CS362 Artificial Intelligence
Second Semester 2018/2019
Project

This is a 3-part project that is intended to familiarize you with Machine Learning (ML). The 3 parts are as follows:

- In the first part, you're required to implement an attribute selection algorithm. Given a dataset of m attributes, the algorithm simply computes the Gain Ratio of each of the attributes and retains the top $\left\lceil \frac{m}{10} \right\rceil$ attributes. This part should be implemented on Alen Shapiro's Chess Dataset.[1] In this dataset, there're 36 attributes, so, your algorithms should pick the 4 ones with the highest Gain Ratio and store the resulting dataset (with only these 4 attributes) into a separate file.
- In the second part, you'll implement the k-nearest neighbor algorithm for classification. Use the Euclidian distance and k=1 and apply your algorithm on the Breast Cancer Wisconsin (Diagnostic) Dataset.[2] However, before implementing the algorithm, split your data into a training set and a testing set. The training set comprises of the first 90% of the instances while the testing set comprises of the remaining 10%. Your algorithm should store its predictions in a separate file and output the accuracy of these predictions.
- In the last part, you'll implement a simple clustering technique that utilizes two versions of the same dataset, a discretized version and a non-discretized (original) version. Specifically, we'll use the Pima Indian Diabetes Dataset Discretized by Mangrove.[3] The dataset has many attributes, but we'll focus only on 5 non-discretized attributes (Age, BMI, Glucose, Insulin, Pregnancies) and 5 discretized ones (LabelPAge, LabelPBMI, LabelPGlucose, LabelPInsulin, LabelPPregnancies). So, the first thing to do is to remove everything but these 10 attributes. The algorithm starts by computing the correlation between every pair of the non-discretized attributes and chooses the pair with the lowest correlation (i.e., with correlation coefficient closest to 0). Let's call this pair a_x and a_y. Then, for these two attributes, it creates a cluster for every possible combination of values for the discretized versions of a_x and a_y. E.g., let's say that the discretized version of a_x has the values *high* and *low* and the discretized version of a_y has the values *large* and *small*. Then, there will be the following 4 clusters:
  - C1: with records containing the values *high* and *large* for a_x and a_y, respectively.
  - C2: with records containing the values *high* and *small* for a_x and a_y, respectively.
  - C3: with records containing the values *low* and *large* for a_x and a_y, respectively.
  - C4: with records containing the values *low* and *small* for a_x and a_y, respectively.

  Your algorithm should create a separate file containing the records of each cluster. It should also evaluate the resulting clustering by computing the maximum Euclidean distance between any two records in the same cluster and minimum Euclidean distance between any two records in different clusters. Note that these distances should be computed based on the 5 non-discretized attributes.

This project must be done individually on Python. Erroneous submissions will be given a zero grade. There will be a discussion session after the submission deadline. Failing to show up for the discussion sessions or failing to answer questions correctly during the discussion will result in a zero grade.

---

[1] https://archive.ics.uci.edu/ml/datasets/Chess+%28King-Rook+vs.+King-Pawn%29

[2] https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29

[3] https://www.kaggle.com/blackbee2016/discretized-datasets-by-mangrove#mangrove_transformed_diabetes.csv