華南理工大學

South China University of Technology

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 .15

1. Topic:

Logistic Regression, Linear Classification and Stochastic Gradient Descent

2. Time:

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

3. Reporter:

Yingying Chen

4. Purposes:

1. Compare and understand the difference between gradient descent and stochastic gradient descent.
2. Compare and understand the differences and relationships between Logistic regression and linear classification.
3. Further understand the principles of SVM and practice on larger data.

5. Data sets and data analysis:

Experiment uses a9a of LIBSVM Data, including 32561/16281(testing) samples and each sample has 123/123 (testing) features.

6. Experimental steps:

Logistic Regression and Stochastic Gradient Descent

1. Load the training set and validation set.
2. Initalize logistic regression model parameters, you can consider initalizing zeros, random numbers or normal distribution.
3. Select the loss function and calculate its derivation, find more detail in PPT.
4. Calculate gradient G toward loss function from **partial samples**.
5. Update model parameters using different optimized methods(**NAG, RMSProp, AdaDelta and Adam**).
6. Select the appropriate threshold, mark the sample whose predict scores **greater than the threshold as positive, on the contrary as negative**. Predict under validation set and get the different optimized method lossL_{NAG} , $L_{RMSProp}$, $L_{AdaDelta}$ and L_{Adam}.
7. Repeate step 4 to 6 for several times, and **drawing graph** of L_{NAG} , $L_{RMSProp}$, $L_{AdaDelta}$ and L_{Adam}**with the number of iterations**.

Linear Classification and Stochastic Gradient Descent

1. Load the training set and validation set.
2. Initalize SVM model parameters, you can consider initalizing zeros, random numbers or normal distribution.
3. Select the loss function and calculate its derivation, find more detail in PPT.
4. Calculate gradient G toward loss function from **partial samples**.
5. Update model parameters using different optimized methods(**NAG, RMSProp, AdaDelta and Adam**).
6. Select the appropriate threshold, mark the sample whose predict scores **greater than the threshold as positive, on the contrary as negative.** Predict under validation set and get the different optimized method loss L_{NAG} , $L_{RMSProp}$, $L_{AdaDelta}$ and L_{Adam}.
7. Repeate step 4 to 6 for several times, and drawing graph of L_{NAG} , $L_{RMSProp}$, $L_{AdaDelta}$ and L_{Adam} **with the number of iterations**.

7. Code:

(Fill in the contents of 8-11 respectively for logistic regression and linear classification)

8. The initialization method of model parameters:

Logistic Regression and Stochastic Gradient Descent

w = np.random.normal(size=(num_features,1))

Linear Classification and Stochastic Gradient Descent

w = np.random.normal(size=(num_features,1))

9. The selected loss function and its derivatives:

Logistic Regression and Stochastic Gradient Descent

hypothesis

$$h(x; w, b) = \frac{1}{1 + e^{-(w^T x + b)}}$$

loss function

$$\mathrm{l} = \frac{1}{2N} * \frac{1}{2}\sum_{i=1}^{N} -\log\big(h(x_i)\big) * (1+y_i) - \log\big(1-h(x_i)\big) * (1-y_i) + \frac{1}{2}\lambda(\|w\|^2 + \|b\|^2)$$

loss function's derivatives

$$\mathrm{l}' = \frac{1}{N}\sum_{i=1}^{N} x_i * (2h(x_i) - (y_i + 1)) + \lambda w$$

Linear Classification and Stochastic Gradient Descent

hypothesis

$$\mathrm{h(x; w, b)} = w^T x + b$$

$$\mathrm{hing\_loss(x, y; w, b, C)} = \max(0,1 - \mathrm{C} * \mathrm{y} * (\mathrm{wx} + \mathrm{b}))$$

loss function

$$l = \frac{1}{N}\sum_{i=1}^{N} \mathrm{hing\_loss}(x_i, y_i) + \frac{1}{2}\lambda(\|w\|^2 + \|b\|^2)$$

its derivatives

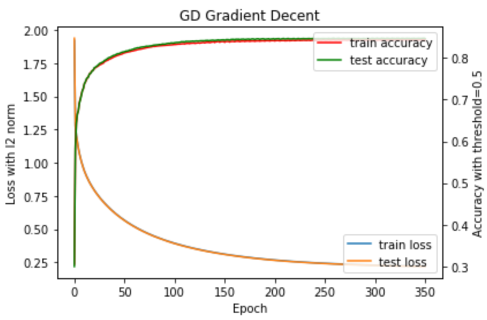
$$l' = -\frac{C}{N}\sum_{i=1}^{N} x_i\, y_i \mathrm{I}[\mathrm{hing\_loss}(x_i, y_i) > 0] + \lambda w$$

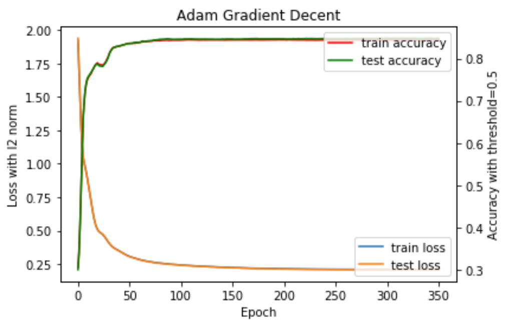
10. Experimental results and curve:(Fill in this content for various methods of gradient descent respectively)

Logistic Regression and Stochastic Gradient Descent

Fixed hyper-parameters are:max_iterate=50, batch_size = 8000
The hyper-parameters' candidate are as follows
param_grid = {

```
'Adam' : {'lamda': [0.01, 0.1],
          'eta': [0.01, 0.05],
          'Adam_beta1': [0.9, 0.95],
          'Adam_beta2' : [0.99, 0.999],
          'threshold': [0.5,0.6]},
'GD' : {'lamda': [0.01, 0.1, 0.5],
        'eta': [0.1, 0.2, 0.3, 0.4, 0.5],
        'threshold': [0.4,0.5,0.6]}}
```

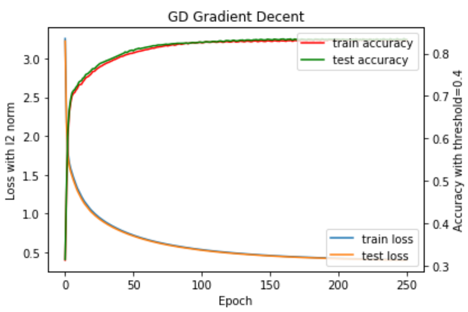


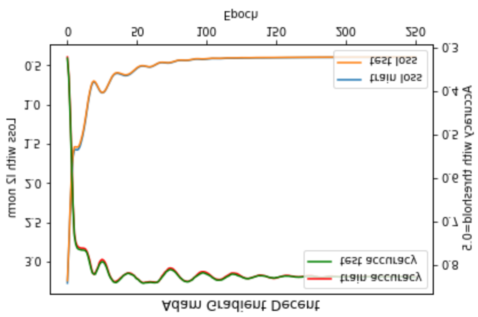


Linear Classification and Stochastic Gradient Descent

Fixed hyper-parameters are: max_iterate=50, batch_size = 8000
The hyper-parameters' candidate are as the above.

Metrics evolution process details:





11. Results analysis:

From figure, we can see that the fitting of the Adam is better than GD. Logistic Regression is better than Linear Classification . With gradient decent , **Adam** reach convergence within 50 epochs but GD needs more than 400epochs.

12. Similarities and differences between logistic regression and linear classification：

To sum up, the essence of the two problems is the same, they are all the fitting of the model. The loss of linear classification is hinge loss, but the logistic regression is negative log likelihood. 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.

13. Summary:

In machine learning, logistic regression and linear classification are the two basic knowledge. Stochastic Gradient Descent is a valid method to optimize both regression problem and classification problem. For classification problem, logistic regression is better than the linear classification.