Assignment 1 Yingnan Zhao id: 260563769

Question 1

1.1

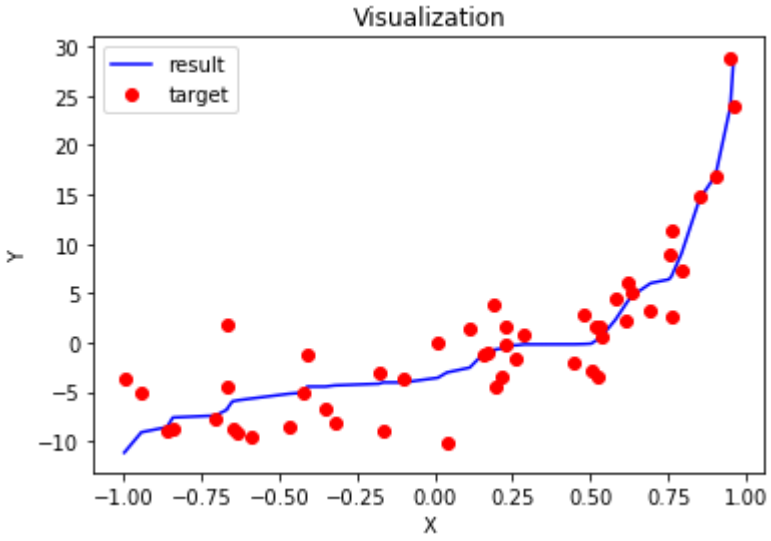
Calculated training MSE and Validation MSE

```
training set MSE = 6.47477662124
validation set MSE = 1418.56775118
```

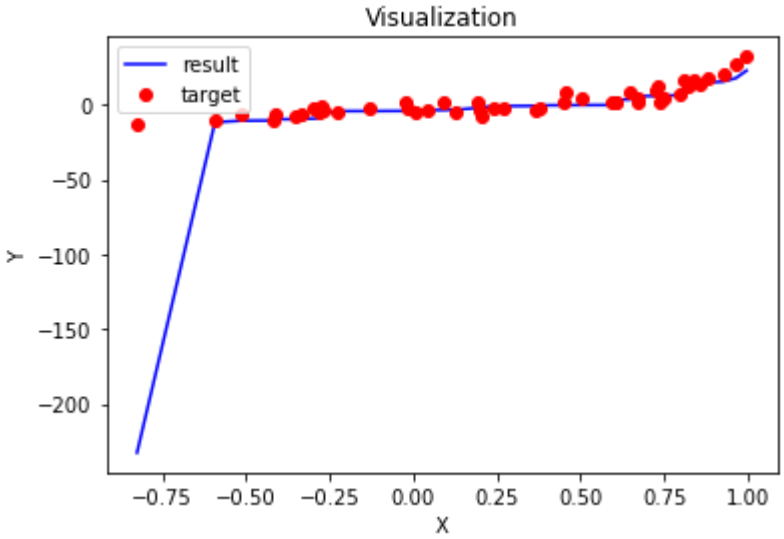
Appearently the MSE Error for the validation set is not very good, we will inspect further when we visualize the fit

Visualizations

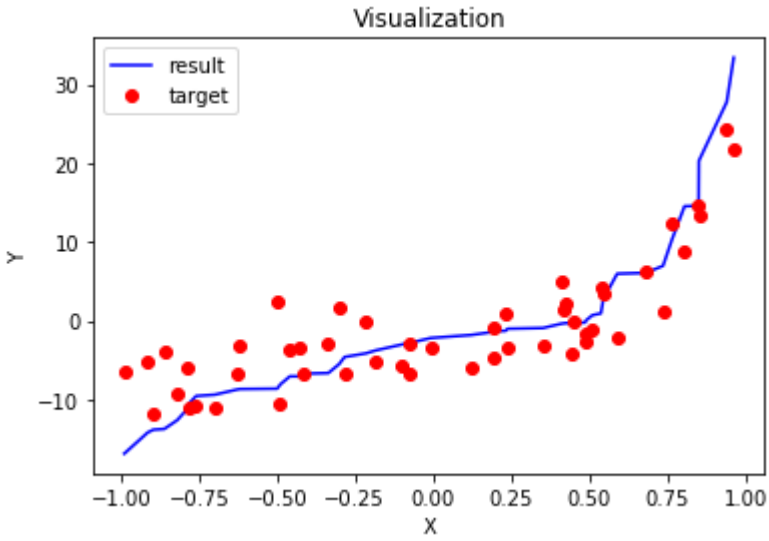
Visualize the fit of the training dataset



Visualize the fit of the validation dataset



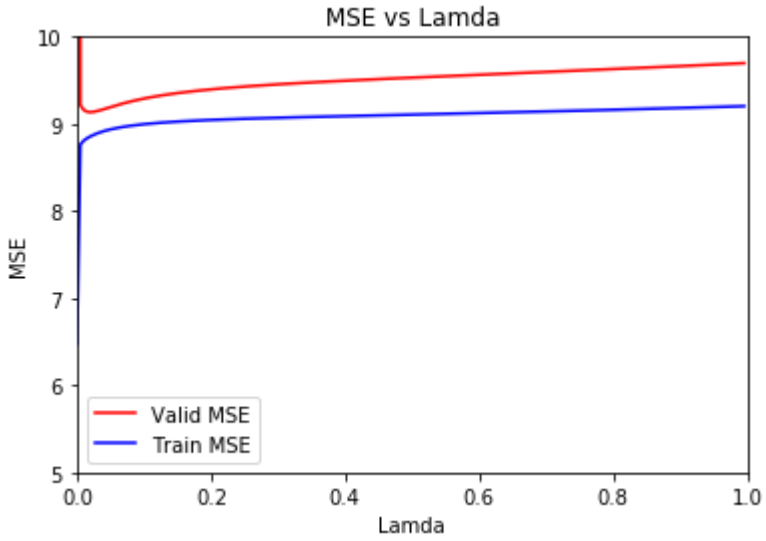
Visualize the fit of the test set



The visualization of the training set and the test set is good. However, for the validation set, as shown above the all the point except for the first one fits well, the reason is the unregulated linear regression model trained by the training set cannot handle the x input x = 0.996344491497.

1.2 L2 regulation

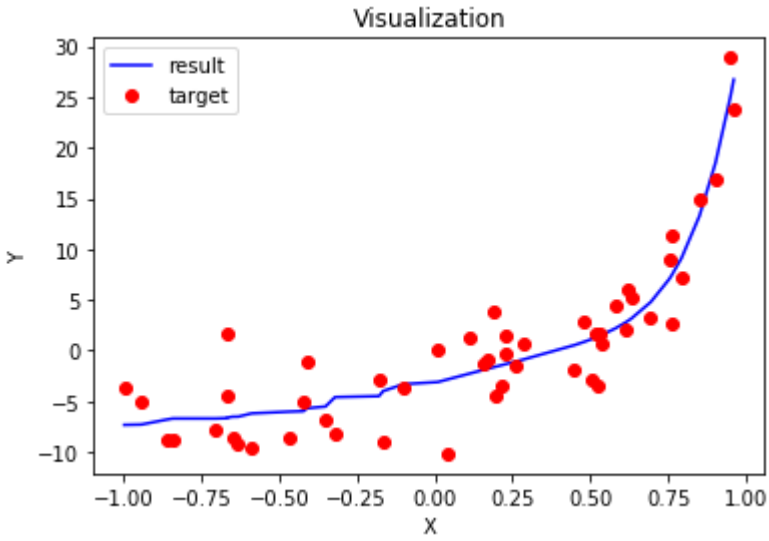
Plot the training MSE and Validation MSE againist lamda, using a for loop to search linearly from lamda = 0 to lamda = 1.



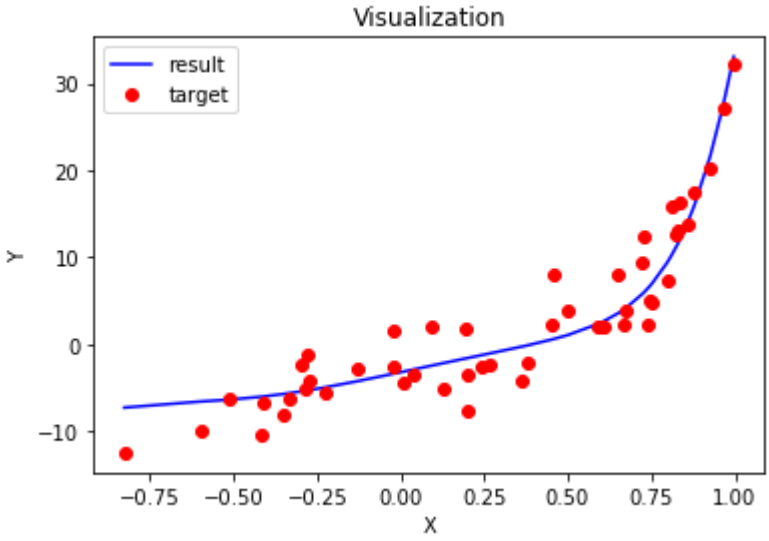

```
the lamda that has minimum error = 0.02, where the MSE for the validation set i
s 9.13509878469
```

The optimum lamda = 0.02, then we will use this value to visualize the fit on the three data set respectively

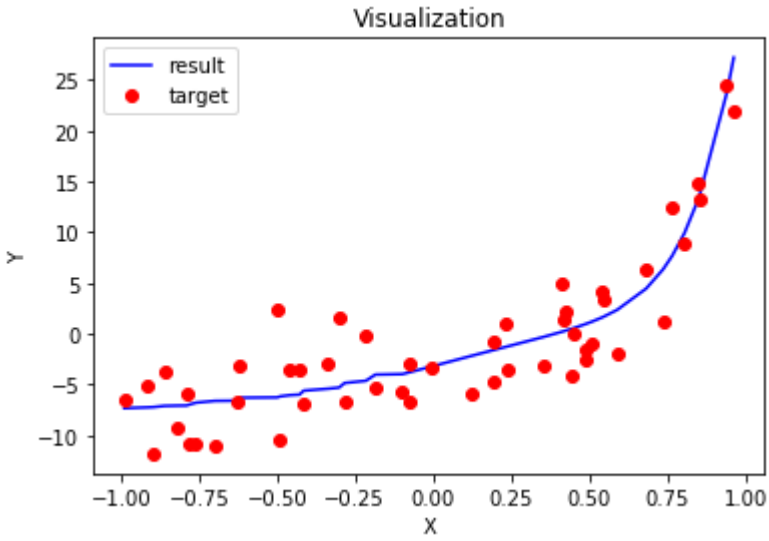
training set



validation set



test set



after adding the L2 regulation, the quality of the fit for the validation set and test set has been greatly improved compare to the nonregulated model

1.3

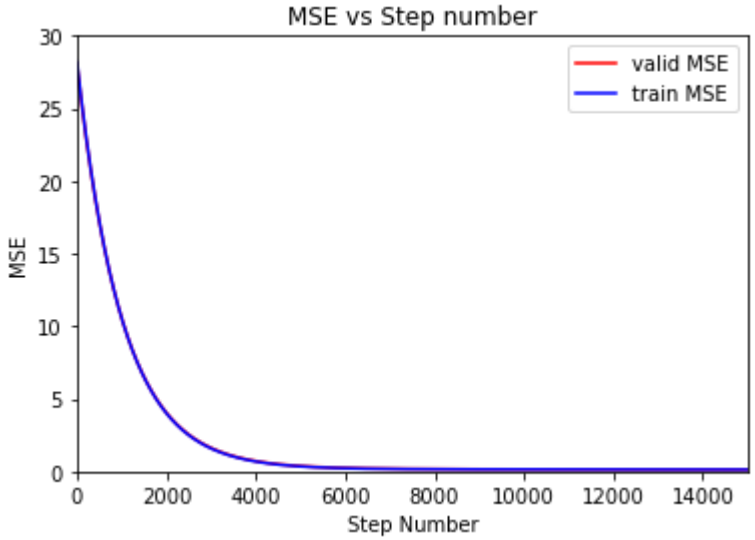
The source does not have a large degree of polynomial, from the visualization shown above, the source looks like a exponential curve

Question 2

2.1

Applying the gradient decent method to our data, using the equation y = w0 + w1x

The learning curve (MSE vs step plot for both training set and validation set) is shown below for w in 15000 steps(1e-6 each step)

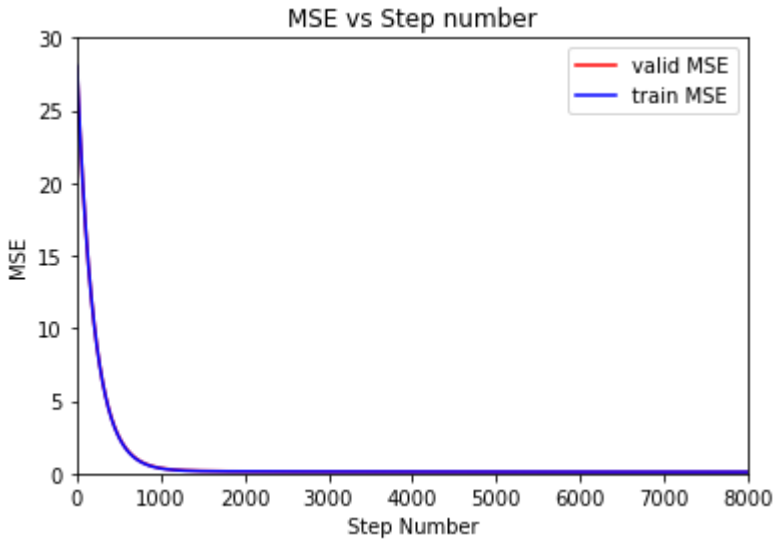


the MSE is droping dramatically as the step number increases

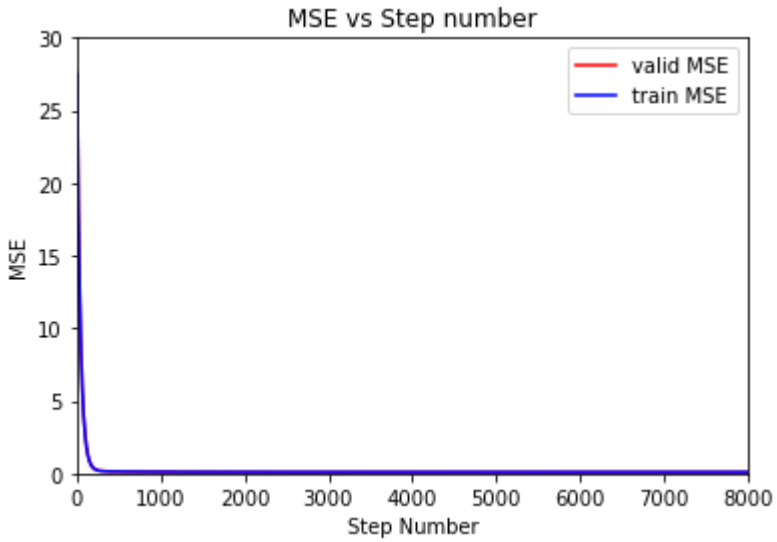
2.2

here we will try different step size to find the optimal one, the goal is to try to find the one with the quickest drop

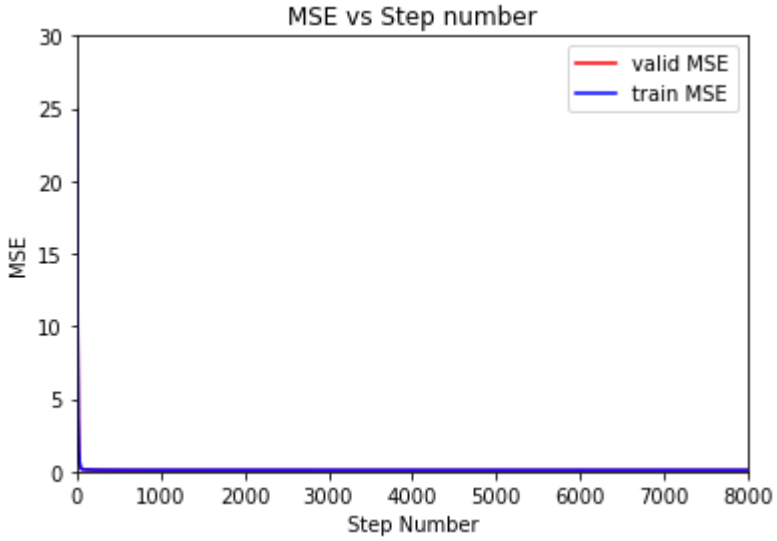
Step Size = 4.9999999999999996e-06



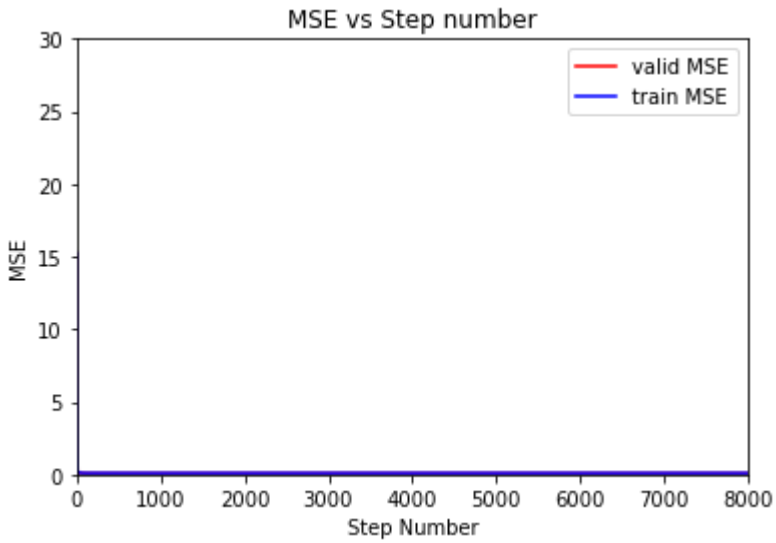
Step Size = 2.4999999999999998e-05



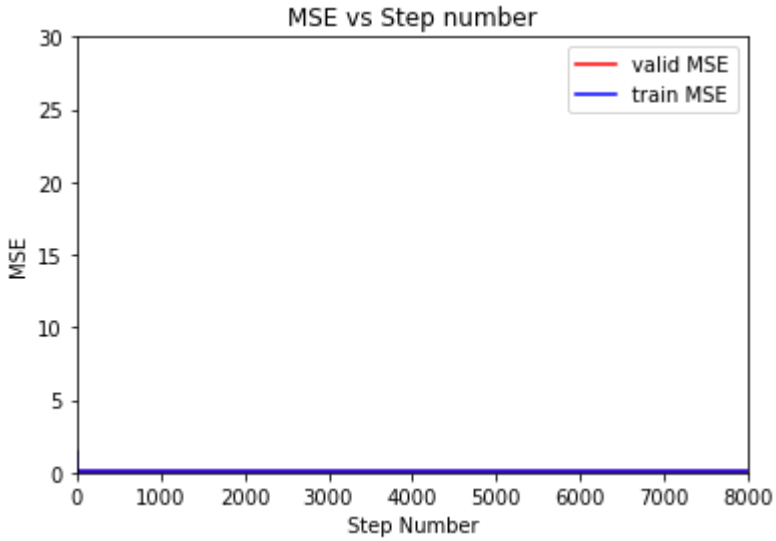
Step Size = 0.000125



```
Step Size = 0.000625
```




```
Step Size = 0.0031249999999999997
```

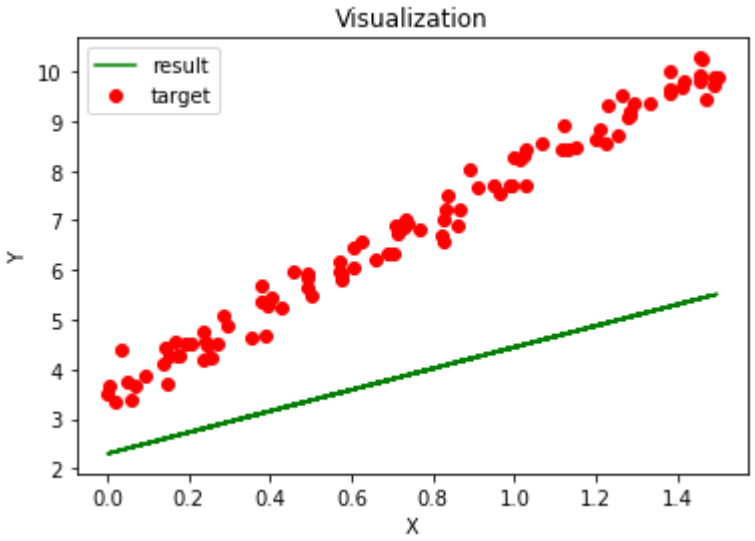


As shown above the largest step we can take is 0.000125 where we can make the function converge in around 100 steps

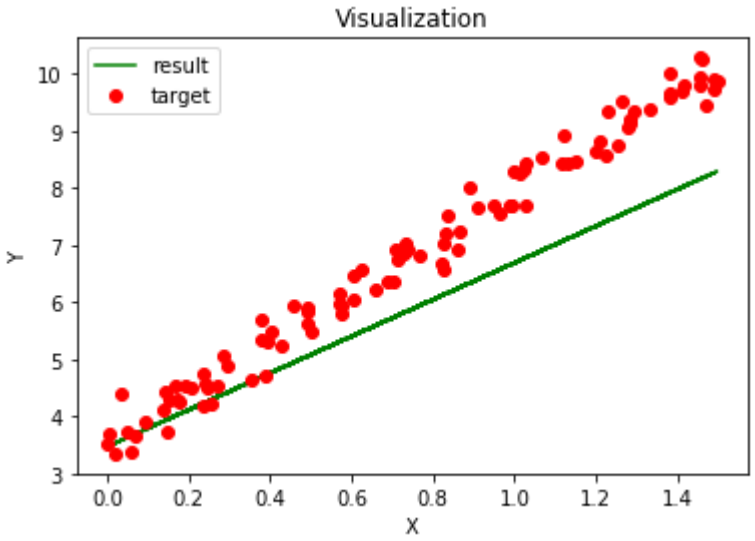
2.3

visualize the training process, the 5 visualizations below shows how the fit changes as we increase the steps

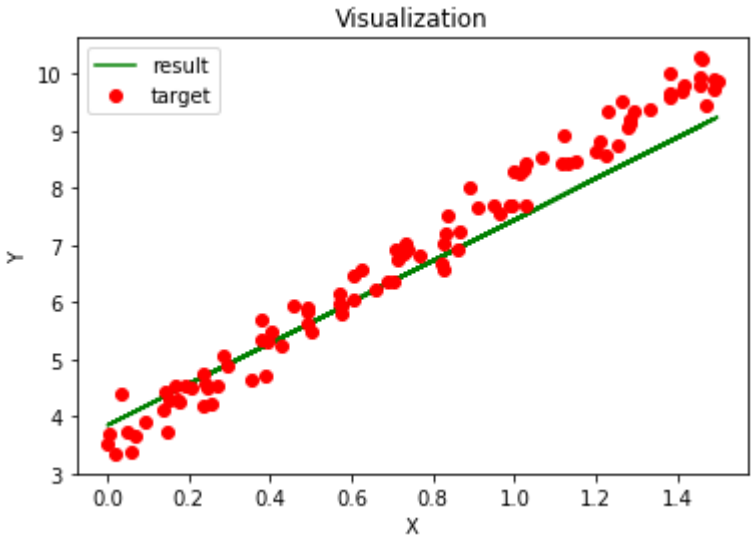
step number = 1100



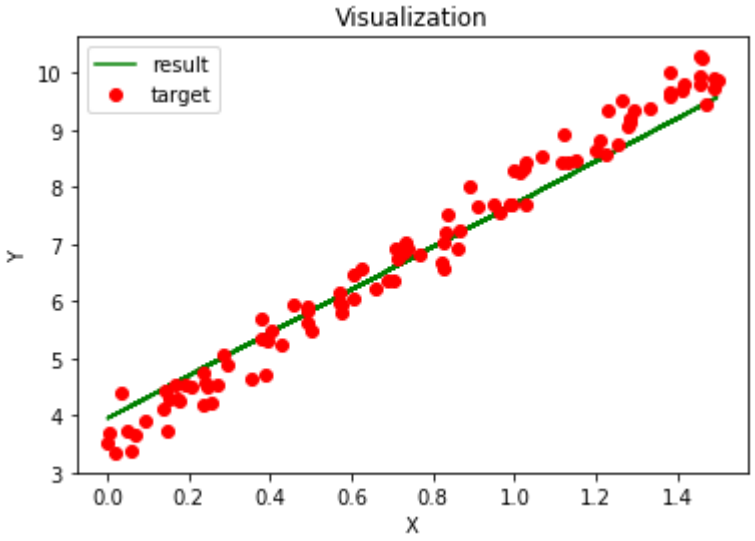
step number = 3300



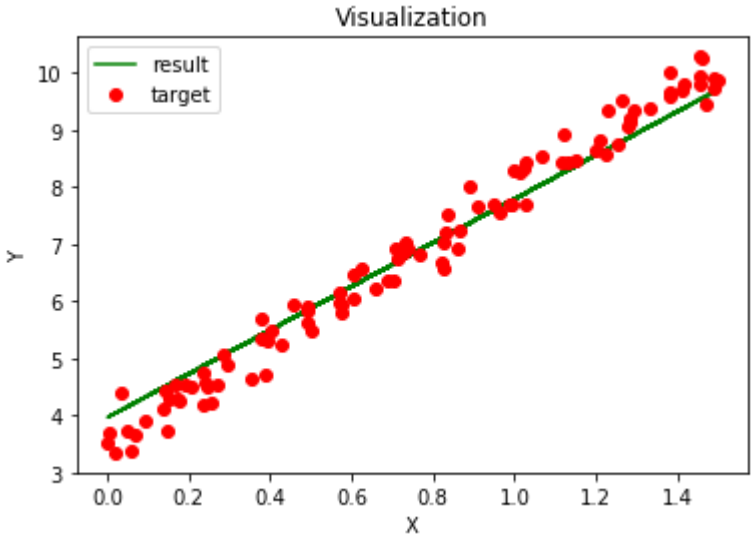
step number = 5500



```
step number = 7700
```




```
step number = 9900
```



As shown above as the training progresses (increasing step number), the visualization gets better.

Question 3

3.1

A function has been created to handle the input data, we have the data itself and the statics of the data in two separate csv files, first the function will complete the dataset by filling all missing attributes using the median of the column. After that, the function will shuffle the data and store them in to 5 pairs of training and testing sets using the 80-20 splits. The mean value is an ok choice, however I believe the median could be a better choice. Since there could be outliers in the dataset, and median is not affect by it. And we could also randomly choose any existing value to fill in the blank to introduce more randomness. The data files are stored in the Dataset folder.

3.2

Here are the results produced by running the code on 5 different 80-20 splits, and parameter learnt is stored in a csv file called ParameterLearntQ3-2.csv in the Dataset folder, in the file there are 123 rows and 5 columns, each row represents a parameter (122 features and one noise term where x = 1)and each column represents an 80-20 split data set.

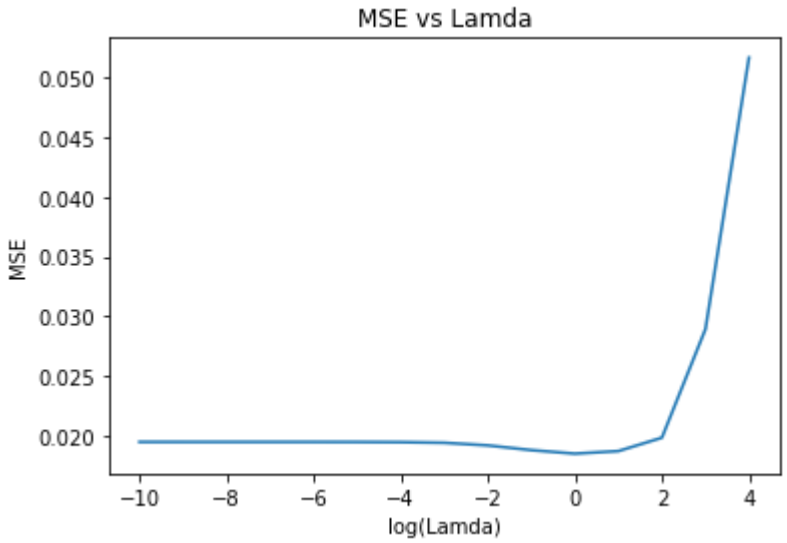
```
the MSE for 80-20 split 1 is 0.0224269076138
the MSE for 80-20 split 2 is 0.019497119073
the MSE for 80-20 split 3 is 0.0183477433374
the MSE for 80-20 split 4 is 0.0197028885344
the MSE for 80-20 split 5 is 55.1123236905
```

The best MSE achieced is 0.0183477433374 on 80-20 split 3

3.3

To find the optimal lamda, the value of lamda has been varied MSE average has been calculated accordingly, the lamda that produce the smallest MSE average is the optimal one.

find minimum lamda



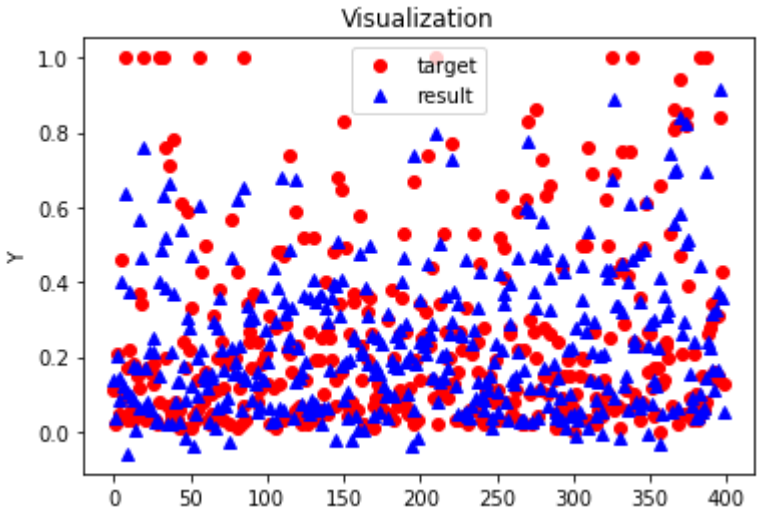

```
the optimal lamda is = 1.0, produce an Average MSE of 0.0185067816579
```

From the parameter matrix w, we could tell some features are redundant or irrelevant. The weight for irrelevant feature must be low thus they can be removed. However, according to the source, all the data has been normalized in a way that does not preserve relationships between values BETWEEN attributes, so there will not be strong corrlation between features.

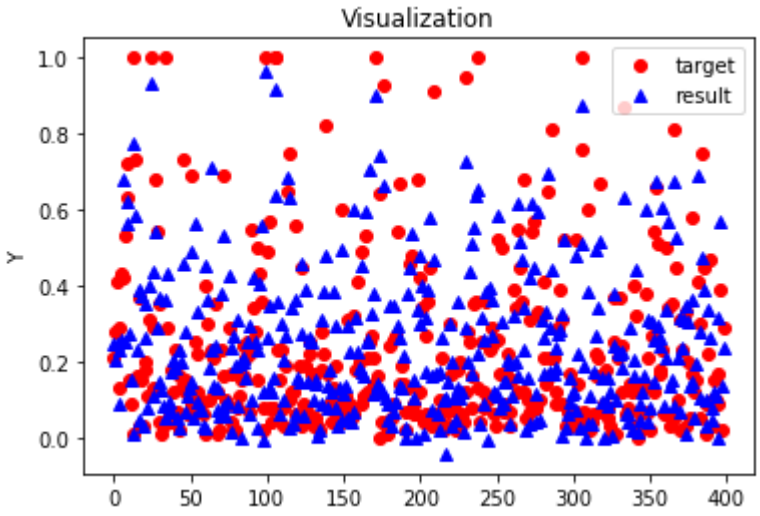
We get the w matrix using lamda = 1.0 and analysis which feature is irrelevant, to analysis the features, the features are stored in a file called ParameterLearntQ3-3.csv in the same fasion as in question 3.2.

Visualizing the fit before feature selection using the optimum lamda

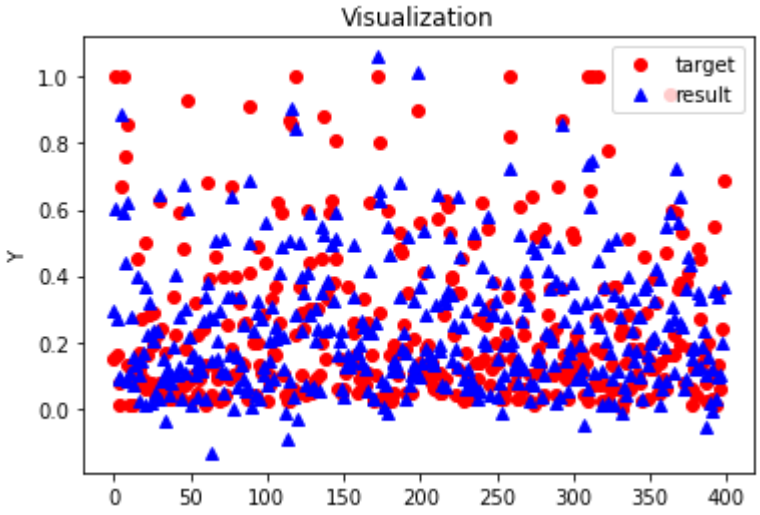
visualization for set 1



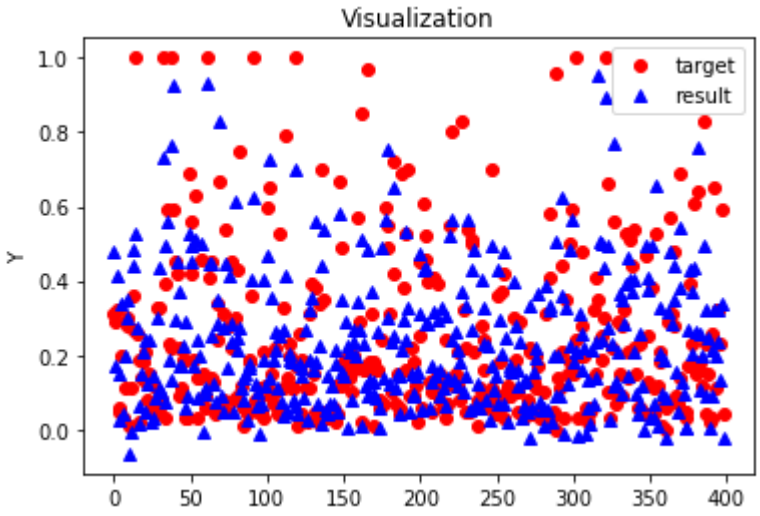
visualization for set 2



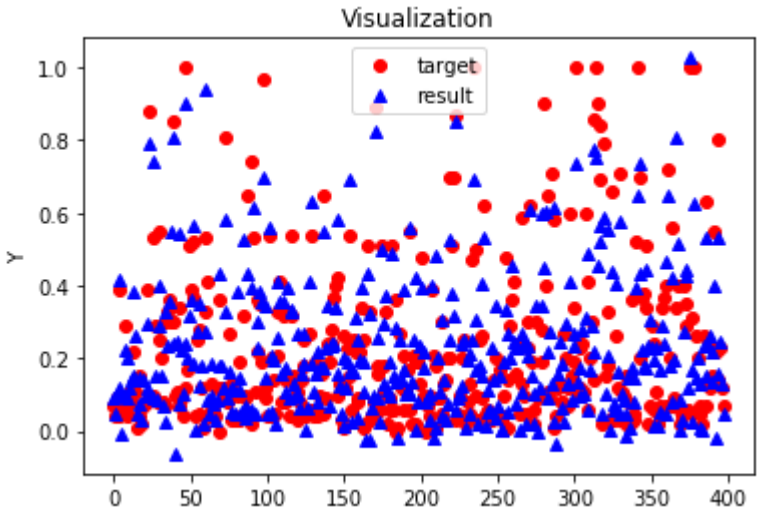
visualization for set 3



visualization for set 4



visualization for set 5



feature selection

The top 53 features that has the smallest absolute value of average weight when lamda equals to 1 will be considered irrelevant thus will be removed, only 70 features will remain. from the model, the MSEs and the fit will be analysised.

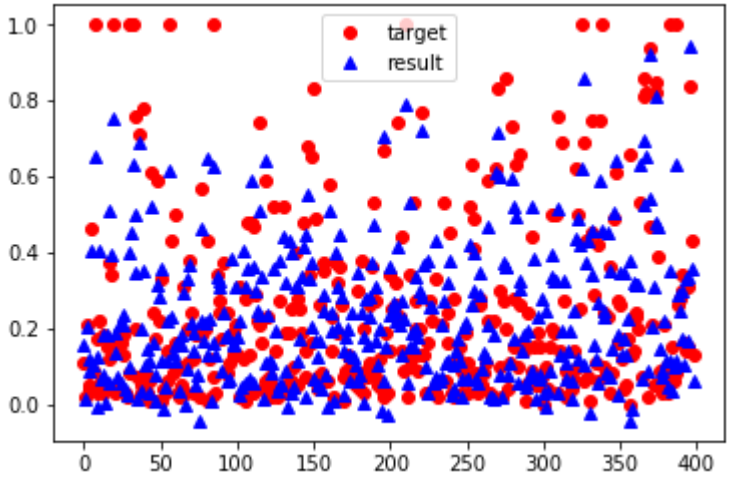
Here are the MSE for each individual data set

```
the MSE for 80-20 split 1 with feature selection is 0.021197917001
the MSE for 80-20 split 2 with feature selection is 0.0189699196655
the MSE for 80-20 split 3 with feature selection is 0.0186198389007
the MSE for 80-20 split 4 with feature selection is 0.0193600318562
the MSE for 80-20 split 5 with feature selection is 0.0187975785119
the mean MSE is 0.0193890571871
```

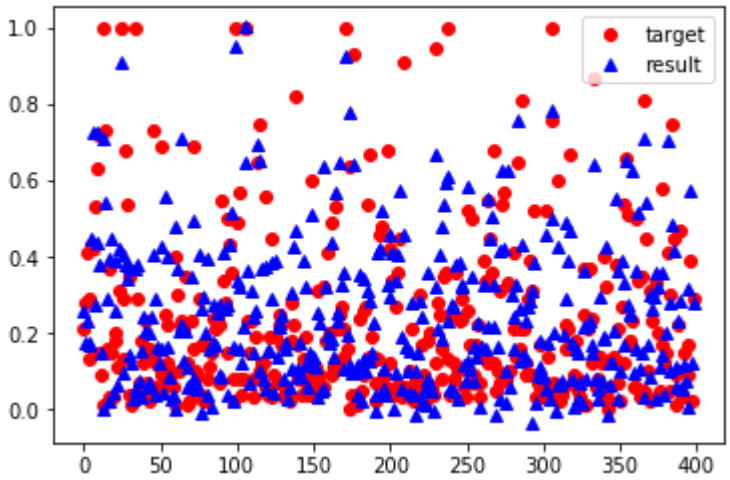
As shown above the MSE of the 70 feature model have not increase by a considerable amount, thus the output could be dominated by a number of features

Visualizing the fit for each 80-20 split set

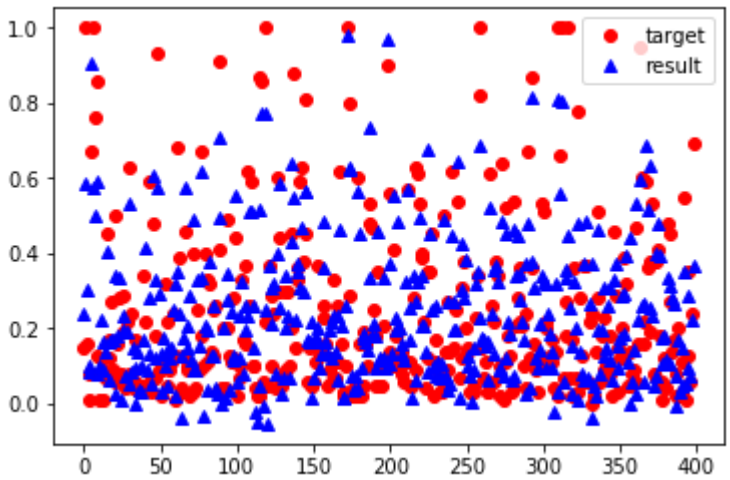
visualization for set 1



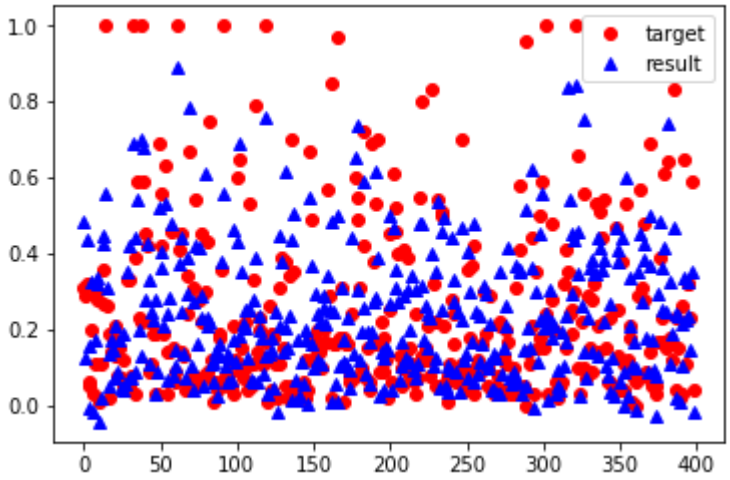
visualization for set 2



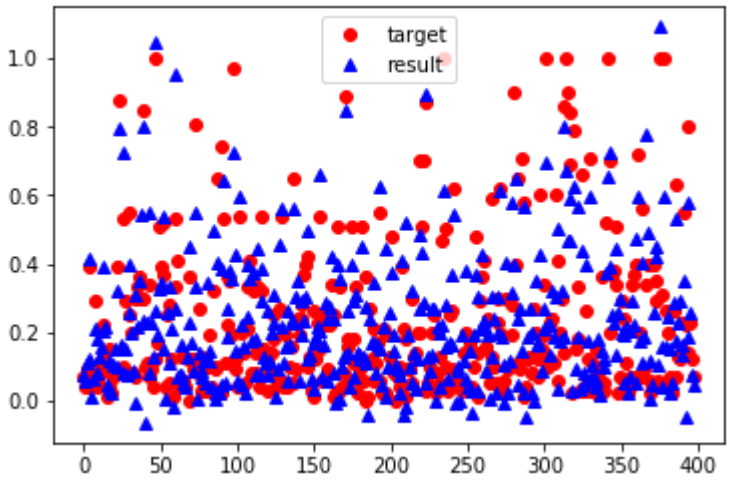
visualization for set 3



visualization for set 4



visualization for set 5



As shown above, the fits looks almost the same as the ones that we have not reduce any features, however the computing time has been reduced by a considerable amount (70 features vs 123 features). Feature selection can also prevent over fitting.