# Sales Forecasting – Project One-Pager

## Business Problem

The dataset contains detailed information at both the item and store levels (e.g., item_id, item_details, store_id, store_details). Each item is sold in 1–10 stores (on average, around 5). The objective is to accurately predict sales while handling missing values, performing categorical encoding, and minimizing the Root Mean Squared Error (RMSE).

## Proposed Approach

### Modeling Strategy

- Baseline: Use gradient boosting models (LightGBM, Random Forest, XGBoost) at the item−store level to capture store effects.
- Alternative: A two-step method—predict total store sales, then distribute across items. However, since tree-based models effectively capture non-linear relationships, the baseline is sufficient.

### Data Preparation

- One-Hot Encoding: Applied to Item_Type, Outlet_Identifier, and Outlet_Type (nominal variables with low cardinality).
- Label Encoding: Applied to Item_Identifier (high cardinality, prevents excessive columns), Item_Fat_Content, Outlet_Size, and Outlet_Location_Type. For ordinal variables like Outlet_Size ('Small', 'Medium', 'High'), label encoding preserves order.
- Standardized category names (e.g., merged 'Low Fat' and 'LF', 'Regular' and 'reg').

### Handling Missing Values

- Item_Weight: Used a dictionary mapping each Item_Identifier to its known weight to fill missing values. Remaining gaps were filled with the global mean.
- Outlet_Size: Applied k-NN imputation for the 3 outlets with missing values.
- Result: All missing values successfully resolved.

### Exploratory Analysis

- No significant outliers found in Item_Weight, MRP, or Visibility.

## Results

Initial experiments were conducted with LightGBM, Random Forest, and XGBoost. Performance improved after refining missing value imputation, and encoding strategies.

### Model Performance Before Hyperparameter Optimization

| | |
|---|---|
| LightGBM | RMSE: 900.43, MAPE: 0.504 |
| Random Forest | RMSE: 421.30, MAPE: 0.211 |
| XGBoost | RMSE: 662.39, MAPE: 0.391 |

### Model Performance After Hyperparameter Optimization (3-Fold CV)

| | |
|---|---|
| Random Forest | Best RMSE: 1098.05 |
| XGBoost | Best RMSE: 1080.38 |
| LightGBM | Best RMSE: 1095.21 |

XGBoost and Random Forest exhibited overfitting. The final model chosen was LightGBM, which offered a balanced trade-off between accuracy and generalization.