AZDC'26

Cosmic Collision Analysis

Kaggle Competition 2026

Azad Hall of Residence
Indian Institute of Technology Kharagpur

Contents

1 Mission Overview **1**
- 1.1 Cosmic Context . . . 1
- 1.2 The Challenge . . . 1
- 1.3 Primary Objective . . . 1
- 1.4 Secondary Objectives . . . 1

2 Evaluation **1**

3 Dataset Description **2**
- 3.1 Data Dictionary . . . 2
- 3.2 Data Quality Notes . . . 3

4 Detailed Task Breakdown **3**
- 4.1 Phase 1: Data Exploration, Understanding and Cleaning . . . 3
  - 4.1.1 1.1 Initial Data Inspection . . . 3
  - 4.1.2 1.2 Statistical Analysis . . . 4
  - 4.1.3 1.3 Visualization . . . 4
  - 4.1.4 1.4 Class Imbalance Analysis . . . 4
- 4.2 Phase 2: Feature Engineering . . . 5
  - 4.2.1 2.1 Temporal Feature Engineering (4 points) . . . 5
  - 4.2.2 2.2 Orbital Mechanics to derive Features . . . 5
  - 4.2.3 2.3 Creative Feature Engineering . . . 6
- 4.3 Phase 3: Data Preprocessing before modeling . . . 6
  - 4.3.1 3.1 Handling Categorical Variables . . . 6
  - 4.3.2 3.2 Feature Scaling and Transformation . . . 6
- 4.4 Phase 4: Model Building and Optimization . . . 6
  - 4.4.1 4.1 Model Selection and Training . . . 6
  - 4.4.2 4.2 Hyperparameter Optimization . . . 6
  - 4.4.3 4.3 Model Evaluation . . . 6

5 Strategic Guidance Framework **7**
- 5.1 Recommended Workflow . . . 7
- 5.2 Critical Technical Hints . . . 7
- 5.3 Common Pitfalls to Avoid . . . 8

6 Timeline and Important Dates **8**

7 Rewards and Submission **8**
- 7.1 Competition Rewards . . . 8
- 7.2 Skill Development Opportunities . . . 8
- 7.3 Kaggle Submission . . . 8

8 Resources and Support **9**
- 8.1 Recommended Resources . . . 9
- 8.2 Support Channels . . . 9

1 Mission Overview

1.1 Cosmic Context

Near-Earth Objects (NEOs) represent both scientific opportunities and potential threats to our planet. Throughout Earth's history, asteroid impacts have shaped evolution, from creating the Moon to causing mass extinctions. Today, with advanced technology, we can predict and potentially mitigate these cosmic threats.

As junior researchers in the Azad Hall Space Research Division, you are tasked with analyzing historical asteroid approach data to develop a predictive system that identifies potentially hazardous asteroids. Your work could contribute to planetary defense systems that protect Earth from cosmic collisions.

1.2 The Challenge

You will analyze a dataset containing asteroid close-approach records. This dataset contains real observational data with all its imperfections: missing values, measurement errors, and complex interrelationships between features.

Your mission is to transform this raw data into actionable intelligence by building a classification system that can accurately determine whether an asteroid poses a hazard to Earth.

1.3 Primary Objective

Develop a machine learning model that classifies asteroids as either **hazardous** or **non-hazardous** based on their orbital characteristics, approach parameters, and physical properties.

1.4 Secondary Objectives

1. Conduct comprehensive exploratory data analysis (EDA)
2. Engineer meaningful features from raw data
3. Handle data quality issues creatively
4. Build and optimize a robust classification model
5. Interpret model results for scientific insight

2 Evaluation

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{1}$$

Where:

- **Precision:** Of all asteroids predicted as hazardous, what proportion are actually hazardous?

- **Recall:** Of all actually hazardous asteroids, what proportion did we correctly identify?

The F1-score balances these two metrics, which is crucial when the consequences of false positives and false negatives differ.

Kaggle competitions use a dual leaderboard system to provide feedback during model development while ensuring fair final evaluation. During the active phase of the competition, performance is shown on the public leaderboard, which is calculated using a small, representative subset of the test data (20% in our case). This allows participants to track progress and compare approaches in real time. However, relying too heavily on the public leaderboard can lead to overfitting, as models may unintentionally adapt to this limited subset. To address this, Kaggle evaluates the remaining hidden portion of the test data using a private leaderboard, which is revealed only after the competition ends and determines the final rankings.

Note: Participants may select up to two submissions for final evaluation, and the best-performing one on the private test set is used for ranking. Hence, the final competition ranking is determined solely by the private leaderboard F1 score.*.

3 Dataset Description

3.1 Data Dictionary

Feature Name	Description	Data Type
`Name`	Unique identifier for the asteroid	Numerical
`Epoch Date Close Approach`	Unix timestamp of closest approach to Earth	Numerical
`Relative Velocity km per sec`	Categorical velocity classification (Very Slow, Slow, Fast, Very Fast)	Categorical
`Relative Velocity km per hr`	Speed relative to Earth in km/h	Numerical
`Miles per hour`	Speed relative to Earth in mph	Numerical
`Miss Dist.(Astronomical)`	Closest distance in Astronomical Units (1 AU = Earth-Sun distance)	Numerical
`Miss Dist.(lunar)`	Closest distance in lunar distances (1 LD = Earth-Moon distance)	Numerical
`Miss Dist.(kilometers)`	Closest distance in kilometers	Numerical
`Miss Dist.(miles)`	Closest distance in miles	Numerical
`Jupiter Tisserand Invariant`	Parameter describing orbit relative to Jupiter (dynamical classification)	Numerical
`Epoch Osculation`	Reference date for orbital calculations (Julian date)	Numerical
`Semi Major Axis`	Average distance from the Sun (defines orbit size)	Numerical

Feature Name	Description	Data Type
`Asc Node Longitude`	Angle where orbit crosses ecliptic plane (degrees)	Numerical
`Perihelion Arg`	Argument of perihelion (angular position of closest approach to Sun)	Numerical
`Aphelion Dist`	Maximum distance from the Sun	Numerical
`Perihelion Time`	Time of closest approach to the Sun	Numerical
`Mean Anomaly`	Position along orbit at specific time (degrees)	Numerical
`Mean Motion`	Average angular speed (degrees/day)	Numerical
`approach_year`	Year of closest approach to Earth	Numerical
`approach_month`	Month of closest approach to Earth	Numerical
`approach_day`	Day of closest approach to Earth	Numerical
`Orbital Period`	Categorical classification of orbital period (Low, Medium, High)	Categorical
`Orbit Uncertainity`	Confidence in orbital calculations (Low, Medium, High)	Categorical
`Hazardous`	**TARGET:** Whether asteroid is potentially hazardous (0/1)	Numerical

3.2 Data Quality Notes

Common Pitfall

Critical Data Issues:

- Missing values distributed non-uniformly across features
- Some features have over 50% missing values
- Temporal data in multiple formats (Unix timestamp, separate year/month/-day)
- Redundant measurements (same distance in different units)
- Categorical variables with inconsistent encoding
- Potential measurement errors and outliers

4 Detailed Task Breakdown

4.1 Phase 1: Data Exploration, Understanding and Cleaning

This phase is the most important phase as You need to understand what the unerlying data wants you to tell. Following breakdown contains few insights on how one can move but its obviously not limited to it.

4.1.1 1.1 Initial Data Inspection

- Determine data types for all features (numerical vs categorical)

- Calculate basic statistics for numerical features (range, mean, median, std, quartiles)
- Identify features with missing values and quantify them
- Separate numerical and categorical features for further analysis
- Implement appropriate imputation strategies for different data types

Step-by-Step Guidance

Step-by-Step:

- Use Pandas to its full potential for this kind of analysis
- Remember: Different units (km, miles) for same quantity - pick one to avoid redundancy.

4.1.2 1.2 Statistical Analysis

- Analyze distributions using histograms and density plots
- Assess skewness and consider normalization/transformation
- Identify outliers using statistical methods (IQR, z-score)
- Explore feature relationships using correlation matrices

Step-by-Step Guidance

1. Plot histograms for all numerical features using.
2. Check skewness: if absolute skewness ¿ 0.5, consider log transformation before modeling.
3. Use boxplots/ Violin plots for very robust analysis.
4. Create correlation matrix. High correlation suggests redundancy - consider dropping one feature.

4.1.3 1.3 Visualization

- Create pairplots for key numerical features
- Interpret diagonal (distribution) vs off-diagonal (relationship) plots
- Identify clusters and patterns in the data

4.1.4 1.4 Class Imbalance Analysis

- Analyze distribution of target variable
- Discuss implications of class imbalance on model performance
- Propose strategies to handle imbalance if present

Step-by-Step Guidance

1. Check class distribution.
2. If imbalance is high, it's significant. Accuracy will be misleading.
3. Solutions: Oversample minority class (SMOTE), undersample majority class, or use class weights in model. (NOTE: EVERYTHING HAS ITS UPS AND DOWNS. USE it smartly)
4. Always use stratified k-fold cross-validation to maintain class ratios.

4.2 Phase 2: Feature Engineering

4.2.1 2.1 Temporal Feature Engineering (4 points)

- Convert Unix timestamps to datetime objects
- Create meaningful temporal features (day of year, season, etc.)
- Calculate "Time Until Approach" from a reference date

4.2.2 2.2 Orbital Mechanics to derive Features

- Derive orbital eccentricity: $e = \frac{Aphelion}{SemiMajor} - 1$
- Orbital period using Kepler's Third Law: $T = 2\pi\sqrt{\frac{a^3}{GM}}$
- Specific orbital energy: $\epsilon = -\frac{GM}{2a}$
- Specific angular momentum: $h = \sqrt{GMa(1-e^2)}$
- Velocity at perihelion: $v_p = \sqrt{\frac{GM}{a}\frac{1+e}{1-e}}$
- Velocity at aphelion: $v_a = \sqrt{\frac{GM}{a}\frac{1-e}{1+e}}$
- Synodic period: $\frac{1}{S} = \left|\frac{1}{T_{asteroid}} - \frac{1}{T_{Earth}}\right|$
- Mean motion: $n = \frac{2\pi}{T}$ (radians per day)

Step-by-Step Guidance

Constants to Use:

- $G = 6.67430 \times 10^{-11}\,\mathrm{m^3 kg^{-1} s^{-2}}$
- $M_\odot = 1.989 \times 10^{30}\,\mathrm{kg}$
- $AU = 1.496 \times 10^{11}\,\mathrm{m}$
- Earth orbital period = 365.25 days

Tips: This problem is not limited to above equations only, try to research and utilise other equations or work to form a more intricate solution.

4.2.3 2.3 Creative Feature Engineering

- Create at least 3 original features based on domain insight
- Justify each feature's relevance to hazard prediction
- Test feature importance

4.3 Phase 3: Data Preprocessing before modeling

4.3.1 3.1 Handling Categorical Variables

- Ordinal encoding for naturally ordered categories (velocity levels)
- One-hot encoding for nominal categories
- Justify encoding choices

4.3.2 3.2 Feature Scaling and Transformation

- Apply appropriate scaling (StandardScaler, RobustScaler, MinMaxScaler)
- Handle skewed distributions with transformations
- Address multicollinearity in redundant features

4.4 Phase 4: Model Building and Optimization

4.4.1 4.1 Model Selection and Training

- Implement at least 3 different classification algorithms
- Use k-fold cross-validation (k=2 to 10)
- Plot learning curves for different k values
- Compare model performance

4.4.2 4.2 Hyperparameter Optimization

- Use systematic hyperparameter tuning (GridSearchCV, RandomSearchCV, Bayesian)
- Prevent overfitting with regularization

4.4.3 4.3 Model Evaluation

- Generate ROC curves and calculate AUC
- Create confusion matrices with proper interpretation
- Use SHAP values or permutation importance for feature interpretation
- Identify most predictive features

5 Strategic Guidance Framework

5.1 Recommended Workflow

- Load data, understand structure, identify issues
- Handle missing values using domain knowledge
- Create basic visualizations, understand distributions
- Create orbital mechanics features
- Build baseline models
- Optimize hyperparameters
- Final model tuning and submission

5.2 Critical Technical Hints

Step-by-Step Guidance

Data Imputation Strategy:

- For orbital parameters: Use Kepler's laws and physical relationships
- For temporal features: Forward-fill within temporal groups
- For categorical variables: Use mode or create "missing" category
- Consider MICE (Multiple Imputation by Chained Equations) for complex cases

Step-by-Step Guidance

Feature Selection Approach:

- Remove redundant distance measurements (keep one unit)
- Use correlation analysis to identify collinear features
- Consider domain importance alongside statistical significance
- Test feature importance through permutation tests
- Try to utilize feature importance approaches present in existing libraries to decide final feature selection.

Step-by-Step Guidance

Modeling Tips:

- Start with simple models (Logistic Regression, Decision Tree) as baselines
- Use ensemble methods (Random Forest, Gradient Boosting) for final model

- Consider neural networks if you have sufficient data preprocessing
- Always validate with proper cross-validation techniques

5.3 Common Pitfalls to Avoid

Common Pitfall

Critical Mistakes:

- **Overlooking domain knowledge:** Physics laws constrain possible values
- **Misinterpreting metrics:** Accuracy is misleading with imbalanced data
- **Feature engineering without validation:** Always test if new features help

6 Timeline and Important Dates

Date	Milestone
Jan 30, 2026	Competition begins, dataset released
Feb 8 2026	Final submission (EOD) & leaderboard goes public
Feb 9/10 2026	Short Presentation for Top 5 teams

7 Rewards and Submission

7.1 Competition Rewards

Top 5 Teams: Special Treat :p and Certificate from Hall

7.2 Skill Development Opportunities

- **Presentation Session:** Top teams present their solution to Azad Hall Seniors
- **Public Speaking Practice:** Receive feedback on presentation skills
- **Networking:** Connect with seniors interested in similar domains
- **Portfolio Building:** Complete project for your resume/GitHub

7.3 Kaggle Submission

- Download the Kaggle submission template (submission_template.csv).
- Use your model to predict on the test set, then fill the template's `Hazardous` column by matching the asteroid names.
- Submit the filled CSV as-is.

- Template format:
  - Column 1: `Name` (asteroid identifiers)
  - Column 2: `Hazardous` (your predictions: 0/1)
- **Do NOT:**
  - Change column names
  - Add/remove columns
  - Reorder rows
  - Modify the `Name` values
- Simply: predict → map by name → submit the CSV unchanged.

Note: For further information, checkout competitions official page on Kaggle

8 Resources and Support

8.1 Recommended Resources

- **Machine Learning:** CampusX 100 Days ML/DL course on YT
- **Visualization:** Matplotlib and Seaborn should be ideally sufficient

8.2 Support Channels

- **WhatsApp Group:** Join the **link** sent in the group
- **Peer Help:** Form study groups with other participants

Remember: Every great data scientist started with their first messy dataset!
Don't get discouraged...ask questions, collaborate, and learn!