COMP 7745/8745: Machine Learning

Instructor: Deepak Venugopal

Fall 2017: Homework 2

Due Date: February 28, 2017 Q1 - Q3 (Hard copy due before the start of class)

Due Date: March 14, 2017 Q5 (Code and answers in soft copy on ecourseware)

1. Naive Bayes (15 points). Given the below dataset,

W	**X**	**Y**	**Class**
T	T	T	T
T	F	T	F
T	F	F	F
F	T	T	F
F	F	F	T

   - Use Naive Bayes to classify the test example, (T, T, F).
   - Suppose I duplicate the last example in the training set 100 times, how will your classification change?

2. Give short answers to the following (25 points)

   - Given a dataset that is not linearly separable, (1) Will logistic regression converge? and (2) Can logistic regression classify all training examples correctly? Give a brief explanation of your answer.
   - State true or false. The 5-Nearest Neighbor algorithm is guaranteed to have no training error. Give a brief explanation of your answer.
   - Which of the following has high bias: (a) logistic regression on linearly-separable data (b) logistic regression on non-linearly-separable data (c) K-NN on non-linearly separable data. Give a brief explanation.
   - Suppose I write a new version of Adaboosting where the probability distribution over the training examples remain unchanged in each iteration. Do you think this will be an effective algorithm? Briefly explain your answer.
   - Can Naive Bayes handle noisy data? That is, in two different training examples, the features values are exactly the same but the labels are different? Briefly explain.

3. You design a new algorithm to retrieve sports pages based on content of the webpage. Out of 150 examples that you tested, If your algorithm returned 50 sports webpages and 100 webpages that are not relevant to sports, what is the precision and recall of your algorithm? (10 points)

4. In this question, you will experiment with logistic regression, Naive Bayes classifier K-NN (with K=3) implementations in Weka on imbalanced data (50)

   You will use the dataset imbalanced.csv. The imbalanced.csv has a far greater number of negative examples (label N) than positive examples (label P). We will refer to the positive examples as minority instances and negative examples as majority instances.

   - We will subsample the majority instances in the imbalanced.csv dataset (reduce the number of negative examples). Use the SpreadSubsampling algorithm implementation of Weka to do this. Specifically, subsampling the instances with larger number of examples will allow us to adjust the ratio of negative to positive examples. In Weka, the distributionspread parameter (M) of SpreadSubsampling allows you to do this. Vary the ratio as $10 : 1, 9 : 1, ..., 1 : 1$. Report the 10-fold cross validated F1-score only for the positive class (label P) for each case. Thus, you will have 30 cases to run (you need to automate this!). Report the results in a table. Briefly explain your results and findings.

     I have changed the earlier WekaRun.java starter code to show you an example of how to call subsampling.