Entropy-SGD: Biasing Gradient Descent Into Wide Valleys

A short report on

Pratik Chaudhari, arXiv:1611.01838

Egorov E.E

Skoltech, ADASE

Moscow, 2018

Recall: Proximal Operator & Trust-Region

Consider **problem**: $\arg\min_x f(x)$

Consider **proximal operator**: $\mathrm{Prox}_f^\alpha(x) := \arg\min_{x'} \left[f(x') + \frac{1}{2}\|x' - x\|_2^2\right]$

Note, that $\mathrm{Prox}_f^\alpha(x^*) = x^*$ iff $x^* = \arg\min_x f(x)$

Hence, we can consider **Disappearing Tikhonov regularization**. At each step k we solve the problem:

$$\arg\min_x f(x) + \frac{1}{2\lambda}\|x - x^k\|_2^2$$

what is equivalent to steps

$$x^{k+1} = \mathrm{Prox}_f^\alpha(x^k)$$

Keep in mind

Motivation:
Improve convergence of some iterative method in such a way that the final result obtained is not affected by the regularization. This is done by shifting the 'center' of the regularization to the previous iterate.

Is it related to this paper or not? Let's discuss at the end.

At least, how solutions are related?

Motivation

Flat minimum is good/robust .etc! How do we estimate "flaness"? Eigenvalues of the hessian

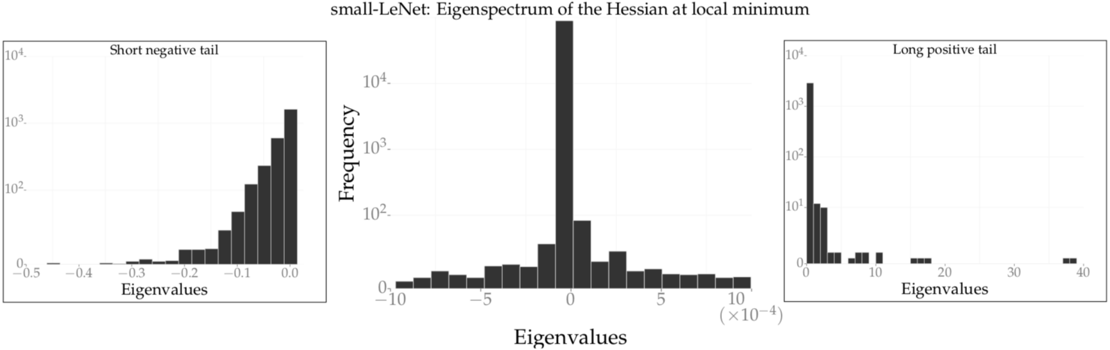


Figure 1: Eigenspectrum of the Hessian at a local minimum of a CNN on MNIST (two independent runs). **Remark:** The central plot shows the eigenvalues in a small neighborhood of zero whereas the left and right insets show the entire tails of the eigenspectrum.

Modified Gibbs Distribution

For any loss function $f(x)$ we can consider tempered Gibbs distribution:

$$P(x;\beta) \propto \exp(-\beta f(x))$$

As $\beta \to \infty$, probability mass concentrates on the global minimum $x^* = \arg\min_x f(x)$

Let's modify Gibbs distributions as:

$$P(x';x,\beta,\gamma) \propto \exp\left([-\beta[f(x') + \frac{\gamma}{2}\|x - x'\|_2^2]\right)$$

- $\gamma << 1$ all mass near x, no respect to $f(x')$
- $\gamma >> 1$ all mass near x^*, no respect to x (Gibss distribution)

Note: consider $\beta = 1$, as behavior depends on $\frac{\gamma}{\beta}$

DNN optimization

$$x \in \mathbb{R}^n := \text{ DNN weights}$$
$$\Xi := \text{ dataset with N samples}, \xi_k := \text{ sample from dataset}$$
$$f(x; \xi_k) := \text{loss value at point } \xi_k \text{ with weights } x$$

Thus, original problem:

$$x^* = \arg\min \frac{1}{N} \sum_{k=1}^{N} f(x; \xi_k)$$

Flat-Biased problem:

$$x_e^* = \arg\min_x -\log \int_{x' \in \mathbb{R}^n} \exp\left(-[f(x') + \frac{\gamma}{2}\|x - x'\|]\right) dx' = \arg\min_x -F(x; \gamma, x')$$

Gradient Step

For stochastic batch Ξ^l let's construct Modified-Gibss distribution:

$$q_e(x'|x, \gamma, \Xi^l) \propto \exp\left[-\left(\frac{1}{m}\sum_{i=1}^{m} f(x';\xi_i)\right) - \frac{\gamma}{2}\|x - x'\|_2^2\right]$$

Then gradient of our optimization problem is simple:

$$-\nabla_x F(x) = -\nabla_x \log \int_{x'} q_e(x')dx' = \gamma(x - \mathbb{E}_{q_e(x')}x')$$

But expectation is intractable.

Evaluation of Expectation

$p(x)$:= prior, $p(\xi_k|x)$:= likelihood

- Expectation is intractable $\to$ MCMC
- Batching MCMC $\to$ Stochastic Langevin Dynamics MCMC

Very brief intuition of this algorithm:

- MCMC: Dynamic + Metropolis-Hastings acceptation rule. Let's make dynamic
- MAP: $\arg\max_x \log p(x|\xi_{k\le N}) = \arg\max_x \log p(x) + \sum_{k=1}^{N} \log p(\xi_k|x) \to$ gradient ascent evolution
- Following Langevin, add random forces -> no convergence to point, fluctuations
- $\Delta x_t = \nabla \log p(x_t) + \sum_{k=1}^{N} \nabla \log p(\xi_k|x_t) + \sqrt{\eta}\varepsilon_t,\ \varepsilon_t \sim N(0,1)$
- Why this dynamics? Because it ok with stochastic batch ((Welling Teh, 2011) Note that in our problem authors consider "flat prior", so its grad vanishes

Algorithm

Algorithm 1: Entropy-SGD algorithm

Input : current weights x, Langevin iterations L

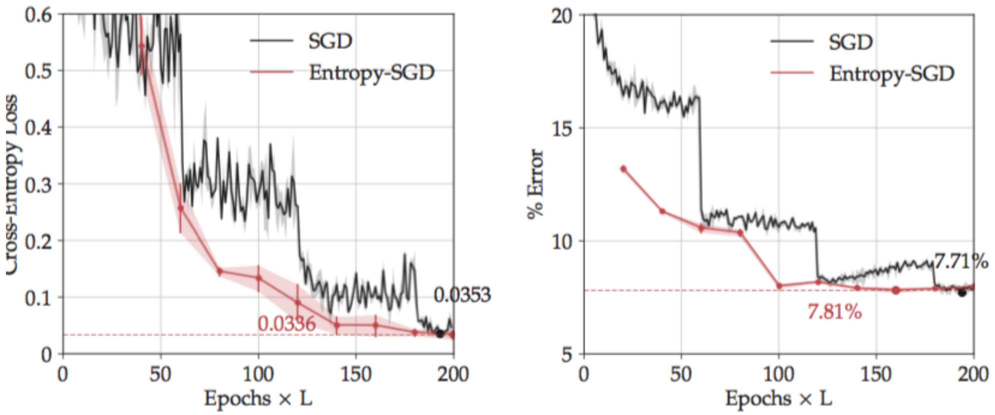
Hyper-parameters: scope γ, learning rate η, SGLD step size η'

```
// SGLD iterations;
x′, μ ← x;
for ℓ ≤ L do
    Ξ^ℓ ← sample mini-batch;
    dx′ ← (1/m) Σ_{i=1}^{m} ∇_{x′} f(x′; ξ_{ℓ_i}) − γ (x − x′);
    x′ ← x′ − η′ dx′ + √η′ ε N(0, I);
    μ ← (1 − α) μ + α x′;

// Update weights;
x ← x − η γ (x − μ)
```

Results: CIFAR

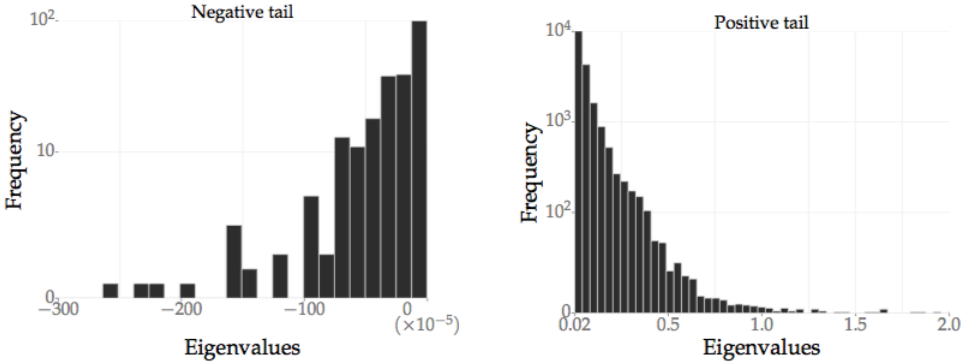
CIFAR-10, no augmentation, 200 epochs, SGD with Nesterov's momentum



(a) All-CNN-BN: Training loss (b) All-CNN-BN: Validation error

Results: CIFAR's Hessian

CIFAR-10, no augmentation, 200 epochs, SGD with Nesterov's momentum



Results

CNN, RNN results

Model	Entropy-SGD		SGD / Adam	
	Error (%) / Perplexity	Epochs	Error (%) / Perplexity	Epochs
mnistfc	1.37 ± 0.03	120	1.39 ± 0.03	66
LeNet	0.5 ± 0.01	80	0.51 ± 0.01	100
All-CNN-BN	7.81 ± 0.09	160	7.71 ± 0.19	180
PTB-LSTM	77.656 ± 0.171	25	78.6 ± 0.26	55
char-LSTM	1.217 ± 0.005	25	1.226 ± 0.01	40