University of Pittsburgh
CS1675 Intro to Machine Learning — Handout 8
Professor Milos Hauskrecht — October 26, 2017

Problem assignment 7

Due: Thursday, November 2, 2017

In this problem we shall continue our investigation of the "Pima" dataset using new classification models: multilayer neural networks, decision trees, and the nearest neighbor classifier.

You can download the dataset (*pima.txt*) and its description (*pima_desc.txt*) from the course web page. In addition to the complete dataset *pima.txt*, you have *pima_train.txt* and *pima_test.txt* you will need to use for training and testing purposes. The dataset has been obtained from the UC Irvine machine learning repository:
$http://www1.ics.uci.edu/\sim mlearn/MLRepository.html$.

Problem 1. Neural network toolbox in Matlab

We start with the neural network toolbox.

- Part a. In homework 5 you were asked to run a gradient algorithm for learning the logistic regression model. However, the logistic regression model is also supported and implemented in Matlab within its Neural Network toolbox. Please familiarize yourself and run *logistic_NN.m* function that is given to you and implements the logistic regression model using the toolbox functions. Try to change the parameters of the model, such as the optimization method and the number of epochs. Report the weights with the best mean misclassification rate for the test set and any graphs you have found interesting.
- Part b. Multilayer neural network. The limitation of the logistic regression model is that it uses a linear decision boundary. One way around this is problem is to use non-linear features in combination with a linear model. However, in this case feature function must be fixed and selected in advance. Multilayer neural networks allow us to represent non-linear models by cascading multiple nonlinear units. Multilayer neural networks can be built with the NN matlab toolbox. Write a program *main1.m* that implements a neural network with two hidden units, that is, there are two nonlinear units we feed the input to, and one unit that combines their results. Run the program for 2000 epochs. Calculate the mean misclassification errors for the training and testing data. Report errors and compare them to results obtained for the logistic regression model for Part a. Which model is better? Why?

- Part c. Experiment with neural networks with 2, 3, 5 and 10 hidden units, while changing other learning parameters, e. g. the optimization method or the number of epochs. Analyze and compare and report the results.

Problem 2. Decision trees

The decision tree approach is yet another classification methods we covered in the course is the decision tree method. The method builds a tree by recursively splitting the training set using one of the attributes by optimizing the gain with respect to some impurity measure.

- Part a. The script *run_DT.m* shows how to train, display, and apply the decision tree. The script first builds a default tree with minimal restrictions on its size, and after that the tree obtained by restricting the number of nodes in the tree. Please run and familiarize yourself with the code. What do you think, which tree is better for prediction, the unrestricted or restricted tree? Why? Should we always try to backprune it?
- Part b. Experiment with the decision tree function fitctree.m and its optional parameters, modifying the algorithm and the tree built. Report the results of your investigations in the report by listing the settings used for the tree learning algorithm and obtained results. You can find the different settings in the matlab help documents.

Problem 3. K-nearest neighbor classifier

Another classification method covered in the course is the k nearest neighbor classifier (kNN).

- Part a. The script *run_kNN.m* shows how to classify examples with the knnclassify method implemented in Matlab. The script uses Euclidean metric and the number of neighbors is set to 3. Please run and familiarize yourself with the code. Please experiment with kNN by modifying the number of neighbors and report the results on the test set. Please attempt 1 and 5 neighbors in addition to 3.
- Part b. Please normalize the data (both the train and test set) before running the kNN classifier with the Euclidean distance again. Did the result improve?