华南理工大学

South China University of Technology

The Experiment Report of Machine Learning

SCHOOL: SCHOOL OF SOFTWARE ENGINEERING

SUBJECT: SOFTWARE ENGINEERING

Author:
Wanting Su

Supervisor:
Qingyao Wu

Student ID：
201530612767

Grade:
Undergraduate

December 9, 2017

Linear Regression, Linear Classification and Gradient Descent

Abstract—

Abstract——In the experiment we compare the difference between gradient descent and stochastic gradient descent. We also compare the differences and relationships between Logistic regression and linear classification. From the experiment we got a further understand the principles of SVM and practice on larger data.

I. INTRODUCTION

Gradient descent is a first-order iterative optimization algorithm for finding the minimum of a function. To find a local minimum of a function using gradient descent, one takes steps proportional to the negative of the gradient (or of the approximate gradient) of the function at the current point. If instead one takes steps proportional to the positive of the gradient, one approaches a local maximum of that function; the procedure is then known as gradient ascent.

Support vector machines are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier.

Linear Classification based on SVM and logic regression are the two models we use to apply gradient decent. There are many ways to update parameters. In this experiment, we compare four different ways of stochastic gradient decent, called NAG，RMSProp，AdaDelta and Adam.

In the experiment we implements the two models mentioned above and use different gradient decent technique to compare the performance of them.

II. METHODS AND THEORY

A. *For logistic regression:*

Loss function is:

$$L(\theta) = -\frac{1}{m}[\sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1-y^{(i)}) \log(1-h_\theta(x^{(i)}))]$$

In which $$h_\theta(X) = \frac{1}{1+e^{-\theta^T \cdot X}}$$

Gradient: $$\frac{\partial L(\theta)}{\partial \theta_j} = \frac{1}{m}\sum_{i=1}^{m}[(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)}]$$

$$\frac{\partial L(\theta)}{\partial \theta} = \frac{1}{m}\sum_{i=1}^{m} X_i(h_\theta(X) - y_i)$$

B. *For linear classification:*

Loss function is:

$$\min_{w,b} L(w,b) = \frac{||w||^2}{2} + \frac{C}{n}\sum_{i=1}^{n} \max(0, 1 - y_i(w^T x_i + b))$$

Gradient: $$\nabla_w L(w,b) = w + \frac{C}{n}\sum_{i=1}^{n} g_w(w_i)$$

$$\nabla_b L(w,b) = \frac{C}{n}\sum_{i=1}^{b} g_b(x_i)$$

In which we have:

$$g_{\mathbf{w}}(\mathbf{x}_i) = \begin{cases} -y_i \mathbf{x}_i & 1 - y_i(\mathbf{w}^\top \mathbf{x}_i + b) >= 0 \\ 0 & 1 - y_i(\mathbf{w}^\top \mathbf{x}_i + b) < 0 \end{cases}$$

C . *Batch Gradient decent and Stochastic Gradient Decent*

$$\begin{aligned} \mathbf{g}_t &\leftarrow \nabla J(\boldsymbol{\theta}_{t-1} - \gamma \mathbf{v}_{t-1}) \\ \mathbf{v}_t &\leftarrow \gamma \mathbf{v}_{t-1} + \eta \mathbf{g}_t \\ \boldsymbol{\theta}_t &\leftarrow \boldsymbol{\theta}_{t-1} - \mathbf{v}_t \end{aligned}$$

Figure 1:NAG update rules

$$\begin{aligned} \mathbf{g}_t &\leftarrow \nabla J(\boldsymbol{\theta}_{t-1}) \\ G_t &\leftarrow \gamma G_t + (1-\gamma)\mathbf{g}_t \odot \mathbf{g}_t \\ \Delta\boldsymbol{\theta}_t &\leftarrow -\frac{\sqrt{\Delta_{t-1} + \epsilon}}{\sqrt{G_t + \epsilon}} \odot \mathbf{g}_t \\ \boldsymbol{\theta}_t &\leftarrow \boldsymbol{\theta}_{t-1} + \Delta\boldsymbol{\theta}_t \\ \Delta_t &\leftarrow \gamma\Delta_{t-1} + (1-\gamma)\Delta\boldsymbol{\theta}_t \odot \Delta\boldsymbol{\theta}_t \end{aligned}$$

Figure 2:AdaDelta update rules

$$\mathbf{g}_t \leftarrow \nabla J(\boldsymbol{\theta}_{t-1})$$
$$\mathbf{m}_t \leftarrow \beta_1 \mathbf{m}_{t-1} + (1-\beta_1)\mathbf{g}_t$$
$$G_t \leftarrow \gamma G_t + (1-\gamma)\mathbf{g}_t \odot \mathbf{g}_t$$
$$\alpha \leftarrow \eta \frac{\sqrt{1-\gamma^t}}{1-\beta^t}$$
$$\boldsymbol{\theta}_t \leftarrow \boldsymbol{\theta}_{t-1} - \alpha \frac{\mathbf{m}_t}{\sqrt{G_t + \epsilon}}$$

Figure 3:Adam update rules

$$\mathbf{g}_t \leftarrow \nabla J(\boldsymbol{\theta}_{t-1})$$
$$G_t \leftarrow \gamma G_t + (1-\gamma)\mathbf{g}_t \odot \mathbf{g}_t$$
$$\boldsymbol{\theta}_t \leftarrow \boldsymbol{\theta}_{t-1} - \frac{\eta}{\sqrt{G_t + \epsilon}} \odot \mathbf{g}_t$$

Figure 4:RMSProp update rules

III. EXPERIMENT

A. *Hyper Parameters Selection*

For logistic regression:

（1）NAG： η =0.005 =0.9 epoch = 300

（2）RMSProp: η =0.005 = 0.9 = 1e-8 epoch =300

（3）AdaDelta: = 0.9 = 1e-8 epoch=100

（4）Adam: η =0.005 = 0.9 = 0.999 = 1e-8 epoch=300

For linear regression:

(1)Learning rate:0.07 Iteration number:500
C:5 Batch size:100

(2)NAG: r=0.9

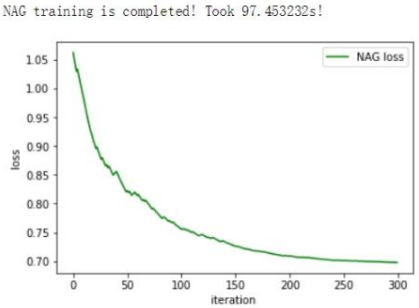
(3)RMSProp: r= 0.9 ε =1e-8

(4)AdaDelta: r=0.95 ε =1e-4

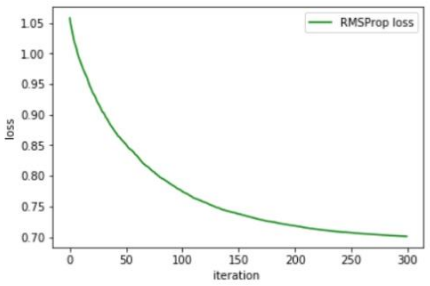
(5)Adam r=0.9 ε =1e-6 β =0.9

B. *Experience Result*

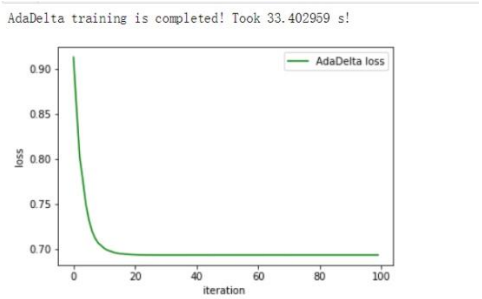
For Logistic Regression:



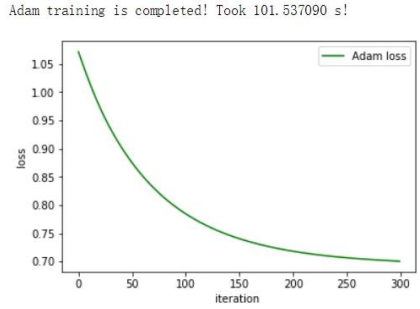
Logistic Regression Loss figure of NAG



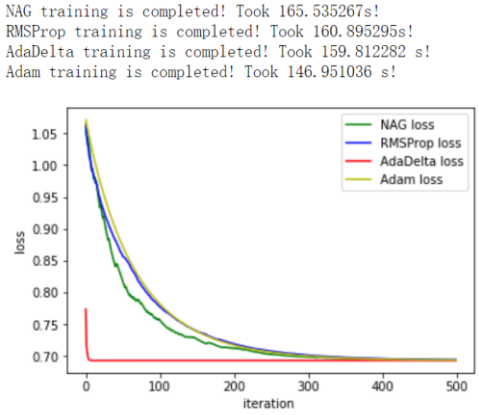
Logistic Regression Loss figure of RMSProp



Logistic Regression Loss figure of AdaDelta



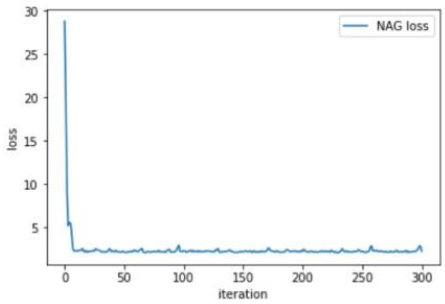
Logistic Regression Loss figure of Adam



- For Linear Regression

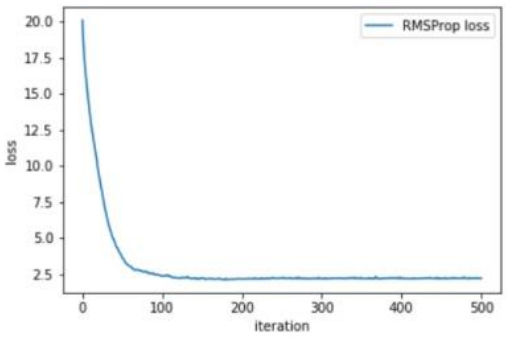
Begin to train
default acc: 0.355015

NAG completed! Took 18.831287 s!



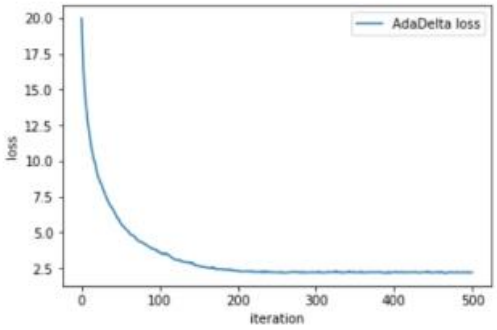
Linear Classification Loss figure of NAG

Begin to train
RMSProp completed! Took 33.205314 s!



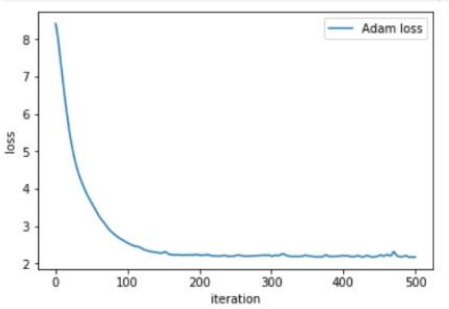
Linear Classification Loss figure of RMSProp

Begin to train
AdaDelta completed! Took 33.477286 s!



Linear Classification Loss figure of AdaDelta

Begin to train
Adam completed! Took 32.911980 s!



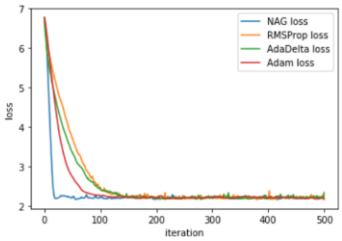
Linear Classification Loss figure of Adam

Begin to train
NAG completed! Took 35.342306 s!

RMSProp completed! Took 34.228036 s!

AdaDelta completed! Took 33.321083 s!

Adam completed! Took 33.640378 s!



IV. CONCLUSION

In this experiment we implement two model and four different algorithm, compare the difference between gradient descent and stochastic gradient descent. From the experiment we got a further understand the principles of SVM and practice on larger data.The detail of different algorithm does benefit the study in the future research .