Assignment 2

Due Date: 10/08/2019

Total Points: 150

In this exercise, you will implement linear regression and logistic regression ***from scratch*** using the programming language of your choice (**without using a toolbox from R, Matlab, Python or any other programming language, please make sure you attach your code in the folder**). You will implement the Gradient Descent Algorithm that we have discussed in class to find out the parameters for $\ominus$. A good way to verify that gradient descent is working correctly is to look at the value of J($\ominus$) and check that it is decreasing with each iteration. For implementing some of the principles of programming, try to modularize the code as much as possible and consider **testing** your algorithm on a smaller known dataset before starting the assignment. Please note, that you can also use datasets such as here: http://data.princeton.edu/wws509/datasets to test your algorithm before testing it on the BSOM dataset. Assignment 2 contains five sections. Please address the subparts in **each** section to receive full credit. Analysis is a crucial aspect of the assignment, so for each subpart try to answer the question in more detail. Also, please divide the data into **training and test data** and use the **test case to evaluate performance**.

1. Linear Regression with One Variables (50 points):
    a. Can you demonstrate linear regression using 'all_mcqs_avg_n20' and 'STEP_1'? Note, here 'STEP_1' is the target variable.
    b. Evaluate performance using metrics (such as Mean Squared Error, Pearson correlation coefficient and R^2). You may also use graphs for explaining your observations.

2. Linear Regression with Two Variables (10 points):
    a. Does adding 'all_NBME_avg_n4' as input improve the performance of the previous model? Please use evaluation metrics or graphs to compare the performance of Question 1 and 2.

3. Logistic Regression with Multiple Variables (50 points):
    a. Can you demonstrate logistic regression using 'all_mcqs_avg_n20', 'all_NBME_avg_n4' and 'LEVEL'? Note, here 'LEVEL' is the target variable.
    b. Evaluate performance using metrics (such as confusion matrix, precision, recall and F1 scores). You may also use graphs for explaining your observations.

4. Regularization and Feature Scaling (20 points):
    a. Does Feature Scaling improve the performance for the model in Question 3?
    b. Does regularization improve the performance for the model in Question 3? Test at least 5 different regularization values to support your answer.
    c. Evaluate performance for each case using metrics (such as confusion matrix, precision, recall and F1 scores).
5. Build the best performance model (20 points):
    a. Select features from input variables 'all_mcqs_avg_n20', 'all_NBME_avg_n4', 'CBSE_01', and 'CBSE_02' to build the best performance logistic regression model to predict target variable 'LEVEL'. Also use feature scaling and regularization techniques if that improves the model performance.
    b. Compare model performances using metrics (such as confusion matrix, precision, recall and F1 scores).

Please make sure to submit a zipped file in Dropbox on Pilot titled YourName_Assignment 2 with the report in pdf format.

Academic Integrity

Discussion of course contents with other students is an important part of the academic process and is encouraged. However, it is expected that course programming assignments, homework assignments, and other course assignments will be completed on an **individual** basis (unless specified otherwise). Students may discuss general concepts with one another, but may not, under any circumstances, work together on the actual implementation of any course assignment. If you work with other students on "general concepts" be certain to acknowledge the collaboration and its extent in the assignment. Unacknowledged collaboration will be considered dishonest. "Code sharing" (including code from previous quarters) is strictly disallowed. "Copying" or significant collaboration on any graded assignments will be considered a violation of the university guidelines for academic honesty.

If the same work is turned in by two or more students, all parties involved will be held equally accountable for violation of academic integrity. You are responsible for ensuring that other students do not have access to your work: do not give another student access to your account, do not leave printouts in the recycling bin, pick up your printouts promptly, do not leave your workstation unattended, etc. If you suspect that your work has been compromised notify me immediately. If you have any questions about collaboration or any other issues related to academic integrity, please see me immediately for clarification. In addition to the policy stated

in this syllabus, students are expected to comply with the Wright State University Code of Student Conduct (http://www.wright.edu/students/judicial/conduct.html) and in particular the portions pertaining to Academic Integrity (http://www.wright.edu/students/judicial/integrity.html) at all times.