Machine Learning for Asteroid Classification

Rachel Hausmann

June 11, 2021

Abstract

Asteroids are small, rocky, debris left over from the formation of our solar system around 4.6 billion years ago. There are currently over 822,000 known asteroids but scientists estimate there are more than a million. Asteroids are classified into clusters around our solar system. Occasionally, asteroids' orbital paths are influenced by the gravitational tug of planets, which cause their paths to alter. Because of the sheer number of asteroids and technical challenges we do not always have all of the observational data on each asteroid that we want. It is beneficial to explore the most important features needed to classify asteroid orbital paths in order to increase efficiency while reducing data dependence. Based on the random tree mean decrease in impurity the three most important features are the semi-major axis, the mean motion degrees per day, and the orbital period in days. Followed by an equal importance of the Perihelion distance and the minimum orbital intersect distance to Earth.

Design

Occasionally, asteroids' orbital paths are influenced by the gravitational tug of planets, which cause their paths to alter. Scientists and engineers are developing plans for warning systems and diversion tactics, just in case an asteroid should ever be found in an orbit that could endanger our planet. One way asteroids are classified is by the orbital cluster it currently resides in. Creating a model that joins that may warn scientists if an asteroid changes classes may tell us if that asteroid's orbital pattern has changed. This project focused on exploring important features of modeling asteroid locations in our solar system.

The classification metric chosen to measure the success of my model is accuracy. Accuracy was chosen for this model because all of the asteroid orbital classes are equally important. We want to accurately classify the asteroids across all classes based on their orbital geometry.

Data

NASA's "known" asteroid dataset was used. The original dataset contained roughly 822,000 observations. The data was cleaned and reduced to 140,000 observations filtered by the presence of a diameter measurement of an asteroid.

All 11 asteroid orbital classes included in the dataset were included in the model. The orbital classes were represented by 3 letter acronyms. There was significant class imbalance in the dataset which was compensated by a stratified k-fold method of data organization.

The features in the model:
Semi_major_axis_AU
Eccentricity
Inclination
Longitude_of_asc_node
Argument_of_perhelion
Perihelion_distance_AU
Aphelion_distance
Absolute_Mag_Parameter
Diameter
Earth_Min_Orbital_Intersect_Distance_AU
Mean_Motion_degrees_per_day
Orbital_Period_Days
Mean_Anomaly_Degrees

Algorithms
Asteroid data was cleaned in a jupyter notebook using pandas. Observations containing any null/NaN values in any of the feature columns were removed from the dataset. Any observations with NaN/Null in the diameter column were removed. This left 140,000 observations for modeling.

Upon running a baseline model for KNN, Log Reg, Decision Tree and Random Forest I was given the best classification metrics with random forest:

- Precision: 1.000
- Recall: 1.000
- F1: 1.000
- Accuracy: 1.000

I was suspicious of the 1.000 scores as overfitting is a serious consideration. Random forest was selected as the model based on the high classification metrics, the knowledge that the model is multi-class, and my dataset was imbalanced. I then conducted 10 stratified k-fold splits on the training data to account for the class imbalance of my dataset. Rerunning the random forest model on the stratified split resulted in a 0.9998 accuracy measure. I then created a feature importance chart based on the mean decrease of impurity. The feature importance chart and confusion matrix inform me that the model is performing as well as it is due to the high importance of features included in the dataset. The model is not expected to be overfitting as the confusion matrix shows only 3 observations were misclassified.

Tools

- Python pandas for EDA and data cleaning
- Scikit-learn for data modeling
- Seaborn and matplotlib for data visualization