Project 1: Supervised Learning Using Baysian Decision Rule - Two Category Classification (Due 09/07)

Objective:

The objective of this project is, first of all, to learn how to implement supervised learning algorithms based on Baysian decision theory. The second objective is to get you familiar with the design flow when applying machine learning algorithms to solve real-world problems. Some practical considerations include, for example, 1) the selection of the right pdf model to characterize the data distribution in the training set, 2) the selection of the right ratio of prior probability, 3) the different ways to evaluate the performance of the learning algorithm, and 4) how differently the same ML algorithm performs when applied to different datasets.

Data Sets:

Two datasets will be used from Ripley's Pattern Recognition and Neural Networks, both are 2-category classification problems. The first is a synthetic dataset with 2 features where there are about equal number of samples in each category. The second is a dataset for diabetes in Pima Indians with 7 features where the number of diabetic patients is much less than that of the normal patients.

- The synthetic dataset: synth.tr (the training set) and synth.te (the test set)

  > Preprocessing you need to do: remove the first row with any text editor. No need to write Python code for that. The labels are 0 and 1.

- The Pima dataset: pima.tr (the training set) and pima.te (the test set)

  > Preprocessing you need to do: 1) Remove the first row with any text editor; 2) Change the labels from "Yes" and "No" to "0" and "1" respectively indicating 'with disease' and 'without disease' - you can do this using the same text editor; and 3) Normalize the data set to make the features comparable (or with the same scale). Suppose x is a data sample, m_i is the mean of each feature i, sigma_i is the standard deviation of each feature i, then normalization is conducted by (x-m_i)/sigma_i. Keep in mind that you also need to normalize the samples in the test set. Be careful which mean and standard deviation you should use. (For each sample in the test set, use the same m_i and sigma_i you derived from the training set.)

Performance Metrics:

Three metrics are used to evaluate the performance of the ML algorithms, including 1) overall classification accuracy, 2) classwise classification accuracy, and 3) run time.

Tasks:

- (5 pts) Show a scatter plot of the training set of the two classes using only the synthetic dataset. From visual inspection, do you think single-modal Gaussian is a good/reasonable model for the pdf?
- (50 pts) Assuming equal prior probability, generate two tables (Tables 1 and 2) summarizing the overall classification accuracy, classwise accuracy, and run time of the three learning algorithms applied on the two datasets, with each row indicating a learning algorithm and each column

indicating a performance metric.

- (10 pts) Provide a comprehensive discussion (0.5 ~ 1 page) on the results shown in the tables, including the effect of using different assumptions of the covariance matrices and different datasets.
- (15 pts) Using only the synthetic dataset, illustrate the three decision boundaries from the three cases of parametric learning algorithms on the same figure as the scatter plot of the testing dataset. Comment on the differences.
- (15 pts) Plot a figure with 3x2 subplots of class-0 accuracy vs. ratio of prior probability on the two datasets of the three cases of MAP classifiers and provide comments and discussions.
- (5 pts) Final discussion.
- Bonus (+10 pts): For the synthetic dataset, if single-modal Gaussian is not the best model for the pdf, revise the model and report the performance.