CS 6923: Machine Learning

Spring 2019

Final Project

Submit on NYU Classes by Friday, May 3rd at 6:00 p.m. You may work together with one other person on this project. If you do that, hand in JUST ONE project for the two of you, with both of your names on it. Your code will be run through a PLAGIARISM CHECK, so ensure that you do not share code with other students.

IMPORTANT SUBMISSION INSTRUCTIONS: Submit your solutions in 3 separate files:
1) a pdf file with your report (report.pdf)
2) your main code file
3) a readme file with instructions on how to run your code (readme.txt)
Once the test set is given to you, submit a csv file with your predictions for the test examples (test_outputs.csv, in the format indicated below)
DO NOT SUBMIT ANY ZIP FILES.

The Game of Quidditch

Quidditch is the most popular game in the wizarding world. Like games played in the real world, Quidditch too has its own set of rules and regulations.

In this project you will be working on a classification problem based on a Quidditch dataset. Your task is to predict whether a player makes it to Professional League Quidditch upon graduation from Hogwarts or not based on other features of the player in the dataset.

We are giving you a training set with labels. You will experiment with this training set in order to develop a good model (hypothesis).

Once you submit the program you used to train your model, the test set (without labels) will be given to you. You will be given a 3 hour window to submit the predicted output labels for this test set.

This is your opportunity to explore and experiment with different machine learning techniques. A large part of your score on this homework will be based on the report in which you describe your experiments and describe how you developed your final learning method.

You are expected to try 3 different supervised learning methods, and to experiment with different parameter settings or techniques associated with your final method. Do NOT just use tools to try different methods with their default parameters! You will receive almost no credit if you do this. You will get more credit for thoughtfully choosing 3 methods, trying them, and then devoting time to improve one of them to get your final program.

Features

The last "feature" is actually the target value, **quidditch_league_player**, that you need to predict. This feature indicates whether the player made it to Professional League Quidditch or not. The other features are the input features.

Often in real-world datasets, there are *missing values* for some of the features in some of the training and the testing examples. This may be because the feature values for those examples were unavailable or were not recorded. Similarly the wizardry-world Quidditch dataset contains missing values for some of the features. It is up to you to decide how to handle these values.

Evaluation Metrics

This is a classification problem and you will choose the best evaluation metric you think is appropriate for this task. You will have to justify your selection.

Input and Output Instructions

Your final program must read in two files: a training file (train.csv) and a test file (test.csv). Your program must use the training file to learn a predictor, apply that predictor to the examples in the test file, and then write a file called test_outputs.csv which gives the predictions for the unlabeled examples in the test file.

IMPORTANT: Your program must be able to take train.csv and test.csv as input. Do not modify these files prior to giving them to your program as input. Your program must write the predictions to the output file directly, in the form indicated below.

The first column of the training and test files is the Id Number. This is different for each example (person), and will be used as the number of the example.

Your output file must be named test_outputs.csv, and it must have just two columns: "id_num" and "quidditch_league_player" (the output). INCLUDE a row at the top of your output file with the column headers, "id_num, quidditch_league_player". Each row of the output file, after the row with the column headers, should have the Id of the example in the test file, with the predicted value for the class. (Do NOT change the Id numbers of any of the examples.) Because your output file should be in .csv format, there should be a comma between the entries in the two columns.

Thus the first three rows of your test file should look something like this (with different Id numbers and different quidditch_league_player labels):

```
id_num,quidditch_league_player
58,NO
289,YES
```

You must write your program in Python. In Python, you may use the tools provided in scikit-learn if you would like to do so, rather than implementing the machine learning methods yourself.

Report

You must submit a report with your program. A template of the report is provided to you. You will follow the template for your submission.

Even if your results on the test set are not good, you can still score well on this project if you have a thoughtful report. Conversely, even if you achieve good results on the test set, you will lose points if your report does not adequately address the items discussed listed out in the template.

Code

You must submit your code file along with instructions how to run it.

1. Your code must read 2 files as input - train.csv and test.csv
2. Your code must output just 1 file - your test_outputs.csv
3. All data cleaning, preprocessing, feature selection, model fitting, cross validation, etc. must be performed by your code, using the above input. Do not modify the given files before you run your code.
4. Your code must be runnable. Include all necessary instructions in a README file.
5. Make your code easy to read and understand.
6. IF YOU DO NOT FOLLOW THESE GUIDELINES YOU ARE LIKELY TO RECEIVE 0 POINTS FOR YOUR CODE AND PREDICTIONS!

Things to think about

1. Before you start fitting your models, you might want to check the distribution of the values of individual features and the distribution of the target value.
2. Which features are actually important? Do you want to use all of them? Do you want to combine any features? Do you want to create new features? Do you want to combine any feature values?