**OPTIMIZING PRICE FOR BOSCH PROJECT PLAN**

**NAME: RAFIAT OLAYINKA AMUSAH**

**COURSE: BUSINESS ANALYTICS - BAN 6800**

**LEARNER ID: 141383**

**PROJECT PLAN**

**DEFINE THE PROBLEM STATEMENT**

**Background:** Bosch Corporation, a global manufacturer leader, has been grappling with optimizing its price strategy for products. Bosch desires to leverage data-driven intelligence for the optimization of its price decision making process, increase customer satisfaction, and maximize profitability. With the application of different business analytics models, Bosch has not been realizing benefits as expected, and this has been leading to inefficiency and an increase in cost.

**Objective:** Create effective prices and consumer tastes. The goal is to achieve the optimal pricing strategy that will bring the maximum revenue, considering customer satisfaction and competitive pricing within the market.

**DATA COLLECTION AND PREPARATION**

**Data Source:** Kaggle- (retail\_price.csv) **Talib, M.**

Talib, Mohammad. “Retail Sales Dataset.” *Kaggle*, 22 Aug. 2023, www.kaggle.com/datasets/mohammadtalib786/retail-sales-dataset?resource=download

**Data Features:** Columns:

* **product\_id**: Each product has a unique product ID
* **product\_category\_name**: Product category name (e.g., bed\_bath\_table, garden\_tools).
* **month\_year**: Date of entry in month and year.
* **qty**: Number of products sold.
* **total\_price**: Total price of products sold.
* **freight\_price**: Freight price of the product.
* **unit\_price**: Each unit contains the unit price for it.
* **product\_name\_length**: Product name length.
* **product\_description\_length**: Length of product description.
* **product\_photos\_qty**: Number of product photos.
* **product\_weight\_g**: Product weight in grams.
* **product\_score**: Customer rating of the product.
* **customers**: Buyers of the product.
* **weekday, weekend, holiday**: Picked up on a weekday, weekend, or holiday.
* **month, year**: Month and year of the sale.
* **s, volume, comp\_1, ps1, fp1, comp\_2, ps2, fp2, comp\_3, ps3, fp3, lag\_price**: Additional metrics and comparisons.

**Data Preprocessing:**

* Loading Data: Loaded dataset using pandas.
* Handling missing values and cleaning data: SimpleImputer is used for imputing missing values in the freight\_price and products\_weight\_g columns with the mean.
* Encode categorical variables: OneHotENcoder is used for encode the product\_categroy\_name column.
* Normalize or standardize numerical features: StandardScaler is used for normarlizing numerical features.
* Splitting the data: Train set and test set are split based on train\_test\_split.
* Preprocessing pipeline usage: Test data as well as train data both consist of preprocessing.

**EXPLORATORY DATA ANALYSIS**

1. Analyze sales trends, customer ratings, and competitor pricing.
2. Identify patterns and correlations.

**Insights:**

* **Product Categories**: There are varieties of products in the dataset such as bed\_bath\_table, garden\_tools, consoles\_games, health\_beauty, and more.
* **Sales Trends**: The dataset includes months and years, used for analysis of sales trends over time.
* **Customer Ratings**: There are customer ratings for products, and customer satisfaction can be obtained from customer ratings.
* **Freight Costs**: Shipping costs are highly variable between products and have the potential to impact total cost.

**Demand Elasticity Analysis**:

* Make estimates of price elasticity of demand in order to judge how proce affects quantity sold.

**Competitor Analysis**:

* Plot Bosch’s prices against competitor prices to see competitive strength or weakness.

**MODEL DEVELOPMENT AND TESTING**

**Model Development**:

* The chosen machine learning algorithm for this project is XGBoost for predicting the price

**Optimization**:

* To find the price that maximizes revenue and profit, optimization techniques will be used

**Validation and Testing**:

* Test the model with historical data.
* Test the prediction of the model and adjust accordingly.

**MODEL DEPLOYMENT**

**Deployment**:

* Execute the model on AWS to get real-time price optimization.

**Launch an EC2 Instance**:

* Choose an Amazon Machine Image (AMI).
* Configure instance details, add storage, and configure security groups.

**Deploy the Flask App**:

* SSH into the EC2 instance.
* Install packages (Python, Flask, etc.).
* Deploy the Flask app to the EC2 instance.
* Run the Flask app.