Loopless stochastic methods

by Vsevolod Klyushev (v.klyushev@innopolis.university)
& Dmitry Beresnev (d.beresnev@innopolis.university)
github.com/dsomni/omml-project-f23

01 Problem

Finite-sum minimization

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

Assumption 1 (L–smoothness) *Functions $f_i : \mathbb{R}^d \to \mathbb{R}$ are L–smooth for some $L > 0$:*

$$f_i(y) \le f_i(x) + \langle \nabla f_i(x), y - x\rangle + \frac{L}{2}\|y - x\|^2, \qquad \forall x, y \in \mathbb{R}^d.$$

Assumption 2 (μ–strong convexity) *Function $f : \mathbb{R}^d \to \mathbb{R}$ is μ–strongly convex for $\mu > 0$:*

$$f(y) \ge f(x) + \langle \nabla f(x), y - x\rangle + \frac{\mu}{2}\|y - x\|^2, \qquad \forall x, y \in \mathbb{R}^d.$$

Note

PAGE

PAGE is used for nonconvex finite-sum problems that satisfies average L-smoothness assumption:

A function $f : \mathbb{R}^d \to \mathbb{R}$ is average L-smooth if $\exists L > 0$,

$$\mathbb{E}[\|\nabla f_i(x) - \nabla f_i(y)\|] \leq L^2\|x - y\|^2, \forall x, y \in \mathbb{R}^d$$

SARAH, SVRG, L-SVRG

SARAH, SVRG and L-SVRG are used for convex finite-sum problems that satisfies L-smoothness and μ-strong convexity

02 SVRG vs L-SVRG

Don't Jump Through Hoops and Remove Those Loops: SVRG and Katyusha are Better Without the Outer Loop

SVRG Algorithm

Stochastic Variance-Reduced Gradient method

Input: learning rate $\gamma > 0$, epoch length m, starting point $x^0 \in \mathbb{R}^d$

$\phi = x^0$

for $s = 0, 1, 2, \ldots$ **do**

  for $k = 0, 1, 2, \ldots, m-1$ **do**

    Sample $i \in \{1, \ldots, n\}$ uniformly at random

    $g^k = \nabla f_i(x^k) - \nabla f_i(\phi) + \nabla f(\phi)$

    $x^{k+1} = \text{prox}_{\gamma R}(x^k - \gamma g^k)$

  end for

  $\phi = x^0 = \frac{1}{m}\sum_{k=1}^{m} x^k$

end for

$$O\left(n + \frac{n^{2/3}}{\epsilon^2}\right)$$

L-SVRG Algorithm

Loopless Stochastic Variance-Reduced Gradient method

Parameters: stepsize $\eta > 0$, probability $p \in (0, 1]$
Initialization: $x^0 = w^0 \in \mathbb{R}^d$
for $k = 0, 1, 2, \ldots$ **do**

$g^k = \nabla f_i(x^k) - \nabla f_i(w^k) + \nabla f(w^k)$ ($i \in \{1, \ldots, n\}$ is sampled uniformly at random)

$x^{k+1} = x^k - \eta g^k$

$$w^{k+1} = \begin{cases} x^k & \text{with probability } p \\ w^k & \text{with probability } 1 - p \end{cases}$$

end for

$$O((\frac{L}{\mu}) \log \frac{1}{\epsilon})$$

L-SVRG Convergence

Gradient learning quantity: $\mathcal{D} = \frac{4\eta^2}{pn}\sum_{i=1}^{n}\|\nabla f_i(w^k) - \nabla f_i(x^*)\|^2$

Lyapunov function: $\Phi^k = \|x^k - x^*\|^2 + \mathcal{D}^k$

Lemma 1:
Upper bounds the expected squared distance of x^{k+1} from x^* in terms of the same distance but for x^k, function suboptimality, and second momentum of g^k.

$$E[\|x^{k+1} - x^*\|^2] \leq (1-\eta\mu)\|x^k - x^*\|^2 - 2\eta(f(x^k) - f(x^*)) + \eta^2 E[\|g^k\|^2]$$

Lemma 2:
Next, we further bound the second moment of g^k in terms of function suboptimality and $\mathcal{D}^k$.

$$E[\|g^k\|^2] \leq 4L(f(x^k) - f(x^*)) + \frac{p}{2\eta^2}\mathcal{D}^k$$

L-SVRG Convergence

Lemma 3:
We bound $E[\mathcal{D}^{k+1}]$ in terms of $\mathcal{D}^k$ and function suboptimality.

$$E[\mathcal{D}^{k+1}] \leq (1-p)\mathcal{D}^k + 8L\eta^2(f(x^k) - f(x^*))$$

Lemma 4:
Putting the above three lemmas together naturally leads to the following result involving Lyapunov function.
Let the step size $\eta \leq \frac{1}{6L}$. Then for all k≥0 the following inequality holds:

$$E[\Phi^{k+1}] \leq (1-\eta\mu)\|x^k - x^*\|^2 + (1-\frac{p}{2})\mathcal{D}^k$$

L-SVRG Convergence

Discussion of Lemma 4:

With $\eta \leq \frac{1}{6L}$ the $(1-\eta\mu)$is at least $1-\frac{\eta}{6\mu}$, thus the complexity cannot be better than $\mathcal{O}(\frac{L}{\mu}\log\frac{1}{\epsilon})$

Also L-SVRG calls the stochastic gradient oracle in expectation $\mathcal{O}(1+pn)$times in each iteration

Combining these facts we get total complexity $\mathcal{O}((\frac{1}{p}+n+\frac{L}{\mu}+\frac{Lpn}{\mu})\log\frac{1}{\epsilon})$

Note that any choice of $p \in [\min\{\frac{c}{n},\frac{c\mu}{L}\},\max\{\frac{c}{n},\frac{c\mu}{L}\}]$, where $c=\Theta(1)$, leads to the optimal complexity $\mathcal{O}((\frac{L}{\mu})\log\frac{1}{\epsilon})$

03 SARAH vs PAGE

PAGE: A Simple and Optimal Probabilistic Gradient Estimator for Nonconvex Optimization

SARAH Algorithm

StochAstic Recursive grAdient algoritHm

Parameters: the learning rate $\eta > 0$ and the inner loop size m.
Initialize: $\tilde{w}_0$
Iterate:
for $s = 1, 2, \dots$ **do**
  $w_0 = \tilde{w}_{s-1}$
  $v_0 = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(w_0)$
  $w_1 = w_0 - \eta v_0$
  Iterate:
  for $t = 1, \dots, m-1$ **do**
    Sample i_t uniformly at random from $[n]$
    $v_t = \nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_{t-1}) + v_{t-1}$
    $w_{t+1} = w_t - \eta v_t$
  end for
  Set $\tilde{w}_s = w_t$ with t chosen uniformly at random from $\{0, 1, \dots, m\}$
end for

$$O\left(n + \frac{\sqrt{n}}{\epsilon^2}\right)$$

PAGE Algorithm

ProbAbilistic Gradient Estimator

Input: initial point x^0, stepsize η, minibatch size b, $b' < b$, probability $\{p_t\} \in (0, 1]$

$g^0 = \frac{1}{b}\sum_{i\in I} \nabla f_i(x^0)$ // I denotes random minibatch samples with $|I| = b$

for $t = 0, 1, 2, \ldots$ **do**

$\quad x^{t+1} = x^t - \eta g^t$

$$g^{t+1} = \begin{cases} \frac{1}{b}\sum\limits_{i\in I} \nabla f_i(x^{t+1}) & \text{with probability } p_t \\ g^t + \frac{1}{b'}\sum\limits_{i\in I'} (\nabla f_i(x^{t+1}) - \nabla f_i(x^t)) & \text{with probability } 1 - p_t \end{cases}$$

end for

Output: $\widehat{x}_T$ chosen uniformly from $\{x^t\}_{t\in[T]}$

$$O\left(n + \frac{\sqrt{n}}{\epsilon^2}\right)$$

PAGE Convergence

Theorem
Suppose that average L-smoothness assumption holds. Choose the step size $\eta \leq \frac{1}{L(1+\sqrt{\frac{1-p}{pb'}})}$, minibatch size b=n, secondary minibatch size b'<√b, and probability p ∈ (0,1].

Then the number of iterations performed by PAGE sufficient to find ϵ-approximate solution of nonconvex finite-sum problem can be bound by

$$T = \frac{2\Delta_0 L}{\epsilon^2}\left(1+\sqrt{\frac{1-p}{pb'}}\right), \text{ where } \Delta_0 = f(x^0) - f^*$$

Moreover according to the gradient estimator of PAGE, we know that it uses pb + (1-p)b' stochastic gradients for each iteration on expectation. Thus, the number of stochastic gradient computations is

$$\#grad = b + T(pb + (1-p)b') = b + \frac{2\Delta_0 L}{\epsilon^2}\left(1+\sqrt{\frac{1-p}{pb'}}\right)(pb + (1-p)b')$$

PAGE Convergence

Corollary

Suppose that average L-smoothness assumption holds. Choose the step size $\eta \leq \frac{1}{L(1+\sqrt{b}/b')}$, minibatch size b=n, secondary minibatch size b'<√b, and probability p=b'/(b+b').

Then the number of iterations performed by PAGE sufficient to find ϵ-approximate solution of nonconvex finite-sum problem can be bound by

$$T = \frac{2\Delta_0 L}{\epsilon^2}\left(1 + \frac{\sqrt{b}}{b'}\right)$$

Moreover, the number of stochastic gradient computations is

$$\#grad \leq n + \frac{8\Delta_0 L\sqrt{n}}{\epsilon^2} = O\left(n + \frac{\sqrt{n}}{\epsilon^2}\right)$$

04 Configuration

Datasets

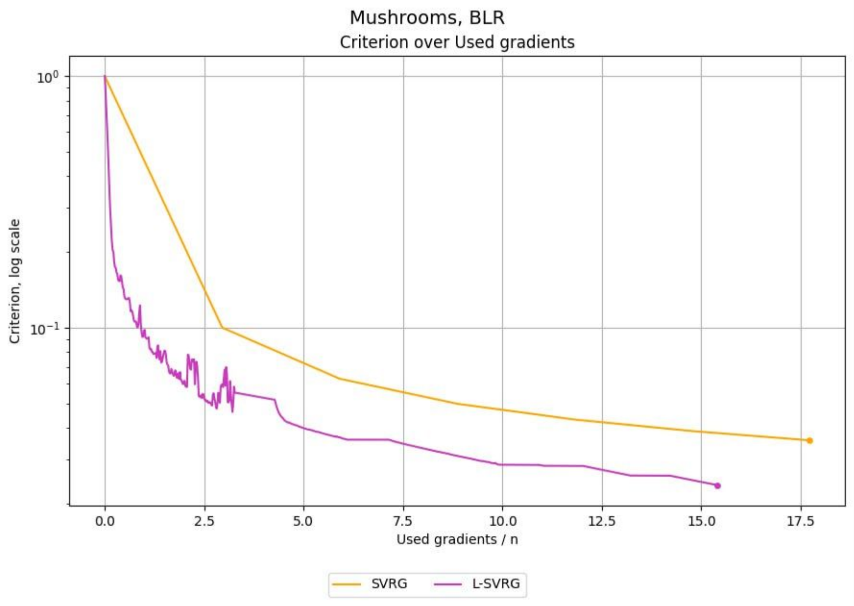
- Mushrooms
  - 8124 data rows
  - 112 features
  - 80/20 -train-test ratio
- MNIST-binary - try to predict whether picture is 0 or 1
  - 2000 data rows
  - 784 features
  - 90/10 train-test ratio
- MNIST
  - 42000 data rows
  - 784 features
  - 90/10 train-test ratio

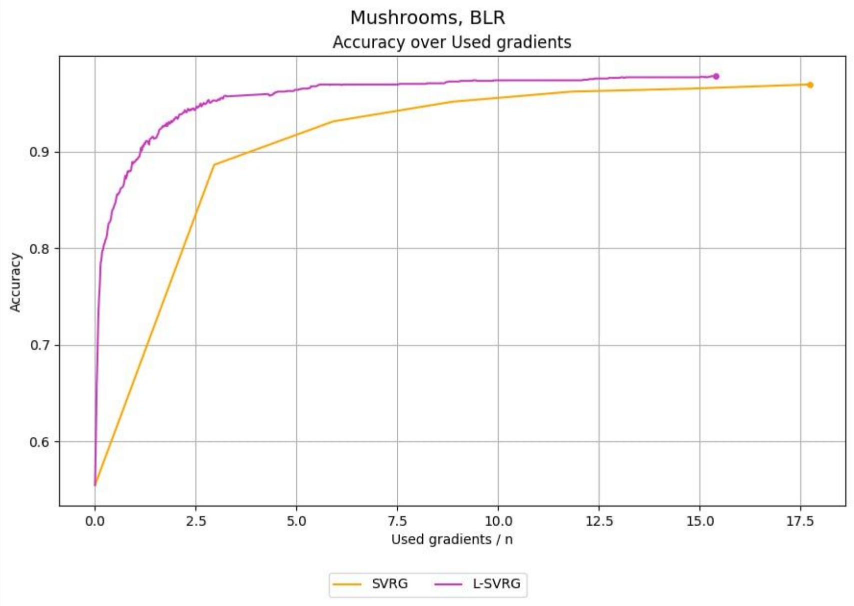
Models

- Binary Logistic Regression (BLR)
  - We always can estimate Lipschitz constant
  - Predicted class is determined by sign of output
  - Suitable only for 2 classes
  - Used in third assignment
- One layer FC
  - torch & Cuda 11.8
  - Architecture *softmax(relu(x@W+b))*
  - Cross Entropy loss
  - Suitable for an arbitrary number of classes
- Simple CNN
  - torch & Cuda 11.8
  - Has architecture *softmax(relu(relu(conv(x, (1,1,3,3)))@W+b))*
  - Cross Entropy loss
  - Suitable for an arbitrary number of classes

05 Experiments

SVRG & L-SVRG

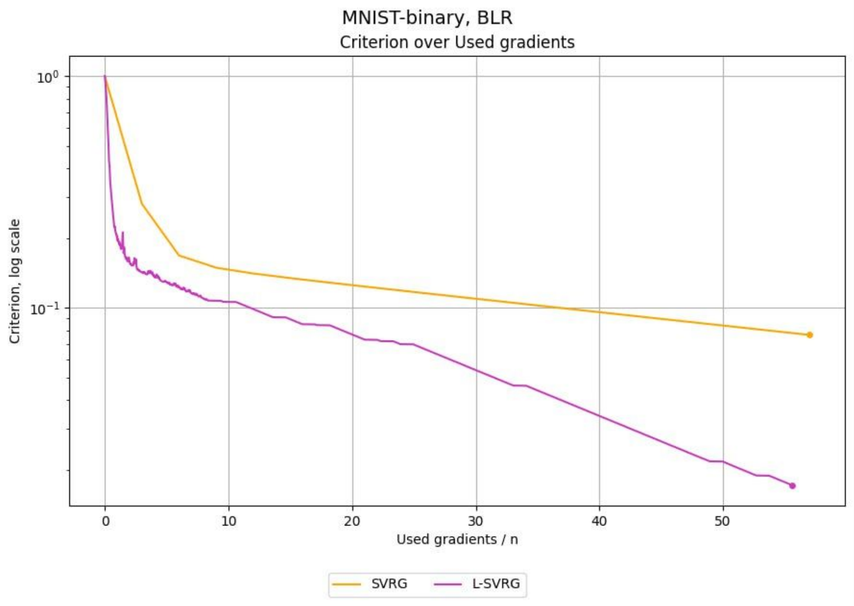


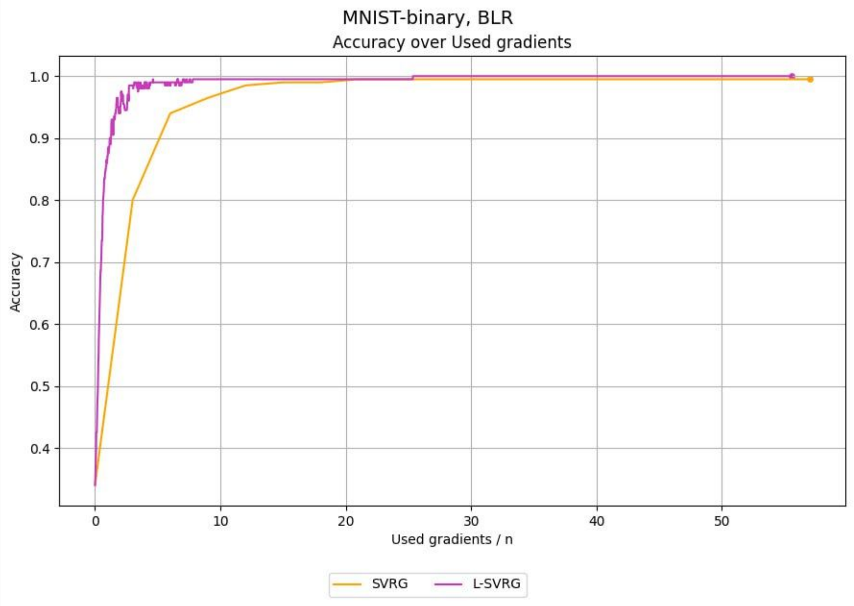


SVRG: η=1/L, n=m=100

L-SVRG: η=1/L, p=1/n=1/100

SVRG & L-SVRG

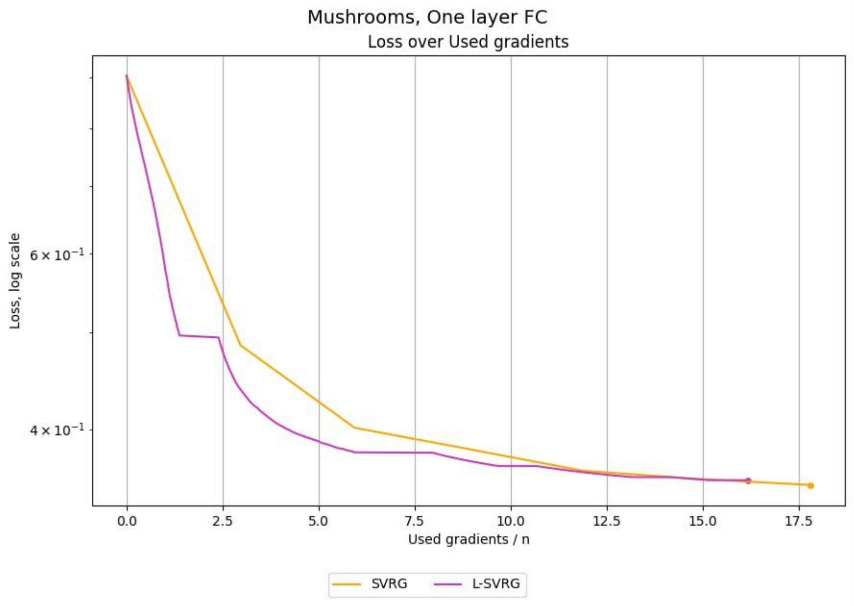


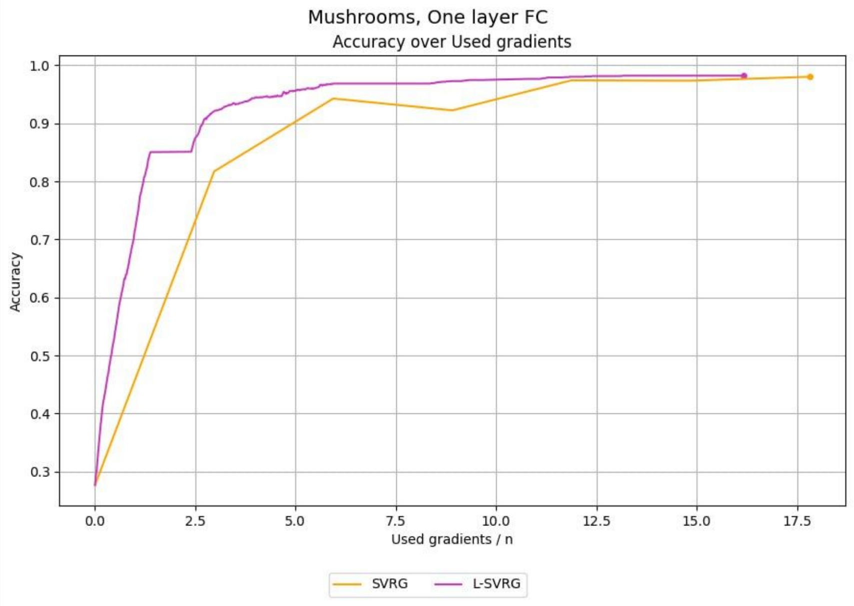


SVRG: $\eta=1/L$, $n=m=100$

L-SVRG: $\eta=1/L$, $p=1/n=1/100$

SVRG & L-SVRG

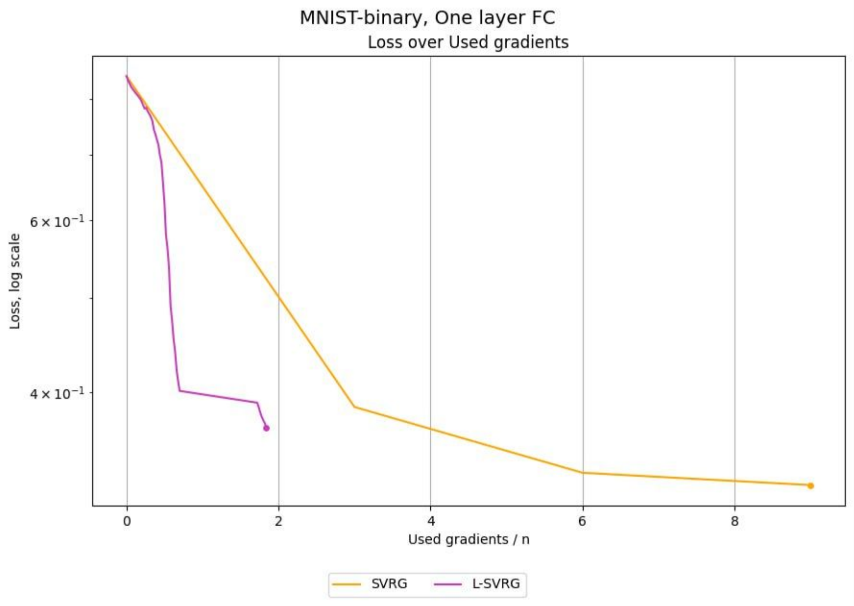


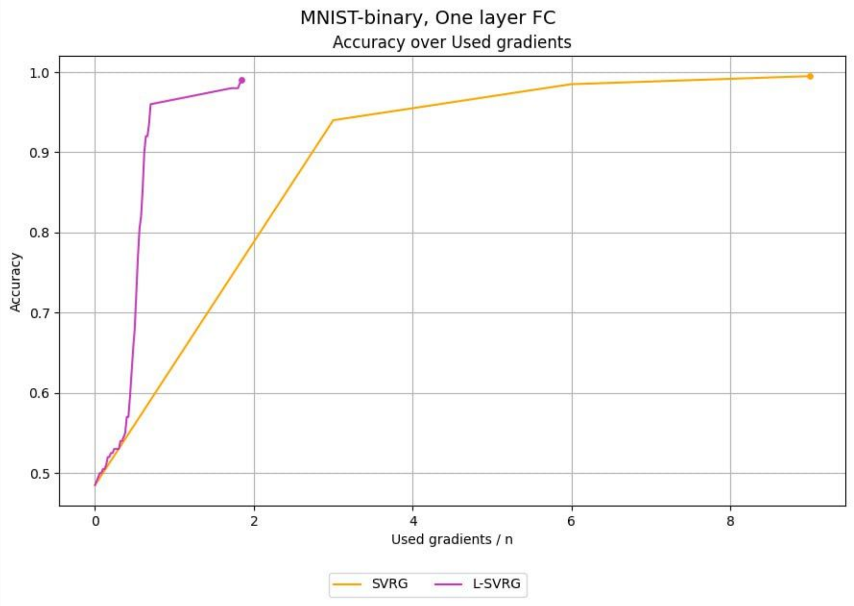


SVRG: η=0.1, n=m=100

L-SVRG: η=0.1, p=1/n=1/100

SVRG & L-SVRG

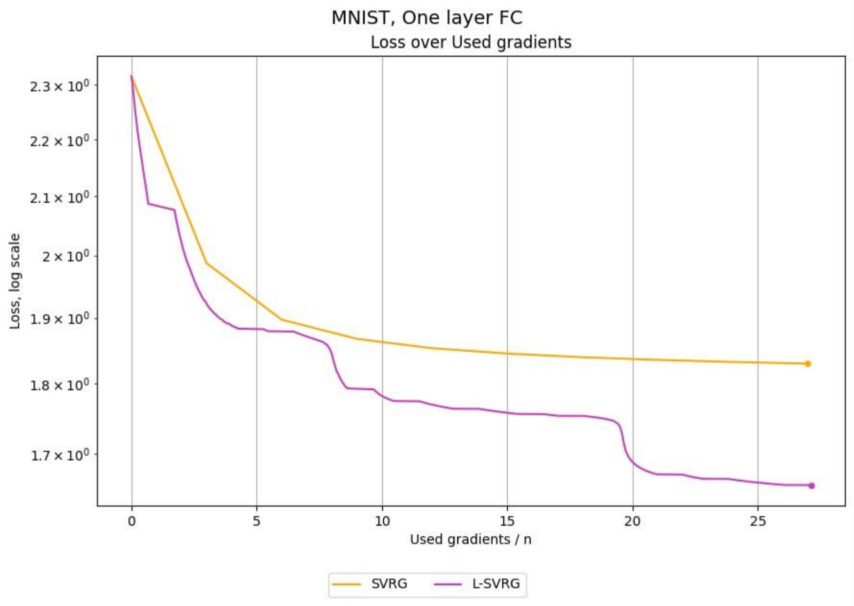




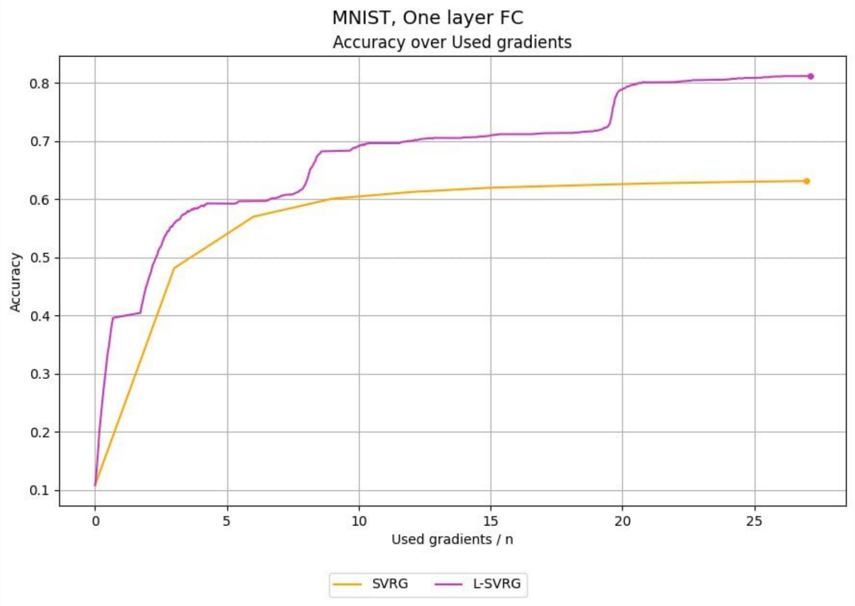
SVRG: η=0.1, n=m=100

L-SVRG: η=0.1, p=1/n=1/100

SVRG & L-SVRG

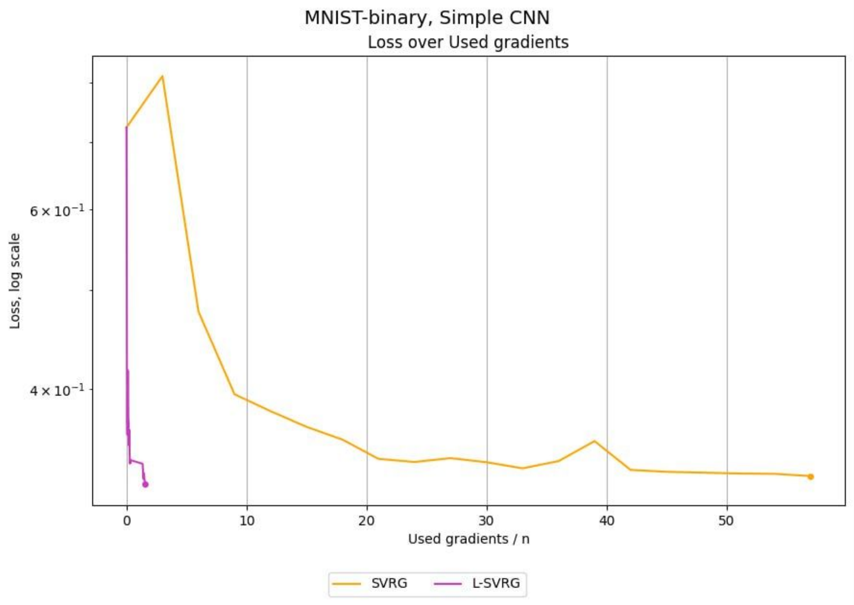


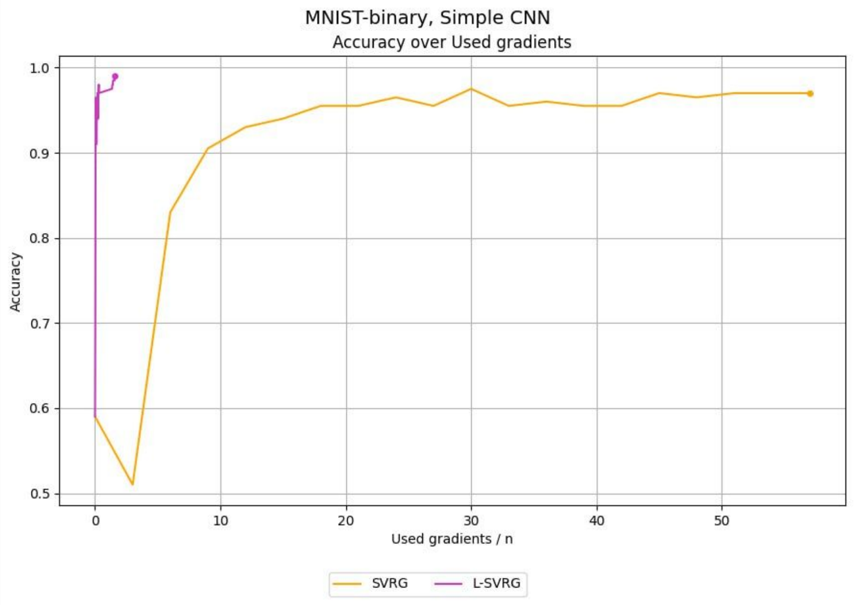
SVRG: η=2, n=m=50



L-SVRG: η=2, p=1/n=1/50

SVRG & L-SVRG

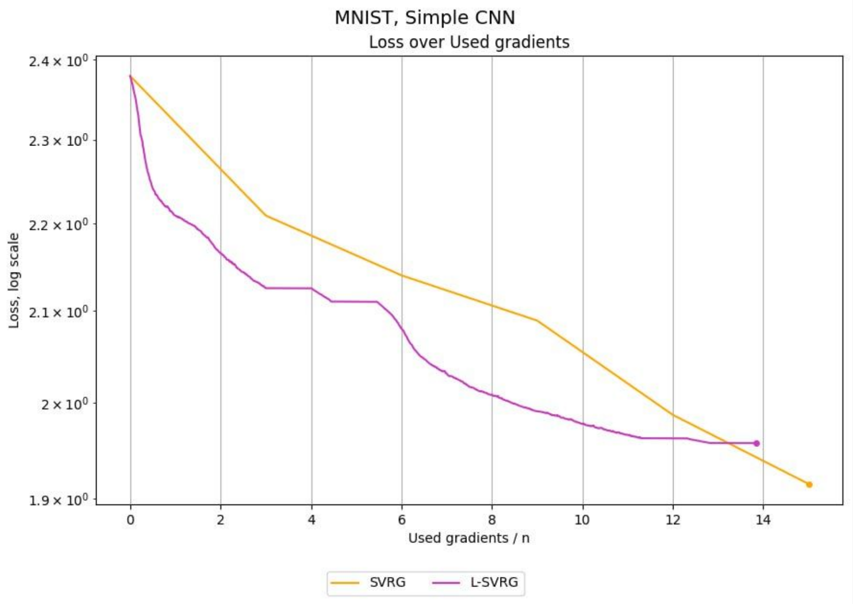


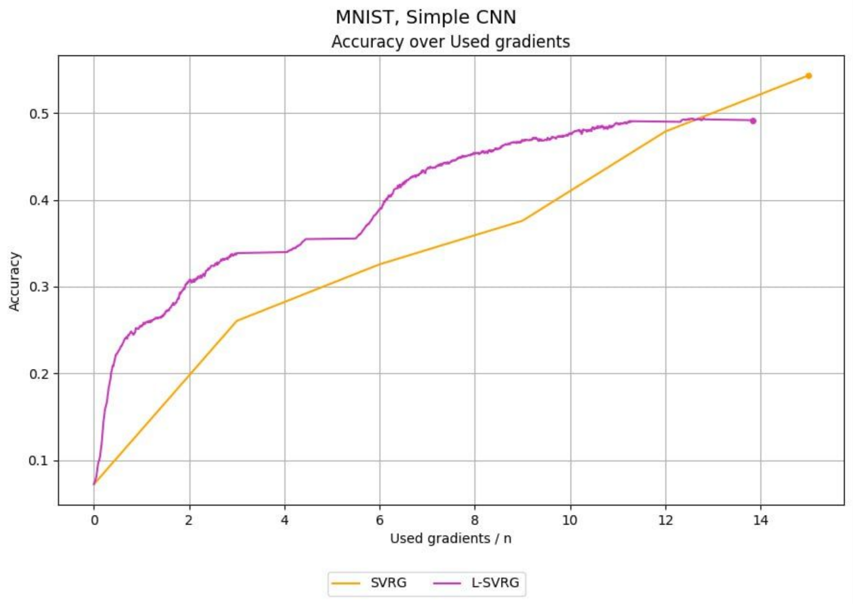


SVRG: $\eta=1$, $n=m=100$

L-SVRG: $\eta=0.1$, $p=1/n=1/100$

SVRG & L-SVRG

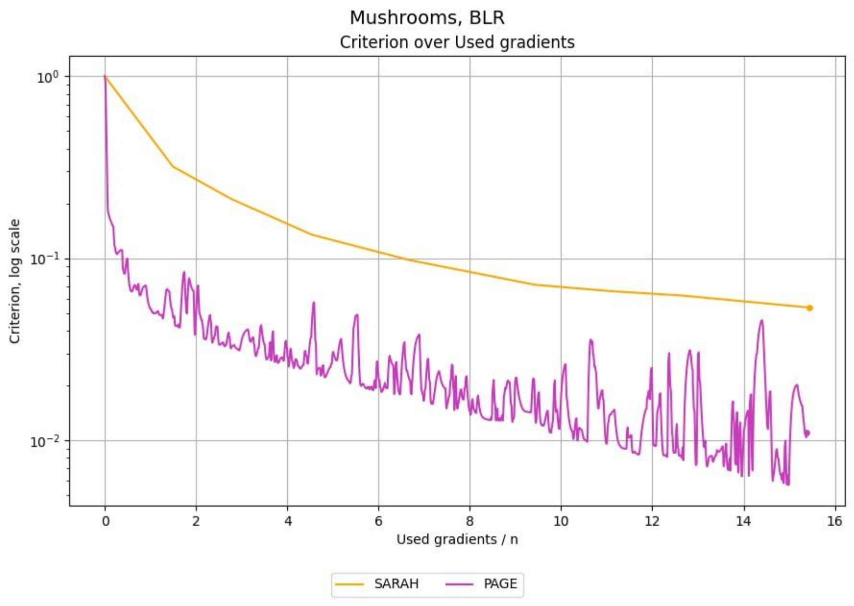


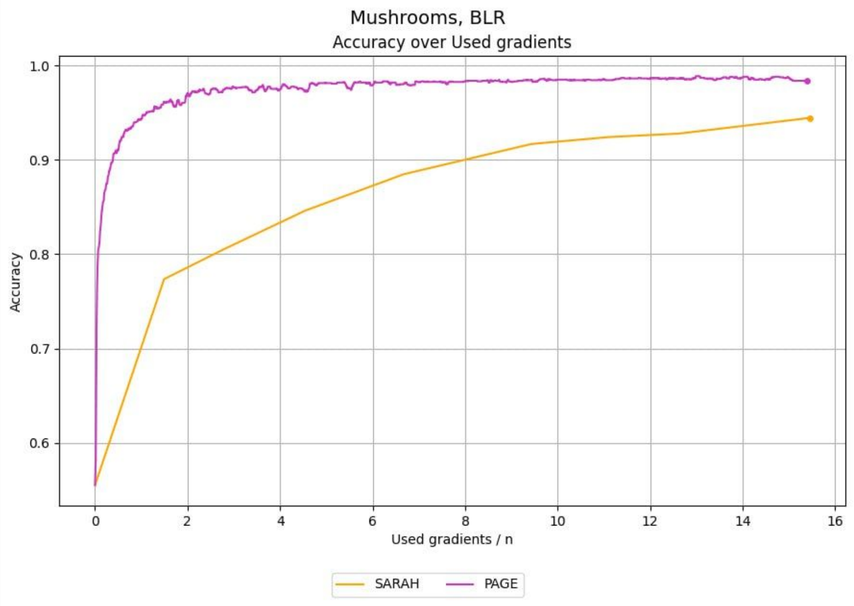


SVRG: η=0.1, n=m=200

L-SVRG: η=0.1, p=1/n=1/200

SARAH & PAGE

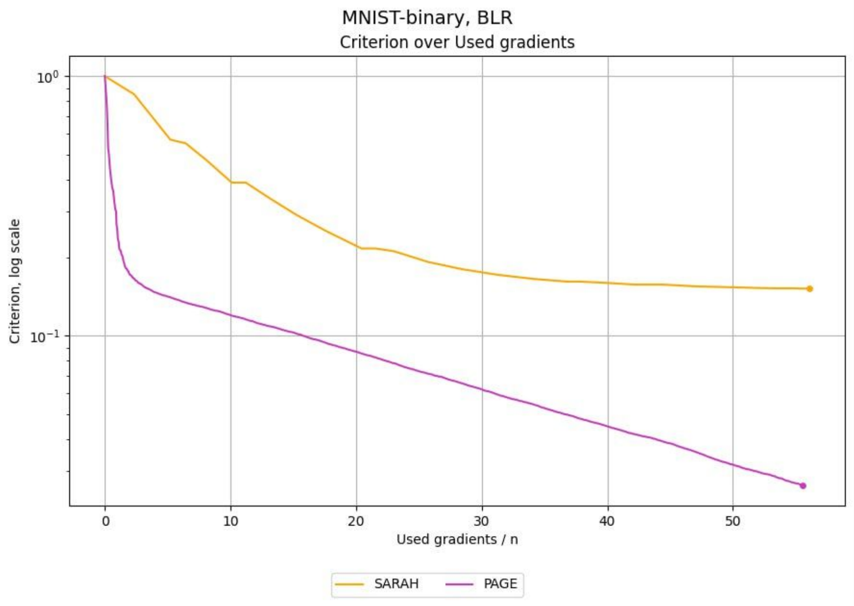


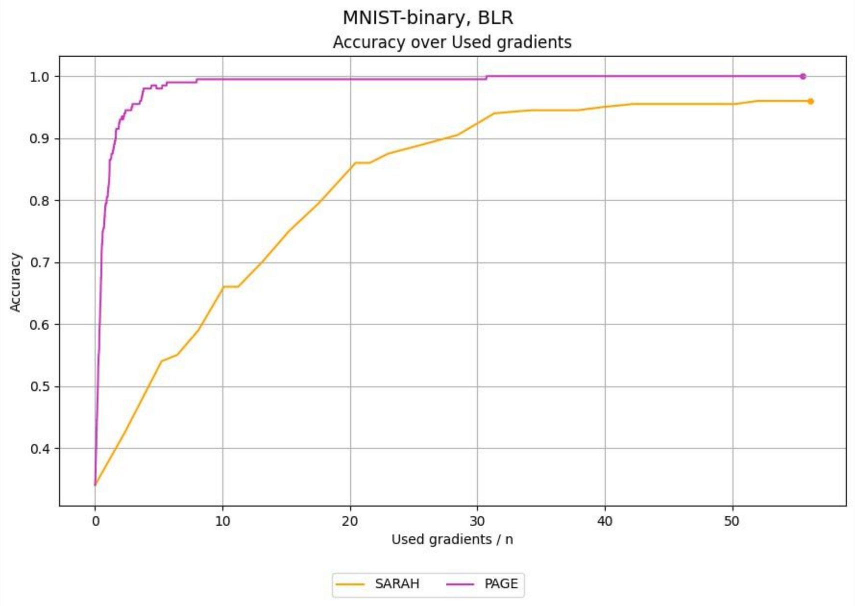


SARAH: η=1/(2L), b=10, m=data_size/b

PAGE: η=1/(2L), b=100, b'=10

SARAH & PAGE

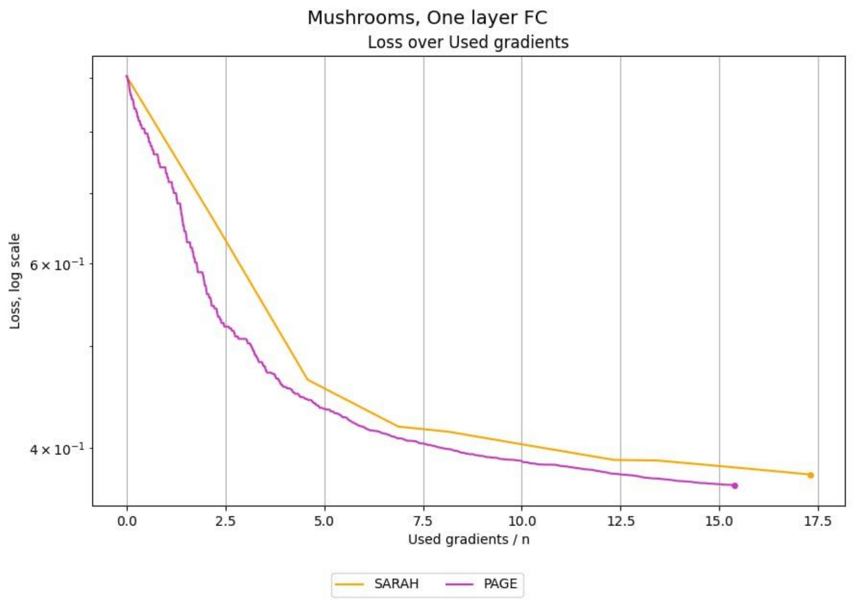


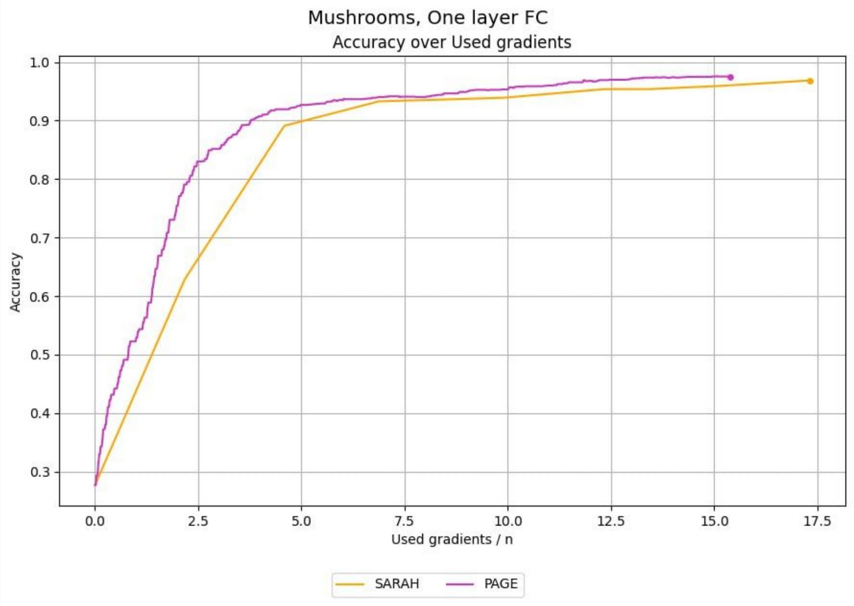


SARAH: η=1/(2L), b=10, m=data_size/b

PAGE: η=1/(2L), b=100, b'=10

SARAH & PAGE

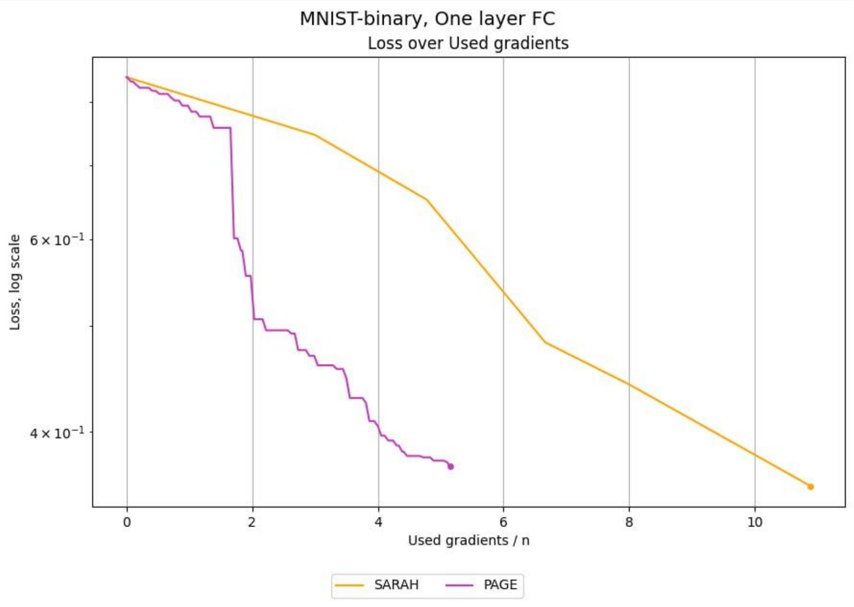


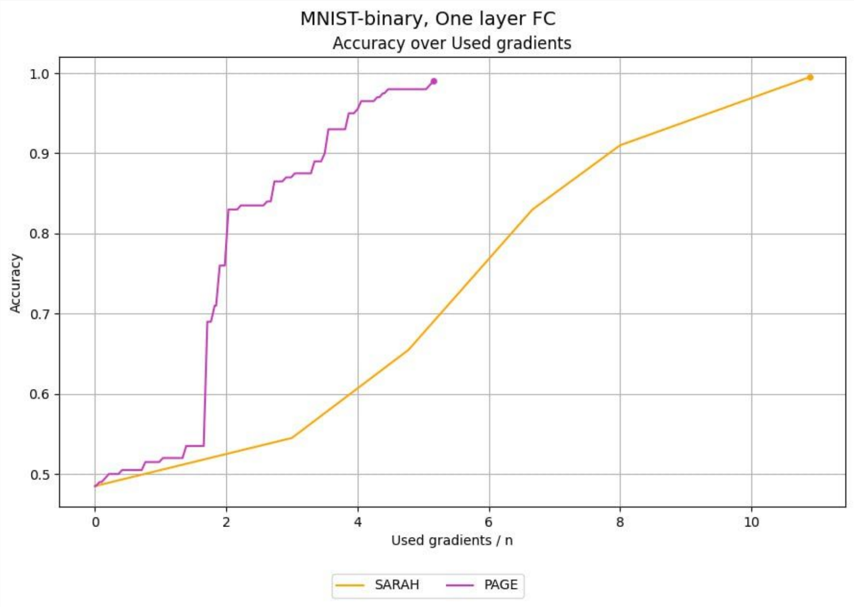


SARAH: η=0.1, b=10, m=data_size/b

PAGE: η=0.1, b=100, b'=10

SARAH & PAGE

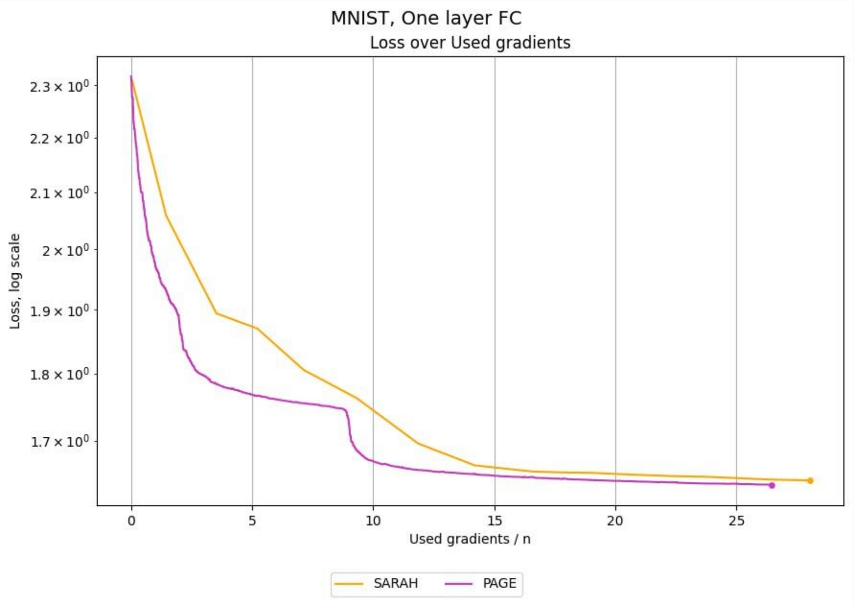


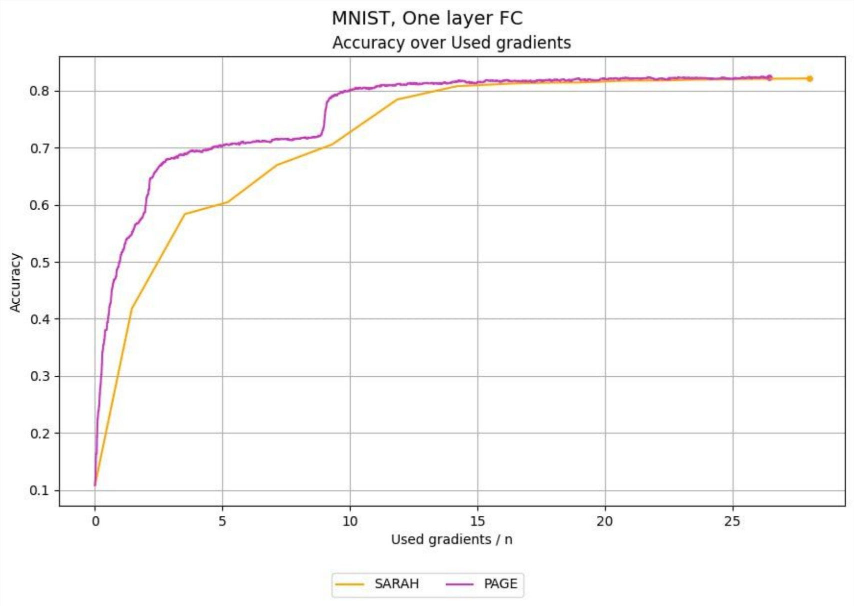


SARAH: η=0.1, b=10, m=data_size/b

PAGE: η=0.1, b=100, b'=10

SARAH & PAGE

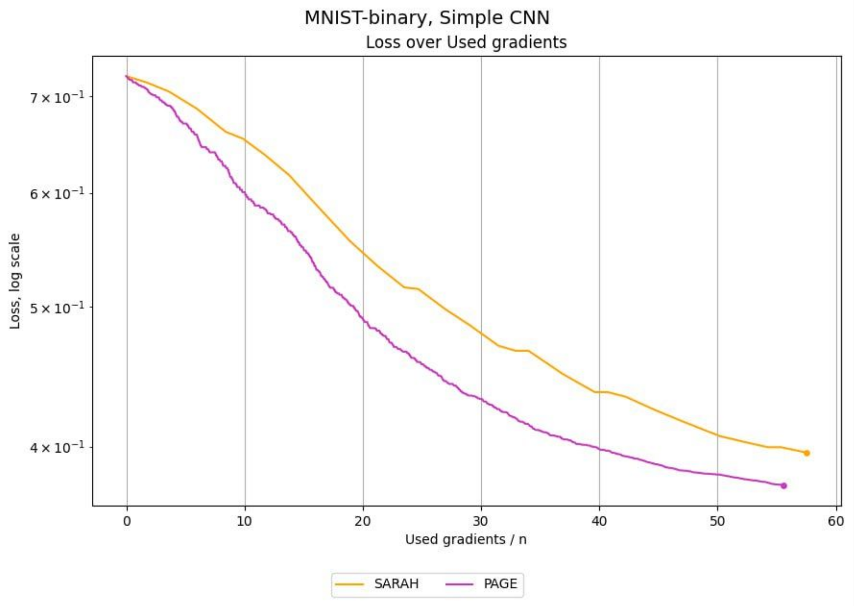


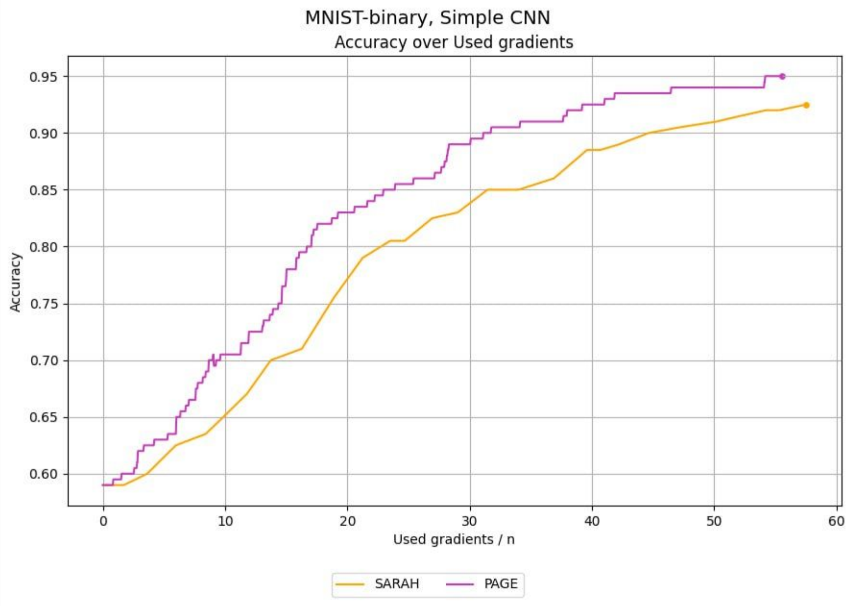


SARAH: η=1, b=50, m=data_size/b

PAGE: η=0.1, b=200, b'=50

SARAH & PAGE

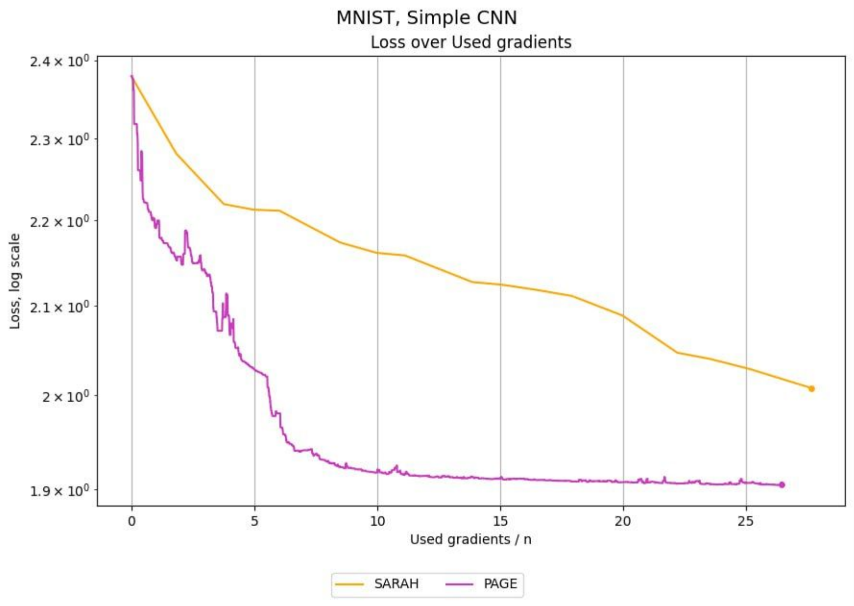


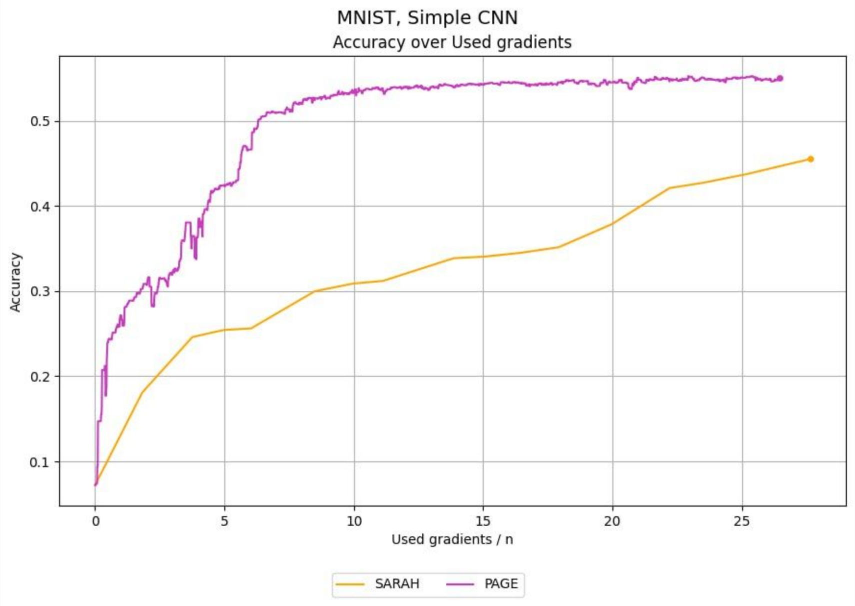


SARAH: η=0.001, b=9, m=data_size/b

PAGE: η=0.001, b=81, b'=9

SARAH & PAGE





SARAH: η=0.1, b=50, m=data_size/b

PAGE: η=0.5, b=400, b'=100

References

- Kovalev, D. (2020, January 28). Don't Jump Through Hoops and Remove Those Loops: SVRG and Katyusha are Better Without the Outer Loop. PMLR. https://proceedings.mlr.press/v117/kovalev20a.html
- Li, Z. (2021, July 1). PAGE: A simple and optimal probabilistic gradient estimator for nonconvex optimization. PMLR. https://proceedings.mlr.press/v139/li21a.html
- Gorbunov, E. (2019, May 27). A unified theory of SGD: variance reduction, sampling, quantization and coordinate descent. arXiv.org. https://arxiv.org/abs/1905.11261
- Nguyen, L. M. (2017, March 1). SARAH: A Novel Method for Machine Learning Problems using Stochastic Recursive Gradient. arXiv.org. https://arxiv.org/abs/1703.00102

Thanks for your attention!

Do not forget to visit our GitHub!
github.com/dsomni/omml-project-f23