Distributed Newton-Type Methods with Communication Compression and Bernoulli Aggregation.

Rustem Islamov Xun Qian Slavomir Hanzely Mher Safaryan Peter Richtárik

MIPT Moscow Institute of Physics and Technology

上海人工智能实验室 Shanghai Artificial Intelligence Laboratory

IST Austria Institute of Science and Technology

جامعة الملك عبدالله للعلوم والتقنية King Abdullah University of Science and Technology

Xun Qian

Researcher

上海人工智能实验室
Shanghai Artificial Intelligence Laboratory

Mher Safaryan

Postdoctoral fellow

IST AUSTRIA
Institute of Science and Technology

Slavomir Hanzely

PhD student

Peter Richtárik

Professor of Computer Science

Outline

1. The Problem
2. 3PC Compression Mechanism
3. The 3 Special Newton-type Methods
4. Newton-3PC
5. Numerical Experiments

Outline

1. **The Problem**
2. 3PC Compression Mechanism
3. The 3 Special Newton-type Methods
4. Newton-3PC
5. Numerical Experiments

The Problem

Strongly convex

clients/devices

2nd order smooth

$$\min_{x \in \mathbb{R}^d} \left\{ f(x) \stackrel{\text{def}}{=} \frac{1}{n} \sum_{i=1}^{n} f_i(x) \right\}$$

model parameters

Empirical risk/loss

Local training data

Local loss function $f_i(x) = \mathbb{E}_{\xi \sim \mathcal{D}_i}[f_\xi(x)]$

Outline

1. The Problem
2. 3PC Compression Mechanism
3. The 3 Special Newton-type Methods
4. Newton-3PC
5. Numerical Experiments

3PC Compression Mechanism

$$\mathcal{C}_{\mathbf{H},\mathbf{Y}} : \underbrace{\mathbb{R}^{d\times d}}_{\mathbf{H}\in} \times \underbrace{\mathbb{R}^{d\times d}}_{\mathbf{Y}\in} \times \underbrace{\mathbb{R}^{d\times d}}_{\mathbf{X}\in} \to \mathbb{R}^{d\times d}$$

Parameters

Input

Peter Richtárik, Igor Sokolov, Ilyas Fatkhullin, Elnur Gasanov, Zhize Li, and Eduard Gorbunov. 3PC:. **Three point compressors for communication-efficient distributed training and a better theory for lazy aggregation.** ICML, 2022.

$$\mathbb{E}\left[\|\mathcal{C}_{\mathbf{H},\mathbf{Y}}(\mathbf{X}) - \mathbf{X}\|_{\mathrm{F}}^2\right] \leq (1-A)\|\mathbf{H}-\mathbf{Y}\|_{\mathrm{F}}^2 + B\|\mathbf{X}-\mathbf{Y}\|_{\mathrm{F}}^2$$

$A \in (0,1]$

$B \geq 0$

3PC Compression Mechanism: Examples

Contractive Compressors:

$$\exists \alpha \in (0,1] : \mathbb{E}\left[\|\mathcal{C}(\mathbf{X}) - \mathbf{X}\|_{\mathrm{F}}^2\right] \leq (1-\alpha)\|\mathbf{X}\|_{\mathrm{F}}^2 \quad \forall \mathbf{X} \in \mathbb{R}^{d\times d}$$

Compressed Lazy Aggregation:

$$\exists \alpha \in (0,1], \zeta \geq 0 : \mathcal{C}_{\mathbf{H},\mathbf{Y}} = \begin{cases} \mathbf{H} + \mathcal{C}(\mathbf{X} - \mathbf{H}) & \text{if } \|\mathbf{H} - \mathbf{X}\|_{\mathrm{F}}^2 > \zeta\|\mathbf{X} - \mathbf{Y}\|_{\mathrm{F}}^2 \\ \mathbf{H} & \text{if otherwise} \end{cases}$$

Contractive compressor with parameter α

Compressed Bernoulli Aggregation:

$$\exists \alpha \in (0,1], p \in (0,1] : \mathcal{C}_{\mathbf{H},\mathbf{Y}} = \begin{cases} \mathbf{H} + \mathcal{C}(\mathbf{X} - \mathbf{H}) & \text{with probability } p \\ \mathbf{H} & \text{with probability } 1-p \end{cases}$$

Outline

Newton Method

$$x^{k+1} = x^k - \Bigg(\frac{1}{n} \sum_{i=1}^{n} \nabla^2 f_i(x^k) \Bigg)^{-1} \Bigg(\frac{1}{n} \sum_{i=1}^{n} \nabla f_i(x^k) \Bigg)$$

Can be computed locally

Expensive to communicate: $\mathcal{O}(d^2)$

Can be computed locally

Easy to communicate: $\mathcal{O}(d)$

- $\mathcal{O}(d)$ communication cost per round
- Implementability in practice
- Local quadratic convergence rate independent of the condition number

Hessian Lipschitz constant

$$\|x^{k+1} - x^*\| \le \frac{L}{2\mu} \|x^k - x^*\|^2$$

Strong convexity constant

Local quadratic rate

Newton Star Method

$$x^{k+1} = x^k - \left(\frac{1}{n} \sum_{i=1}^{n} \boxed{\nabla^2 f_i(x^*)} \right)^{-1} \left(\frac{1}{n} \sum_{i=1}^{n} \boxed{\nabla f_i(x^k)} \right)$$

Can NOT be computed locally

Single communication of $\mathcal{O}(d^2)$

Can be computed locally

Easy to communicate: $\mathcal{O}(d)$

Rustem Islamov, Xun Qian and Peter Richtárik
Distributed second order methods with fast rates and compressed communication,
ICML 2021.

- ✓ $\mathcal{O}(d)$ communication cost per round
- ✗ Implementability in practice
- ✓ Local quadratic convergence rate independent of the condition number

Hessian Lipschitz constant

$$\|x^{k+1} - x^*\| \leq \frac{L}{2\mu} \|x^k - x^*\|^{2}$$

Strong convexity constant

Local quadratic rate

Newton Zero Method

$$x^{k+1} = x^k - \left(\frac{1}{n}\sum_{i=1}^{n} \nabla^2 f_i(x^0)\right)^{-1} \left(\frac{1}{n}\sum_{i=1}^{n} \nabla f_i(x^k)\right)$$

Can be computed locally

Single communication of $\mathcal{O}(d^2)$

Can be computed locally

Easy to communicate: $\mathcal{O}(d)$

- ✓ $\mathcal{O}(d)$ communication cost per round*
- ✓ Implementability in practice
- ✗ Local quadratic convergence rate

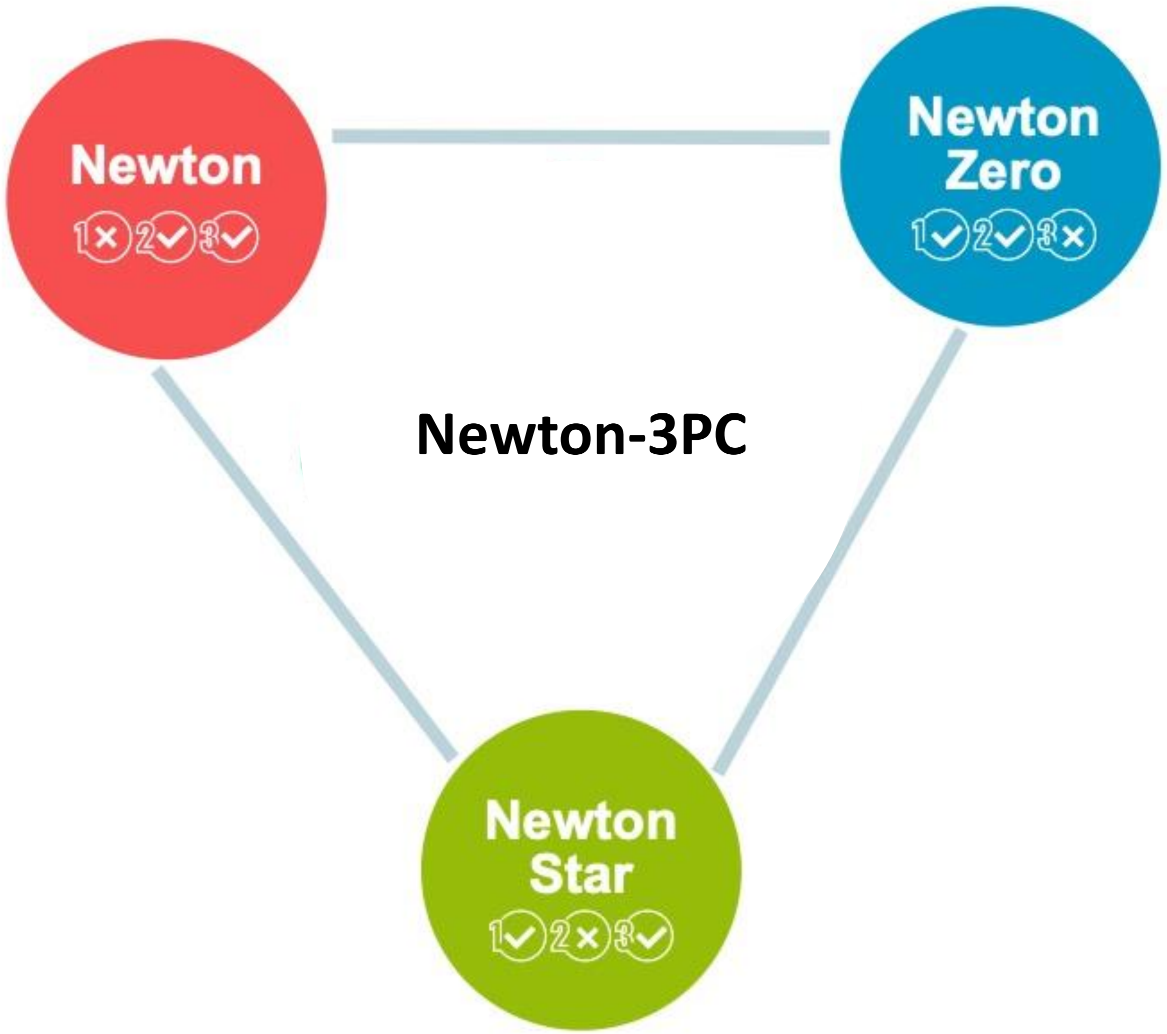
$$\|x^{k+1} - x^*\| \leq \frac{1}{2}\|x^k - x^*\|$$

Local (fixed) linear rate: $\|x^0 - x^*\| \leq \frac{\mu}{2L}$

Strong convexity constant

Hessian Lipschitz constant

“Newton Triangle”



Outline

1. The Problem
2. Brief Comparison with Related Works
3. The 3 Special Newton-type Methods
4. Newton-3PC
5. Numerical Experiments

Learning the Optimal Hessian Matrices

Newton Star

$$x^{k+1} = x^k - \left(\frac{1}{n}\sum_{i=1}^n \nabla^2 f_i(x^*)\right)^{-1} \nabla f(x^k)$$

Idea! Learn the optimal Hessians $\nabla^2 f_i(x^*)$ in communication efficient manner:

$\boxed{(i)}\ \mathbf{H}_i^k \to \nabla^2 f_i(x^*)$ as $k \to \infty$ $\quad \boxed{(ii)}\ \mathbf{H}_i^{k+1} - \mathbf{H}_i^k$ is compressed

$$x^{k+1} = x^k - \left(\frac{1}{n}\sum_{i=1}^n \mathbf{H}_i^k\right)^{\boxed{-1}} \nabla f(x^k)$$

$$= x^k - \left(\mathbf{H}^k\right)^{\boxed{-1}} \nabla f(x^k)$$

PDF Adobe

Rustem Islamov, Xun Qian and Peter Richtárik
Distributed second order methods with fast rates and compressed communication,
ICML 2021.

FedNL: Two Options for Updating the Global Model

Option 1

$$x^{k+1} = x^k - \left(\left[\mathbf{H}^k\right]_{\boxed{\mu}}\right)^{-1} \nabla f(x^k)$$

Projection onto the cone of positive definite matrices

Option 2

$$x^{k+1} = x^k - \left(\mathbf{H}^k + \boxed{l^k}\mathbf{I}\right)^{-1} \nabla f(x^k)$$

$$l^k = \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{H}_i^k - \nabla^2 f_i(x^k)\|_{\mathrm{F}}$$

Newton-3PC: New Hessian Learning Technique

$$\mathbf{H}_i^{k+1} = \mathcal{C}_{\mathbf{H}_i^k, \nabla^2 f_i(x^k)}\left(\nabla^2 f_i(x^{k+1})\right)$$

3PC- Compression operator

Peter Richtárik, Igor Sokolov, Ilyas Fatkhullin, Elnur Gasanov, Zhize Li, and Eduard Gorbunov. 3PC:. **Three point compressors for communication-efficient distributed training and a better theory for lazy aggregation.** ICML, 2022.

FedNL: Hessian Learning Rate Options

Assumption 4.1. *The average loss f is μ-strongly convex, and all local losses $f_i(x)$ have Lipschitz continuous Hessians.*

$$\|\nabla^2 f_i(x) - \nabla^2 f_i(y)\| \quad \leq L_* \|x - y\|,$$

$$\Phi^k := \mathcal{H}^k + 6\left(\frac{1}{A} + 3AB\right) L_{\mathrm{F}}^2 \|x^k - x^*\|^2, \quad \text{where} \quad \mathcal{H}^k := \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{H}_i^k - \nabla^2 f_i(x^*)\|_{\mathrm{F}}^2$$

Can be relaxed just for k=0

Theorem 4.2. *Let Assumption 4.1 hold. Assume $\|x^0 - x^*\| \leq \frac{\mu}{\sqrt{2D}}$ and $\mathcal{H}^k \leq \frac{\mu^2}{4C}$ for all $k \geq 0$. Then,* Newton-3PC *(Algorithm 1) with any 3PC mechanism converges with the following rates:*

$$\|x^k - x^*\|^2 \leq \frac{1}{2^k}\|x^0 - x^*\|^2, \qquad \mathbb{E}\left[\Phi^k\right] \leq \left(1 - \min\left\{\frac{A}{2}, \frac{1}{3}\right\}\right)^k \Phi^0,$$

$$\mathbb{E}\left[\frac{\|x^{k+1} - x^*\|^2}{\|x^k - x^*\|^2}\right] \leq \left(1 - \min\left\{\frac{A}{2}, \frac{1}{3}\right\}\right)^k \left(C + \frac{AD}{12(1+3AB)L_{\mathrm{F}}^2}\right)\frac{\Phi^0}{\mu^2}.$$

Algorithm 1 Newton-3PC (Newton's method with three point compressor)

Input: $x^0 \in \mathbb{R}^d$, $\mathbf{H}_1^0, \ldots, \mathbf{H}_n^0 \in \mathbb{R}^{d \times d}$, $\mathbf{H}^0 := \frac{1}{n}\sum_{i=1}^n \mathbf{H}_i^0$, $l^0 = \frac{1}{n}\sum_{i=1}^n \|\mathbf{H}_i^0 - \nabla^2 f_i(x^0)\|_{\mathrm{F}}$.

on server

  Option 1: $x^{k+1} = x^k - [\mathbf{H}^k]_\mu^{-1} \nabla f(x^k)$

  Option 2: $x^{k+1} = x^k - [\mathbf{H}^k + l^k \mathbf{I}]^{-1} \nabla f(x^k)$

  Broadcast x^{k+1} to all nodes

for each device $i = 1, \ldots, n$ in parallel **do**

  Get x^{k+1} and compute local gradient $\nabla f_i(x^{k+1})$ and local Hessian $\nabla^2 f_i(x^{k+1})$

  Apply 3PC and update local Hessian estimator to $\mathbf{H}_i^{k+1} = \mathcal{C}_{\mathbf{H}_i^k, \nabla^2 f_i(x^k)}\left(\nabla^2 f_i(x^{k+1})\right)$

  Send $\nabla f_i(x^{k+1})$, $\mathbf{H}_i^{k+1}$ and $l_i^{k+1} := \|\mathbf{H}_i^{k+1} - \nabla^2 f_i(x^{k+1})\|_{\mathrm{F}}$ to the server

end for

on server

  Aggregate $\nabla f(x^{k+1}) = \frac{1}{n}\sum_{i=1}^n \nabla f_i(x^{k+1}), \mathbf{H}^{k+1} = \frac{1}{n}\sum_{i=1}^n \mathbf{H}_i^{k+1}, l^{k+1} = \frac{1}{n}\sum_{i=1}^n l_i^{k+1}$

FedNL: Assumptions

FedNL: Local Convergence Theory

$$\|x^k - x^*\|^2 \boxed{\leq} \frac{1}{2^k}\|x^0 - x^*\|^2$$

Local (fixed) linear rate

Lyapunov function

$$\Phi^k = \mathcal{H}^k + 6\left(\frac{1}{\boxed{A}} + 3\boxed{A}\boxed{B}\right) L^2\|x^k - x^*\|^2$$

$$\mathbb{E}[\boxed{\Phi^k}] \boxed{\leq} \left(1 - \min\left\{\frac{\boxed{A}}{2}, \frac{1}{3}\right\}\right)\boxed{\Phi^0}$$

Local linear rates

$x^k \to x^*$

$\mathbf{H}_i^k \to \nabla^2 f_i(x^*)$ for all $i \in [n]$

Constant depending on the choice of the compressor and stepsize

Outline

Experiments: Regularized Logistic Regression

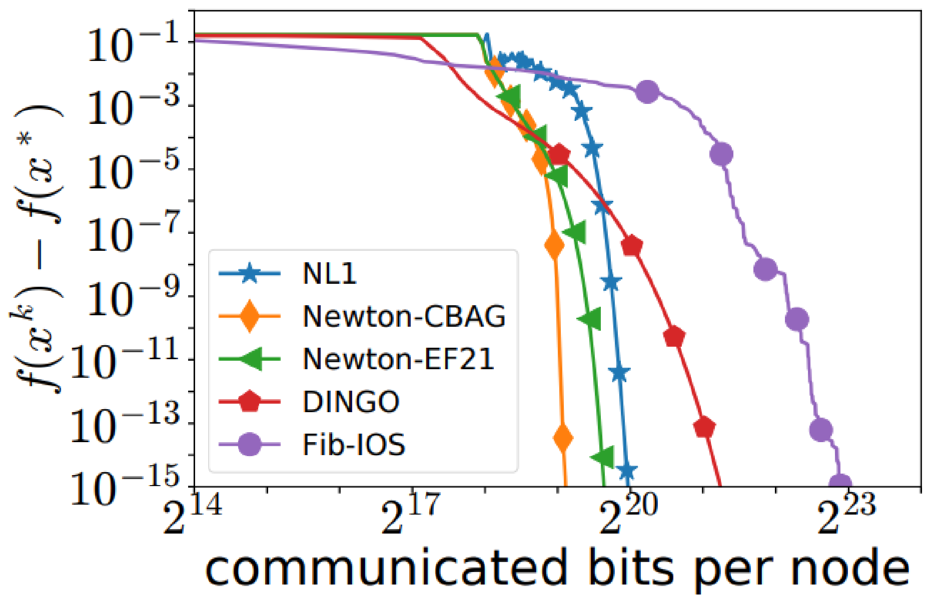
Regularization parameter

Training data points

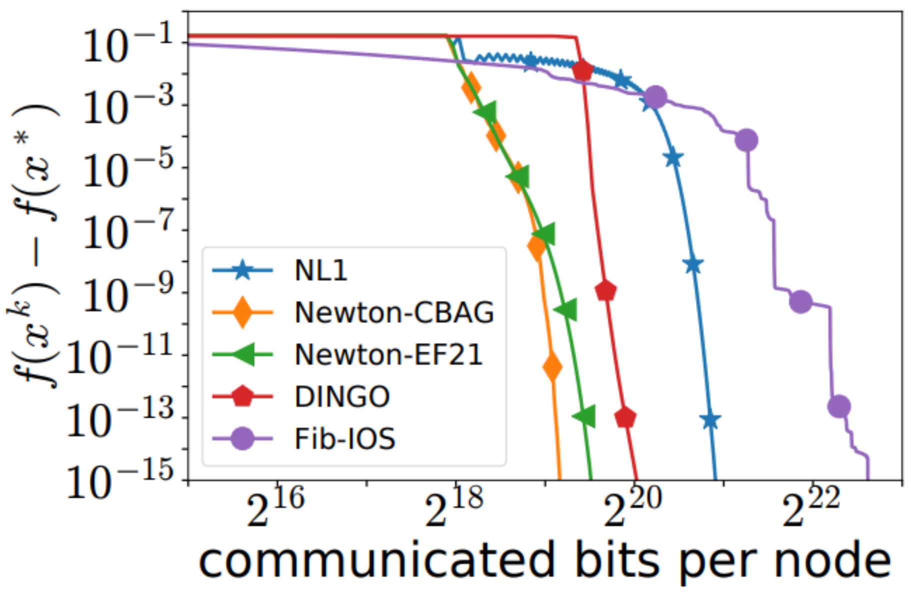
$$\min_{x \in \mathbb{R}^d} \left\{ f(x) := \frac{1}{n} \sum_{i=1}^{n} f_i(x) + \frac{\lambda}{2} \|x\|^2 \right\}, \qquad f_i(x) = \frac{1}{m} \sum_{j=1}^{m} \log\left(1 + \exp(-b_{ij} a_{ij}^\top x)\right),$$

where $\{a_{ij}, b_{ij}\}_{j \in [m]}$ are data points at the i-th device. The datasets were taken from LibSVM library [Chang and Lin, 2011]: `a1a`, `a9a`, `w7a`, `w8a`, and `phishing`.

Experiments: Local Comparisong against SOM

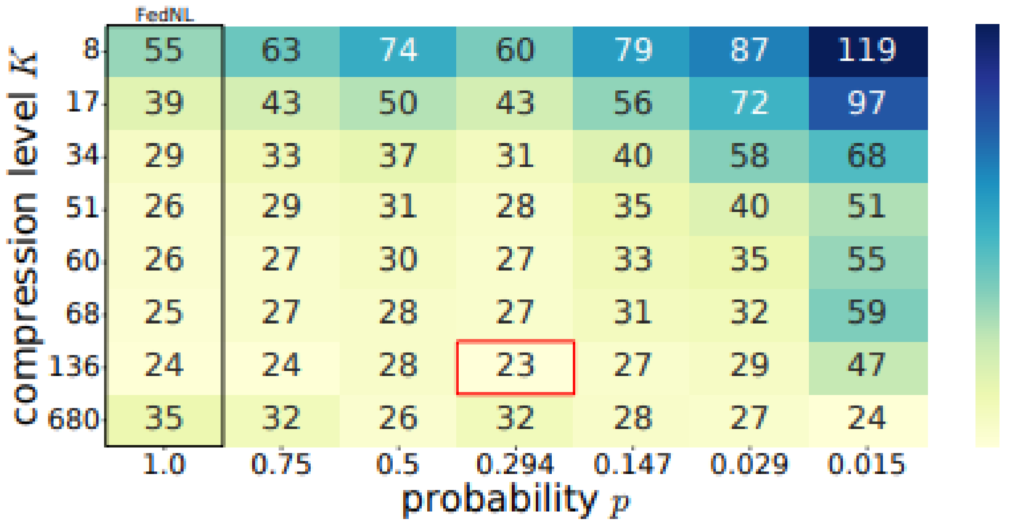


(a) a1a, $\lambda = 10^{-3}$

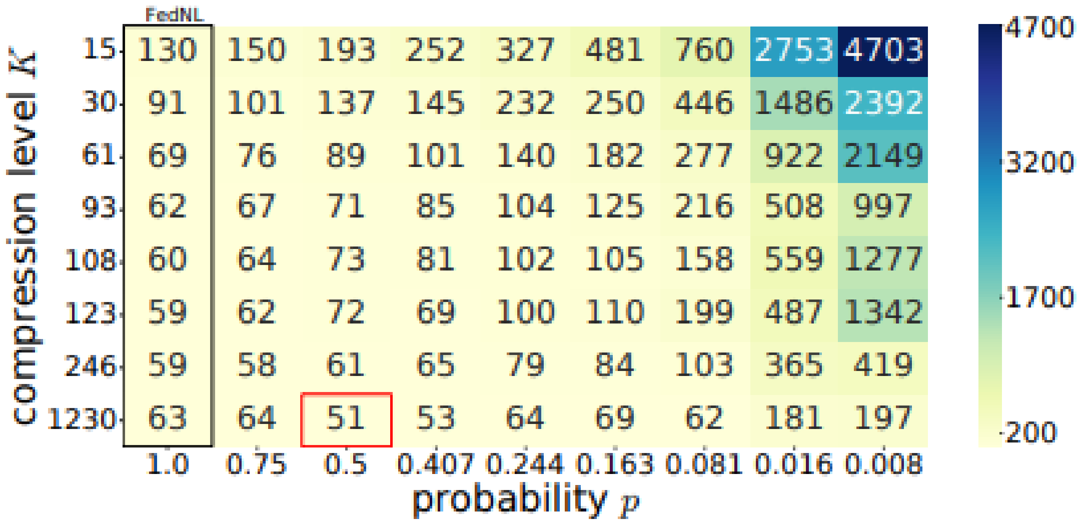


(c) a9a, $\lambda = 10^{-3}$

Experiments: Does Bernoulli Aggregation Bring Benefit?



(e) phishing, $\lambda = 10^{-3}$



(f) a1a, $\lambda = 10^{-4}$

Thank you