COSC74/174: Machine Learning and Statistical Data Analysis
Homework 1. Due: 11:59PM on Friday, 11 Oct 2019
Instructor: Prof. Soroush Vosoughi

Part 1) Binary classification (70 points)

You are given a training dataset in CSV format. The training data has 5,600 rows:

- Columns 1 through 6 of the given CSV file represent independent variables
- The last column ("Label") represents the dependent variable (or the class label) (0 or 1)

A) You are required to implement the following models and train them using this dataset:
   1) Implement a Multinomial Naïve Bayes classifier, with smoothing (you can set the smoothing value to 1). (20 points)
   2) Implement a Gaussian Bayes classier. (20 points)
   3) **Bonus (10 points):** Implement a decision tree using the c4.5 algorithm covered in class. The decision tree function should take in as input a splitting criterion (either entropy or Gini). As this is a bonus problem, you have a lot of leeway for what to implement (for instance you can use any stopping criterion you want).

B) You are required to implement a function for K-fold cross-validation of the models. You should implement and use the F1-metric as your performance metric. (5 points)

C) Compare the performance of all implemented models (Multinomial NB, Gaussian NB and if you are attempting the bonus, the decision tree models using entropy and Gini) using cross-validation and the F1-score. (Printing the performance number is fine, you do not need to create any figures). (5 points)

D) Next, you are required to do everything you did above using the scikit-learn library (https://scikit-learn.org/stable/). Compare the results of the scikit-learn library to your own implementation. (Printing the performance number is fine, you do not need to create any figures).) Note that there might be differences between your models' performance and the performance of the scikit-learn library, do not worry about that, you will still get full grade. (5 points)

We will test your predictor by giving you a test set with 2,400 rows for which the last column is blank (i.e. you do not know the true class to which rows in the test data belong). You should use your best performing model (NOT the scikit-learn library implementation but your own) to classify the data in the test set. (15 points)

You will submit both your results and code. The grade you receive for your project will depend upon the accuracy of your best model relative to our benchmark, and the quality of your code. The scikit-learn implementation only counts for 5 points, the rest is based on your implementation and results.

Please include the following in your submissions:

1. A new CSV file of the test set with an added column, “Label”, showing the dependent variable (0 or 1) that you predicted using your best model.

2. Your code (e.g. Jupyter Notebook or python script).

NOTES:

- Your code should contain functions that abstract the training and prediction phases:
  - o train(data), where input data is the CSV filename of the training dataset, and output is a classifier C .
  - o predict(C, row), where input C is a classifier object on train(...) was called, input row is an array corresponding to some row of IVs in the CSV test dataset (a single row only, to account for variable dataset sizes), output is 0 or 1.

Part 2) Association Rules (30 points)

A) Implement the aprior algorithm with lift. The function should take three parameters: support, confidence, and lift and return rules that satisfy these conditions. (25 points)
B) Run the algorithm on the market basket data from the lecture slides with the following parameters: (5 points)
   1) Support=0.1, Confidence =0.30, minimum lift = 1
   2) Support=0.2, Confidence =0.65, minimum lift = 0
   3) Support=0.2, Confidence =0.65, minimum lift = 1.1
   4) Support=0.2, Confidence =0.65, minimum lift = 2
   5) Support=0.2, Confidence=0.50, minimum lift = 1.1
   6) Support=0.3, Confidence=0.70, minimum lift = 1.1

Generate a csv file with the results. Each column in the csv file should correspond to one of the conditions (in order). The rows correspond to the rules learned (so if you have learned 3 rules for condition 1, the first column will have 3 rows, one per rule, plus the header that specifies the condition number). Note that some conditions might not generate any rules, and that is fine.

You will submit both your results and code. The grade you receive for your project will depend upon the rules in the CSV files as compared to our benchmark, and the quality of your code.

Important:

- Projects must be coded in Python
- You are responsible for installing the Scikit-learn library and getting it to work.
- You are responsible for making sure that your project is properly submitted, and your code can be properly run.
- Please be sure to submit all parts of your homework in the required format.

• All work must be your own. Academic Honor Principle applies to all parts of the project. Please refer to http://student-affairs.dartmouth.edu/policy/academic-honor-principle for more detail.
• All projects will be due by the deadline posted. Late submission policy is explained on Canvas.