Astronomy & Astrophysics manuscript no. 0head_ML_3FGL
January 24, 2020

Machine learning classification of unassociated *Fermi* LAT sources

A. Bhat [⋆1] and D. Malyshev [⋆⋆1]

Erlangen Centre for Astroparticle Physics, Erwin-Rommel-Str. 1, Erlangen, Germany

Received September 15, 1996; accepted March 16, 1997

ABSTRACT

Context. Classification of sources is one of the most important tasks in astronomy. Sources detected in one wavelength band, e.g., in gamma rays, may have several possible associations in other wavebands or there may be no plausible association candidates.
Aims. In this work, we take unassociated sources in the third *Fermi*-LAT point source catalog (3FGL) and suggest associations to known classes of gamma-ray sources using machine learning methods trained on associated sources in the 3FGL.
Methods. We use several machine learning methods to separate *Fermi*-LAT sources into two major classes: pulsars and active galactic nuclei (AGNs). We evaluate the dependence of results on meta-parameters of the ML methods, such as the depth of the tree in tree-based classification methods and the number of layers in neural networks. We test the performance of the methods with a test sample drawn from the associated sources in 3FGL. We compare the predictions with the preliminary forth *Fermi*-LAT catalog (4FGL).
Results. Summary of results

Key words. Methods: statistical – Catalogs

Contents

⋆ e-mail: aakash.bhat@fau.de
⋆⋆ e-mail: dmitry.mayshev@fau.de

1. Introduction

Catalogs of gamma-ray sources such as 3FGL or 4FGL contain many sources without associations, e.g., X for 3FGL and Y for 4FGL. In order for a source to be associated with a known source, such as a blazar or a pulsar, it has to pass relatively strict selection criteria, which ensure that the rate of errors in associations is low. However, in some problems the desired property is completeness rather than purity. In other words, one may need to select as many members of a particular class as it is possible, which usually comes at the expense of higher rate of false associations. For example, this situation can arise if we look for something new or unusual, in this case one would like to pay attention to all possible members of the class, even though there may be many non-members added to this list. In this case it is useful to have a softer selection criterion, such as a probability to belong to a certain class, rather than a discrete classification. Then one has an option to select a pure subsample with very high probability to belong to the class, or a larger subsample where the probability to miss a source is low but for some of the sources the probability to belong to the class can be also not high, e.g., even less than 50%.

Discrete or probabilistic classification of unassociated gamma-ray sources can be achieved with machine learning (ML) algorithms. The basic idea is that the algorithms are trained to classify sources based on their characteristics, such as the spectral index and position on the sky, for a set of sources with known classifications. Then the application of the classification algorithm for a sources with unknown classification allows one to make a prediction to which class it belongs. The main goal of this paper is to apply several ML algorithms for the classification of sources in 3FGL and 4FGL catalogs. In particular we will use logistic regression, decision trees, random forest, neural networks algorithms. We will

1. Train the classifies on associated sources in the 3FGL catalog in Section 2;
2. Make prediction for unassociated sources in the 3FGL catalog and compare with the associations in the newer 4FGL catalog in Section 3;
3. Retrain the classifies for the 4FGL catalog and make predictions for the unassociated sources in 4FGL in Section 4.

Discussion of literature and what is new in this paper.

2. Methods

Our methodology for classification was dependent on two things: The data that we had, which needed to be cleaned and the algorithms that we needed to apply. For this we decided on using the 3rd catalog of F-LAT (3FGL from hereon) for initial training and testing, the 4th catalog (FL8Y from hereon) for further testing and predictions, and machine learning algorithms like Random Forests, Logistic Regression, Decision Trees, and Neural Networks. All of the machine learning algorithms were taken from the python module sklearn, including Neural Networks. A neural network using Keras was also attempted; however, due to the classification being on only two classes, we discarded it in favour of the sklearn algorithm which was much faster.

Our data was similar to that used by Parkinson et. al. We cleaned the 3FGL catalog to have sources which were both associated and unassociated but with no missing values. We then used the associated sources which were classified as either AGNs (with multiplpe labels) or Pulsars, to get a list of 1905 sources. The rest of the sources without problematic values were then used as unassociated sources, which we used later on for testing and prediction. The FL8Y presented us with another way of testing the accuracy of our methods. We predicted the classifications of unasssociated sources in the 3FGL and used the FL8Y to check how many of these unassociated sources, which now had associations in the FL8Y, actually had the right prediction.

The raw data of the catalog had a lot of different features that could be used for classification. However, going by the previous studies, we decided on using the most important features, which included Flux density and the error on it, spectral index, the curvature, hardness ratios (as defined by Parkinson et. al.), variablity, and also the galactic latitude, the last of which was used even in the classification of AGN and Pulsars (as opposed to Parkinson, who used it only for the young and milli-second pulsar distinction). In features where the values were high, we used the logarithmic scale to better seperate the sources. The complete list of sources, along with some statistics, is given in the table below. The influence of the features on the classification, especially the differences in the various methodologies is discussed in much more detail in the next section.

One of the main aims of our project was to understand and optimize the machine learning methods which we were using. So apart from the features which were in the data itself, we also theorized and experimented with the parameters of the algorithms themselves. We wanted to find the fastest and cost-effective way of using certain methods, without going into regimes of under and over-fitting the data. Parameters which we studied range from Depth and Number of trees in Forest based methods to the number of hidden layers and epochs in neural networks. The details are given in the next section, where we discuss our expectations and the resulting behaviour of our algorithms.

In our general the Methodology was as follows.

1. Split the PS with known classification into learning and test samples.
2. Use the learning sample for training and for selection of features. In particular, continuous parameters, such as the thresholds in the decision trees or mixing matrices in neural networks, are determined from the learning sample.
3. Meta-parameters, which encode the complexity of the methods, such as the depth of the decision trees, are determined from the best performance on the test sample.

After the above had been completed we were ready both with our final data and our optimum algorithms. We then applied and sought the results using both the catalogs in our possession. This is discussed in detail in section 4 and 5.

2.1. Details of the analysis

2.2. Data and Features

The total number of sources, including unassociated and associated, in the two catalogs is show below.
[Add Table]

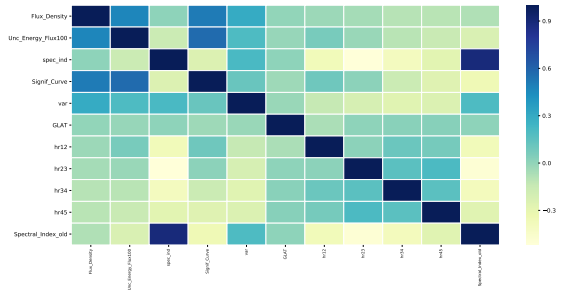

Fig. 1. Correlation matrix for the most important features

The features used for our analysis follow the same idea as the previous studies. The features, along with statistical and methodological details, are given below. A correlation matrix is presented for the most important features as well. The matrix is important for the case where there might be redundant features, in which case using only one of the two features would be a better idea.

[Add Table of features for both catalogs]

Our initial hypothesis was that certain features would be more important for classification than others. For instance, as shown below, one can see a clear distinction between the regimes of AGNs and Pulsars, based on spectral idex and significant curvature. [Add image] While not clearly obvious from the get go, we were also interested in comparing the importance of features based on the algorithms that we were using. Due to the difference in the basic method of Random Forests and Neural Networks, we expected a slight shift in their reliance on certain features. Despite that we hypothesized that features with the most contribution would be among spectral index, variability, and the curvature; as already observed by Parkinson et. al.

2.3.

1. Describe the features that we use for the analysis.
2. Describe the objective function for minimization (accuracy of classification on learning sample). Weighted objective function: give more weight to pulsars, since there are fewer of them in the catalog.
3. Learning curve using all features? *Plot: classification accuracy using the total list of features for learning and test sample as a function of complexity parameter.*
4. Selection of the most important features. *Table: features vs algorithms. Columns: algorithms, rows: features, values: significance.*
5. Selection of meta-parameters. *Plot: classification results for the test sample using a subset of features.*
6. Train the final classifier. *Table: classification accuracy of the final classifiers for different algorithms using the test sample from 3FGL.*

Discuss the general features of the optimal algorithms: which features turn out to be important, what is the depth of the trees, the number of trees in random forests, the depth and number of internal nodes in the neural networks.

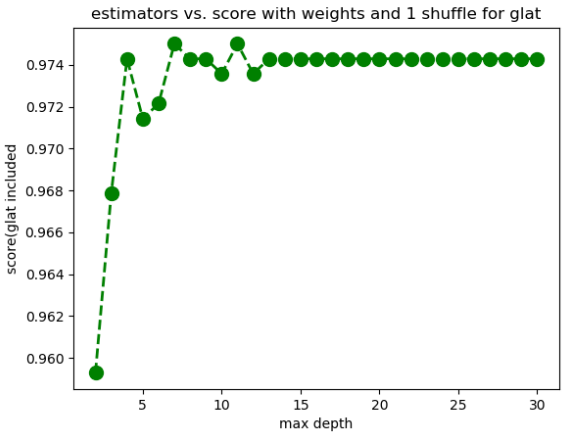

Fig. 2. Example of a figure for one column.

2.4. Comparison of the classification algorithms

Plot: classification domains for a pair of features (or different pairs of features, e.g., latitude vs index, index vs curvature, latitude vs variability).

Probabilistic classification? Result: probability for a source to belong to a particular class. Result of classification: table of sources with probabilities for different algorithms. Final probability: the probability for one of the algorithms (for the most precise one?) and uncertainties determined from the other algorithms.

Discuss a few examples where algorithms give different predictions (are these sources at the boundaries of the domains).

Discuss examples where algorithms misclassify sources from the test sample.

In the case of test data, we worked with three different classification algortithms, namely Random Forests, Ada Boost, and Neural Networks. Here we were mostly concerned with tweaking the parameters of the

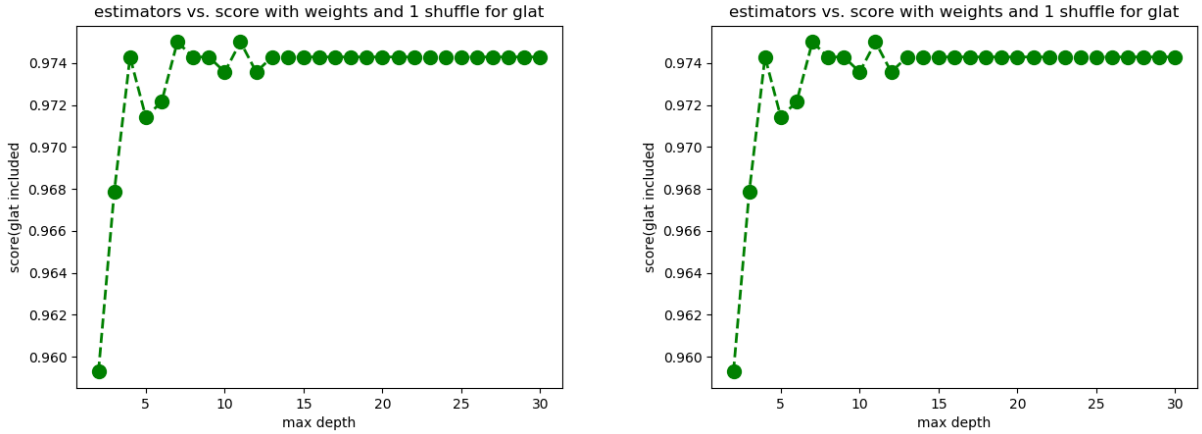

Fig. 3. Example of a figure for both columns.

classification algorithms involved, minimizing the cost of computation and aiming for the most efficient way of classification.

2.4.1. Random Forests

The two main parameters involved in Random Forests are the number of trees and the maximum depth of the trees involved. Figures below shows one instance of the accuracy as a function of maximum depth when the number of trees was kept constant, and as a function of number of trees when the maximum depth was kept constant.

2.4.2. Neural Networks

In the case of neural networks we were concerned with the number of epochs that one would need to tweak, along with a dependence on the number of neurons in the hidden layers. A final improvement involved checking whether multiple hidden layers would actually add to such a classification algorithm or not.
As can be seen in the figure, a complex network with two hidden layers (100 and 5 neurons) reaches the maximum accuracy pretty fast. The results becoming more consistent at higher epochs. A similar result was found for a network having two hidden layers but with only 20 neurons in the first layer. However, such networks could also lead to overtraining, and therefore it is important to check whether such a high accuracy could perhaps be reached by less complicated algorithms, which would drastically reduce the chances of overtraining and allow for a more flexible classificaition methodology.

A consistent and accurate result is found even for networks with only one hidden layer with 20 and 5 neurons in the hidden layer respectively. There seems to be no significant dependence for the number of epochs above 30, and even a simple network with one layer and 5 neurons shows a high accuracy for only 30-40 epochs.

3. Prediction for unassociated sources in 3FGL and comparison with 4FGL

Apply the algorithms on unassociated sources in 3FGL.

Plot: add unassociated sources on the plots with domains for the best algorithm.

Create a table with sources which are more likely to be pulsars (select about 20 the most likely candidates). Compare the accuracy of the algorithm for the sources which have an associate now in the 4FGL catalog.

4. Prediction for unassociated source in the 4FGL catalog

5. Conclusions

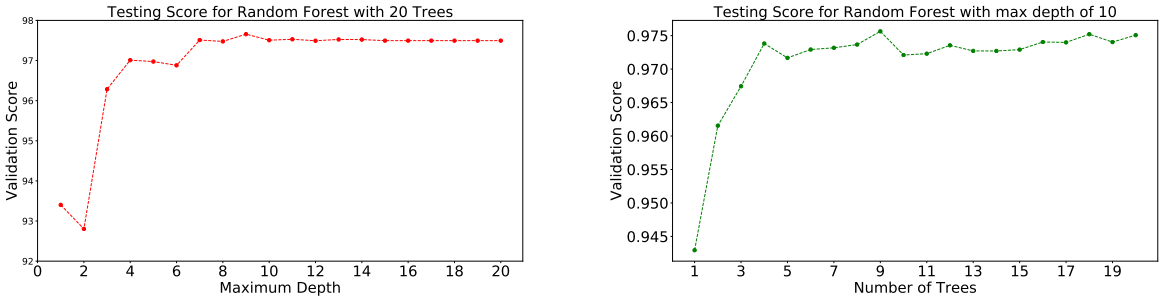

Fig. 4. Random Forests on Testing Data

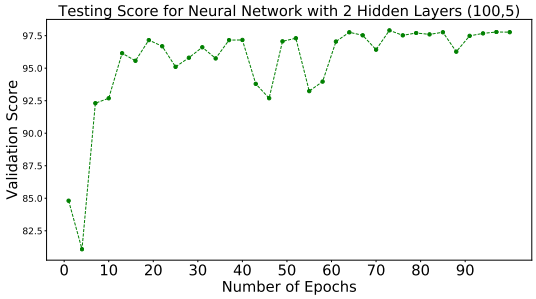

Fig. 5. Example of a figure for one column.

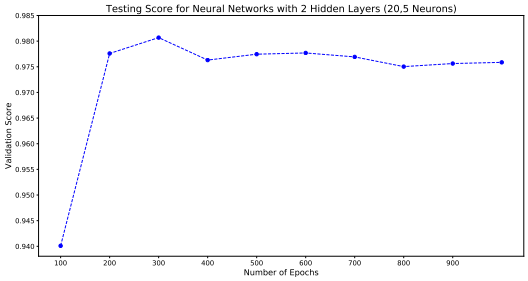

Fig. 6. Neural Network

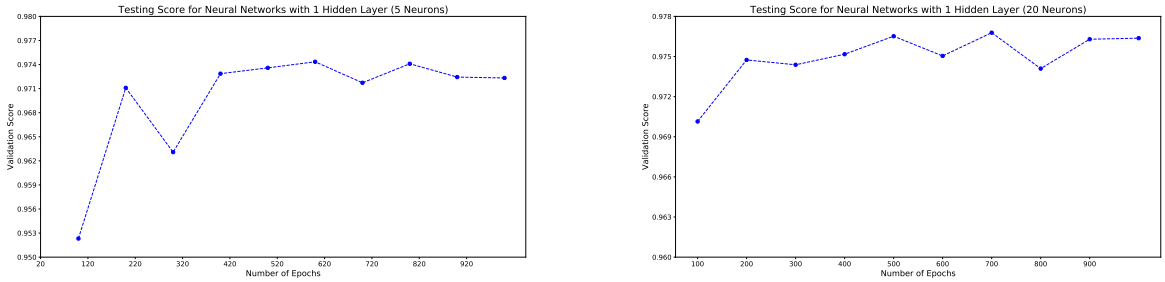

Fig. 7. Neural networks

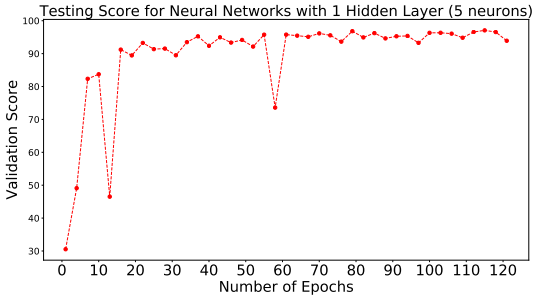

Fig. 8. Neural Network

Appendix A: Appendix

If we need one.