Choosing Best Learning Rate for each Iteration

Indravardhan Reddy[#1], Vamsi Motupalli[*2], Harsha Koduri[#3]

***Abstract*— Learning Rate determines the step size of the weights that are updated during the training. The learning rate is the most important hyperparameter for effective training of neural networks. In this Paper we demonstrate how to choose the best learning rate for better optimization. We have followed three different approaches for choosing the best learning rate 1.General Search, 2.Binary Search. 3. Decaying methods**

***Keywords*— Neural networks, Optimization, Learning Rate, Decaying Learning Rate, Binary Search**

I. Introduction

Learning Rate is the step size of the weights that are updated during the training. The learning rate is the most important hyperparameter. It is a hyperparameter used in the training of neural networks which ranges from 0 to 1.0. This controls the adapting rate of a model for a certain problem. In this Paper we demonstrate how to choose the best learning rate for better optimization. We have followed three different approaches for choosing the best learning rate 1. Choosing the best learning rate in each iteration, 2.Binary Search Optimization and when the user gives a single learning rate then we can use decaying methods

II. Problem Statement

Learning rate plays an important role in determining the accuracy of our model. Convergence is not good when we choose very low and very high learning rates. So we came up with three different approaches for better optimization.

III. Dataset

We have used a California dataset for training the model. It contains the information from the 1990 California census. Columns contain Longitude, Latitude, Housing Median Age, Total Rooms, Total Bedrooms, Population, Households, Median Income, Median House Value. We used median Home Value as the target variable and remaining all as input variables Table 1 provides descriptions for each feature in the data set.

Column title	Description
Longitude	A measure of how far west a house is; a higher value is farther west
Latitude	A measure of how far north a house is; a higher value is farther north
Housing Median Age	Median age of a house within a block; a lower number is a newer building
Total Rooms	Total number of rooms within a block
Total Bedrooms	Total number of bedrooms within a block
Population	Total number of people residing within a block
Households	Total number of households, a group of people residing within a home unit, for a block
Median Income	Median income for households within a block of houses (measured in tens of thousands of US Dollars)
Median House Value	Median house value for households within a block (measured in US Dollars)

Table. 1 Data set columns with detailed description

IV. Effects of Learning Rate

A. Low Learning Rate

Low learning rates require more training as the smaller changes made to weights makes it learning slow and takes more time for training.Larger learning rates result in quick changes and require fewer training epochs.

B. High Learning Rate

Larger Learning Rate can cause the model to converge too quickly to a suboptimal solution and may diverge if the alpha value is too large. It is one of the most common problems in optimization.

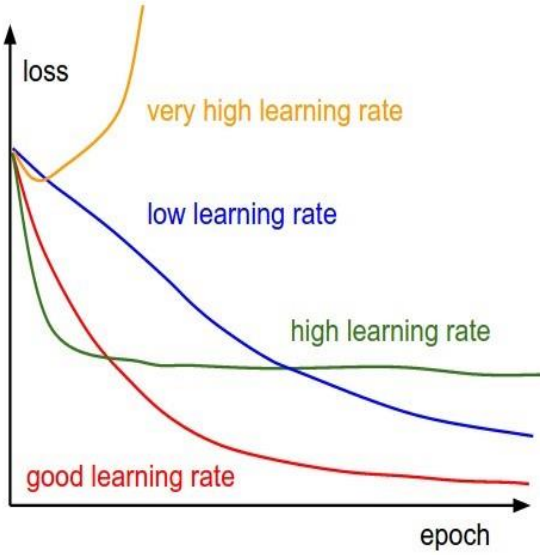

Fig. 1 Optimization curves for different learning rates

V. Choosing Best Learning Rate In Each Iteration

In this method the user inputs a list of learning rates and our optimization algorithms chooses the best learning rate out of the given list of learning and it repeats this process for each iteration. Algorithm for this method is shown in fig 2.

```
Initialize weights
Loop for each iteration:
  Initialize cost with inf
  Compute predicted outputs(Feedforward)
  Compute cost(MSE between true labels and predicted outputs)
  Computer gradients(Backpropagation)
  Loop for each learning rate in a list:
    Do gradient update
    Compute new predicted outputs form updated weights
    Compute new cost
    If new cost < cost :
      cost = new cost
      weights=updated weights
```

Fig. 2 Algorithm for choosing the best learning rate in each iteration

VI. Binary search Optimization

In Binary Search Optimization we use the Binary Search algorithm for searching the best learning rate. User inputs a list of learning rates, initially the algorithm sorts the learning rate list with an ascending order, then it takes the first and last values in the learning rate list and computes their corresponding costs. Finally it compares both the cost and if the learning rate chosen from the last index has lower cost then the algorithm updates the first index with the average of first and last indexes of the list, else it updates last index with the average of first and last indexes, and this process repeates for every iteration.

VII. Decaying Optimization

In Decaying Learning Rate the learning rate converges fast down the slope and reduces the converging which is a learning rate when it is nearing the minima with time.

Learning rate decay is dependent on the time as the time increases the value of learning rate decreases which helps in reaching the minima fast to the minima and there will be a less chance of diverging cost.

We perform two different methods of decaying 1.Exponential Decaying Learning Rate, 2.Inverse Time Decaying Learning Rate

A. Exponential Decaying Learning Rate

In this method learning rate decays exponentially with time (iterations). The remodel gradient descent update update step is

$$W_t = W_{t-1} - \gamma^t \Delta L_\theta$$

is new updated weight, is old weight, is a constantW^t parameter with range $W0^t$to^{-1}1, t is time(iteration),γ $\Delta L_\theta$is gradient of loss wrt weight parameters.

B. Inverse Time Decaying Learning Rate

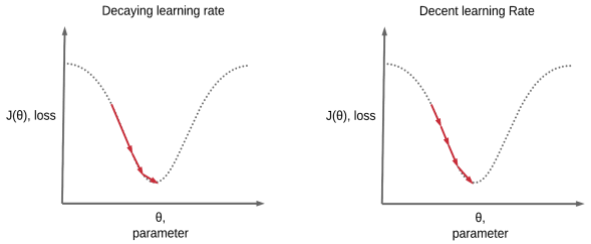

Fig. 3 Optimization curves for Decaying learning rate and Descent Learning Rate

VIII. RESULTS

1. *General Search*

No of epochs: 500

```
best alpha in 1 iteration : 0.01
best alpha in 2 iteration : 0.01
best alpha in 3 iteration : 0.01
best alpha in 4 iteration : 0.01
best alpha in 5 iteration : 0.01
best alpha in 6 iteration : 0.01
best alpha in 7 iteration : 1e-06
best alpha in 8 iteration : 1e-06
best alpha in 9 iteration : 1e-06
```

Fig. 4 choosing best alpha for each iteration

In this method learning rate decay is proportional to the inverse of time(iterations). The remodel gradient descent update update step is

$$W_t = W_{t-1} - 1 / (1 + t\gamma)^p \Delta L_\theta$$

is new updated weight, is old weight, is a constantW^t parameter with range $W0^t$to^{-1}1, t is time(iteration),γ p is an integer(2 is prefered), $\Delta L_\theta$is gradient of loss wrt weight parameters.

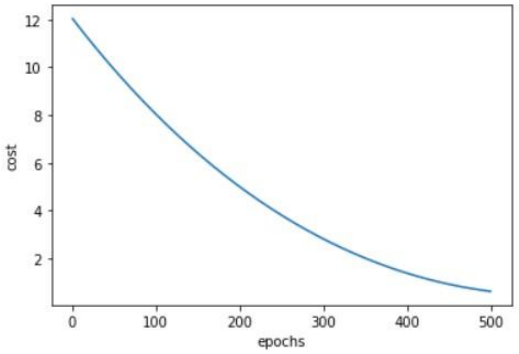

Fig 7 cost vs epoch plot

3. *Decaying Methods*

A. Exponential Decaying Learning Rate

No of epochs: 500

```
Exponential decaying alpha in 1 iteration : 0.001
Exponential decaying alpha in 2 iteration : 0.00098
Exponential decaying alpha in 3 iteration : 0.0009603999999999999
Exponential decaying alpha in 4 iteration : 0.000941192
Exponential decaying alpha in 5 iteration : 0.0009223681599999999
Exponential decaying alpha in 6 iteration : 0.0009039207967999999
Exponential decaying alpha in 7 iteration : 0.0008858423808639999
Exponential decaying alpha in 8 iteration : 0.00086812553324672
Exponential decaying alpha in 9 iteration : 0.0008507630225817855
```

Fig. 8 Exponential decaying alpha for each iteration

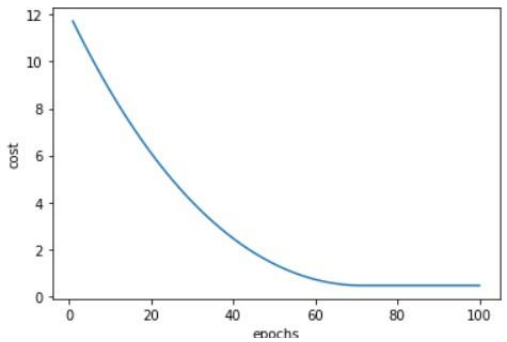

Fig 5 cost vs epoch plot

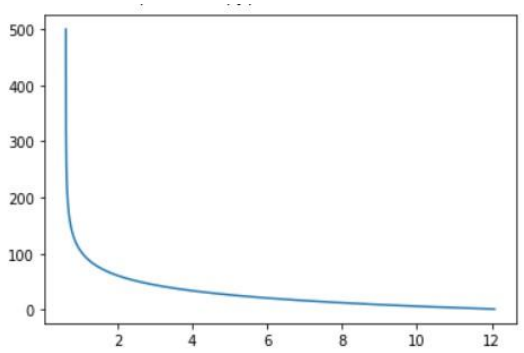

Fig. 9 epochs vs cost plot

2. Binary Search B. Inverse Time Decaying Learning Rate

No of epochs: 500

```
left most alpha :1e-06
right most alpha :0.0001
cost for left most alpha :12.077121254662488
cost for right most alpha :12.031955555251237
left most alpha :3e-05
right most alpha :0.0001
cost for left most alpha :12.018292263191361
cost for right most alpha :11.986446894785532
left most alpha :5e-05
right most alpha :0.0001
cost for left most alpha :11.963736650222172
cost for right most alpha :11.941051833557953
```

Fig. 10

Inverse Decaying alpha for each iteration

No of epochs: 500

```
inverse decaying alpha in 1 iteration : 0.01
inverse decaying alpha in 2 iteration : 0.006944444444444445
inverse decaying alpha in 3 iteration : 0.005102040816326531
inverse decaying alpha in 4 iteration : 0.003906249999999999
inverse decaying alpha in 5 iteration : 0.0030864197530864196
inverse decaying alpha in 6 iteration : 0.0025
inverse decaying alpha in 7 iteration : 0.0020661157024793385
inverse decaying alpha in 8 iteration : 0.0017361111111111106
```

Fig 6 comparing cost and choosing the next interval

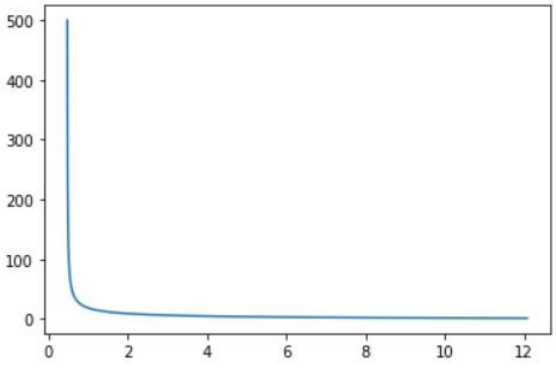

Fig. 11 epochs vs cost plot

IX. Conclusion

In this report we presented different techniques for choosing best learning rate for each iteration if the user gives us a list of learning rate then we could use general and binary search methods, if user gives a single learning rate then we could do better optimization using decaying methods

References

[1] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learningapplied to document recognition,"Proceedings of the IEEE, vol. 86,no. 11, pp. 2278–2324, Nov 1998.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classificationwith deep convolutional neural networks," inAdvances in NeuralInformation Processing Systems 25, F. Pereira, C. J. C. Burges,L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012,pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[3] T. Schaul, S. Zhang, and Y. LeCun, "No more pesky learningrates," inProceedings of the 30th International Conference onInternational Conference on Machine Learning - Volume 28, ser.ICML'13.JMLR.org, 2013, pp. III–343–III–351. [Online]. Available:http://dl.acm.org/citation.cfm?id=3042817.3042975

[4] S. Ruder, "An overview of gradient descent optimization algorithms,"arXiv e-prints, p. arXiv:1609.04747, Sep 2016

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for imagerecognition," inThe IEEE Conference on Computer Vision and PatternRecognition (CVPR), June 2016.

[6] L. Bottou, "Stochastic gradient descent tricks," inNeural networks:Tricks of the trade. Springer, 2012, pp. 421–436.