Assignment 3

CS6510: Applied Machine Learning
IIT-Hyderabad
Aug-Nov 2018

Max Points: 35
Due: 26 Nov 2018 11:59 pm

This homework is intended to cover programming exercises in the following topics:

- Regression, Clustering

Instructions

- Please use Google Classroom to upload your submission by the deadline mentioned above. Your submission should comprise of a single file (PDF/ZIP), named `<Your_Roll_No>_Assign3`, with all your solutions.
- For late submissions, 10% is deducted for each day (including weekend) late after an assignment is due. Note that each student begins the course with 15 grace days for late submission of assignments. Late submissions will automatically use your grace days balance, if you have any left. You can see your balance on the `CS6510 Marks and Grace Days` document under the course Google drive.
- We recommend using PYTHON for the programming assignments, although you can use Java/C/C++/R if you like too.
- Please read the department plagiarism policy. Do not engage in any form of cheating - strict penalties will be imposed for both givers and takers. Please talk to instructor or TA if you have concerns.

1 Questions

1. **(15 points)** We will now implement Linear Regression to predict the age of Abalone (a type of snail). The data set is made available as part of the provided zip archive (linregdata). You can read more about the dataset at this UCI repository link. We are interested in predicting the last column of the data that corresponds to the age of the abalone using all the other attributes.

   (a) The first column in the data denotes the attribute that encodes-female, infant and male as 0, 1 and 2 respectively. The numbers used to represent these values are symbols and

therefore should not be ordinal. 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). [*0.5 points*]

(b) Before performing linear regression, we must first standardize the independent variables, which includes everything except the last attribute (target attribute). 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. [*0.5 points*]

(c) Implement the following functions: (i) `mylinridgereg(X, Y, λ)` that calculates the linear least squares solution with the ridge regression penalty parameter (λ) and returns the regression weights; (ii) `mylinridgeregeval(X, weights)` that returns a prediction of the target variable given the input variables and regression weights; and (iii) `meansquarederr(T, Tdash)` that computes the mean squared error between the predicted and actual target values. [*6 points*]

(d) Partition the dataset into 80% training and 20% testing (Let's call this the partition fraction, in this case 0.2). Now, use your `mylinridgereg` with different λ values to fit the penalized linear model to the training data and predict the target variable for both training and testing data. [*1 point*]

(e) Identify the λ with the best performance 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. [*1 point*]

(f) We now would like to ask the question: Does the effect of λ on error change for different partitions of the data into training and test sets? To do this, change the partition fraction (a value between 0 and 1, as defined earlier) with at least 4 other values. Repeat the following steps 25 times for each partition fraction:

- Randomly divide data into training and test sets.
- Standardize the training input variables, and standardize the testing input variables using the means and standard deviations from the training set.
- Follow step (d) for each such partition.

For each partition fraction, plot a figure with λ on the x-axis, and MSE on the y-axis. For each figure, include 2 graphs - one for the training MSE and one for the test MSE. (You should then have 5 figures in total, with 2 plots on each figure.) [*3 points*]

(g) Do the above figures give you clarity? Also, plot two more figures. In the first graph, plot the minimum average mean squared testing error versus the partition fraction values. In the second graph, plot the λ value that produced the minimum average mean squared testing error versus the partition fraction. [*1 point*]

(h) How good is your model? So far, we have been looking at only the mean squared error. We might also be interested in understanding the contribution of each prediction towards the error. Maybe the error is due to a few samples with large errors and 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 axes 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. [*2 points*]

Include all the plots and your observations in your submission.

2. **(14 points)** DBSCAN is density based clustering algorithm. In this question you need to implement your own DBSCAN algorithm. You can read more about it from paper that proposed this method [link]

   (a) Use the Kmeans clustering algorithm from python's sklearn package and find the number of clusters in dataset1. Plot the data points with different colors for different clusters. [*3 points*]

   (b) Implement your own DBSCAN algorithm on the same dataset1 and plot the data points. [*6 points*]

   (c) What differences do you see between the DBSCAN and k-means methods, and why? [*2 points*]

   (d) Consider dataset2 with three clusters. Use (a) and (b) for dataset2, and compare the performance. List your observations clearly, and make conclusions on pros and cons of DBSCAN and k-means. [*3 points*]

3. **(6 points) Hierarchical Clustering:** Given below is the distance matrix for 6 data points

	x_1	x_2	x_3	x_4	x_5	x_6
x_1	0					
x_2	0.12	0				
x_3	0.51	0.25	0			
x_4	0.84	0.16	0.14	0		
x_5	0.28	0.77	0.70	0.45	0	
x_6	0.34	0.61	0.93	0.20	0.67	0

   (a) Draw a dendrogram for the final result of hierarchical clustering with single link. [*2 points*]

   (b) Draw a dendrogram for the final result of hierarchical clustering with complete link. [*2 points*]

   (c) Change two values from the matrix so that the answer to the last two questions is same. [*2 points*]

2 Theory and Practice Questions (No submission required)

The questions below are only for your practice - no submission required.

1. **Logistic Regression:**

   (a) Plot the sigmoid function $1/(1+e^{-w\mathbf{x}})$ vs $\mathbf{x} \in \mathbb{R}$ or increasing weight $w \in \{1, 5, 100\}$. A qualitative sketch is enough. Use these plots to argue why a solution with large weights can cause logistic regression to overfit.

(b) To prevent overfitting, we want the weights to be small. To achieve this, instead of maximum likelihood estimation MLE for logistic regression:

$$\max_{w_0,\cdots,w_d} \prod_{i=1}^{n} P(Y_i|X_i, w_0, \cdots, w_d)$$

we can consider maximum a posterior (MAP) estimation:

$$\max_{w_0,\cdots,w_d} \prod_{i=1}^{n} P(Y_i|X_i, w_0, \cdots, w_d)P(w_0, \cdots, w_d)$$

where $P(w_0, \cdots, w_d)$ is a prior on the weights. Assuming a standard Gaussian prior $\mathcal{N}(0, I)$ for the weight vector (I = Identity matrix), derive the gradient ascent update rules for the weights.

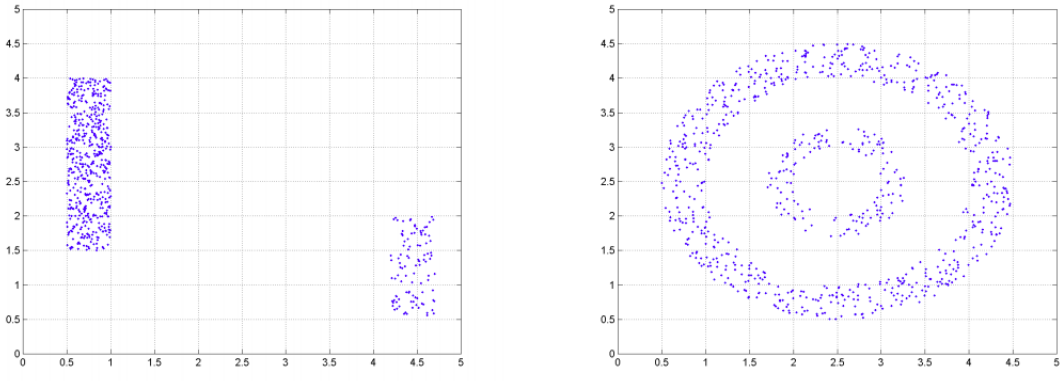
(c) One way to extend logistic regression to multi-class (say, K class labels) setting is to consider $(K-1)$ sets of weight vectors and define:

$$P(Y = y_k|X) \propto \exp(w_{k0} + \sum_{i=1}^{d} w_{ki}X_i) \text{ for } k = 1, \cdots, K-1$$

What model does this imply for $P(Y = y_k|X)$? What would be the classification rule in this case?

(d) Draw a set of training data with three labels and the decision boundary resulting from a multi-class logistic regression. (The boundary does not need to be quantitatively correct but should qualitatively depict how a typical boundary from multi-class logistic regression would look like.)

2. **Spectral Clustering:** In this problem we will analyze the operation of one of the variants of spectral clustering methods on two datasets shown in the adjoining figure. For each of the datasets, please answer the following questions.



(a) (b)

(a) The first step is to build an affinity matrix. The matrix defines the degree of similarity between points. Suppose we use the L_2 norm to construct the following affinity matrix (let x_i denote an ith datapoint):

$$A(i,j) = A(j,i) = \begin{cases} 1 & \text{if } |x_i - x_j|_2 < \theta \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

What θ value would you choose and why?

(b) Suppose instead we use Gaussian kernel for our affinity matrix:

$$A(i,j) = \exp(\frac{-|x_i - x_j|_2}{2\sigma^2}) \tag{2}$$

What σ value would you choose and why?

(c) The second step is to compute first k dominant eigenvectors of the affinity matrix, where k is the number of clusters we want to have. For the dataset in Figure 1(a) and the affinity matrix defined by equation 1, is there a value of θ for which you can compute analytically eigenvalues corresponding to the first two dominant eigenvectors? If not, explain why not. If yes, compute and write these eigenvalues down.

(d) The third step is to cluster the rows of the matrix Y into k clusters using K-means (or a similar algorithm), where Y is constructed by placing k dominant eigenvectors into columns and re-normalizing the rows (to make each row a unit vector). For the dataset in Figure 1(a) and the affinity matrix defined by equation 1, write down your best guess for the coordinates of $k = 2$ cluster centers.

(e) Finally, given the clusters on matrix Y, a point x_i is declared to be in cluster j iff the ith row of Y is in cluster j. What are the final clusters you would expect to obtain for each of the datasets? Provide a rough sketch of the clusters to give an idea.