

**The Experiment Report of**

***Machine Learning***

**College Software College**

**Subject Software Engineering**

**Members**   **Jin Cao**

**Student ID 201530611111**

**E-mail 2512046562@qq.com**

**Tutor**   **MingKui Tan**

**Date submitted** **2017.12.2**

**1. Topic:** Linear Regression, Linear Classification and Gradient Descent

**2. Time: 2017/12/2**

**3. Reporter: Jin Cao**

**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:**

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

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

**6. Experimental steps:**

(1) 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: . η 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.

(2) 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 as .

7)Update model: . ηis 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 trainin 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.

**7. Code:**

(1) Linear Regression and Gradient Descent

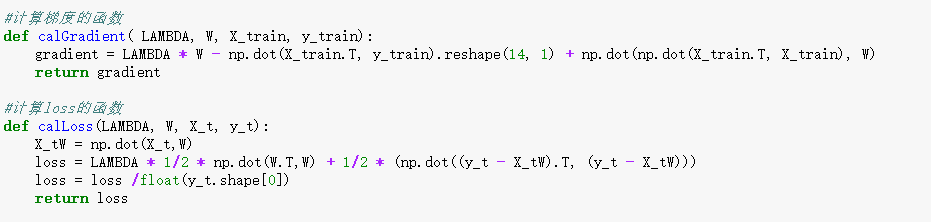


Fig1. The calculation of gradient and loss

(2) Linear Classification and Gradient Descent

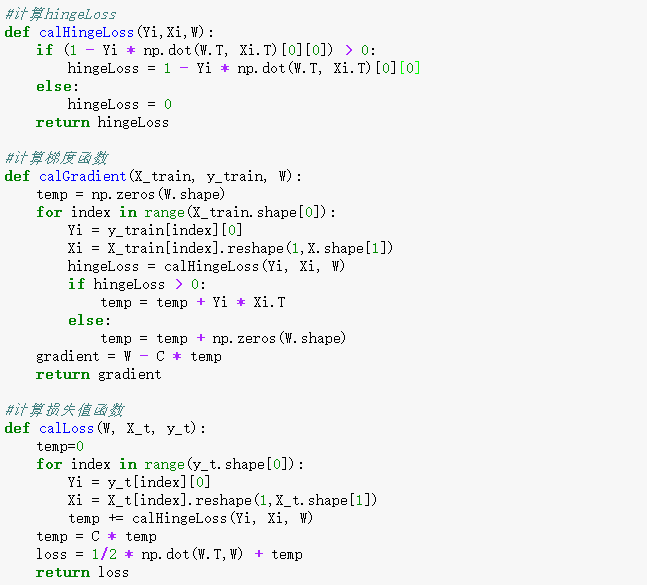


Fig2. The calculation of gradient and loss

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

Hold-out, Divide dataset into training set and validation set. I set the 3/10 of dataset to be the validation set, the left 7/10 of dataset to be the training set.

**9. The initialization method of model parameters:**

(1) Linear Regression and Gradient Descent

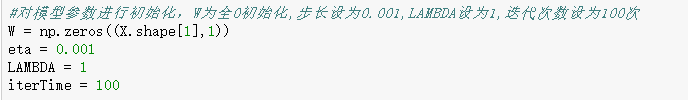


Fig3.the parameter of linear regression

(2) Linear Classification and Gradient Descent



Fig4. the parameter of linear classification

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

(1) Linear Regression and Gradient Descent

Loss Function:

Derivative:

(2) Linear Classification and Gradient Descent

Loss Function:

Derivative:

The derivative of loss is:

**11. Experimental results and curve:**

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

(1) Linear Regression and Gradient Descent

(2) Linear Classification and Gradient Descent

C = 1

threshold = 0

## Assessment Results (based on selected validation):

(1) Linear Regression and Gradient Descent

In the iteration process, the min average loss of training set is: 12.011219498587989

In the iteration process, the min average loss of validation set is: 12.672467869280991

(2) Linear Classification and Gradient Descent

In the iteration process, the min average loss of training set is: 0.29377026470702783

In the iteration process, the min average loss of validation set is: 0.30935252603752267

## Predicted Results (Best Results):

(1) Linear Regression and Gradient Descent

the min average loss of validation set is: 12.672467869280991

(2) Linear Classification and Gradient Descent

the min average loss of validation set is: 0.30935252603752267

## Loss curve:

(1) Linear Regression and Gradient Descent

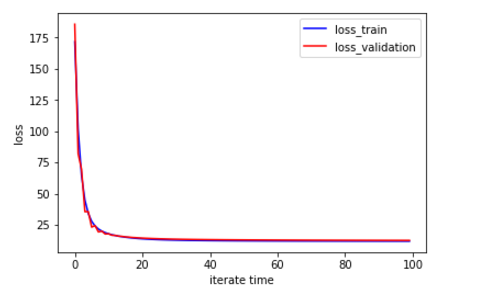


Fig5. The loss curve of linear regression

(2) Linear Classification and Gradient Descent

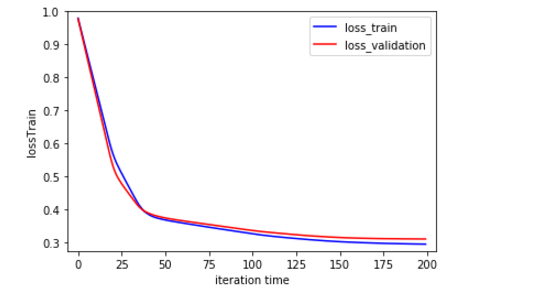


Fig6. The loss curve of linear classififcation

**12. Results analysis:**

Parameter optimization and selection is very important, different parameters will affect the calculation process, such as different learning rates, different convergence rate

(1) Linear Regression and Gradient Descent

The learning rate η were taken 0.1,0.01,0.001,0.0001, but 0.1 is too large, the loss value does not converge, the loss value of the image will be an upward trend, and the 0.01 is the same situation, take 0.001 loss value image began to show a downward trend, convergence, taking 0.0001 convergence rate decreases. The specific situation as shown below:

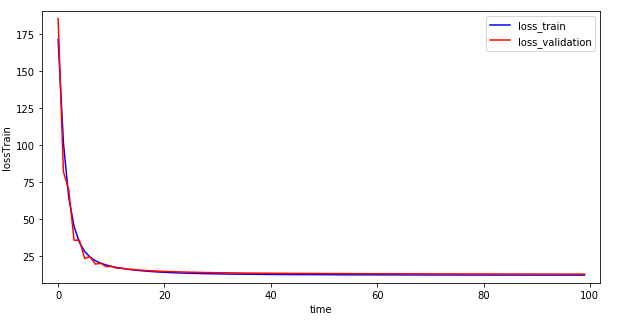


Fig7.η=0.001

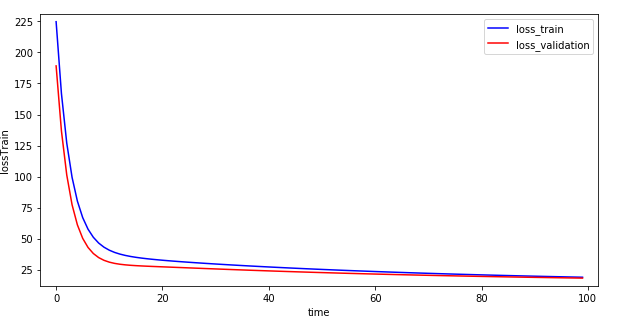


Fig8. η=0.0001

(2) Linear Classification and Gradient Descent

The learning rate η were taken 0.1,0.01,0.001,0.0001,0.00005, found that 0.1 value loss when the image oscillation amplitude is large, take 0.01 when the loss value of the image oscillation slightly reduced, take 0.001 when the oscillation and then decrease, take 0.0001 and found to be better, taking 0.00005 due to the small learning rate, so slow convergence, the specific situation as shown below:

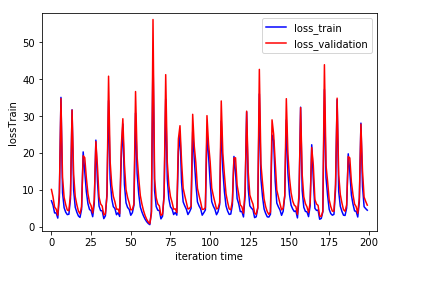


Fig9. η=0.1

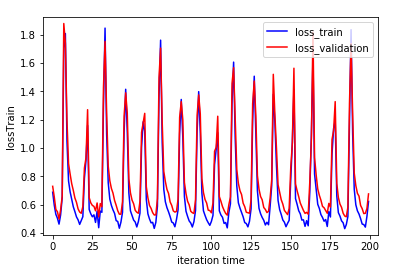


Fig10. η=0.01

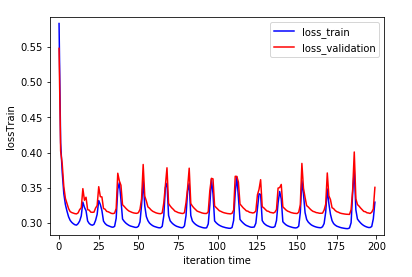


Fig11. η=0.001

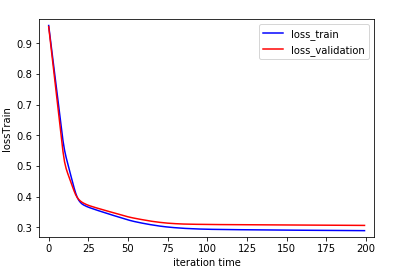


Fig12. η=0.0001

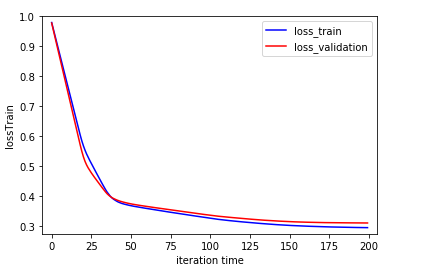


Fig12. η=0.00005

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

(1) The same point

Linear Regression and Linear Classification all use the linear function to be the prediction function. Linear Regression and Linear Classification all use gradient to optimize parameters.

(2)The different point

The loss function of Linear Regression and Linear Classification is different. After training, getting the linear function, the Linear Regression will get the prediction when we load the data into the linear function. But the Linear Classification need a threshold to distinguish the prediction into positive class and negative class.

**14. Summary:**

Both linear regression and linear classification use data sets to train a linear function, and optimize the parameters with a certain optimization method. In this experiment, we use gradient descent to optimize the parameters. For linear regression, find the linear function, the sample into the linear function, you can get the predicted value. For linear regression, set a certain threshold, and compare the predicted value with the threshold to judge whether the prediction result is positive or negative.

Through this experiment, I understand more deeply about linear regression and linear classification, and also have a deeper understanding of the gradient descent optimization method. In the experiment, different parameter choices, such as the learning rate, have a great influence on the experimental results. This experiment also made me have a deeper understanding of the reference parameters.