Assignment 2

CS780/880: Introduction to Machine Learning

Due: By 12:40PM Tue Feb 21st, 2017
Submission: Turn in as a PDF on myCourses, or printed and turned in at class
Discussion forum: https://piazza.com/unh/spring2017/cs780cs880

Problem 1 [15%] In logistic regression, the probability is predicted using the *logistic function*:

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}$$

Using simple algebraic manipulation, show the equivalence to *odds*:

$$\frac{p(X)}{1-p(X)} = e^{\beta_0+\beta_1 X}$$

Problem 2 [20%] This problem relates to the QDA model, in which the observations within each class are drawn from a normal distribution with a class-specific mean vector and a class specific covariance matrix. We consider the simple case where $p = 1$; i.e. there is only one feature. Suppose that we have K classes, and that if an observation belongs to the k-th class then X comes from a one-dimensional normal distribution, $X \sim \mathcal{N}(\mu_k, \sigma_k^2)$. Recall that the density function for the one-dimensional normal distribution is given in equation (4.11) in ISL. Prove that in this case, the Bayes' classifier is **not** linear. Argue that it is in fact quadratic.
Hint: For this problem, you should follow the arguments laid out in ISL Section 4.4.2, but without making the assumption that $\sigma_1^2 = \ldots = \sigma_K^2$.

<u>CS880 Graduate</u>: Problem 3 [35%] When the number of features p is large, there tends to be a deterioration in the performance of KNN and other local approaches that perform prediction using only observations that are near the test observation for which a prediction must be made. This phenomenon is known as the *curse of dimensionality*, and it ties into the fact that non-parametric approaches often perform poorly when p is large. We will now investigate this curse.

(a) Suppose that we have a set of observations, each with measurements on $p = 1$ feature, X. We assume that X is uniformly (evenly) distributed on $[0, 1]$. Associated with each observation is a response value. Suppose that we wish to predict a test observationâĂŹs response using only observations that are within 10% of the range of X closest to that test observation. For instance, in order to predict the response for a test observation with X = 0.6, we will use observations in the range $[0.55, 0.65]$. On average, what fraction of the available observations will we use to make the prediction?

(b) Now suppose that we have a set of observations, each with measurements on $p = 2$ features, X_1 and X_2. We assume that $(X1, X2)$ are uniformly distributed on $[0, 1] \times [0, 1]$. We wish to predict a test observationâĂŹs response using only observations that are within 10% of the range of X_1 and within 10% of the range of X_2 closest to that test observation. For instance, in order to predict the response for a test observation with $X_1 = 0.6$ an $X_2 = 0.35$, we will use observations in the range $[0.55, 0.65]$ for X_1 and in the range $[0.3, 0.4]$ for X_2. On average, what fraction of the available observations will we use to make the prediction?

(c) Now suppose that we have a set of observations on $p = 100$ features. Again the observations are uniformly distributed on each feature, and again each feature ranges in value from 0 to 1. We wish to predict a test observationâĂŹs response using observations within the 10% of each featureâĂŹs range that is closest to that test observation. What fraction of the available observations will we use to make the prediction?

(d) Using your answers to parts (a)âĂŞ(c), argue that a drawback of KNN when p is large is that there are very few training observations âĂIJnearâĂİ any given test observation.

(e) Now suppose that we wish to make a prediction for a test observation by creating a p-dimensional hypercube centered around the test observation that contains, on average, 10% of the training observations. For $p = 1, 2$, and 100, what is the length of each side of the hypercube? Comment on your answer.

Note: A hypercube is a generalization of a cube to an arbitrary number of dimensions. When $p = 1$, a hypercube is simply a line segment, when $p = 2$ it is a square, and when $p = 100$ it is a 100-dimensional cube.

CS780 Undergraduate: Problem 3 [35%] Suppose that we take a data set, divide it into equally-sized training and test sets, and then try out two different classification procedures. First we use logistic regression and get an error rate of 20% on the training data and 30% on the test data. Next we use 1-nearest neighbors (i.e. $K = 1$) and get an average error rate (averaged over both test and training data sets) of 18%. Based on these results, which method should we prefer to use for classification of new observations? Why?

Problem 4 [30%] This question should be answered using the Weekly data set, which is part of the ISLR package. This data is similar in nature to the Smarket data from this chapterâĂŹs lab, except that it contains 1 089 weekly returns for 21 years, from the beginning of 1990 to the end of 2010.

(a) Produce some numerical and graphical summaries of the Weekly data. Do there appear to be any patterns?

(b) Use the full data set to perform a logistic regression with Direction as the response and the five lag variables plus Volume as predictors. Use the summary function to print the results. Do any of the predictors appear to be statistically significant? If so, which ones?

(c) Compute the confusion matrix and overall fraction of correct predictions. Explain what the confusion matrix is telling you about the types of mistakes made by logistic regression.

(d) Now fit the logistic regression model using a training data period from 1990 to 2008, with Lag2 as the only predictor. Compute the confusion matrix and the overall fraction of correct predictions for the held out data (that is, the data from 2009 and 2010).

(e) Repeat (d) using LDA.

(f) Repeat (d) using QDA.

(g) Repeat (d) using KNN with K = 1.

(h) Which of these methods appears to provide the best results on this data?

(i) Experiment with different combinations of predictors, including possible transformations and interactions, for each of the methods. Report the variables, method, and associated *confusion matrix* that appears to provide the best results on the held out data. Note that you should also experiment with values for K in the KNN classifier.