1 Introduction

1.1 Description/formulation of the problem

Concrete is the most important material in civil engineering, and concrete compressive strength is the measure people value a lot. In order to explore the relationship between concrete compressive strength and concrete's age and ingredients, I analyzed the data provided by Prof. I-Cheng Yeh applying the regression models and gradient descent algorithm. We can design the concrete mixture and predict the concrete compressive strength according to the result.

1.2 Pre-process

1.2.1 Description of data

The data has 1030 instances within 9 attributes. It includes 8 quantitative input variables: cement, blast furnace slag, fly ash, water, superplasticizer, coarse aggregate, fine aggregate and age; and 1 quantitative output variable: concrete compressive strength. All the variables are numerical.

1.2.2 Data splitting

I created 2 part of the dataset including 900 instances for training and 130 instances for testing, randomly selected.

1.2.3 Missing value

```
Cement                           0
Blast Furnace Slag               0
Fly Ash                          0
Water                            0
Superplasticizer                 0
Coarse Aggregate                 0
Fine Aggregate                   0
Age                              0
Concrete compressive strength    0
dtype: int64
```

Figure 1, missing value

There is no missing value in the data.

1.2.4 Outliers

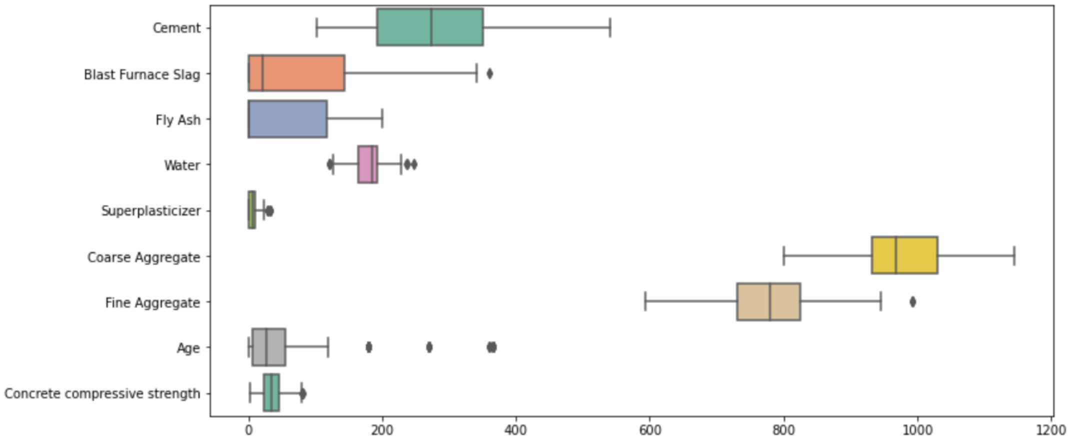

Figure 2, checking outliers

From the above boxplot we can see that there are outliers in some columns. I replaced them with the median of their column.

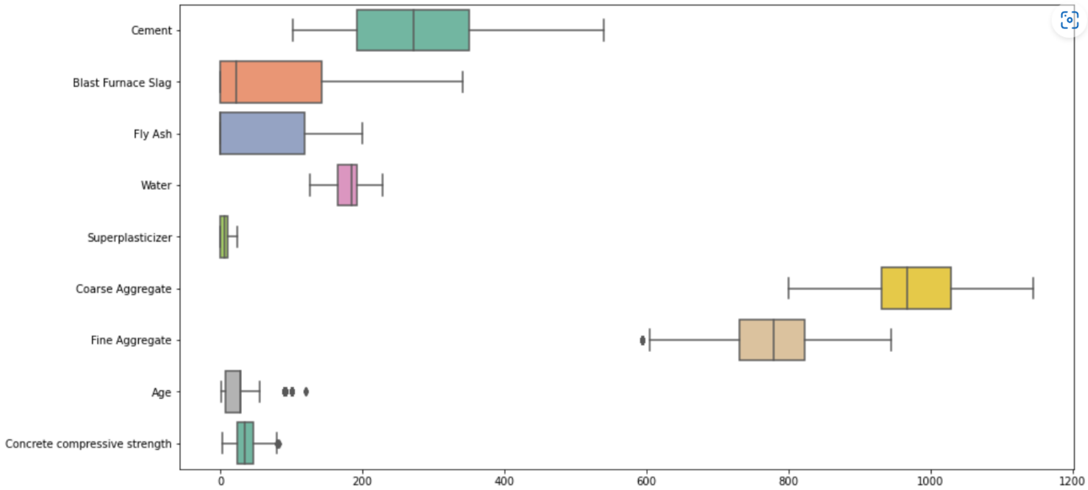

Figure 3, after replacing outliers

1.2.5 Standardization

	count	mean	std	min	25%	50%	75%	max
Cement	1030.0	281.165631	104.507142	102.000000	192.375000	272.900000	350.000000	540.000000
Blast Furnace Slag	1030.0	73.895485	86.279104	0.000000	0.000000	22.000000	142.950000	359.400000
Fly Ash	1030.0	54.187136	63.996469	0.000000	0.000000	0.000000	118.270000	200.100000
Water	1030.0	181.566359	21.355567	121.750000	164.900000	185.000000	192.000000	247.000000
Superplasticizer	1030.0	6.203112	5.973492	0.000000	0.000000	6.350000	10.160000	32.200000
Coarse Aggregate	1030.0	972.918592	77.753818	801.000000	932.000000	968.000000	1029.400000	1145.000000
Fine Aggregate	1030.0	773.578883	80.175427	594.000000	730.950000	779.510000	824.000000	992.600000
Age	1030.0	45.662136	63.169912	1.000000	7.000000	28.000000	56.000000	365.000000
Concrete compressive strength	1030.0	35.817836	16.705679	2.331808	23.707115	34.442774	46.136287	82.599225

Figure 4, raw data information

From above we can see that Mean and the median is nearly same for the Cement, Water, Superplastic, Coarse Aggregate, Fine Aggregate, Strength so we can say it is approximately normally distributed. Slag, Ash, Age are having much values at the maximum portion so we can say it is skewed towards right side.

I used Zscore standardization:

$$z = (x - \mu) / \sigma$$

After normalization, the data becomes:

	count	mean	std	min	25%	50%	75%	max
Cement	1030.0	4.335798e-17	1.000486	-1.715219	-0.850026	-0.079130	0.658977	2.477918
Blast Furnace Slag	1030.0	4.248489e-16	1.000486	-0.858191	-0.858191	-0.600407	0.814184	3.150353
Fly Ash	1030.0	1.267056e-15	1.000486	-0.847132	-0.847132	-0.847132	1.001836	2.281122
Water	1030.0	-4.093813e-16	1.000486	-2.673240	-0.813466	0.162552	0.502458	2.250549
Superplasticizer	1030.0	-3.282164e-17	1.000486	-1.090890	-1.090890	0.069134	0.749621	3.183845
Coarse Aggregate	1030.0	-9.011131e-17	1.000486	-2.212137	-0.526514	-0.063289	0.726766	2.214232
Fine Aggregate	1030.0	-5.716301e-16	1.000486	-2.269697	-0.528758	0.087340	0.631232	2.192293
Age	1030.0	-1.662101e-16	1.000486	-1.124724	-0.908821	-0.153159	-0.153159	3.157360
Concrete compressive strength	1030.0	4.486163e-16	1.000486	-2.005443	-0.725298	-0.082351	0.617961	2.801689

Figure 5, data information after standardization

the features are rescaled and their properties of a standard normal distribution changed to μ=0 and σ=1.

1.3 Details of algorithm： MSE

The algorithm is y=mx+b (univariate) or M·X+b (multivariate) . I set the initial weights m and bias b randomly. I used batch gradient descent since the number of samples is not very large, batch gradient descent gives optimal solution given sufficient time to converge. I used MSE $L(m,b) = \frac{1}{n}\sum_{i=1}^{n}(y_i - (mx_i + b))^2$as loss function to train the model. The update rules for m is $m_{new} = m_{old} - \frac{\alpha}{n}\sum_{i=1}^{n} -2x_i(\, y_i - (m_{old}x_i + b_{old}))$, for b is $m_{new} = b_{old} - \frac{\alpha}{n}\sum_{i=1}^{n} -2(\, y_i - (m_{old}x_i + b_{old}))$ The gradient descent would stop when it's reach 200000 criterions or the enhance of MSE loss value is smaller than 1e-5. Then, I chose the learning rate for univariate as 0.00001 and for multivariate as 0.1, since their variance explained is the best among [0.01,0.001,0.0001,0.00001,0.000001,0.0000001].

1.4 (Optional extension 1)Details of algorithm： MAE

Changes from using MAE from using MSE:

I used MSE $L(m,b) = \frac{1}{n}\sum_{i=1}^{n}|y_i - f(x_i)|$as loss function to train the model. The update rules for m is $m_{new} = m_{old} - \frac{\alpha}{n}\sum_{i=1}^{n} -gx_i$, for b is $b_{new} = b_{old} - \frac{\alpha}{n}\sum_{i=1}^{n} -g$. (g is -1, where y_pred>y_true, and +1 where y_pred<y_true) Then, I chose the learning rate for univariate as 0.000001 and for multivariate as 0.001, since their variance explained is the best among [0.01,0.001,0.0001,0.00001,0.000001,0.0000001]. In addition, I used R-squared = SSR/SST to calculate Variance explained.

1.5 (Optional extension 2)Details of algorithm： Ridge Regression

Changes from using Ridge Regression from using MSE:

I used Ridge Regression $L(m,b) = \frac{1}{n}\sum_{i=1}^{n}(\, y_i - (mx_i + b)\,)^2 + \lambda\|m\|_2^2$ as loss function to train the model. The update rules for m is $m_{new} = m_{old} - \frac{\alpha}{n}\sum_{i=1}^{n} -2x_i(\, y_i - (m_{old}x_i + b_{old})) + 2 * lam * m_{old}$, for b is $b_{new} = b_{old} - \frac{\alpha}{n}\sum_{i=1}^{n} -2x_i(\, y_i - (m_{old}x_i + b_{old})) + 2 * lam * b_{old}$. Then, I chose the learning rate for univariate as 0.000001 and for multivariate as 0.1, since the variance explained is the best among [0.01,0.001,0.0001,0.00001,0.000001, 0.0000001]. In addition, I used R-squared = SSR/SST to calculate Variance explained. Finally, for Ridge Regression, we needed to add a lambda value. I set lambda value as 0.01 for multivariate and 0.001 for univariate since their variance explained is the best among [0.1,0.01,0.001,0.0001,0.00001]

1.6 Pseudo-code

1. Input predictor values X, response value Y, learning rate alpha, iteration stop number,

error tolerance, (lambda for Ridge Regression)

2. Set initial weights m and bias b
3. Compute loss function
4. Compute the ∂L/∂w and ∂L/∂b gradient.
5. Update m. `m_new = m_old -` α * ∂L/∂m.
6. Update b. `B_new = B_old -` α * ∂L/∂w
7. Compute new loss function
8. If new loss function-old loss function < error tolerance, or reach iteration stop number, exit.
9. Else, repeat 4,5,6,7,8 steps.

2 Results

2.1 Variance explained on the training dataset: MSE

2.1.1 univariate regression: raw data

Cement
Variance explained on the training dataset: 0.24823163940056947
Blast Furnace Slag
Variance explained on the training dataset: 0.01399897223388269
Fly Ash
Variance explained on the training dataset: 0.007803885255368792
Water
Variance explained on the training dataset: 0.09082049901952183
Superplasticizer
Variance explained on the training dataset: 0.1348944801643418
Coarse Aggregate
Variance explained on the training dataset: -599294.1763787885
Fine Aggregate
Variance explained on the training dataset: -7306.139281350159
Age
Variance explained on the training dataset: 0.22094244011728992

2.1.2 multivariate regression: raw data

Variance explained on the training dataset:0. 35558992698300307(standardized)
Variance explained on the training dataset:-3.4680764579360472(not standardized)

2.1.3univariate regression: pre-processed data

Cement
Variance explained on the training dataset: 0.24823163940056947
Blast Furnace Slag
Variance explained on the training dataset: 0.014724912864741757
Fly Ash
Variance explained on the training dataset: 0.023178258591711742
Water
Variance explained on the training dataset: 0.10111962125335994
Superplasticizer
Variance explained on the training dataset: 0.1203917591842465

Coarse Aggregate
Variance explained on the training dataset: -599294.1763787885
Fine Aggregate
Variance explained on the training dataset: -7306.139281350159
Age
Variance explained on the training dataset: 0.25402277518261673

2.1.4 multivariate regression: pre-processed data

Variance explained on the training dataset: 0.5987593150879904

2.2 Variance explained on the testing dataset: MSE

2.2.1 univariate regression: raw data

Cement
Variance explained on the testing dataset: 0.2418292928498601
Blast Furnace Slag
Variance explained on the testing dataset: 0.03829535032724951
Fly Ash
Variance explained on the testing dataset: 0.007803860786620653
Water
Variance explained on the testing dataset: 0.028437995103204106
Superplasticizer
Variance explained on the testing dataset: 0.11487819917456632
Coarse Aggregate
Variance explained on the testing dataset: -544748.3420680033
Fine Aggregate
Variance explained on the testing dataset: -6460.95050638732
Age
Variance explained on the testing dataset: -0.8708838295047261

2.2.2 multivariate regression: raw data

Variance explained on the testing dataset: 0.4041892776655298(standardized)
Variance explained on the training dataset: -2.8817420624297454 (not standardized)

2.2.3 univariate regression: pre-processed data

Cement
Variance explained on the testing dataset: 0.2418292928498601

Blast Furnace Slag
Variance explained on the testing dataset: 0.03829535032724951
Fly Ash
Variance explained on the testing dataset: 0.023178258591711742
Water
Variance explained on the testing dataset: 0.04987308767082699
Superplasticizer
Variance explained on the testing dataset: 0.11120337777671929
Coarse Aggregate
Variance explained on the testing dataset: -544748.3420680033
Fine Aggregate
Variance explained on the testing dataset: -6460.95050638732
Age
Variance explained on the testing dataset: -0.5727325146769036

2.2.4 multivariate regression: pre-processed data

Variance explained on the testing dataset: 0.6085246664730262

2.3 Plot

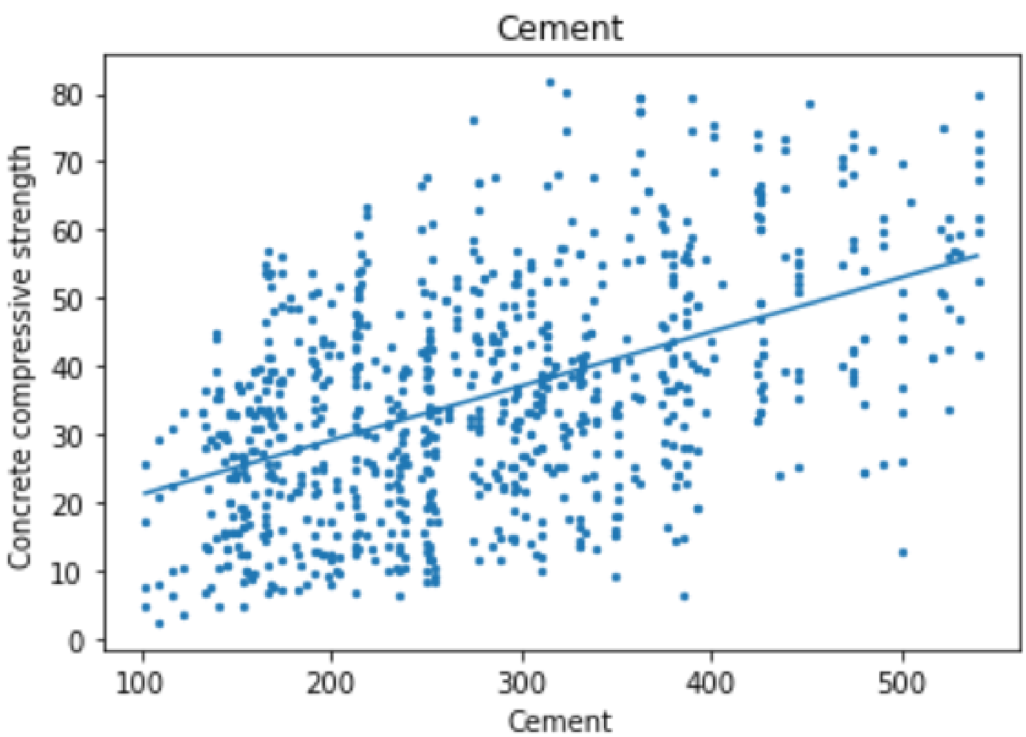

Figure 6, Cement (kg in a m3 mixture)

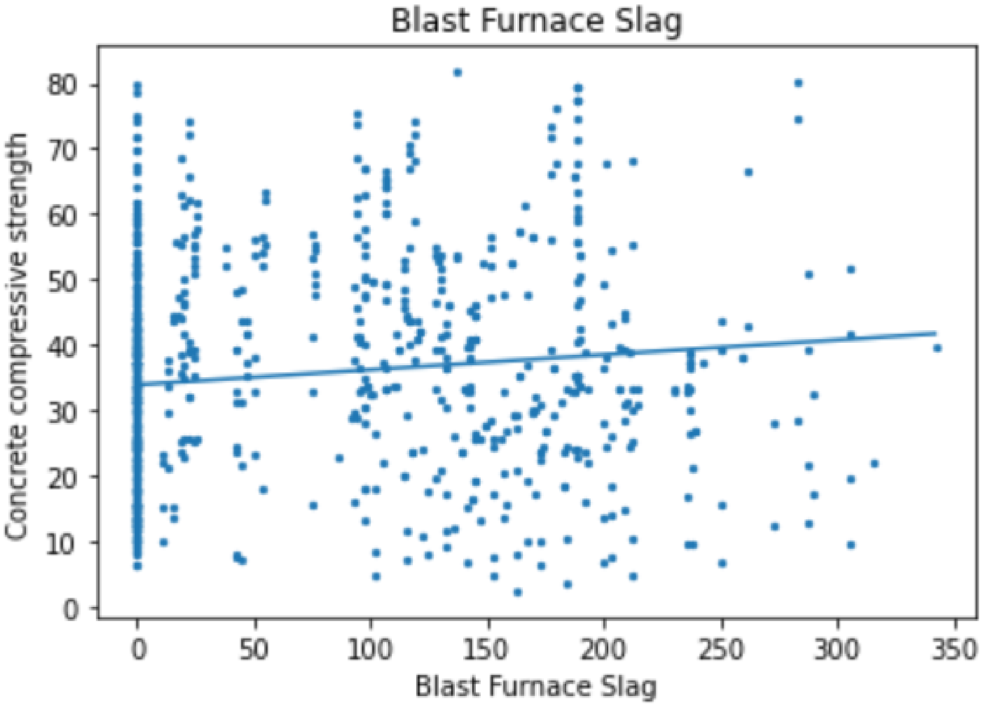

Figure 7, Blast Furnace Slag (kg in a m3 mixture)

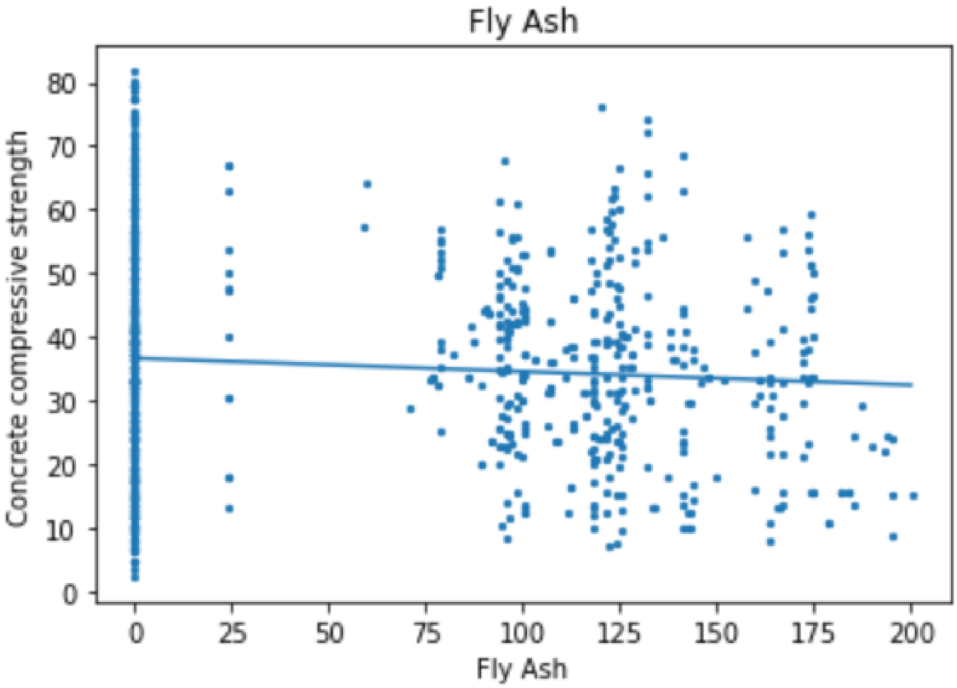

Figure 8, Fly Ash(kg in a m3 mixture)

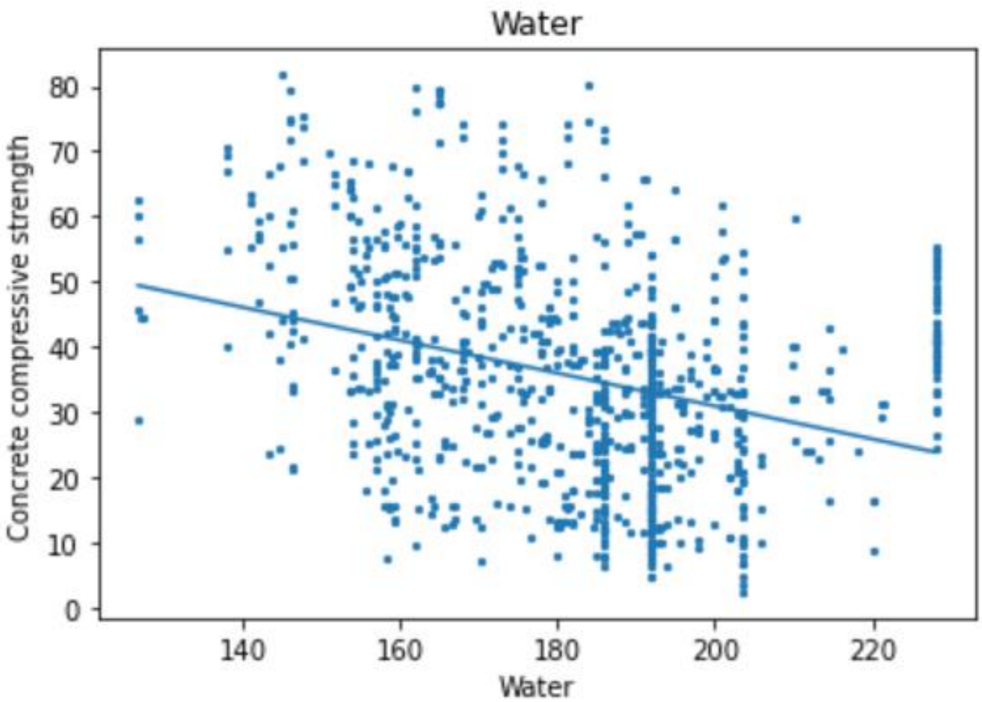

Figure 9, Water (kg in a m3 mixture)

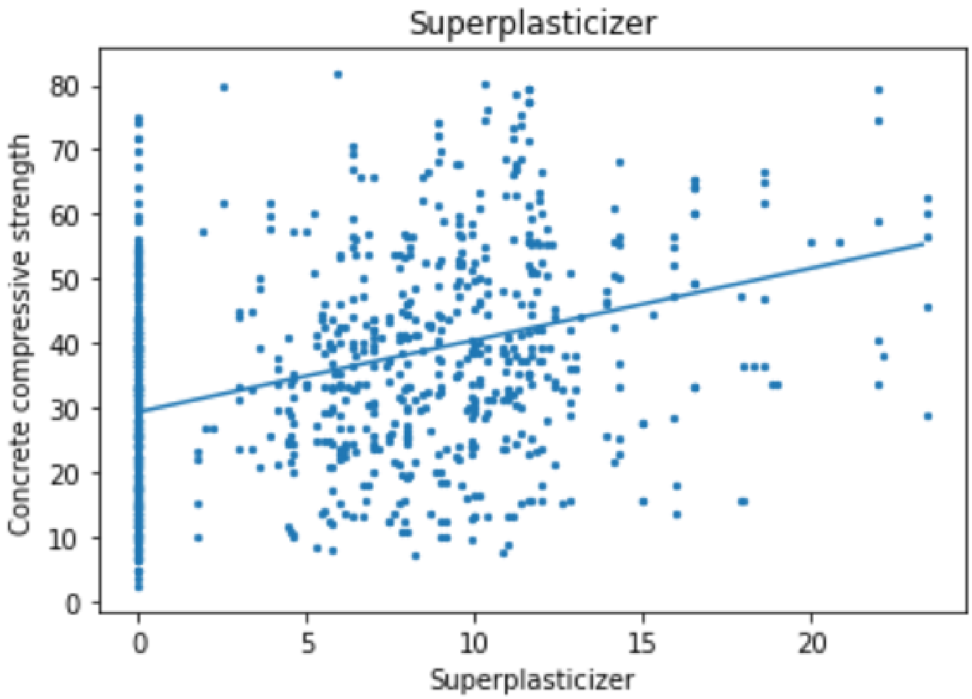

Figure 10 Superplasticizer (kg in a m3 mixture)

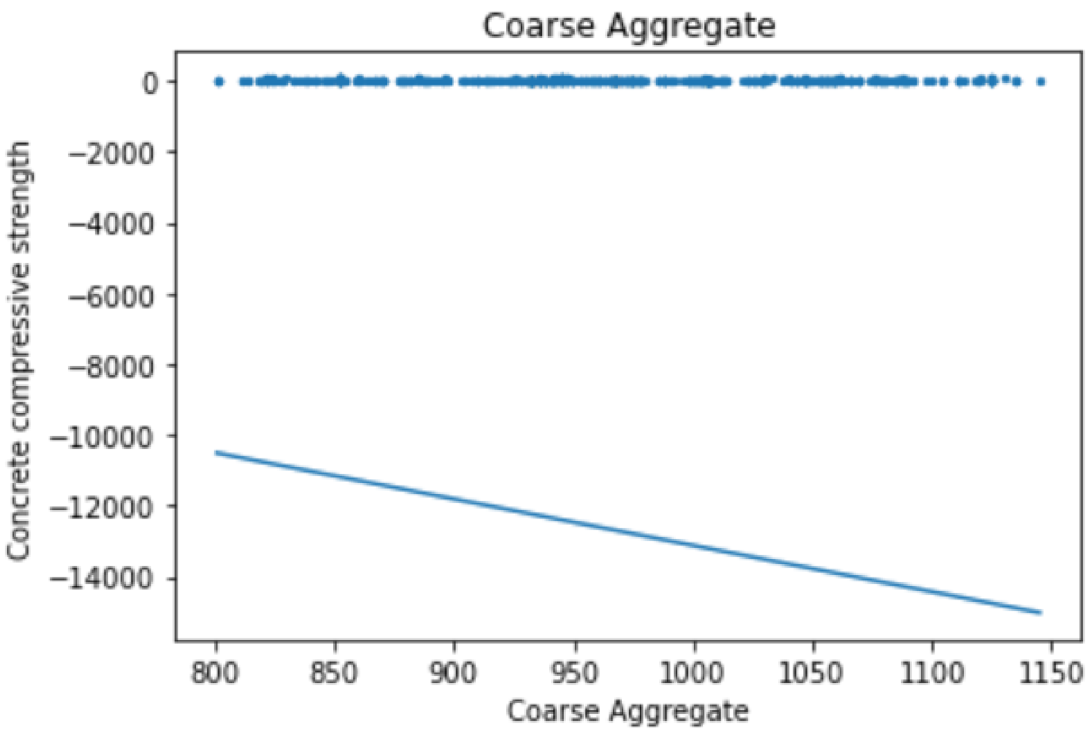

Figure 11, Coarse Aggregate (kg in a m3 mixture)

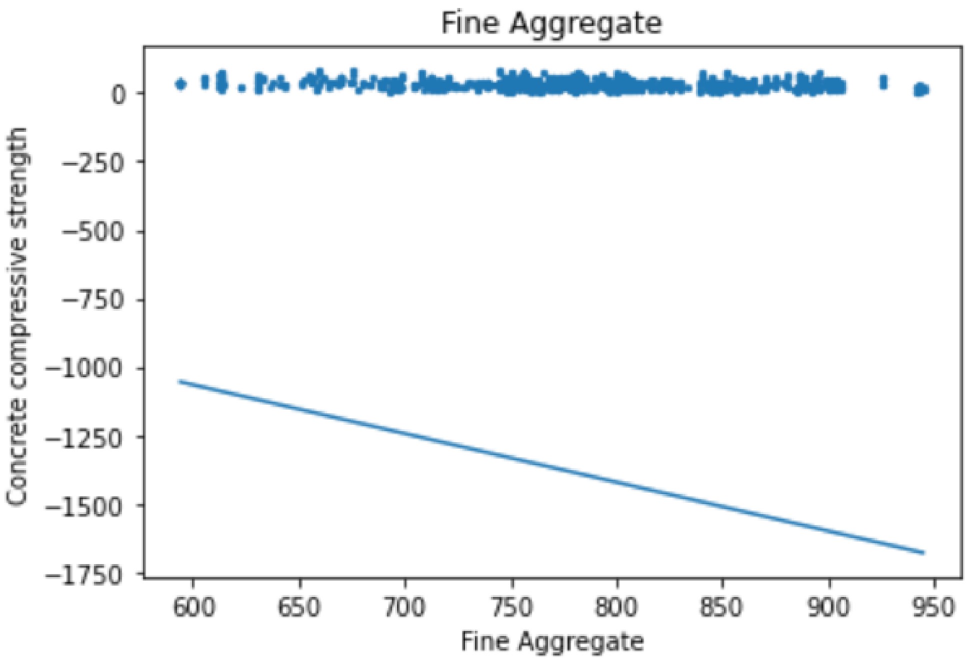

Figure 12, Fine Aggregate (kg in a m3 mixture)

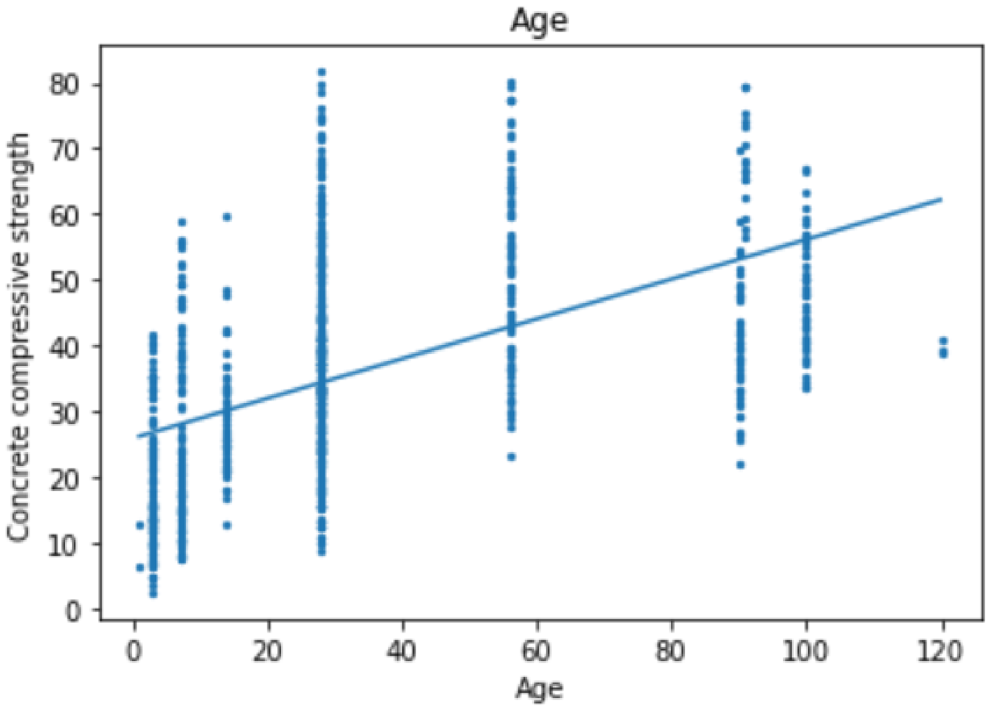

Figure 12, Age (day)

2.4 (Option1)Variance explained: MAE

2.4.1 univariate regression: training data

Cement
Variance explained on the training dataset: 0.18236600840777903
Blast Furnace Slag
Variance explained on the training dataset: -2.405754707577723
Fly Ash
Variance explained on the training dataset: -2.71811246893949
Water
Variance explained on the training dataset: -0.2091419778948237
Superplasticizer
Variance explained on the training dataset: -2.276082892123231
Coarse Aggregate
Variance explained on the training dataset: -0.08267511975398642
Fine Aggregate
Variance explained on the training dataset: -0.1285015002044996
Age
Variance explained on the training dataset: -0.7786927926837799

2.4.2 multivariate regression: training data

Variance explained of multivariate regression models on training data: 0.625815774401514

2.4.3 univariate regression: testing data

Cement
Variance explained on the testing dataset: 0.14748243473662762
Blast Furnace Slag
Variance explained on the testing dataset: - 2.1125698619042814
Fly Ash
Variance explained on the testing dataset: -3.190842299716907
Water
Variance explained on the testing dataset: -0.22232951479634544
Superplasticizer
Variance explained on the testing dataset: -2.453970388872871
Coarse Aggregate
Variance explained on the testing dataset: -0.1178951978476935
Fine Aggregate
Variance explained on the testing dataset: -0.1511333810564888
Age
Variance explained on the testing dataset: -0.8523853309749477

2.4.4 multivariate regression: testing data

Variance explained of multivariate regression models on testing data: 0.6963324828490879

2.5 (Option2)Variance explained: Ridge Regression

2.5.1 univariate regression: training data

Cement
Variance explained on the training dataset: 0.18236600840777903
Blast Furnace Slag
Variance explained on the training dataset: 0.013628927388471442
Fly Ash
Variance explained on the training dataset: 0.006700191405976424
Water
Variance explained on the training dataset: 0.10011968260397719
Superplasticizer

Variance explained on the training dataset: 0.11941966025716717
Coarse Aggregate
Variance explained on the training dataset: -223.5801138191168
Fine Aggregate
Variance explained on the training dataset: -5.307176689030549
Age
Variance explained on the training dataset: 0.2531929721903466

2.5.2 multivariate regression: training data

Variance explained of multivariate regression models on training data: 0.7127438425051839

2.5.3 univariate regression: testing data

Cement
Variance explained on the testing dataset: 0.235953132034705
Blast Furnace Slag
Variance explained on the testing dataset: 0.03083992419657654
Fly Ash
Variance explained on the testing dataset: 0.015609284877684764
Water
Variance explained on the testing dataset: 0.04250834199014793
Superplasticizer
Variance explained on the testing dataset: 0.10430758022628554
Coarse Aggregate
Variance explained on the testing dataset: -66605.45014122291
Fine Aggregate
Variance explained on the testing dataset: -574.2385473659172
Age
Variance explained on the testing dataset: 0.2531929721903466

2.5.4 multivariate regression: testing data

Variance explained of multivariate regression models on testing data: 0.7295609071683106

3Discussion

3.1 Compare and contrast your models.

3.1.1 Did the same models that accurately predicted the training data also accurately predict the testing data?

Some models predicted the testing data accurately, but some models did not. For example, the variance explained of age model is positive on training data, but is negative on testing data.

When using MAE, the performance of MAE models is not good on preprocessed data. Only multivariate model and univariate model of Cement have a positive variance explained. These two models on testing data also got a variance explained.

When using Ridge Regression, the models that accurately predicted the training data also accurately predict the testing data. These models include multivariate model and univariate model of Cement, Blast Furnace Slag, Fly Ash, Water, Superplasticizer and Age

3.1.2Did different models take longer to train or require different hyperparameter values?

Yes. For example, I set the learning rate for univariate as 0.00001, but for multivariate as 0.1. Since for univariate models, if I use 0.1, I'll get negative variance explained. And for multivariate models, I'll get negative variance explained when use 0.00001 as learning rate.

When using MAE, learning rates are also different for univariate and multivariate models. In addition, some models just need about 300 steps to converge, some models cannot converge after 200000 steps.

When using Ridge Regression, besides learning rate, the lambda value is also different for univariate and multivariate models.

3.1.3 How did pre-processing change your results or optimization approach?

No matter using MSE, MAE or Ridge Regression, the variance explained is better on the data which is replaced outliers than on the raw data.

In addition, when training univariate models with MSE, MAE or Ridge Regression, if I use the data after standardization, I cannot get positive variance explained. So, I train the univariate models on the data which is not standardized.

3.2 Draw some conclusions about what factors predict concrete compressive strength. What would you recommend for making the hardest possible concrete?

Firstly comparing the variance explained among several models using MSE, MAE or Ridge Regression, we can find the none univariate models is better than multivariate models. It means that we'd better not conclude the concrete compressive strength only based on one factor.

Then, for MSE, final estimate of b and m: -0.0017760885150005443 [0.6692779521994532, 0.39037369316614534, 0.19809771676483542, -0.1836265027253171, 0.07460770616951481, -0.007422413024896934, -0.046961730435405004, 0.5128194355414].

For MAE, final estimate of b and m: -0.12963284493758204 [0.8849591734592834, 0.5552779828450896, 0.2383112822073779, 0.3156984423377398, 0.462127532298658, 0.27109534424783954, 0.24108739676388863, 0.5295181082228482]

For Ridge Regression final estimate of b and m: -0.0022576339343107652. [0.6214290851919095, 0.34577336570096295, 0.15867223227760113, -0.2063108559663113, 0.07972011898301898, -0.03283542618163383, -0.0806185065281442, 0.5083499316297132].

We can conclude from the result that if we want to make the hardest possible concrete, we'd better increase the quantity of Cements and length of age as possible as we can since these factors contribute to strength best. In addition, Coarse Aggregate may reduce the strength. Finally, other factors may also help increase the strength.