Variable selection in continuous optimization: Some possible directions

Marc Schoenauer

Équipe TAO – INRIA Futurs – Orsay, France

Marc.Schoenauer@inria.fr

Overview

- Optimization vs Classification
- Evolution Strategies
  - Adaptive and self-adaptive Gaussian mutations
  - The Covariance Matrix Adaptation
  - Variable Selection using the Covariance Matrix?
- Feature selection in Machine Learning
  - A survey
  - Feature ranking with ROGER
- Conclusion

Optimization vs Classification

- We are interested in optimization problems
- Machine Learning and Data Mining have designed many methods for Feature Selection . . .
- for classification problems

So what?

Learnable Evolutionary Models

Michalski 99

- Evaluate the population
- Sort according to fitness
- Label as *Good* the best third, and as *Bad* the worst third
- Learn a classifier from those examples
- Generate next population by sampling the *Good* region

Can be viewed as an Estimation of Distribution Algorithm, that evolve a distribution on the search space.

Variable selection

Two possible directions:

- Use Data Mining tools for Feature Selection on the successive classification problems as defined in LEM
- Use the state-of-the-art Evolutionary Algorithm (CMA) that learns the Covariance Matrix of the objective function

Overview

- Optimization vs Classification
- Evolution Strategies
  - Adaptive and self-adaptive Gaussian mutations
  - The Covariance Matrix Adaptation
  - Variable Selection using the Covariance Matrix?
- Feature selection in Machine Learning
  - A survey
  - Feature ranking with ROGER
- Conclusion

Learning from examples

- Given a set of examples

$$(x_i, y_i) \in \mathbf{R}^d \times \{0,1\}$$

- Find an hypothesis H s.t.

$H(x_i) < 0$ if $y_i = 0$ and $H(x_i) > 0$ if $y_i = 1$

or minimizing $\sum (H(x_i) - y_i)^2$, or ...

- Such algorithm is called a learner.
- Well-known examples:
  - ID3, AQ15, ... — Symbolic learners
  - Neural networks, SVMs, /ldots — Numerical learners

Feature Selection: A hot topic

- Data are growing in size — in all directions :-)
- Genetic data, medical data, Web data, . . .
- Most learners do not scale up well with the number of features

Special Issue of *Journal of Machine Learning Research* on Feature Selection in 2003.

Feature Selection: a (very) brief survey

Still (almost) up-to-date:
M. Dash and H. Liu, Feature selection for classification, *Intelligent Data Analysis*, **1**(3), 1997.

Shift of paradigm

- Find the subset of features that gives the same empirical accuracy
  or does not decrease accuracy too much
- Find the optimal subset of size M w.r.t. accuracy
- Find the minimal size for a given accuracy

Feature Selection: methods

Whatever the target, it is a combinatorial problem

- Try all subsets — Does not scale up!
- Forward selection: add features one by one
- Backward selection: remove features one by one
- Stochastic: e.g. using Evolutionary Algorithms

In any case, need for a criterion

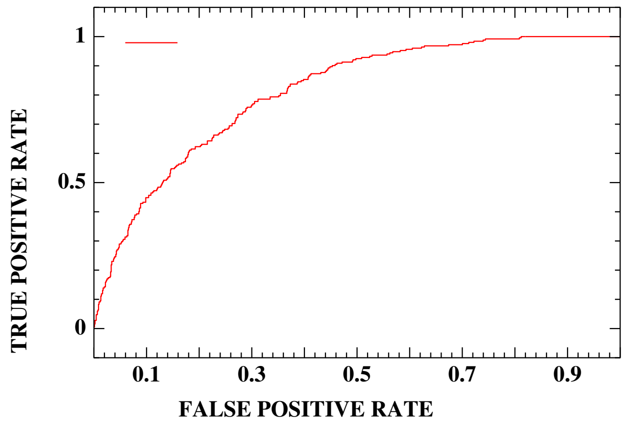
Feature Selection: Criteria

- Use a learner to compute accuracy — Wrapper method
  - Results depend on the learner
  - Costly
- Use a measure on the feature space
  - Entropy — The most discriminant feature w.r.t. class
  - Correlation — between features

ROGER

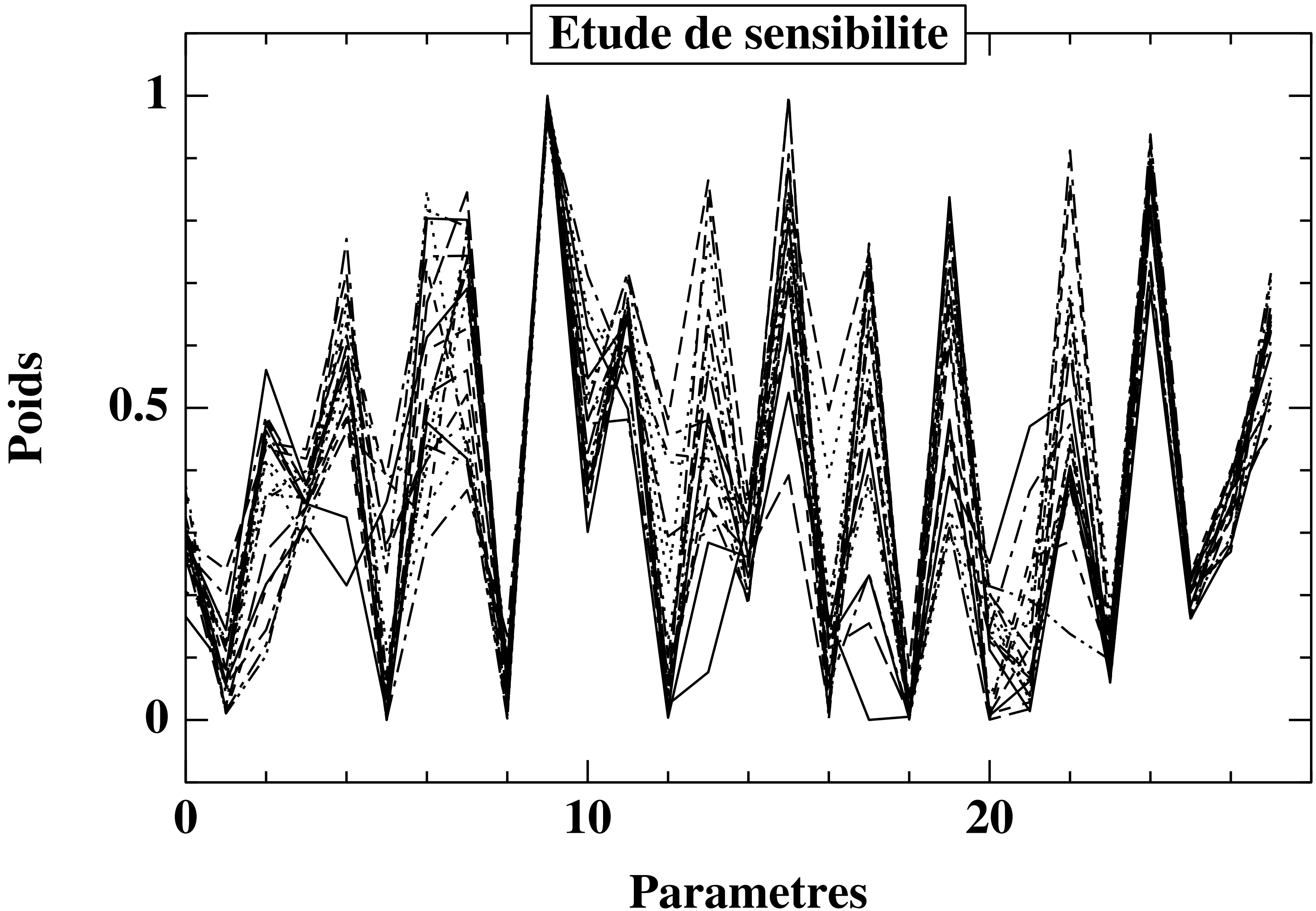
Sebag et al., 2003

- Does not try to “fit” the data



- Optimizes the Area under the ROC curve using . . . Evolution Strategies
- Look for ω_i and h_i s.t. $H(x) = \sum |w_i x_i - h_i|$ optimizes the ranking of the examples
- Look at the weights for each feature accross different evolutionary runs

Feature ranking with ROGER



Overview

- Optimization vs Classification
- Evolution Strategies
  - Adaptive and self-adaptive Gaussian mutations
  - The Covariance Matrix Adaptation
  - Variable Selection using the Covariance Matrix?
- Feature selection in Machine Learning
  - A survey
  - Feature ranking with ROGER
- Conclusion

Evolution Strategies

- μ parents
- generate λ offspring
- using normal mutations

$$X := X + \sigma\mathcal{N}(0, C)$$

- deterministically choose who will survive
  - Best μ among λ offspring $(\mu, \lambda) - ES$
  - Best μ among μ parents plus λ offspring $(\mu + \lambda) - ES$

Issue: Tune σ (the step-size) and C (the covariance matrix)

Adaptation of Gaussian mutation

History

- $\sigma \propto^{-1} t$ — Simulated annealing like
  Not adaptive
- $\sigma \propto^{-1}$ fitness — Early EP, difficult to tune
  Adaptive, individual
- The 1/5^{t}h rule: Modify σ w.r.t. # successful mutations
  Adaptive, population
- Self-adaptive mutations, allele or individual
- Covariance Matrix Adaptation — "Derandomized self-adaptation"
  Adaptive, population

Self-adaptive mutations

- Isotropic: One σ per variable, $C = I_d$

$$\begin{cases} \sigma := \sigma \ e^{\tau N_0(0,1)} \\ X_i := X_i + \sigma N_i(0,1) \ \ i = 1, \ldots, d \end{cases}$$

- Non -isotropic: d σ's per individual, $C =$ diag$(\sigma_1, \ldots, \sigma_d)$

$$\begin{cases} \kappa = \tau N_0(0,1) \\ \sigma_i := \sigma_i \ e^{\kappa + \tau' N_i(0,1)} \ \ i = 1, \ldots, d \\ X_i := X_i + \sigma_i N_i'(0,1) \ \ i = 1, \ldots, d \end{cases}$$

N_i and $N_i 1$ are independent

Self-adaptive mutations

- Correlated: C positive definite:

$$\vec{N}(0, C(\vec{\sigma}, \vec{\alpha})) = \prod_{i=1}^{d-1} \prod_{j=i+1}^{d} R(\alpha_{ij}) \vec{N}(0, \vec{\sigma})$$

$d(d-1)/2$ rotations

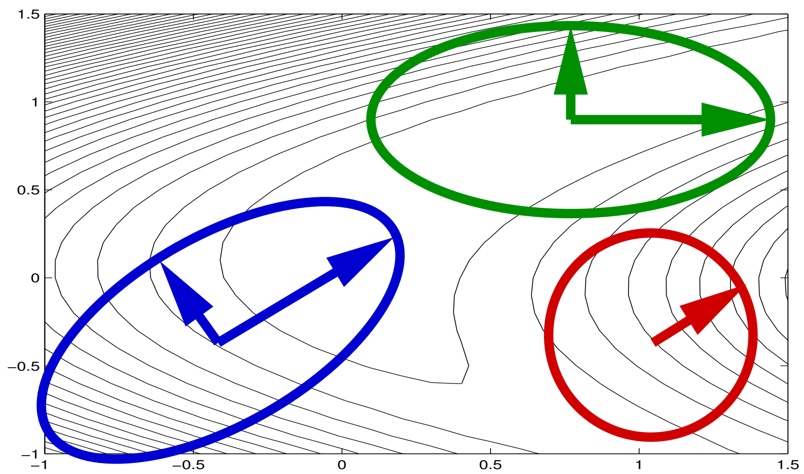
$$\begin{cases} \sigma_i = \sigma_i e^{\tau' N_0(0,1) + \tau N_i(0,1)} & i = 1, \ldots, d \\ \alpha_j := \alpha_j + \beta N_j(0,1) & j = 1, \ldots, d(d-1)/2 \\ \vec{X} := \vec{X} + \vec{N}(0, C(\vec{\sigma}, \vec{\alpha})) \end{cases}$$

- From Schwefel: $\tau \propto \frac{1}{\sqrt{2\sqrt{d}}}$, $\tau' \propto \frac{1}{\sqrt{2d}}$, $\beta = 0.0873$ (=5^o)

Isotropic mutation

Non–isotropic mutation

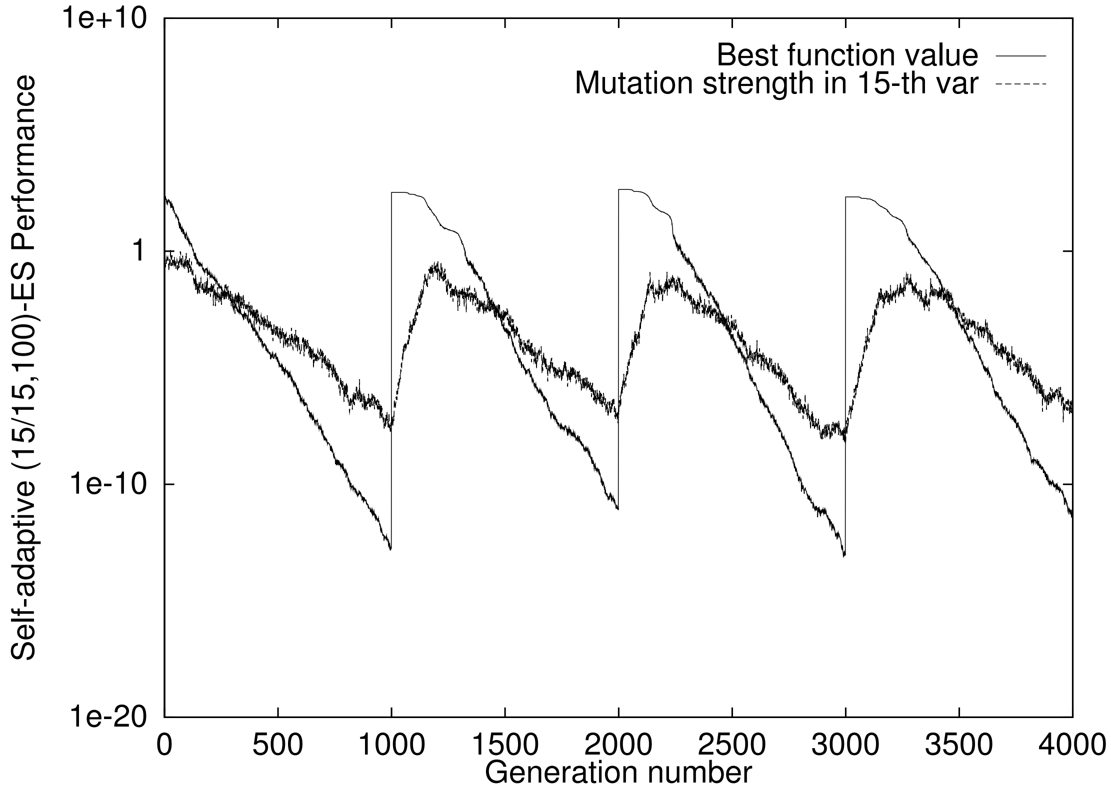
Correlated mutation



Evidence of self-adaptivity – SA-ES Deb & Beyer 01

Experiments on dynamic landscape

Slightly elliptic function with random moves of minimum every K generations



Fitness and σ_{15} for non-isotropic mutation

Self-adaptivity – discussion

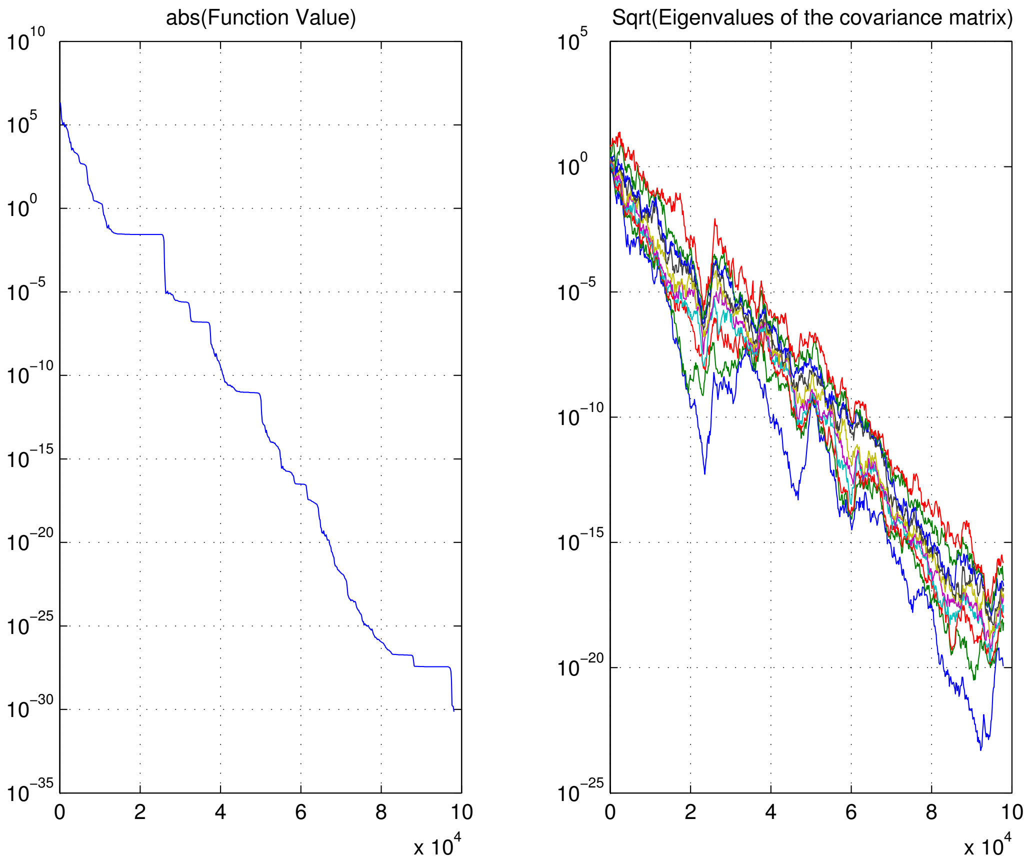
- Stability: Correlated mutation and DE require crossover
- Sensitivity to the characteristic basis of the fitness: Correlated mutation (and DE) perform poorly on rotated (elliptic) functions
- Speed: Adaptation can be very slow

But what covariance matrix should be learned?

Good reasons to believe it's $(\frac{1}{2}H)^{-1}$

H Hessian matrix of fitness

What does SA-ES learn?



$$f_H = \frac{1}{2} \sum_{i=1}^{n} (10^6)^{\frac{i-1}{n-1}} x_i^2$$

$$(\tfrac{1}{2} H)^{-1} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \vdots & 0 & 10^{-6} \end{pmatrix}$$

Fitness and square-root of eigenvalues for SA-ES on f_H $(n = 10)$

Back to adaptivity

Hansen & Ostermeier 96-01

- The actual path contains local information on the landscape
- It is lost through the self-adaptive mutation process

$\longrightarrow$ Derandomized Evolution Strategies

- Consecutive steps in colinear directions
  $\rightarrow$ increase step-size

  and vice-versa
- Add direction information to the covariance matrix
- Also, use a $(\mu/\mu, \lambda) - ES$ better with small populations Scheel, 85
  i.e. generate offspring from $<X>^{n+1} = \sum_{i=1}^{\mu} w_i X_{i:\lambda}^n$

  $X_{i:\lambda}^n, i = 1, \ldots, \mu$: best μ offspring from the λ mutations of $<X>^n$

Cumulative Step-size Adaptation

A $(\mu/\mu, \lambda) - ES$ with covariance matrix I_n — or $diag(\sigma_1, \ldots, \sigma_n)$

- Compute the cumulative path p^n using

$$p_\sigma^{n+1} = (1 - c_\sigma)p_\sigma^n + \sqrt{c_\sigma(2 - c_\sigma)}\frac{<X>^{n+1} - <X>^n}{\sigma^n}$$

- Update the step-size by — e.g. isotropic

$$\sigma^{n+1} = \sigma^n \exp\left(\frac{1}{d_\sigma}\left(\frac{||p_\sigma^{n+1}||}{E(||\mathcal{N}(0, I_d)||)} - 1\right)\right).$$

- Rationale:
  - if $p_\sigma^n \sim \mathcal{N}(0, I_d)$ and $\frac{<X>^{n+1} - <X>^n}{\sigma^n} \sim \mathcal{N}(0, I_d)$
    and they are independent,
    then $p_\sigma^{n+1} \sim \mathcal{N}(0, I_d)$
  - if there is no selection — e.g. $\lambda = \mu$
    then $\sigma^{n+1} = \sigma^n$ — Nothing should happen

Covariance Matrix Adaptation

Hansen & Ostermeier, 01

A $(\mu/\mu, \lambda) - ES$ with full covariance matrix C^n

- Update the (global) step-size
  - $p_\sigma^{n+1} = (1 - c_\sigma) p_\sigma^n + \sqrt{c_\sigma(2 - c_\sigma)} (C^n)^{-\frac{1}{2}} \frac{<X>_\mu^{n+1} - <X>_\mu^n}{\sigma^n}$
  - $\sigma^{n+1} = \sigma^n \exp\left(\frac{1}{d_\sigma}\left(\frac{||p_\sigma^{n+1}||}{E(||\mathcal{N}(0, I_d)||)} - 1\right)\right).$
- Rationale: idem CSA with a full covariance matrix C^n
- Note: $E[\|\mathcal{N}(0, I_d)\|] = \sqrt{2}\Gamma(\frac{n+1}{2})/\Gamma(\frac{n}{2})$ is approximated practically by $\sqrt{d}(1 - \frac{1}{4d} + \frac{1}{21d^2})$

CMA (2)

Hansen & Ostermeier, 01

- Update the Covariance Matrix: Rank 1 update
  - compute the cumulated path
$$p_c^{n+1} = (1 - c_c)p_c^n + \sqrt{c_c(2 - c_c)}\frac{<X>_\mu^{n+1} - <X>_\mu^n}{\sigma^n}.$$
  - $C^{n+1} = (1 - c_{\mathrm{cov}})C^n + c_{\mathrm{cov}} p_c^{n+1} p_c^{n+1\,T}$
- Rationale:
  - p_c^{n+1} is (roughly) the descent direction
  - C^n is updated with the rank 1 matrix $p_c^{n+1} p_c^{n+1\,T}$ whose eigenvector is p_c^{n+1}

- Use all μ best offspring to update C^n:
  - $U^{n+1} = \sum_{i=1}^{\mu} \frac{(X_{i:\lambda} - <X>_\mu^n)(X_{i:\lambda} - <X>_\mu^n)^T}{(\sigma^n)^2}$ Rank μ
  - $C^{n+1} = (1 - c_{\text{cov}})C^n + c_{\text{cov}}(\alpha_{\text{cov}} p_c^{n+1} p_c^{n+1\,T} + (1 - \alpha_{\text{cov}})U^{n+1})$
- Increase the speed of adaptation in high dimensions

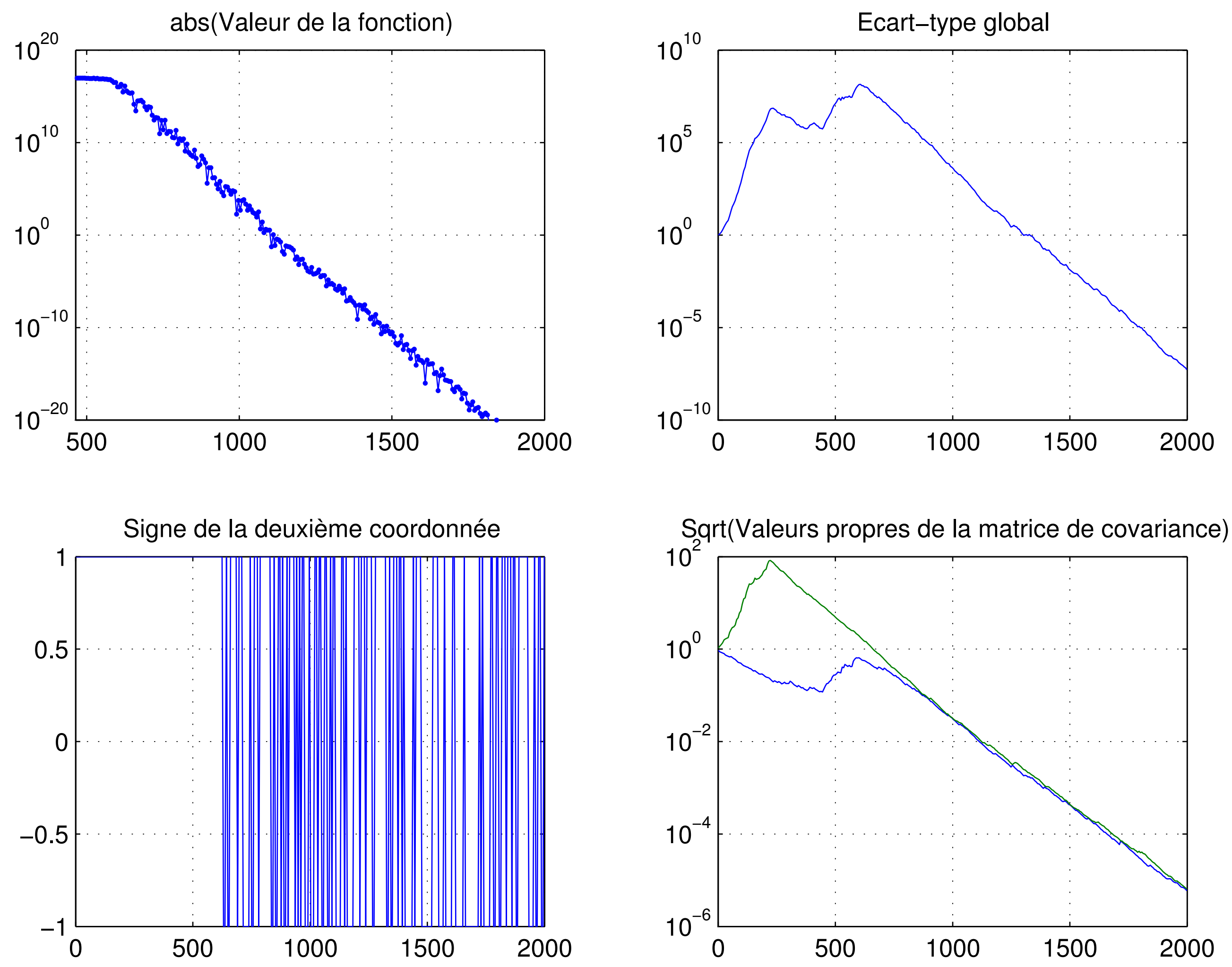
CMA parameters

$$c_c = \frac{4}{d+4},\ c_\sigma = \frac{10}{d+20},\ d_\sigma = \max(1, \frac{3\mu}{d+10}) + \frac{1}{c_\sigma}$$

$$c_{\text{cov}} = \frac{1}{\mu}\frac{2}{(d+\sqrt{2})^2} + (1 - \frac{1}{\mu})\min(1, \frac{2\mu - 1}{(d+2)^2 + \mu})$$

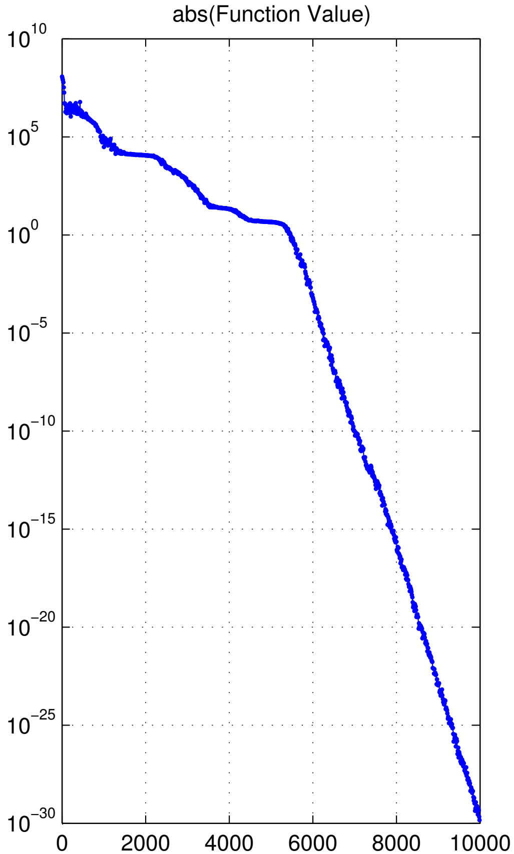
with initial values $p_\sigma^0 = 0$, $p_c^0 = 0$ and $C^0 = I_d$.

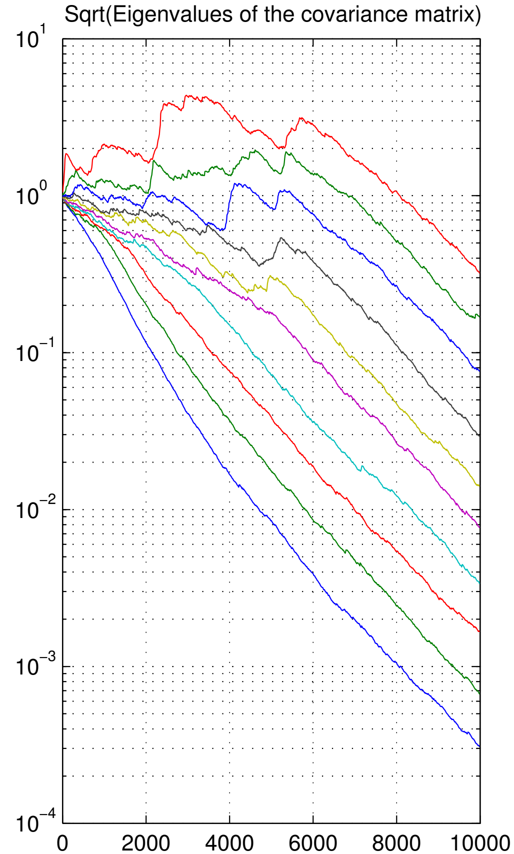
CMA-ES at work



Sphere function, $n = 2$, initial point $(0, 10^9)$.
Fitness, (global) step-size, sign(x_2) and sqrt(eigenvalues)

What does CMA-ES learn?





$$f_H = \frac{1}{2}\sum_{i=1}^{n}(10^6)^{\frac{i-1}{n-1}} x_i^2$$

$$(\tfrac{1}{2}H)^{-1} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \vdots & 0 & 10^{-6} \end{pmatrix}$$

Fitness and square-root of eigenvalues for CMA-ES on f_H $(n = 10)$

Toward variable selection

Idea

- Once the covariance matrix has been learned
- select the eigenvectors with the smallest eignevalues

But

- How good is the approximation? Error criterion Auger, PhD 04
- What threshold? Cross-validate with other measures

  Entropy, covariance, . . .
- A moving target The interesting variables might change along evolution

Conclusion

Nothing yet