IoT Analytics- Project 4: Neural Networks Project

Wen-Han Hu(whu24)

Task 1. Automatic grid search:

Data preparation:

After reading the data from the dataset, we first spilt the data randomly into 2000 training data and 300 testing data. The summary of both dataset as following.

Train Data:

	x1	x2	x3	x4	x5
count	2000.000000	2000.000000	2000.000000	2000.000000	2000.000000
mean	9.027309	20.258701	50.144458	53.800349	68.214058
std	10.122487	10.029083	9.699734	9.938293	10.015642
min	-30.873000	-10.122000	14.165000	18.628000	28.221000
25%	2.280400	13.411250	43.380250	47.065000	61.484250
50%	9.006800	20.552000	50.354500	53.920000	68.290500
75%	15.828500	26.691250	57.017250	60.351500	74.764000
max	40.836000	57.969000	80.023000	89.961000	104.710000

Test Data

	x1	x2	x3	x4	x5
count	300.000000	300.000000	300.000000	300.000000	300.000000
mean	9.819267	20.676267	49.793113	53.943373	68.892227
std	9.553811	9.871697	9.571632	10.468692	10.520026
min	-16.954000	-3.793400	20.370000	26.082000	40.287000
25%	2.719225	13.475250	43.020000	46.871250	62.144750
50%	9.565100	20.667500	49.860500	54.185500	69.050500
75%	16.534750	27.330750	56.809500	61.170500	75.618250
max	35.158000	52.949000	76.563000	78.909000	94.622000

By evaluating the mean, standard deviation above, we can observe that the train and test dataset are very similar distribution which make sure there is not much difference between these two subsets.

Run Automatic grid search:

In order to establish a multilayers neural nets model, we need to decide hyperparameters before we train the NN model. The hyperparameters includes the learning rate γ, the momentum, the regularization parameter λ, and the size of the minibatch. (Reference: textbook section 10.10) If we just simply run everything on automatic grid search, we will have tremendous combination and may spend endless time on waiting the results. Instead of placing everything into the automatic grid search. We use the approach on section 10.10 of textbook which is starting a single hidden layer with 2 or 3 nodes, then we search for the best values of the hyper-parameters, determined the best values, run the previous NN model on a data set, increase the number of nodes and layers, and calculate the cost on both train and test data.

The first layer automatic grid search results as following.

```
Best score: 0.8630777870595906
Best params: {'alpha': 0.1, 'hidden_layer_sizes': 21, 'learning_rate_init': 0.1}
```

```
loss: 888.9365247543599
```

*Note: the alpha here is the same parameter as regularization parameter λ.
*Note: the activation function used sigmoid function.
*Note: the score is the indicator similar to R-squared after 3-fold cross validation. The higher score means the NN model is better, low score mean the model may be overfitting.
*Note: the loss here we use mean squared error as an indicator to evaluate our NN modal performance.

The second layer automatic grid search results as following.

```
Best score: 0.9409307668174363
Best params: {'alpha': 0.0001, 'hidden_layer_sizes': (21, 20), 'learning_rate_init': 0.1}
```

```
loss: 232.1343410560544
```

We have two hidden layers with first layer with 21 nodes and 20 nodes on the second one.

The third layer automatic grid search results as following.

```
Best score: 0.15959943737900834
Best params: {'alpha': 0.1, 'hidden_layer_sizes': (21, 20, 18), 'learning_rate_init': 0.1}
```

```
loss: 124.16405197579618
```

Third hidden layer with 18 nodes.

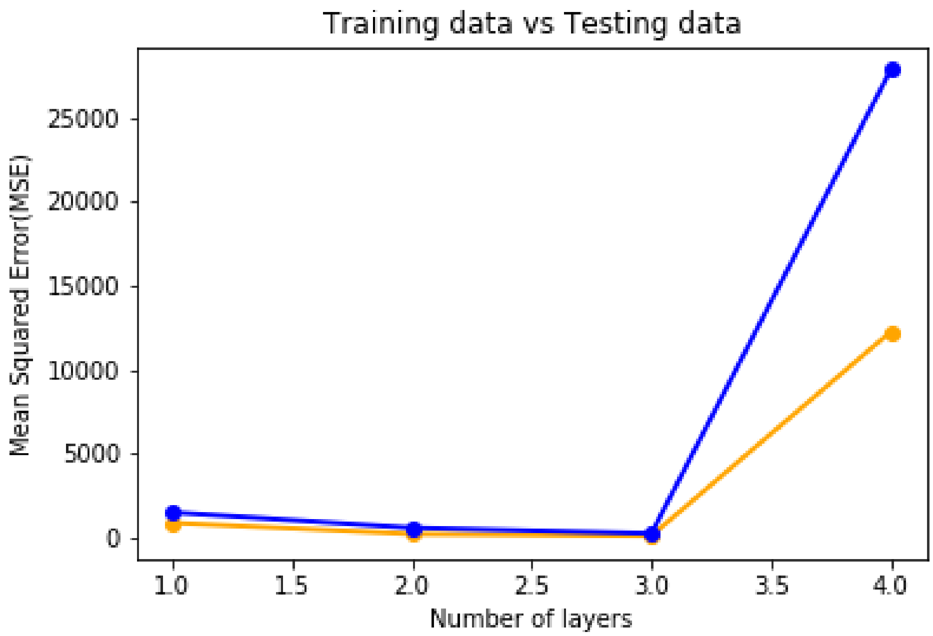
The fourth layer automatic grid search results as following.

```
Best score: -0.0017981275850611595
Best params: {'alpha': 1e-06, 'hidden_layer_sizes': (21, 20, 18, 14), 'learning_rate_init': 0.1}
```

```
loss: 12259.020420123194
```

Fourth hidden layer with 14 nodes.

Based on the metrices above, we can find that even though our NN model with 3 hidden layers has minimum MSE with 124, the score 0.16 indicates that the model is overfitting. The 4 hidden layer model, of course, the situation of overfitting is worse.



The orange line represents to the training MSE cost, the blue line represents to the testing MSE cost.

<u>Conclusion:</u>

Although we have better MSE with 3 hidden layers NN model, the 3-fold cross validation score with 0.16 indicates that the model is overfitting. On the other hand, the NN model with 2 hidden layers (21, 20) has the best cross validation score with 0.94 which indicates that the model is very good fit for our training data. Thus, we choose the neural network model with 2 hidden layer which is the first layer contains 21 neurons and second layer contain 20 neuron, and learning rate = 0.1, λ = 0.0001 as our best ANN model.

Task 2. Compare the trained neural network with multivariable regression:

In the previous multivariable regression model, we trained the regression model with all 5 variables having best R squared score. Hence, we also used the same model for comparison. The coefficients of the multivariable regression model are on the next page.

```
                            OLS Regression Results
=======================================================================================
Dep. Variable:                      y   R-squared (uncentered):                   0.998
Model:                            OLS   Adj. R-squared (uncentered):              0.998
Method:                 Least Squares   F-statistic:                          2.333e+05
Date:                Tue, 19 Nov 2019   Prob (F-statistic):                        0.00
Time:                        21:15:15   Log-Likelihood:                         -12835.
No. Observations:                2300   AIC:                                  2.568e+04
Df Residuals:                    2295   BIC:                                  2.571e+04
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
x1             7.9736      0.133     60.042      0.000       7.713       8.234
x2             3.4508      0.131     26.270      0.000       3.193       3.708
x3             6.1477      0.118     51.899      0.000       5.915       6.380
x4             6.3730      0.112     56.895      0.000       6.153       6.593
x5             9.4455      0.099     95.281      0.000       9.251       9.640
==============================================================================
Omnibus:                     1463.625   Durbin-Watson:                   2.000
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            18255.249
Skew:                           2.856   Prob(JB):                         0.00
Kurtosis:                      15.565   Cond. No.                         10.6
==============================================================================
```

The result shows that we have a very good fit regression model with nearly 1 R-squared score. The regression function will be as following.

$$Y = 7.9736X_1 + 3.4508X_2 + 6.1477X_3 + 6.373X_4 + 9.4455X_5$$

The p-values of coefficients are less than 0.01 which indicates that we reject the null hypothesis (coefficient is equal to 0), we don't need to remove any independent variables.
The p-value is 0 of F-statistic result indicate that we reject the null hypothesis of F statistic which is all of the regression coefficients are equal to zero. This result is consistent with the analysis of coefficients.

The SSE performance results as following.

```
Best ANN's SSE: 77758.6941079954
Best Regression's SSE: 633402884.1532137
```

The sum of squared error of best ANN model is 77758.
The sum of squared error of best Regression model is 633402884.
Since less SSE means the model has better fit, ANN model is better than Regression model.

<u>Conclusion:</u>

The SSE results indicate that even though we have quite decent multivariable regression model, the SSE still much higher than our best ANN model. Because the structure of neural network, we have better chance to adjust our weight/coefficients to allow the variable fitting our outcome. Thus, ANN model is better than regression model in our case.