Nearest Earth Objects – A traditional and a Neural model

Valentina Ravest

1. Introduction
   1. Description of the dataset

This dataset is a compilation of the list provided by NASA that condenses all the objects that are at a distance from the earth, their characteristics and whether they are potentially dangerous.

In this project it was decided to develop a classifier that could predict whether an object orbiting near the earth was dangerous or not. This classification task is binary because there are only two possible labels, dangerous or not, which can be translated to 1 or 0.

1. Dataset exploration

Before starting with the implementation of the models, an inspection was carried out to verify that the data were optimal for the predictions. First, the datatype of each feature was checked. Then we checked for missing values, since they were not present in the dataset. Then some graphs were made to account for the distribution of the values, showing many outliers.

Since hazardous is the column to be predicted, additional tests were performed to account for the distribution. We found 81996 classified as non hazardous and only 8840 hazardous, which shows that the dataset is unbalanced, having only 10% of hazardous values.

* 1. Imbalanced data

To deal with the imbalance of the data, a downsampling was performed, reducing the number of False labels to the same number of True labels, with this, the dataset contains a total of 17680 rows with 8840 True and 8840 False. And, to deal with the difference between the dataset values we used the MinMaxScaler

1. Traditional model
   1. Random forest

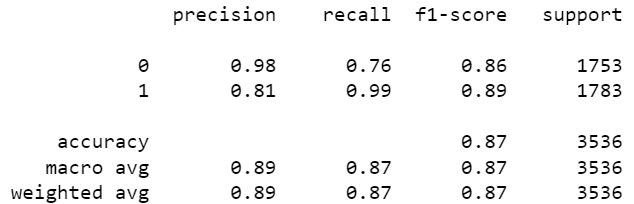
A pipeline was created using the random forest model. This model fits decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

A GridSearch was performed to identify the best parameters for the model. The best parameters were:

RandomForestClassifier(max\_depth=10, min\_samples\_leaf=5, min\_samples\_split=6)

* 1. Results

After applying the gridsearch the model obtained a macro f1 score accuracy of 0.87, increasing by 1 point.



1. Neural model
   1. Deep keras sequential

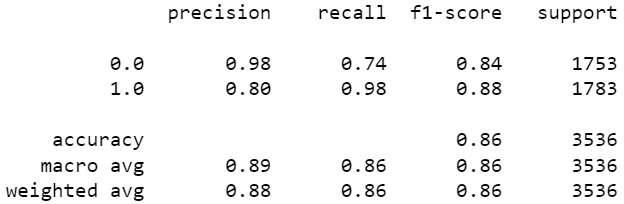
The deep learning keras sequential model was used. This model allows the construction of several layers of neural networks having a desired input and output. The data follow a flow from start to finish in which they are processed through the different layers.

* 1. Parameters

Because the problem is a binary qualsification problem, two layers with a softmaz activation function and the final layer with the sigmoid function were used, the optimizer selected was Adam, the loss function BinaryCrossEntropy and the measure binary accuracy.

* 1. Results

The ressult of the model is 0.86 macro f1 accuracy.



1. Comparing models

The traditional model obtained better results, but required more time to perform the gridsearch. In this case, the neural model, being extremely simple, obtained slightly lower results but in a more efficient way.