Hyper-parameter Optimization with Poisson Disc Sampling and Heuristics

Abhinav Shaw
UMass Amherst
Amherst, MA 01003
abhinavshaw@umass.edu

Abstract

Hyper-parameter Tuning has been an integral part of training Neural Networks, the existing techniques such as Grid Search, Random Search, etc. may not be able to perform a comprehensive search in the Search Space. There are other limitations that are discussed in this report. This project is an attempt to sample from the Search Space using Poisson Disc sampling and some additional Heuristic techniques. The algorithm for obtaining samples through Poisson Disc Sampling is based on a paper by Robert Bridson on Fast Poisson Disc Sampling which allows us to compute samples in linear time. The search for Hyper-parameter has been divided into two parts, a Sparse / Coarse search for finding the hot regions in the Search Space which is followed by a denser, more concentrated search. Since for the given Data Set we have prior insight that lower learning-rate gives us better convergence we try to sample more from the lower range of the Search Space. Hence in this project I have sampled the learning-rate in the log space. Thus, reinforcing the notion of how prior knowledge and insight of the Optimization problem at hand can be crucial in narrowing our Search Space and eliminating unimportant Hyper-parameters. The performance of each method is decided by comparing the Average Accuracy, Variance and Standard Deviation of 15 independent experiments on the test set.

1. Introduction

Performance of Neural Networks are dependent on Hyper-parameters, thus making it essential to find the best Hyper-parameters. This is another reminder to us that how critical, optimizing search algorithms for finding best Hyper-parameters is. It has great significance not only in the Industry but also in Machine Learning Research. We find that Pure Grid is taking every possible combination of the Hyper-parameters in the search Space, this is very granular and suffers from curse of Dimensionality on high number of dimensions [2]. Moreover, estimating the performance of the Neural Net over so many samples will be computationally intensive. Hence we use less granular Grid Search which has a possibility of skipping best Hyper-parameters in the Search Space. Random search has better results than grid search[2] but does not reproduce the results consistently. Hexagonal Search would do a more comprehensive search than Grid Search but it falls prey to patterns that can be observed from any direction as depicted in Fig 1. This is also known as the **The Corn Field Problem**.

This project is tries to explore Poisson Disc Sampling with some additional Heuristic techniques as an optimization algorithm over the search space. The algorithms used in the project are simple to understand and improve upon the best model obtained by Pure Grid Search.

2. Background

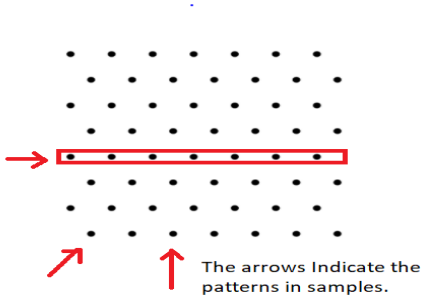
2.1. Literature Review

The classification problem on CIFAR-10 Data Set be denoted by *N*. To get the Best Model we minimize over the Loss function, however the optimization problem is dependent on Hyper-parameters λ. Hence we optimize over different λs and this problem of optimizing over Hyper-parameters is called *Hyper-parameter Optimization*[2]. Let $\psi(\lambda)$ denote the accuracy of the Neural Net for particular set of Hyper-parameters λ and ω denote the total number of samples from the Hyper-parameter search space from our sampling algorithm. The optimization problem becomes.

$$\lambda^* \approx \arg\max_{\lambda \in \omega} \psi(\lambda) = \hat{\lambda} \tag{1}$$

where $\hat{\lambda}$ is the Best Hyper-parameter. Now that we have defined the literature for our experiment, we discuss the literature for the Poisson Disc Sampling algorithm.

The implementation of Poisson Disc Sampling in this project is based on the algorithm in the paper *Fast Poisson Disk Sampling in Arbitrary Dimensions* by *Robert Bridson*. The Search Space is bounded by a rectangle with a length *l* and breadth *b*. In the algorithm we select *r* to be the minimum distance by which the samples have to be apart. *n* is the number of dimensions, in our case it is analogous to



(a) Hexagonal Sampling

(b) A corn field

Figure 1: Corn Field problem

the number of Hyper-parameter. k is the number of random samples generated within the spherical annulus[1] of r and $2r$. The Algorithms scales in Linear Time with a complexity of $O(n)$. This is because k is kept constant.

2.2. Random Experiment Evaluation

We will be taking ≈400 samples for each experiment. This is a safe number of samples for obtaining the best λ that lives in the Hyper-parameter Search Space. We can increase the number of sample much more than that, but conducting our experiment over too many samples will become computationally unfeasible. If we are worried about computation so much, one must arise with a question that why don't we reduce the number of samples to a very small number, this again will be detrimental to our experiment because the probability of us finding the best λ will be very low with very less samples. This will make our experiment highly unreliable.

Algorithm 1 Fast Poisson Disc Sampling

procedure GET SAMPLES
    Initialize n-dimensional *Grid* for storing samples.
    $CellSize \leftarrow r/\sqrt{n}$
    CellElements Initialized to -1
    $Sample \leftarrow$ random initial sample X_0
    $Grid \leftarrow Sample$
    $ActiveList.append(Sample)$
loop:While $ActiveListNotEmpty$
    Generate k random points between r and $2r$.
For each point:
    Check if anyother point within distance r.
    Only Near by Neighbours checked.
    If point point far apart, select point as new sample and add it to the $ActiveList$.
    Reject if no such point found and remove from $ActiveList$.

3. Approach

The project optimizes over two Hyper-parameter which are the learning-rate and the momentum. We do experiments on only these two Hyper-parameters because $\psi(\lambda)$ usually has very low effective dimensionality [2], for our model, these two are very important for convergence. Our Algorithms search the Search Space twice, first with a sparse search which enables us to find *Hot* regions in the Search Space. The best learning-rate and momentum from the sparse search decide the region for a concentrated search. An offset parameter (in percentage) is used to decide the new range for dense search with respect to the best Hyper-parameter found in the sparse search. The advantage of this method over doing an extremely dense search on the Search Space is that we save a lot of computation as the sampling doesn't remain in real time as we increase the number of samples. This is evident from the plot of number of samples with respect to the time taken to compute them Fig 2. The idea of hot regions is presented in Fig 3 where we can see that there are certain samples that give us higher accuracy than others and these can be considered *Hot* samples which eventually lead us to λ^*(Best λ).

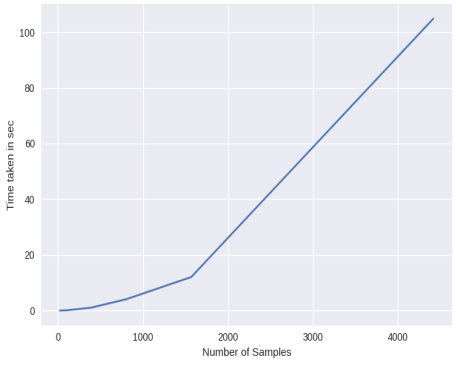


Figure 2: Number of samples vs Time Taken

The rectangular bound for Poisson Disc Search is fixed at *l=20* and *b=20*. The minimum distance between the samples is fixed at *r* = 2. The sampling range of the learning-rate is [1e-5,1e-2] and momentum is [0.7,1]. The first sample is always at (1e-5,0.7), which acts as a seed for other samples. The number of samples in Poisson Disc Sampling is given by the following equations.

$$Size = \frac{r}{\sqrt{n}} \tag{2}$$

$$n = NumberOfDimensions = 2 \tag{3}$$

$$Size = \frac{r}{\sqrt{2}} \tag{4}$$

$$NumberOfSamples = \frac{l}{Size} \times \frac{b}{Size} \tag{5}$$

The samples are collected in the scale of rectangular bounds i.e. between 0 and 20. These are converted to the range of the Hyper-parameter with the following formulae.

$$StepSize = UpperLr - LowerLr \tag{6}$$

$$lr = LowerLr + \frac{StepSize \times i}{l} \tag{7}$$

where $i \in range(0, (l/r+1))$ and LowerLr and UpperLr are the lower and upper limits of the learning-rate. The momentum can be calculated using the same equations. Once the best learning-rate and momentum are obtained from the sparse search we find the new upper and lower limits which will act as bounds for the dense search. Since the offset are in percentages we need to apply the following formulae to obtain the new ranges.

$$NewLrLower = BestLr - BestLr \times \frac{DenseLr_{Ofst}}{100} \tag{8}$$

$$NewLrUpper = BestLr + BestLr \times \frac{DenseLr_{Ofst}}{100} \tag{9}$$

The project also exploits prior knowledge of the optimum learning-rate. I convert the learning-rate by taking the log of it and taking the samples in the log space and then pushing it back to the original space by applying the transformation 10^{sample}. We can see how samples gather at the bottom in Fig 4. The figure also depicts how our Heuristic search will look like in the Search Space. This works better for CIFAR-10 Data Set, and would do so on other data sets as long as they converge better at lower ranges of learning-rate, generally of the order of $10^{-3} or 10^{-4}$. This is essentially a way of taking more samples at the Lower range of the learning-rate and then gradually reducing the number of samples as we move away from the LowerLimit. The samples are densely populated at the lower end of the Search Space which can be seen in Fig 4.

Table 1: Results of 5 Experiments.

Sr.	PureGS	PD HR	GS HR	PD HL	GS HL
1	34.7	35.5	36.1	36.3	35
2	34.7	34.6	34.7	35.9	35.6
3	33.5	36.4	35.4	36.6	37
4	33.6	35	35.7	35.4	35.4
5	34.5	36.9	36.9	35.4	37.7
6	34.6	36.3	35.3	35.4	36.6
7	34.9	35.3	35.9	35	37.6
8	34.4	37.2	34.4	35.3	35.8
9	34.9	36.7	34.7	36	36.2
10	34.6	36.1	36.3	36.3	36.6
11	34.7	37.5	36.1	35.2	35.8
12	35.3	35.9	34.2	35.3	35.3
13	37.1	35	36.4	36.2	37.5
14	36	35.9	34.3	35.6	33.6
15	33.9	35.8	36.2	36.1	37.3
μ	34.76	36.05	35.51	35.73	36.2
σ^2	0.81	0.71	0.75	0.24	1.31
σ	0.89	0.84	0.87	0.49	1.14

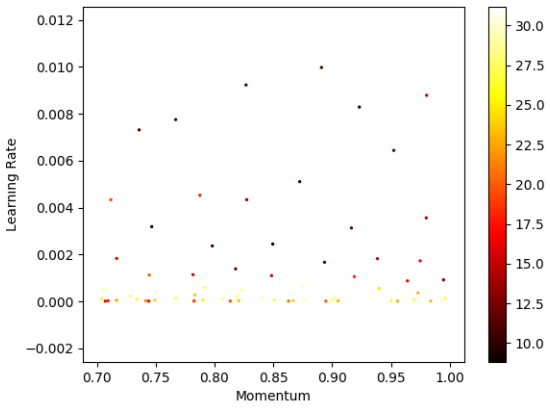
4. Experiment

4.1. Procedure

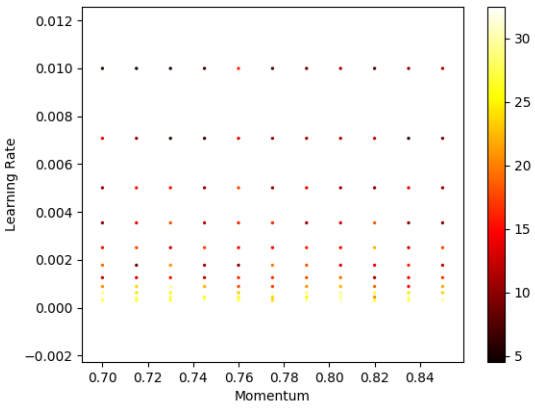
The Neural Network is a 2-layer Fully Connected Neural Net implemented in PyTorch. The best accuracy achieved by manually tuning the network was around 34.1% with a learning-rate of 10-3 and momentum of 0.85. To expedite the process, the model's weren't trained on the whole data set but a sub-set of 10000 images. The test set was a sample of 1000 images. The sub sampling doesn't affect our results because both the Grid Search and Poisson Disc Search will be subjected to the same constraints.

The Neural Network has 50 hidden dimensions and a loss function of Cross Entropy loss. The update rule for the Neural Net is *SGD*. The network is trained for 50 epochs, this is also a Hyper-parameter which could be sampled from Poisson Disc Search but has been kept fixed due to computational constraints. Moreover more than 50 epoch does not improve the test accuracy and just over fits the Neural Net on the training set.

Neural Network was trained with λs obtained form sparse search. Accuracy $\psi(\lambda)$ were recorded for every λ. The best λ form these λs was selected and the new Search Space was decided w.r.t to the selected λ. Accuracy from new λs were recorded, the λ with the highest $\psi(\lambda)$ was selected to be the best configuration for our Neural Net. This experiment was run 15 times for each Poisson Disc Search with and without logspace for learning-rate and Grid Search with and without logspace for learning-rate. The independent Random Experiments gave us a list of best Accuracy,

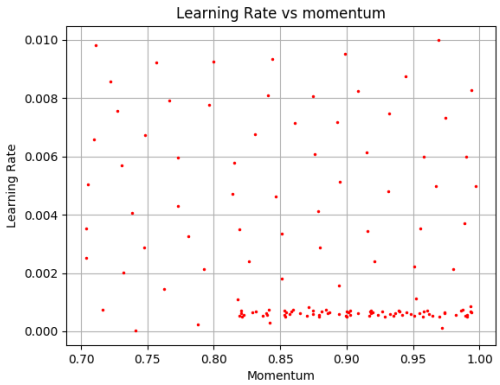


(a) Poisson Heat Scatter Plot

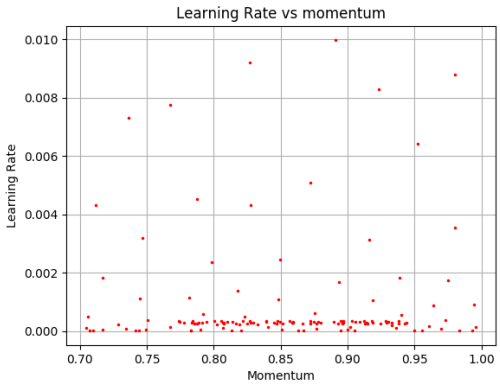


(b) Grid Heat Scatter Plot

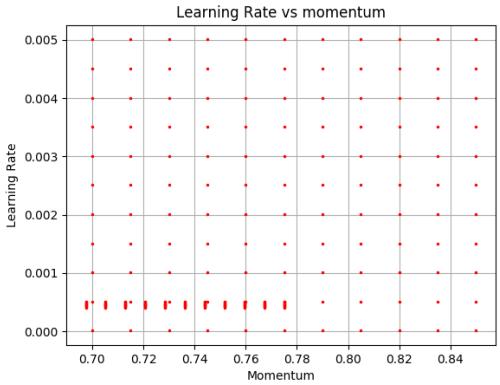
Figure 3: Heat Scatter plot, Color-bar represents the accuracy. The scatter plot is plotted with taking samples for the learning rate in log space hence we can observe the samples to be gathered at the bottom of the plot.



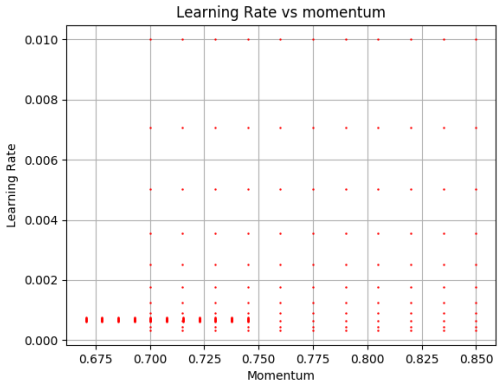
(a) Poisson Samples in regular space



(b) Poisson Samples in log space



(c) Grid Samples in regular space



(d) Grid Samples in log space

Figure 4: Sparse with Dense search for Poisson Disc and Grid Samples with learning-rate in Regular space and Log space.

on which the Mean , Variance and Standard Deviation were calculated.

4.2. Results

The results are summarized in the Table 1.

- GS - Grid Search.
- PD - Poisson Disc Sampling.
- H - Heuristic of Sparse Search and Dense Search.
- R - Learning-rate in Regular Space.
- L - Learning-rate in Log Space.

We see that Pure GS performs the worst out of all the algorithms experimented in this project. GS HR. which is Grid Search with Heuristics and Learning-rate in Regular Space gives us weaker performance when compared to PD HR, PD HL and GS HL. The best Average Accuracy is achieved by Grid Search With heuristics and Learning-rate in Log space. However the the Random Experiment has really high Variance and Standard Deviation, thus making the algorithm Unreliable. Overall PD HR gives us good results, it has high Average Accuracy and low Variance and Standard Deviation.

5. Future Work

There are certain techniques that the project does not explore.

- Hyper-parameter optimization was done in a very small subspace of the true Search Space. In the future we can extend the sub space to more number of dimensions and find out how our Algorithms perform.
- The Poisson Disc samples could be computed fast by using numPy optimization and a Quad Tree Implementation [4].
- These Experiments can be backed by rigourous theoretical proof.

6. Conclusion

PD HR, Poisson Disc Sampling with Heuristics can be considered as an optimal algorithm with consistent results and good mean Accuracy. Through the project's empirical evidence we have shown that configuring Neural Networks with Pure Grid Search might not give you great results and one must look deeper into the Search Space to find better models. Thus utilizing the Algorithms proposed in this report could bring us closer to the best configurations in the Search Space.

References

[1] Robert Bridson *Fast Poisson Disk Sampling in Arbitrary Dimensions.* University of British Columbia

[2] James Bergstra et al. *Random Search for Hyper-Parameter Optimizationn.* University of Montreal.

[3] Mohamed S. Ebeida and Scott A. Mitchell et al. *A Simple Algorithm for Maximal Poisson-Disk Sampling in High Dimensions.* Sandia National Laboratories, University of California Davis.

[4] Code for Poisson Disc Sampling