During domain analysis, we noticed that some columns were categorical while others were numerical, leading to imbalanced data. This made data cleaning and feature engineering challenging. When choosing the right machine learning model for our business case, we realized it was a classification problem.

However, the dataset was vague and imbalanced, making it tough to get reliable results. After evaluating the model, we found the accuracy wasn’t great. So, we tried hyperparameter tuning, which finally gave us better accuracy.

On what basis you have created the model.

**Model Design Analysis for Liver Disease Prediction**

**1. Problem Definition**

The objective is to predict the presence of liver disease using demographic and biochemical data. The target variable is **Target** with two classes:

* **1 = Liver Disease**
* **2 = No Liver Disease**

**2. Data Exploration and Understanding**

* **Rows:** 583
* **Columns:** 11 (10 features + 1 target)
* **Data Types:** Numerical and Categorical
* **Class Distribution:** Imbalanced (likely more cases of one class than the other)

**3. Feature Analysis**

* **Numerical Features:**
  + Age, Total\_Bilirubin, Direct\_Bilirubin, Alkaline\_Phosphotase, Alanine\_Aminotransferase, Aspartate\_Aminotransferase, Total\_Proteins, Albumin, Albumin\_and\_Globulin\_Ratio
  + **Analysis Performed:**
    - Descriptive statistics (mean, median, standard deviation)
    - Distribution plots (histograms, box plots) to check for outliers and skewness
* **Categorical Feature:**
  + Gender (Male/Female)
  + **Analysis Performed:**
    - Frequency distribution
    - Impact on target variable using cross-tabulation

**4. Data Preprocessing**

* **Handling Missing Values:** Checked for null values and decided on imputation or removal.
* **Outlier Detection and Treatment:** Used box plots and z-scores to identify and handle outliers.
* **Categorical Encoding:**
  + Gender was encoded using One-Hot Encoding.
* **Feature Scaling:**
  + Standardization applied to numerical features for uniform scale.

**5. Addressing Imbalanced Data**

* The target classes were imbalanced, affecting model performance.
* **Techniques Used:**
  + **Oversampling:** SMOTE (Synthetic Minority Over-sampling Technique)
  + **Undersampling:** Random undersampling of majority class

**6. Feature Engineering**

* Created interaction features (e.g., combining bilirubin levels).
* Derived new features such as age groups to enhance model performance.

**7. Model Selection**

After analyzing the problem, the classification models considered were:

* **Logistic Regression** – As a baseline model
* **Random Forest** – For handling non-linear relationships and feature importance
* **XGBoost** – For its efficiency with imbalanced data
* **Support Vector Machine (SVM)** – To explore boundary-based classification

**8. Model Evaluation Metrics**

* **Accuracy** – Initial measure of overall performance
* **Precision, Recall, and F1-Score** – To evaluate performance on imbalanced classes
* **ROC-AUC Score** – To check model discrimination capability

**9. Model Performance and Tuning**

* Initial models showed poor accuracy due to imbalanced data.
* **Hyperparameter Tuning:**
  + Used GridSearchCV and RandomizedSearchCV to optimize model parameters.
  + Tuned parameters like learning rate, depth of trees, and number of estimators.
* **Final Model Selection:**
  + **XGBoost** gave the best performance with balanced accuracy and improved precision and recall.

**10. Final Decision and Justification**

* XGBoost was selected as the final model due to:
  + Better handling of imbalanced data
  + Higher accuracy and AUC-ROC score compared to other models
  + Interpretability of feature importance for medical diagnosis