Week 4 Report

Sales Forecasting Based on Past Sales Statistics

Nguyen Minh Duc - 20225437

Week 4 Objectives:

- Implement machine learning models (e.g., tree-based models such as XGBoost/LightGBM)
- Initial training for machine learning models.

---

1 Model Overview

1.1 LightGBM: Mechanism and Rationale

Light Gradient Boosting Machine (LightGBM) is a tree-based gradient boosting framework that constructs an ensemble of decision trees in a sequential manner. At each iteration, a new tree is trained to minimize the residual errors of the current ensemble, typically by optimizing a differentiable loss function via gradient descent. Unlike traditional gradient boosting implementations, LightGBM grows trees in a *leaf-wise* manner rather than level-wise, which allows it to focus splits on regions of the feature space that yield the largest reduction in loss. This strategy often leads to faster convergence and improved predictive performance, especially in tabular datasets with complex non-linear interactions.
LightGBM further improves computational efficiency through techniques such as histogram-based feature binning and gradient-based one-side sampling. These properties make it particularly suitable for datasets with a relatively large number of engineered features, as is the case in this project.

1.2 Model Configuration and Usage in This Project

In this project, LightGBM is employed as a machine learning baseline on the weekly aggregated sales time series augmented with behavioral and transactional features. The model is trained and evaluated using a *walk-forward evaluation* strategy, similar to the baselines.
Key configuration choices include:

- **Minimum history**: A minimum of six weeks (`MIN_HISTORY = 6`) is required before predictions are made. This is identical to the settings of baseline models and ensures that the model has sufficient observations to estimate meaningful splits and reduces instability in early predictions.
- **Feature selection**: From the full pool of engineered features (lags, rolling statistics, behavioral ratios, and category shares), only features with non-zero usage in tree splits were retained. Empirical testing showed that removing a large fraction of unused or weakly informative features had no impact on predictive performance, indicating strong redundancy in the original feature set.

- **Hyperparameters**: Conservative default hyperparameters were used (limited tree depth, moderate learning rate) to reduce overfitting given the small sample size (approx. 65 weekly observations).

1.3 Model Performance

The LightGBM model was evaluated using three metrics: Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and Mean Error (ME). RMSE captures overall forecast accuracy, MAPE provides a scale-independent measure of relative error, and ME reflects systematic bias.

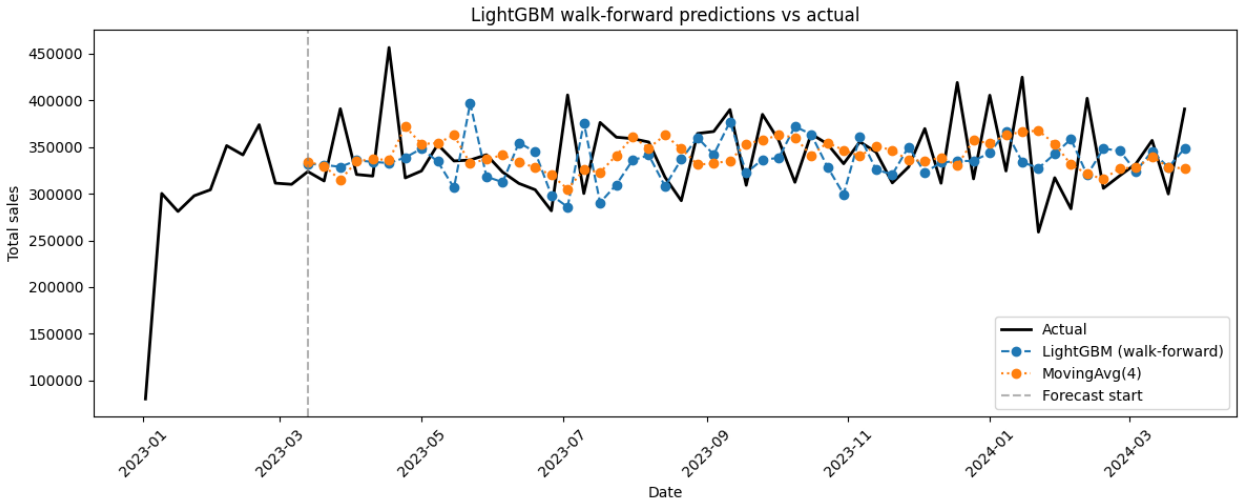

Figure 1: Predictions of LightGBM as compared with MovingAvg(4)

Table 1: Comparison with Baseline and Statistical Models

Model	RMSE	MAPE (%)	ME
LightGBM	**43 743.25**	**9.93**	**84.16**
Naive	63 428.41	13.90	-664.72
Moving Average	43 698.10	9.99	-1 325.87
Seasonal Naive	52 395.30	12.14	-2 456.74
ARIMA(1,1,2)	54 110.44	11.81	1 239.32
Prophet	52 868.51	13.96	28 891.20

1.4 Discussion and Conclusions

The results demonstrate that LightGBM's performance is comparable to that of Moving Average, the best model among baseline and statistical models. In particular, the near-zero Mean Error suggests that the

model exhibits minimal systematic bias, in contrast to Prophet and ARIMA, which tend to underpredict sales. This is beneficial from a commercial viewpoint, as the model accurately predicts how much restocking is needed.

2 Feature Selection and Comparison

2.1 Methodology

To assess which engineered features contribute most to the predictive performance of the LightGBM model, we analyze *feature importance* based on information gain. In tree-based gradient boosting models, features with higher gain provide more informative partitions of the data and are more influential in model predictions.

2.2 Top Features by Gain

Table 2 presents the most influential features in a representative model run, ranked by total information gain. The split count indicates how frequently each feature was used for decision splits.

Table 2: Top Features by Information Gain (Single Run)

Feature	Gain	Split Count
total_sales_lag_1	1.49×10^{11}	186
total_sales_rollstd_3	1.30×10^{11}	136
avg_qty_per_order	1.23×10^{11}	114
total_sales_lag_4	1.16×10^{11}	153
pct_returning_lag_1	1.01×10^{11}	150
avg_session_duration_lag_1	7.40×10^{10}	200
total_quantity_lag_1	6.72×10^{10}	113
avg_pages_viewed_lag_1	5.87×10^{10}	117
week_idx	5.63×10^{10}	134
total_sales_lag_2	3.99×10^{10}	122
total_sales_rollmean_3	2.64×10^{10}	70
total_sales_rollmean_4	2.44×10^{10}	82

2.3 Aggregated Feature Importance Across Walk-Forward Steps

To evaluate feature stability across time, gains were aggregated over all walk-forward training iterations. Table 3 reports both the cumulative gain and the average gain per usage.

Table 3: Aggregated Feature Importance Across Walk-Forward Evaluation

Feature	Gain (Sum)	Gain (Average)
total_sales_lag_4	4.47×10^{12}	8.13×10^{10}
avg_qty_per_order	4.16×10^{12}	7.56×10^{10}
total_sales_lag_1	3.26×10^{12}	5.92×10^{10}
total_sales_rollstd_3	2.84×10^{12}	5.17×10^{10}
avg_pages_viewed_lag_1	2.21×10^{12}	4.01×10^{10}
total_quantity_lag_1	1.83×10^{12}	3.34×10^{10}
avg_session_duration_lag_1	1.77×10^{12}	3.22×10^{10}
week_idx	1.69×10^{12}	3.08×10^{10}
total_sales_lag_2	1.44×10^{12}	2.61×10^{10}
pct_returning_lag_1	1.41×10^{12}	2.56×10^{10}
total_sales_rollmean_4	1.05×10^{12}	1.90×10^{10}
total_sales_rollmean_3	6.92×10^{11}	1.26×10^{10}

2.4 Remarks and Interpretation

Several important observations can be drawn from the feature importance analysis:

- **Dominance of recent lags**: Lagged sales variables (especially 1-week and 4-week lags) consistently provide the largest information gains, confirming that short-term momentum is the strongest predictor of near-future sales.
- **Volatility matters**: Rolling standard deviation features outperform rolling means in terms of gain, suggesting that recent sales variability contains valuable predictive signals beyond average levels.
- **Behavioral features are informative**: Variables such as average quantity per order, session duration, pages viewed, and the proportion of returning customers contribute meaningfully, indicating that user behavior adds complementary information to pure sales history.

- **Limited role of calendar indexing**: The week index appears with moderate importance, capturing coarse temporal progression rather than true seasonality, which aligns with earlier EDA findings showing weak seasonal patterns.
- **Feature redundancy**: Many engineered features were either unused or had negligible gain. Removing large subsets of such features did not affect predictive performance, highlighting substantial redundancy and reinforcing the need for parsimonious feature selection in small-sample settings.

Overall, the analysis confirms that LightGBM primarily relies on a compact set of recent sales and behavioral indicators. This explains both the strong performance of the model and the observed robustness to aggressive feature pruning.