黃偉祥 X1136010

Loss function : (Y\_true – Y\_predict)2 + \*|w|2

Gradient : 2

Update Weight = Weight – learning\_rate \* Gradient

Validation Loss = (Yvalidation – Yvalidation\_predict)2 / number of Y

Basic part:

Y\_predict = Xin \* W + noise

Weight\_dimentions : {noise , Xin}

Advanced part:

X’ = {noise , X0 , X1 , X2 , X3 , X4 , X5 , X6}

Ypredict = noise + X0\*W0 +X1\*W 1 + X2\*W2 + X3\*W3 + X4\*W4 + X5\*W5 + X6\*W6

Weight\_dimentions : {noise , X0,X1,X2,X3,X4,X5,X6}

In basic part the input of X is only {weight} variable, but in advanced part we have 7 variables which are {age , gender , height , weight , bodyfat , diastolic , systolic} as X.

First, I don’t know why my python cannot recognize the function np.isnan(), so I change my data into pandas.DataFrame and use fillna() to replace missing value(np.nan).

Second, I realize that my model doesn’t work so well, then I start to check how to make model perfect, but then I found that the main point is the outliers, so I remove the outliers using IQR and setting lower bound as Q1 – 1.5\*IQR and upper bound as Q3 + 1.5\*IQR.

Third, the second column of advanced part (gender) is given a string(F or M), then it cannot be doing mathematical operation, so I change it to 0.0 and 1.0.

Fourth, I tried to apply the non-linear basis functions to X, but only improve little in the result.

Last but not least, I realize that my model keep running even after convergences, and this is wasting a lot of time and may cause to overfitting, so I used the loss of validation dataset to have a early stopping point which are:

1. When the loss of validation dataset is not going down. (loss > previous\_loss)
2. When the loss is going down very little. (previous\_loss – loss < 0.0000001)

Last, I realize that each time of my result will be different, so I set the random seed and run a for loop to get the best random seed of my model.

In conclusion, I tried to apply some features engineering to the dataset and set some early stopping point to prevent overfitting and time waste.