VOIMachine

Sharmad Navelkar
Ninan Sajeeth Philip
Virtual Observatory India

What is it?

- VOIndia's attempt to make Machine Learning (ML) tools easily accessible.
- Provide web based access to ML services running on a server.
- Similar to online image cut-out and galaxy morphology services, namely VOI Mosaic service and Pymorph.
- Interfacing ML tools with VO tools and services.

Immediate Goals

- Creation of a functional model consisting, for now, of a single ML tool.
- The ML tool we have selected for the pilot attempt, is called a Decision Based Neural Network (DBNN). C++ code by Ninan Sajeeth Philip.
- Development of a user-friendly interactive GUI.
- Integrating it with visualization and statistical packages, namely VOPlot and Astrostat.

Motivation behind DBNN

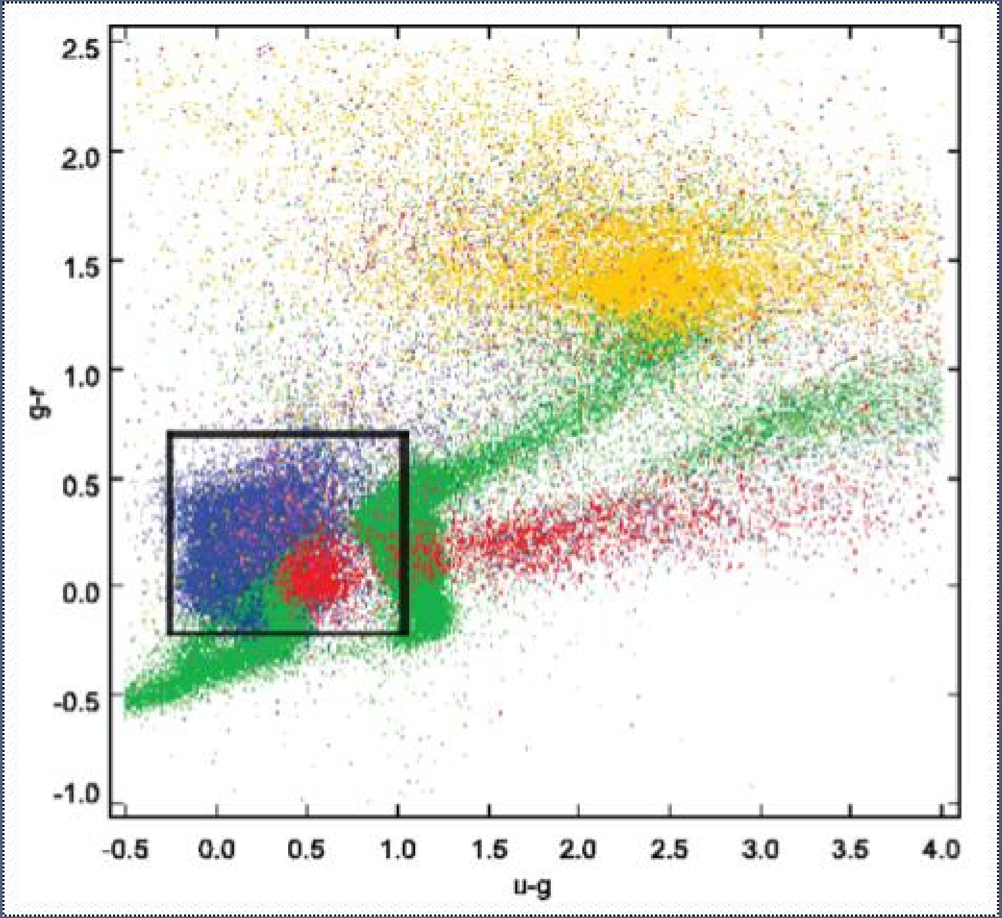
Motivations behind DBNN

- The primary advantage of ML tools in general is their ability to handle huge datasets.
- Cosmology, for instance, depends on the number counts of each type of object, rather than their individual details, to properly understand and build a realistic model of the universe.
- The type or nature of objects, can traditionally be established only through expensive methods like spectroscopy.

Motivations behind DBNN

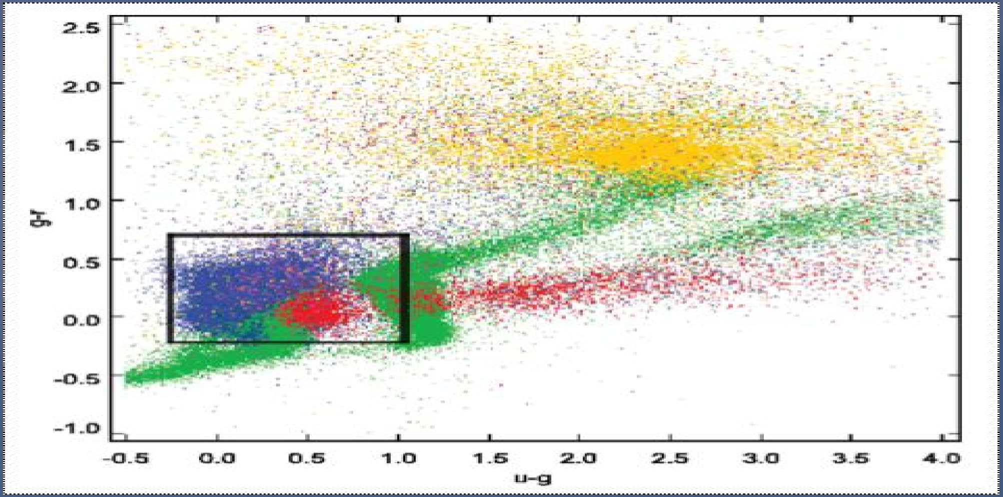
- The SDSS over it's 12 years of service could take spectra of about 1% of the objects in it's imaging survey.
- How do we determine the nature of the remaining 99% of objects that do not have an observed spectra?
- With an ML tool like DBNN, we can classify objects based on their nature.

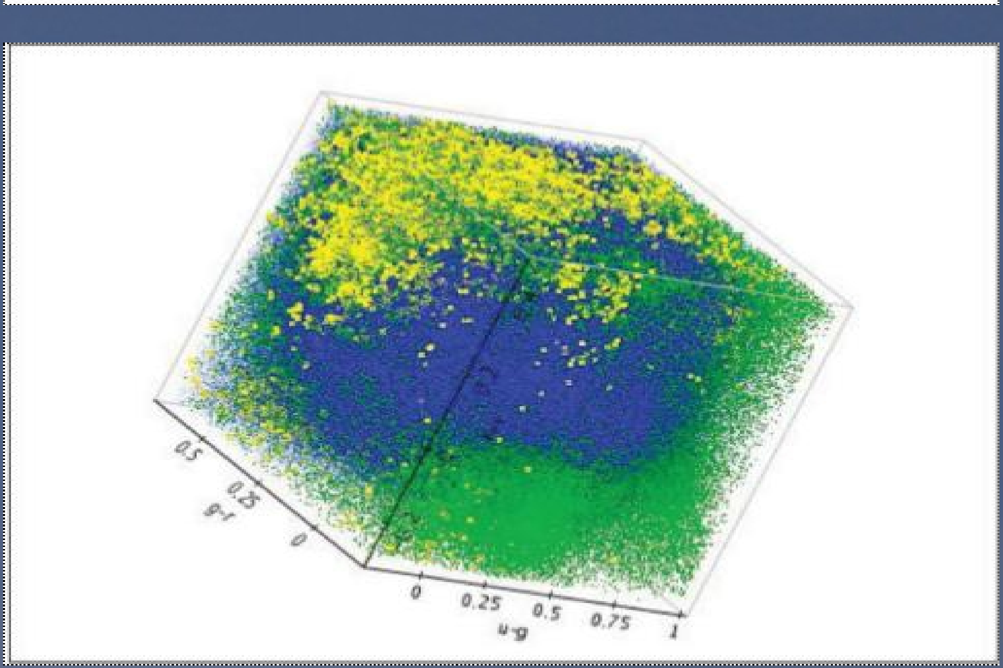
Motivations behind DBNN



- An alternative option to spectroscopy is color.
- Color depends on the amount of light in every frequency, which in turn is related to the spectra of the image that emits it.

Motivations behind DBNN





- The ML algorithm could determine the probable class of about 6 million objects where as spectra is available only for about 100,000 of them.
- Once we have predicted the class of the object, we can selectively observe the interesting candidates to speed up the confirmation process.

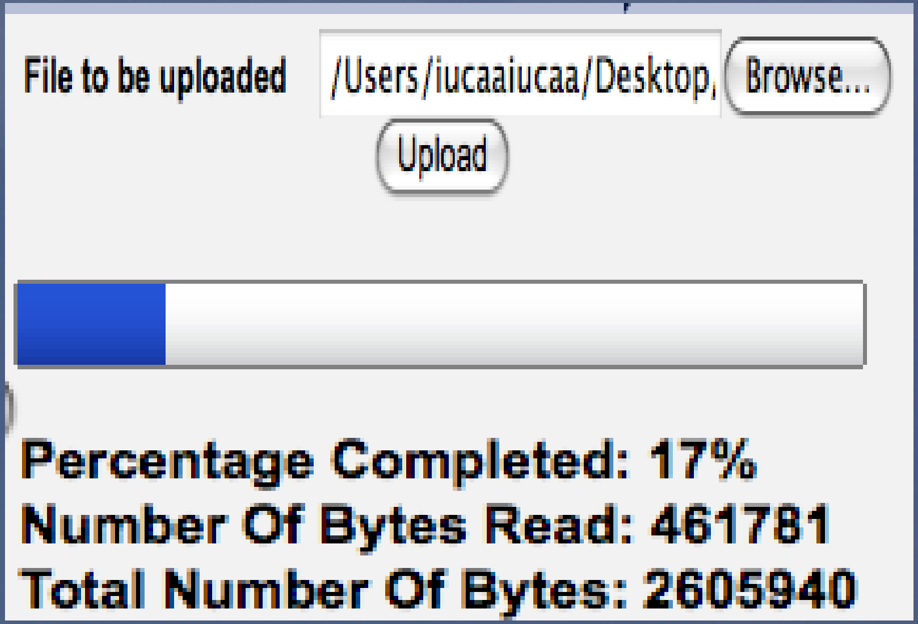
DBNN in it's original form

- DBNN is a Bayesian classifier for machine learning applications
- The software implementation works on Linux platforms.
- Requires data to be in a text file with space delimited values.
- Requires user to create parameter files based on their data.

Functionality

Functionality

Data Definition

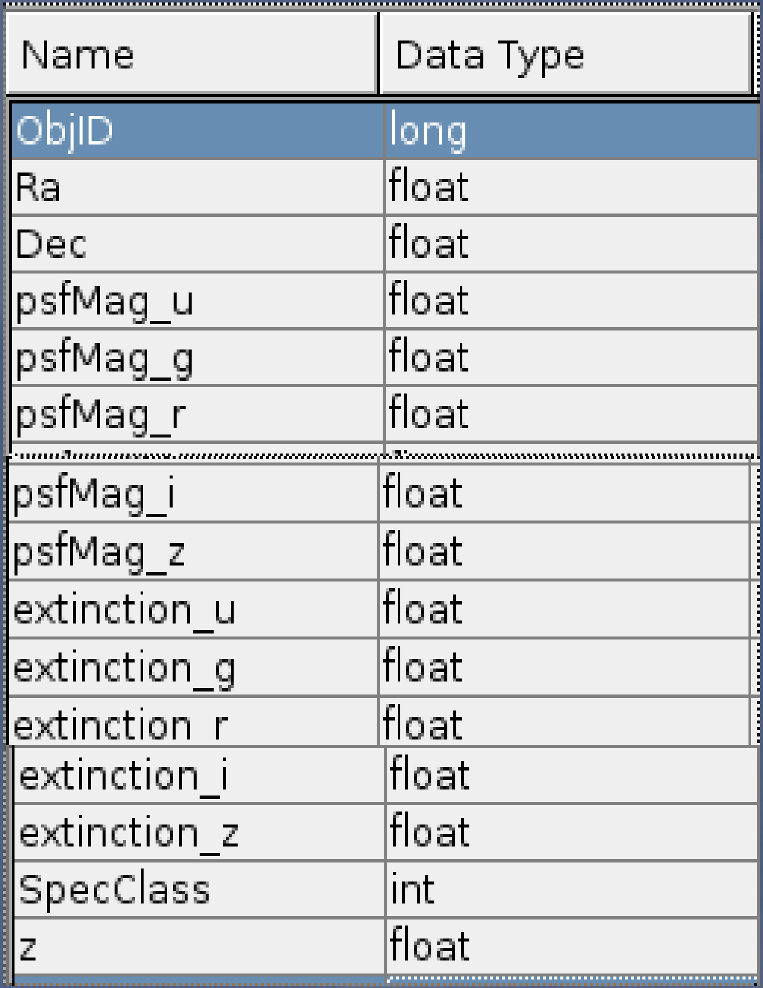


Percentage Completed: 17%
Number Of Bytes Read: 461781
Total Number Of Bytes: 2605940

- With the online interface, we will allow users to upload their own data files, or import from VO enabled catalogs like VizieR.
- Conversion to DBNN accepted format will be done by the system behind the scene.
- Parameter files will be generated by the system on the fly.

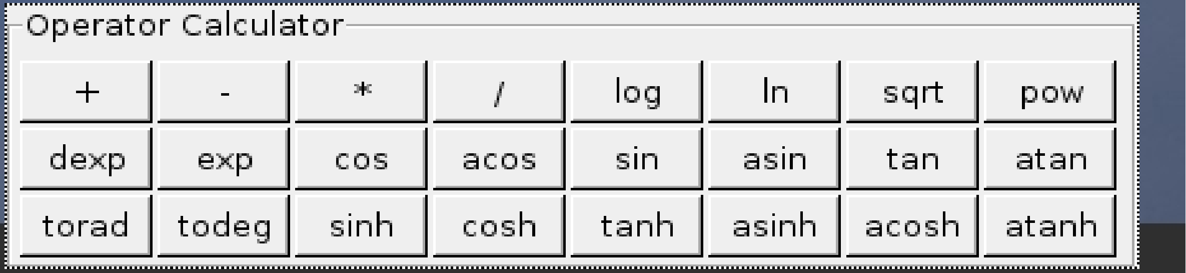
Functionality

Data Manipulation



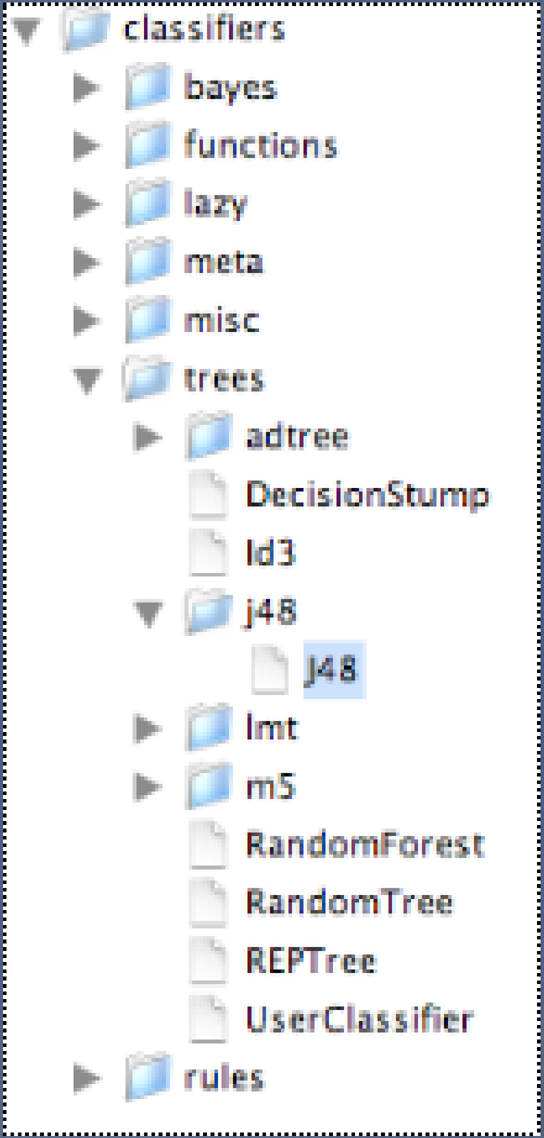
Name	Data Type
ObjID	long
Ra	float
Dec	float
psfMag_u	float
psfMag_g	float
psfMag_r	float
psfMag_i	float
psfMag_z	float
extinction_u	float
extinction_g	float
extinction_r	float
extinction_i	float
extinction_z	float
SpecClass	int
z	float

- In case of data imported from catalogs, selection of columns users wish to work with.
- Column manipulation, creation of new columns by applying standard operations.
- Filtering the data.



Functionality

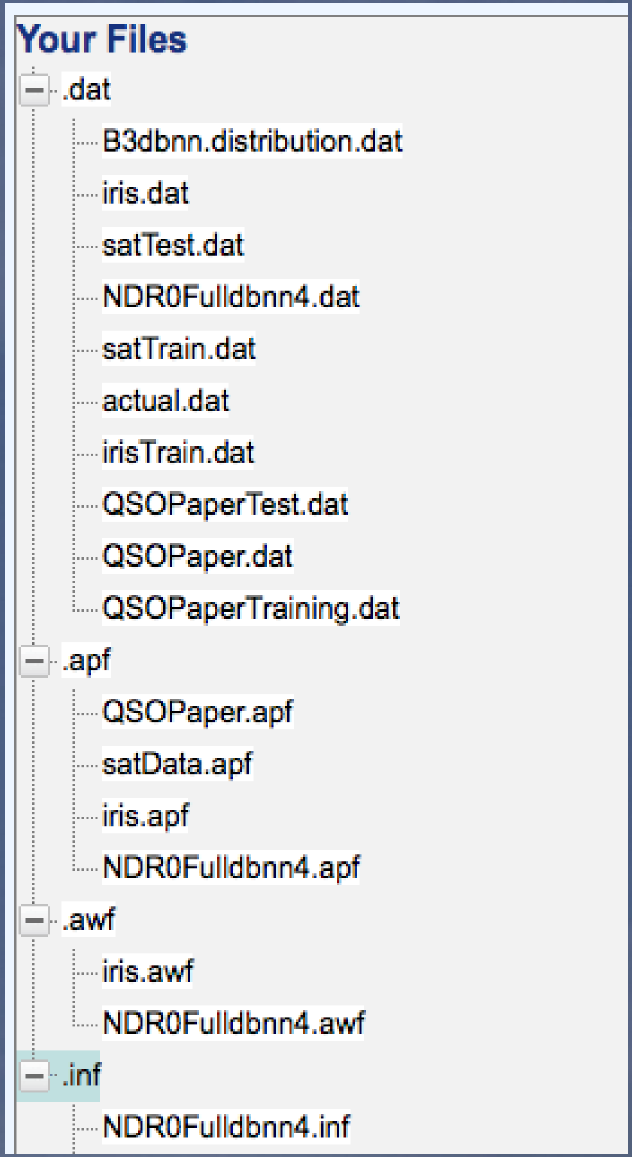
Training the ML tool



- Select the ML tool that you wish to train with the data. ML tools classified by types. (currently only one)
- Generate meta-data and parameter files for the chosen data and ML tool.
- Specify training and testing data percentage from the loaded data.
- Train the network.

Functionality

Classify



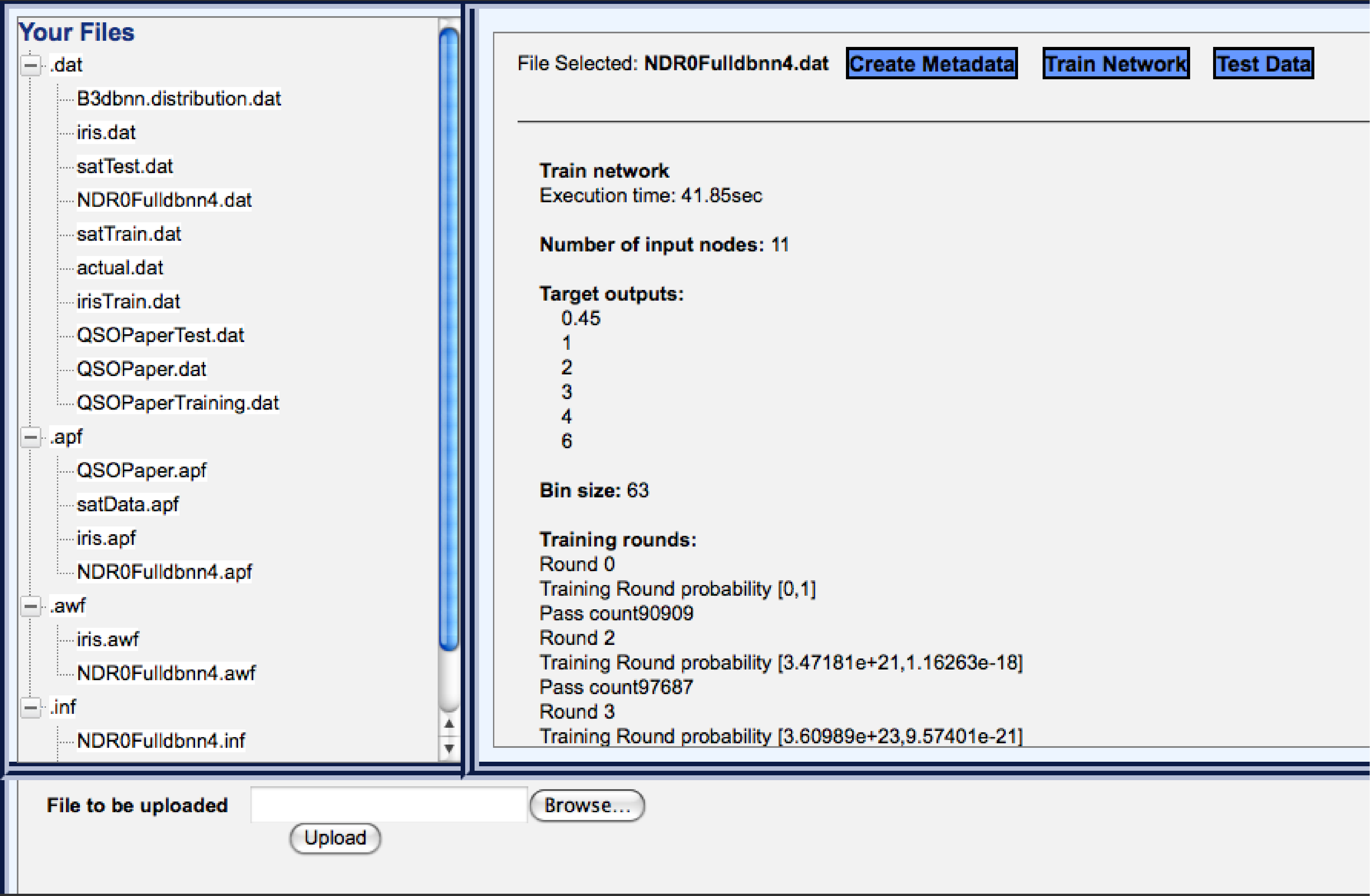
Your Files

- .dat
  - B3dbnn.distribution.dat
  - iris.dat
  - satTest.dat
  - NDR0Fulldbnn4.dat
  - satTrain.dat
  - actual.dat
  - irisTrain.dat
  - QSOPaperTest.dat
  - QSOPaper.dat
  - QSOPaperTraining.dat
- .apf
  - QSOPaper.apf
  - satData.apf
  - iris.apf
  - NDR0Fulldbnn4.apf
- .awf
  - iris.awf
  - NDR0Fulldbnn4.awf
- .inf
  - NDR0Fulldbnn4.inf

- The ML tool is now ready to accept data for classification process.
- Users could also specify a target class to test the predictions with and to generate an accuracy figure. This could be a column in the data itself or as a separate file.
- Save predictions to file or append to test data as new column.

Long Term Goals

- Integrate more machine learning tools in a modular fashion.
- More comprehensive UI with plenty of graphics and figures accompanying the results.
- Authoring tutorials and examples.



File Selected: **NDR0Fulldbnn4.dat**

Train network
Execution time: 41.85sec

Number of input nodes: 11

Target outputs:
0.45
1
2
3
4
6

Bin size: 63

Training rounds:
Round 0
Training Round probability [0,1]
Pass count90909
Round 2
Training Round probability [3.47181e+21,1.16263e-18]
Pass count97687
Round 3
Training Round probability [3.60989e+23,9.57401e-21]

File to be uploaded

THANK YOU