Algorithms for Learning Good Step Sizes

Brian Zhang, Manikant Tiwari

with the guidance of Prof. Tim Roughgarden

Problem

- Let D be a hidden distribution of functions; i.e. we don't know D, but we can sample functions from D.
- Want to run gradient descent to optimize functions in D, but don't know what step size to choose
- Hope: Sample some functions from D. Choose some step sizes to test. Then use the step size that performs best on the sample.
- Question: How many samples to take from D? How many step sizes to try on each sample?

Motivation / Previous Work

- Common algorithms in machine learning rely on a good step size, to work well.
- Current techniques: brute force grid or random search on entire training set.
- Gupta and Roughgaraden [1] have derived a theoretical bound on the number of samples to take from D and the number of step sizes to try on each sample.
  - Bound predicts that, in order to learn a "good enough" step size with high probability, it suffices to sample $O(H^3)$ functions from D, and use a grid of size $(p_u - p_\ell)/K$ where H, K are quantities computable from properties of the functions in D, and ρ_u, ρ_ℓ are the smallest and largest step size intended to be tried.
  - Unknown if the above bound is tight, or what constant factors are.

Our Goal

- To test these bounds empirically in some simple cases
- To compute the constant factors in the big-O
- To apply the intuition behind this technique to a real world scenario

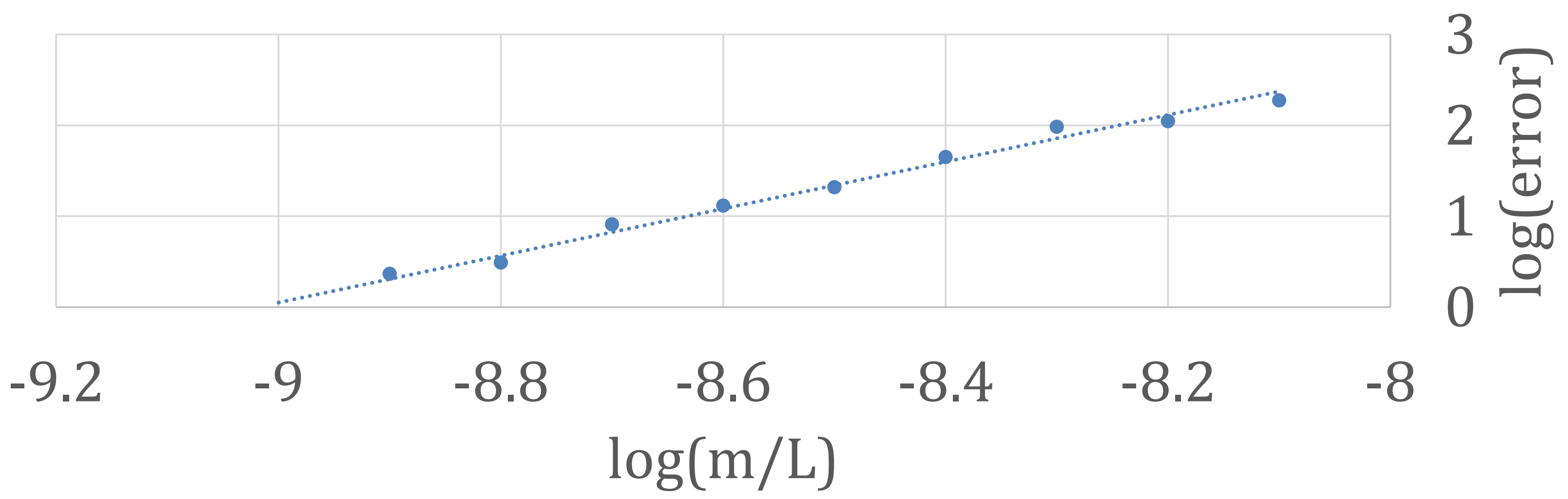
A Simple Distribution

- Consider quadratic functions $\frac{1}{2}ax^2$ for $a \in [m, L]$ distributed according to some distribution $p(a)$.
- Want to pick step size that maximizes progress per step toward minimum at $x = 0$. Can explicitly write down expression for best step size ρ^*:

$$\rho^* = \arg\max_{\rho} \int_m^L \frac{p(\alpha)\, d\alpha}{\log|1 - \alpha\rho|}$$

- Testing claim: take p to be the uniform distribution, vary parameters (e.g. ratio m/L). Then compute H and K, sample $O(H^3)$ functions, find $\hat{\rho}$ to minimize error on sample, and see if error (defined as difference in expected convergence time between $\hat{\rho}$ and ρ^*) changes.

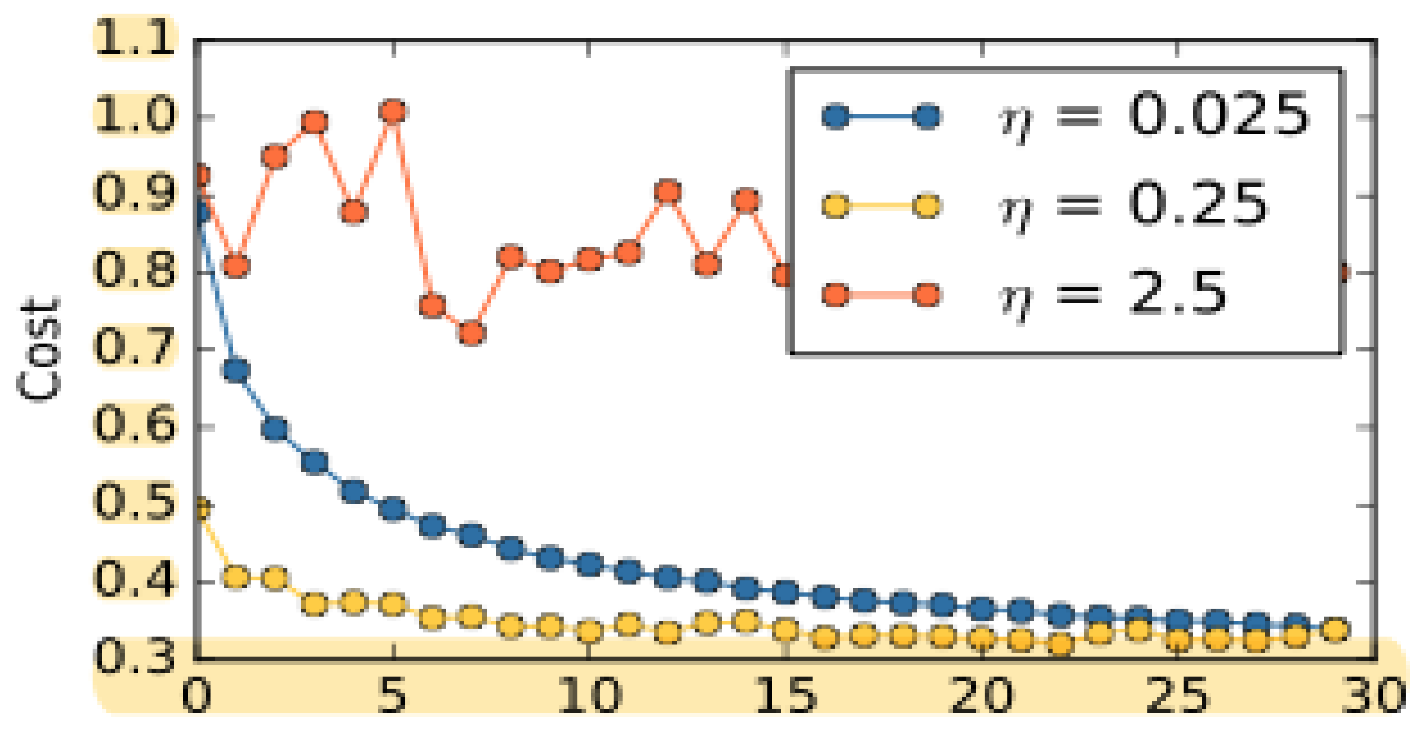
Findings & Analysis



- As distribution gets "harder" (e.g. m/L decreases) , error *decreases.* Suggests that bound can be tighter, at least for quadratic functions
  - If we instead sample about $O(\sqrt{H})$ samples (graph not depicted), error has nearly no relationship with parameters. Much better than $O(H^3)$!
- K is prohibitively small for reasonable parameter values. Can multiply K by (very large) constant factor without sacrificing error
- *In general, very few (<100) samples and very small grid (<100) needed to find good step size!*
- *Conjecture* (from empirical testing): Values are the same for general quadratic forms, independent of dimension!

Application: Neural Networks

- Basic model: three-layer neural network for digit classification problem (large training sets available).
- Still need to do more work to understand neural networks enough to apply theoretical results exactly
- General principle: with a small sample and a small grid search, we can find a step size that performs relatively well.
- Preliminary test: Take grid of size 3 (step sizes 2.5, 0.25, 0.025), vary sample size between 30 and 10000. See whether small samples can be used to tell which step size is best.
- Preliminary Results: Even with samples of size 100, we could discern that, of the three, $\eta = 0.25$ gave the best convergence properties. Below is a graph of the learning curves of the three step sizes for a fixed sample size.



References

1. R. Gupta and T. Roughgarden, "A PAC approach to application-specific algorithm selection," in *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, pp. 123-134, ACM, 2016.