ASTEROID HAZARD PREDICTION SYSTEM USING MACHINE LEARNING

Snehil Raj

B.Tech (Computer Engineering)

GitHub Repository: https://github.com/Snehil7903/asteroid-hazard-prediction

Live Application: https://asteroid-hazard-prediction-by-snehil.streamlit.app/

Abstract

Near-Earth Objects (NEOs), particularly asteroids, pose a potential threat to Earth due to close-approach events. Early identification of potentially hazardous asteroids is critical for planetary defense and risk mitigation. This project presents an end-to-end machine learning–based system for predicting asteroid hazard levels using real-world NASA Near-Earth Object (NEO) data. Multiple supervised learning models—including Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Decision Tree classifiers—were trained and evaluated using ROC-AUC to address class imbalance. Additionally, unsupervised learning techniques such as K-Means clustering combined with Principal Component Analysis (PCA) were applied to discover natural groupings within asteroid data. The trained models were deployed through an interactive Streamlit web application, enabling real-time hazard prediction with confidence scores. The system demonstrates the practical application of machine learning in astronomical risk assessment and public-facing scientific tools.

Keywords

Asteroid Hazard Prediction, NASA NEO, Machine Learning, ROC-AUC, Classification, Clustering, Streamlit, PCA

1. Introduction

Asteroids that approach Earth closely can cause significant damage in the event of an impact. Although most Near-Earth Objects (NEOs) pass safely, some meet specific criteria that make them potentially hazardous. NASA monitors these objects continuously, but analyzing large-scale asteroid data manually is inefficient.

Machine learning techniques provide an automated and scalable approach for identifying hazardous asteroids by learning patterns from historical data. This project focuses on building an end-to-end system that not only predicts asteroid hazard levels but also provides interpretability, comparison across models, and real-time user interaction through a deployed web application.

2. Problem Statement

The goal of this project is to develop a machine learning system capable of predicting whether an asteroid is potentially hazardous based on its physical and orbital characteristics. The system must handle imbalanced data, provide meaningful risk estimates, and allow real-time predictions through an interactive interface.

3. Dataset Description

The dataset used in this project is based on publicly available **NASA Near-Earth Object (NEO)** records.

Features Used

- **Absolute Magnitude (H):** Indicates asteroid brightness and indirectly its size.
- **Estimated Diameter (Min & Max):** Estimated size range of the asteroid in meters.
- **Relative Velocity (km/s):** Speed of the asteroid relative to Earth.
- **Miss Distance (AU):** Closest approach distance between the asteroid and Earth.
- **Hazardous:** Binary target variable indicating whether the asteroid is potentially hazardous.

NASA Hazard Criteria

According to NASA:

- **Miss Distance ≤ 0.05 AU**
- **Absolute Magnitude (H) < 22**

Asteroids satisfying both conditions are classified as potentially hazardous.

4. Data Preprocessing

The following preprocessing steps were applied:

- Removal of missing and inconsistent values
- Feature selection based on relevance to hazard prediction
- Standardization using Standard Scaler for scale-sensitive models
- Train-test split to ensure unbiased evaluation

5. Exploratory Data Analysis (EDA)

EDA revealed:

- Significant class imbalance, with non-hazardous asteroids dominating the dataset
- Miss distance as the most influential feature
- Larger asteroids with closer approaches are more likely hazardous

These insights guided feature selection and model evaluation choices.

6. Machine Learning Models

6.1 Supervised Learning Models

The following classifiers were implemented and compared:

Logistic Regression

A probabilistic model that provides smooth confidence estimates. It is well-suited for risk estimation and was used as the primary model for probability output.

K-Nearest Neighbors (KNN)

A distance-based classifier that predicts hazard based on similarity to known asteroid cases.

Support Vector Machine (SVM)

A margin-based classifier that focuses on separating safe and hazardous asteroids. It produces conservative predictions, especially in imbalanced datasets.

Decision Tree

A rule-based model that provides interpretable decision logic but produces hard classifications with poorly calibrated probabilities.

6.2 Evaluation Metric

Due to class imbalance, **ROC-AUC (Receiver Operating Characteristic – Area Under the Curve)** was used instead of accuracy.

ROC-AUC measures the probability that a model ranks a hazardous asteroid higher than a non-hazardous one, making it ideal for imbalanced binary classification problems.

7. Unsupervised Learning

7.1 K-Means Clustering

K-Means clustering was applied to identify natural groupings of asteroids based on physical and orbital features.

7.2 Principal Component Analysis (PCA)

PCA was used to reduce feature dimensions to two components, enabling visualization of clusters.

Clustering was used for exploratory analysis and insight generation, not for direct hazard prediction.

8. System Architecture

The system follows a modular pipeline:

1. Data ingestion (NASA NEO dataset)
2. Data preprocessing and scaling

3. Model training and evaluation
4. Model serialization
5. Streamlit web application
6. Cloud deployment

9. Web Application Deployment

An interactive web application was developed using **Streamlit**.

Features

- User input for asteroid parameters
- Model selection
- Real-time hazard prediction
- Confidence visualization
- Educational explanations of parameters and models

Live Application

https://asteroid-hazard-prediction-by-snehil.streamlit.app/

GitHub Repository

https://github.com/Snehil7903/asteroid-hazard-prediction

The GitHub repository provides complete source code for transparency and reproducibility.

10. Results and Discussion

- Logistic Regression provided the most reliable probability estimates.
- SVM and Decision Tree models were conservative due to class imbalance.
- Miss distance dominated hazard prediction.
- Clustering revealed distinct asteroid groupings but did not directly predict hazard.

11. Limitations

- Limited to historical data
- No real-time NASA API integration
- No orbital simulation or future trajectory modeling
- Probability calibration for tree-based models can be improved

12. Future Scope

- Integration with live NASA APIs
- Ensemble models such as Random Forest and XGBoost
- Real-time asteroid alert systems
- Advanced probability calibration
- Further mobile UI optimization

13. Conclusion

This project successfully demonstrates the application of machine learning for asteroid hazard prediction using real NASA data. By combining supervised classification, unsupervised clustering, and cloud deployment, the system provides both predictive capability and educational value. The deployed web application highlights the effectiveness of machine learning in scientific risk assessment and public awareness.

14. Tools and Technologies Used

- **Programming Language:** Python
- **Libraries:** Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn
- **Machine Learning:** Classification, Clustering, PCA
- **Deployment:** Streamlit, Streamlit Cloud
- **Version Control:** Git, GitHub

15. References

1. NASA Near-Earth Object Program
2. Scikit-learn Documentation
3. Streamlit Documentation
4. GitHub Repository – Asteroid Hazard Prediction System

   https://github.com/Snehil7903/asteroid-hazard-prediction