Stochastic Gradient Descent: Going As Fast As Possible But Not Faster

Alice Schoenauer-Sebag[1], Marc Schoenauer[2], Michèle Sebag[2]
(1) Altschuler & Wu lab, UCSF
(2)TAO, CNRS – INRIA – LRI – Université Paris-Sud

Jan. 9th, 2018

Position of the talk

Given a black-box optimization algorithm, and its hyper-parameters:

Offline setting AUTO-ML

- Find good hp values, used during the whole run
  e.g. kernel; tree depth; regularization weights...

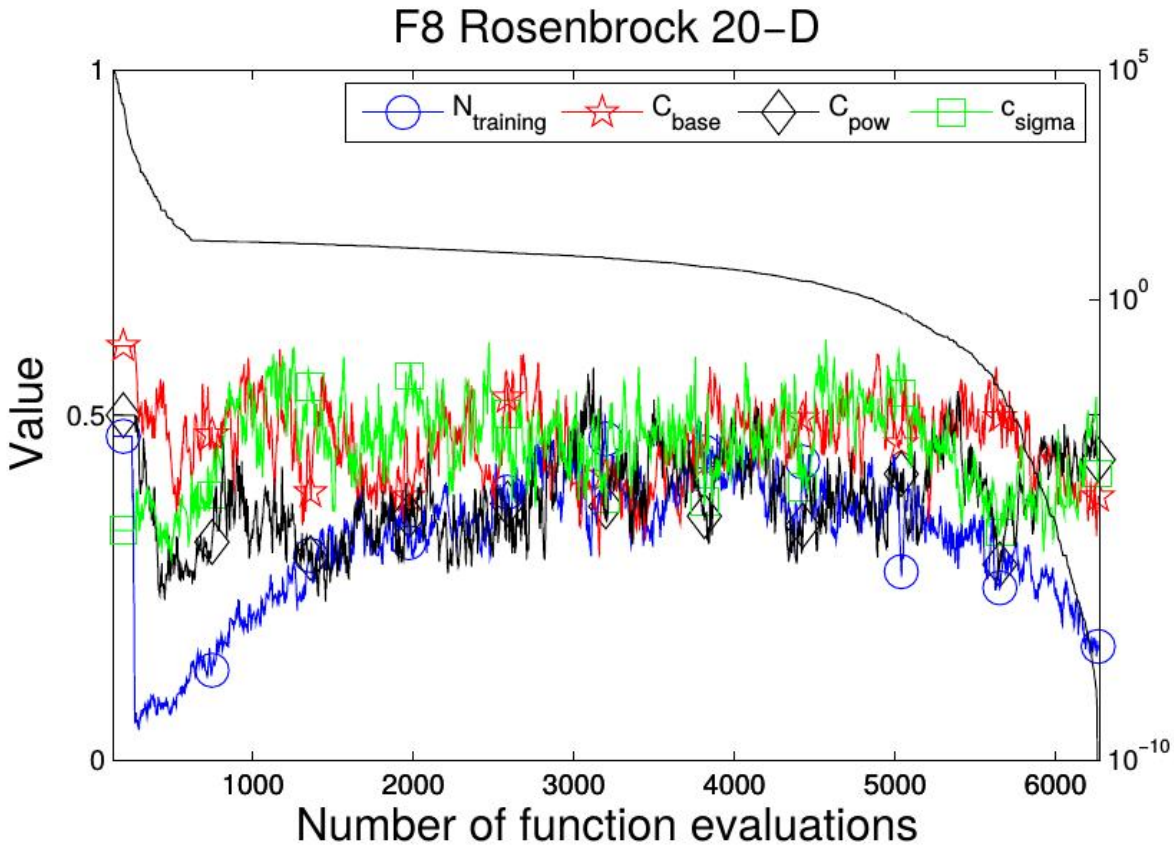
Online setting

- Adjust the hp values **during** the run
- NB: when considering a *fixed* schedule (e.g. $\eta_t = \frac{\eta}{\sqrt{t}}$), you are in the offline setting.

Offline vs Online hyper-parameter setting

PROS/CONS

- More computationally expensive by orders of magnitude
- Potentially huge gains



from Loshchilov et al. 10-14

Focusing on online adaptation

- Large Scale Machine Learning
  Increasing adoption of Stochastic Gradient Descent

$$\theta_{t+1} = \theta_t - \eta_t \nabla \mathcal{L}_\theta$$

- Adjusting η_t ?

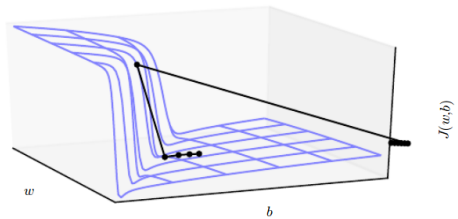
  Robbins & Munro 51; Jacobs 88; Nesterov 83
  Sutskever et al. 13; Duchi et al. 13; Kingma & Ba 14

- Vanishing and Exploding gradients
- Most usually

$$\eta_t \sim \frac{1}{\sqrt{t}} \text{ or } \frac{1}{t}$$

Claims and Goal

Goal: Go as fast as possible



Issue: There are explosions

Claim

- We cannot predict them with reasonable cost
- We can observe them
- We can cure them

Remark

- If explosions are controlled, we can go faster
  speed-up and brake

SALeRA: Safe Agnostic Learning Rate Adaptation

1st ingredient: Going faster

- Compare gradient directions with an agnostic cumulative path

2nd ingredient: Catastrophic event manager

- Change detection test: Page-Hinkley Page 54, Hinkley 70
- Instant cure

$$\theta_t = \theta_{t-1}$$

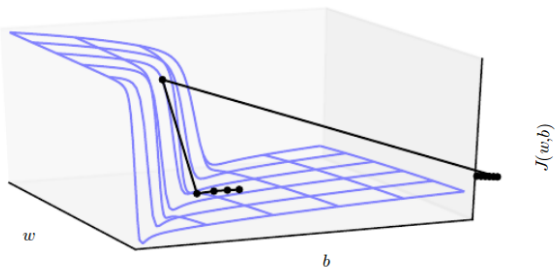
$$\eta_t = \frac{1}{2}\eta_{t-1}$$

Claim: both questions can be addressed using statistical tests

Catastrophic events in Deep Learning

There exist steep cliff structures in the derivative landscape

Goodfellow et al. 16



Catastrophic episodes

Avoiding cliff neighbourhood:

- Regularization: L_1, L_2, Max-norm
- Gradient clipping

  Pascanu et al., 13
- Batch normalization

  Ioffe et Szegedy, 15
- Weight initialization

  Glorot et Bengio, 10

Gradient descent

Given $\mathcal{D} = \{(x_i, y_i), i \in [[1, N]]\}$,
Minimize loss function $\mathcal{L}$ on parameter space $\Theta \subset \mathcal{R}^d$

Gradient descent: Iterate

- Compute full gradient

$$g_t = \frac{1}{n} \sum_{1}^{N} \nabla_\theta \mathcal{L}(x_i; \theta_t)$$

- Update

$$\theta_{t+1} = \theta_t - \eta_t g_t$$

with θ_t solution at step t, η_t learning rate

Stochastic gradient descent

Stochastic gradient descent (SGD)

- Sample a mini-batch MB (*) of size $n < N$ (**) from $\mathcal{D}$
- Compute gradient

$$g_t = \frac{1}{n} \sum_{MB} \nabla_\theta \mathcal{L}(x; \theta_t)$$

- Update

$$\theta_{t+1} = \theta_t - \eta g_t$$

(*) About selecting the minibatch a.k.a curriculum learning

- Start with easy examples — Bengio et al. 09
- Self-Paced Learning — Kumar et al. 10
- Prefer hardest examples — Loshchilov et al. 16
- It depends — Lopedriza et al. 16

(**) $n << N$? — Keskar et al. 17

Learning rate deterministic schedule (1/2)

$$g_t = \frac{1}{n} \sum_{MB} \nabla_\theta \mathcal{L}(x; \theta_t)$$

Learning rate decay Robbins and Monroe, 51

$$\eta_t = \frac{\eta_0}{1 + \nu t}$$

$$\theta_{t+1} = \theta_t - \eta_t g_t$$

With η_0: initial learning rate, ν hyper-parameter

Relaxing the gradient (2/2)

$$g_t = \frac{1}{n} \sum_i \nabla_\theta \mathcal{L}(x_i; \theta_t - \gamma v_{t-1})$$

Nesterov accelerated gradient Nesterov 83; Sutskever et al., 13

$$v_t = \gamma v_{t-1} + \eta g_t$$
$$\theta_{t+1} = \theta_t - v_t$$

With η learning rate, γ hyper-parameter $[.8, .999]$

Learning rate adaptive schedule 1/2

Adagrad Duchi et al., 13

For each coordinate $j \in [[1, d]]$

$$g_{t,j} = \frac{1}{n} \sum_i \frac{\partial \mathcal{L}}{\partial \theta_j}(x_i; \theta_t)$$

$$G_{t,j} = \sum_{k=1}^{t} g_{k,j}^2$$

$$\eta_{t,j} = \frac{\eta_0}{\sqrt{G_{t,j} + \epsilon}}$$

$$\theta_{t+1,j} = \theta_{t,j} - \eta_{t,j} g_{t,j}$$

With η_0 initial learning rate, $\epsilon << 1$

Learning rate adaptive schedule 2/2

Adam Kingma & Ba, 14

For each coordinate $j \in [[1, d]]$

$$g_{t,j} = \frac{1}{n} \sum_i \frac{\partial \mathcal{L}}{\partial \theta_j}(x_i; \theta_t)$$

$$m_{t,j} = \beta_1 m_{t-1} + (1 - \beta_1) g_{t,j}$$
$$v_{t,j} = \beta_2 v_{t-1} + (1 - \beta_2) g_{t,j}^2$$
$$\hat{m}_t = m_t / (1 - \beta_1^t) \qquad \hat{v}_t = v_t / (1 - \beta_2^t)$$
$$\theta_{t+1,j} = \theta_{t,j} - \frac{\eta_0}{\sqrt{\hat{v}_{t,j}} + \epsilon} \hat{m}_{t,j}$$

With η_0: initial learning rate, $\epsilon << 1$ and $\beta_1 < \beta_2$ (default $\beta_1 = .9$, $\beta_2 = .999$)

The delta-bar-delta rule

Asymmetrical schedule Jacobs, 88

For each coordinate $j \in [[1, d]]$

$$g_{t,j} = \frac{1}{n} \sum_{MB} \frac{\partial \mathcal{L}}{\partial \theta_j}(x; \theta_t)$$

$$\begin{aligned} v_{t,j} &= \gamma v_{t-1,j} + (1-\gamma) g_{t,j} \\ \eta_{t,j} &= \eta_{t-1,j} + \beta \text{ if } \operatorname{sgn}(g_{t,j}) = \operatorname{sgn}(v_{t,j}) \\ &= \eta_{t-1,j}/\nu \quad \text{otherwise} \\ \theta_{t+1} &= \theta_t - \eta_t g_t \end{aligned}$$

with ν, γ, β hyper-parameters

Additive increase
Multiplicative decrease

Background

- Catastrophic events in Deep Learning
- SGD and learning rate adaptation

S-ALeRA

- Speeding up: Agnostic learning rate adaptation
- Catastrophic event management

Experiments

The $(\mu, \lambda)-$Evolution Strategy

Minimize $f : \mathbb{R}^n \to \mathbb{R}$

Initialize distribution parameters $\boldsymbol{m}, \sigma, \mathbf{C}$, set population size $\lambda \in \mathbb{N}$

While not terminate

1. Sample distribution $\mathcal{N}\left(\boldsymbol{m}, \sigma^2 \mathbf{C}\right) \to \boldsymbol{x}_1, \ldots, \boldsymbol{x}_\lambda \in \mathbb{R}^n$

$$\mathbf{x}_i \sim \boldsymbol{m} + \sigma\, \mathcal{N}_i(\mathbf{0}, \mathbf{C}) \qquad \text{for } i = 1, \ldots, \lambda$$

2. Evaluate $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_\lambda$ on f

$$\text{Compute } f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_\lambda)$$

3. Update parameters

$$\boldsymbol{m}, \sigma, \mathbf{C} \leftarrow F(\boldsymbol{m}, \sigma, \mathbf{C}, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_\lambda, f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_\lambda))$$

The $(\mu, \lambda)-$Evolution Strategy (2)

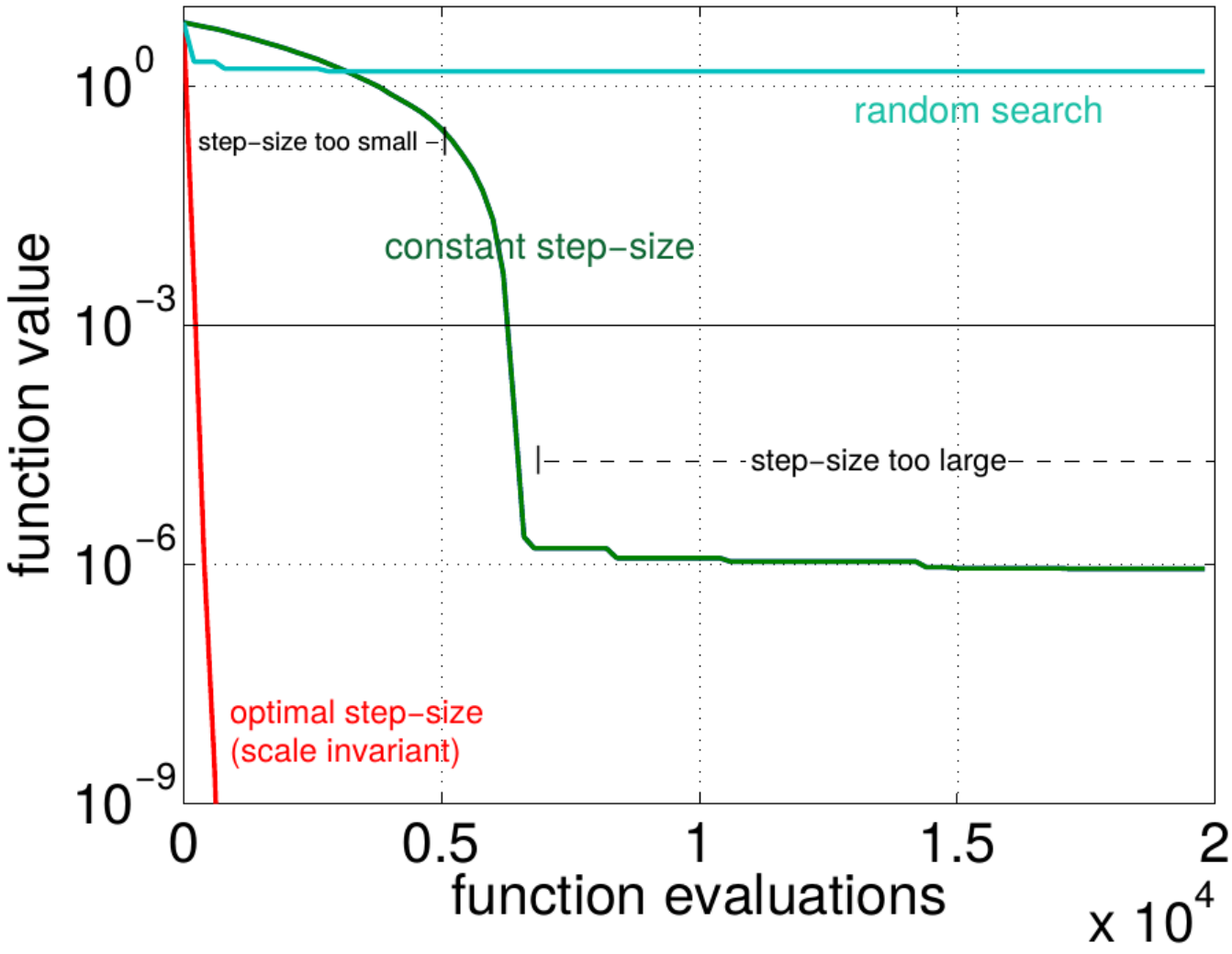
Gaussian Mutations

- mean vector $\boldsymbol{m} \in \mathbb{R}^n$ is the current solution
- the so-called step-size $\sigma \in \mathbb{R}_+$ controls the *step length*
- the covariance matrix $\mathbf{C} \in \mathbb{R}^{n \times n}$ determines the shape of the distribution ellipsoid

How to update $\boldsymbol{m}$, σ, and $\mathbf{C}$?

- $m = \frac{1}{\mu} \sum_{i=1}^{i=\mu} x_{i:\lambda}$ Selection of best μ samples
- Adaptive σ and $\mathbf{C}$

Need for adaptation

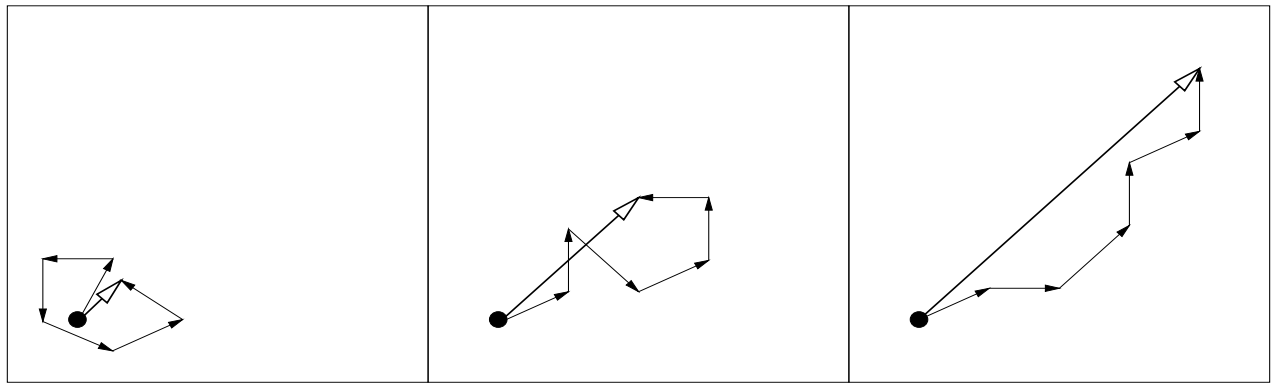


Sphere function $f(x) = \sum x_i^2, n = 10$

Cumulative Step-Size Adaptation (CSA)

Measure the length of the *evolution path*

the pathway of the mean vector $\boldsymbol{m}$ in the generation sequence



$\downarrow$

decrease σ

Rationale

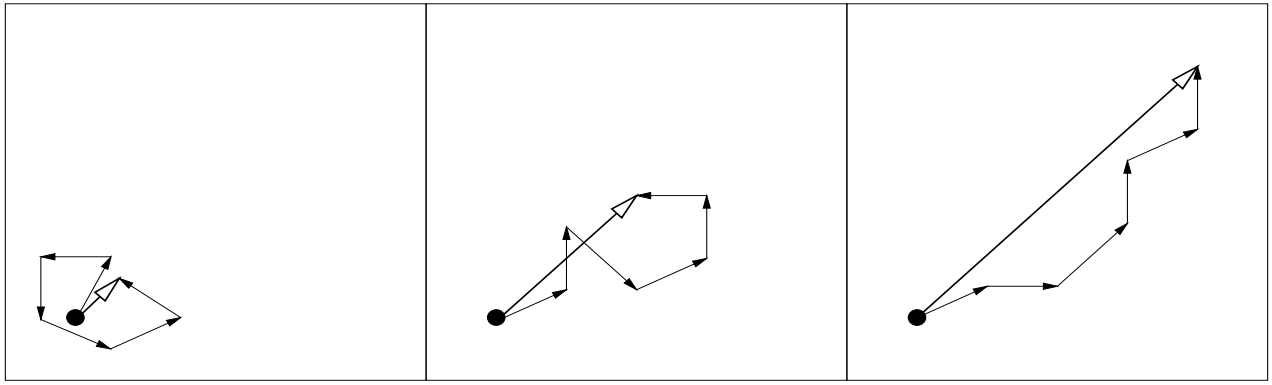
If successive moves are 'less correlated' than a random walk, step size should be decreased

hovering around the optimum

Cumulative Step-Size Adaptation (CSA)

Measure the length of the *evolution path*

the pathway of the mean vector $\boldsymbol{m}$ in the generation sequence



↓

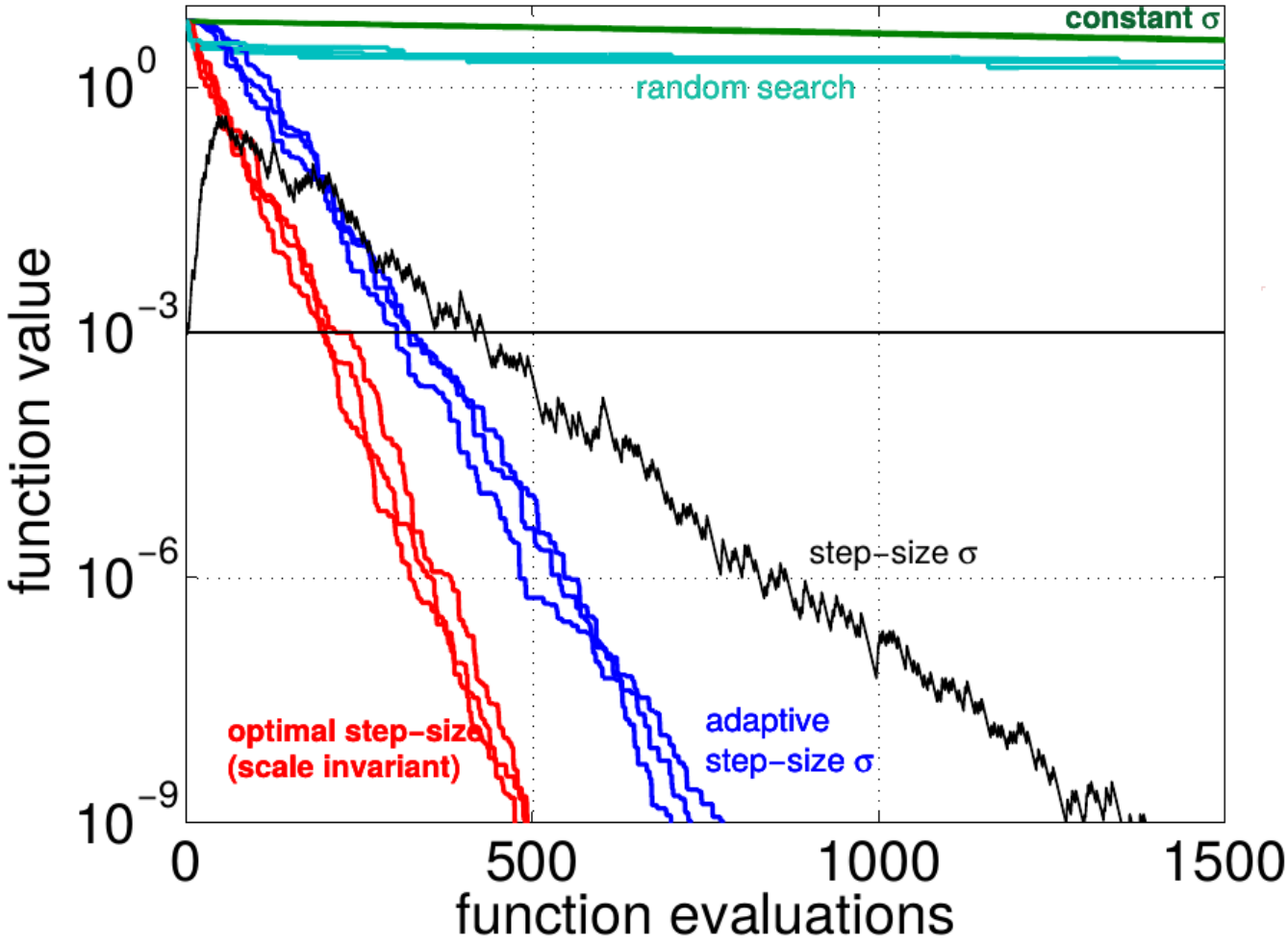
increase σ

Rationale

If successive moves are 'more correlated' than a random walk, step size should be increased

still far from the optimum

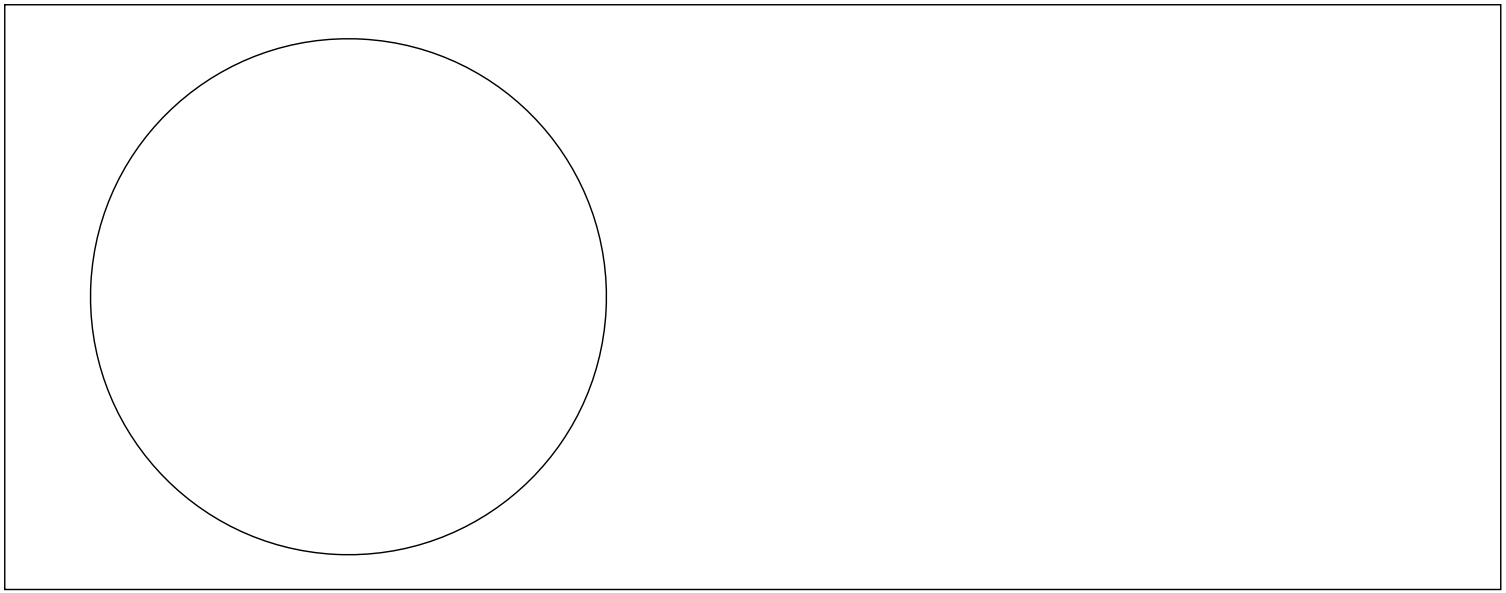
Step-size Adaptation at Work



Sphere function $f(x) = \sum x_i^2, n = 10$

Rank-One Update

$$\boldsymbol{m} \leftarrow \boldsymbol{m} + \sigma \boldsymbol{y}_w, \quad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i \, \boldsymbol{y}_{i:\lambda}, \quad \boldsymbol{y}_i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C})$$

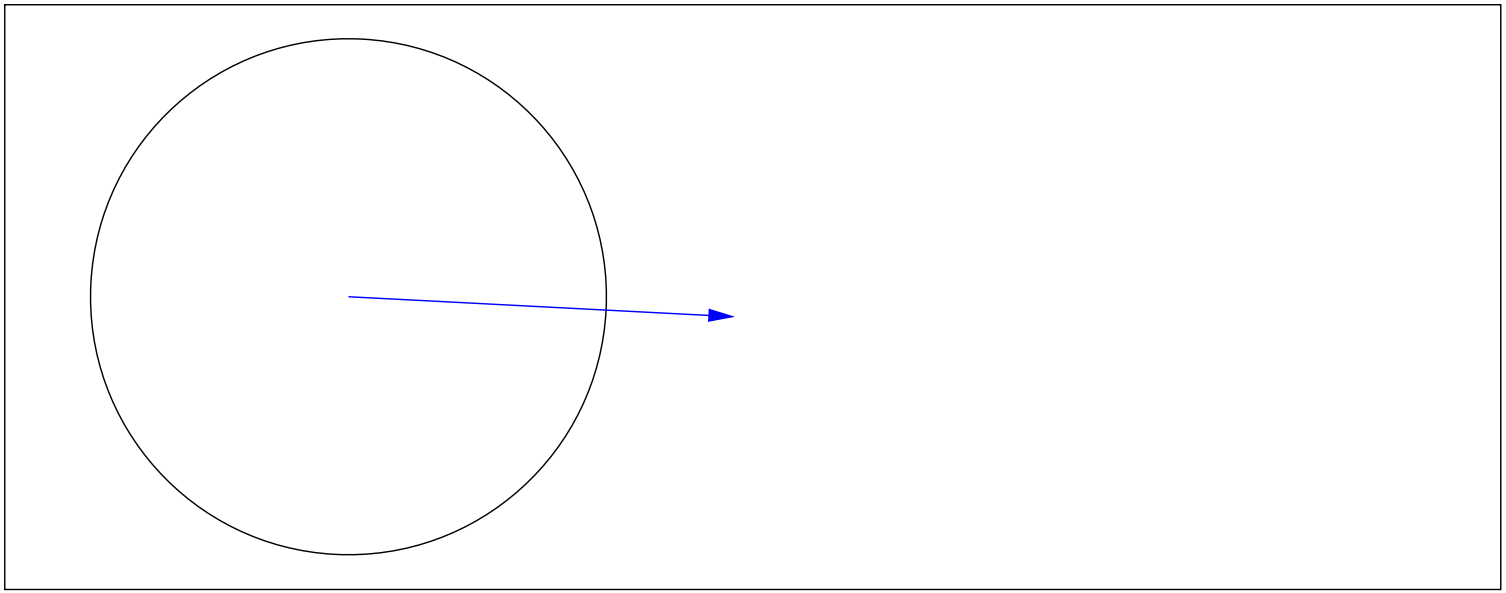


initial distribution, $\mathbf{C} = \mathbf{I}$

- new distribution: $\mathbf{C} \leftarrow 0.8 \times \mathbf{C} + 0.2 \times \boldsymbol{y}_w \boldsymbol{y}_w^{\mathrm{T}}$
- ruling principle: the adaptation increases the probability of successful steps, $\boldsymbol{y}_w$, to appear again

Rank-One Update

$$\boldsymbol{m} \leftarrow \boldsymbol{m} + \sigma \boldsymbol{y}_w, \quad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i \, \boldsymbol{y}_{i:\lambda}, \quad \boldsymbol{y}_i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C})$$

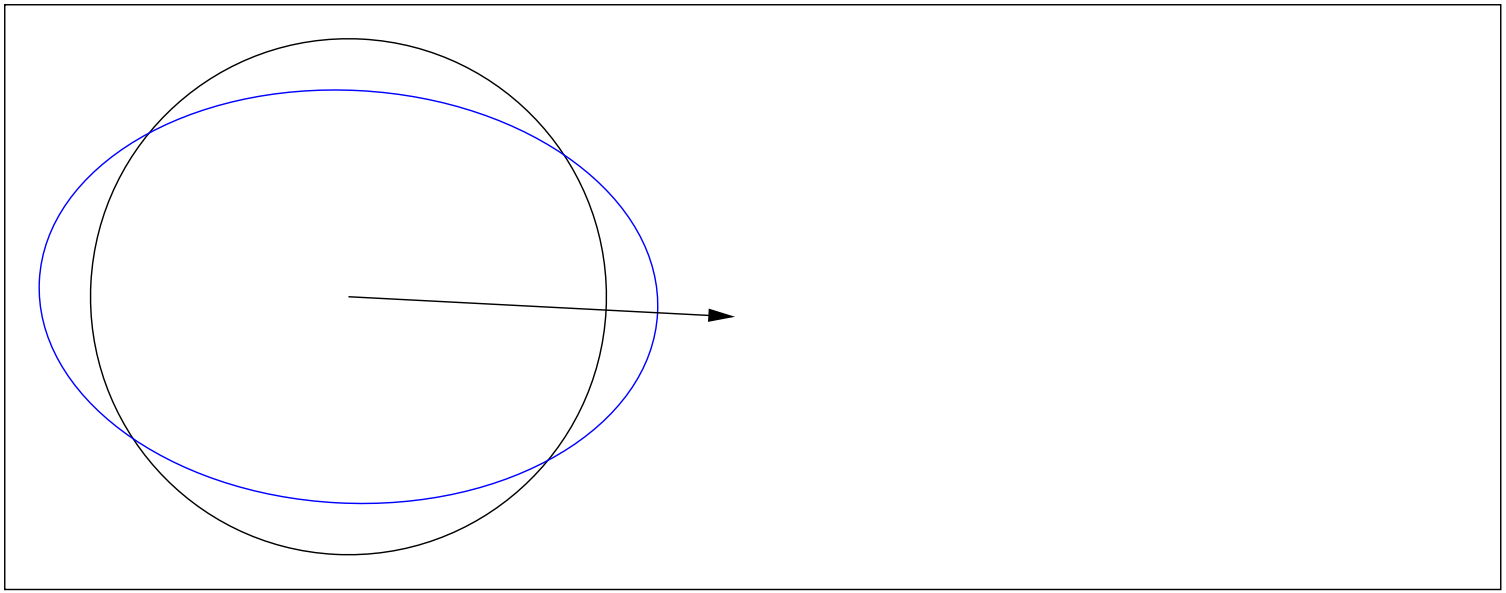


$\boldsymbol{y}_w$, movement of the population mean $\boldsymbol{m}$ (disregarding σ)

- new distribution: $\mathbf{C} \leftarrow 0.8 \times \mathbf{C} + 0.2 \times \boldsymbol{y}_w \boldsymbol{y}_w^{\mathrm{T}}$
- ruling principle: the adaptation increases the probability of successful steps, $\boldsymbol{y}_w$, to appear again

Rank-One Update

$$\boldsymbol{m} \leftarrow \boldsymbol{m} + \sigma \boldsymbol{y}_w, \quad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i \, \boldsymbol{y}_{i:\lambda}, \quad \boldsymbol{y}_i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C})$$

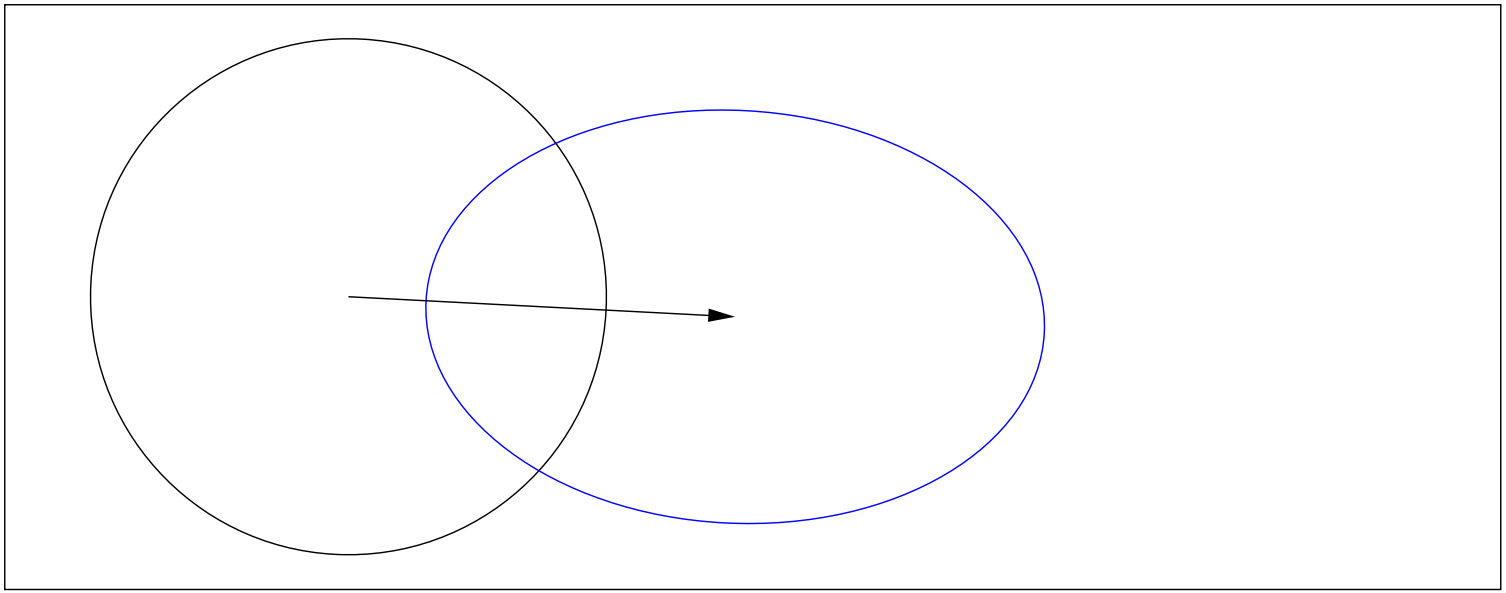


mixture of distribution $\mathbf{C}$ and step $\boldsymbol{y}_w$, $\mathbf{C} \leftarrow 0.8 \times \mathbf{C} + 0.2 \times \boldsymbol{y}_w \boldsymbol{y}_w^{\mathrm{T}}$

- new distribution: $\mathbf{C} \leftarrow 0.8 \times \mathbf{C} + 0.2 \times \boldsymbol{y}_w \boldsymbol{y}_w^{\mathrm{T}}$
- ruling principle: the adaptation increases the probability of successful steps, $\boldsymbol{y}_w$, to appear again

Rank-One Update

$$\boldsymbol{m} \leftarrow \boldsymbol{m} + \sigma \boldsymbol{y}_w, \quad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i \, \boldsymbol{y}_{i:\lambda}, \quad \boldsymbol{y}_i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C})$$

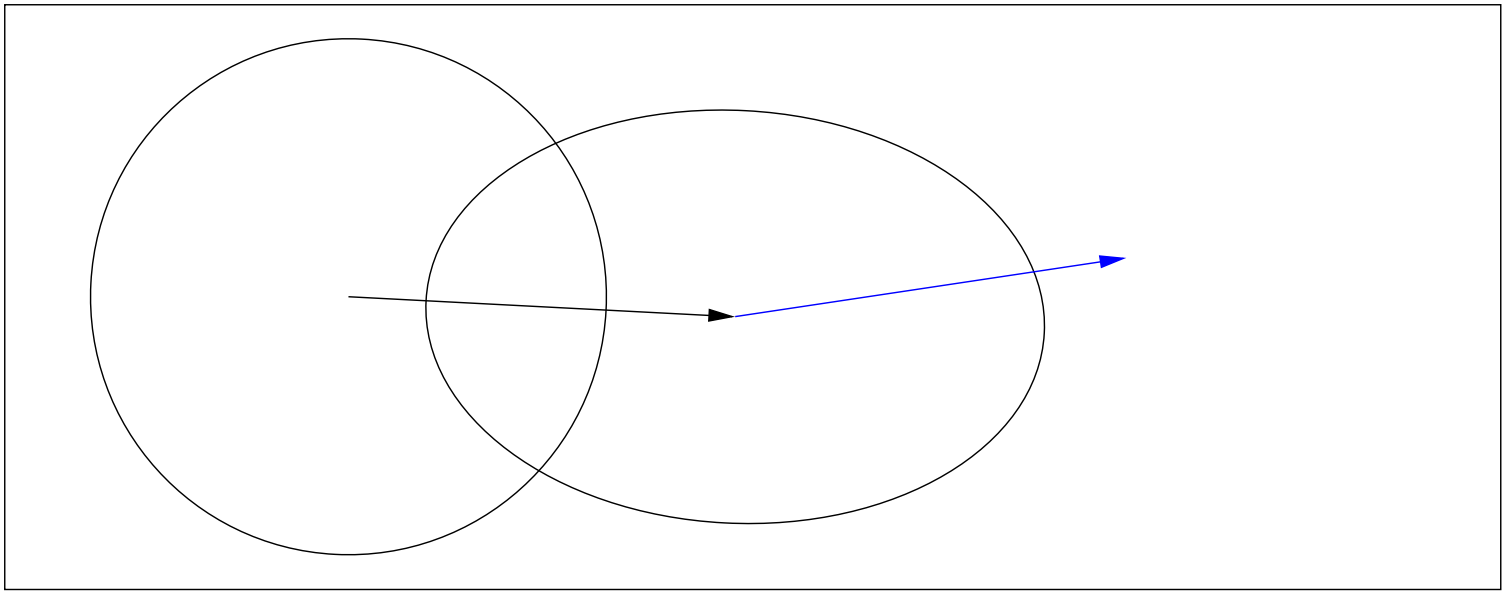


new distribution (disregarding σ)

- new distribution: $\mathbf{C} \leftarrow 0.8 \times \mathbf{C} + 0.2 \times \boldsymbol{y}_w \boldsymbol{y}_w^{\mathrm{T}}$
- ruling principle: the adaptation increases the probability of successful steps, $\boldsymbol{y}_w$, to appear again

Rank-One Update

$$\boldsymbol{m} \leftarrow \boldsymbol{m} + \sigma \boldsymbol{y}_w, \quad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i \, \boldsymbol{y}_{i:\lambda}, \quad \boldsymbol{y}_i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C})$$

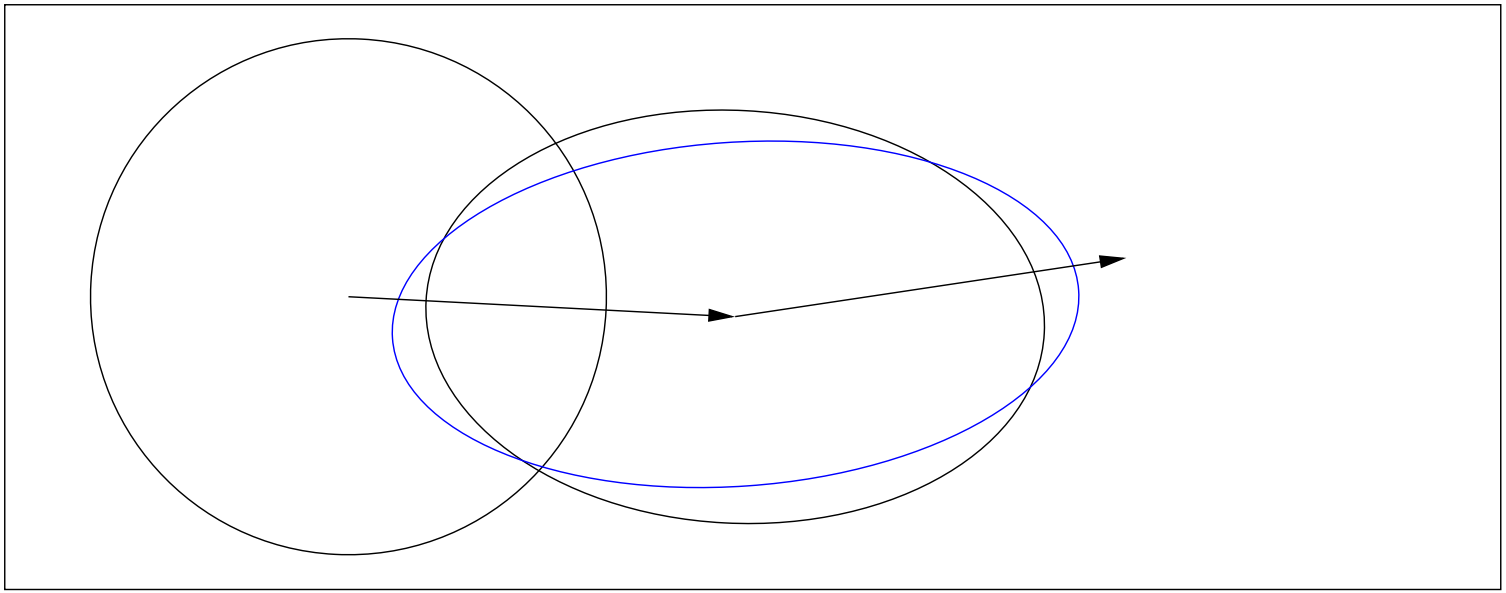


movement of the population mean $\boldsymbol{m}$

- new distribution: $\mathbf{C} \leftarrow 0.8 \times \mathbf{C} + 0.2 \times \boldsymbol{y}_w \boldsymbol{y}_w^{\mathrm{T}}$
- ruling principle: the adaptation increases the probability of successful steps, $\boldsymbol{y}_w$, to appear again

Rank-One Update

$$\boldsymbol{m} \leftarrow \boldsymbol{m} + \sigma \boldsymbol{y}_w, \quad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i \, \boldsymbol{y}_{i:\lambda}, \quad \boldsymbol{y}_i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C})$$

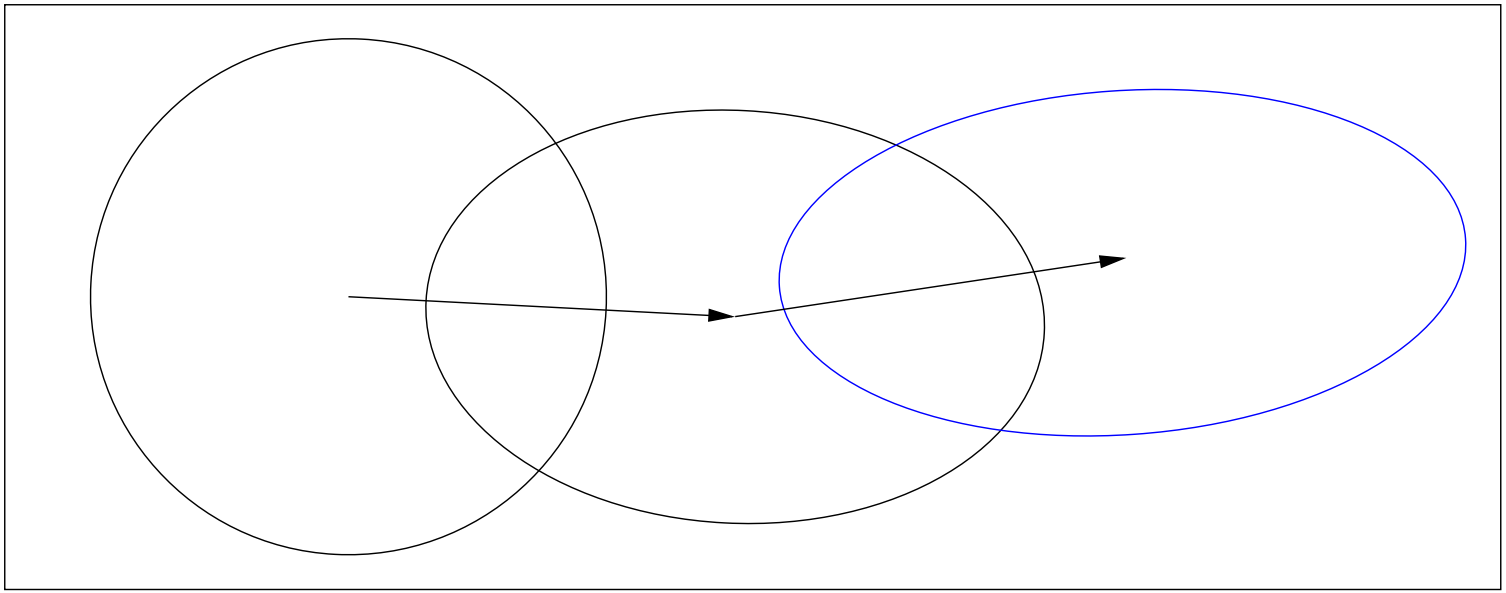


mixture of distribution $\mathbf{C}$ and step $\boldsymbol{y}_w$,
$\mathbf{C} \leftarrow 0.8 \times \mathbf{C} + 0.2 \times \boldsymbol{y}_w \boldsymbol{y}_w^{\mathrm{T}}$

- new distribution: $\mathbf{C} \leftarrow 0.8 \times \mathbf{C} + 0.2 \times \boldsymbol{y}_w \boldsymbol{y}_w^{\mathrm{T}}$
- ruling principle: the adaptation increases the probability of successful steps, $\boldsymbol{y}_w$, to appear again

Rank-One Update

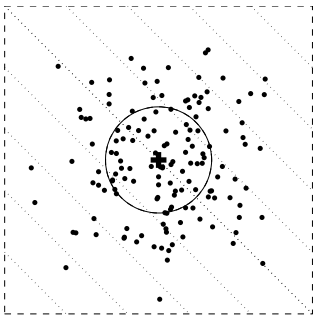
$$\boldsymbol{m} \leftarrow \boldsymbol{m} + \sigma \boldsymbol{y}_w, \quad \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i \, \boldsymbol{y}_{i:\lambda}, \quad \boldsymbol{y}_i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C})$$



- new distribution: $\mathbf{C} \leftarrow 0.8 \times \mathbf{C} + 0.2 \times \boldsymbol{y}_w \boldsymbol{y}_w^{\mathrm{T}}$
- ruling principle: the adaptation increases the probability of successful steps, $\boldsymbol{y}_w$, to appear again

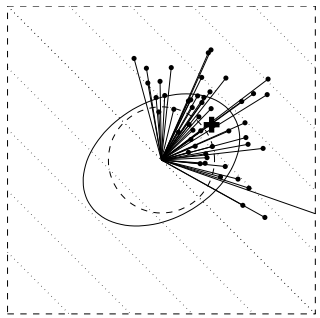
Rank-μ Update

$$\mathbf{x}_i = \boldsymbol{m} + \sigma\, \mathbf{y}_i, \qquad \mathbf{y}_i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C}),$$
$$\boldsymbol{m} \leftarrow \boldsymbol{m} + \sigma \mathbf{y}_w \qquad \mathbf{y}_w = \sum_{i=1}^{\mu} w_i\, \mathbf{y}_{i:\lambda}$$



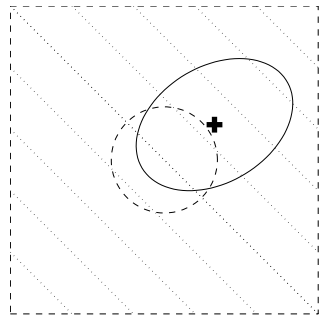
$\mathbf{x}_i = \boldsymbol{m} + \sigma\, \mathbf{y}_i, \quad \mathbf{y}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{C})$

sampling of $\lambda = 150$ solutions where $\mathbf{C} = \mathbf{I}$ and $\sigma = 1$



$\mathbf{C}_\mu = \frac{1}{\mu} \sum \mathbf{y}_{i:\lambda} \mathbf{y}_{i:\lambda}^{\mathrm{T}}$
$\mathbf{C} \leftarrow (1-1) \times \mathbf{C} + 1 \times \mathbf{C}_\mu$

calculating $\mathbf{C}$ from $\mu = 50$ points, $w_1 = \cdots = w_\mu = \frac{1}{\mu}$



$\boldsymbol{m}_{\text{new}} \leftarrow \boldsymbol{m} + \frac{1}{\mu} \sum \mathbf{y}_{i:\lambda}$

new distribution

Remark: the old (sample) distribution shape has a great influence on the new distribution $\longrightarrow$ iterations needed

CMA-ES in one page

Input: $\boldsymbol{m} \in \mathbb{R}^n$, $\sigma \in \mathbb{R}_+$, λ
Initialize: $\mathbf{C} = \mathbf{I}$, and $\boldsymbol{p}_\mathbf{c} = \mathbf{0}$, $\boldsymbol{p}_\sigma = \mathbf{0}$,
Set: $c_\mathbf{c} \approx 4/n$, $c_\sigma \approx 4/n$, $c_1 \approx 2/n^2$, $c_\mu \approx \mu_w/n^2$, $c_1 + c_\mu \leq 1$, $d_\sigma \approx 1 + \sqrt{\frac{\mu_w}{n}}$, and $w_{i=1\ldots\lambda}$ such that $\mu_w = \frac{1}{\sum_{i=1}^{\mu} w_i{}^2} \approx 0.3\,\lambda$

While not terminate

$$\mathbf{x}_i = \boldsymbol{m} + \sigma\,\boldsymbol{y}_i, \quad \boldsymbol{y}_i \sim \mathcal{N}_i(\mathbf{0}, \mathbf{C}), \quad \text{for } i = 1, \ldots, \lambda \qquad \text{sampling}$$

$$\boldsymbol{m} \leftarrow \sum_{i=1}^{\mu} w_i\,\mathbf{x}_{i:\lambda} = \boldsymbol{m} + \sigma\,\boldsymbol{y}_w \quad \text{where } \boldsymbol{y}_w = \sum_{i=1}^{\mu} w_i\,\boldsymbol{y}_{i:\lambda} \qquad \text{update mean}$$

$$\boldsymbol{p}_\mathbf{c} \leftarrow (1 - c_\mathbf{c})\,\boldsymbol{p}_\mathbf{c} + \mathbb{1}_{\{\|\boldsymbol{p}_\sigma\| < 1.5\sqrt{n}\}} \sqrt{1 - (1 - c_\mathbf{c})^2}\sqrt{\mu_w}\,\boldsymbol{y}_w \qquad \text{cumulation for } \mathbf{C}$$

$$\boldsymbol{p}_\sigma \leftarrow (1 - c_\sigma)\,\boldsymbol{p}_\sigma + \sqrt{1 - (1 - c_\sigma)^2}\sqrt{\mu_w}\,\mathbf{C}^{-\frac{1}{2}}\,\boldsymbol{y}_w \qquad \text{cumulation for } \sigma$$

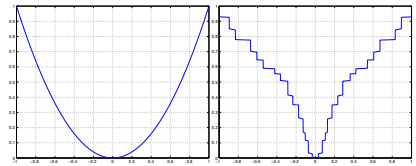
$$\mathbf{C} \leftarrow (1 - c_1 - c_\mu)\,\mathbf{C} + c_1\,\boldsymbol{p}_\mathbf{c}\,\boldsymbol{p}_\mathbf{c}{}^{\mathrm{T}} + c_\mu \sum_{i=1}^{\mu} w_i\,\boldsymbol{y}_{i:\lambda}\boldsymbol{y}_{i:\lambda}^{\mathrm{T}} \qquad \text{update } \mathbf{C}$$

$$\sigma \leftarrow \sigma \times \exp\left(\frac{c_\sigma}{d_\sigma}\left(\frac{\|\boldsymbol{p}_\sigma\|}{\mathrm{E}\|\mathcal{N}(\mathbf{0},\mathbf{I})\|} - 1\right)\right) \qquad \text{update of } \sigma$$

Not covered on this slide:
termination, restarts, active or mirrored sampling, outputs, and boundaries

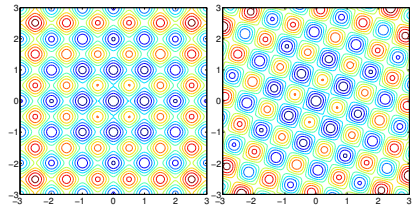
Invariances: Guarantee for Generalization

Invariance properties of CMA-ES



- Invariance to order preserving transformations in function space

  like all comparison-based algorithms



- Translation and rotation invariance

  to *rigid transformations* of the search space

CMA-ES is almost parameterless

- Tuning of a small set of functions — Hansen & Ostermeier 01
- Default values generalize to whole classes
- Exception: population size for multi-modal functions

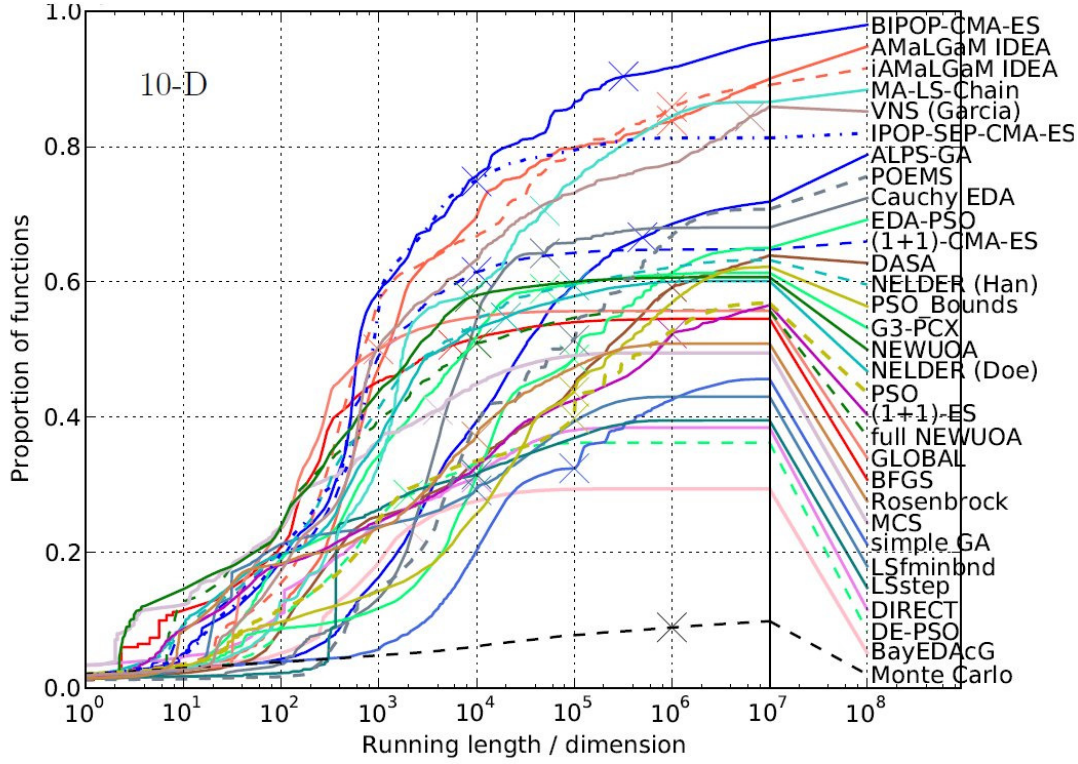
  but see IPOP-CMA-ES Auger & Hansen 05
  and BIPOP-CMA-ES Hansen 09

BBOB – Black-Box Optimization Benchmarking

- ACM-GECCO workshops ∼ yearly since 2009
- Set of 25 benchmark functions, dimensions 2 to 40 COCO platform
- With known difficulties (ill-conditioning, non-separability, . . .)
- Noisy and non-noisy versions

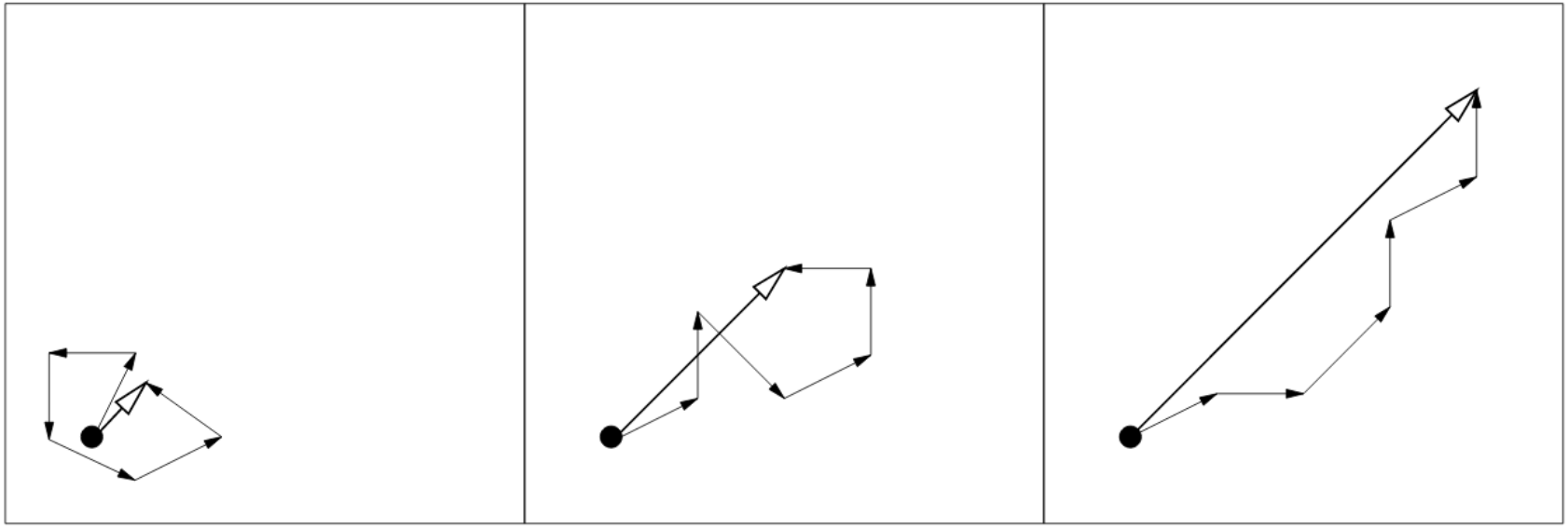
Competitors include

- BFGS (Matlab version),
- Fletcher-Powell,
- DFO (Derivative-Free Optimization, Powell 04)
- Differential Evolution
- Particle Swarm Optimization
- and many more



Speeding up: Agnostic learning rate adaptation

sketching the solution trajectory in 2d...



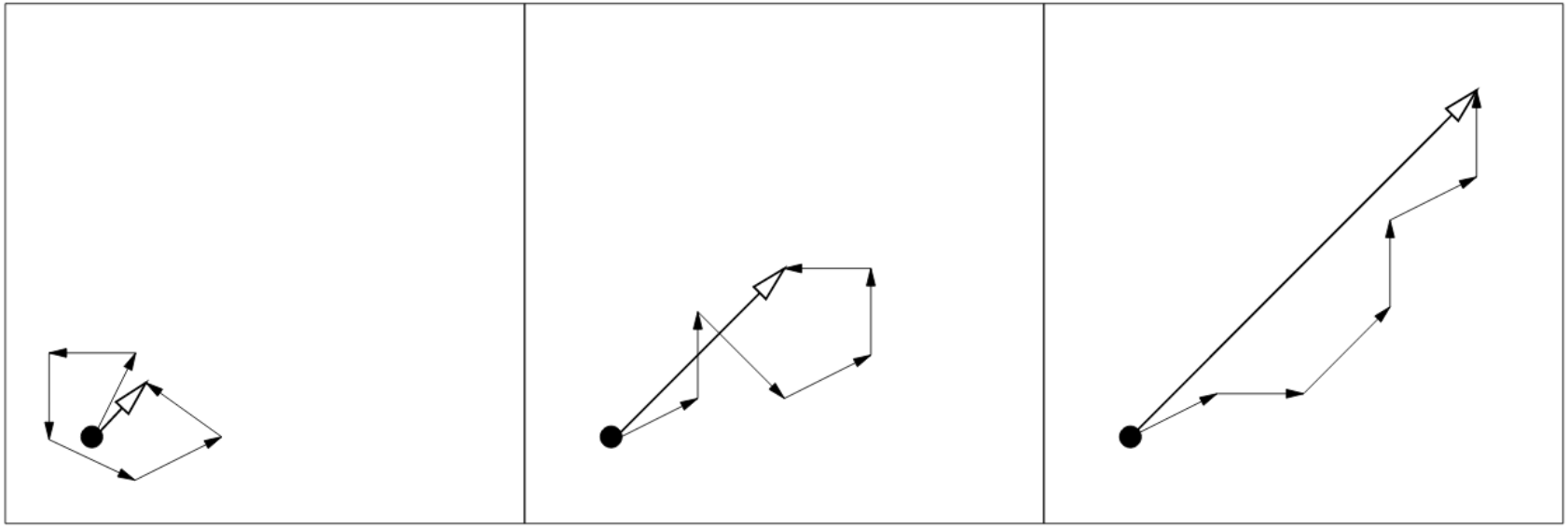
Intuition

If successive gradient directions are **more** correlated than random directions, learning rate can be **increased**.

Inspiration CMA-ES Hansen and Ostermeier, 01; Ollivier et al., 17

Speeding up: Agnostic learning rate adaptation

sketching the solution trajectory in 2d...



Intuition

If successive gradient directions are **less** correlated than random directions, learning rate can be **decreased**.

Inspiration CMA-ES Hansen and Ostermeier, 01; Ollivier et al., 17

Speeding up: Agnostic learning rate adaptation

Given memory parameter α

Define

- Gradient cumulative path

$$p_0 = 0$$
$$p_t = \alpha \frac{g_t}{\|g_t\|} + (1-\alpha)p_{t-1}$$

- Random cumulative path

$$\mathrm{r}_0 = 0$$
$$u_t \sim \mathbb{U}_{\mathcal{S}_d}$$
$$\mathrm{r}_t = \alpha u_t + (1-\alpha)\mathrm{r}_{t-1}$$

Speeding up: Agnostic learning rate adaptation

Expectation and std deviation of random cumulative path norm

$$\mathbb{E}(\|r_t\|^2) = \frac{\alpha}{2-\alpha}[1-(1-\alpha)^{2t}] \stackrel{def}{=} \mu(t)$$

$$std(\|r_t\|^2) = \frac{1}{d}\frac{2\alpha^2(1-\alpha)^2}{(2-\alpha)^2((1-\alpha)^2+1)}[1-(1-\alpha)^t][1-(1-\alpha)^{t-1}] \stackrel{def}{=} \sigma(d,t)$$

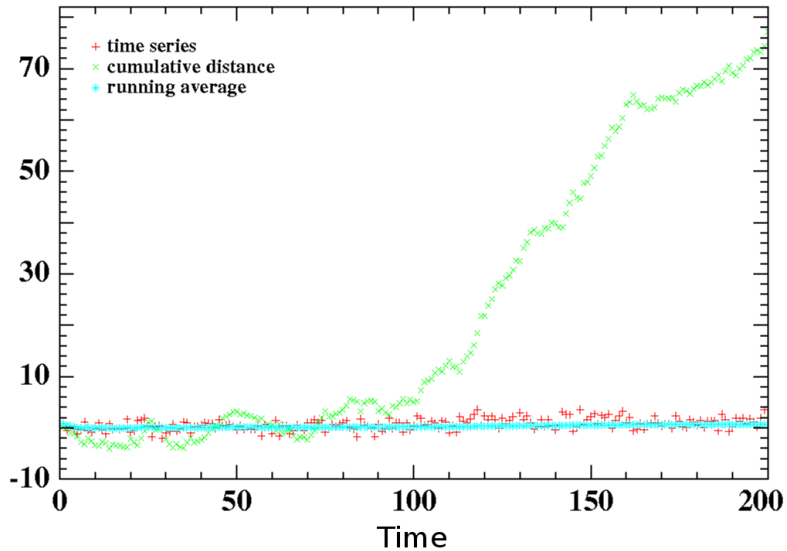
Agnostic momentum adaptation

$$\eta_{t+1} = \eta_t \exp\left(C\frac{\|p_t\|^2 - \mu(t)}{\sigma(d,t)}\right)$$

with C hyper-parameter

Change detection test

Example: given time series $\ell_{t<100} \sim \mathcal{N}(0,1),\ \ell_{t>100} \sim \mathcal{N}(1,1)$

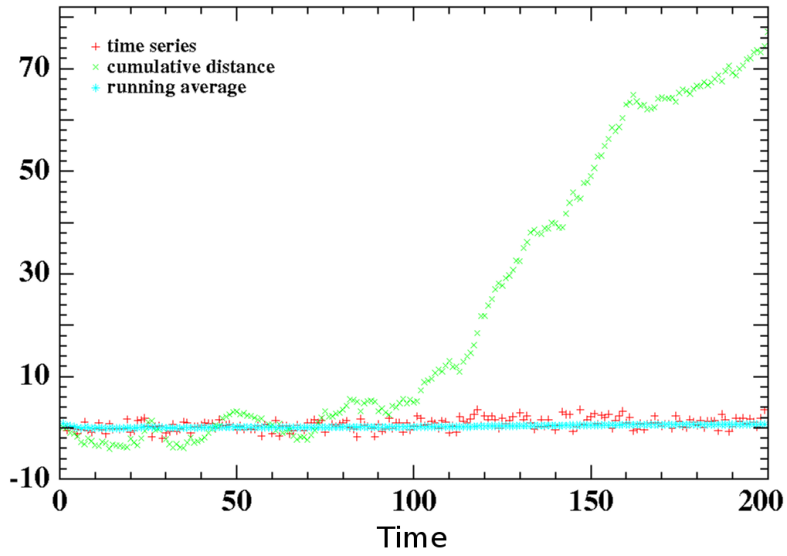


Page-Hinkley change detection test

Page 54; Hinkley 70

- Parameter Δ, alarm threshold
- Given a time series $(\ell_1, \ldots, \ell_t)$, compute
  - $\bar{\ell}_t = 1/t \sum_1^t \ell_i$
  - $L_t = \sum_1^t \ell_i - \bar{\ell}_i$
  - $L_{min} = min_i L_i, \qquad L_{max} = max_i L_i$
  - Test triggered if $\quad PH_t = \{L_t - L_{min} > \Delta\}$ increase detected
    $PH_t = \{L_{max} - L_t > \Delta\}$ decrease detected

Catastrophic event detection: Meeting the cliff



Page-Hinkley change detection test

Page 54; Hinkley 70

- $\bar{\ell}_t = 1/t \sum_1^t \ell_i$
- $L_t = \sum_1^t \ell_i - \bar{\ell}_i$
- $L_{min} = min_i L_i$
- $\mathrm{PH}_t = \{L_t - L_{min} > \Delta\}$

Monitoring the learning curve

- Signal: weighted average of mini-batch loss

$$\ell_t = \rho \mathcal{L}(\theta, \mathrm{MB}_t) + (1 - \rho)\ell_{t-1}$$

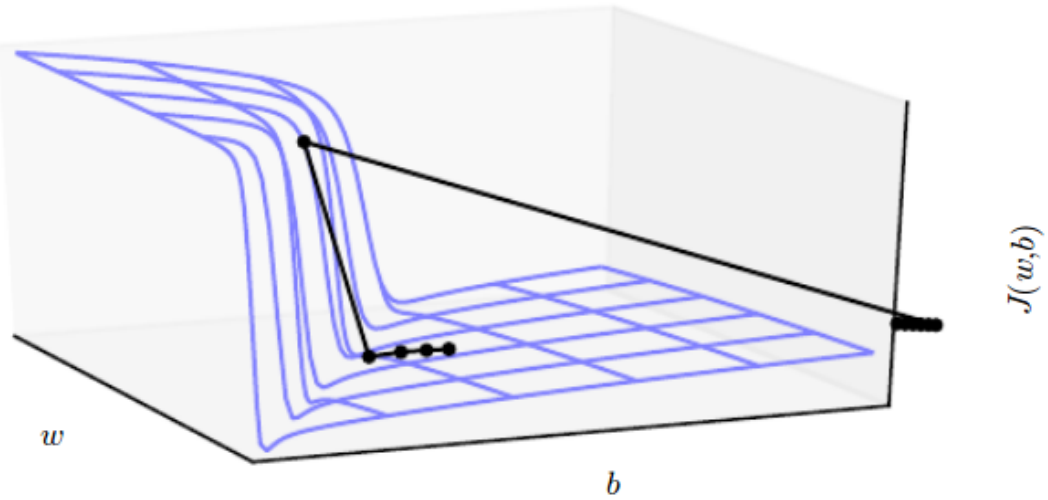
- Only monitor **loss increase**
- Alarm threshold

$$\Delta = \frac{1}{10}\mathcal{L}(\theta, \text{first mini-batch})$$

Catastrophic event curation

Usual catastrophic episodes

Goodfellow et al., 2016



Curation: If PH_t:

- Backtrack

$$\theta_{t+1} = \theta_t$$

- Halve learning rate

$$\eta_{t+1} = \frac{1}{2}\eta_t$$

S-ALeRA

Algorithm 1: S-ALeRA: *Agnostic LEarning Rate Adaptation* and Page-Hinkley change detection

Input: Model with loss function $\mathcal{L}$

Parameters : Memory rate α, factor C //algorithm parameters

Initial learning rate η_0, mini-batch ratio ρ // run parameters

Initialize : $t \leftarrow 0; p \leftarrow 0; \eta \leftarrow \eta_0$; init$(\theta)$ // initialization

$L, L_{min}, \ell, \bar{\ell}, \leftarrow 0; \Delta \leftarrow \mathcal{L}(\theta, \text{first mini-batch})/10$ // initialize Page-Hinkley

while *stopping criterion not met* **do**

  $MB \leftarrow$ new mini-batch ; $t \leftarrow t+1$ // perform forward pass

  $\ell = \rho\mathcal{L}(\theta, \text{MB}) + (1-\rho)\ell$

  $\bar{\ell} \leftarrow (\ell + t\bar{\ell})/(t+1)$ // empirical mean of batch losses

  $L \leftarrow L + (\ell - \bar{\ell})$ // cumulated deviations from mean

  $L_{min} \leftarrow min(L_{min}, L)$ // lower bound of deviations

  if $L - L_{min} > \Delta$ **then**

    $\theta \leftarrow \theta^{(b)}; \eta \leftarrow \eta/2$ // Page-Hinkley triggered: backtrack

    $L, L_{min}, \bar{\ell}, \ell \leftarrow 0; t \leftarrow 0$ // and re-initialize Page-Hinkley[2]

  else

    $\theta^{(b)} \leftarrow \theta$ // save for possible backtracks

    $g \leftarrow \nabla_\theta \mathcal{L}(\theta, \text{MB})$ // compute gradient with backward pass

    $p \leftarrow \alpha g/\|g\| + (1-\alpha)p$ // exponential moving average of normalized gradients

    $\eta \leftarrow \eta \exp\left(C(\|p\|_2^2 - \mu)/\sigma(d)\right)$ // agnostic learning rate update

    $\theta \leftarrow \theta - \eta g$ // standard parameter update

  end

end

Background

- Catastrophic events in Deep Learning
- SGD and learning rate adaptation

S-ALeRA

- Speeding up: Agnostic learning rate adaptation
- Catastrophic event management

Experiments

Experimental setting 1/2

- Datasets
  - MNIST — Le Cun et al., 1998
  - CIFAR-10 — Krizhevsky, 2009
- Four architectures, activation ReLU
  - M0: softmax classification (no hidden layers)
  - M2: 2 fully connected layers
  - M2b: M2 with batch normalization (BN) in each layer
  - M4: 2 convolutional layers followed by 2 fully connected layers (BN in each layer, max-pooling).
- Environment 46 GPUs; Torch — Collobert et al., 2011
- Avg classification error (mean and std. on 5 runs)

Experimental setting 2/2

- Baselines:
  - Nesterov Accelerated Gradient (NAG) $\gamma \in \{.8, .9, .99, .999\}$
  - ADAGRAD
  - ADAM $\beta_1 \in \{.8, .9, .99\}$, $\beta_2 \in \{.99, .999, .9999\}$
- New algorithms:
  - S-ALERA $\alpha \in \{.001, .01, .1, .25\}$, $C \in \{3.10^{-8}, 3.10^{-7}, 3.10^{-6}, 3.10^{-5}\}$
  - ALERA: S-ALERA w/o PH idem
  - AL-ADAM: ADAMwith ALERA idem, with $\beta_1 = 0.9$, $\beta_2 = 0.999$

Results - Learning performances

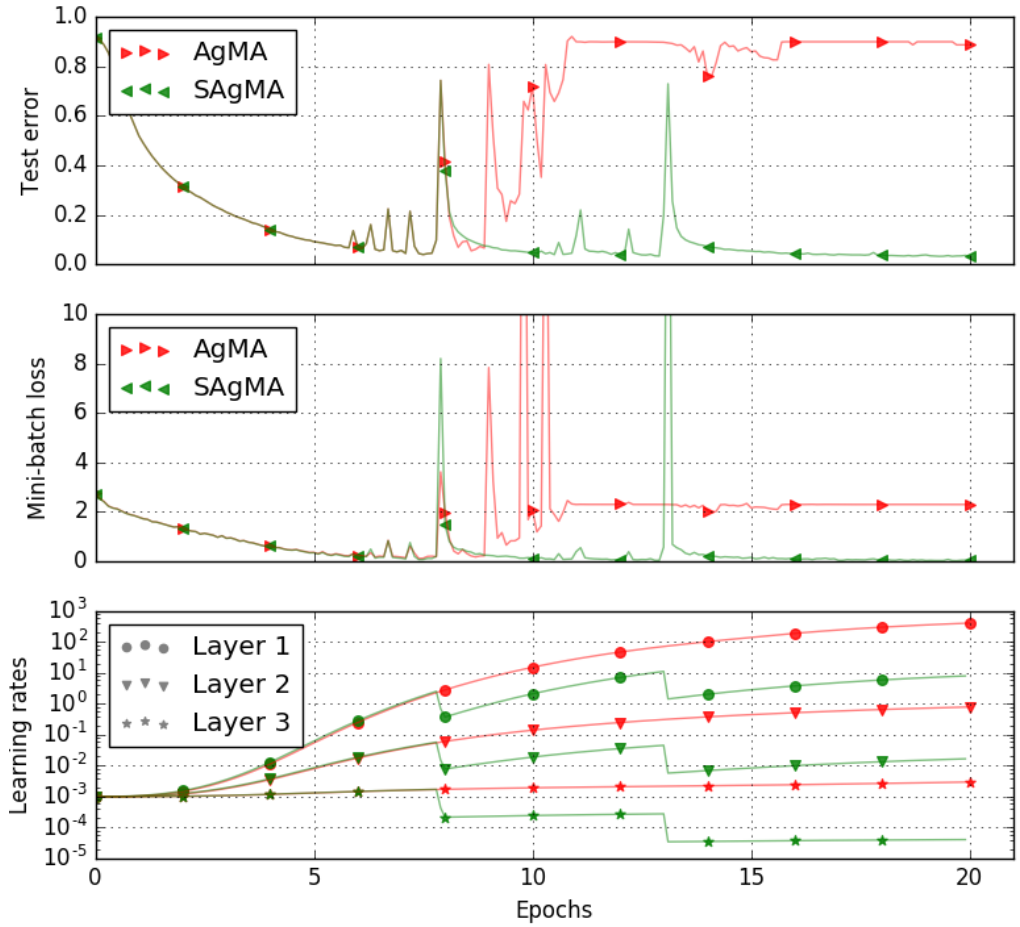
			NAG	Adagrad	ADAM	AG-ADAM	AGMA	S-AGMA
MNIST	M0	5ep.	7.75 (.14)	7.73 (.06)	7.72 (.12)	**7.31 (.09)**	7.51 (.09)	7.51 (.09)
		20ep.	7.59 (.07)	7.51 (.09)	7.43 (.10)	**7.29 (.09)**	7.43 (.03)	7.44 (.04)
	M2	5ep.	1.95 (.17)	2.00 (.06)	2.07 (.11)	1.93 (.17)	**1.86 (.11)**	1.87 (.05)
		20ep.	1.58 (.08)	1.71 (.08)	1.56 (.06)	1.57 (.04)	**1.55 (.10)**	1.59 (.09)
	M2b	5ep.	1.82 (.13)	1.72 (.07)	1.81 (.07)	1.66 (.08)	**1.59 (.08)**	**1.59 (.08)**
		20ep.	1.47 (.10)	1.48 (.06)	1.57 (.94)	1.53 (.05)	**1.43 (.04)**	1.48 (.09)
	M4b	5ep.	.85 (.08)	1.02 (.09)	.89 (.31)	.91 (.06)	**.82 (.30)**	**.82 (.30)**
		20ep.	.72 (.09)	.82 (.08)	.80 (.08)	.79 (.05)	**.63 (.05)**	.64 (.07)
CIFAR	M0	5ep.	60.37 (.55)	60.49 (.71)	60.60 (.45)	**59.62 (.27)**	59.89 (.19)	59.69 (.33)
		20ep.	59.73 (.19)	59.76 (.36)	59.81 (.24)	59.34 (.24)	**59.31 (.11)**	**59.31 (.25)**
	M2	5ep.	45.82 (.93)	44.81 (.62)	45.68 (.39)	44.91 (.42)	44.69 (17.58)	**44.42 (.24)**
		20ep.	45.08 (.32)	43.59 (.51)	44.43 (.50)	43.25 (.40)	43.19 (.21)	**42.72 (.41)**
	M2b	5ep.	45.01 (.84)	44.18 (.62)	44.30 (.96)	43.33 (.33)	**43.08 (.17)**	**43.08 (.17)**
		20ep.	42.50 (.48)	43.79 (.25)	43.60 (.64)	42.72 (.33)	**42.12 (.15)**	42.50 (.29)
	M4b	5ep.	27.74 (.48)	34.93 (.96)	28.50 (.68)	**25.60 (.29)**	28.61 (.50)	28.61 (.50)
		20ep.	27.45 (.39)	29.15 (.67)	27.84 (.59)	**25.30 (.18)**	26.35 (.64)	25.93 (.64)

Table 1: Best performances across all hyper-parameter settings at 5 and 20 epochs (average and standard deviation of test error on 5 runs). Bold: best result/line, red: less than 1 std.dev. away.

- AL-ADAM: outperforms ADAM
- S-ALERA: same as ALERA, less sensitive (less catastrophes)

Robust S-ALERA setting: $\alpha = .01,\ C = 3.10^{-6}$

ALeRA vs S-ALeRA on MNIST, M2, same seed



Comments First catastrophic episode around epoch 8: S-ALeRA halves the learning rate for all three layers.
Further catastrophic events met are more rare and less severe for S-ALeRA than ALeRA.

S-ALeRA eventually yields a better training loss and a considerably better test error than ALeRA.

Analysis of the dividing factor

Considering the 1D parabola problem

$$F(\theta) = \frac{1}{2} a \theta^2$$

Notations

- Optimal learning rate: $\eta^* = \frac{1}{a}$
- For $\eta > \eta_- = 2\eta^*$, loss increases

Phase 1: decreasing learning rate

Let ζ the dividing factor

1. Assume $\eta > \eta_-$ loss increase
2. Thus divide η by ζ until falling in $[\frac{\eta_-}{\zeta}, \eta_-[$

Number of divisions $S = \log(\frac{\eta}{\eta_-}) / \log(\zeta)$.

Phase 2: increasing learning rate

- Simplification:
  - If $\eta > \eta^*$, $\eta \to \eta\,(1 - \varepsilon)$
  - Else $\eta \to \eta\,(1 + \varepsilon)$

Analysis of the dividing factor, 2

Case 1, $\zeta < 2$

- At the end of Phase 1, $\eta > \eta^*$; hence, multiplied by $(1-\varepsilon)$ until reaching η^*.

Case 2, $\zeta > 2$

- At the end of Phase 1, η/η_- in $[\frac{1}{\zeta}, 1[$
- Case 2.1: η/η_- in $[\frac{1}{\zeta}, \frac{1}{2}]$
- Case 2.2: η/η_- in $[\frac{1}{2}, 1[$

Case 2.1: Expectation of time to reach η^*

- Probability $\frac{1}{2}\frac{\zeta-2}{\zeta-1}$
- Expectation of number of multiplications by $(1+\varepsilon)$

$$\begin{aligned} U &= \frac{1}{\varepsilon}\int_{\frac{1}{\zeta}}^{\frac{1}{2}} -\log(2u)du = \frac{1}{\varepsilon}[-t\log(t)+t-t\log(2)]_{\frac{1}{\zeta}}^{\frac{1}{2}} \\ &= \frac{1}{\varepsilon}\left(\frac{1}{2} - \frac{\log(\zeta)}{\zeta} - \frac{1-\log(2)}{\zeta}\right) \end{aligned}$$

Analysis of the dividing factor, 3

Case 2.2: Expectation of time to reach η^*

- Probability $\frac{1}{2}\frac{\zeta}{\zeta-1}$
- Expectation of number of multiplications by $(1-\varepsilon)$

$$\begin{aligned} T &= \tfrac{1}{\varepsilon}\int_{\frac{1}{2}}^{1}\log(2u)du = \tfrac{1}{\varepsilon}[t\log(t)-t+t\log(2)]_{\frac{1}{2}}^{1} \\ &= \tfrac{1}{\varepsilon}(\log(2)-\tfrac{1}{2}) \end{aligned}$$

Overall expected cost

c_C: catastrophic regime (forward pass)
c_S: standard regime (forward + backward pass)

$$c_C S + c_S\left(\frac{1}{2}\frac{\zeta-2}{\zeta-1}U + \frac{1}{2}\frac{\zeta}{\zeta-1}T\right)$$

Optimal ζ: minimizing

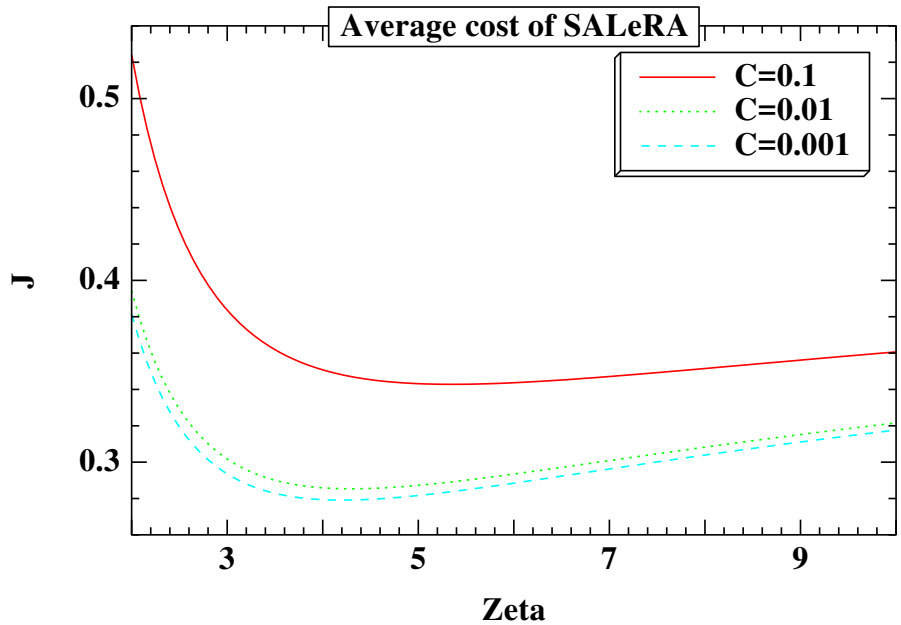
$$J(\zeta) = \frac{C}{\log(\zeta)} + \frac{1}{2}\frac{\zeta-2}{\zeta-1}\left(\frac{1}{2} - \frac{\log(\zeta)}{\zeta} - \frac{1-\log(2)}{\zeta}\right) + \frac{1}{2}\frac{\zeta}{\zeta-1}\left(\log(2)-\frac{1}{2}\right)$$

with

$$C = \frac{c_C}{c_S}\varepsilon\log\left(\frac{\eta}{\eta_-}\right)$$

Optimal dividing factor

Assuming C small ($c_C < c_S$, ε small)



Cost $J(\zeta)$ for different values of C: Optimal ζ in $[3, 5]$

Conclusions

- A principled way to address catastrophic episodes
  - Detection: Page-Hinkley change detection test
  - Curation: backtrack and halve learning rates
- Enables a more aggressive learning rate adaptation scheme
  - Comparing the cumulative gradient path to that of random unit vectors

Remarks

- PH does reduce the catastrophic event rate (test error $> 80\%$) by $\approx 40\%$

Paper: https://arxiv.org/abs/1709.01427
Code: https://github.com/lalil⊘u/salera/

Perspectives

- More datasets
- Extension to Recurrent Neural Networks
- Adapt PH threshold Δ during learning
- Consider other learning signals

Acknowledgements

- Steve Altschuler and Lani Wu
- Yann Ollivier
- Sigurd Angenent