CSC2515 Fall 2010
Assignment 1
Due: Oct. 15, 2010, by 3 pm

October 3, 2010

Instructions

There are several questions on this assignment; only the last question involves coding. Do not attach your code to the writeup. Instead, email your implementation, and your write-up to csc2515ta@cs.

Late assignments will have 25% subtracted from the total out of which they are graded for each day or part of a day that they are late. They will be more than one day late if they are not emailed within 24 hours of the due date/time.

1 Introduction

Please write a few sentences describing your background, and why you are interested in machine learning.

2 Extending $K$-nearest neighbors (10 points)

The choice of $k$ is somewhat arbitrary, but important to the KNN algorithm. One possible solution to this problem is to take into account the distances of the neighbors to the query point in their contribution to the classifier. Describe how you could modify $k$-nearest neighbors to do this, both in words and equations. Can you use this approach to remove the dependence on the parameter $k$? Describe situations in which you expect this method to do well and not do well.
3 Naive Bayes and Logistic Regression

1. (5 points) Briefly describe the functions Naive Bayes (NB) and Logistic Regression (LR) optimize.

2. (10 points) The parametric form of $P(Y|X)$ used by LR is implied by the assumptions of a NB classifier, for some specific class-conditional densities. Prove this for a Gaussian NB classifier for continuous input values.

3. (15 points) Consider the two class problem where class label $y \in \{T, F\}$ and each training example $X$ has 2 binary attributes $X_1, X_2 \in \{T, F\}$. Let the class prior be $P(Y = T) = 0.5$ and also let $P(X_1 = T|Y = T) = 0.8$ and $P(X_1 = F|Y = F) = 0.7$, $P(X_2 = T|Y = T) = 0.5$ and $P(X_2 = F|Y = F) = 0.9$. So, attribute $X_1$ provides slightly stronger evidence about the class label than $X_2$. For this problem, you should assume that the true distribution of $X_1, X_2, Y$ satisfies the Naive Bayes assumption of conditional independence with the above parameters.

   (a) Assume $X_1$ and $X_2$ are truly independent given $Y$. Write down the Naive Bayes decision rule given $X_1 = x_1$ and $X_2 = x_2$.

   (b) Show that if Naive Bayes uses both attributes, $X_1$ and $X_2$, the error rate is $0.235$. Is it better than using only a single attribute ($X_1$ or $X_2$)? Why? The error rate is defined as the probability that each class generates an observation where the decision rule is incorrect.

   (c) Now, suppose that we create a new attribute $X_3$, which is an exact copy of $X_2$. So, for every training example, attributes $X_2$ and $X_3$ have the same value, $X_2 = X_3$. Are $X_2$ and $X_3$ conditionally independent given $Y$? What is the error rate of Naive Bayes now? [Hint: The true distribution has not changed.]

   (d) Explain what is happening with Naive Bayes. Does Logistic Regression suffer from the same problem? Explain why or why not.

4 Experimenting with Logistic Regression and $K$-Nearest Neighbors

For this part you will compare $K$-Nearest Neighbors (KNN) and Logistic Regression (LR) on a data set you can download from the course website. We have provided you with an implementation of a $k$ nearest neighbor classifier, and a template for a Logistic Regression classifier. You will need to fill in the remaining components of the LR classifier, including an L2 regularizer. Use checkgrad.m to make sure that your gradients are correct.

The dataset contains some images of handwritten digits, where there are three classes, corresponding to the three digits: 4, 6 and 9. You’ll see that the dataset has been split into three separate sets. The training data for the digits has 30 examples of each digit; the
validation data has 100 examples of each digit; and the test data has approximately 1000 examples of each class.

For LR, you will need to choose some parameters. Reasonable parameters for training are: weight initialization with 0.01*randn; a learning rate between 0.0005 and 0.005; and an L2 penalty coefficient of about 100.

You should submit, for each algorithm:

1. (10 points) A brief description of your implementation (pseudocode). The description must include equations for estimating the classification parameters and for classifying a new example. This should not be a printout of your code, but a high-level outline.

2. (15 points) Train the weights in logistic regression with no regularization for 200-500 iterations. Plot training and validation curves. Explain how you can stop training based on validation error. Compare the performance of the system if you stop based on the training error versus the validation error.

3. (15 points) Add the regularization, and again plot the training and validation curves. Compare your results to those you obtained without the regularizer. Explain the effects of the regularization.

4. (15 points) Now try \(k\)-nearest neighbor on the same dataset. Experiment with different values of \(k\), and describe the effect of these changes. Implement the \(k\)-free method you described above, and experiment with this as well.