

**The Experiment Report of**

***Machine Learning***

**College Software College**

**Subject Software Engineering**

**Members**  张超弦

**Student ID 201530613597**

**E-mail 864858406@qq.com**

**Tutor**   **谭明奎**

**Date submitted** **2017. 12 . 7**

**1. Topic:** Linear Regression, Linear Classification and  [Gradient descen](http://www.so.com/link?url=http://dict.youdao.com/search?q=Gradient descent&keyfrom=hao360&q=%E6%A2%AF%E5%BA%A6%E4%B8%8B%E9%99%8D%E7%9A%84%E8%8B%B1%E6%96%87&ts=1512646095&t=8f89ee073208b902da579e240918efa" \t "https://www.so.com/_blank)t

**2. Time:** 2017.12.2

**3. Reporter:** 张超弦

**4. Purposes:**

1) Further understand of linear regression and gradient descent.

2)Conduct some experiments under small scale dataset.

3)Realize the process of optimization and adjusting parameters.

**5. Data sets and data analysis:**

1）Linear Regression uses Housing in LIBSVM Data, including 506 samples and each sample has 13 features.

2) Linear classification uses australian in LIBSVM Data, including 690 samples and each sample has 14 features.

**6. Experimental steps:**

**Linear Regression and Gradient Descent:**

1) Load the experiment data. You can use load\_svmlight\_file function in sklearn library.

2) Devide dataset. You should divide dataset into training set and validation set using train\_test\_split function. Test set is not required in this experiment.

3) Initialize linear model parameters. You can choose to set all parameter into zero, initialize it randomly or with normal distribution.

4) Choose loss function and derivation: Find more detail in PPT.

5) Calculate gradient ***G*** toward loss function from all samples.

6) Denote the opposite direction of gradient ***G*** as ***D***.

7) Update model: .  is learning rate, a hyper-parameter that we can adjust.

8) Get the loss  under the training set and by validating under validation set.

9) Repeate step 5 to 8 for several times, and **drawing graph of**  **as well as**  **with the number of iterations**.

**Linear Classification and Gradient Descent:**

1) Load the experiment data.

2) Divide dataset into training set and validation set.

3) Initialize SVM model parameters. You can choose to set all parameter into zero, initialize it randomly or with normal distribution.

4) Choose loss function and derivation: Find more detail in PPT.

5) Calculate gradient ***G*** toward loss function from all samples.

6) Denote the opposite direction of gradient ***G*** as ***D***.

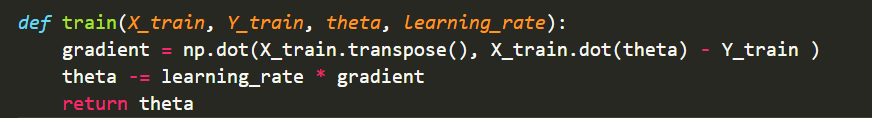
7) Update model: . is learning rate, a hyper-parameter that we can adjust.

8) **Select the appropriate threshold, mark the sample whose predict scores greater than the threshold as positive, on the contrary as negative**. Get the loss  under the training set and  by validating under validation set.

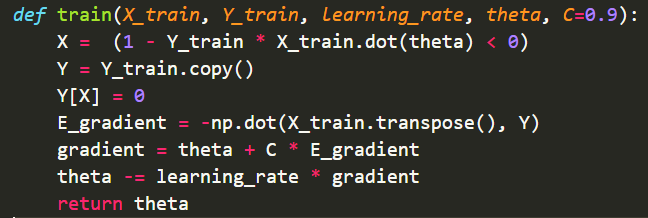
9) Repeate step 5 to 8 for several times, and **drawing graph of**  **as well as**  **with the number of iterations**.

1. **Code:**

For linear regression:



For linear classification:



**8. Selection of validation (hold-out, cross-validation, k-folds cross-validation, etc.):** hold-out

**9. The initialization method of model parameters:** set all parameter into zero

**10. The selected loss function and its derivatives:**

For linear regression:

Loss function:



Its derivatives:



For linear classification:

Loss function:



Its derivatives:



**11. Experimental results and curve:**

## Hyper-parameter selection (η, epoch, etc.):

For linear regression:

epoch = 1000

For linear classification:

epoch = 2000

η = C = 0.9

## Assessment Results (based on selected validation):

For linear regression:

Loss\_train = 

For linear classification:

Loss\_train = 

## Predicted Results (Best Results):

For linear regression:

Loss\_validation = 

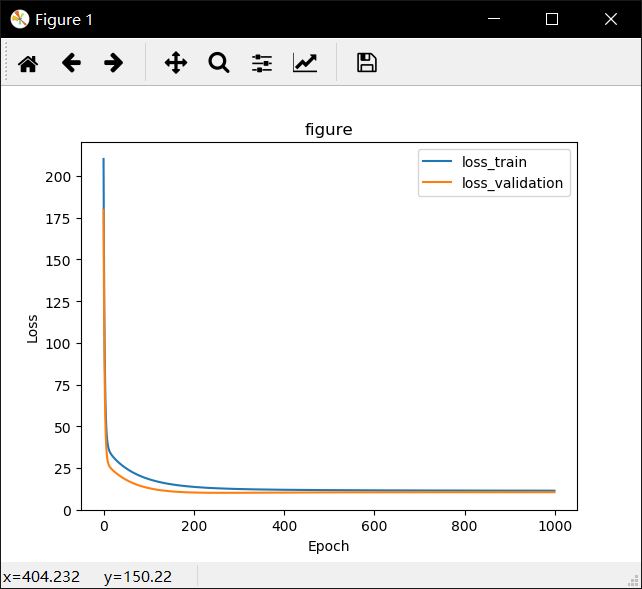
For linear classification:

Loss\_validation = 

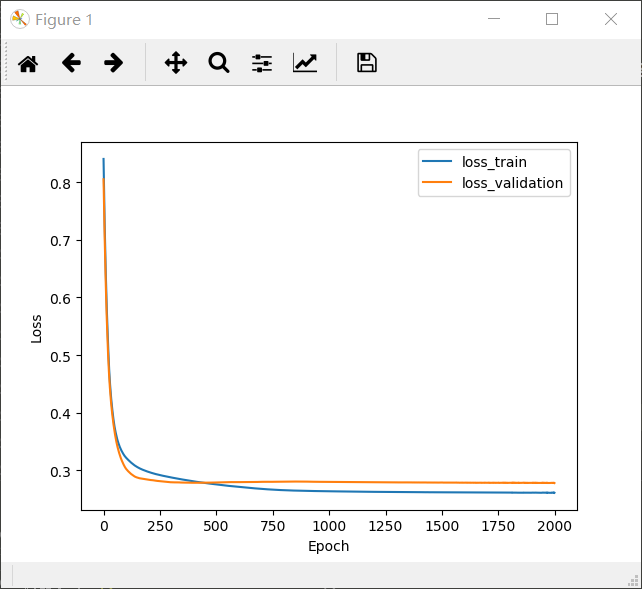
Accuracy = 

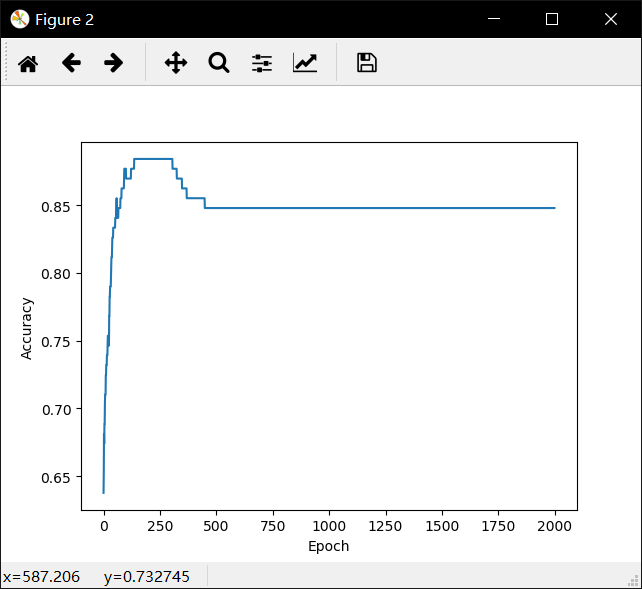
## Loss curve:

For linear regression:



For linear classification:





1. **Results analysis:**

For linear regression, the value of loss decreases rapidly and maintain about 11.

For linear classification,the value of loss reduces rapidly and maintain about 0.26. The value of accuracy increases rapidly first and than decreases a little. And the accuracy is always bigger than 0.8 but no more than 0.9 at last.

**13. Similarities and differences between linear regression and linear classification:**

In general, the two problems are essentially the same, that is, the fitting of the model. But the y value of classification problem, also known as label, is more discrete. Moreover, the same y value may correspond to a large number of X. These X have a certain range.

So the classification problem is more (x in some areas) corresponding to (a Y). And the regression problem model is more inclined to (x in the small area, or usually a x) corresponds to (a Y)

1. **Summary:**

In this experiment, I understand the essence of linear regression and linear classification more. Not only that, I initially learned and used Python. Machine learning is a magic course, I find it that the computer uses algorithms to learn from a large number of data and be smarter. How amazing this is !I met a lot of problems in this experiment, and I got solutions to these problems by looking up the Internet and asking my classmates. By the way, adjusting parameters is an interesting process !