**BIG MART SALES PREDICTION**

**Problem understanding:**

To build a robust predictive model that accurately forecasts the sales of individual products within specific Big Mart outlets. This is a regression problem, where the target variable is Item\_Outlet\_Sales.

The primary goal for BigMart is to gain actionable insights into which product and outlet characteristics are the most significant drivers of sales.

**Data understanding:**

Analyze the distribution of variables, identify relationships between features and the target, spot outliers, and understand the nature of missing values. This step is crucial for informing feature engineering.

The Outlet sales is correlated more with Item\_MRP (0.57)

The Starchy food has more sales followed by seafood compared to other Items

Also observed that the Health and Hygiene has outliers wrt to the MRP of Items infers more pricy for the Items available

Seafood is the only Item with No outliers with sales

There are two major fields where there are missing data points that is crucial for analysis that are Item weight and Outlet size. These are important because the sales depends on the outlet size and Item weight. Also there is a column Item Visibility which plays a key role in the outlet sales where the visibility less than 0.2 has more sales, also observed that there are zeros in Item visibility which are equivalent to null values in other columns.

So for imputing the Visibility and Weight, we performed the grouping with Item Identity because the weight and visibility of Item are dependent on the Item type. As the Weight is normally distributed the imputation is done with the mean of the data. Visibility is left skewed and was imputed with mean to preserve the central tendency of data.

For Imputing the outlet size, the outlet size depend on the type of outlet as well as location so based on observations imputed the size of outlet to small where there is Grocery store as store type and location as tier2

In Item fat content there are values such as LF reg which corresponds to Low fat and Regular, and change the values to numeric variable for better model building

Feature engineering strategy

Created feature called Age of outlet as the sales we are predicting is wrt to 2013 the years value wouldn’t help us in analysis and prediction so the column is changed to Age

Item\_MRP to Weight column is created as the Item weight the ration has impact on the sales that happen because the better the ratio the better the sales can be. So the column is created

Item\_MRP\_segmentation segmenting the values to categories for better model prediction as this should be categerised

Created Label Encoding and One hot encoding to test the data with models for accuracy building but for Catboost only few columns are considered for Label encoding.

Modeling strategy:

Linear regression: Started with Linear Regression as a solid baseline. It's like checking the fundamental pulse of the data. If sales were simply a straightforward combination of factors like price, visibility, or store size, Linear Regression would quickly show that.

Ridge regression: This is still a linear model, but it's a bit smarter. It helps **prevent overfitting** by gently shrinking the importance of less impactful features. Given that real-world data can be a bit noisy and some features might be correlated (like maybe larger stores tend to be in Tier 1 cities), Ridge helps to make the model more stable and generalize better to new, unseen sales data

Random forest: It's an **ensemble method** that works by building many individual decision trees. Think of it like gathering a diverse committee of experts, where each expert (tree) looks at the problem slightly differently.

XGBoost: This is often a go-to for its speed, performance, and ability to handle various data types. It's highly optimized and includes built-in regularization to further combat overfitting.

Catboost: It's **specifically designed to handle categorical features natively and efficiently**. Unlike other models where I'd have to pre-process categories (e.g., one-hot encode)

Challenges and solutions: Challenges is like low accuracy with base line models and then performed hyper parameter tuning for the random forest, xgboost and Gradient boosting for improving accuracy but these models were not able to provide good satisfactory results. So tried with catboost using categorical features. Major challenge is to identify the right set of model