Assignment 1 (Linear regression, regularization, logistic regression, multi-class classification, probabilistic classifiers)

Submission due date: Sept 26, 2020

1. Implement the linear regression algorithm to estimate the weight parameters. You can use batch gradient descent algorithm for the implementation. (a) Plot the cost function vs the number of iterations. (b) Plot the cost function (J) vs w1 and w2 in a contour or 3D surf graph (w= [w0 w1 w2]). (c) Evaluate the mean square error between the actual test output and predicted test output. Please use the files such as Training feature matrix, training output vector, test feature matrix, test output vector for this question (Use for or while loop for the implementation of linear regression).

2. Implement mini-batch and stochastic gradient descent algorithms for the linear regression problem in question number 1. (a) Plot the cost function vs the number of iterations. (b) Plot the cost function vs w1 and w2. (c) Evaluate the mean square error between the actual test output and predicted test output. Please use the same data files as in question number 1 (Use for or while loop for the implementation).

3. Implement the ridge regression problem by considering batch, mini-batch and stochastic gradient descent algorithms. (a) Plot the cost function vs the number of iterations for all three cases. (b) Plot the cost function (J) vs w1 and w2 in a contour or 3D surf graph for all three cases. (c) Evaluate the mean square error between the actual test output and predicted test output. Please use the same data files as in question number 1 (Use for or while loop for the implementation).

4. Derive the least angle regression (LAR) weight update rule. Clearly show all steps (cost function of LAR, evaluation of the derivative of the cost function, and weight update rule). Upload the derivations of this question in the assignment 1 part of the Google classroom.

4. Implement Least angle regression to estimate the weight parameters for the feature matrix (X) and the class label vector (y) by considering batch, mini-batch and stochastic gradient descent-based algorithms. (a) Plot the cost function vs the number of iterations for all three cases. (b) Plot the cost function (J) vs w1 and w2 in a contour or 3D surf graph for all three cases. (c) Evaluate the mean square error between the actual test output and predicted test output. Please use the same data files as in question number 1 (Use for or while loop for the implementation).

5. Implement the vectorized linear regression, ridge regression and least angle regression problems to evaluate the weight parameters for question number 1. Compare the weight parameters of vectorization-based methods with the weights obtained using batch, mini-batch and stochastic gradient descent-based algorithms. Please use the same data files as

in question number 1. Evaluate the mean square error between the actual test output and predicted test output.

6. Implement K-means clustering based unsupervised learning algorithm for the dataset ("data2.xlsx"). Plot the estimated class labels vs features. Use the number of clusters as K=3.

7. Implement the logistic regression algorithm for the binary classification using the dataset ("data3.xlsx"). Divide the dataset into training and testing using hold-out cross-validation technique with 60 % of instances as training and the remaining 40% as testing. Evaluate the accuracy, sensitivity and specificity values for the binary classifier.

8. Implement the multiclass logistic regression algorithm using both "One VS All" and "One VS One" multiclass coding techniques. Evaluate the performance of the multiclass classifier using individual class accuracy and overall accuracy measures. Use the hold-out cross-validation approach (60% training and 40% testing) for the selection of training and test instances. (Please use the dataset "data4.xlsx")

9. Evaluate the performance of multiclass logistic regression classifier using 5-fold cross-validation approach. Evaluate the individual class accuracy and overall accuracy measures for the multiclass classifier along each fold. (Please use the dataset "data4.xlsx")

10. Use the likelihood ratio test (LRT) for the binary classification using the dataset ("data3.xlsx"). Divide the dataset into training and testing using hold-out cross-validation technique with 60 % of instances as training and the remaining 40% as testing. Evaluate the accuracy, sensitivity and specificity values for the binary classifier.

11. Implement the Maximum a posteriori (MAP) decision rule for multiclass classification task. Use the hold-out cross-validation approach (70% training and 30% testing) for the selection of training and test instances. (Please use the dataset "data4.xlsx").

12. Implement the Maximum likelihood (ML) decision rule for multiclass classification task. Use the hold-out cross-validation approach (70% training and 30% testing) for the selection of training and test instances. (Please use the dataset "data4.xlsx").

13. Please write in your own words that what you have learned by solving the Assignment 1.

Dataset description

Training_feature_matrix.mat: contains feature vectors for all training instances.
Training_output.mat: contains output for all training instances
Test_feature_matrix.mat: contains feature vectors for all test instances

Test_output.mat: contains output for all test instances (for evaluating mean square error only)

data4.xlsx: (This dataset contains 4 features for different instances as four columns, fifth column is for class label)
data3.xlsx: (This dataset contains 4 features for different instances as four columns, fifth column is for class label)
data2.xlsx: (This dataset contains 4 features for different instances as four columns)

packages used in python for assignment 1

You can use pandas, NumPy, Matplotlib, and Scipy (only for matrix part)

Keras, scikit-learn, tensorflow etc. not allowed. If you use these packages, then questions will not be checked.