CSCI 390 Machine Learning – HW2

DUE: Friday, October 11 - 11:59PM

In this assignment you will be exploring the application of multilayer perceptron models to a pair of datasets that I have provided.

Part 1 (50 points)

In part 1, you will implement a simple regression MLP using sklearn, and will investigate the effects of network size and regularization.

First, develop a program to open, preprocess and apply a regression MLP to the dataset "`Census_Supplement.xlsx`" (the same one used in HW1). This time, you will be predicting the continuous quantity HDIVVAL, using AGI, A_AGE, A_SEX and WKSWORK.

Modify your code to collect the learning history on the training and test data (use `partial_fit()` inside a for loop and evaluate your network after each epoch). Finally, add code to capture the NN weights after each epoch (they are available in the classifier variable `coefs_`). Store the weights as a history that can be plotted by epoch. Each time you train the classifier, generate two plots: one of the loss on training and test sets versus epoch (the learning history) and one of the weights versus epoch (this plot is too busy to read unless you use a small network: four hidden nodes, perhaps). I found the curves were useful for 100 epochs.

Now, generate plots (and corresponding MSE metrics) for several ANNs: hidden nodes from 3 to 6, and L2 regularization coefficients of 0, 0.0001, 0.001 and 0.01. Put the plots into your homework submission and analyze them. What do you observe about the weights for various sizes and regularization settings? Is this what you expected?

Part 2 (50 points)

In part 2, you will apply a classification ANN to the dataset "iris". Use the sklearn function `sklearn.datasets.load_iris()` to stream the dataset without directly downloading it. Search the sklearn documentation to find proper usage of this dataset.

Experiment with network sizes and other parameters to find the maximum accuracy possible. What is it? Document each experiment and your findings.

Part 3 (50 points)

For part 3, you will be again working with the "`Census_Supplement,xlsx`" dataset. Use **TensorFlow** to develop a regression MLP to predict HDIVVAL from the other features (as listed in Part 1). You can use any network size that you wish. Find the best model, in your opinion, and justify your choice. Be sure to report the final model performance using suitable metrics and plot a learning history for your chosen model.

Tips

- While imputation and feature transformation may or may not be needed in this assignment, be sure and follow the recommended steps in model development and clearly indicate them using comments. If one of the steps is not used (feature transformation, for example), leave the comment in as a placeholder.
- There is sample code everywhere on the Web for many of these functions; you are welcome to use it if you clearly note which lines you borrowed and clearly list the source URL.
- When working with larger datasets, I often find it helpful to create a small version of the dataset for testing my code (it loads and runs much faster). The modeling results are NOT useful, because I am not randomly sampling from the larger file, but I can use it to debug my program. In this case, I copied the first 1000 rows of the spreadsheet into a file called "`Census_Supplement_1000.xlsx`" and used that for code development.

Submission

Your submission will consist of a Word or pdf file, containing the items below, and your source code for all problems. Your Word file should contain:

- Your plots and discussion for part 1
- Your model performance, architecture, and observations for part 2
- Your model performance, architecture, learning history plot and discussion for part 3
- Your complete Python code for parts 1-3