[CSC 5825 Fall 2022]

Homework 2

Due 11:59 pm, October 12, 2022

Full credit: 100 points

Question. Generative Classifier Implementation

In this question, you are asked to train two generative classifiers (Naive Bayes and k-NN) from scratch to dealing with a classification task. The heart failure prediction dataset contains 11 common features (e.g., age, sex, and several medical predictor variables) from 918 patient examples that can be used to predict whether the patient has heart disease (1) or not (0). You can find more details of the dataset from Kaggle website `https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction`.

Tasks:

- Split the 918 instances into training and test sets (8:2) for Naive Bayes classifier. This is because the features have already been manually selected and no further tuning is necessary. While for the k-NN classifier, you need to split the instances into training, validation, and test sets as (6:2:2). The validation set is used to fine-tune the hyper-parameter k. (10 points)
- Train your Naive Bayes and k-NN Classifiers on the training set. (60 points)
- After training, test your classifier on the test set, report the confusion matrices and accuracy, precision, recall, and F1-score, these metrics are calculated from confusion matrix. (30 points)

Guidelines (for k-NN only):

- Use Euclidean distance (L2) to compute distances between instances. As the attributes in Heart Disease dataset are either categorical or continuous. In the case of mix of these two, the categorical variables may be mapped to numerical values (through one-hot encoding) before applying the k-NN algorithm.
- Each continuous feature should be normalized separately from all other features. Specifically, for both training and testing instances, each feature should be transformed using function $F(X) = (X - mean)/std$, using the mean and std of the values of that feature on the training data.

Submission Instructions

Homework must be submitted electronically through Canvas website on/before the due date/time. Homework must be typed with LaTeX, or a Python IDE, or Word. The code can be submitted as .py file or .ipynb file. Late homeworks will not be accepted unless with legitimate excuses with documents. Do NOT use models in scikit-learn package directly in the homeworks.