LEARNING FROM TIME-DEPENDENT STREAMING DATA WITH ONLINE STOCHASTIC ALGORITHMS

Nicklas Werge

Ph.D. Defence: Thursday 29th September, 2022

Supervised by Antoine Godichon-Baggioni and Olivier Wintenberger

SORBONNE UNIVERSITÉ

LPSM LABORATOIRE DE PROBABILITÉS STATISTIQUE & MODÉLISATION

Learning schemes

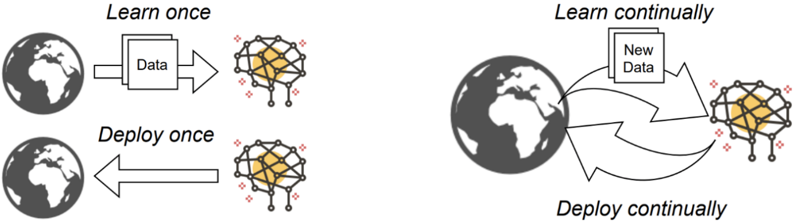


Figure 1: Large- and small-scale learning vs. learning from streaming data

Examples of streaming data. Internet traffic (e.g., tweets, search engines, advertising), self-driving cars, financial investments, electricity management from solar or wind, weather data and other sensor data.

Why use SG-based methods for streaming data?

Common optimization problem,

$$\min_{\theta \in \mathbb{R}^d} \left\{ L_n(\theta) = \frac{1}{n} \sum_{t=1}^{n} l_t(\theta) \right\}, \quad \text{(empirical risk)} \tag{1}$$

where (l_t) is a sequence of random differentiable functions from $\mathbb{R}^d$ to $\mathbb{R}$.

Why use SG-based methods for streaming data?

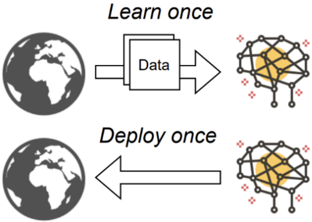
Common optimization problem,

$$\min_{\theta \in \mathbb{R}^d} \left\{ L_n(\theta) = \frac{1}{n} \sum_{t=1}^{n} l_t(\theta) \right\}, \quad \text{(empirical risk)} \tag{1}$$

where (l_t) is a sequence of random differentiable functions from $\mathbb{R}^d$ to $\mathbb{R}$.

What is the computational cost of **solving** (1)?

- Batch gradient descent costs $\mathcal{O}(ndk)$ with k iterations.
- Stochastic Gradient (SG) descent costs $\mathcal{O}(nd)$.[a]



[a] BB07.

Why use SG-based methods for streaming data?

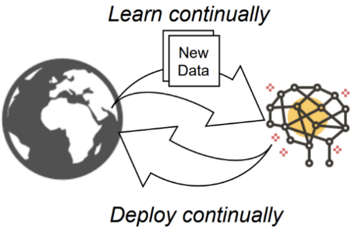
Common optimization problem,

$$\min_{\theta \in \mathbb{R}^d} \left\{ L_n(\theta) = \frac{1}{n} \sum_{t=1}^{n} l_t(\theta) \right\}, \quad \text{(empirical risk)} \tag{1}$$

where (l_t) is a sequence of random differentiable functions from $\mathbb{R}^d$ to $\mathbb{R}$.

What is the computational cost of **updating** (1)?

- Batch gradient descent costs $\mathcal{O}(ndk)$ with k iterations.
- Stochastic Gradient (SG) descent costs $\mathcal{O}(d)$.



Why use SG-based methods for streaming data?

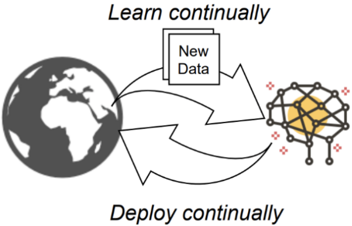
Common optimization problem,

$$\min_{\theta \in \mathbb{R}^d} \left\{ L_n(\theta) = \frac{1}{n} \sum_{t=1}^{n} l_t(\theta) \right\}, \quad \text{(empirical risk)} \tag{1}$$

where (l_t) is a sequence of random differentiable functions from $\mathbb{R}^d$ to $\mathbb{R}$.

What is the computational cost of **updating** (1)?

- Batch gradient descent costs $\mathcal{O}(ndk)$ with k iterations.
- Stochastic Gradient (SG) descent costs $\mathcal{O}(d)$.



Takeaway. For streaming with large n (and d) $\Rightarrow$ SG-based methods.

Examples of applications for (1)

Let $X_t \in \mathcal{X}$ (inputs) and $Y_t \in \mathcal{Y}$ (outputs/labels),

$$l_t(\theta) = l(Y_t, h_\theta(X_t)) + \lambda\Omega(\theta), \quad \lambda \geq 0, \tag{2}$$

where $h_\theta(X_t) : \mathcal{X} \to \mathbb{R}$ (predictor), $l : \mathcal{Y} \times \mathbb{R} \to \mathbb{R}$ (loss) and $\Omega(\theta) : \mathbb{R}^d \to \mathbb{R}$ (regularizer).

Examples of applications for (1)

Let $X_t \in \mathcal{X}$ (inputs) and $Y_t \in \mathcal{Y}$ (outputs/labels),

$$l_t(\theta) = l(Y_t, h_\theta(X_t)) + \lambda\Omega(\theta), \quad \lambda \geq 0, \tag{2}$$

where $h_\theta(X_t) : \mathcal{X} \to \mathbb{R}$ (predictor), $l : \mathcal{Y} \times \mathbb{R} \to \mathbb{R}$ (loss) and $\Omega(\theta) : \mathbb{R}^d \to \mathbb{R}$ (regularizer).

Typical examples:

- **Regression**: $\mathcal{Y} = \mathbb{R}$, $h_\theta(X_t) = \langle \theta, X_t \rangle$, $l = \frac{1}{2}(Y_t - h_\theta(X_t))^2$, $\Omega(\theta) = \|\theta\|_1$ or $\Omega(\theta) = \|\theta\|_2^2$.
- **Classification**: $\mathcal{Y} = \{-1, 1\}$, $h_\theta(X_t) = \langle \theta, X_t \rangle$, $l = \phi(Y_t h_\theta(X_t))$, where ϕ, e.g., is $\max\{0, 1-u\}$ or $\log(1+e^{-u})$.

Examples of applications for (1)

Let $X_t \in \mathcal{X}$ (inputs) and $Y_t \in \mathcal{Y}$ (outputs/labels),

$$l_t(\theta) = l(Y_t, h_\theta(X_t)) + \lambda\Omega(\theta), \quad \lambda \geq 0, \tag{2}$$

where $h_\theta(X_t) : \mathcal{X} \to \mathbb{R}$ (predictor), $l : \mathcal{Y} \times \mathbb{R} \to \mathbb{R}$ (loss) and $\Omega(\theta) : \mathbb{R}^d \to \mathbb{R}$ (regularizer).

Other examples:

- Geometric median (our example in this talk).
- Quasi-maximum likelihood for non-linear time series models.
- Neural networks for deep learning.

Examples of applications for (1)

SORBONNE UNIVERSITÉ

Let $X_t \in \mathcal{X}$ (inputs) and $Y_t \in \mathcal{Y}$ (outputs/labels),

$$l_t(\theta) = l(Y_t, h_\theta(X_t)) + \lambda\Omega(\theta), \quad \lambda \geq 0, \tag{2}$$

where $h_\theta(X_t) : \mathcal{X} \to \mathbb{R}$ (predictor), $l : \mathcal{Y} \times \mathbb{R} \to \mathbb{R}$ (loss) and $\Omega(\theta) : \mathbb{R}^d \to \mathbb{R}$ (regularizer).

Other examples:

- Geometric median (our example in this talk).
- Quasi-maximum likelihood for non-linear time series models.
- Neural networks for deep learning.

Takeaway. There are many examples for applications, e.g., see Teo et al. [Teo+07], Hastie et al. [Has+09], Kushner and Yin [KY03], and Nesterov et al. [Nes+18] for examples of losses and their derivatives.

Research aims and objectives

Main goals. The central theme of this thesis is to learn from time-dependent streaming data, where traditional optimization techniques are unsustainable due to their high computational cost.

Research aims and objectives

Main goals. The central theme of this thesis is to learn from time-dependent streaming data, where traditional optimization techniques are unsustainable due to their high computational cost.

We want to explore the robustness and convergence guarantees of SG-based methods under various settings. In short, the main goals are

1. to allow learning algorithms to handle streaming data and
2. to improve learning by adapting streaming learning to the difficulty of the problem; the level of dependence, noise, and convexity.

Research aims and objectives

Main goals. The central theme of this thesis is to learn from time-dependent streaming data, where traditional optimization techniques are unsustainable due to their high computational cost.

Summary of Ph.D.:

- Chapter 2 [GBWW21]: Antoine Godichon-Baggioni, Nicklas Werge, and Olivier Wintenberger. "Non-Asymptotic Analysis of Stochastic Approximation Algorithms for Streaming Data". In: *arXiv preprint arXiv:2109.07117* (2021).
- Chapter 3 [GBWW22]: Antoine Godichon-Baggioni, Nicklas Werge, and Olivier Wintenberger. "Learning from time-dependent streaming data with online stochastic algorithms". In: *arXiv preprint arXiv:2205.12549* (2022).
- Chapter 4 [WW22]: Nicklas Werge and Olivier Wintenberger. "AdaVol: An adaptive recursive volatility prediction method". In: *Econometrics and Statistics* 23 (2022), pp. 19–35.

Appendix [Wer21]: Nicklas Werge. "Predicting risk-adjusted returns using an asset independent regime-switching model". In: *Expert Systems with Applications* 184 (2021), p. 115576. ISSN: 0957-4174.

Research aims and objectives

Main goals. The central theme of this thesis is to learn from time-dependent streaming data, where traditional optimization techniques are unsustainable due to their high computational cost.

For this talk:

- Chapter 2 [GBWW21]: Learning from streaming data.
- Chapter 3 [GBWW22]: Learning from time-dependent streaming data.

Stochastic optimization

Stochastic Optimization (SO) problem

Minimize objectives $L : \mathbb{R}^d \to \mathbb{R}$, defined by

$$\theta^* := \arg\min_{\theta \in \mathbb{R}^d} \{L(\theta) := \mathbb{E}[l_t(\theta)]\}, \tag{3}$$

with $l_t : \mathbb{R}^d \to \mathbb{R}$ some random differentiable functions.

Stochastic optimization

Stochastic Optimization (SO) problem

Minimize objectives $L : \mathbb{R}^d \to \mathbb{R}$, defined by

$$\theta^* := \arg\min_{\theta \in \mathbb{R}^d} \{L(\theta) := \mathbb{E}[l_t(\theta)]\}, \tag{3}$$

with $l_t : \mathbb{R}^d \to \mathbb{R}$ some random differentiable functions.

How do we find the unique global minimizer θ^* of L in (3)?[1]

- L is minimized without evaluating it directly.
- Instead, we **only** use noisy gradients of $l_t(\theta)$ as estimates.

[1] Robbins and Monro [RM51]

Stochastic optimization

Stochastic Optimization (SO) problem

Minimize objectives $L : \mathbb{R}^d \to \mathbb{R}$, defined by

$$\theta^* := \arg\min_{\theta \in \mathbb{R}^d} \{L(\theta) := \mathbb{E}[l_t(\theta)]\}, \tag{3}$$

with $l_t : \mathbb{R}^d \to \mathbb{R}$ some random differentiable functions.

How to extend the SO problem to a streaming setting

At each time $t \in \mathbb{N}$, a **block** of $n_t \in \mathbb{N}$ random differentiable functions arrive,

$$l_t := (l_{t,1}, \ldots, l_{t,n_t}).$$

Some examples for streaming applications

Following (1) and (2), for some parameterization $\{h_\theta\}_{\theta \in \mathbb{R}^d}$, this requires to minimize

$$L_{N_t}(\theta) = \frac{1}{N_t} \sum_{i=1}^{t} l_i(\theta), \quad \text{(empirical risk)}$$

where $N_t := \sum_{i=1}^{t} n_i$ denotes the accumulated sum of observations; here

$$l_t(\theta) = \sum_{j=1}^{n_t} l(Y_{t,j}, h_\theta(X_{t,j})) + \lambda \Omega(\theta),$$

where $X_t := (X_{t,1}, \dots, X_{t,n_t})$ and $Y_t := (Y_{t,1}, \dots, Y_{t,n_t})$ are the blocks of n_t observations that arrive at each t (a.k.a. streaming-batches).

How to we solve the SO problem in a streaming setting?

Stochastic Streaming Gradient (SSG)

The SSG is defined by the following recursion,

$$\theta_t = \theta_{t-1} - \frac{\gamma_t}{n_t} \sum_{i=1}^{n_t} \nabla_\theta l_{t,i} \left(\theta_{t-1} \right), \quad \theta_0 \in \mathbb{R}^d, \tag{4}$$

with learning rate (γ_t) satisfying $\sum_{i=1}^{\infty} \gamma_i = \infty$ and $\sum_{i=1}^{\infty} \gamma_i^2 < \infty$.

- $n_t = 1 \Rightarrow$ SG descent (SGD) [RM51].
- n_t constant $\Rightarrow$ online mini-batches.
- n_t time-varying $\Rightarrow$ streaming-batches.

Acceleration by averaging

Averaged SSG (ASSG)

The ASSG is derived for all $t \in \mathbb{N}$ by the recursion,

$$\bar{\theta}_t = \frac{1}{N_t} \sum_{i=0}^{t-1} n_{i+1} \theta_i, \ \bar{\theta}_0 = 0, \text{ with } (\theta_t) \text{ following (4)}, \tag{5}$$

where $N_t = \sum_{i=1}^{t} n_i$ denotes the accumulated sum of observations.

- $n_t = 1 \Rightarrow$ Polyak-Ruppert averaging SGD (ASGD) [PJ92; Rup88].
- n_t constant $\Rightarrow$ online Polyak-Ruppert averaged mini-batches.
- n_t time-varying $\Rightarrow$ Polyak-Ruppert averaged streaming-batches.

Acceleration by averaging

Averaged SSG (ASSG)

The ASSG is derived for all $t \in \mathbb{N}$ by the recursion,

$$\bar{\theta}_t = \frac{1}{N_t} \sum_{i=0}^{t-1} n_{i+1} \theta_i, \ \bar{\theta}_0 = 0, \text{ with } (\theta_t) \text{ following (4)}, \tag{5}$$

where $N_t = \sum_{i=1}^{t} n_i$ denotes the accumulated sum of observations.

Stochastic streaming algorithms combines SG-based methods'

1. applicability,
2. computational benefits,
3. variance-reducing properties through mini-batching, and
4. the accelerated convergence from Polyak-Ruppert averaging.

Overview of stochastic streaming algorithms (pseudo code)

Algorithm 1: Stochastic streaming algorithms (SSG/ASSG)

Inputs : $\theta_0 \in \mathbb{R}^d$, average: **True** or **False**
Outputs: θ_t, $\bar{\theta}_t$ (resulting estimates)
$\bar{\theta}_0 = 0$
for *each* $t \geq 1$, *a block of* n_t *data arrives* **do**
  $\theta_t \leftarrow \theta_{t-1} - \frac{\gamma_t}{n_t} \sum_{i=1}^{n_t} \nabla_\theta l_{t,i} \left(\theta_{t-1}\right)$
  if *average* **then**
    $\bar{\theta}_t \leftarrow (N_{t-1}/N_t)\bar{\theta}_{t-1} + (n_t/N_t)\theta_{t-1}$ `/* average */`

Takeaway. Each update is cheap with a computational costs of $\mathcal{O}(n_t d)$. A batch gradient costs $\mathcal{O}(N_t dk)$ after k iterations.

Overview of stochastic streaming algorithms (pseudo code)

SORBONNE UNIVERSITÉ

Algorithm 2: Stochastic streaming algorithms (SSG/ASSG)

Inputs : $\theta_0 \in \mathbb{R}^d$, average: **True** or **False**
Outputs: θ_t, $\bar{\theta}_t$ (resulting estimates)
$\bar{\theta}_0 = 0$
for *each* $t \geq 1$, *a block of* n_t *data arrives* **do**
  $\theta_t \leftarrow \theta_{t-1} - \frac{\gamma_t}{n_t} \sum_{i=1}^{n_t} \nabla_\theta l_{t,i}(\theta_{t-1})$
  if *average* **then**
    $\bar{\theta}_t \leftarrow (N_{t-1}/N_t)\bar{\theta}_{t-1} + (n_t/N_t)\theta_{t-1}$ `/* average */`

Takeaway. Each update is cheap with a computational costs of $\mathcal{O}(n_t d)$. A batch gradient costs $\mathcal{O}(N_t dk)$ after k iterations.

Projected stochastic streaming algorithms $\rightarrow$ [GBWW21; GBWW22].

What is our goals? How do we evaluate?

- Our **objective** is to provide non-asymptotic bounds of

$$\delta_t = \mathbb{E}[\|\theta_t - \theta^*\|^2] \quad \text{and} \quad \bar{\delta}_t = \mathbb{E}[\|\bar{\theta}_t - \theta^*\|^2].$$

What is our goals? How do we evaluate?

- Our **objective** is to provide non-asymptotic bounds of

$$\delta_t = \mathbb{E}[\|\theta_t - \theta^*\|^2] \quad \text{and} \quad \bar{\delta}_t = \mathbb{E}[\|\bar{\theta}_t - \theta^*\|^2].$$

- Learning rates (γ_t) on the form:

$$\gamma_t = C_\gamma n_t^\beta t^{-\alpha},$$

with $C_\gamma > 0$, $\beta \in [0, 1)$ and α chosen accordingly to the streaming-batches.

What is our goals? How do we evaluate?

- Our **objective** is to provide non-asymptotic bounds of

$$\delta_t = \mathbb{E}[\|\theta_t - \theta^*\|^2] \quad \text{and} \quad \bar{\delta}_t = \mathbb{E}[\|\bar{\theta}_t - \theta^*\|^2].$$

- Learning rates (γ_t) and streaming-batches (n_t) on the form:

$$\gamma_t = C_\gamma n_t^\beta t^{-\alpha} \quad \text{and} \quad n_t = C_\rho t^\rho,$$

  with $C_\gamma > 0$, $C_\rho \in \mathbb{N}$, $\beta, \rho \in [0, 1)$ and α chosen accordingly to the streaming-batches.
  - **Classical SG-based methods**: $n_t = 1$, i.e., $\{C_\rho = 1, \rho = 0\}$.
  - **Constant** streaming-batches (online mini-batch): $n_t = C_\rho$, i.e., $\{C_\rho \in \mathbb{N}, \rho = 0\}$, with **streaming-batch size** C_ρ.
  - **Time-varying** streaming-batches: $n_t = C_\rho t^\rho$ with $C_\rho \in \mathbb{N}$ and **streaming rate** $\rho \in [0, 1)$.[1]

[1] Note that [GBWW21] considered $\rho \in (-1, 1)$.

What is our goals? How do we evaluate?

- Our **objective** is to provide non-asymptotic bounds of

$$\delta_t = \mathbb{E}[\|\theta_t - \theta^*\|^2] \quad \text{and} \quad \bar{\delta}_t = \mathbb{E}[\|\bar{\theta}_t - \theta^*\|^2].$$

- Learning rates (γ_t) and streaming-batches (n_t) on the form:

$$\gamma_t = C_\gamma n_t^\beta t^{-\alpha} \quad \text{and} \quad n_t = C_\rho t^\rho,$$

with $C_\gamma > 0$, $C_\rho \in \mathbb{N}$, $\beta, \rho \in [0, 1)$ and α chosen accordingly to the streaming-batches.

What has been done until now?

- Classical setting with $n_t = 1$ (i.e., $\{C_\rho = 1, \rho = 0\}$) using **independent unbiased** gradients [MB11].
- Streaming setting using **independent unbiased** gradients [GBWW21].
- Streaming setting using **dependent biased** gradients [GBWW22].

Convexity and smoothness of the objectives

Assumption (Convexity and smoothness of the objectives)

Assume the following about the objectives $L : \mathbb{R}^d \to \mathbb{R}$*:*

- L *has unique global minimizer* $\theta^* \in \mathbb{R}^d$ *such that* $\nabla_\theta L(\theta^*) = 0$.

Convexity and smoothness of the objectives

Assumption (Convexity and smoothness of the objectives)

Assume the following about the objectives $L : \mathbb{R}^d \to \mathbb{R}$:

- *L has unique global minimizer $\theta^* \in \mathbb{R}^d$ such that $\nabla_\theta L(\theta^*) = 0$.*
- *L is μ-quasi-strongly convex;*[1]

$$\exists \mu > 0, \forall \theta \in \mathbb{R}^d, L(\theta^*) \geq L(\theta) + \langle \nabla_\theta L(\theta), \theta^* - \theta \rangle + \frac{\mu}{2} \|\theta^* - \theta\|^2.$$

[1] E.g., see Bach and Moulines [BM13] and Gadat and Panloup [GP17] for non-convex objectives.

Convexity and smoothness of the objectives

Assumption (Convexity and smoothness of the objectives)

Assume the following about the objectives $L : \mathbb{R}^d \to \mathbb{R}$:

- *L has unique global minimizer $\theta^* \in \mathbb{R}^d$ such that $\nabla_\theta L(\theta^*) = 0$.*
- *L is μ-quasi-strongly convex;*

$$\exists \mu > 0, \forall \theta \in \mathbb{R}^d, L(\theta^*) \geq L(\theta) + \langle \nabla_\theta L(\theta), \theta^* - \theta \rangle + \frac{\mu}{2}\|\theta^* - \theta\|^2.$$

- *L has C_∇-Lipschitz continuous gradients;*

$$\exists C_\nabla > 0, \forall \theta, \theta' \in \mathbb{R}^d, \|\nabla_\theta L(\theta) - \nabla_\theta L(\theta')\| \leq C_\nabla \|\theta - \theta'\|. \tag{6}$$

- *The Hessian of L is C'_∇-Lipschitz-continuous;*

$$\exists C'_\nabla > 0, \forall \theta, \theta' \in \mathbb{R}^d, \|\nabla^2_\theta L(\theta) - \nabla^2_\theta L(\theta')\| \leq C'_\nabla \|\theta - \theta'\|. \tag{7}$$

Observe that the Lipschitz smoothness assumptions in (6) and (7) **only** needs to hold for the averaged estimate $\bar{\theta}_t$ in (5).

Learning from streaming data

Let (l_t) be a sequence of **independent** differentiable random functions possibly non-convex and their gradients **unbiased** estimates of $\nabla_\theta L$.[1]

Assumption 1 (unbiased gradients, κ-expected smoothness, σ-gradient noise)

Assume the following about $l_{t,i}$ for each $t \in \mathbb{N}$ with $i = 1, \ldots, n_t$. For some positive integer p, there exists $\kappa, \sigma > 0$ such that

- $\mathbb{E}[\nabla_\theta l_{t,i}(\theta)] = \nabla_\theta L(\theta)$,

[1] E.g., see Nesterov et al. [Nes+18] for definitions and properties of such functions.

Learning from streaming data

Let (l_t) be a sequence of **independent** differentiable random functions possibly non-convex and their gradients **unbiased** estimates of $\nabla_\theta L$.[1]

Assumption 1 (unbiased gradients, κ-expected smoothness, σ-gradient noise)

Assume the following about $l_{t,i}$ for each $t \in \mathbb{N}$ with $i = 1, \ldots, n_t$. For some positive integer p, there exists $\kappa, \sigma > 0$ such that

- $\mathbb{E}[\nabla_\theta l_{t,i}(\theta)] = \nabla_\theta L(\theta)$,
- $\mathbb{E}[\|\nabla_\theta l_{t,i}(\theta) - \nabla_\theta l_{t,i}(\theta')\|^p] \leq \kappa^p \mathbb{E}[\|\theta - \theta'\|^p]$,

[1] E.g., see Nesterov et al. [Nes+18] for definitions and properties of such functions.

Learning from streaming data

Let (l_t) be a sequence of **independent** differentiable random functions possibly non-convex and their gradients **unbiased** estimates of $\nabla_\theta L$.[1]

Assumption 1 (unbiased gradients, κ-expected smoothness, σ-gradient noise)

Assume the following about $l_{t,i}$ for each $t \in \mathbb{N}$ with $i = 1, \dots, n_t$. For some positive integer p, there exists $\kappa, \sigma > 0$ such that

- $\mathbb{E}[\nabla_\theta l_{t,i}(\theta)] = \nabla_\theta L(\theta)$,
- $\mathbb{E}[\|\nabla_\theta l_{t,i}(\theta) - \nabla_\theta l_{t,i}(\theta')\|^p] \leq \kappa^p \mathbb{E}[\|\theta - \theta'\|^p]$,
- $\mathbb{E}[\|\nabla_\theta l_{t,i}(\theta^*)\|^p] \leq \sigma^p$, $\forall \theta, \theta' \in \mathbb{R}^d$.

[1] E.g., see Nesterov et al. [Nes+18] for definitions and properties of such functions.

Learning from streaming data

SORBONNE UNIVERSITÉ

Let (l_t) be a sequence of **independent** differentiable random functions possibly non-convex and their gradients **unbiased** estimates of $\nabla_\theta L$.[1]

Assumption 1 (unbiased gradients, κ-expected smoothness, σ-gradient noise)

Assume the following about $l_{t,i}$ for each $t \in \mathbb{N}$ with $i = 1, \dots, n_t$. For some positive integer p, there exists $\kappa, \sigma > 0$ such that

- $\mathbb{E}[\nabla_\theta l_{t,i}(\theta)] = \nabla_\theta L(\theta)$,
- $\mathbb{E}[\|\nabla_\theta l_{t,i}(\theta) - \nabla_\theta l_{t,i}(\theta')\|^p] \leq \kappa^p \mathbb{E}[\|\theta - \theta'\|^p]$,
- $\mathbb{E}[\|\nabla_\theta l_{t,i}(\theta^*)\|^p] \leq \sigma^p$, $\forall \theta, \theta' \in \mathbb{R}^d$.

Takeaway. For SSG, we need Assumption 1 with $p = 2$, whereas for ASSG, we need $p = 4$.

[1] E.g., see Nesterov et al. [Nes+18] for definitions and properties of such functions.

Learning from streaming data

Classical setting with $n_t = 1$ (i.e., $\{C_\rho = 1, \rho = 0\}$).

Theorem 1 (Moulines and Bach [MB11])

Under Assumption 1 with $p = 2$, there exists explicit constants $C_\delta, C'_\delta, C''_\delta > 0$ such that for $\alpha \in (1/2, 1)$:

$$\delta_t \leq \frac{C_\delta \sigma^2}{\mu N_t^\alpha} + C'_\delta \exp(-\mu C''_\delta N_t^{1-\alpha}). \tag{8}$$

The bound in (8) can be divided into

- a **noise term** $C_\delta \sigma^2 / \mu N_t^\alpha$ and
- a sub-exponential term $C'_\delta \exp(-\mu C''_\delta N_t^{1-\alpha})$.

Takeaway. We should focus on **reducing the noise term** without harming the natural decay of the sub-exponential term.

Learning from streaming data

Streaming setting with $n_t = C_\rho$ (i.e., $\{C_\rho \in \mathbb{N}, \rho = 0\}$).

Theorem 2 (SSG)

Under Assumption 1 for $p = 2$, there exists explicit constants $C_\delta, C'_\delta, C''_\delta > 0$ such that for $\alpha \in (1/2, 1)$:

$$\delta_t \leq \frac{C_\delta \sigma^2}{\mu C_\rho^{1-\alpha-\beta} N_t^\alpha} + C'_\delta \exp\left(-\frac{\mu C''_\delta N_t^{1-\alpha}}{C_\rho^{1-\alpha-\beta}}\right). \tag{9}$$

Takeaway.

- The **noise term** in (9) is divided by $C_\rho^{1-\alpha-\beta}$, implying we achieve **variance reduction** by taking $\alpha + \beta < 1$.
- But this will not increase the convergence rate, which still is determined by $\alpha \in (1/2, 1)$.

Learning from streaming data

Streaming setting with $n_t = C_\rho t^\rho$ (i.e., $\{C_\rho \in \mathbb{N}, \rho \in [0,1)\}$).

Theorem 3 (SSG)

Under Assumption 1 for $p = 2$, there exists explicit constants $C_\delta, C'_\delta, C''_\delta > 0$ such that for $\alpha - \beta\rho \in (1/2, 1)$:

$$\delta_t \leq \frac{C_\delta \sigma^2}{\mu C_\rho^{1-\beta-\phi} N_t^\phi} + C'_\delta \exp\left(-\frac{\mu C''_\delta N_t^{1-\phi}}{C_\rho^{1-\beta-\phi}}\right), \tag{10}$$

with $\phi = ((1-\beta)\rho + \alpha)/(1+\rho)$.

Takeaway.

- The **noise term** is scaled by $C_\rho^{1-\beta-\phi}$, implying we should take $\alpha + \beta < 1$ to obtain **variance reduction**.
- **Increasing streaming rates** $(i.e., \rho > 0)$ can **accelerate convergence**, e.g., $\alpha = 2/3, \beta = 0$, gives $\delta_t = \mathcal{O}(N_t^{-(2/3+\rho)/(1+\rho)})$.

Learning from streaming data

Acceleration by averaging. Consider the averaging estimate $(\bar{\theta}_n)$ given in (5) derived with use of (θ_t) from (4).

Assumption 2 (Covariance of scores $(\nabla_\theta l_{t,i}(\theta^*))$)

There exists a non-negative self-adjoint operator Σ such that $\mathbb{E}[\nabla_\theta l_{t,i}(\theta^*) \nabla_\theta l_{t,i}(\theta^*)^\top] \preceq \Sigma$.

Learning from streaming data

Theorem 4 (ASSG)

Under Assumption 1 for $p = 4$ and Assumption 2, we have for $\alpha - \beta\rho \in (1/2, 1)$:

$$\bar{\delta}_t^{1/2} \leq \frac{\Lambda^{1/2}}{N_t^{1/2}} + \mathcal{O}(\max\{N_t^{-1+\phi/2}, N_t^{-\phi}\}), \tag{11}$$

where $\Lambda = \mathrm{Tr}(\nabla_\theta^2 L(\theta^*)^{-1} \Sigma \nabla_\theta^2 L(\theta^*)^{-1})$ and $\phi = ((1-\beta)\rho + \alpha)/(1+\rho)$.

Takeaway.

- Λ / N_t achieves the desirable **Cramer-Rao bound**, obtaining the optimal rate of $\bar{\delta}_t = \mathcal{O}(N_t^{-1})$.
- $\mathcal{O}(\max\{N_t^{-1+\phi/2}, N_t^{-\phi}\})$ insinuate that $\phi = 2/3$, e.g., by $\alpha = 2/3$ and $\beta = 1/3 \Rightarrow$ robustly achieve $\mathcal{O}(N_t^{-4/3})$, $\forall \rho \in [0, 1)$.

Learning from streaming data

Geometric median[2] is a generalization of the real median [Hal48], defined by

$$\theta^* := \arg\min_{\theta \in \mathbb{R}^d} \{L(\theta) := \mathbb{E}[\|X - \theta\| - \|X\|]\},$$

with gradient $\nabla_\theta L(\theta) = \mathbb{E}[\nabla_\theta l_t(\theta)]$, $\nabla_\theta l_t(\theta) = -(X_t - \theta)/\|X_t - \theta\|$.

Experiments

- Set $d = 10$ and fix $C_\gamma = \sqrt{10}$ and $\alpha = 2/3$ [CCZ13].
- (X_t) is standard Gaussian centered at $\theta = (-4, -3, 2, 1, 0, 1, 2, 3, 4, 5)^T \in \mathbb{R}^{10}$.
- Explore the errors for various data streams $n_t = C_\rho t^\rho$ with $N_t = 100000$ observations.

[2] E.g., see Kemperman [Kem87], Gervini [Ger08], and Godichon-Baggioni [GB16].

Learning from streaming data

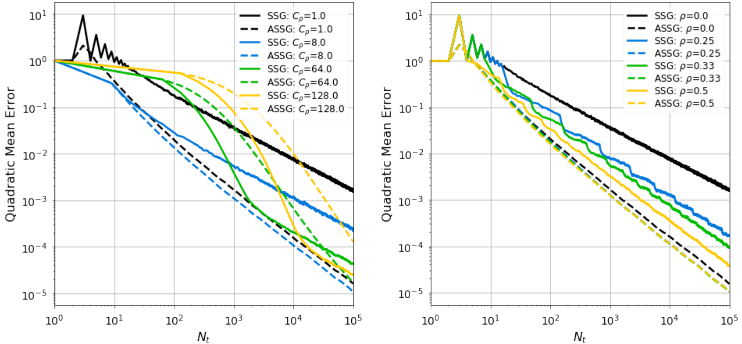


Figure 2: LHS: Constant streaming-batches, $\rho = 0$, $\beta = 0$. RHS: Varying streaming-batches, $C_\rho = 1$, $\beta = 0$.

Takeaway.

- Increasing mini-batch $\Rightarrow$ variance reduction.
- Increasing streaming rates $\Rightarrow$ increasing convergence rates (SSG).

Learning from streaming data

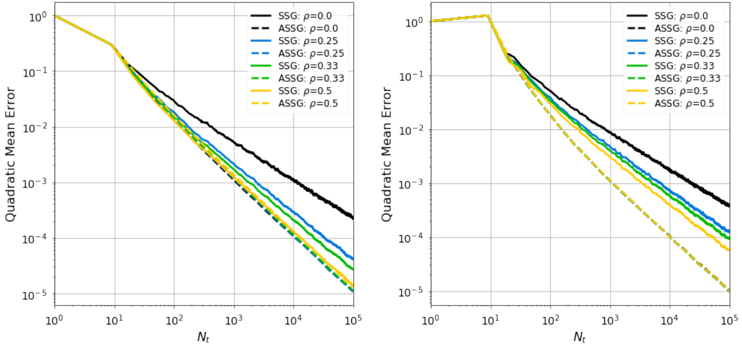


Figure 3: LHS: Varying streaming-batches, $C_\rho = 8$, $\beta = 0$. RHS: Varying streaming-batches, $C_\rho = 8$, $\beta = 1/3$.

Takeaway.

- Combining mini-batches with increasing streaming rates $\Rightarrow$ variance reduction and better convergence rates.
- $\alpha = 2/3$ and $\beta = 1/3 \Rightarrow$ ASSG robustly decay $\forall \rho \in [0, 1)$.

Learning from time-dependent streaming data

Assumption 2 ($D_\nu \nu_t$-dependence, $B_\nu \nu_t$-bias, κ_t-expected smoothness, σ_t-gradient noise)

Assume the following about l_t for each $t \in \mathbb{N}$. For some positive integer p, there exists positive sequences (ν_t), (κ_t), (σ_t) and $D_\nu, B_\nu \geq 0$,

- $\mathbb{E}[\|\mathbb{E}[\nabla_\theta l_t(\theta)|\mathcal{F}_{t-1}] - \nabla_\theta L(\theta)\|^p] \leq \nu_t^p (D_\nu^p \mathbb{E}[\|\theta - \theta^*\|^p] + B_\nu^p)$,

Learning from time-dependent streaming data

Assumption 2 ($D_\nu \nu_t$-dependence, $B_\nu \nu_t$-bias, κ_t-expected smoothness, σ_t-gradient noise)

Assume the following about l_t for each $t \in \mathbb{N}$. For some positive integer p, there exists positive sequences (ν_t), (κ_t), (σ_t) and $D_\nu, B_\nu \geq 0$,

- $\mathbb{E}[\|\mathbb{E}[\nabla_\theta l_t(\theta)|\mathcal{F}_{t-1}] - \nabla_\theta L(\theta)\|^p] \leq \nu_t^p(D_\nu^p \mathbb{E}[\|\theta - \theta^*\|^p] + B_\nu^p)$,
- $\mathbb{E}[\|\nabla_\theta l_t(\theta) - \nabla_\theta l_t(\theta')\|^p] \leq \kappa_t^p \mathbb{E}[\|\theta - \theta'\|^p]$,

Learning from time-dependent streaming data

Assumption 2 ($D_\nu \nu_t$-dependence, $B_\nu \nu_t$-bias, κ_t-expected smoothness, σ_t-gradient noise)

Assume the following about l_t for each $t \in \mathbb{N}$. For some positive integer p, there exists positive sequences (ν_t), (κ_t), (σ_t) and $D_\nu, B_\nu \geq 0$,

- $\mathbb{E}[\|\mathbb{E}[\nabla_\theta l_t(\theta)|\mathcal{F}_{t-1}] - \nabla_\theta L(\theta)\|^p] \leq \nu_t^p (D_\nu^p \mathbb{E}[\|\theta - \theta^*\|^p] + B_\nu^p)$,
- $\mathbb{E}[\|\nabla_\theta l_t(\theta) - \nabla_\theta l_t(\theta')\|^p] \leq \kappa_t^p \mathbb{E}[\|\theta - \theta'\|^p]$,
- $\mathbb{E}[\|\nabla_\theta l_t(\theta^*)\|^p] \leq \sigma_t^p$, $\forall \theta, \theta' \in \mathbb{R}^d$.

Learning from time-dependent streaming data

SORBONNE UNIVERSITÉ

Assumption 2 ($D_\nu \nu_t$-dependence, $B_\nu \nu_t$-bias, κ_t-expected smoothness, σ_t-gradient noise)

Assume the following about l_t for each $t \in \mathbb{N}$. For some positive integer p, there exists positive sequences (ν_t), (κ_t), (σ_t) and $D_\nu, B_\nu \geq 0$,

- $\mathbb{E}[\|\mathbb{E}[\nabla_\theta l_t(\theta)|\mathcal{F}_{t-1}] - \nabla_\theta L(\theta)\|^p] \leq \nu_t^p (D_\nu^p \mathbb{E}[\|\theta - \theta^*\|^p] + B_\nu^p)$,
- $\mathbb{E}[\|\nabla_\theta l_t(\theta) - \nabla_\theta l_t(\theta')\|^p] \leq \kappa_t^p \mathbb{E}[\|\theta - \theta'\|^p]$,
- $\mathbb{E}[\|\nabla_\theta l_t(\theta^*)\|^p] \leq \sigma_t^p$, $\forall \theta, \theta' \in \mathbb{R}^d$.

- $\nu_t = n_t^{-\nu}$, $\kappa_t = C_\kappa n_t^{-\kappa}$ and $\sigma_t = C_\sigma n_t^{-\sigma}$ with $\nu \in (0, \infty)$, $\kappa, \sigma \in [0, 1/2]$, and $C_\kappa, C_\sigma > 0$.

Learning from time-dependent streaming data

Assumption 2 ($D_\nu \nu_t$-dependence, $B_\nu \nu_t$-bias, κ_t-expected smoothness, σ_t-gradient noise)

Assume the following about l_t for each $t \in \mathbb{N}$. For some positive integer p, there exists positive sequences (ν_t), (κ_t), (σ_t) and $D_\nu, B_\nu \geq 0$,

- $\mathbb{E}[\|\mathbb{E}[\nabla_\theta l_t(\theta)|\mathcal{F}_{t-1}] - \nabla_\theta L(\theta)\|^p] \leq \nu_t^p(D_\nu^p \mathbb{E}[\|\theta - \theta^*\|^p] + B_\nu^p)$,
- $\mathbb{E}[\|\nabla_\theta l_t(\theta) - \nabla_\theta l_t(\theta')\|^p] \leq \kappa_t^p \mathbb{E}[\|\theta - \theta'\|^p]$,
- $\mathbb{E}[\|\nabla_\theta l_t(\theta^*)\|^p] \leq \sigma_t^p$, $\forall \theta, \theta' \in \mathbb{R}^d$.

- $\nu_t = n_t^{-\nu}$, $\kappa_t = C_\kappa n_t^{-\kappa}$ and $\sigma_t = C_\sigma n_t^{-\sigma}$ with $\nu \in (0, \infty)$, $\kappa, \sigma \in [0, 1/2]$, and $C_\kappa, C_\sigma > 0$.
- Long-range dependence is when $\nu \in (0, 1/2)$ and $\kappa, \sigma < 1/2$.

Learning from time-dependent streaming data

Assumption 2 ($D_\nu \nu_t$-dependence, $B_\nu \nu_t$-bias, κ_t-expected smoothness, σ_t-gradient noise)

Assume the following about l_t for each $t \in \mathbb{N}$. For some positive integer p, there exists positive sequences (ν_t), (κ_t), (σ_t) and $D_\nu, B_\nu \geq 0$,

- $\mathbb{E}[\|\mathbb{E}[\nabla_\theta l_t(\theta)|\mathcal{F}_{t-1}] - \nabla_\theta L(\theta)\|^p] \leq \nu_t^p (D_\nu^p \mathbb{E}[\|\theta - \theta^*\|^p] + B_\nu^p)$,
- $\mathbb{E}[\|\nabla_\theta l_t(\theta) - \nabla_\theta l_t(\theta')\|^p] \leq \kappa_t^p \mathbb{E}[\|\theta - \theta'\|^p]$,
- $\mathbb{E}[\|\nabla_\theta l_t(\theta^*)\|^p] \leq \sigma_t^p$, $\forall \theta, \theta' \in \mathbb{R}^d$.

- $\nu_t = n_t^{-\nu}$, $\kappa_t = C_\kappa n_t^{-\kappa}$ and $\sigma_t = C_\sigma n_t^{-\sigma}$ with $\nu \in (0, \infty)$, $\kappa, \sigma \in [0, 1/2]$, and $C_\kappa, C_\sigma > 0$.
- Long-range dependence is when $\nu \in (0, 1/2)$ and $\kappa, \sigma < 1/2$.
- Short-range dependence is when $\nu \in [1/2, \infty)$ and $\kappa, \sigma = 1/2$

Learning from time-dependent streaming data

SORBONNE UNIVERSITÉ

Assumption 2 ($D_\nu \nu_t$-dependence, $B_\nu \nu_t$-bias, κ_t-expected smoothness, σ_t-gradient noise)

Assume the following about l_t for each $t \in \mathbb{N}$. For some positive integer p, there exists positive sequences (ν_t), (κ_t), (σ_t) and $D_\nu, B_\nu \geq 0$,

- $\mathbb{E}[\|\mathbb{E}[\nabla_\theta l_t(\theta)|\mathcal{F}_{t-1}] - \nabla_\theta L(\theta)\|^p] \leq \nu_t^p (D_\nu^p \mathbb{E}[\|\theta - \theta^*\|^p] + B_\nu^p)$,
- $\mathbb{E}[\|\nabla_\theta l_t(\theta) - \nabla_\theta l_t(\theta')\|^p] \leq \kappa_t^p \mathbb{E}[\|\theta - \theta'\|^p]$,
- $\mathbb{E}[\|\nabla_\theta l_t(\theta^*)\|^p] \leq \sigma_t^p$, $\forall \theta, \theta' \in \mathbb{R}^d$.

- $\nu_t = n_t^{-\nu}$, $\kappa_t = C_\kappa n_t^{-\kappa}$ and $\sigma_t = C_\sigma n_t^{-\sigma}$ with $\nu \in (0, \infty)$, $\kappa, \sigma \in [0, 1/2]$, and $C_\kappa, C_\sigma > 0$.
- Long-range dependence is when $\nu \in (0, 1/2)$ and $\kappa, \sigma < 1/2$.
- Short-range dependence is when $\nu \in [1/2, \infty)$ and $\kappa, \sigma = 1/2$
- Independent unbiased case is when $\nu \to \infty$ and $\sigma = \kappa = 1/2$.

Learning from time-dependent streaming data

Assumption 2 ($D_\nu \nu_t$-dependence, $B_\nu \nu_t$-bias, κ_t-expected smoothness, σ_t-gradient noise)

Assume the following about l_t for each $t \in \mathbb{N}$. For some positive integer p, there exists positive sequences (ν_t), (κ_t), (σ_t) and $D_\nu, B_\nu \geq 0$,

- $\mathbb{E}[\|\mathbb{E}[\nabla_\theta l_t(\theta)|\mathcal{F}_{t-1}] - \nabla_\theta L(\theta)\|^p] \leq \nu_t^p (D_\nu^p \mathbb{E}[\|\theta - \theta^*\|^p] + B_\nu^p)$,
- $\mathbb{E}[\|\nabla_\theta l_t(\theta) - \nabla_\theta l_t(\theta')\|^p] \leq \kappa_t^p \mathbb{E}[\|\theta - \theta'\|^p]$,
- $\mathbb{E}[\|\nabla_\theta l_t(\theta^*)\|^p] \leq \sigma_t^p$, $\forall \theta, \theta' \in \mathbb{R}^d$.

- $\nu_t = n_t^{-\nu}$, $\kappa_t = C_\kappa n_t^{-\kappa}$ and $\sigma_t = C_\sigma n_t^{-\sigma}$ with $\nu \in (0, \infty)$, $\kappa, \sigma \in [0, 1/2]$, and $C_\kappa, C_\sigma > 0$.
- Long-range dependence is when $\nu \in (0, 1/2)$ and $\kappa, \sigma < 1/2$.
- Short-range dependence is when $\nu \in [1/2, \infty)$ and $\kappa, \sigma = 1/2$
- Independent unbiased case is when $\nu \to \infty$ and $\sigma = \kappa = 1/2$.

Takeaway. Assumption 2 allows dependent and biased gradients. For SSG, we need $p = 2$, whereas for ASSG, we need $p = 4$.

Learning from time-dependent streaming data

SORBONNE UNIVERSITÉ

Theorem 5 (SSG)

Under Assumption 2 with $p = 2$ *and* $\mu_\nu = \mu - \mathbb{1}_{\{\rho=0\}} 2D_\nu C_\rho^{-\nu} > 0$, *there exists* $C_\delta, C'_\delta, C''_\delta > 0$ *such that for* $\alpha - \rho\beta \in (1/2, 1)$*:*

$$\delta_t \leq \frac{C_\delta C_\sigma^2}{\mu_\nu C_\rho^{\frac{2\sigma-\beta-\alpha}{1+\rho}} N_t^{\frac{\rho(2\sigma-\beta)+\alpha}{1+\rho}}} + \frac{C'_\delta B_\nu^2}{\mu\mu_\nu C_\rho^{\frac{2\nu}{1+\rho}} N_t^{\frac{2\rho\nu}{1+\rho}}} + \pi_t, \qquad (12)$$

with $\pi_t = \mathcal{O}(\exp(-\mu C''_\delta N_t^{(1+\rho\beta-\alpha)/(1+\rho)} / C_\rho^{(1-\beta-\alpha)/(1+\rho)}))$.

Learning from time-dependent streaming data

SORBONNE UNIVERSITÉ

Theorem 5 (SSG)

Under Assumption 2 with $p = 2$ and $\mu_\nu = \mu - \mathbb{1}_{\{\rho=0\}} 2D_\nu C_\rho^{-\nu} > 0$, there exists $C_\delta, C'_\delta, C''_\delta > 0$ such that for $\alpha - \rho\beta \in (1/2, 1)$:

$$\delta_t \leq \frac{C_\delta C_\sigma^2}{\mu_\nu C_\rho^{\frac{2\sigma-\beta-\alpha}{1+\rho}} N_t^{\frac{\rho(2\sigma-\beta)+\alpha}{1+\rho}}} + \frac{C'_\delta B_\nu^2}{\mu\mu_\nu C_\rho^{\frac{2\nu}{1+\rho}} N_t^{\frac{2\rho\nu}{1+\rho}}} + \pi_t, \tag{12}$$

with $\pi_t = \mathcal{O}(\exp(-\mu C''_\delta N_t^{(1+\rho\beta-\alpha)/(1+\rho)} / C_\rho^{(1-\beta-\alpha)/(1+\rho)}))$.

- Taking $\alpha + \beta < 2\sigma \Rightarrow$ **variance reduction** for mini-batches $C_\rho > 1$.
- **Increasing streaming rates** $(\rho > 0) \Rightarrow$ **accelerate convergence**.
- **Bias** term B_ν is **independent** of the learning rate γ_t.
- **Positivity** of the dependence penalised **convexity constant** μ_ν is essential in all terms of (12) for attaining convergence.

Learning from time-dependent streaming data

Theorem 5 (SSG)

Under Assumption 2 with $p = 2$ *and* $\mu_\nu = \mu - \mathbb{1}_{\{\rho=0\}} 2 D_\nu C_\rho^{-\nu} > 0$*, there exists* $C_\delta, C'_\delta, C''_\delta > 0$ *such that for* $\alpha - \rho\beta \in (1/2, 1)$*:*

$$\delta_t \leq \frac{C_\delta C_\sigma^2}{\mu_\nu C_\rho^{\frac{2\sigma-\beta-\alpha}{1+\rho}} N_t^{\frac{\rho(2\sigma-\beta)+\alpha}{1+\rho}}} + \frac{C'_\delta B_\nu^2}{\mu \mu_\nu C_\rho^{\frac{2\nu}{1+\rho}} N_t^{\frac{2\rho\nu}{1+\rho}}} + \pi_t, \tag{12}$$

with $\pi_t = \mathcal{O}(\exp(-\mu C''_\delta N_t^{(1+\rho\beta-\alpha)/(1+\rho)} / C_\rho^{(1-\beta-\alpha)/(1+\rho)}))$.

- Taking $\alpha + \beta < 2\sigma \Rightarrow$ **variance reduction** for mini-batches $C_\rho > 1$.
- **Increasing streaming rates** $(\rho > 0) \Rightarrow$ **accelerate convergence**.
- **Bias** term B_ν is **independent** of the learning rate γ_t.
- **Positivity** of the dependence penalised **convexity constant** μ_ν is essential in all terms of (12) for attaining convergence.

Takeaway. Taking $\rho > 0$ and C_ρ large enough to ensure that $\mu_\nu > 0 \Rightarrow$ convergence even under long-range dependence and biased gradients.

Learning from time-dependent streaming data

Acceleration by averaging. In continuation of Assumption 2 with $\sigma_t = C_\sigma n_t^{-\sigma}$ for $\sigma \in [0, 1/2]$, we make the following assumption:

Assumption 3 (Covariance of scores $(\nabla_\theta l_t(\theta^*))$)

There exists a non-negative self-adjoint operator Σ such that $\forall t \geq 1$,

$$n_t^{2\sigma} \mathbb{E}[\nabla_\theta l_t(\theta^*) \nabla_\theta l_t(\theta^*)^\top] \preceq \Sigma + \Sigma_t,$$

where Σ_t is a positive symmetric matrix with Tr$(\Sigma_t) = C'_\sigma n_t^{-2\sigma'}$ for $C'_\sigma \geq 0$ and $\sigma' \in (0, 1/2]$.

- Assumption 3 is verified with $\sigma = 1/2$ and $C'_\sigma = 0$ in the unbiased i.i.d. case [GBWW21], e.g., see Assumption 2.

Learning from time-dependent streaming data

Theorem 6 (ASSG, $\sigma = 1/2$)

Under Assumption 2 with $p = 4$, Assumption 3 and $\mu_\nu = \mu - \mathbb{1}_{\{\rho=0\}} 2D_\nu C_\rho^{-\nu} > 0$, we have for $\alpha - \rho\beta \in (1/2, 1)$:

$$\bar{\delta}_t^{1/2} \leq \frac{\Lambda^{1/2}}{N_t^{1/2}} + \tilde{\mathcal{O}}\left(\max\left\{N_t^{-\frac{2+\rho(1+\beta)-\alpha}{2(1+\rho)}}, N_t^{-\frac{\rho(1-\beta)+\alpha}{1+\rho}}\right\}\right) + \mathbb{1}_{\{B_\nu \neq 0\}} \Psi_t,$$

with $\Lambda = \mathrm{Tr}(\nabla_\theta^2 L(\theta^*)^{-1} \Sigma \nabla_\theta^2 L(\theta^*)^{-1})$ and

$$\Psi_t = \tilde{\mathcal{O}}\left(\max\left\{N_t^{-\frac{\rho(1/2+\nu)}{2(1+\rho)}}, N_t^{-\frac{\rho(1-\beta+2\nu)+\alpha}{4(1+\rho)}}, N_t^{-\frac{\rho\nu}{1+\rho}}\right\}\right).$$

Takeaway.

- Streaming rates $\rho > 0$ or mini-batches $C_\rho > 1 \Rightarrow \mu_\nu > 0$.
- Cramer-Rao's bound is obtainable for $\sigma = 1/2$ under short-range dependence and biasedness $B_\nu \neq 0$.

Learning from time-dependent streaming data

Real-life time-dependent streaming data using geometric median

- Historical **hourly weather data**.[3]
- Dataset contains around five years (roughly 45000 data points) with dimension $d = 36$.
- Our geometric median is compared to the one calculated by the Weiszfeld's algorithm [WP09].

[3] The historical hourly weather dataset can be found on `https://www.kaggle.com/datasets/selfishgene/historical-hourly-weather-data`.

Learning from time-dependent streaming data

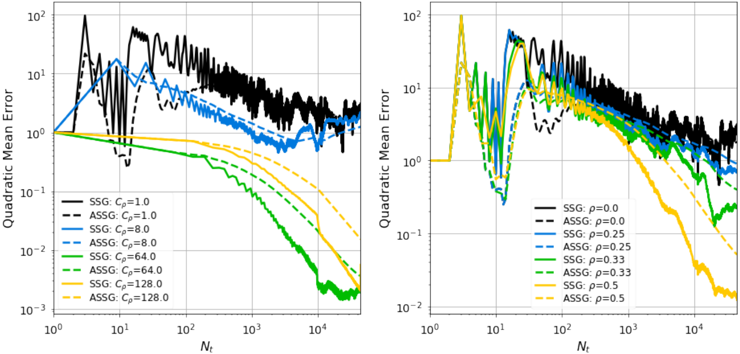


Figure 4: LHS: Constant streaming-batches, $\rho = 0$, $\beta = 0$. RHS: Varying streaming-batches, $C_\rho = 1$, $\beta = 0$.

Takeaway.

- Large mini-batches C_ρ ensures convexity through $\mu_\nu > 0$.
- Increasing streaming-batches ($\rho > 0$) ensures convexity, $\mu_\nu > 0$.

Learning from time-dependent streaming data

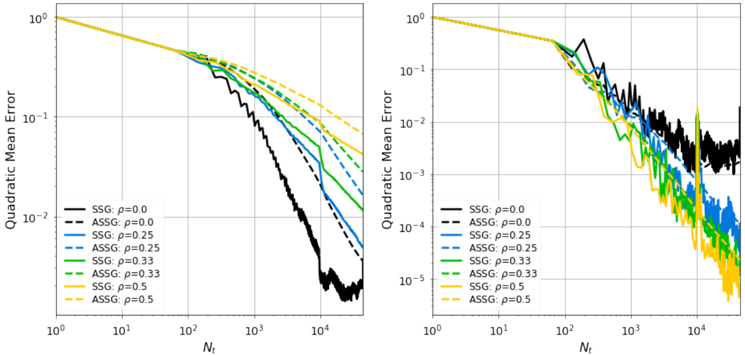


Figure 5: LHS: Varying streaming-batches, $C_\rho = 64$, $\beta = 0$. RHS: Varying streaming-batches, $C_\rho = 64$, $\beta = 1/3$.

Takeaway.

- Large C_ρ and increasing $(\rho > 0)$ streaming-batches accelerate learning, ensure convexity and break dependence.
- Obtain **a final error of only** 10^{-5} with $C_\rho = 64$, $\rho > 0$, $\beta = 1/3$.

Some final remarks

Some conclusions:

- Examined the SO problem in a streaming framework using time-dependent and biased gradients.
- Theoretical results formed heuristics that links the level of dependency and convexity to the SO problem parameters.
- SG-based methods can break long- and short-term dependence by using increasing streaming-batches.

Some final remarks

Some perspectives:

- Adaptive stochastic streaming gradient methods.
- Non-strongly convex objectives.
- Higher order stochastic streaming gradient methods.
- Probabilistic bounds; for any $\delta \in (0, 1)$, with probability at least $1 - \delta$, we bound the sequences $\{\|\theta_t - \theta^*\| : t \in \mathbb{N}\}$ and $\{\|L(\theta_t) - L(\theta^*)\| : t \in \mathbb{N}\}$.

Thank you for your attention!

References I

[BM13] Francis Bach and Eric Moulines. "Non-strongly-convex smooth stochastic approximation with convergence rate O (1/n)". In: *Advances in neural information processing systems* 26 (2013).

[BP11] Gérard Biau and Benoît Patra. "Sequential Quantile Prediction of Time Series". In: *Information Theory, IEEE Transactions on* 57 (Apr. 2011), pp. 1664 –1674.

[BB07] Léon Bottou and Olivier Bousquet. "The tradeoffs of large scale learning". In: *Advances in neural information processing systems* 20 (2007).

[Bra05] Richard C Bradley. "Basic properties of strong mixing conditions. A survey and some open questions". In: *Probability surveys* 2 (2005), pp. 107–144.

[CCZ13] Hervé Cardot, Peggy Cénac, and Pierre-André Zitt. "Efficient and fast estimation of the geometric median in Hilbert spaces with an averaged stochastic gradient algorithm". In: *Bernoulli* 19.1 (2013), pp. 18–43.

[Dou94] Paul Doukhan. "Mixing". In: *Mixing*. Springer, 1994, pp. 15–23.

[Dou12] Paul Doukhan. *Mixing: properties and examples*. Vol. 85. Springer Science & Business Media, 2012.

References II

[DHS11] John Duchi, Elad Hazan, and Yoram Singer. "Adaptive subgradient methods for online learning and stochastic optimization.". In: *Journal of machine learning research* 12.7 (2011).

[FZ19] Christian Francq and Jean-Michel Zakoian. *GARCH models: structure, statistical inference and financial applications*. John Wiley & Sons, 2019.

[FZH11] Christian Francq, Jean-Michel Zakoïan, and Lajos Horvath. "Merits and Drawbacks of Variance Targeting in GARCH Models". In: *Journal of Financial Econometrics* 9 (Sept. 2011), pp. 619–656.

[GP17] Sébastien Gadat and Fabien Panloup. "Optimal non-asymptotic bound of the Ruppert-Polyak averaging without strong convexity". In: *arXiv preprint arXiv:1709.03342* (2017).

[Ger08] Daniel Gervini. "Robust functional estimation using the median and spherical principal components". In: *Biometrika* 95.3 (2008), pp. 587–600.

[GB16] Antoine Godichon-Baggioni. "Estimating the geometric median in Hilbert spaces with stochastic gradient algorithms: Lp and almost sure rates of convergence". In: *Journal of Multivariate Analysis* 146 (2016), pp. 209–222.

References III

[GBWW21] Antoine Godichon-Baggioni, Nicklas Werge, and Olivier Wintenberger. "Non-Asymptotic Analysis of Stochastic Approximation Algorithms for Streaming Data". In: *arXiv preprint arXiv:2109.07117* (2021).

[GBWW22] Antoine Godichon-Baggioni, Nicklas Werge, and Olivier Wintenberger. "Learning from time-dependent streaming data with online stochastic algorithms". In: *arXiv preprint arXiv:2205.12549* (2022).

[Hal48] JBS Haldane. "Note on the median of a multivariate distribution". In: *Biometrika* 35.3-4 (1948), pp. 414–417.

[Has+09] Trevor Hastie et al. *The elements of statistical learning: data mining, inference, and prediction*. Vol. 2. Springer, 2009.

[Kem87] JHB Kemperman. "The median of a finite measure on a Banach space". In: *Statistical data analysis based on the L1-norm and related methods (Neuchâtel, 1987)* (1987), pp. 217–230.

[KY03] H. J. Kushner and G. G. Yin. *Stochastic Approximation and Recursive Algorithms and Applications*. Springer-Verlag, 2003.

[MB11] Eric Moulines and Francis Bach. "Non-asymptotic analysis of stochastic approximation algorithms for machine learning". In: *Advances in neural information processing systems* 24 (2011).

References IV

SORBONNE UNIVERSITÉ

[Nes+18] Yurii Nesterov et al. *Lectures on convex optimization*. Vol. 137. Springer, 2018.

[PJ92] Boris T Polyak and Anatoli B Juditsky. "Acceleration of stochastic approximation by averaging". In: *SIAM journal on control and optimization* 30.4 (1992), pp. 838–855.

[Rio17] Emmanuel Rio. *Asymptotic theory of weakly dependent random processes*. Vol. 80. Springer, 2017.

[RM51] Herbert Robbins and Sutton Monro. "A stochastic approximation method". In: *The annals of mathematical statistics* (1951), pp. 400–407.

[Rup88] David Ruppert. *Efficient estimations from a slowly convergent Robbins-Monro process*. Tech. rep. Cornell University Operations Research and Industrial Engineering, 1988.

[She20] Kevin Sheppard. *bashtage/arch:* Release 4.15 (Version 4.15). Zenodo, June 2020. DOI: `https://doi.org/10.5281/zenodo.593254`.

[Teo+07] Choon Hui Teo et al. "A scalable modular convex solver for regularized risk minimization". In: *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2007, pp. 727–736.

References V

[WWB18] Rachel Ward, Xiaoxia Wu, and Leon Bottou. *AdaGrad stepsizes: Sharp convergence over nonconvex landscapes, from any initialization*. 2018. arXiv: `1806.01811 [stat.ML]`.

[WP09] Endre Weiszfeld and Frank Plastria. "On the point for which the sum of the distances to n given points is minimum". In: *Annals of Operations Research* 167.1 (2009), pp. 7–41.

[Wer19] Nicklas Werge. "AdaVol". In: *GitHub repository* (2019). URL: `\url{https://github.com/nhwerge/AdaVol.git}`.

[Wer21] Nicklas Werge. "Predicting risk-adjusted returns using an asset independent regime-switching model". In: *Expert Systems with Applications* 184 (2021), p. 115576. ISSN: 0957-4174.

[WW22] Nicklas Werge and Olivier Wintenberger. "AdaVol: An adaptive recursive volatility prediction method". In: *Econometrics and Statistics* 23 (2022), pp. 19–35.

[Zin03] Martin Zinkevich. "Online convex programming and generalized infinitesimal gradient ascent". In: *Proceedings of the 20th international conference on machine learning (icml-03)*. 2003, pp. 928–936.

Projected SSG and ASSG

Projected SSG (PSSG)

The PSSG is defined by the following recursion,

$$\theta_t = \mathcal{P}_\Theta \left(\theta_{t-1} - \frac{\gamma_t}{n_t} \sum_{i=1}^{n_t} \nabla_\theta l_{t,i} \left(\theta_{t-1} \right) \right), \quad \theta_0 \in \Theta, \tag{13}$$

where Θ is a closed convex set in $\mathbb{R}^d$ and $\mathcal{P}_\Theta$ denotes the Euclidean projection onto Θ, i.e., $\mathcal{P}_\Theta(\theta) = \arg\min_{\theta' \in \Theta} \|\theta - \theta'\|_2$.

Projected ASSG (PASSG)

The PASSG is derived for all $t \in \mathbb{N}$ by the recursion,

$$\bar{\theta}_t = \frac{1}{N_t} \sum_{i=0}^{t-1} n_{i+1} \theta_i, \ \bar{\theta}_0 = 0, \ \text{with } (\theta_t) \text{ following (13)}, \tag{14}$$

where $N_t = \sum_{i=1}^{t} n_i$ denotes the accumulated sum of observations.

Learning from streaming data – *random streaming batches*

SORBONNE UNIVERSITÉ

Theorem (SSG)

Under Assumption 1 for $p = 2$, there exists explicit constants $C_\delta, C'_\delta, C''_\delta > 0$ such that for $\alpha - \beta\rho \in (1/2, 1)$:

$$\delta_t \leq \frac{C_\delta \sigma^2}{\mu C_\rho^{1-\beta-\phi} N_t^\phi} + C'_\delta \exp\left(-\frac{\mu C''_\delta N_t^{1-\phi}}{C_\rho^{1-\beta-\phi}}\right),$$

with $\phi = ((1-\beta)\rho + \alpha)/(1+\rho)$.

- Theorem 3 could be expanded to include *random* streaming batches where n_t is given such that

$$C_L t^{\rho_L} \leq n_t \leq C_H t^{\rho_H},$$

  with $\rho_L, \rho_H \in (-1, 1)$ and $C_L, C_H \geq 1$.
- This yields the modified convergence rate

$$\phi' = ((1-\beta)\rho_L + \alpha)/(1+\rho_H).$$

Verifying Assumption 2 using α-mixing conditions

Assumption 2 $\approx$ α-mixing condition for weakly dependence sequences.

- Assumption 2 can be verified using moment inequalities for partial sums of strongly mixing sequences [Rio17]; short-term dependence.
- For any positive integer p, Assumption 2 can be upper bounded by

$$\mathbb{E}[\|\mathbb{E}[\nabla_\theta l_t(\theta)|\mathcal{F}_{t-1}] - \nabla_\theta L(\theta)\|^p] \leq n_t^{-p}\mathbb{E}[\|S_t\|^p], \qquad (15)$$

  using Jensen's inequality, where $S_t = \sum_{i=1}^{n_t}(\nabla_\theta l_{t,i}(\theta) - \nabla_\theta L(\theta))$ is a d-dimensional vector.
- Under sufficient conditions, $\mathbb{E}[\|S_t\|^p] = \mathcal{O}(n_t^{p/2})$, meaning, (15) is at most $\mathcal{O}(n_t^{-p/2})$, i.e., ν_t^p is $\mathcal{O}(n_t^{-p/2})$.
- Examples: linear, non-linear and Markovian time series [Bra05; Dou12].

Verifying Assumption 2 for AR processes

Sequence of real-valued time-series (X_s); here s is short notation for indexing the sequence of observations, $(X_{N_t}, X_{N_t-1}, \ldots, X_{N_t-n_t} \equiv X_{N_{t-1}}, X_{N_{t-1}-1}, \ldots)$ with $N_t = \sum_{i=1}^{t} n_t$.

- Stationary AR(1) process $X_s = \theta X_{s-1} + \epsilon_s$ where $|\theta| < 1$ and (ϵ_s) is white noise with zero mean and variance σ_ϵ^2.
- Assumption 2 is verified for $p = 2$ if (X_s) has bounded moments; this is fulfilled by the natural constraint that $|\theta^*| < 1$.
- One can show $\mathbb{E}[\|\mathbb{E}[\nabla_\theta l_t(\theta)|\mathcal{F}_{t-1}] - \nabla_\theta L(\theta)\|^2]$ is less than

$$\frac{4(\theta-\theta^*)^2(1-(\theta^*)^{2n_t})^2\sigma_\epsilon^2}{(1-(\theta^*)^2)^4 n_t^2}\left(\sigma_\epsilon^2 + \frac{1}{1-(\theta^*)^2}\right),$$

- Thus, $D_\nu > 0$, $B_\nu = 0$, and ν_t is $\mathcal{O}(n_t^{-1})$.
- The remaining assumptions can be verified in the same way, κ_t and σ_t is $\mathcal{O}(n_t^{-1/2})$.
- Assumption 3 with $\Sigma = 4\sigma_\epsilon^4/(1-(\theta^*)^2)$ and $\Sigma_t = 0$.

Verifying Assumption 2 for MA processes

SORBONNE UNIVERSITÉ

- Assume that the underlying data generating process follows the MA(1)-process, $X_s = \epsilon_s + \phi^* \epsilon_{s-1}$, with $\phi^* \in \mathbb{R}$.
- One can show that $\theta = \phi^*/(1+(\phi^*)^2)$, thus, for any $\phi^* \in \mathbb{R}$ then $\theta \in (-1/2, 1/2)$.
- This yields,

$$\mathbb{E}[\|\mathbb{E}[\nabla_\theta l_t(\theta)|\mathcal{F}_{t-1}] - \nabla_\theta L(\theta)\|^2] = \frac{4(\theta-\theta^*)^2}{n_t^2} f_{\phi^*}(\epsilon_{N_{t-1}}),$$

  where $f_{\phi^*}(\epsilon_{N_{t-1}})$ is finite function depending on the moments of $(\epsilon_{N_{t-1}})$ and ϕ^*.
- Hence, we have $D_\nu > 0$ and $B_\nu = 0$ with ν_t being $\mathcal{O}(n_t^{-1})$.
- Similarly, it can be verified that κ_t and σ_t are $\mathcal{O}(n_t^{-1/2})$ by use of the reparametrization trick

Verifying Assumption 2 for ARCH processes

A process (ϵ_s) is called an ARCH(1) process with parameters α_0 and α_1 if it satisfies

$$\begin{cases} \epsilon_s = \sigma_s z_s, \\ \sigma_s^2 = \alpha_0 + \alpha_1 \epsilon_{s-1}^2, \end{cases} \tag{16}$$

where $\alpha_0 > 0$ and $\alpha_1 \geq 0$ ensures the non-negativity of the conditional variance process (σ_s^2), and the innovations (z_s) is white noise.

- Verification of Assumption 2 can be done using mixing conditions; Francq and Zakoian [FZ19, Theorem 3.5] showed that stationary ARCH processes are geometrically β-mixing, which implies α-mixing as well.

Verifying Assumption 2 for AR-ARCH processes

The process (X_s) is called an AR(1)-ARCH(1) process with parameters θ, α_0 and α_1 if it satisfies

$$\begin{cases} X_s = \theta X_{s-1} + \epsilon_s, \\ \epsilon_s = \sigma_s z_s, \\ \sigma_s^2 = \alpha_0 + \alpha_1 \epsilon_{s-1}^2. \end{cases} \tag{17}$$

where the innovations (z_s) is weak white noise.

- The statistical inference of this model is done using the squared loss for the AR-part and the QMLE for the ARCH part.
- Assumption 2 can be verified by Doukhan [Dou94, Proposition 6], which showed that ARMA-ARCH processes are β-mixing.

Alternative version of Theorem 1

Classical setting with $n_t = 1$ (i.e., $\{C_\rho = 1, \rho = 0\}$).

Theorem (Moulines and Bach [MB11])

Under Assumption 1 with $p = 2$, there exists explicit constants $C_\delta, C'_\delta, C''_\delta > 0$ such that for $\alpha \in (1/2, 1)$:

$$\delta_t \leq \frac{C_\delta \sigma^2}{\mu N_t^\alpha} + C'_\delta \exp\left(-\mu C''_\delta N_t^{1-\alpha}\right).$$

Hence, for any desired error $\epsilon > 0$, we have after

$$t > \max\left\{\left(\frac{C_\delta \sigma^2}{\mu \epsilon}\right)^{\frac{1}{\alpha}}, \left(\frac{1}{\mu C''_\delta} \log\left(\frac{C'_\delta}{\epsilon}\right)\right)^{\frac{1}{1-\alpha}}\right\}$$

iterations that $\delta_t < \epsilon$.

Alternative version of Theorem 2

Streaming setting with $n_t = C_\rho$ (i.e., $\{C_\rho \in \mathbb{N}, \rho = 0\}$).

Theorem (SSG)

Under Assumption 1 for $p = 2$*, there exists explicit constants* $C_\delta, C'_\delta, C''_\delta > 0$ *such that for* $\alpha \in (1/2, 1)$*:*

$$\delta_t \leq \frac{C_\delta \sigma^2}{\mu C_\rho^{1-\alpha-\beta} N_t^\alpha} + C'_\delta \exp\left(-\frac{\mu C''_\delta N_t^{1-\alpha}}{C_\rho^{1-\alpha-\beta}}\right).$$

Hence, for any desired error $\epsilon > 0$, we have after

$$t > \max\left\{\left(\frac{C_\delta \sigma^2}{\mu C_\rho^{1-\beta}\epsilon}\right)^{\frac{1}{\alpha}}, \left(\frac{1}{\mu C''_\delta C_\rho^\beta}\log\left(\frac{C'_\delta}{\epsilon}\right)\right)^{\frac{1}{1-\alpha}}\right\}$$

iterations that $\delta_t < \epsilon$.

Alternative version of Theorem 3

Streaming setting with $n_t = C_\rho t^\rho$ (i.e., $\{C_\rho \in \mathbb{N}, \rho \in [0,1)\}$).

Theorem (SSG)

Under Assumption 1 for $p = 2$, there exists explicit constants $C_\delta, C'_\delta, C''_\delta > 0$ such that for $\alpha - \beta\rho \in (1/2, 1)$:

$$\delta_t \leq \frac{C_\delta \sigma^2}{\mu C_\rho^{1-\beta-\phi} N_t^\phi} + C'_\delta \exp\left(-\frac{\mu C''_\delta N_t^{1-\phi}}{C_\rho^{1-\beta-\phi}}\right),$$

with $\phi = ((1-\beta)\rho + \alpha)/(1+\rho)$.

Hence, for any desired error $\epsilon > 0$, we have after

$$t > \max\left\{ \left(\frac{C_\delta \sigma^2}{\mu C_\rho^{1-\beta} \epsilon}\right)^{\frac{1}{(1-\beta)\rho+\alpha}}, \left(\frac{1}{\mu C''_\delta C_\rho^\beta} \log\left(\frac{C'_\delta}{\epsilon}\right)\right)^{\frac{1}{1+\beta\rho-\alpha}} \right\}$$

iterations that $\delta_t < \epsilon$.

Alternative version of Theorem 5

Streaming setting with $n_t = C_\rho t^\rho$ (i.e., $\{C_\rho \in \mathbb{N}, \rho \in [0,1)\}$).

Theorem (SSG)

Under Assumption 2 with $p = 2$ and $\mu_\nu = \mu - \mathbb{1}_{\{\rho=0\}} 2D_\nu C_\rho^{-\nu} > 0$, there exists $C_\delta, C'_\delta, C''_\delta, C'''_\delta > 0$ such that for $\alpha - \rho\beta \in (1/2, 1)$:

$$\delta_t \leq \frac{C_\delta C_\sigma^2}{\mu_\nu C_\rho^{\frac{2\sigma-\beta-\alpha}{1+\rho}} N_t^{\frac{\rho(2\sigma-\beta)+\alpha}{1+\rho}}} + \frac{C'_\delta B_\nu^2}{\mu\mu_\nu C_\rho^{\frac{2\nu}{1+\rho}} N_t^{\frac{2\rho\nu}{1+\rho}}} + \pi_t,$$

with $\pi_t = C'''_\delta \exp(-\mu C''_\delta N_t^{(1+\rho\beta-\alpha)/(1+\rho)} / C_\rho^{(1-\beta-\alpha)/(1+\rho)})$.

Hence, for any desired error $\epsilon > 0$, we have after

$$t > \max\left\{ \left(\frac{C_\delta C_\sigma^2}{\mu_\nu C_\rho^{2\sigma-\beta}\epsilon}\right)^{\frac{1}{(2\sigma-\beta)\rho+\alpha}}, \left(\frac{C'_\delta B_\nu^2}{\mu\mu_\nu C_\rho^{2\nu}\epsilon}\right)^{\frac{1}{2\rho\nu}}, \left(\frac{1}{\mu C''_\delta C_\rho^\beta}\log\left(\frac{C'''_\delta}{\epsilon}\right)\right)^{\frac{1}{1+\beta\rho-\alpha}} \right\}$$

iterations that $\delta_t < \epsilon$.

Alternative version of Theorem 6

SORBONNE UNIVERSITÉ

Theorem (ASSG)

Under Assumption 2 with $p = 4$, Assumption 3 and $\mu_\nu = \mu - \mathbb{1}_{\{\rho=0\}} 2D_\nu C_\rho^{-\nu} > 0$, we have for $\alpha - \rho\beta \in (1/2, 1)$:

$$\bar{\delta}_t^{1/2} \leq \frac{\Lambda^{1/2}}{N_t^{1/2}} \mathbb{1}_{\{\sigma=1/2\}} + \mathcal{O}\left(N_t^{-\frac{1+2\rho\sigma}{2(1+\rho)}}\right) \mathbb{1}_{\{\sigma<1/2\}} + \mathcal{O}\left(N_t^{-\frac{1+2\rho(\sigma+\sigma')}{2(1+\rho)}}\right) + \tilde{\mathcal{O}}\left(\max\left\{N_t^{-\frac{2+\rho(2\sigma+\beta)-\alpha}{2(1+\rho)}}, N_t^{-\frac{\rho(2\sigma-\beta)+\alpha}{1+\rho}}\right\}\right) + \mathbb{1}_{\{B_\nu \neq 0\}} \Psi_t,$$

with $\Lambda = \mathrm{Tr}(\nabla_\theta^2 L(\theta^*)^{-1} \Sigma \nabla_\theta^2 L(\theta^*)^{-1})$ and

$$\Psi_t = \tilde{\mathcal{O}}\left(\max\left\{N_t^{-\frac{\rho(\sigma+\nu)}{2(1+\rho)}}, N_t^{-\frac{1+\rho(\beta+\nu)-\alpha}{1+\rho}}, N_t^{-\frac{1+2\rho\nu}{2(1+\rho)}}, N_t^{-\frac{\delta/2+\rho\nu}{2(1+\rho)}}, N_t^{-\frac{2\rho\nu}{1+\rho}}\right\}\right),$$

where $\delta = \mathbb{1}_{\{B_\nu=0\}}(\rho(2\sigma-\beta)+\alpha) + \mathbb{1}_{\{B_\nu\neq 0\}} \min\{\rho(2\sigma-\beta)+\alpha, 2\rho\nu\}$.

Learning from time-dependent streaming data

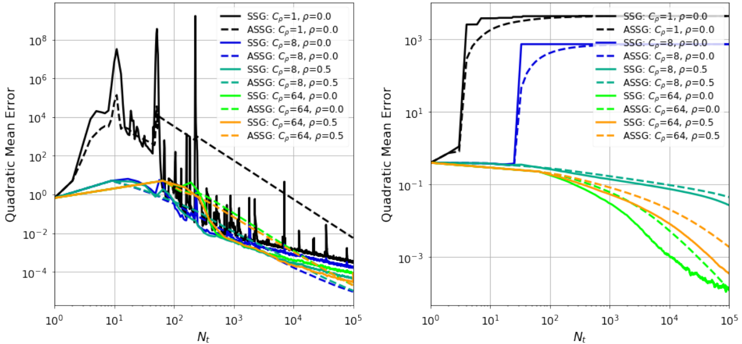


Figure 6: LHS: AR(1)-process, $X_t = \theta X_{t-1} + \epsilon_t$ with noise from fractional Brownian motion and Student't dist. with $df > 4$. RHS: ARCH(1)-process, $\epsilon_t = \sigma_t z_t$, $\sigma_t^2 = \alpha_0 + \alpha_1 \epsilon_{t-1}^2$, with Gaussian innovations z_t.

Takeaway. Large C_ρ and increasing $(\rho > 0)$ streaming-batches accelerate learning, ensure convexity and break dependence.

Learning from time-dependent streaming data

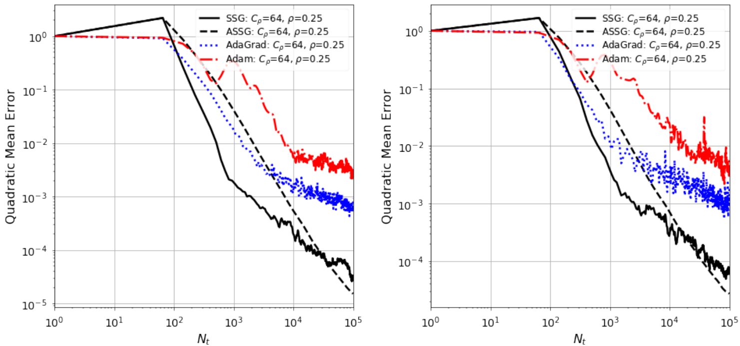


Figure 7: LHS: AR(1)-process with Gaussian noise. RHS:AR(1)-process with noise from fractional Brownian motion and Student't dist. with $df > 4$.

Takeaway. (1) SSG/ASSG could accelerate adaptive learning rates, e.g., AdaGrad and Adam. (2) Adaptive learning rates could ease the use of SSG/ASSG.

AdaVol: Objective

The **aim** is to make a natural adaption of the classical Quasi-Maximum Likelihood (QML) procedure to a *streaming setting* (where observations arrive continuously).

AdaVol is a recursive QML estimation procedure for GARCH models relying on the principles from stochastic approximations.

AdaVol is beneficial in at least three ways:

- *Estimation is faster and more memory-efficient* with a cost of only $\mathcal{O}(d)$ computations per recursion (compared to $\mathcal{O}(ndk)$).
- *Reducing numerical issues in convergence* when QML is combined with the Variance Targeting Estimation (VTE) technique[4].
- *Adaption to time-varying parameters* as AdaVol only treats new observations once.

[4] E.g., see Francq, Zakoïan, and Horvath [FZH11].

GARCH(p, q) Models

Let us recall the Generalized AutoRegressive Conditional Heteroskedasticity (GARCH) model:

- A process (X_t) is called a GARCH(p, q) process with parameter vector $\theta = (\omega, \alpha_1, \ldots, \alpha_p, \beta_1, \ldots, \beta_q)^T$ if it satisfies

$$\begin{cases} X_t = \sigma_t Z_t, \\ \sigma_t^2 = \omega + \sum_{i=1}^p \alpha_i X_{t-i}^2 + \sum_{j=1}^q \beta_j \sigma_{t-j}^2, \end{cases} \tag{18}$$

  where ω, α_i, and β_j for $1 \le i \le p$ and $1 \le j \le q$ are non-negative parameters ensuring the non-negativity of the conditional variance process (σ_t^2).
- The innovations (Z_t) is a sequence of i.i.d. random variables with $\mathbb{E}[Z_0] = 0$ and $\mathbb{E}[Z_0^2] = 1$.

GARCH(p,q) Models combined with VTE

Combine GARCH in (18) with VTE:

- The VTE reparametrization is obtained by defining $\omega = \gamma^2(1 - \sum_{i=1}^{p} \alpha_i - \sum_{j=1}^{q} \beta_j)$, where γ is the sample volatility.
- The volatility process of the GARCH(p,q) process in (18) can then be rewritten as

$$(\sigma_t^2 - \gamma^2) = \sum_{i=1}^{p} \alpha_i (X_{t-i}^2 - \gamma^2) + \sum_{j=1}^{q} \beta_j (\sigma_{t-j}^2 - \gamma^2).$$

- The remaining parameters $\theta = (\alpha_1, \ldots, \alpha_p, \beta_1, \ldots, \beta_q)^T \in \mathbb{R}_+^{p+q}$ is estimated by the QML method.
- Note that one does **not** need VTE.

QML of GARCH(p, q) Models combined with VTE

- Quasi likelihood loss is given by $\widehat{l}_t(\theta) = 2^{-1}(X_t^2/\widehat{\sigma}_t^2(\theta) + \log \widehat{\sigma}_t^2(\theta))$ with first derivative

$$\nabla \widehat{l}_t(\theta) = \nabla \widehat{\sigma}_t^2(\theta) \left(\frac{\widehat{\sigma}_t^2(\theta) - X_t^2}{2\widehat{\sigma}_t^4(\theta)} \right),$$

  where $\nabla \widehat{\sigma}_t^2(\theta) = \vartheta_t(\theta) + \sum_{j=1}^q \beta_j \nabla \widehat{\sigma}_{t-j}^2(\theta)$ with $\vartheta_t(\theta) = (X_{t-1}^2 - \gamma^2, \ldots, X_{t-p}^2 - \gamma^2, \widehat{\sigma}_{t-1}^2(\theta) - \gamma^2, \ldots, \widehat{\sigma}_{t-q}^2(\theta) - \gamma^2)^T \in \mathbb{R}^{p+q}$.
- Parameter space:

$$\mathcal{K} = \left\{ (\alpha_1, \cdots, \alpha_p, \beta_1, \ldots, \beta_q) \in \mathbb{R}_+^{p+q} \middle| \sum_{i=1}^p \alpha_i + \sum_{j=1}^q \beta_j < 1 \right\}.$$

Adaptive recursive QML estimation for GARCH(p, q) Models

- Our recursive QML method relies on stochastic approximations[5].

The recursive method is given by

$$\hat{\theta}_t = \hat{\theta}_{t-1} - \eta_{t-1} \nabla_\theta \hat{l}_t(\hat{\theta}_{t-1}),$$

where the *learning sequence* (η_t) is a decreasing sequence of positive numbers satisfying $\sum_{i=1}^t \eta_i = \infty$ and $\sum_{i=1}^t \eta_i^2 < \infty$ as $t \to \infty$.

[5] Robbins and Monro [RM51]
[6] Duchi, Hazan, and Singer [DHS11]
[7] Ward, Wu, and Bottou [WWB18]
[8] Zinkevich [Zin03]

Adaptive recursive QML estimation for GARCH(p, q) Models

- Our recursive QML method relies on stochastic approximations[5].
- Adaptive learning with AdaGrad[6], which has shown promising results in non-convex optimization[7].

The recursive method is given by

$$\hat{\theta}_t = \hat{\theta}_{t-1} - \frac{\eta}{\sqrt{\sum_{i=1}^{t} \nabla_\theta \hat{l}_i(\hat{\theta}_{i-1})^2 + \epsilon}} \nabla_\theta \hat{l}_t(\hat{\theta}_{t-1}),$$

where $\eta, \epsilon > 0$.

[5]Robbins and Monro [RM51]
[6]Duchi, Hazan, and Singer [DHS11]
[7]Ward, Wu, and Bottou [WWB18]
[8]Zinkevich [Zin03]

Adaptive recursive QML estimation for GARCH(p, q) Models

- Our recursive QML method relies on stochastic approximations[5].
- Adaptive learning with AdaGrad[6], which has shown promising results in non-convex optimization[7].
- Project $\hat{\theta}_t$ onto $\mathcal{K}$, preventing large jumps and enforcing convergence[8].

Our recursive method is given by

$$\hat{\theta}_t = \text{Projection}_{\mathcal{K}} \left[\hat{\theta}_{t-1} - \frac{\eta}{\sqrt{\sum_{i=1}^{t} \nabla_\theta \hat{l}_i(\hat{\theta}_{i-1})^2 + \epsilon}} \nabla_\theta \hat{l}_t(\hat{\theta}_{t-1}) \right],$$

where $\eta, \epsilon > 0$ with parameter space $\mathcal{K} = \{(\alpha_1, \ldots, \alpha_p, \beta_1, \ldots, \beta_q) \in \mathbb{R}_+^{p+q} | \sum_{i=1}^{p} \alpha_i + \sum_{j=1}^{q} \beta_j < 1\}$.

[5] Robbins and Monro [RM51]
[6] Duchi, Hazan, and Singer [DHS11]
[7] Ward, Wu, and Bottou [WWB18]
[8] Zinkevich [Zin03]

Applications - Real-life Observations

Real-life observations:

- Consider daily log-returns (r_t) of stock market indices.
- GARCH$(1,1)$ model with initial value $\hat{\theta}_0 = \tilde{\theta}_0 = (5 \cdot 10^{-5}, 0.05, 0.9)^T$.

Stock Market Index		Period
Standard & Poor's 500	(S&P500)	Jan. 1950 - Sep. 2020

Table 1: The observations consist of daily log-returns which are defined as log differences of the closing prices of the index between two consecutive days.

Iterative QMLE $\tilde{\theta}_n$:

- Estimated at every two thousand incremental using all observations up to this point, i.e., $(\tilde{\theta}_t)_{(k-2000)+1 \le t \le k}$ is estimated using $(X_t)_{1 \le t \le k}$ for $k = 2000, 4000, \dots, n$ (**i.e., forward-looking with at most** 2000 **observations**).
- We use the (bounded) *L-BFGS* algorithm to solve for $\tilde{\theta}_n$.

Applications - Real-life Observations - S&P500

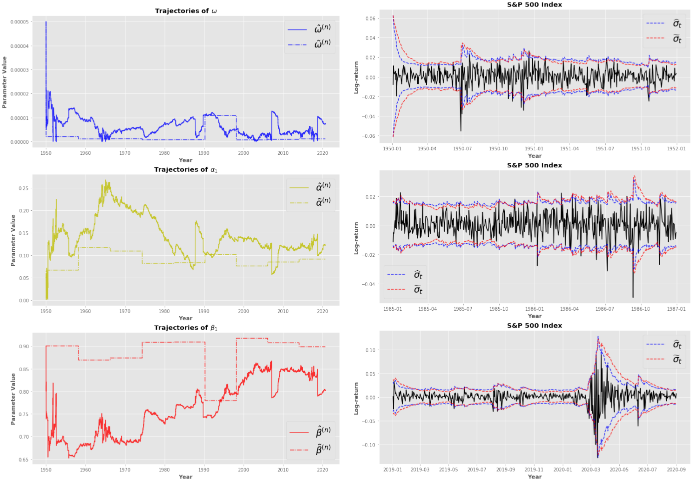


Figure 8: Left: Trajectory of QML estimates. Right: Log-returns r_t with confidence intervals in three different periods.

Applications - Accuracy Score

- Measure the accuracy by studying the conditional quantiles using the predicted volatility processes[9].
- Under the assumption of standard Gaussian innovations, X_t is Gaussian with zero mean and variance σ_t^2.
- For any $\alpha \in (0,1)$, the α-quantile of a Gaussian distribution $\mathcal{N}(0,\sigma_t^2)$ is $\sigma_t \Phi^{-1}(\alpha)$ ($\Phi^{-1}(\alpha)$ is the α-quantile of the standard Gaussian one).
- The α-quantile loss function is defined as

$$\rho_\alpha(X_t,\sigma_t) = \begin{cases} \alpha\left(X_t - \Phi^{-1}(\alpha)\sigma_t\right), & \text{for } X_t > \Phi^{-1}(\alpha)\sigma_t, \\ (1-\alpha)\left(\Phi^{-1}(\alpha)\sigma_t - X_t\right), & \text{for } X_t \le \Phi^{-1}(\alpha)\sigma_t, \end{cases}$$

with tilting parameter $\alpha \in (0,1)$.

[9] Biau and Patra [BP11]

Applications - Accuracy Score

- We evaluate across the α-quantile scores ρ_α of (σ_t) by the (normalized) cumulative α-quantile scoring function QS_α:

$$QS_\alpha(X_n, \sigma_n) = \frac{1}{n} \sum_{t=1}^{n} \sum_{m=1}^{M} \rho_{\alpha_m}(X_t, \sigma_t),$$

  with M as the number of quantiles $\alpha = \{\alpha_1, \ldots, \alpha_M\}$.
- The **lowest** QS_α score indicates the **best** ability of volatility forecast.

Applications - Real-life Observations - S&P500

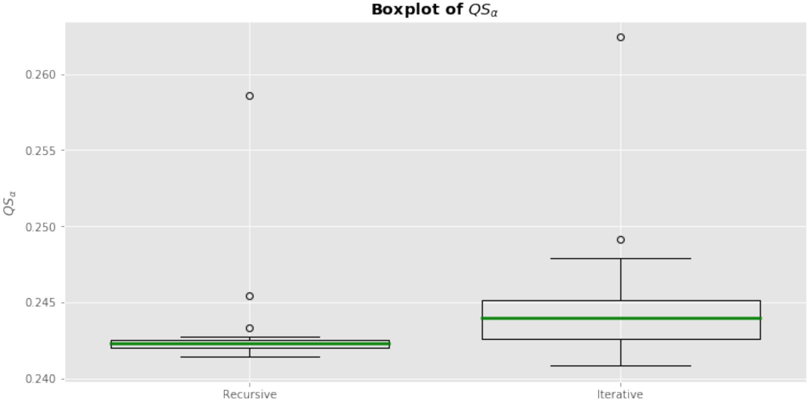


Figure 9: Boxplot of QS_α scores for $\alpha = \{0.01, 0.02, \ldots, 0.99\}$, using the GARCH$(1,1)$ model on the log-returns r_t of S&P500 Index with **random initial value** in $\mathcal{K}$.

Summary

- AdaVol; an adaptive approach to recursively estimate GARCH model parameters in a streaming setting using the VTE technique.
- AdaVol's design showed to produce robust and adaptive estimates.
- Time-varying parameters was an advantage for real-life observations.
- AdaVol is computationally efficient.

Model	n	AdaVol	`arch`
GARCH$(1,1)$	1000	1.00	204.89
	2000	1.00	233.86
GARCH$(2,2)$	1000	1.00	322.33
	2000	1.00	328.50

Table 2: Relative speed comparison between AdaVol implementation in `Python` [Wer19] and `arch` version 4.15 [She20]. A value of 1.00 means the method is the fastest. A value of 204.89 means the estimation time of the method is 204.89 times larger than the fastest.