

**The Experiment Report of**

***Deep Learning***

**College Software College**

**Subject Software Engineering**

**Members**  **Yiqun Wu**

**Student ID 201720145082**

**E-mail** [**1406754983@qq.com**](mailto:1406754983@qq.com)

**Tutor**   **Mingkui Tan**

**Date submitted** **2017.12.7**

## Topic: Linear Regression, Linear Classification and Gradient Descent

## 2. Time:：2017.12.7

## 3. Reporter: Yiqun Wu

## 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**: use the dataset “Housing” (scale version, scaled to [-1,1]) from LIBSVM Data. “Housing” contains 506 samples and each sample has 13 features.
2. **Linear Classification**: use the dataset “australian” (scaled to [-1,1]) from LIBSVM Data.. “australian” contains 690 samples and each samples has 14 features. Totally there are 2 classes.

## 6. Experimental steps:

**Linear Regression**:

1. Use load\_svmlight\_file function in sklearn library to load the experiment data.
2. Divide dataset into training set and validation by using train\_test\_split function.
3. Initialize linear model parameters by setting all parameter into zero.
4. Choose loss function and derivation.
5. Calculate gradient toward loss function from all samples.
6. Update model: . In which is learning rate and is the gradient.
7. Get the loss under the training set and by validating under validation set.
8. Repeate step 5 to 7 for several times, and drawing graph of as well as with the number of iterations.

**Linear Classification**:

1. Load the experiment data.
2. Divide dataset into training set and validation set.
3. Initialize SVM model parameters by setting all parameter into zero.
4. Choose loss function and derivation.
5. Calculate gradient toward loss function from all samples.
6. Update model:. is learning rate.
7. 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.
8. Repeat step 5 to 7 for several times, and drawing graph of as well as with the number of iterations.

## 7. 代码内容:

The code is shown in ClassificationExperiment.ipynb and RegressionExperiment.ipynb.

**Part 8-12 of Linear Regression**

## 8. Selection of validation：hold-out.

## 9. The initialization method of model parameters: setting all parameter into zero.

## 10. The selected loss function and its derivatives

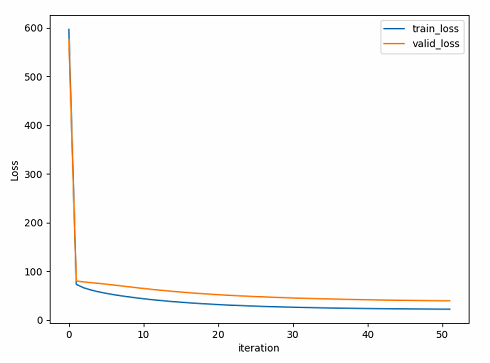
## 11. Experimental results and curve

## Hyper-parameter selection：η=0.1

## Assessment Results (based on selected validation)：the loss function value on both training and validation dataset declines over iterations and finally converge to an optimal value.

## Predicted Results (Best Results): the value of loss function on training dataset declines as the number of iterations, and the same as the loss on validation dataset.

## Loss curve:



## 12. Results analysis:

As shown in the above graph, the value of loss function on training dataset declines with the number of iterations, and the same as the loss on validation dataset. Finally the loss converge to an optimal value . This confirms that the gradient descent is effective and the linear regression fitting the data effectively.

**Part 8-12 of Linear Regression**

## 8. Selection of validation：hold-out.

## 9. The initialization method of model parameters: setting all parameter into zero.

## 10. The selected loss function and its derivatives

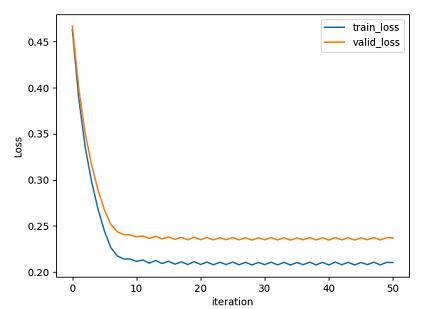
## 11. Experimental results and curve

## Hyper-parameter selection：η=0.1

## Assessment Results (based on selected validation)：the loss function value on validation data set declines over iterations and finally converge to an optimal value.

## Predicted Results (Best Results): the value of loss function on training dataset declines as the number of iterations, and the same as the loss on validation dataset.

## Loss curve:



## 12. Results analysis:

As shown in the above graph, the value of loss function on training dataset declines with the number of iterations, and the same as the loss on validation dataset. Finally the loss converge to an optimal value . This confirms that the gradient descent is effective. Also, the linear classification model divides most of the data points into true class.

## 13. Similarities and differences between linear regression and linear classification

**Similarities:** the linear classification and linear regression belong to linear model.

**Differences:** The linear classification uses hinge-loss in loss function, while the linear regression takes the mean square error. Linear classification uses the symbol(class) of result as output, but linear regression takes the direct result as output.

## 14. Summary：From this lab, I understand the principle of linear regression and linear classification better, and master the derivation and implementation of the gradient descent, which benefit me a lot.