Christopher Gjersing
CS 422 1002 | Fall 2021
16 November 2021

# Assignment 5: MLP Regressor

The dataset I used for this assignment is the same as assignment 3 and comes from the University of California, Irvine Machine Learning Repository. The name of the dataset is "Auto MPG" and lists information about cars. Not all of the attributes are continuous real-valued, so I trimmed the dataset down to only MPG, displacement, horsepower, weight, and acceleration. Using these attributes, I decided to see if the multi-layer perceptron algorithms could predict the MPG of a car when provided with the other 4 attributes.

From assignment 3 I learned that the attributes may not correlate very effectively with the output, MPG, but want to compare the algorithms nonetheless. I used the standard scaler, just like assignment 3, to assist the algorithms. For my MLP experiments I changed the number of layers, neurons, activation function, solver, and learning rate for the algorithms. The differences are shown in the following table:

| Number | # Layers | # Neurons | Activation | Solver | Learning Rate |
|---|---|---|---|---|---|
| MLP #1 | 1 | 100 | Hyperbolic tan function (tanh) | sgd | Constant (0.01) |
| MLP #2 | 2 | 80, 40 | Identity (no-op activation) | sgd | invscaling |
| MLP #3 | 4 | 60,40,20,10 | Relu (rectified linear unit) | lbfgs | N/A |

The results and corresponding metrics for the three algorithms (plus Ast3 LR) are as follows:

```
MLP1 Train Metrics (MAE|MSE|R2):  2.2317  |  10.1508 |  0.8231
MLP1 Test Metrics   (MAE|MSE|R2):  2.9055  |  16.0168 |  0.7829
MLP1 Iterations:  436
----------------------------------------------------------------------------------------------------
MLP2 Train Metrics (MAE|MSE|R2):  3.1520  |  17.3347  |  0.6980
MLP2 Test Metrics   (MAE|MSE|R2):  3.5290  |  21.4125  |  0.7098
MLP2 Iterations:  550
----------------------------------------------------------------------------------------------------
MLP3 Train Metrics (MAE|MSE|R2):  0.4717  |   0.4635   |  0.9919
MLP3 Test Metrics   (MAE|MSE|R2):  4.6030  |  41.4784  |  0.4379
MLP3 Iterations:  2000
----------------------------------------------------------------------------------------------------
AST3 Train Metrics (MAE|MSE|R2):  3.2745 | 18.2952 |  0.7060
AST3 Test Metrics  (MAE|MSE|R2):   3.0813 | 16.1596 |  0.7056
AST3 Iterations:  1196
```

My linear regression algorithm from assignment 3 had an 0.70 R2 score for both the training and test datasets. MLP1, the single layer 100 neurons hyperbolic tan activation algorithm was able to increase the R2 score to 0.82 in training and 0.78 in testing. MLP2, the 2 layer identity activator, had a similar result to my findings with linear regression.

The MLP3 results are quite interesting. This model is the most dramatic difference between the other MLP and the LR models, with the ‘lbfgs’ solver, an optimizer in the family of quasi-Newton methods, instead of the stochastic gradient descent used by the others.With an 0.99 R2 score on the training data, I thought that this algorithm was optimized for this dataset and was capable of results the other algorithms were not. However, MLP3 had dismal test scores at 0.43. I’m assuming this is a case of overfitting for the algorithm, especially since it went all the way to the max of 2000 iterations without convergence.

I tried manipulating some of the traits of MLP3 while preserving the 4 layers and rectified linear unit activator but found little success with other options. Experimenting with different numbers of neurons had small effects on the results as well. For MLP1 I tried counts as low as 5 and as high as 1000 for the neurons with variations of ~0.15 to the R2 score. For MLP2 and 3 I experimented with splitting the neurons evenly between layers versus having the count vary between layers and found that varying counts of neurons seemed to produce slightly better results.

Surprisingly, it seems the simpler algorithms perform better on this dataset. The single layer MLP and assignment 3 linear regression algorithms produced the best R2 scores in testing. I believe that the MLP regressor would outperform the linear regression algorithm with this dataset due to it’s customization and different parameters that can be better tweaked to find a best fit. I still believe that the inputs, displacement, horsepower, weight, and acceleration, have weak correlations to the output, MPG, resulting in a dataset that cannot be solved to a high R2 score by default.