

**The Experiment Report of**

***Machine Learning***

**College Software College**

**Subject Software Engineering**

**Members**   **Feng Chen**

**Student ID 201530611159**

**E-mail 781909556@qq.com**

**Tutor**   **Mr. Tan**

**Date submitted** **2017.12 .8**

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

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

**3. Reporter:Feng Chen**

**4. Purposes:**

a. Further understand of linear regression and gradient descent.

b. Conduct some experiments under small scale dataset.

c. Realize the process of optimization and adjusting parameters.

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

a. Linear Regression uses [Housing](https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/regression.html#housing) in [LIBSVM Data](https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/), including 506 samples and each sample has 13 features. You are expected to download scaled edition. After downloading, you are supposed to divide it into training set, validation set.

b. Linear classification uses [australian](https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html#australian) in [LIBSVM Data](https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/), including 690 samples and each sample has 14 features. You are expected to download scaled edition. After downloading, you are supposed to divide it into training set, validation set.

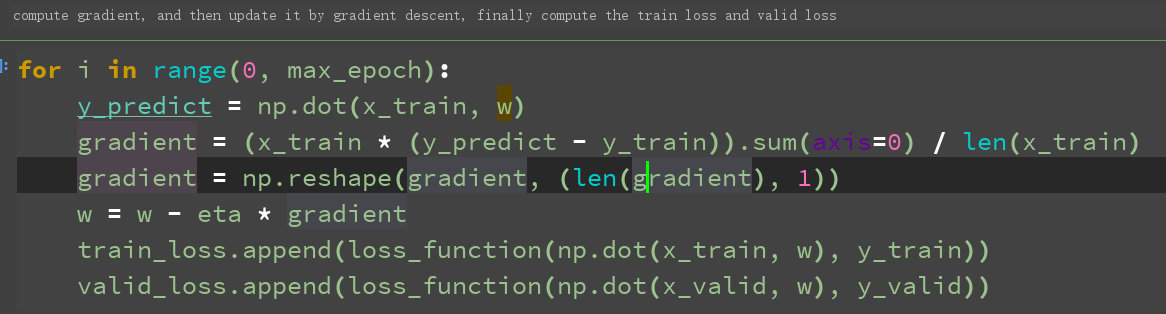
**6. Experimental steps:**

(1)**.** Linear Regression and Gradient Descent

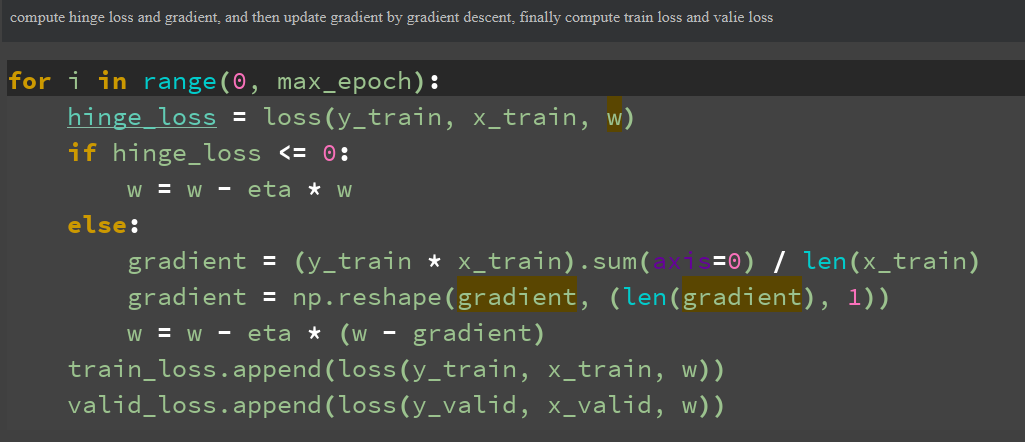
1. Load the experiment data. You can use [load\_svmlight\_file](http://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_svmlight_file.html) function in sklearn library.
2. Devide dataset. You should divide dataset into training set and validation set using [train\_test\_split](http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html) 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: Wt=Wt-1+ηD. ηis learning rate, a hyper-parameter that we can adjust.
8. Get the loss Ltrain under the training set and Lvalidation by validating under validation set.
9. Repeate step 5 to 8 for several times, and **drawing graph of** Ltrain **as well as** Lvalidation **with the number of iterations**.
   1. Linear Classification and Gradient Descent
10. Load the experiment data.
11. Divide dataset into training set and validation set.
12. Initialize SVM model parameters. You can choose to set all parameter into zero, initialize it randomly or with normal distribution.
13. Choose loss function and derivation: Find more detail in PPT.
14. Calculate gradient G toward loss function from all samples.
15. Denote the opposite direction of gradient G as D.
16. Update model: Wt=Wt-1+ηD.η is learning rate, a hyper-parameter that we can adjust.
17. **Select the appropriate threshold, mark the sample whose predict scores greater than the threshold as positive, on the contrary as negative.** Get the loss Ltrain under the training set and Lvalidation by validating under validation set.
18. Repeate step 5 to 8 for several times, and **drawing graph of** Ltrain **as well as** Lvalidation **with the number of iterations**.

**7. Code:**

(1).Linear Regression and Gradient Descent



(2). Linear Classification and Gradient Descent



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

(1). Linear Regression and Gradient Descent

adjust parameters:eta and epoch = 500

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| eta loss | 0.00001 | 0.0001 | 0.001 | 0.01 | 0.1 |
| train loss | 314.55865566 | 201.62550871 | 36.21810476 | 15.86955501 | 10.95595234 |
| valid loss | 371.05927964 | 287.26451149 | 39.83058521 | 20.82835596 | 11.90529436 |

(2) Linear Classification and Gradient Descent

adjust parameters:eta , C and epoch = 500

the format of each unit is (train loss, valid loss)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| C eta | 0.00001 | 0.0001 | 0.001 | 0.01 | 0.1 |
| 0.2 | (0.998075047564,1.06187842837) | (0.87677882839,0.870162747113) | (0.708699675525,0.684598170824) | (0.49549886568,0.402767389856) | (0.492545340191,0.399586048792) |
| 0.4 | (1.02131769356,0.974795921973) | (0.65771353769,0.635826807032) | (0.673554919288,0.706729053016) | (0.493547925029,0.401366960197) | (0.492545340191,0.399586048792) |
| 0.6 | (0.786501556525,0.829404930229) | (1.09281050152,1.12802107925) | (0.926785484827,0.88505934528) | (0.495660972662,0.403531504698) | (0.492545340191,0.399586048792) |
| 0.8 | (0.824419539813,0.803265178817) | (1.20099357639,1.24542382054) | (0.78895317803,0.740166984679) | (0.494751904258,0.402433407668) | (0.492545340191,0.399586048792) |
| 1.0 | (0.643288988054,0.633141650514) | (1.37271138469,1.38306236762) | (0.691832555136,0.697837833285) | (0.492678513605,0.400728154773) | (0.492545340191,0.399586048792) |

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

(1)**.** Linear Regression and Gradient Descent

initialize randomly

(2). Linear Classification and Gradient Descent

initialize randomly

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

(1)**.** Linear Regression and Gradient Descent

Loss function:

Derivatives:

(2). Linear Classification and Gradient Descent

Loss function:

Derivatives: w, if

w + C \* , if

**11. Experimental results and curve:**

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

1. Linear Regression and Gradient Descent

η is in [0.000001,0.00001,0.001,0.01,0.1] and epoch = 500

1. Linear Classification and Gradient Descent

η is in [0.000001,0.00001,0.001,0.01,0.1] , C is in [0.2,0.4,0.6,0.8,1.0]and epoch = 500

## Assessment Results (based on selected validation):

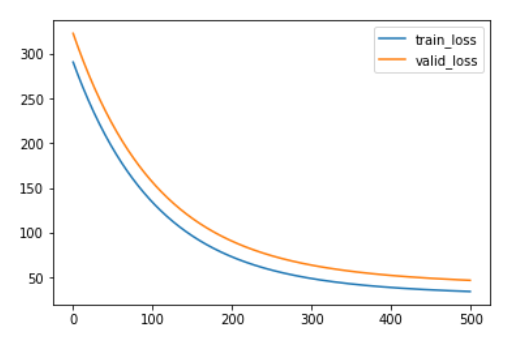
Shown in the table above

## Predicted Results (Best Results):

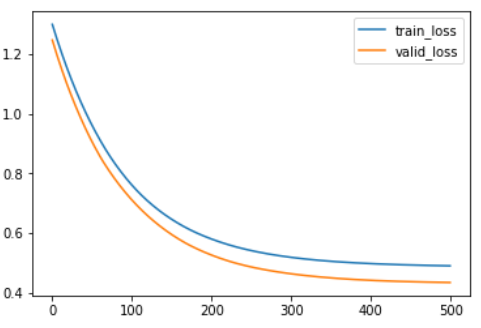
According to the table above, we can know that ‘eta=0.1’ is the best result to the Linear Regression and Gradient Descent and ‘eta=0.1, C=0.2’ is the best result to the Linear Classification and Gradient Descent

## Loss curve:

1. Linear Regression and Gradient Descent



1. Linear Classification and Gradient Descent



**12. Results analysis:**

1. Linear Regression and Gradient Descent

By increasing the learning rate, the loss of the training set and the verification set decreases gradually, but the parameters at this time may not be optimal

1. Linear Classification and Gradient Descent

By increasing the learning rate and C, the loss of the training set and the verification set decreases gradually, decreases rapidly at the beginning and then gradually slows down, eventually converges

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

Overall, the two questions are essentially the same, that is, the fitting (matching) of the model. However, the y value (also known as label) of the classification problem is more discretized, and the same y value may correspond to a large number of x, which is of a certain range.  
Therefore, the classification problem is more (some x in a certain region) corresponds to (a y), and the model of regression problem is more inclined to (x in a very small region, or generally x) to (a y).

**14. Summary:**

Through this experiment, I got a deeper understanding of linear classification and linear regression and gradient descent. I really understand how they are realized. I used to know some and only use libraries before.