|  |
| --- |
| **Machine Learning Report of Pro1.1** |

**Haorui Dong**

**UBIT number: 50291149**

Department of Computer Science and Engineering

University at Buffalo

[*haoruido@buffalo.edu*](mailto:haoruido@buffalo.edu)

**Abstract**

In this project, we use machine learning to solve a problem called the Learning to Rank (LeToR) problem. We should first train a linear regression model using a closed-form solution. Then we need to train a linear regression model using stochastic gradient descent. We modified some hyper-parameters in order to find what values those hyper-parameters should take so as to give better performance.

1. **The original result**

First I run the sample code to get the original result. I change some print to see the number and plot of accuracy.

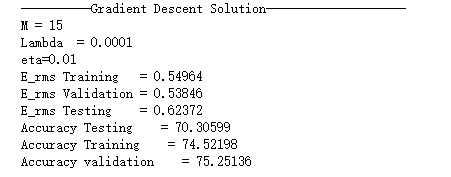


Figure1: SGD result

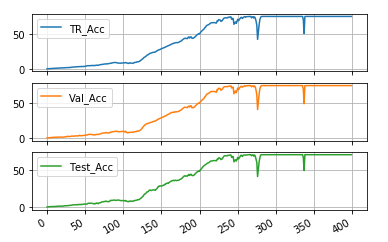
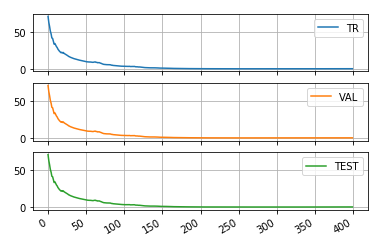


Figure2: Erms of SGD Figure3: Accuracy of SGD

This result is under the condition that: M = 10, C\_Lambda = 0.03.

The Testing Accuracy got 70.3. The E\_rms of testing got 0.062372.

1. **Change the value of C\_Lambda**

I change the value of C\_Lambda from 0.03 to 0.9 to see how it influence the result.

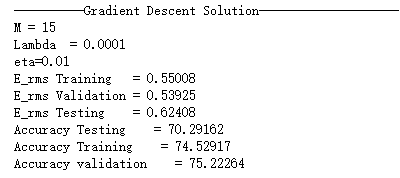


Figure4: Result of 0.9 C\_Lambda

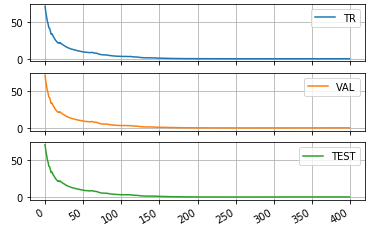
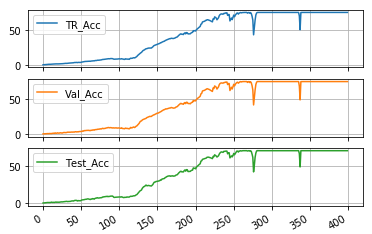
 

Figure5: Erms of 0.9 C\_Lambda Figure6: Accuracy of 0.9 C\_Lambda

Then I change C\_Lambda to 0.001.

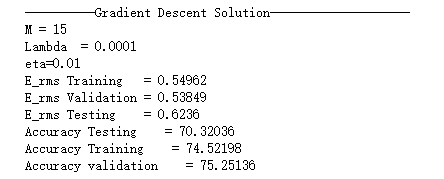


Figure7: Result of C\_Lambda = 0.001

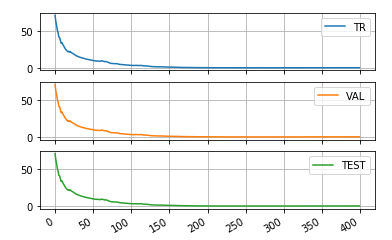
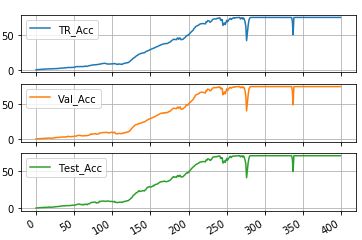
 

Figure8: Erms of C\_Lambda = 0.001 Figure9: Accuracy of C\_Lambda = 0.01

It doesn’t change the result a lot.

1. **Change the value of M**

M determines how many cluster will be separated. So I change the M from 10 to 2 and 50 to see the change of result.

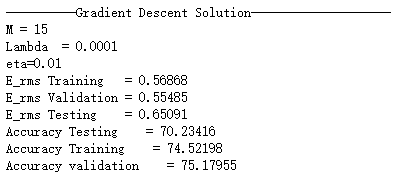


Figure10: Result of M = 2

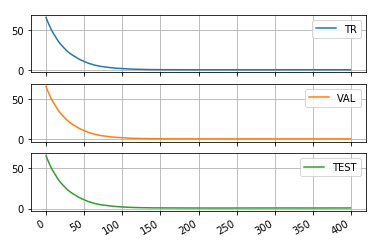
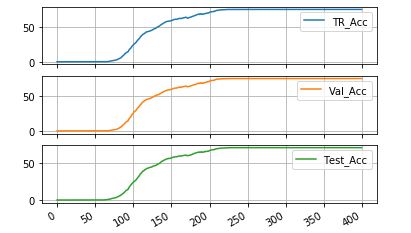
 

Figure11: Erms of M = 2 Figure12: Accuracy of M = 2

The curve got more smooth.

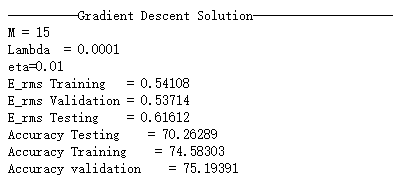


Figure13: Result of M = 50

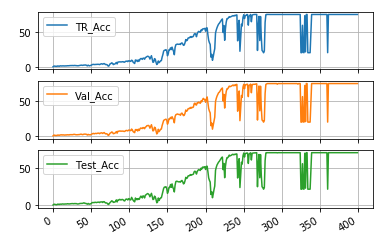
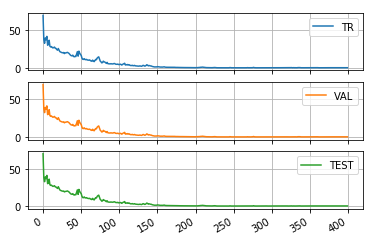


Figure14: Erms of M = 50 Figure15: Accuracy of M = 50

The final result seems same as the original one. But the shapes of curves are changed a lot.

1. **Change some other parameters**

In the part of Grandient Descent, I keep the learning rate and change the La to 10.

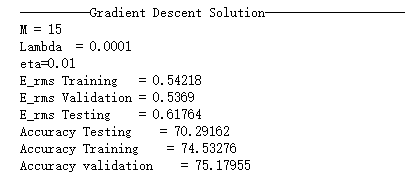


Figure16: Result of La = 10

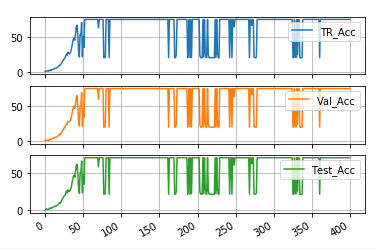
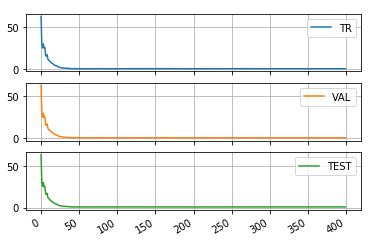


Figure17: Erms of La=10 Figure18: Accuracy of La=10

We found that the results do not change a lot, but it changed to final value in a higher speed.

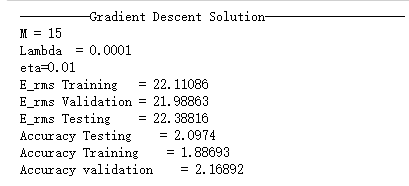
Then I change the La to a very small value such as 0.1.

Figure19: Result of La = 0.1

Obviously, it had a huge influence to result. And we got a very bad result, so I will not put the plot of curve.

I keep the La and change the Learning rate to 0.05 from 0.1. And the program cannot run with math range error.