Decision Trees

The Tic-Tac-Toe dataset is composed of 10 columns which we have split up into 9 features being:

Top-Left square,
Top-middle square,
Top-right square,
Middle-left square,
Middle-middle square,
Middle-right square,
Bottom-left square,
Bottom-middle square,
Bottom-right square,

that all take up the values ‘x’, ‘o’, or ‘b’ and a target class:

Outcomes

which takes up the values, positive and negative.

Upon execution of the algorithm which is in the form of a python script (on Jupyter Notebook), the outputs for each request are displayed and the program is spilt into separate blocks to ensure that the outputs may be distinguished from their requests.

The following libraries were used:
Numpy
Pandas
Sklearn

Logistics:

The program reads in the dataset as a type .csv then renames the columns to make for easier access to the features.
From there the dataset descriptions (How many rows and columns) are printed then an example of how it would look for 6 entires is displayed.

Thereafter the number of times a value may appear for each feature is show, target class probabilities are defined and the number of times a positive or negative outcome may appear for a given feature value is defined and later on , within the while loop, specified.
These values are used for the definition of the entropy and gain functions.
And the gain values for each feature are then displayed.

String values are difficult to run entropy calculations and fitting on, so a preprocessing label encoder from the sklearn library is used to convert the dataset into integer values.
The dataset is then split into X, Y, training and testing data sets and there after printed (please see DecisionTreesFinalJupyter.pdf)
The classifaction entropy function is then defined and the training data is fitted to it and from the fit an array of predictions from the training data is made.

The fit at first returned an accuracy of 100%, but with constant modifications to the program, later regressed to 65%

A sample input for testing was then parsed into the model and the correct answer was return.