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:
Qiaoyuan Wen

Supervisor:
Qingyao Wu

Student ID：
201530612989

Grade:
Undergraduate

December 9, 2017

Logistic Regression, Linear Classification and Stochastic Gradient Descent

Abstract—

I. Introduction

In statistics, logistic regression is a regression model where the dependent variable (DV) is categorical. This article covers the case of a binary dependent variable—that is, where the output can take only two values, "0" and "1", which represent outcomes such as pass/fail, win/lose, alive/dead or healthy/sick. Cases where the dependent variable has more than two outcome categories may be analysed in multinomial logistic regression, or, if the multiple categories are ordered, in ordinal logistic regression.In the terminology of economics, logistic regression is an example of a qualitative response/discrete choice model.

Stochastic gradient descent (often shortened to SGD), also known as incremental gradient descent, is a stochastic approximation of the gradient descent optimization and iterative method for minimizing an objective function that is written as a sum of differentiable functions. In other words, SGD tries to find minima or maxima by iteration.

II. Methods and Theory

NAG

Further proposals include the NAG method, Stochastic gradient descent with momentum remembers the update Δ w at each iteration, and determines the next update as a linear combination of the gradient and the previous update:

Delta w:=alpha Delta w-eta nabla Q_{i}(w-gamma v)

w:=w+Delta w

that leads to:

w:=w-eta nabla Q_{i}(w)+alpha Delta w

where the parameter w which minimizes Q(w) is to be estimated, and eta is a step size (sometimes called the learning rate in machine learning).

The name momentum stems from an analogy to momentum in physics: the weight vector w, thought of as a particle traveling through parameter space, incurs acceleration from the gradient of the loss ("force"). Unlike in classical stochastic gradient descent, it tends to keep traveling in the same direction, preventing oscillations.

RMSProp

RMSProp (for Root Mean Square Propagation) is also a method in which the learning rate is adapted for each of the parameters. The idea is to divide the learning rate for a weight by a running average of the magnitudes of recent gradients for that weight.[19] So, first the running average is calculated in terms of means square,

v(w,t):=gamma v(w,t-1)+(1-gamma)(nabla Q_{i}(w))^{2}

Where gamma is the forgetting factor.And the parameters are updated as,w:=w-{frac {eta }{sqrt {v(w,t)}}}nabla Q_{i}(w)

RMSProp has shown excellent adaptation of learning rate in different applications. RMSProp can be seen as a generalization of Rprop and is capable to work with mini-batches as well opposed to only full-batches.

AdaDelta

\begin{align*} \Vg_t & \gets \nabla J(\Vtheta_{t-1}) \\ G_t & \gets \gamma G_t + (1-\gamma) \Vg_t \odot \Vg_t \\ \Delta \Vtheta_t & \gets - \frac{\sqrt{\Delta_{t-1} + \epsilon}}{\sqrt{G_t + \epsilon}} \odot \Vg_t \\ \Vtheta_t & \gets \Vtheta_{t-1} + \Delta \Vtheta_t \\ \Delta_t & \gets \gamma \Delta_{t-1} + (1-\gamma) \Delta\Vtheta_t \odot \Delta \Vtheta_t \end{align*}

Adam

Adam(short for Adaptive Moment Estimation) is an update to the RMSProp optimizer. In this optimization algorithm, running averages of both the gradients and the second moments of the gradients are used. Given parameters w^{(t)} and a loss function L^{(t)}, where t indexes the current training iteration (indexed at 1), Adam's parameter update is given by:

m_{w}^{(t+1)}leftarrow beta _{1}m_{w}^{(t)}+(1-beta _{1})nabla _{w}L^{(t)}

v_{w}^{(t+1)}leftarrow beta _{2}v_{w}^{(t)}+(1-beta _{2})(nabla _{w}L^{(t)})^{2}

{hat {m}}_{w}={frac {m_{w}^{(t+1)}}{1-beta _{1}^{t}}}

{hat {v}}_{w}={frac {v_{w}^{(t+1)}}{1-beta _{2}^{t}}}

w^{(t+1)}leftarrow w^{(t)}-eta {frac {{hat {m}}_{w}}{sqrt {{hat {v}}_{w}}+epsilon }}

where epsilon is a small number used to prevent division by 0, and {beta _{1} andbeta _{2} are the forgetting factors for gradients and second moments of gradients, respectively.

III. Experiment

Motivation of Experiment

Compare and understand the difference between gradient descent and stochastic gradient descent.

Compare and understand the differences and relationships between Logistic regression and linear classification.

Further understand the principles of SVM and practice on larger data.

Dataset

Experiment uses a9a of LIBSVM Data, including 32561/16281(testing) samples and each sample has 123/123 (testing) features. Please download the training set and validation set.

Environment for Experiment

python3, at least including following python package: sklearn，numpy，jupyter，matplotlib

It is recommended to install anaconda3 directly, which has built-in python package above.

Experiment Step

The experimental code and drawing are completed on jupyter.

Logistic Regression and Stochastic Gradient Descent

Load the training set and validation set.

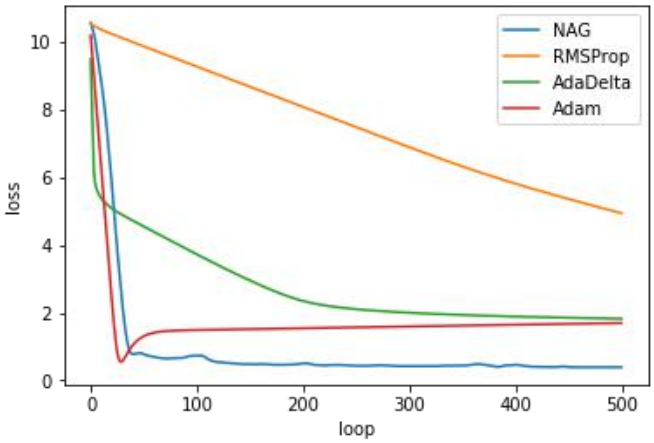
Initalize logistic regression model parameters, you can consider initalizing zeros, random numbers or normal distribution.

Select the loss function and calculate its derivation, find more detail in PPT.

Calculate gradient toward loss function from partial samples.

Update model parameters using different optimized methods(NAG，RMSProp，AdaDelta and Adam).

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



Linear Classification and Stochastic Gradient Descent

Load the training set and validation set.

Initalize SVM model parameters, you can consider initalizing zeros, random numbers or normal distribution.

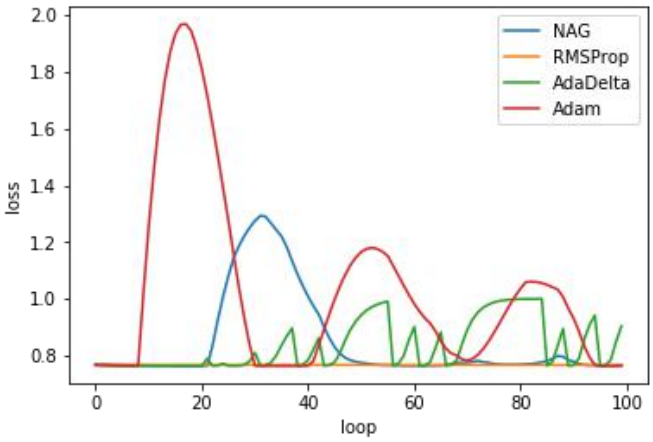
Select the loss function and calculate its derivation, find more detail in PPT.

Calculate gradient toward loss function from partial samples.

Update model parameters using different optimized methods(NAG，RMSProp，AdaDelta and Adam).

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

Result:



IV. CONCLUSION

Through this experiment, I have a deeper understanding of Logistic Regression, Linear Classification and Stochastic Gradient Descent.For different tasks, you should try different optimization algorithms.Different optimization algorithms have different learning speed, and the final convergence is not necessarily the same. In some cases SGD may also be the best choice.