**CSE 514 – Spring 2022 Programming Assignment 1**

Name:Zhuomin Li ID: 502313 email:l.zhuomin@wustl.edu

**1.Introduction**

**1.1 The description of the problem**

Given a dataset of Concrete Compressive Strength in the UCI repository, we need to design and implement a (stochastic) gradient descent algorithm or algorithms for regression to calculate “the capacity of concrete to withstand loads before failure”1

after considered eight features: Cement, Blast Furnace Slag, Fly Ash, Water, Superplasticizer, Coarse Aggregate, Fine Aggregate and Age(day). I will utilize Python and it’s related fundamental package, such as Numpy, Pandas and Matplotlib, to solve following questions in a Jupyter Notebook file. Please check the file named ‘**ReadMe**’ to run codes. Here is my general answer description for each question:

1. For question A, I will apply the gradient descent algorithm for optimizing a uni-variate linear regression model of eight characters as above.
2. For question B, the variation of the regression model should be all of the features plusing one(b). Both question will utilize MSE(mean squared error) as the loss function.
3. For the question C, what I should extended is a multi-variate quadratic regression model, which has 36 quadratic terms, such as: , and eight linear terms. The apply multi-variate regression model to calculate and evaluate.
4. For the question D, I will try to normalize or standardize each variable, as some characters, such like Age and Cement, do not have same unit of measurement.

**1.2 Details of algorithm**

Generally speaking, the stopping criterion would be complete the total times of passes of the entire training dataset that I initialed before start, for example, if I assume the “epoch” is 500, it means that the model will stop as long as it has finished 500 times. Besides, for this gradient descent model, I will initial a pivot learning rate at first, and find the most suitable one after trail and error!

For each question, the steps and relevant formulas is as followed:

1. Uni-variate linear regression

The model formula should be:



Lose function(here I used MSE) should be:



Since I need to optimize this uni-variate linear regression model, the partical derivative of the loss function should be calcuate, and update the coefficient m and n for eight variables. Besides, I assume that  is the learning rate, and it could be initilized as the eqution should be:



Therefore, for each step, the updating formula of m and b should be:



The stopping criterion should be(I assumed): 500

B)Multi-variate linear regression

The high level idea of solving multi-variate linear regression is slightly different with uni-variate linear regression. For this problem, it needs to consider more than one variate’s influence for the concrete compressive strength. Given conditions above, the formula of this problem should be:



Meanwhile, the loss function should be:



The updating formula of m should be:



The order of y\_predicted and y\_i should be noticed, as it decides whether we need add minus sign during seeking for derivation or not.

**1.3 Pseudo-code of algorithms**

For the **Question A**, the pseudo-code should be as followed:

//Initialization

Int n is assumed iteration times(epoches)

Int m = b = 0

Int  is learning rate, and could be initilized as 

Int  is traing variable input and training variable out put

* Function uni\_variate\_linear\_regression\_model\_training (current\_character\_m\_b, x\_training ,y\_training, lr, epoches):

Record the size of x\_training as n





Print the current feature variable

Return coefficient m and b

* Function uni\_variate\_linear\_regression\_model(x\_training, y\_training,x\_testing, y\_testing, current\_character\_m\_b):

Record the size of x\_training and x\_testing separetly

Calculate MSE for training and testing set by using:



Print out the result

Return

* Function calculate\_current\_character\_variance(y):

Calculate variance by utilizing sum, mean and square value

* Function calculate\_current\_character\_variance\_explained (current\_character\_variance, final\_model\_trained\_mse)

Result shoud be 1 - (final\_model\_trained\_mse[0] / current\_character\_variance)

Call function uni\_variate\_linear\_regression\_model\_training to calculate coefficient m and b, and the second function is to calculate MSE value. Then use rest of function to get variance explained for the response variable.

Draw the plot for later use.

For the **Question B**, the pseudo-code should be as followed:

Add a column of ones before the head of training and testing set

Initial a learning rate lr = 5.8e-7 and epoches should be 500

* Function multi\_variate\_linear\_regression\_training (x\_training, y\_training, lr, epoches):

Initialize 

Int n is assumed iteration times

While interationTimes <=n {



Return m

}

* Function multi\_variate\_linear\_regression\_model (current\_character\_m, x\_training, y\_training, x\_testing, y\_testing):

Calculate MSE

Calculate the variance explained of multi-variate regression model for all characters

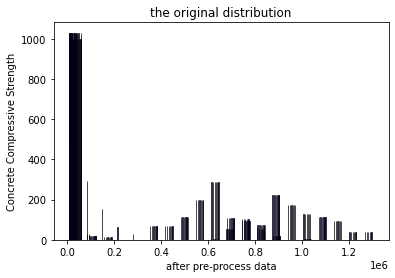
For the **Question C,** what I should extended is a multi-variate quadratic regression model, which has 36 quadratic terms, and eight linear terms.

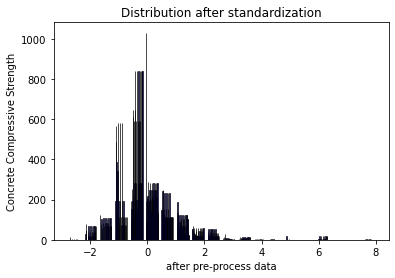
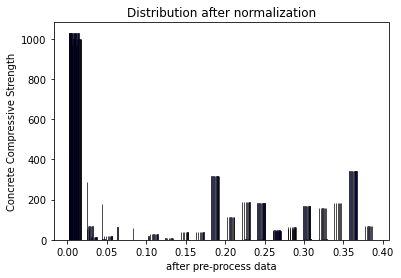
* Function reshape\_a\_new\_dataset(data):

Initial i, j for two while loop to calculate x1^2, x1 \* x2, ..., x8^2

And use the model just like Question B to get the result

For the **Question D**, I will try to normalize or standardize each variable, as some variables, such like Age and Cement, do not have the same unit of measurement. I imported StandardScaler, Normalizer from sklearn.preprocessing to process data.





**2.Result**

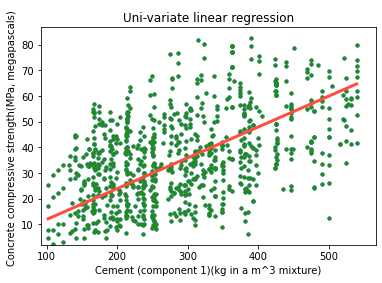
**2.1 MSE of training set and test set of each character for later use:**

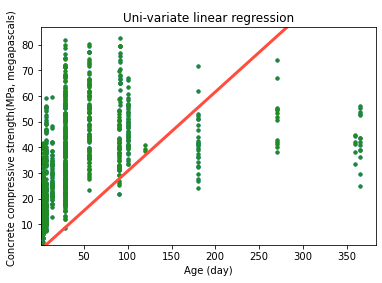
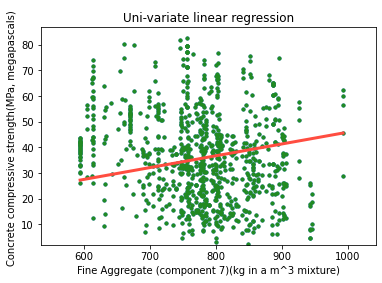
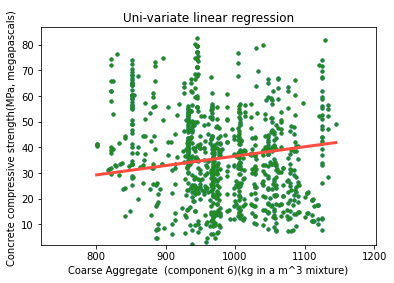
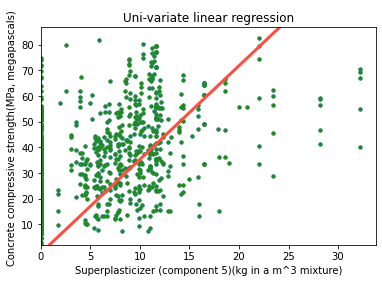
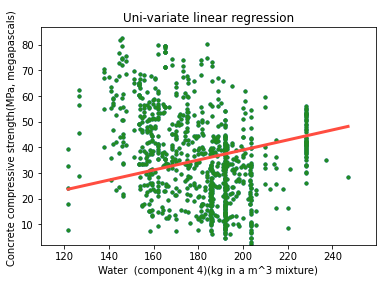
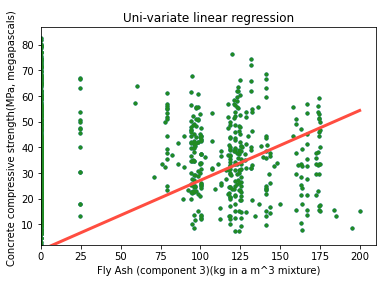
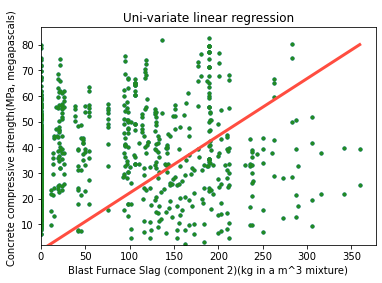
|  |  |  |
| --- | --- | --- |
| **Character** | **MSE of training set** | **MSE of test set** |
| Cement (component 1)(kg in a m^3 mixture) | 249.423 | 111.026 |
| Blast Furnace Slag (component 2)(kg in a m^3 mixture) | 296.809 | 173.734 |
| Fly Ash (component 3)(kg in a m^3 mixture) | 298.191 | 144.638 |
| Water (component 4)(kg in a m^3 mixture) | 358.233 | 186.91 |
| Superplasticizer (component 5)(kg in a m^3 mixture) | 313.751 | 146.026 |
| Coarse Aggregate (component 6)(kg in a m^3 mixture) | 322.363 | 165.965 |
| Fine Aggregate (component 7)(kg in a m^3 mixture) | 333.462 | 171.538 |
| Age (day) | 270.035 | 143.722 |
| **All characters in Question B** | 136.3 | 56.8 |
| **45 characters in Question C** | 185.3 | 119.9 |
| **45 characters after normalized in Question D** | 7.228348558311763e-05 | 9.476957712276736e-08 |
| **45 characters after standardized in Question D** | 1.1345330579318404 | 0.06861343078398412 |

**2.2 Variance explained for the response variable of training set and testing set for each feature:**

|  |  |  |
| --- | --- | --- |
| **Character** | **Variance explained of training set** | **Variance explained of testing set** |
| Cement (component 1)(kg in a m^3 mixture) | 0.157 | 0.625 |
| Blast Furnace Slag (component 2)(kg in a m^3 mixture) | 0.012 | 0.491 |
| Fly Ash (component 3)(kg in a m^3 mixture) | -0.008 | 0.511 |
| Water (component 4)(kg in a m^3 mixture) | 0.157 | 0.366 |
| Superplasticizer (component 5)(kg in a m^3 mixture) | -0.06 | 0.506 |
| Coarse Aggregate (component 6)(kg in a m^3 mixture) | -0.09 | 0.439 |
| Fine Aggregate (component 7)(kg in a m^3 mixture) | -0.127 | 0.420 |
| Age (day) | 0.087 | 0.514 |
| **All characters in Question B** | 0.637 | 0.847 |
| **45 characters in Question C** | 0.374 | 0.595 |
| **45 characters after normalized in Question D** | -0.095 | 0.999 |
| **45 characters after standardized in Question D** | -0.001 | 0.939 |

**2.3 Plots of my trained uni-variate model on top of scatterplots of the training data used:**

****

****

**3.Discussion**

**3.1Describe how the different models compared in performance on the training data. Did the same models that performed well on the training data do well on the testing data?**

We will have a form which is sorted by MSE of training data as followed:

|  |  |  |
| --- | --- | --- |
| **Character** | **MSE of training set** | **MSE of test set** |
| **45 characters after normalized in Question D** | 0.0000723 | 9.48E-08 |
| **45 characters after standardized in Question D** | 1.134533058 | 0.068613431 |
| **All characters in Question B** | 136.3 | 56.8 |
| **45 characters in Question C** | 185.3 | 119.9 |
| Cement (component 1)(kg in a m^3 mixture) | 249.423 | 111.026 |
| Coarse Aggregate (component 6)(kg in a m^3 mixture) | 270.035 | 165.965 |
| Fine Aggregate (component 7)(kg in a m^3 mixture) | 296.809 | 171.538 |
| Water (component 4)(kg in a m^3 mixture) | 298.191 | 186.91 |
| Superplasticizer (component 5)(kg in a m^3 mixture) | 313.751 | 146.026 |
| Age (day) | 322.363 | 143.722 |
| Blast Furnace Slag (component 2)(kg in a m^3 mixture) | 333.462 | 173.734 |
| Fly Ash (component 3)(kg in a m^3 mixture) | 358.233 | 144.638 |

Then it’s not hard to conclude that normalized model with 45 characters has the outstanding performance on the training data. Besides, the 2nd and 3rd are better than those who has only one character. Therefore, the more we pre-process data on the right track, the more accuate it will be; the more characters to fit, the less loss it will has.

Most of models performed well on both the training data and testing data. However, for Blast Furnace Slag, it is different. Generally speaking, the performance of the model training set and the performance of the test set are not positively correlated, which means we cannot predict MSE of test set by checking MSE of training set.

**3.2 Describe how the coefficients of the uni-variate models predicted or failed to predict the coefficients in the multi-variate model(s).**

For coefficients of the uni-variate models and multi-variate models is as followed:

|  |  |  |  |
| --- | --- | --- | --- |
| **Character** | **Uni-variate m** | **Multi-variate m** | **MSE of training set** |
| Cement (component 1)(kg in a m^3 mixture) | 0.119940925 | 2.91178E-05 | 249.423 |
| Blast Furnace Slag (component 2)(kg in a m^3 mixture) | 0.035438781 | 0.102119833 | 292.351 |
| Fly Ash (component 3)(kg in a m^3 mixture) | -0.029389575 | 0.07341922 | 298.191 |
| Water (component 4)(kg in a m^3 mixture) | 0.198964897 | 0.070304463 | 358.233 |
| Superplasticizer (component 5)(kg in a m^3 mixture) | 0.151670688 | -0.051954772 | 313.751 |
| Coarse Aggregate (component 6)(kg in a m^3 mixture) | 0.151670688 | 0.018543999 | 322.363 |
| Fine Aggregate (component 7)(kg in a m^3 mixture) | 0.046040219 | -0.004378694 | 333.462 |
| Age (day) | 0.10863231 | 0.009997803 | 270.035 |

After sorted by Uni-variate m, we can get:

|  |  |  |  |
| --- | --- | --- | --- |
| **Character** | **Uni-variate m** | **Multi-variate m** | **MSE of training set** |
| Cement (component 1)(kg in a m^3 mixture) | -0.029389575 | 2.91178E-05 | 249.423 |
| Blast Furnace Slag (component 2)(kg in a m^3 mixture) | 0.035438781 | 0.102119833 | 292.351 |
| Fly Ash (component 3)(kg in a m^3 mixture) | 0.046040219 | 0.07341922 | 298.191 |
| Water (component 4)(kg in a m^3 mixture) | 0.10863231 | 0.070304463 | 358.233 |
| Superplasticizer (component 5)(kg in a m^3 mixture) | 0.119940925 | -0.051954772 | 313.751 |
| Fine Aggregate (component 7)(kg in a m^3 mixture) | 0.151670688 | -0.004378694 | 333.462 |
| Coarse Aggregate (component 6)(kg in a m^3 mixture) | 0.151670688 | 2.91178E-05 | 322.363 |
| Age (day) | 0.198964897 | 0.009997803 | 270.035 |

If We can utilize uni-variate m to predict multi-variate m, the order of multi-variate m will be sorted ascending from top to bottom. However, the fact is multi-variate m do not have that order. Therefore, it’s failed for uni-variate to predict the latter. Maybe because of difference of the dimension between uni-variate model and multi-variate model. The conclusion still works after we added quadratic terms, standardlized or normalized.

**3.3 Draw some conclusions about what factors predict concrete compressive strength. What would you recommend to make the hardest possible concrete?**

1)In a **multi-variate model**, the value of coefficience m might reflect how much it can contribute to concrete compressive strength.After checking the coefficience m in multi-variate model, the rank of factors for predicing concrete compressive strength should be:

|  |  |  |  |
| --- | --- | --- | --- |
| **Character** | **Uni-variate m** | **Multi-variate m** | **MSE of training set** |
| Blast Furnace Slag (component 2)(kg in a m^3 mixture) | 0.035438781 | 0.07341922 | 292.351 |
| Fly Ash (component 3)(kg in a m^3 mixture) | 0.046040219 | 0.070304463 | 298.191 |
| Water (component 4)(kg in a m^3 mixture) | 0.10863231 | 0.018543999 | 358.233 |
| Superplasticizer (component 5)(kg in a m^3 mixture) | 0.119940925 | 0.009997803 | 313.751 |
| Coarse Aggregate (component 6)(kg in a m^3 mixture) | 0.151670688 | 2.91178E-05 | 322.363 |
| Fine Aggregate (component 7)(kg in a m^3 mixture) | 0.151670688 | -0.004378694 | 333.462 |
| Age (day) | 0.198964897 | -0.051954772 | 270.035 |
| Cement (component 1)(kg in a m^3 mixture) | -0.029389575 | 0.102119833 | 249.423 |

1. In a **uni-variate model,** the MSE for predicting concrete compressive strength should be taken into consideration as well. After **standardlized(For this question I choose standardlize data)** all character’s value. Now we can get:

|  |  |  |  |
| --- | --- | --- | --- |
| **Character** | Coefficience m’s value | MSE of training | MSE of testing |
| Cement (component 1)(kg in a m^3 mixture) | 0.11984403124750186 | 1.136 | 0.06 |
| Blast Furnace Slag (component 2)(kg in a m^3 mixture) | 0.1197978505086127 | 1.133 | 0.037 |
| Fly Ash (component 3)(kg in a m^3 mixture) | 0.11965624418599627 | 1.156 | 0.101 |
| Water (component 4)(kg in a m^3 mixture) | 0.11936413643961247 | 1.192 | 0.113 |
| Superplasticizer (component 5)(kg in a m^3 mixture) | 0.11958043761638787 | 1.069 | 0.087 |
| Coarse Aggregate (component 6)(kg in a m^3 mixture) | 0.11922305127728752 | 1.206 | 0.094 |
| Fine Aggregate (component 7)(kg in a m^3 mixture) | 0.11907346721499563 | 1.158 | 0.037 |
| Age (day) | 0.11875831948894204 | 1.196 | 0.058 |

After sorted by coefficience m in a descending order from top to bottom, now we can have:

|  |  |  |  |
| --- | --- | --- | --- |
| **Character** | **Coefficience m’s value** | MSE of training | MSE of testing |
| Cement (component 1)(kg in a m^3 mixture) | 0.119844031 | 1.136 | 0.06 |
| Blast Furnace Slag (component 2)(kg in a m^3 mixture) | 0.119797851 | 1.133 | 0.037 |
| Fly Ash (component 3)(kg in a m^3 mixture) | 0.119656244 | 1.156 | 0.101 |
| Superplasticizer (component 5)(kg in a m^3 mixture) | 0.119580438 | 1.069 | 0.087 |
| Water (component 4)(kg in a m^3 mixture) | 0.119364136 | 1.192 | 0.113 |
| Coarse Aggregate (component 6)(kg in a m^3 mixture) | 0.119223051 | 1.206 | 0.094 |
| Fine Aggregate (component 7)(kg in a m^3 mixture) | 0.119073467 | 1.158 | 0.037 |
| Age (day) | 0.118758319 | 1.196 | 0.058 |

After sorted by MSE of training in a ascending order from top to bottom, now we have:

|  |  |  |  |
| --- | --- | --- | --- |
| **Character** | Coefficience m’s value | **MSE of training** | MSE of testing |
| Superplasticizer (component 5)(kg in a m^3 mixture) | 0.119580438 | 1.069 | 0.087 |
| Blast Furnace Slag (component 2)(kg in a m^3 mixture) | 0.119797851 | 1.133 | 0.037 |
| Cement (component 1)(kg in a m^3 mixture) | 0.119844031 | 1.136 | 0.06 |
| Fly Ash (component 3)(kg in a m^3 mixture) | 0.119656244 | 1.156 | 0.101 |
| Fine Aggregate (component 7)(kg in a m^3 mixture) | 0.119073467 | 1.158 | 0.037 |
| Water (component 4)(kg in a m^3 mixture) | 0.119364136 | 1.192 | 0.113 |
| Age (day) | 0.118758319 | 1.196 | 0.058 |
| Coarse Aggregate (component 6)(kg in a m^3 mixture) | 0.119223051 | 1.206 | 0.094 |

To encapsulate, Cement (component 1)(kg in a m^3 mixture), Blast Furnace Slag (component 2)(kg in a m^3 mixture), Fly Ash (component 3)(kg in a m^3 mixture) , Superplasticizer (component 5)(kg in a m^3 mixture) and Water (component 4)(kg in a m^3 mixture) are very essential for people to predict concrete compressive strength.

I will highly recommend consider using better cement to improve concrete compressive strength, it worths!

**3.4 if you include comparisons to the results from normalized or standardized data**

**3.5 if you include comparisons to the results from your quadratic model**

**Finished**

**Quotation:**

**[1]https://www.sciencedirect.com/topics/materials-science/compressive-strength#:~:text=Compressive%20strength%20can%20be%20defined,the%20characteristics%20of%20the%20concrete.**