

Model report

**Team Members :**

**vishnu patil - 24-996093**

**Yash Somkuwar - 24-111757**

**Dev Mahatme - 24-675666**

Team : Nada but just

**Team ID :T-0818232**

**1. Exploratory Data Analysis (EDA)**

**1.1 Data Inspection**

* **Inspect the dataset and determine the data types of all features (numerical, categorical).**  
  We began by loading the dataset and inspecting the data types of each feature. The dataset contains both numerical (e.g., Relative Velocity, Miss Distance) and categorical (e.g., Orbit Class) features.
* **Calculate and analyze basic statistics for each numerical feature, including range, mean, median, standard deviation, and quartiles.**  
  Basic statistics were calculated using the .describe() method from pandas, revealing insights such as:
  + Mean Miss Distance: x.xx
  + Median Relative Velocity: x.xx
  + Standard Deviation of Orbital Period: x.xx
  + Range of Hazardous column: 0 to 1
* **Identify features that have missing values.**  
  The dataset was checked for missing values using the isnull().sum() method, revealing that several features contained missing values, including Orbital Period and Miss Distance.
* **Identify the numerical and categorical features of the dataset to use for further analysis.**  
  Numerical features included Relative Velocity, Miss Distance, and Orbital Period, while categorical features included Orbit Class and Hazardous.
* **Use imputation to fill the null values in the dataset. How is this process different for numerical and categorical columns?**  
  Numerical features were filled with the mean, while categorical features were filled with the mode using the Simple Imputer from Scikit-learn. This approach ensures that numerical distributions remain unchanged while maintaining categorical feature integrity.

**1.2 Statistical Inference (6 points)**

* **Plot the distribution of numerical features to assess the skewness of the data.**  
  Histograms were plotted for numerical features to check for skewness. The Relative Velocity feature showed positive skewness, indicating a need for normalization.
* **Identify potential outliers in the numerical columns using any statistical technique.**  
  Box plots were generated to identify outliers. Several outliers were observed in the Miss Distance and Relative Velocity columns.
* **Explore the relationship between different features using scatter plots or correlation matrices.**  
  A correlation matrix was created, showing strong correlations between Relative Velocity and Miss Distance. Scatter plots were used to visually confirm these relationships.

**1.3 Visualization**

* **Create a pairplot using Seaborn to visualize relationships between multiple numerical features simultaneously.**  
  A pairplot was generated, illustrating the relationships between Relative Velocity, Miss Distance, and Orbital Period.
* **What do you infer from these plots?**  
  The diagonal plots represent the distribution of each feature, while off-diagonal plots show relationships between pairs of features. For instance, the scatter plots revealed that higher Relative Velocity correlates with lower Miss Distance.

**1.4 Tackling Class Imbalance**

* **Is there a classification bias (class imbalance) in this dataset?**  
  Yes, the dataset exhibited class imbalance, with significantly more non-hazardous asteroids than hazardous ones.
* **How would you tackle it?**  
  Techniques like oversampling the minority class (SMOTE) or undersampling the majority class were proposed to balance the classes.
* **Discuss the implication of class imbalance on model performance.**  
  Class imbalance can lead to biased predictions towards the majority class, decreasing the model’s ability to detect hazardous asteroids effectively.

**2. Numerical Interpretation and Mathematical Analysis**

**2.1 Feature Engineering**

* **Combine the approach\_date, month, and year features into a single feature representing the day of the year.**  
  A new feature day\_of\_year was created by converting the approach date into a datetime format.
* **Calculate the ratio of Miss Distance vs. Semi-major axis. Create a 'Time Until Approach' feature.**  
  The Miss Distance ratio and Time Until Approach were calculated based on the difference between Epoch Date Close Approach and the current date.
* **Calculate the eccentricity of the orbit, average orbital velocity, and orbital period using Kepler’s Law.**  
  Using Kepler's laws, we derived the values for eccentricity, average velocity, and orbital period.
* **Calculate the heliocentric distance, escape velocity, and specific orbital energy.**  
  These calculations were performed using the relevant orbital parameters.
* **Calculate the Specific Angular Momentum using the formula: h=sqrt(GMa(1−e²)).**  
  The specific angular momentum was computed for all asteroids.
* **Calculate the velocity at Perihelion and Aphelion.**  
  The velocities were derived using relevant orbital equations.
* **Average the Miss distance of various categories and find the closest approach distance.**  
  The average Miss Distance was calculated across different orbit classes.
* **Calculate Synodic Period and Mean Motion using the orbital period.**  
  The synodic period and mean motion were calculated using the derived orbital period.

**2.2 Additional Features**

* **Create additional features as per your understanding of the problem for improving accuracy.**  
  Additional features included:
  + Orbital Stability Index
  + Impact Probability Score
  + Historical Close Approaches

**3. Handling Binned Values**

* **Modify the binned features that have an ordinal relationship.**  
  Binned features were encoded into numerical values, such as very slow = 0, slow = 1, fast = 2, very fast = 3.
* **One-hot encode the binned features whose relationship is not strictly ordinal.**  
  Non-ordinal binned features were one-hot encoded to facilitate their use in modeling.

**4. Hazardous Classification**

* **Build a robust and efficient classifier to classify asteroids as Hazardous (1) or Not Hazardous (0).**  
  A LightGBM classifier was built, leveraging its efficiency for large datasets.
* **Implement K-Fold Cross Validation for training.**  
  K-Fold cross-validation (K=2 to 10) was implemented, and loss and accuracy were plotted versus epochs.
* **Optimize all the hyperparameters used in the classifier by selecting an appropriate optimization method.**  
  Hyperparameter optimization was performed using Grid Search, resulting in improved model performance.
* **Plot the ROC curve and Confusion Matrix to quantify the performance of your classifier.**  
  The ROC curve and confusion matrix indicated high sensitivity and specificity of the model.
* **Use SHAP Values, Permutation Importance, or Partial Dependence Plots.**  
  SHAP values were utilized to analyze feature importance, revealing that Relative Velocity and Miss Distance were the most significant predictors of hazard classification.

**5. Anomaly Detection**

* **Perform anomaly detection using an inbuilt library of your choice.**  
  Anomaly detection was performed using Isolation Forest, successfully detecting anomalies in the dataset.
* **Writing your own anomaly detection algorithm.**  
  A custom algorithm using Z-score and IQR methods was implemented to identify anomalies, yielding similar results.
* **Store the results as a new column in the dataset.**  
  Anomaly detection results were stored, and the number of anomalies detected by each method was printed, confirming the effectiveness of both techniques.