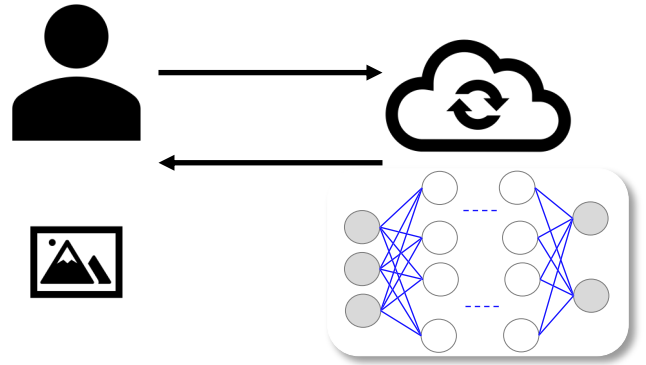
On Polynomial Approximations for Privacy-Preserving and Verifiable ReLU Networks

Ramy E. Ali, Jinhyun So and Salman Avestimehr

PPML Workshop, NeurIPS 2020

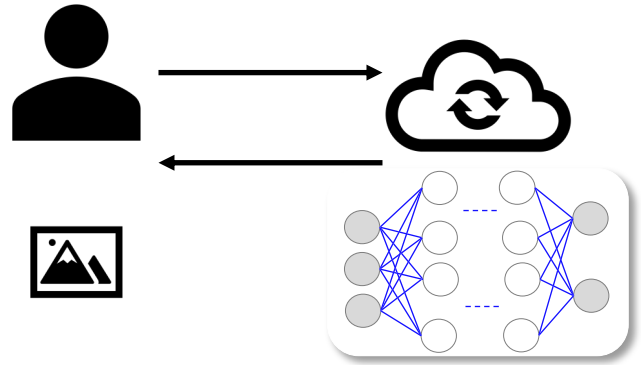
Introduction

- Outsourcing inference tasks raises several privacy and integrity concerns.
  - The users must verify the correctness of the computations.
  - The users may want to keep their data private.
  - The cloud also may want to keep its model private.



Introduction

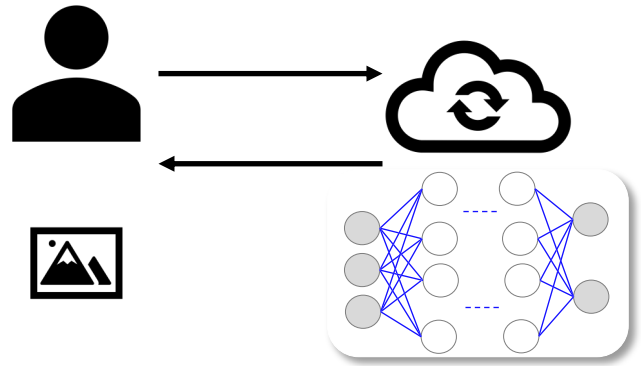
- Outsourcing inference tasks raises several privacy and integrity concerns.
  - The users must verify the correctness of the computations.
  - The users may want to keep their data private.
  - The cloud also may want to keep its model private.



- While there are many efficient privacy-preserving and verifiable techniques for polynomial-based computations [1-4], neural networks involve non-polynomial computations.

Introduction

- Outsourcing inference tasks raises several privacy and integrity concerns.
  - The users must verify the correctness of the computations.
  - The users may want to keep their data private.
  - The cloud also may want to keep its model private.



- While there are many efficient privacy-preserving and verifiable techniques for polynomial-based computations [1-4], neural networks involve non-polynomial computations.

- Hence, several frameworks as CryptoNets [5] and SafetyNets [6] replace or approximate the non-polynomial functions with polynomial functions.

Previous Work

- Much previous works as [5, 6] replace the ReLU function

  $$\sigma_r(x) = \max(x, 0)$$

  with the square function

  $$\sigma_{square}(x) = x^2.$$

- Max-pooling layers also are replaced with sum-pooling layers.
- This was shown empirically to work well for networks with small number of activation layers (3 or 4 layers).

This Work

- We empirically show that $\sigma_{square}(x) = x^2$ does not work well for deeper networks.

This Work

- We empirically show that $\sigma_{square}(x) = x^2$ does not work well for deeper networks.

- We instead propose

$$\sigma_{poly}(x) = x^2 + x.$$

This Work

- We empirically show that $\sigma_{square}(x) = x^2$ does not work well for deeper networks.
- We instead propose

$$\sigma_{poly}(x) = x^2 + x.$$

- This is motivated by the Minimax polynomial approximation

$$p_2(x) = \frac{1}{2}x^2 + \frac{1}{2}x + \frac{1}{16}.$$

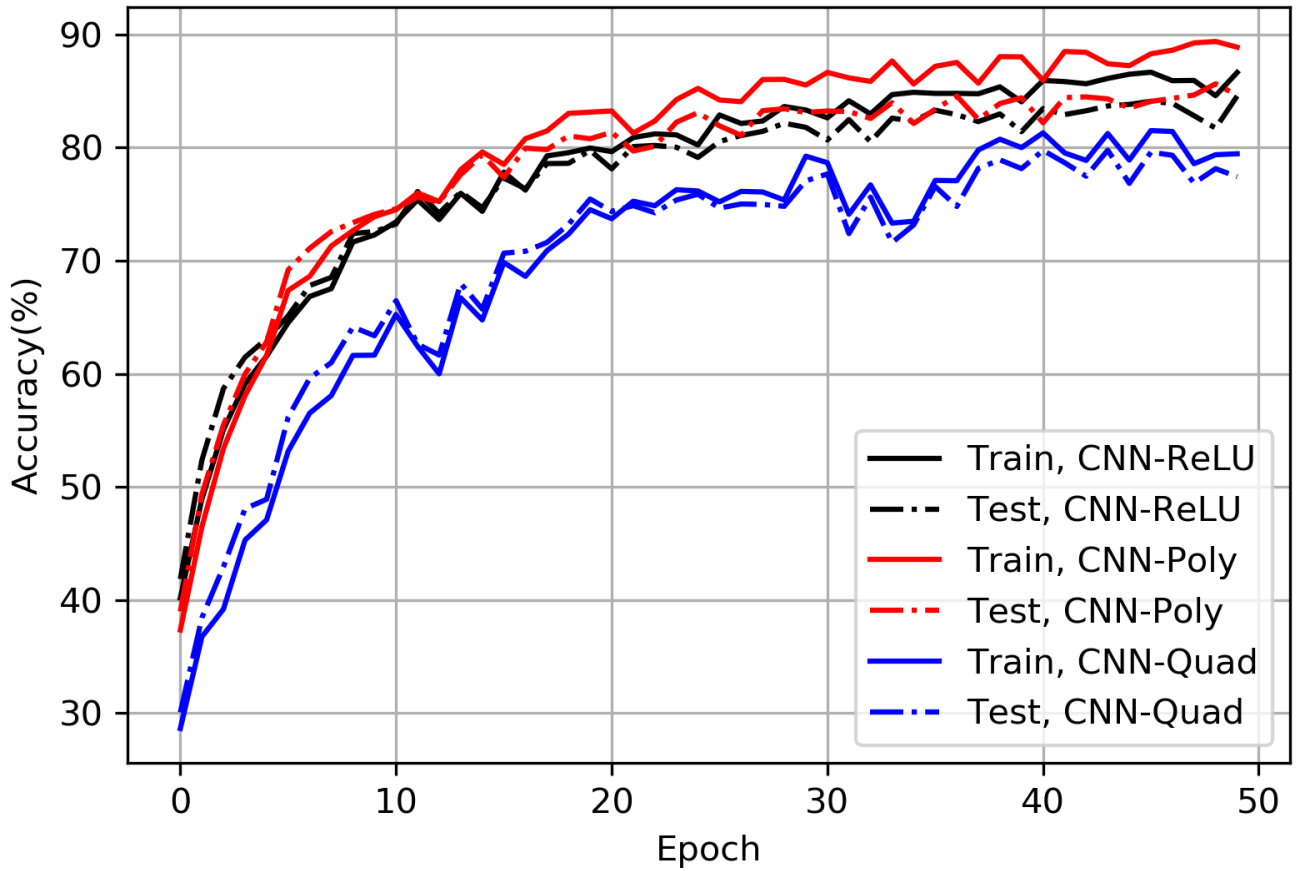
- Scaling this polynomial and ignoring the bias term lead to our activation function.

Evaluation

1. We consider the convolutional network of [7] with the CIFAR-10 dataset.

The network has

- 7 convolutional layers
- 7 ReLU activation layers
- 2 max-pooling layers
- a fully connected layer and
- a Softmax activation layer.

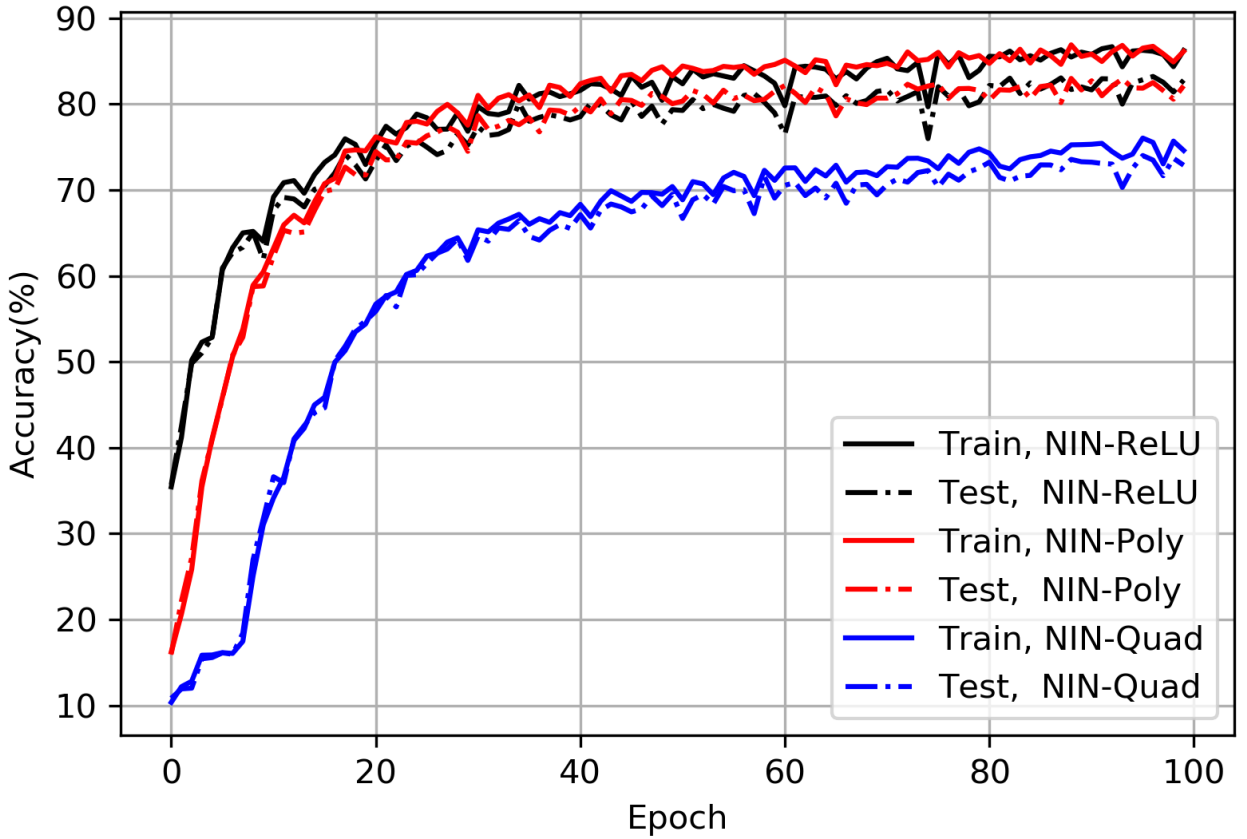


Evaluation

2. We consider the “Network In Network” architecture of [8] with the CIFAR-10 dataset.

The network has

- 9 convolutional layers
- 9 ReLU activation layers
- 2 max-pooling layers, Global pooling layer and
- a Softmax activation layer.



Discussion

- We have that empirically show that $\sigma_{poly}(x) = x^2 + x$ significantly outperforms $\sigma_{square}(x) = x^2$.

- Our future work aims to test our activation function on deeper networks and other datasets and to investigate its optimality.

Questions?

Thank you

E-mail: reali@usc.edu

References

[1] Ronald L Rivest, Len Adleman, Michael L Dertouzos, et al. **On data banks and privacy homomorphisms**. Foundations of secure computation, 4(11):169–180, 1978.

[2] Craig Gentry. **Fully homomorphic encryption using ideal lattices**. In Proceedings of the fortyfirst annual ACM symposium on Theory of computing, pages 169–178, 2009.

[3] Carsten Lund, Lance Fortnow, Howard Karloff, and Noam Nisan. **Algebraic methods for interactive proof systems**. Journal of the ACM (JACM), 39(4):859–868, 1992.

[4] Joppe W Bos, Kristin Lauter, Jake Loftus, and Michael Naehrig. **Improved security for a ring-based fully homomorphic encryption scheme**. In IMA International Conference on Cryptography and Coding, pages 45–64. Springer, 2013.

[5] Ran Gilad-Bachrach, Nathan Dowlin, Kim Laine, Kristin Lauter, Michael Naehrig, and John Wernsing. **Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy**. In International Conference on Machine Learning, pages 201–210, 2016.

[6] Zahra Ghodsi, Tianyu Gu, and Siddharth Garg. **Safetynets: Verifiable execution of deep neural networks on an untrusted cloud**. In Advances in Neural Information Processing Systems, pages 4672–4681, 2017.

[7] Jian Liu, Mika Juuti, Yao Lu, and Nadarajah Asokan. **Oblivious neural network predictions via minionn transformations**. In Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, pages 619–631, 2017.

[8] Min Lin, Qiang Chen, and Shuicheng Yan. **Network in network**. ICLR, 2014.