

**The Experiment Report of**

***Deep Learning***

**College Software College**

**Subject Software Engineering**

**Members**  **(Yingying Chen)陈莹莹**

**Student ID 201710106604**

**E-mail 1569640579@qq.com**

**Tutor**   **Xiaowei Yang(杨晓伟)**

**Date submitted**  **2017. 12 . 8**

**1. Topic:**

Linear Regression, Linear Classification and Gradient Descent

**2. Time:**

2017-12-02 9:00-12:00 AM B7-138/238

**3. Reporter:**

Yingying Chen

**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](https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/regression.html#housing) in [LIBSVM Data](https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/).

, 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/).

**6. Experimental steps:**

Linear Regression and Gradient Descent

1. Load the experiment data. We use load\_svmlight\_file function in skearn library to load the data.
2. Divide dataset. We 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.
3. Initialize linear model parameters. We choose to set all parameter into zero, initialize it randomly or with normal distribution.
4. Choose loss function and derivation:
5. Calculate gradient G toward loss function from all samples.
6. Denote the opposite direction of gradient as D.
7. Update model w(t)=w(t-1)+a D. a 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. Repeat step 5 to 8 for several times, and **drawing graph of the training loss as well as the testing loss 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. We choose to set all parameter into zero, initialize it randomly or with normal distribution.
4. Choose loss function and derivation: loss=1.0/float(num\_records)\*e.sum()+1.0/2\*lambda\*w.transspose()\*dot.w
5. Calculate gradient G toward loss function from all samples.
6. Denote the opposite direction of gradient as D.
7. Update model w(t)=w(t-1)+a D ,a 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.**
9. Repeat step 5 to 8 for several times, and **drawing graph of the training loss as well as the testing loss with the number of iterations**.

**7. Code:**

(Fill in the contents of 8-12 respectively for linear regression and linear classification)

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

We hold out a test set of 25% of the whole dataset.

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

For linear regression and linear classification,

w = np.random.normal(size=(num\_features,1))

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

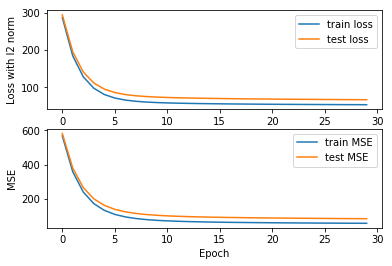
For linear regression ,

For linear classification,

**11. Experimental results and curve:**

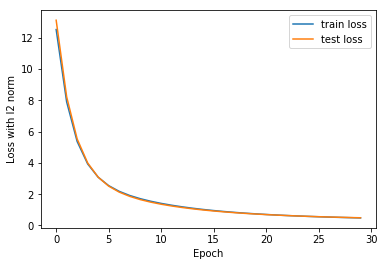
For linear regression,

threshold = 0.5



For linear classification,

threshold = 0.5

****

**12. Results analysis:**

From figure, we can see that the fitting of the MSE is better than the loss with 12 norm for the linear regression and the fitting of loss with 12 norm is good for the linear classification. Gradient decent is a valid method to optimize both regression problem and classification problem.

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

To sum up, the essence of the two problems is the same, they are all the fitting of the model. But for the linear classification, the label is more discrete and the same x corresponds to multiply y and the threshold is an important factor to influence the accuracy.

. However, for the linear regression, one x always corresponds to one y and the evaluation metric is mean square error.

**14. Summary:**

In machine learning, linear classification and regression are the two basic knowledge. Gradient decent is a valid method to optimize both regression problem and classification problem. For these two linear problems, we can write the following two steps：

1) How can we get a good loss function which assesses fitting degree and should be convex?

2) How can we choose a best model parameter?