My Strategy for the Challenge

I started off with Exploratory Data Analysis (EDA), spending a good amount of time understanding the dataset. The focus was on both univariate and multivariate analysis to identify patterns, correlations, and potential transformations. This helped in making key decisions for feature engineering.

Model Selection & Early Experiments

For encoding, I chose target encoding for Outlet_Identifier and label encoding for other categorical variables, as I was clear from the beginning that I wanted to use tree-based models.

Initially, I explored RandomForestRegressor, but after evaluating its limitations, I switched to LightGBM due to its advantages with categorical variables, faster training time, and better handling of large datasets. This challenge also gave me an opportunity to explore LightGBM, which I hadn't used extensively before.

Feature Engineering - The Messy Phase

This was where things got complicated. I tried various permutations & combinations (P&C) of features, including:

- Target encoding for different outlet-related features.
- Feature importance analysis to prioritize variables.
- Principal Component Analysis (PCA) to reduce dimensionality.

However, instead of improving performance, the feature set became too complex, and my leaderboard rank remained above 2000. Even hyperparameter tuning didn't bring significant improvements.

Reassessing the Approach - The Breakthrough

At this point, I decided to step back and rethink my feature engineering strategy. Instead of sequentially dropping features, I flipped the approach:

- Started with a minimal feature set.
- Gradually added new features, monitoring performance at each step.

This reverse approach worked well, and LightGBM's strength in handling categorical variables helped refine the feature set further.

For hyperparameter tuning, I used Optuna, which streamlined the process of finding the best combination of parameters. As I fine-tuned my features, my rank started improving.

Final Refinements & Success

After extensive P&Cs on the feature set, I reached Rank ~1100. To push further, I:

- Reduced features to a bare minimum while maintaining performance.
- Fine-tuned hyperparameters further, as the leaderboard rankings were now shifting by decimal points.

The final breakthrough came with the right balance of feature selection and hyperparameter tuning, which pushed my rank to #730. That's when I decided to call it a day.

Key Takeaways & Learnings

- Feature selection matters more than the number of features. A clean, well-selected set performs much better than an overloaded one.

- Iterative experimentation is key. The best results came when I built up the feature set gradually rather than overcomplicating it.
- Tree-based models like LightGBM shine with the right tuning. Their ability to handle categorical variables efficiently was a game-changer.
- Validation RMSE is not always the best indicator. Towards the end, leaderboard rank was shifting with decimal-point differences, showing the importance of fine-tuning.

This challenge was a great learning experience, allowing me to explore LightGBM, feature engineering strategies, and hyperparameter tuning techniques in depth.