PHUNG THANH HAI (馮清海)
Student ID : 0880803

REPORT

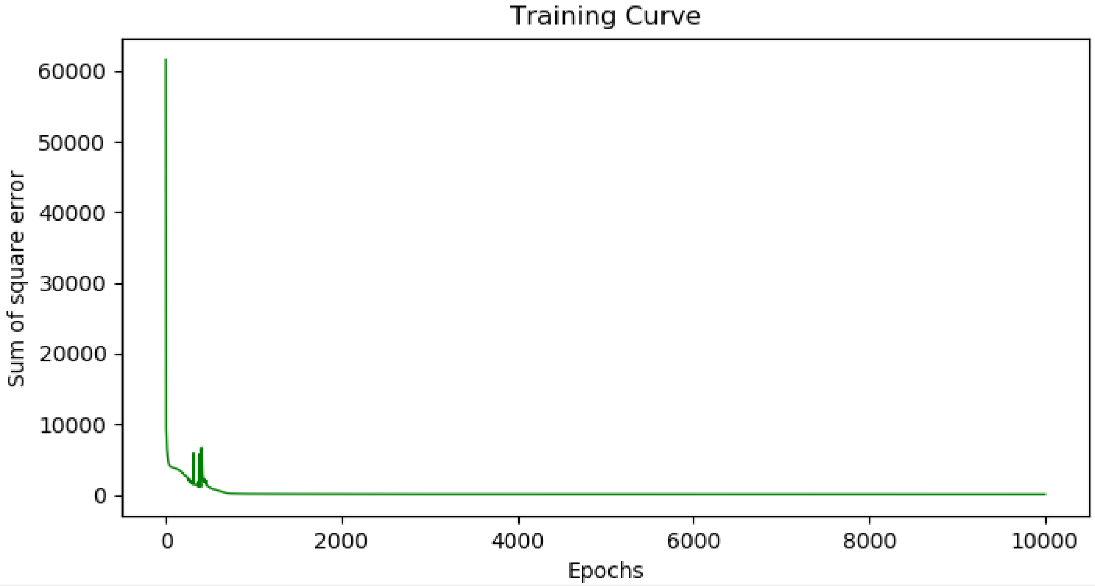
DEEP LEARNING HOMEWORK 1

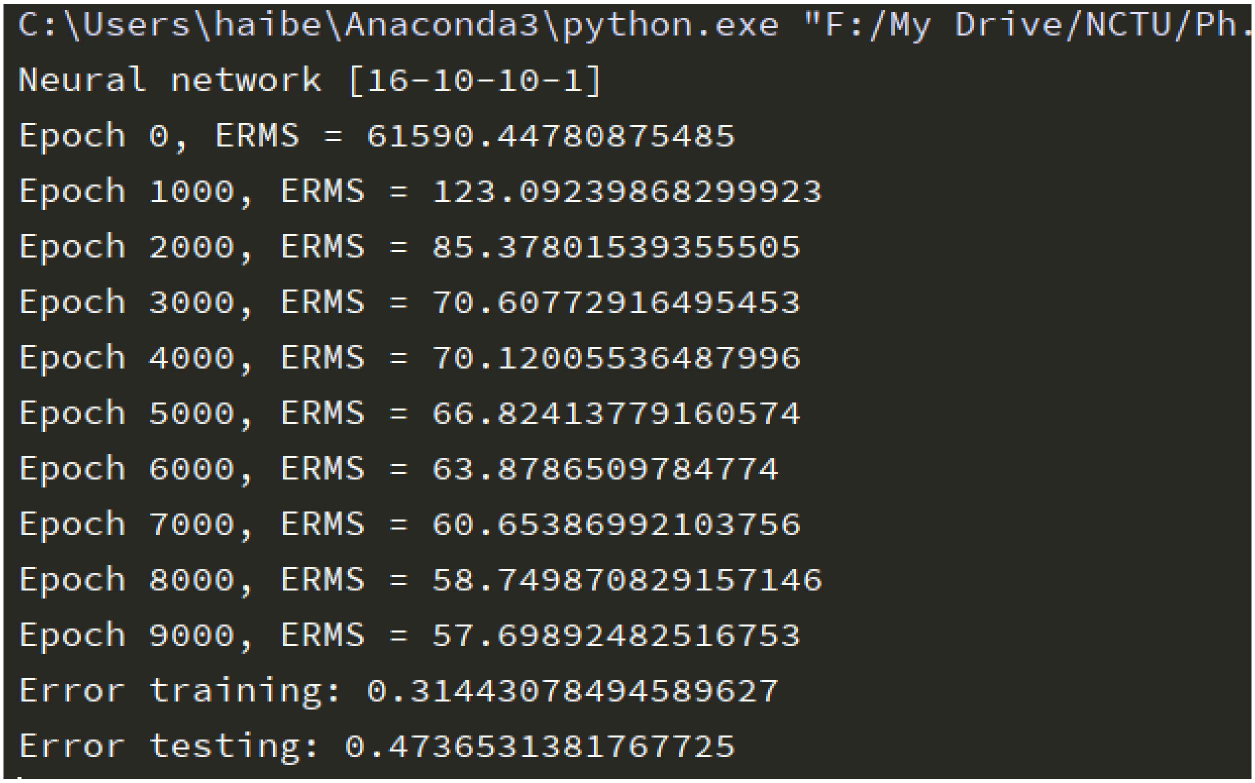
1. Regression

a. Network Architecture

Network Architecture	16 – 10 – 10 - 1
Selected features	in range (8)
Training ERMS	**0.31443078494589627**
Testing ERMS	**0.4736531381767725**
Epochs	10000
Learning rate	0.002
Batch Size	32
Training / Testing Size	75% - 25%

b. Learning Curve

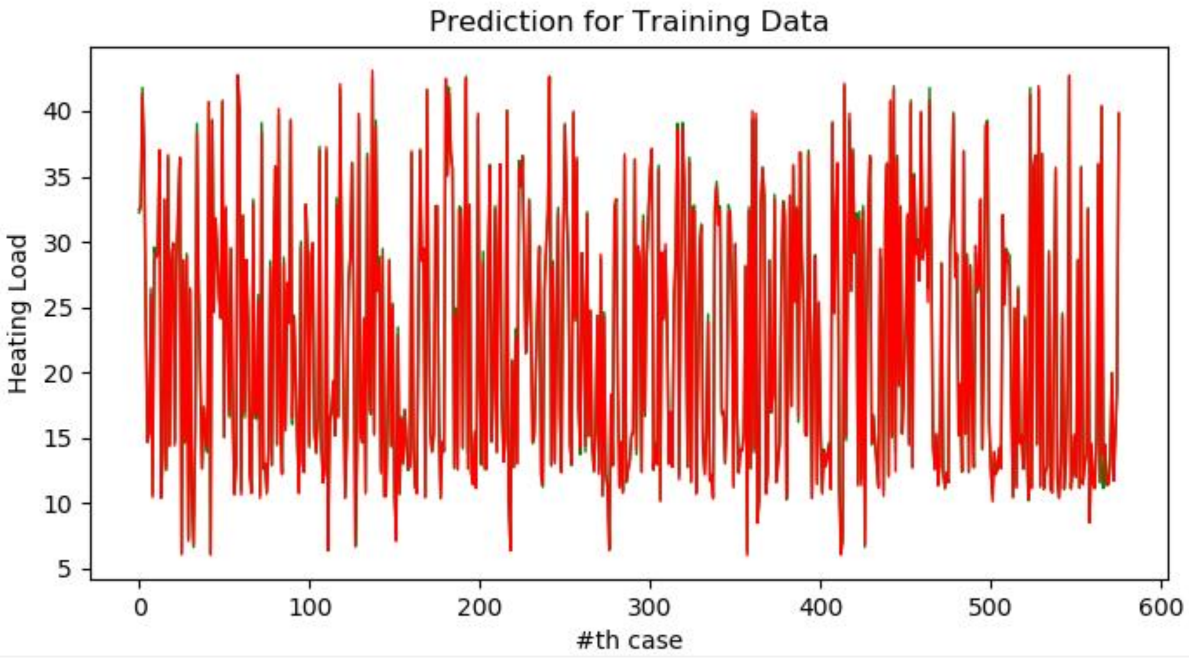




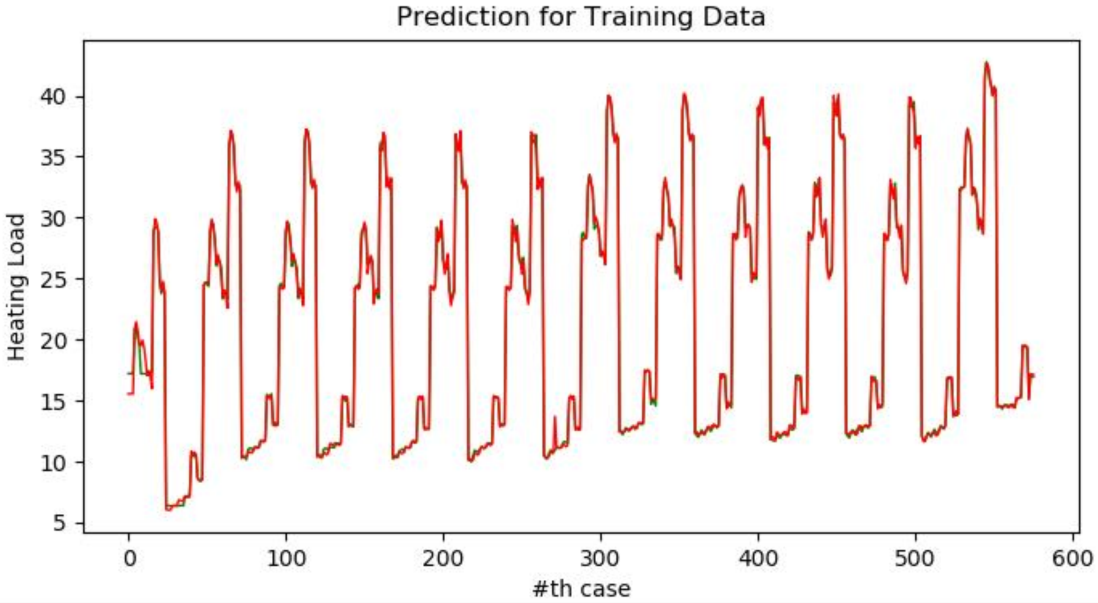
```
C:\Users\haibe\Anaconda3\python.exe "F:/My Drive/NCTU/Ph
Neural network [16-10-10-1]
Epoch 0, ERMS = 61590.44780875485
Epoch 1000, ERMS = 123.09239868299923
Epoch 2000, ERMS = 85.37801539355505
Epoch 3000, ERMS = 70.60772916495453
Epoch 4000, ERMS = 70.12005536487996
Epoch 5000, ERMS = 66.82413779160574
Epoch 6000, ERMS = 63.8786509784774
Epoch 7000, ERMS = 60.65386992103756
Epoch 8000, ERMS = 58.749870829157146
Epoch 9000, ERMS = 57.69892482516753
Error training: 0.31443078494589627
Error testing: 0.4736531381767725
```

c. Regression result with training label

- **Shuffle = TRUE**

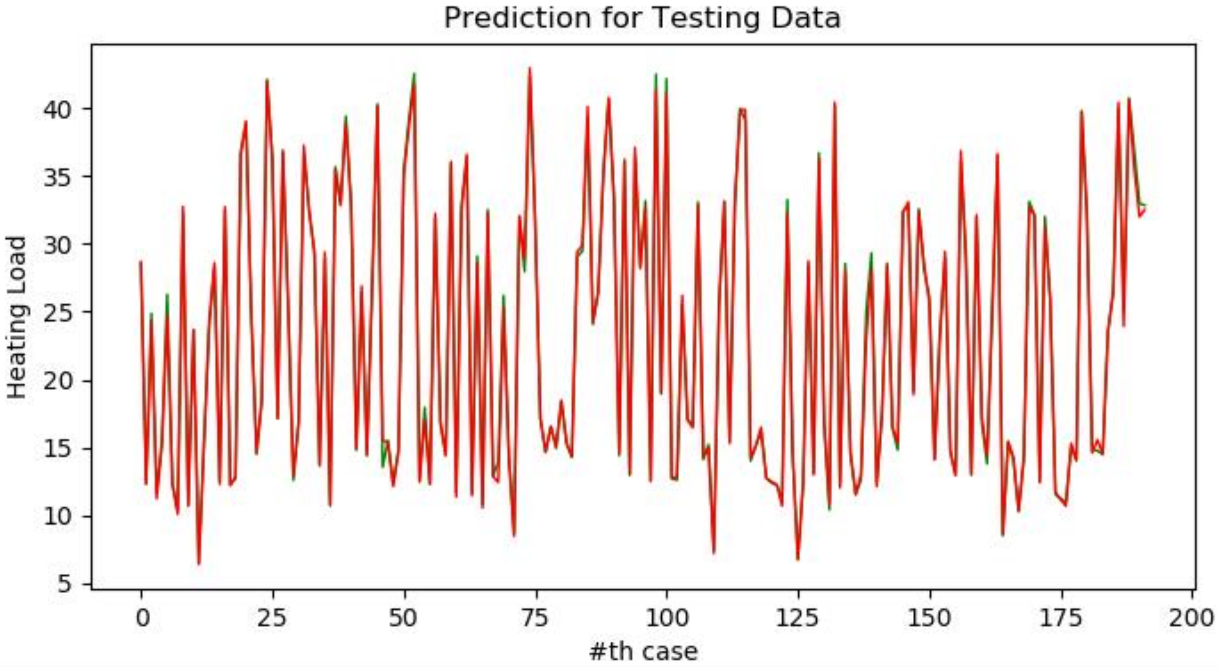


- **Shuffle = FALSE**

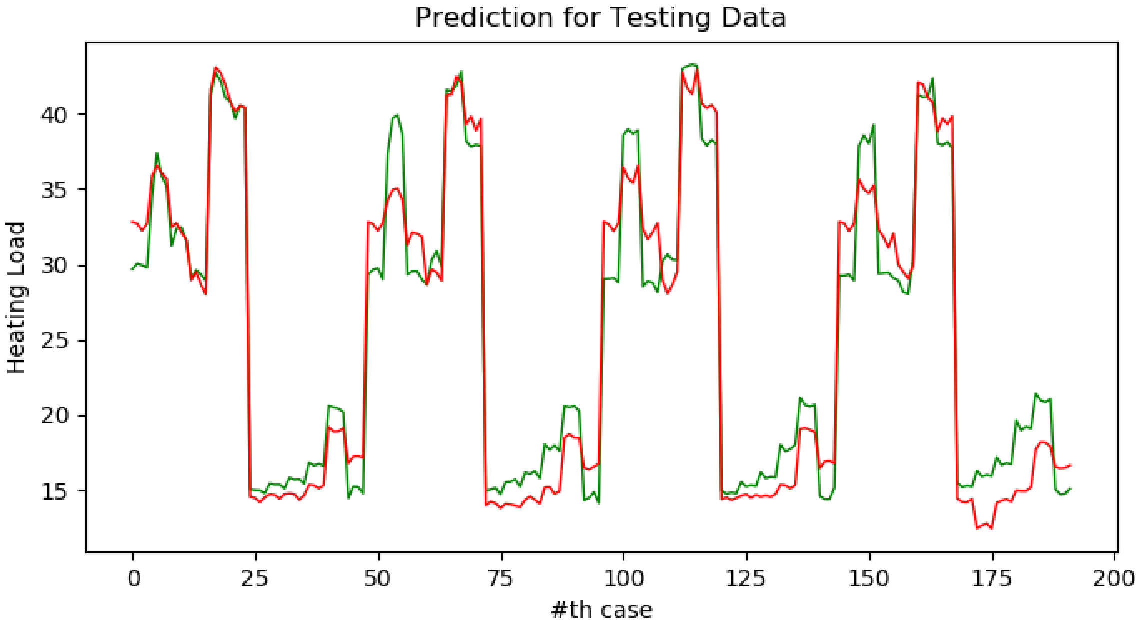


d. Regression result with testing label.

- **Shuffle = TRUE**



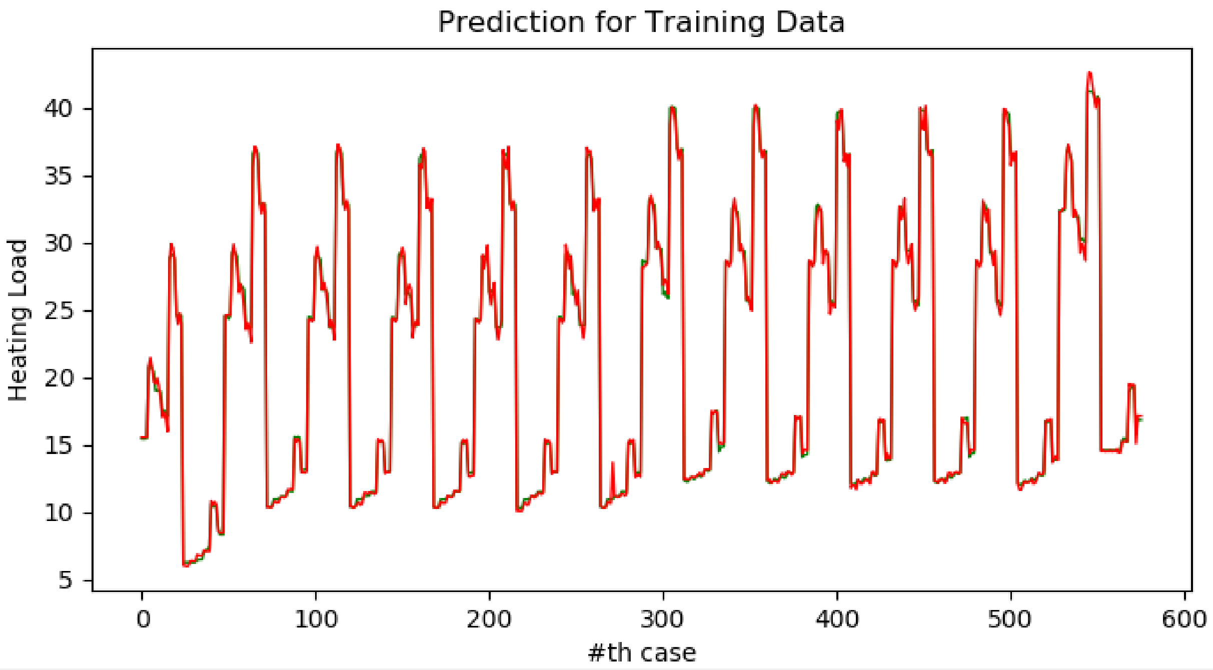
- **Shuffle = FALSE**

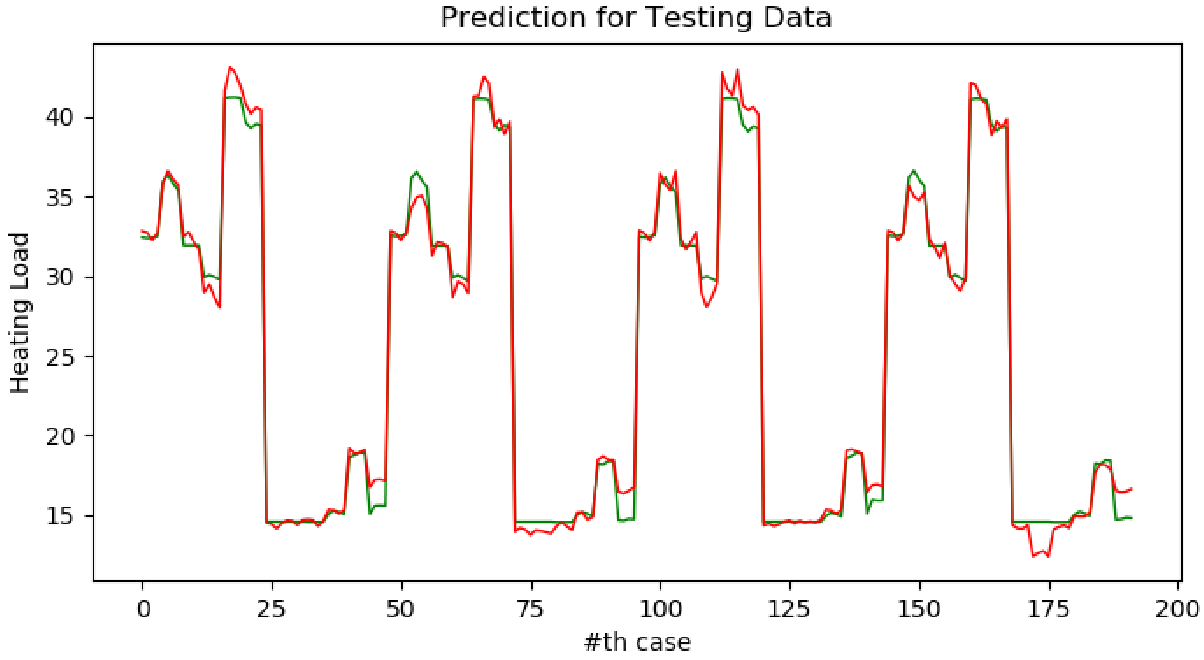


- A method to identify which feature that has the most influent to the result is that adding noise to the feature. Assuming that in practice, the collected data comes from the sensors might be noise, and which feature that we need to reduce the noise as small as possible. The error is generated by Gaussian ((μ, σ), zero mean and be normalize from 0 to 1, saying as how many percent that data was noised.

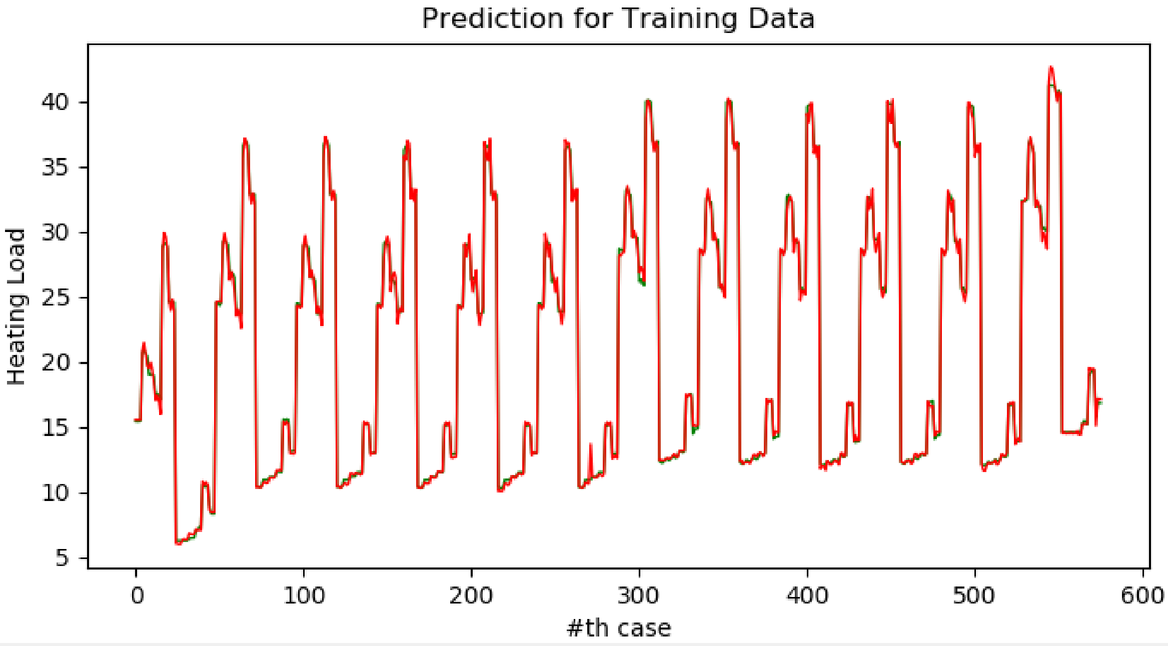
**ERMS/each data*

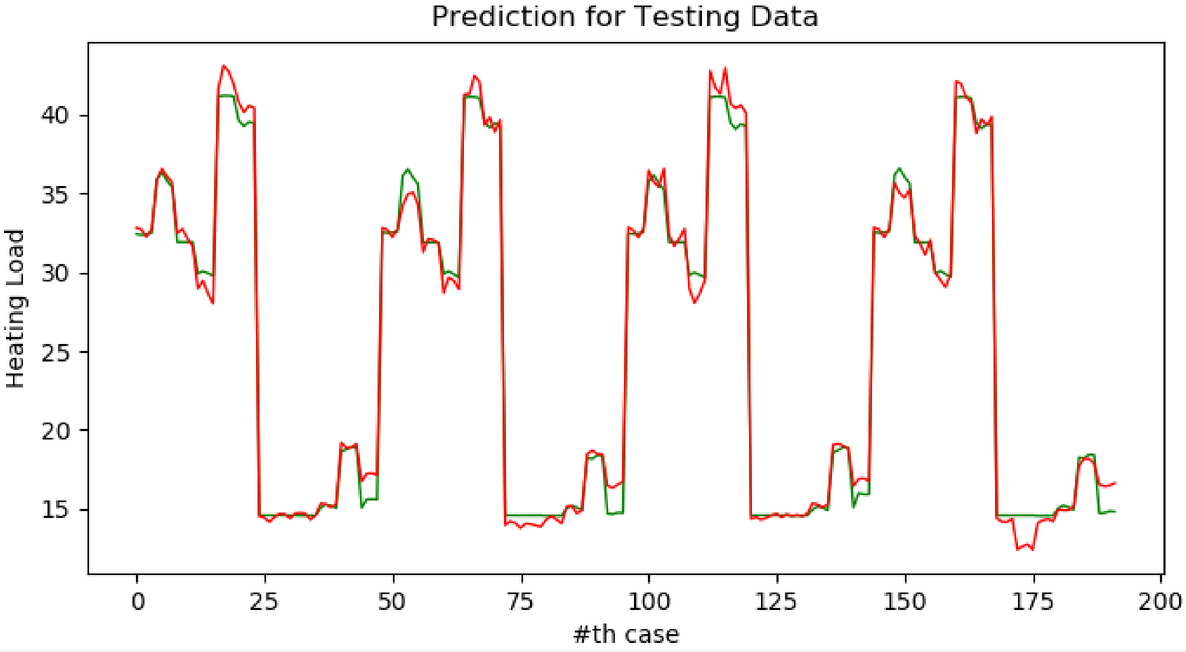
- By this test, we could say that the 3rd features influent to the result more than the others two feature. We had tried this method to find out which features is more important than the others feature.
- By the way, this is the state – of – art questions on Machine Learning, as well as how we configure the parameter and the structure of the network.
- Another method to select features is by the variance of the features. Removing the features have the variance that is smaller than the threshold. We could expect that the features that have small variance will not effect to result.
- Remove 1 feature (2nd feature) with minimum variance. The result is quite similar to when we use all of the features, shown below:



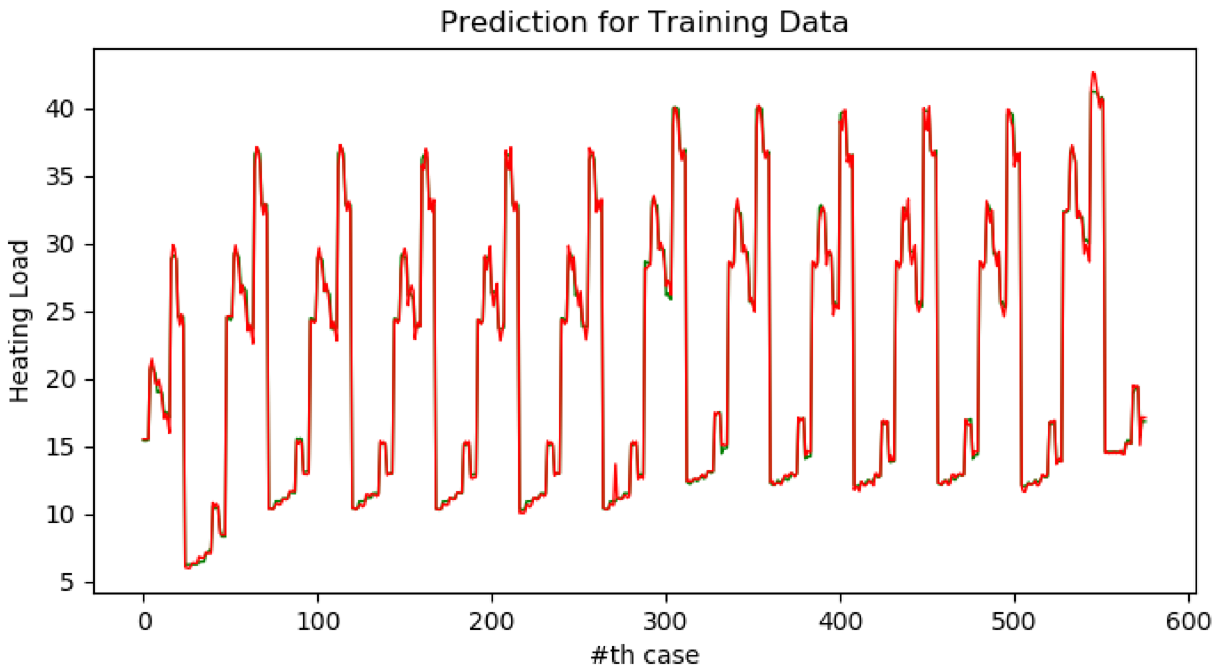


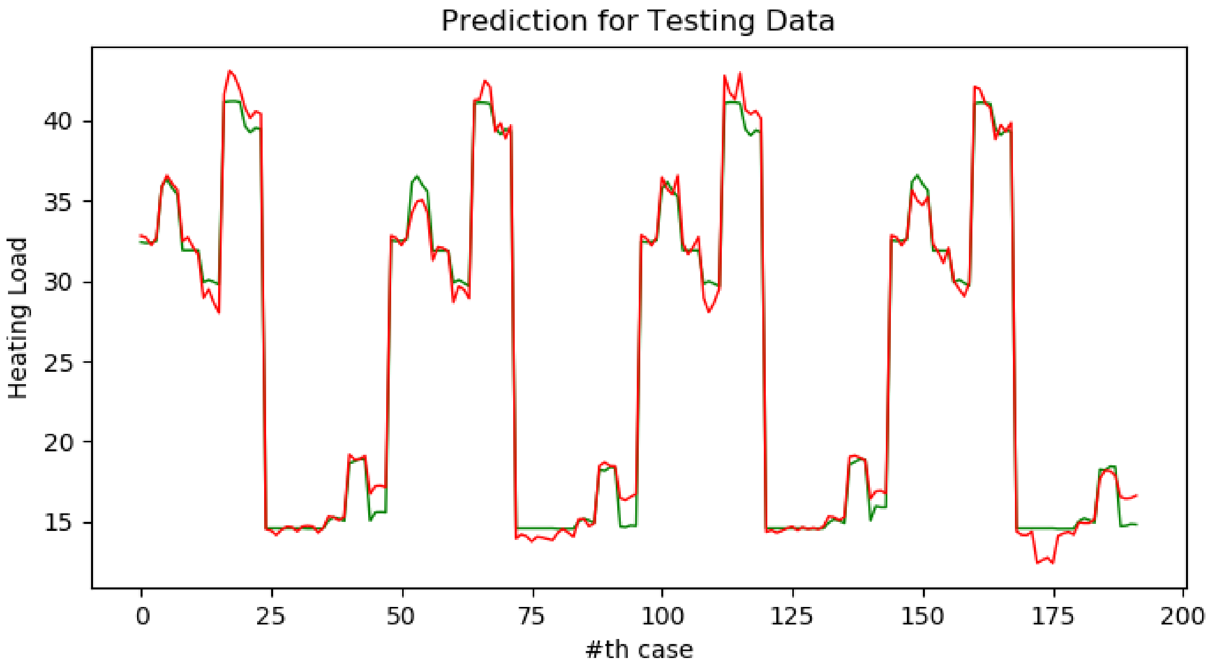
- Remove 2 features (0th and 2nd features) with minimum variance. The result shown below:





- Remove 1 features (5th features) with maximum variance. Shown below:





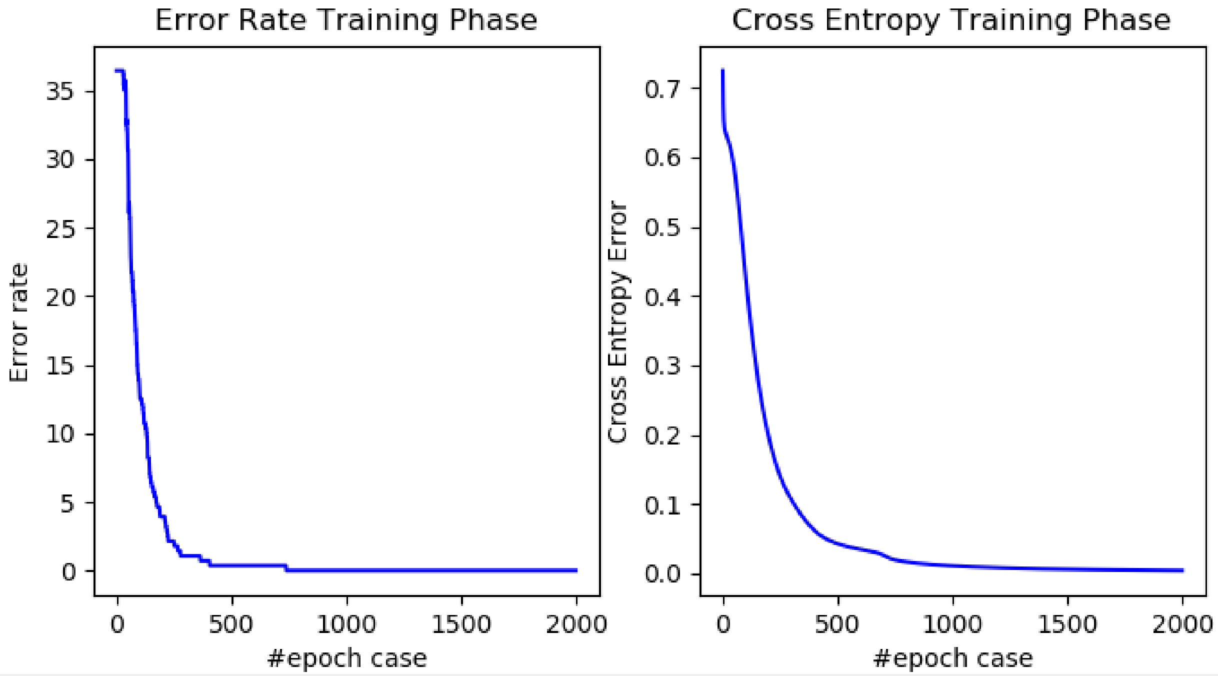
2. Classification

a. Network Architecture

Network Architecture	128-64-32-3-2
Selected features	all
Error Rate Training	**0.0%**

Error Rate Testing	**5.63%**
Epochs	2000
Learning rate	0.005
Batch Size	32
Training / Testing Size	80% - 20%

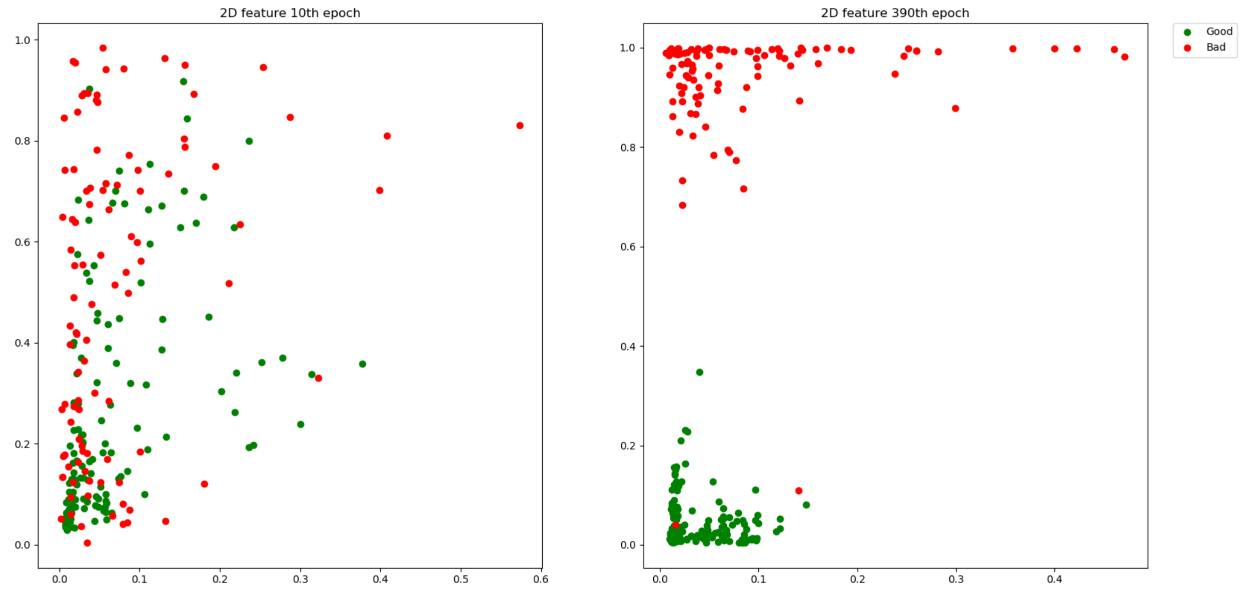
b. Learning Curve



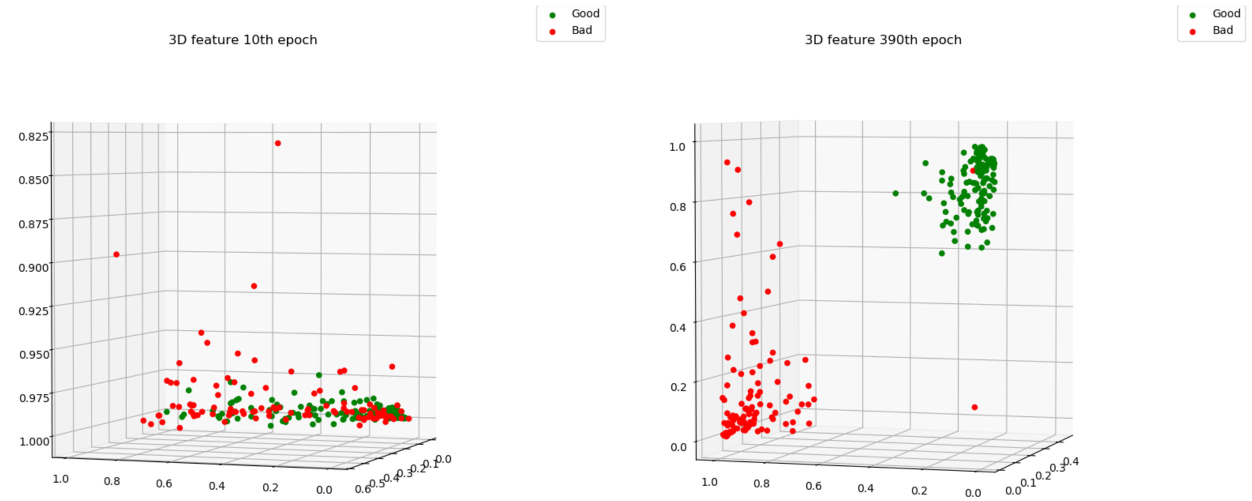
Error & Loss during Training Phase

c. Plotting latent features at different training state

By changing the hidden layer before output layer, using 2 or 3 nodes, we then compare the result by visualize (2D or 3D) the distribution of this hidden layer at different training state.



The result by visualize (2D)



The result by visualize (3D)

***Usage:**

To run regression program:

```
>> python Dregression.py
```

To run Classification program:

```
>> python Dclassification.py
```

If there is anything goes wrong thru my code, please don't hesitate to contact me to check it out. Thank you very much!

Email: haiphung106@gmail.com