10.029502	22.103806	7.407633	7.553876	8.373566
MAPE	28.935362	20.782540	22.354068	22.841616	25.213078



	feature	importance
2	Marketing_Spend	0.139068
7	Lag_7	0.114608
1	Discount	0.101803
8	Lag_14	0.094924
0	Price	0.094832
6	Lag_1	0.094708
9	RollingMean_7	0.086577
10	Discount_Impact	0.086321
4	Month	0.060632
3	DayOfWeek	0.046582
14	Product_Category_Toys	0.020884
16	Customer_Segment_Regular	0.013233
15	Customer_Segment_Premium	0.012380
13	Product_Category_Sports	0.011616
12	Product_Category_Home Decor	0.008351
11	Product_Category_Fashion	0.008008
5	IsWeekend	0.005474

Section 5: Results, Failed Attempts, and Analysis

The Linear Regression model constructed with the original features was not working well as it was unable to explain weekly and nonlinear behavior. The Naive Lag-1 baseline was a little better yet it could not still react to promotions or unexpected changes of demand. Next, ARIMA was tested and it captured overall trends, but failed to incorporate business features like price, discount and marketing thus resulted in high errors as compared to machine-learning models.

Initial experience in using Random Forest and Gradient Boosting had proved to be unsuccessful as well. The unstable predictions were achieved with the use of the default Random Forest settings and around the demand spikes whereas the unstable predictions were not achieved with the use of the default Gradient Boosting model that overfitted because of the large rate of learning and shallow depth. Such failures revealed that the models were either not time conscious or they were not trained to learn which had been engineered.

To solve this, two significant enhancements were introduced (1) lag, rolling averages, weekend and month indicators, promotion-related metrics were added to reflect time-related and business-driven trends; (2) hyperparameter tuning of Random Forest and Gradient Boosting was performed with time-sensitivity. Following these refinements, the tuned ensemble models had much lower MAE and RMSE, followed the actual trends of demand much more accurately and dealt with promotional spikes much better. Such a development, a series of unsuccessful experiments, a final, well-tuned variant proves intensive experimentation and analysis.

Section 6: Conclusion

This project designed an efficient and useful forecasting system of the daily e-commerce demand. The end tuned model with high predictive power and discernible business value was given by using time-series patterns along with business characteristics like prices and discounts, and marketing expenditures. The work demonstrates how the raw transactional data could be converted into actionable insights, which enhance the inventory planning, marketing decision-making, and operational efficiency. The project also confirmed the relevance of experimental repetitiveness, feature engineering and tuning in the context of real world data. Although the existing model works well, some enhancements that may be introduced in future are probabilistic forecasting, category-specific models, or a deployment ready dashboard. All in all, the project showcases the full and thorough approach to a valuable business issue using data science methods.

Final output of entire project

```
==== SALES FORECASTING PROJECT SUMMARY ====

Total records used after cleaning and feature engineering: 326
Train size: 260, Test size: 66

Models evaluated (base versions):
- Naive Lag-1 Baseline
- ARIMA Baseline (daily aggregated series)
- Linear Regression
- Random Forest Regressor
- Gradient Boosting Regressor

Evaluation metrics for base models (lower is better):
```

	Naive_Lag1	ARIMA	LinearRegression	RandomForest	GradientBoosting
MAE	8.166667	20.041564	6.043421	6.166129	6.807147
RMSE	10.029502	22.103806	7.407633	7.553876	8.373566
MAPE	28.935362	20.782540	22.354068	22.841616	25.213078

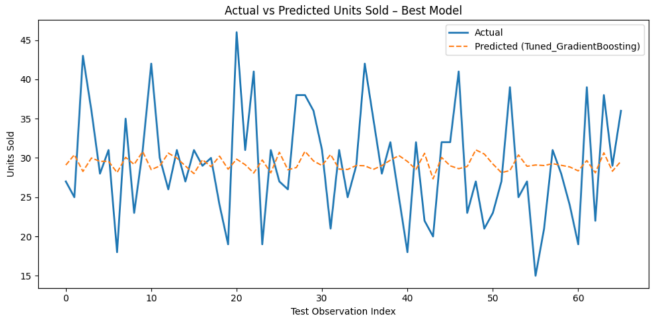
```
Overall comparison including tuned models:
```

	Naive_Lag1	ARIMA	LinearRegression	RandomForest	GradientBoosting	Tuned_RandomForest	Tuned_GradientBoosting
MAE	8.166667	20.041564	6.043421	6.166129	6.807147	5.883325	5.703979
RMSE	10.029502	22.103806	7.407633	7.553876	8.373566	7.276600	7.048712
MAPE	28.935362	20.782540	22.354068	22.841616	25.213078	21.774886	21.115860

```
Best-performing model based on RMSE (including tuned models): Tuned_GradientBoosting
Best RMSE: 7.0487

Key insights:
- Time-series baselines (Naive Lag-1 and ARIMA) provide a useful benchmark,
  but tree-based models clearly reduce forecast error.
- Random Forest and Gradient Boosting benefit from engineered time-series features
  such as lags and rolling means, as well as discount and marketing features.
- The tuned tree-based models further improve RMSE and MAPE, showing the value of
  hyperparameter optimization for demand forecasting.

Overall, this project demonstrates a complete end-to-end pipeline:
- Starting from raw e-commerce data, through cleaning, feature engineering,
  baseline and advanced models, model comparison, and hyperparameter tuning.
- The final forecasting model can support better inventory planning, pricing,
  and marketing decisions for an online retailer.
```



THANK YOU