South China University of Technology

The Experiment Report of *Machine Learning*

SCHOOL: SCHOOL OF SOFTWARE ENGINEERING

SUBJECT: SOFTWARE ENGINEERING

Author:
Haipeng Deng

Supervisor:
Mingkui Tan

Student ID:
201730686193

Grade:
Undergraduate

November 6, 2019

Lab 1 : Linear Regression and Stochastic Gradient Descent

***Abstract*—This report introduces my work in Lab 1 : Linear Regression and Stochastic Gradient Descent, where I constructed a linear regression model and implemented two basic methods of optimization i.e. closed-formed solution and stochastic gradient descent.**

I. Introduction

LINEAR regression, closed-formed solution and stochastic gradient descent are three fundamental concepts in machine learning. Although linear regression may seemed quite too simple compared to those complicated state-of-the-art models. It has been a readily comprehensible model for a machine learning starter. This lab is conducted in order to further understand the three concepts, conduct some experiments under small scale dataset, realize the process of optimization and adjusting parameters.

II. Methods and Theory

Simple linear regression describes the linear relationship between a variable, plotted on the x-axis, and a response variable y, plotted on the y-axis

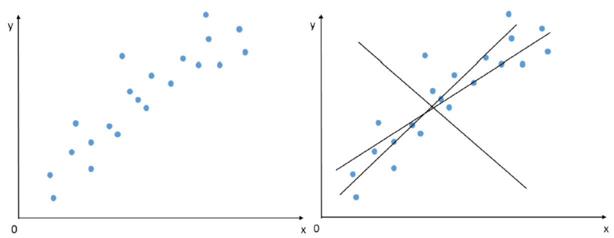


Fig. 1. Simple linear 1D regression

The model function of linear regression is shown below:

$$\begin{aligned} f(x;b,w) &= b + w_1x_1 + ... + w_mx_m \\ &= \sum_{j=1}^{m} w_jx_j + b \\ &= w^Tx + b \end{aligned}$$

In this experiment, the least squared loss function is chosen to judge the "quality" of the model:

$$\begin{aligned} L_D(w) &= \frac{1}{2}\sum_{i=1}^{n}(y_i - f(x_i;w))^2 \\ &= \frac{1}{2}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \end{aligned}$$

In closed-form solution, we directly derive the w* by analytical methods i.e. letting $\frac{\partial L(w)}{\partial w} = -X^Ty + X^TXw = 0$. And finally we can get $w* = (X^TX)^-1X^Ty$

In stochastic gradient descent method, after the loss function is chosen, a random sample will be selected to calculate the gradient of the model.

$$Gradient = \frac{\partial L(w)}{\partial w} = -X^Ty + X^TXw \tag{1}$$

We then optimize the linear model by adjusting w via following the opposite direction of the gradient. In this procedure, a hyperparameter η will be set manually to control the "learning rate" of the descent process. The process can be described as:

$$w' \to w - \eta\frac{\partial L(w)}{\partial w} \tag{2}$$

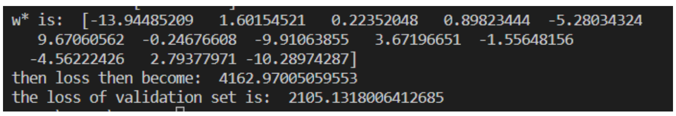
III. Experiments

A. Dataset

The dataset used in the experiment is "Housing" in LIBSVM Data, including 506 samples and each sample has 13 features. The dataset is split into a training set and a validation set as testing set is not required in this lab. The training set contains $\frac{2}{3}$ of the whole dataset and the remaining data is used to examine how good is the model(validation set).

B. Implementation

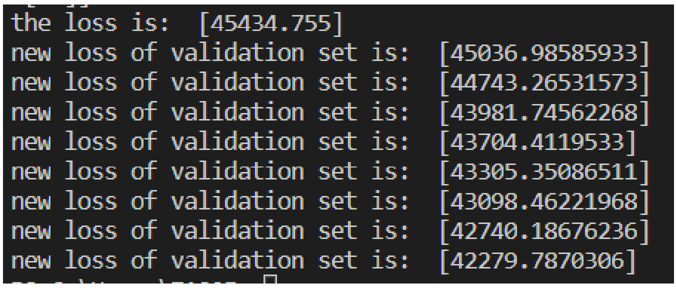
In closed-form solution, I firstly initialize the training set and validation set using python method, write the code of deriving w* and the code of computing loss of the model. Then I finished this part by directly invoking the functions i wrote, which is quite simple.



```
w* is:  [-13.94485209   1.60154521   0.22352048   0.89823444  -5.28034324
   9.67060562  -0.24676608  -9.91063855   3.67196651  -1.55648156
  -4.56222426   2.79377971 -10.28974287]
then loss then become:  4162.97005059553
the loss of validation set is:  2105.1318006412685
```

Fig. 2. result of closed-form solution

In stochastic gradient descent method, I firstly split the dataset using the same strategy above. Then I initialize my model parameters w by setting all parameters into 0. After checking the original loss of my model, I start to randomly choose a sample in my dataset and compute its gradient corresponding to the loss function. The following step is to update my w by applying $W_t = W_{t+1} + \eta D$ and compute the new loss. The two step above is conducted several times to observe how stochastic gradient descent helps optimize the model.



```
the loss is:  [45434.755]
new loss of validation set is:  [45036.98585933]
new loss of validation set is:  [44743.26531573]
new loss of validation set is:  [43981.74562268]
new loss of validation set is:  [43704.4119533]
new loss of validation set is:  [43305.35086511]
new loss of validation set is:  [43098.46221968]
new loss of validation set is:  [42740.18676236]
new loss of validation set is:  [42279.7870306]
```

Fig. 3. Code output of stochastic gradient descent

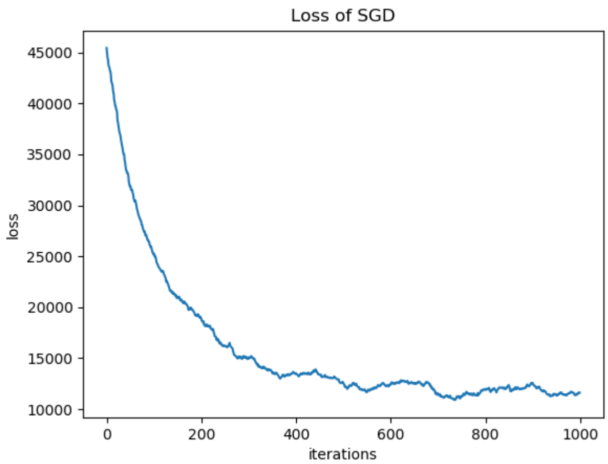


Fig. 4. stochastic gradient descent result

IV. Conclusion

After the lab, I feel I have understood the knowledge that I have learned in the course better. I encountered many situations when I was completing my lab code. But as I consulted the slices and asked search engines again and again. I had a completely new understanding of machine learning. Many thanks to the professor and those graduate students who offered help.