Intractability of Learning the Discrete Logarithm with Gradient-Based Methods

Rustem Takhanov[1] Maxat Tezekbayev[1] Artur Pak[1] Arman Bolatov[1] Zhibek Kadyrsizova[1] Zhenisbek Assylbekov[2]

[1]Nazarbayev University [2]Purdue University Fort Wayne

Discrete logarithm problem

Definition

Let $(\mathcal{G},\circ)$ be a finite group, $a \in \mathcal{G}$ an element of prime order p, and $x \in \langle a\rangle$, where $\langle a\rangle := \{\underbrace{a\circ a\circ\ldots\circ a}_{k \text{ times}} : 0 \le k \le p-1\}$ is the cyclic group generated by a. The **discrete logarithm problem (DLP)** is finding the integer k, $0 \le k \le p-1$, such that

$$\underbrace{a\circ a\circ\ldots\circ a}_{k \text{ times}} = x.$$

This integer k is called the *index* of x to the base a, and we will denote it by $\mathrm{ind}_a x$.

Example (summation modulo p): $(\mathcal{G},\circ) = (\mathbb{Z}_p,+)$

Let $\cdot$ denote multiplication modulo p. The **discrete logarithm problem (DLP)** is finding the integer k, $0 \le k \le p-1$, such that

$$\underbrace{a+a+\ldots+a}_{k \text{ times}} = a\cdot k = x.$$

I.e.

$$\mathrm{ind}_a x = x\cdot a^{-1},$$

where a^{-1} is an inverse of a in multiplicative group $(\mathbb{Z}_p^*,\cdot)$.

Thus, for $(\mathcal{G},\circ) = (\mathbb{Z}_p,+)$ the DLP is equivalent to simple modular multiplication by a^{-1}.

Example (the elliptic curve group): $(\mathcal{G},\circ) = E(\mathbb{F}_p)$

Let $\mathbb{F}_p = (\mathbb{Z}_p,+,\cdot)$ and

$$E(\mathbb{F}_p) = \{(x,y)\in\mathbb{F}_p^2 \mid y^2 = x^3 + Ax + B\},$$

be an elliptic curve equipped with standard group structure. If $|E(\mathbb{F}_p)| = p$, then the **discrete logarithm problem (DLP)** over $E(\mathbb{F}_p)$ is also tractable.

So, DLP can be easy, but

Main claim of paper: the mapping $x \to \mathrm{ind}_a x$ for any group of prime cardinality p is hard to train using gradient-based algorithms.

GD Framework

Ohad Shamir considered the following class of gradient-based algorithms:

- Let a be some parameter that is randomly chosen in the beginning;
- The objective that an algorithm optimizes is $F(\mathbf{w},a)$ and $F(\mathbf{w},a)$ is highly sensitive to the choice of a;
- At every iteration $t = 1,\cdots,T$, the algorithm chooses a point $\mathbf{w}_t$ and receives (from an oracle) a vector $\mathbf{g}_t$ such that $\|\nabla F(\mathbf{w}_t,a) - \mathbf{g}_t\| < \varepsilon$.
- $\mathbf{w}_{t+1} = r_t(\{\mathbf{w}_i\}_1^t,\{\mathbf{g}_i\}_1^t)$.

Informally: If the $\mathrm{Var}_a[\nabla F(\mathbf{w},a)]$ is very small ($\ll T^{-3}$), then the latter algorithm will not succeed, because an information content of the gradient about the key parameter a is too small.

GD Framework: application to our case

- Let a be a base that is uniformly sampled from $\mathcal{G}/\{1\}$;
- Let $f_{\mathbf{w}} : \mathcal{G}/\{1\} \to \mathbb{R}$ be our architecture of NN;
- Let $l : \mathbb{R} \to \mathbb{R}$ be some 1-Lipshitz loss function;
- The objective is

$$F(\mathbf{w},a) = \mathbb{E}_{x\sim\mathcal{G}/\{1\}} l((-1)^{\mathrm{ind}_a x} f_{\mathbf{w}}(x)),$$

or

$$F(\mathbf{w},a) = \mathbb{E}_{x\sim\mathcal{G}/\{1\}} (\mathrm{ind}_a x - f_{\mathbf{w}}(x))^2$$

Informally: We prove that $\mathrm{Var}_a[\nabla F(\mathbf{w},a)] = \tilde{\mathcal{O}}(\frac{1}{\sqrt{p}})$. This means that the number of iterations needed should behave at least like $p^{1/6}$. For $p \sim 2^{512}$ we have $T \sim 10^{25}$.

Learning the last bit of $\mathrm{ind}_a x$

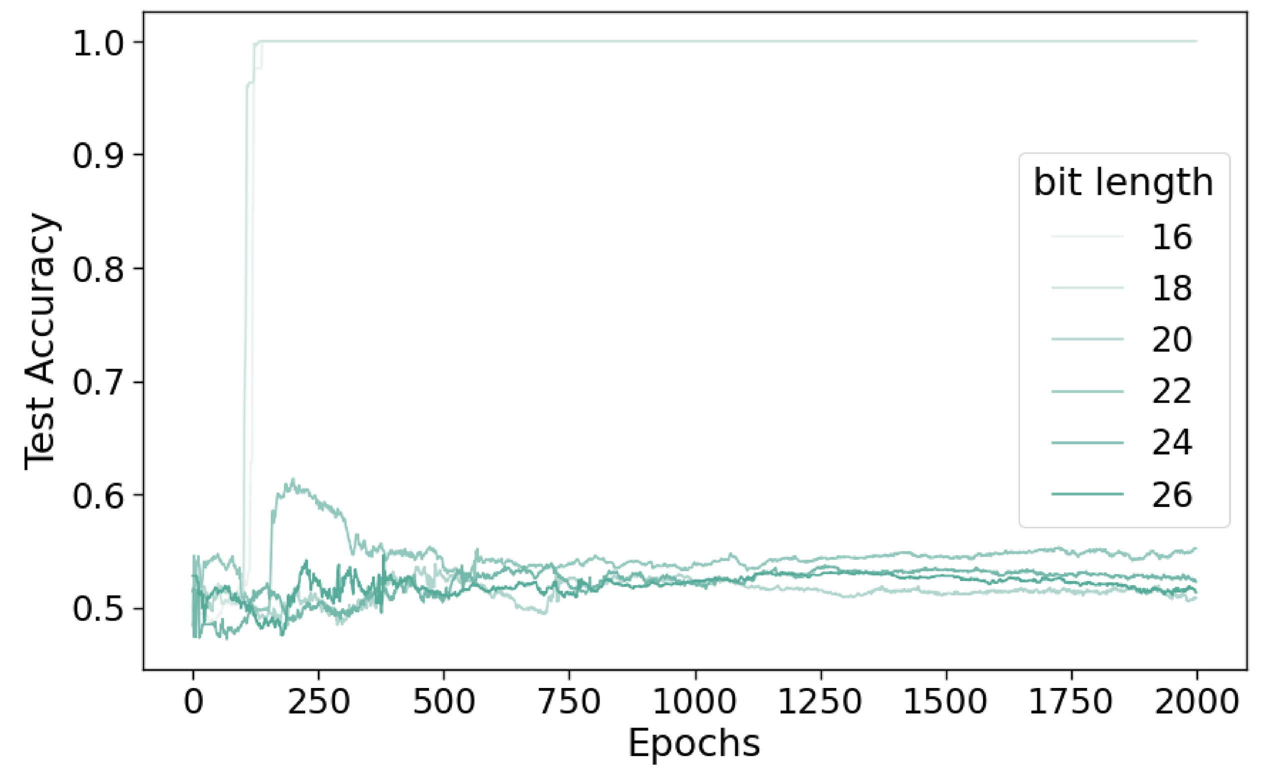

Figure 1. Learning with a 3-layer width-1000 dense network. Darker shades correspond to longer bitlengths. For each bitlength n, the group order p is chosen randomly from the prime numbers in the interval $[2^{n-1}, 2^n - 1]$.

Learning all bits

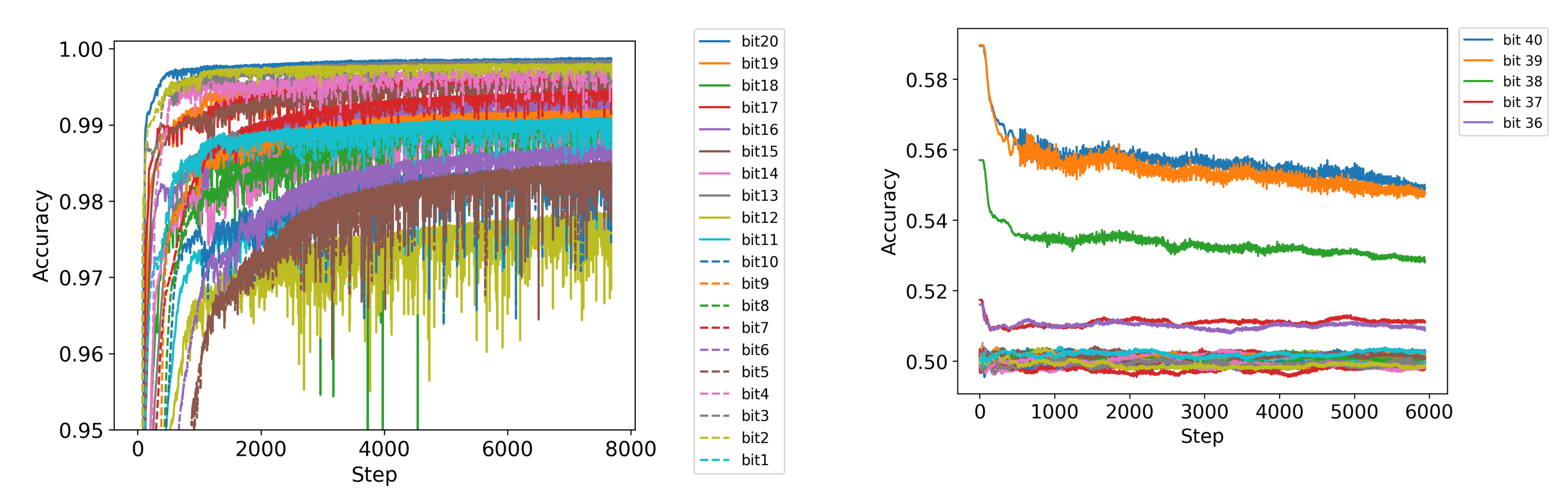

Figure 2. Test Accuracies when learning *all* bits of the discrete logarithm in $(\mathbb{Z}_p,+)$ with a single neural network. Bitlengths of p: 20 (left) and 40 (right).

Main result

Theorem

Suppose that $f_{\mathbf{w}}(\mathbf{x})$ is differentiable w.r.t. $\mathbf{w}$, and for some scalar $d(\mathbf{w})$, satisfies $\mathbb{E}_{X\sim\mathcal{G}\setminus\{1\}}\left[\left\|\frac{\partial}{\partial\mathbf{w}}f_{\mathbf{w}}(X)\right\|^2\right] \le d(\mathbf{w})^2$. Let the loss function ℓ be either the square loss $\ell(\hat{y},y) = \frac{1}{2}(\hat{y}-y)^2$ or a classification loss of the form $\ell(\hat{y},y) = s(\hat{y}\cdot y)$ for some 1-Lipschitz function s. Then

$$\mathop{\mathbb{E}}_{A\sim\mathcal{G}\setminus\{1\}} \|\nabla F(\mathbf{w},A) - \boldsymbol{\mu}(\mathbf{w})\|^2 \le \frac{c\cdot d(\mathbf{w})^2 \ln p}{\sqrt{p}}, \quad (1)$$

where $\boldsymbol{\mu}(\mathbf{w}) := \mathbb{E}_{A\sim\mathcal{G}\setminus\{1\}} \nabla F(\mathbf{w},A)$, and c is an absolute constant.

Low correlation of discrete logarithms

We computed the mean squared covariance

$$\mathop{\mathbb{E}}_{A,B\sim\mathbb{Z}_p^*}\left(\mathop{\mathrm{Cov}}_{X\sim\mathbb{Z}_p^*}[\mathrm{ind}_A X, \mathrm{ind}_B X]\right)^2 \quad (2)$$

for prime numbers in the interval $[3,500]$. The results are shown in Figure 3.

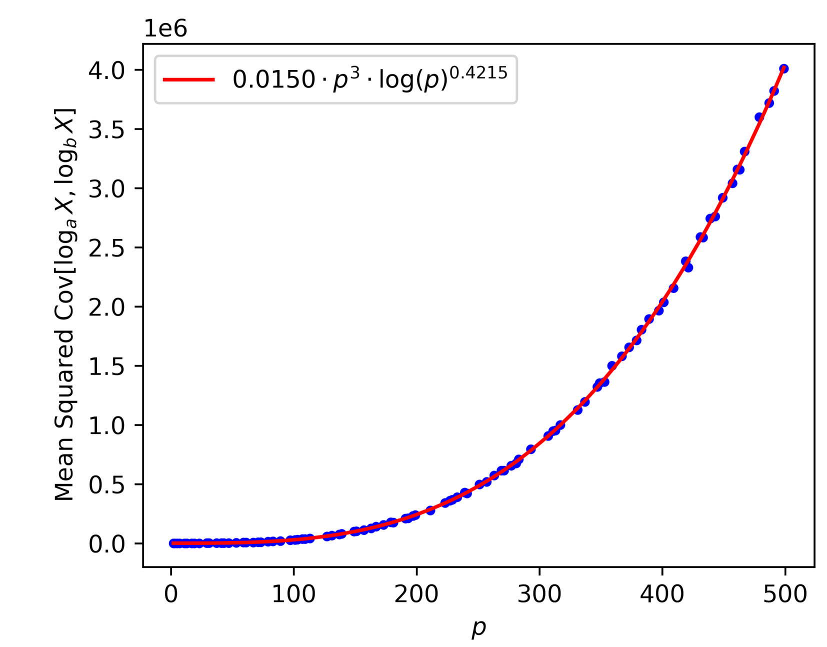

Figure 3. Mean squared covariance between two logarithms, $\mathrm{ind}_a X$ and $\mathrm{ind}_b X$, when X is a random variable uniformly distributed on $\mathbb{Z}_p^*$.

References

[1] Shai Shalev-Shwartz, Ohad Shamir, and Shaked Shammah. Failures of gradient-based deep learning. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 3067–3075. PMLR, 2017.

[2] Ohad Shamir. Distribution-specific hardness of learning neural networks. *J. Mach. Learn. Res.*, 19:32:1–32:29, 2018.

zassylbe@pfw.edu