Introduction to Machine Learning (CSCI-UA.473): Homework 4

Instructor: Lerrel Pinto

October 18, 2022

Submission Instructions

You must typeset the answers using LaTeX and submit the report as a single PDF file. Name the report as ⟨Your-NetID⟩_hw4.pdf. The naming guidelines for the program scripts have been provided in the respective sections. Submit the files through the following Google Form - https://forms.gle/g5t7GQ9i2hqwg3sL9 The due date is **November 3, 2022, 11:59 PM**. You may discuss the questions with each other but each student must provide their own answer to each question.

Problem Statement

In this assignment, you will use Decision Trees and Random Forests to classify emails into spam or non-spam categories. You have to submit the report file in pdf format. **The programs used for the homework are required to be submitted**.

Data Set Description

An email is represented by various features like frequency of occurrences of certain keywords, length of capitalized words etc. A data set containing about 4601 instances are available in this link (data folder):
https://archive.ics.uci.edu/ml/datasets/Spambase

The data format is also described in the above link. You have to randomly pick 70% of the data set as training data and the remaining as test data.

Questions

This section comprises two parts. You must provide a single report comprising the results obtained in both parts. Also, mention the libraries used in your

solution along with details about the methodology such as reading the data, data normalization used, etc.

Part 1: Spam email classification using Decision Trees (50 Points)

In this part, you must classify the above data set using Decision Trees. The code must be written in **Python** and you can use any Python package to solve the question. You must report the best classification accuracy achieved across 20 different seeds. The accuracies must be reported for the decision tree created using (1) Gini impurity, and (2) Shannon information gain. The program script for this part must be named ⟨NetID⟩_hw4_part1.py.

Criterion	**Best Test Accuracy**
Gini Impurity	
Shannon I.G.	

Shannon I.G. refers to the Shannon Information Gain.

Part 2: Spam email classification using Random Forests (50 Points)

In this part, you must classify the above data set using Random Forests. The code must be written in **Python** and you can use any Python package to solve the question. For this part, you must fill up the following table with the best classification accuracy achieved across 20 different seeds.

Criterion	**# Estimators**	1	3	5	10	15	20	40	70
Gini Impurity	Test Accuracy								
Shannon I.G.	Test Accuracy								

Shannon I.G. refers to the Shannon Information Gain. Provide an intuition for the results observed for the different hyperparameters used. The program script for this part must be named ⟨NetID⟩_hw4_part2.py.

Extra Credit Question (25 Points)

Since this is an extra credit question, no additional clarification or help will be provided. If you are solving this, you are expected to go through the details provided and solve the question accordingly.

Problem Statement

Write a program to learn a decision tree and use it to predict class labels of test data. Decision tree learning should use information gain as the criterion for choosing the attribute for splitting. If there is a tie between two attributes, the lower attribute number should be chosen (e.g., if there is a tie between $x2$ and

$x5$; $x2$ should be chosen). **Tree pruning should not be performed.** The learned tree should be tested on test instances with unknown class labels, and the predicted class labels for the test instances should be printed as output.

Data Set Description

Training Data Filename: data2.csv

Training Data File Format: Boolean input attributes $(x1, x2, \ldots, x8)$ in first 8 columns. The last (9^{th}) column represents the Boolean class label (y). Each row is a training instance. There are 24 training instances.

Test Data Filename: test2.csv

Test Data File Format: Boolean input attributes $(x1, x2, \ldots, x8)$ in each of the 8 columns. Note that, there is no class label column. Each row is a test instance. There are 4 test instances. The row number corresponds to the instance number of the test instances.

Please STRICTLY follow the program input/output format specified below.
Input Format: Assume the data files *data2.csv* and *test2.csv* is present in the same directory and contain the training and test data. Thus, your program should not require any input from the user and should read from these files. Strictly use these filenames only.
Output Format: Predicted class labels (0/1) for the test data exactly in the order in which the test instances are present in the test file. Put a blank space between printed class labels. (e.g., output 1 0 1 0, if the predicted class labels are - Test Instance 1: 1, Test Instance 2: 0, Test Instance 3: 1, Test Instance 4: 0). Output, in above format, should be printed to the file: ⟨NetID⟩_extracredit.out (e.g., lp91_extracredit.out). Strictly use this filename format.

Submission Guidelines: You must use Python for this part of the homework. You should name your file as ⟨NetID⟩_extracredit.py (e.g., lp91_extracredit.py). Your program should be standalone and should not use any special-purpose library. NumPy may be used. You should submit both the program file and the output file.

Extra Resources

1. Decision Trees - https://www.youtube.com/watch?v=$_L 39rN6gz7Y$
2. Random Forests - https://www.youtube.com/watch?v=J4Wdy0Wc$_x Q$
3. Gini Impurity - https://blog.quantinsti.com/gini-index/
4. Information Gain for Decision Trees - https://www.youtube.com/watch?v=FuTRucXB9rA