AS4501 - Homework 2

A new generation of survey telescopes is monitoring the sky in a systematic fashion thanks to a combination of large apertures and large digital detectors, notably the Zwicky Transient Facility (ZTF), which is already operating from Mount Palomar, USA, and the Vera C. Rubin Observatory and its Legacy Survey of Space and Time (LSST), which will start its operations in Chile in 2024. These telescopes allow us to study the variable sky, including new or variables objects, that are reported in real-time in streams of astronomical alerts. In order to process these alerts a new generation of astronomical alert brokers has been created. In fact, seven brokers have been officially selected as LSST Comunity Brokers. One of them is located in Chile: the ALeRCE broker (Automatic Learning for the Rapid Classification of Events).

In this project we ask you to implement a classifier for the ALeRCE broker. In particular, we want you to use pandas and its methods more effectively and to use sklearn.

1. Data curation using pandas:

   In this section we will read the data excercising pandas. We recommend that you do the 10 minutes to pandas tutorial if you haven't done so. When using pandas you should try to avoid for loops, many calculations can be done in one line. In fact, the following tasks should be done with one line when indicated.

   a. Read the labels dataframe dfcrossmatches_small.pickle (one line) and display it (one line). Note that the dataframe has already been indexed by the object's identifier (oid). Extract only the 'classALeRCE' column (one line). Do a bar plot of the number of classes using the value_counts function and pandas plot.bar (one line).
   b. Read the features dataset features_small.pickle (one line) and display it (one line) . You will see that the dataframe has already been indexed by oid and that there are many features for the same object (oid) and band (fid). Use the pandas pivot function so that the new column names are given by the values in the 'name' and 'fid' columns and the values are given by the 'value' column (one line). Display it (one line). This will create multiindex columns, play with the columns and discover how to recover one element in the dataframe.
   c. Display the median of every column using the pandas median function (one line). Replace all nan values in the features dataframe by the median value in the column using the fillna function (one line) and display it (one line).
   d. Concatenate the two previous dataframes (labels and features) using the pandas concat function, noting that you are concatenating columns, i.e. axis=1 (one line) and display the result (one line). Note that there are some oids which did not have features. Drop all rows with nan values using the pandas dropna function with inplace=True (one line). Display the result (one line).
   e. Do a bar plot of the number of classes using the value_counts function and the pandas plot.bar function (one line).

f. Group the classes “EB/EW” and “EA” as “EB” (eclipsing binaries) using the replace function (one line). Drop the classes “NLAGN”, “NLQSO”, “TDE” and “ZZ” using the pandas loc and isin functions (one line). Display the resulting dataframe. Now you have a clean dataset to build the design matrix X and the labels vector y!

g. Build the design matrix X and the labels vector y.

2. Visualize the data

a. Select 10 random examples from each class and use the ALeRCE client to plot the light curves of these objects. You can learn about the ALeRCE client in here https://alerce.readthedocs.io/en/latest/.

b. Select 30 random examples from the entire dataset and plot their light curves using the ALeRCE client. Are you able to visually classify them?

c. Use some dimensionality reduction technique to plot the design matrix in two dimensions (e.g. UMAP). Color code elements based on their class. Do you see any structure?

3. Building the training and test sets and train the model.

a. Separate the training and test sets using 30% of the sample for the test with sklearn.

b. Choose two different methods seen in classes and train them using the sklearn API. Use stratified k-fold cross-validation to find relevant hyperparameters

4. Model metrics

a. Plot the ROC curve for each classifier

b. Show the confusion matrix for each classifier. Do you see a problem with your classifier?

c. Compute the micro and macro averaged accuracy and f1 scores for each classifier.

5. Fixing the data Imbalance

a. Choose one of the two models and try at least two methods to fix the imbalanced nature of the training set. Train the models and plot the confusion matrix in each case. Do you see any improvement?

Fecha de entrega: 9 de Mayo
Formato: jupyter notebook

PD: se recomienda utilizar pandas, ver
https://pandas.pydata.org/pandas-docs/stable/getting_started/10min.html