CSL407 Machine Learning
Homework 1

Due on 15/8/2014, 11.59pm

Instructions: Upload to your moodle account one zip file containing the following. Please do not submit hardcopy of your solutions. In case moodle is not accessible email the zip file to the instructor at ckn@iitrpr.ac.in. Late submission is not allowed without prior approval of the instructor. You are expected to follow the honor code of the course while doing this homework.

1. A neatly formatted PDF document with your answers for each of the questions in the homework. You can use latex, MS word or any other software to create the PDF.
2. Include a separate folder named as 'code' containing the scripts for the homework along with the necessary data files. Name the scripts using the problem number.
3. Include a README file explaining how to execute the scripts.
4. Name the ZIP file using the following convention rollnumber_hwnumber.zip

---

In this homework you will be implementing linear ridge regression and solving few other problems to gain a better understanding on linear regression.

1. (32 points)In this problem you will be implementing Linear Regression to predict the age of Abalone (is a type of snail). You can read more about the dataset at the UCI repository (http://archive.ics.uci.edu/ml/datasets/Abalone).
   - Load the dataset into the Matlab workspace. We are primarily interested in predicting the last column of the data that corresponds to the age of the abalone using all the other attributes.
   - The first column in the data encodes the attribute that encodes-female, infant and male as 0, 1 and 2 respectively. Though numbers have been used to represent the attribute, these are only symbols and therefore are not ordered. Transform this attribute into a three column binary representation. For example, represent female as (1, 0, 0), infant as (0, 1, 0) and male as (0, 0, 1).
   - Before performing linear regression, we must first standardize the independent variables, which includes everything except the last attribute (target attribute) - the number of rings. Standardizing means subtracting each attribute by its mean and dividing by its standard deviation. Standardization will transform the attributes to possess zero mean and unit standard deviation. You can use this fact to verify the correctness of your code.
   - Implement the function named mylinridgereg(X, T, lambda) that calculates the linear least squares solution with the ridge regression penalty parameter lambda (λ) and returns the regression weights. Implement the function mylinridgeregeval(X, weights) that returns a prediction of the target variable given the input variables and regression weights. Before applying these functions to the dataset, randomly partition the data into a training and test set. Let's refer to the partition fraction as frac. If we want to use a 20%/80% training/testing split, then the value of frac will be 0.2. Now use your mylinridgereg with a variety of λ values to fit the penalized linear model to the training data and predict the target variable for the training and also for the testing data using two calls to your mylinridgeregeval function.

- Implement the function meansquarederr(T, Tdash) that computes the mean squared error between the predicted and actual target values. Store the mean squared error of the regression function on the training and test splits in a variable. We will use it later for visualizations.
- Pick a value for λ and examine the weights of the ridge regression model. Which are the most significant attributes? Try removing two or three of the least significant attributes and observe how the mean squared errors change.
- Let us now try to answer some questions
  - Does the effect of λ on error change for different partitions of the data into training and testing sets?
  - How do we know if we have learned a good model?

  To answer these questions, modify your code to perform the following steps
  - For different training set fractions, repeat 100 times
    (a) Randomly divide data into training and testing partitions.
    (b) Standardize the training input variables.
    (c) Standardize the testing input variables using the means and standard deviations from the training set.
    (d) For different values of lambda
      i. Fit a linear model to the training data for the given lambda
      ii. Use it to predict the number of rings in the training data and calculate the mean squared error (MSE)
      iii. Do this again, using the same linear model applied to the testing data.
  - Calculate the average mean squared error over the 100 repetitions for each combination of training set fraction and lambda value
- To see if the training set fraction affects the effect of lambda on error, plot the effect in multiple graphs, one for each training set fraction, by building the following figure. Make one figure of multiple graphs, one for each training set fraction, each graph being a plot of the average mean squared training error versus λ values and a plot of the average mean squared testing error versus λ. To enable the comparison across graphs, force each graph to have the same error (y axis) limits. You will find subplot, plot, hold on and ylim Matlab functions useful for plotting these graphs.
- The figures provide some insight, but is not very clear right? So let us draw two more graphs. In the first graph plot the minimum average mean squared testing error versus the training set fraction values. In the second graph, plot the λ value that produced the minimum average mean squared testing error versus the training set fraction.
- So far we have been looking at only the mean squared error. We might also be interested in understanding the contribution of the each prediction towards the error. Maybe the error is due to a few samples with huge errors and all others have tiny errors. One way to visualize this information is to a plot of predicted versus actual values. Use the best choice for the training fraction and λ, make two graphs corresponding to the training and testing set. The X and Y axis in these graphs will correspond to the predicted and actual target values respectively. If the model is good then all the points will be close to a 45 degree line through the plot.
- Include all the plots and your observations in the report.

2. (4 points)I have collected a set of data (N=50 observations) containing a single attribute and a target response. I then fit a linear regression model to the data, as well as a separate quartic regression, i.e. $y(x) = w_0 + w_1 x + w_2 x^2 + w_3 x^3 + w_4 x^4$

   (a) Suppose that the true relationship between X and T is linear, i.e. $t = w_0 + w_1 x + \epsilon$. Consider the training residual sum of squares (RSS) for linear regression and quartic regression. Would we expect one to lower than the other, would we expect them to be the same, or is there not enough information to tell? Justify your answer.

(b) Answer (a) using test rather than training RSS.

(c) Suppose that the true relationship between X and T is not linear, but we don't know how far is it from linear. Consider the training RSS for linear and quartic regressions. Would we expect one to be lower than the other, would we expect them to be the same, or is there no enough information to tell? Justify your answer.

(d) Answer (c) using test rather than training RSS.

3. (4 points)Consider a data set in which each data point t_n is associated with a weighting factor $r_n > 0$, so that the sum-of-squares error function becomes

$$E_D(w) = \frac{1}{2}\sum_{n=1}^{N} r_n\{t_n - w^T x_n\}^2$$

Find an expression for the solution $\hat{w}$ that minimizes this error function.