Bridging Optimization over Manifolds and Evolutionary Computation

Luigi Malagò

Romanian Institute of Science and Technology - RIST
malago@rist.ro

PPSN 2016, Edinburgh September 17, 2016

Outline of the Tutorial

A gentle introductions to

- Optimization over manifolds: Riemannian optimization
- Geometry(-ies) of statistical models: Information Geometry

Outline of the Tutorial

A gentle introductions to

- Optimization over manifolds: Riemannian optimization
- Geometry(-ies) of statistical models: Information Geometry

Relevant applications in Evolutionary Computation

- A geometric framework for model-based meta-heuristics: Stochastic Relaxation
- Generalizations of population-based meta-heuristics: Riemannian PSOs

"One geometry cannot be more true than another; it can only be more convenient". Henri Poincaré, Science and Hypothesis, 1902.

Optimization Over Manifolds

Riemannian optimization refers to the optimization of a cost function defined over a manifold

$$f : \mathcal{M} \to \mathbb{R}$$

Optimization Over Manifolds

Riemannian optimization refers to the optimization of a cost function defined over a manifold

$$f : \mathcal{M} \to \mathbb{R}$$

Informally, a manifold $\mathcal{M}$ is a non-linear space that generalizes the notion of a Euclidean vector space, since it admits a structure that looks "locally" Euclidean

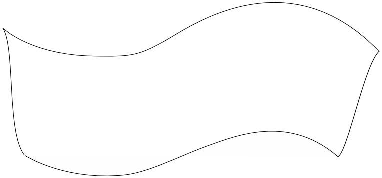
Optimization Over Manifolds

Riemannian optimization refers to the optimization of a cost function defined over a manifold

$$f : \mathcal{M} \to \mathbb{R}$$

Informally, a manifold $\mathcal{M}$ is a non-linear space that generalizes the notion of a Euclidean vector space, since it admits a structure that looks "locally" Euclidean

Intuitively, think to lower-dimensional surface embedded in $\mathbb{R}^n$



Why Manifolds?

Manifolds appear naturally whenever we have some symmetry or invariance properties in the cost function or in the constraints

They play an important role in linear algebra, signal processing, robotics, machine learning, statistics, and many other fields

Why Manifolds?

Manifolds appear naturally whenever we have some symmetry or invariance properties in the cost function or in the constraints

They play an important role in linear algebra, signal processing, robotics, machine learning, statistics, and many other fields

In general, by taking into account the structure of the problem, more efficient numerical procedures can be developed

A mathematical framework for manifold optimization provides the basis for convergence analysis of the optimization algorithms

Optimization Algorithms on Matrix Manifolds
P.-A. Absil, R. Mahony, and R. Sepulchre
Princeton University Press, 2008

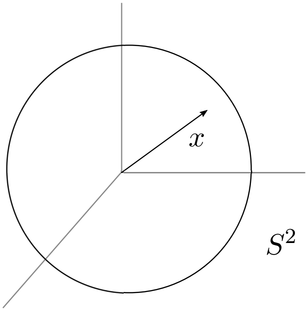
Some Examples of Manifolds

- The n–sphere
- The torus
- The set of rotation matrices
  $SO_3 = \{R : RR^{\mathrm{T}} = 1 \wedge \det(R) = 1\}$
- The Special Euclidean group $SE_3 = SO_3 \times \mathbb{R}^3$
- The cone of positive definite matrices
- The set of rank-k matrices
- The Gaussian distribution and more in general any exponential family

The n-sphere

The n-sphere is one of the simplest examples of manifold

Its structure arises for example by imposing a normalization constraint on a Euclidean vector space



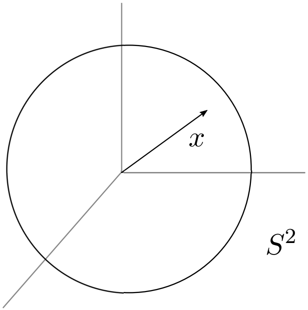
On the n-sphere $\|\boldsymbol{x}\| = 1$

E.g., the space of the eigenvectors of a matrix

The n-sphere

The n-sphere is one of the simplest examples of manifold

Its structure arises for example by imposing a normalization constraint on a Euclidean vector space



On the n-sphere $\|\boldsymbol{x}\| = 1$

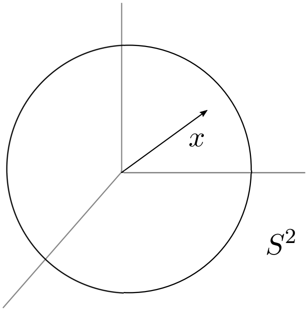
E.g., the space of the eigenvectors of a matrix

S^n is a n-dimensional manifold, which can be embedded in $\mathbb{R}^{n+1}$

The n-sphere

The n-sphere is one of the simplest examples of manifold

Its structure arises for example by imposing a normalization constraint on a Euclidean vector space



On the n-sphere $\|\boldsymbol{x}\| = 1$

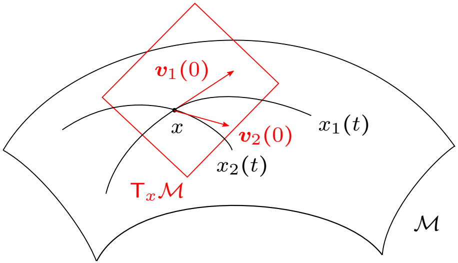
E.g., the space of the eigenvectors of a matrix

S^n is a n-dimensional manifold, which can be embedded in $\mathbb{R}^{n+1}$

The Tangent Space

To implement first-order calculus, we need a differentiable structure

This is obtained by defining a tangent bundle $\mathsf{T}\mathcal{M}$, i.e., the set of tangent spaces $\mathsf{T}_p\mathcal{M}$ for all $p \in \mathcal{M}$



Intuitively tangent spaces can be identified by the set the velocity vectors to all smooth curves passing through x

Riemannian Metric

The tangent space is a vector space for which we can define an inner product called Riemannian metric

$$g(\boldsymbol{v}, \boldsymbol{w}) = \langle \boldsymbol{v}, \boldsymbol{w} \rangle_p : \mathsf{T}_p\mathcal{M} \times \mathsf{T}_p\mathcal{M} \to \mathbb{R}$$

Riemannian Metric

The tangent space is a vector space for which we can define an inner product called Riemannian metric

$$g(\boldsymbol{v}, \boldsymbol{w}) = \langle \boldsymbol{v}, \boldsymbol{w} \rangle_p : \mathsf{T}_p\mathcal{M} \times \mathsf{T}_p\mathcal{M} \to \mathbb{R}$$

The inner product induces a norm

$$\|\boldsymbol{v}\|_p = \sqrt{\langle \boldsymbol{v}, \boldsymbol{v} \rangle_p}$$

Riemannian Metric

The tangent space is a vector space for which we can define an inner product called Riemannian metric

$$g(\boldsymbol{v}, \boldsymbol{w}) = \langle \boldsymbol{v}, \boldsymbol{w} \rangle_p : \mathsf{T}_p\mathcal{M} \times \mathsf{T}_p\mathcal{M} \to \mathbb{R}$$

The inner product induces a norm

$$\|\boldsymbol{v}\|_p = \sqrt{\langle \boldsymbol{v}, \boldsymbol{v} \rangle_p}$$

The inner product can be used to measure the length of a curve $x(t)$ with $t \in [a, b]$

$$L(x(t)) = \int_a^b \sqrt{\langle \dot{\boldsymbol{x}}(t), \dot{\boldsymbol{x}}(t) \rangle_p}\, \mathrm{d}t$$

Riemannian Metric

The tangent space is a vector space for which we can define an inner product called Riemannian metric

$$g(\boldsymbol{v}, \boldsymbol{w}) = \langle \boldsymbol{v}, \boldsymbol{w} \rangle_p : \mathsf{T}_p\mathcal{M} \times \mathsf{T}_p\mathcal{M} \to \mathbb{R}$$

The inner product induces a norm

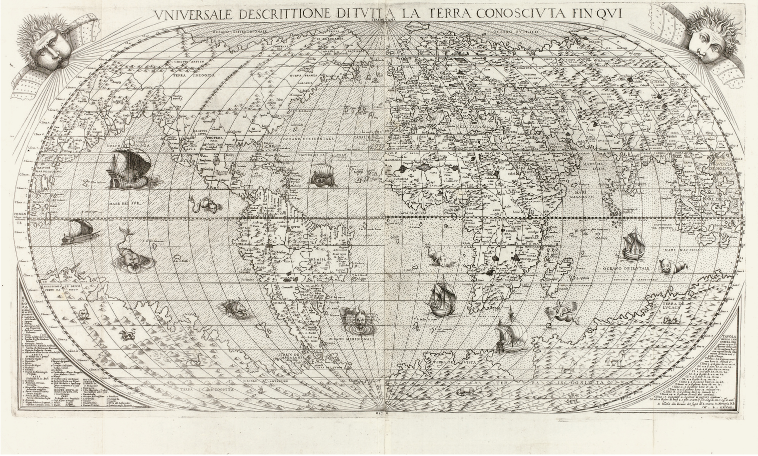
$$\|\boldsymbol{v}\|_p = \sqrt{\langle \boldsymbol{v}, \boldsymbol{v} \rangle_p}$$

The inner product can be used to measure the length of a curve $x(t)$ with $t \in [a, b]$

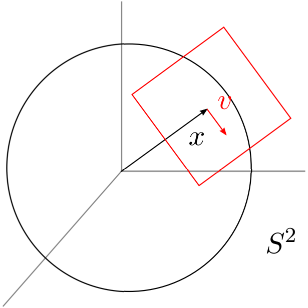
$$L(x(t)) = \int_a^b \sqrt{\langle \dot{\boldsymbol{x}}(t), \dot{\boldsymbol{x}}(t) \rangle_p}\, \mathrm{d}t$$

Geodesics are length minimizing curves between two points

Tangent Space of the n-sphere



Tangent Space of the n-sphere



The tangent space $\mathsf{T}_x\mathcal{M}$ is given by all orthogonal vectors, i.e.,

$$\{\boldsymbol{v} \in \mathbb{R}^n \text{ such that } \boldsymbol{v}^{\mathrm{T}}\boldsymbol{x} = 0\}$$

The inner product inherited by the embedding the Euclidean space is the standard inner product on $\mathbb{R}^n$

Geodesics are the great circles of the sphere

Riemannian Gradient

Let $f(x) : \mathcal{M} \mapsto \mathbb{R}$ be a smooth function over $(\mathcal{M}, g)$

Riemannian Gradient

Let $f(x) : \mathcal{M} \mapsto \mathbb{R}$ be a smooth function over $(\mathcal{M}, g)$

For each vector field v over $\mathcal{M}$, the Riemannian gradient of $f(x)$, i.e., the direction of steepest ascent is the unique vector that satisfies

$$g(\operatorname{grad} f, v) = \mathrm{D}_v\, f,$$

where $\mathrm{D}_v\, f$ is the directional derivative of f in the direction v

Riemannian Gradient

Let $f(x) : \mathcal{M} \mapsto \mathbb{R}$ be a smooth function over $(\mathcal{M}, g)$

For each vector field v over $\mathcal{M}$, the Riemannian gradient of $f(x)$, i.e., the direction of steepest ascent is the unique vector that satisfies

$$g(\operatorname{grad} f, v) = \mathrm{D}_v\, f,$$

where $\mathrm{D}_v\, f$ is the directional derivative of f in the direction v

Given a coordinate system ξ for $\mathcal{M}$ we have

$$\operatorname{grad} f(\xi) = \sum_{i,j=1}^{d} g^{ij} \frac{\partial f_\xi}{\partial \theta_i} \frac{\partial}{\partial \theta_j} = G_\xi(\xi)^{-1} \nabla f_\xi(\xi)$$

Riemannian Gradient

Let $f(x) : \mathcal{M} \mapsto \mathbb{R}$ be a smooth function over $(\mathcal{M}, g)$

For each vector field v over $\mathcal{M}$, the Riemannian gradient of $f(x)$, i.e., the direction of steepest ascent is the unique vector that satisfies

$$g(\operatorname{grad} f, v) = \mathrm{D}_v\, f,$$

where $\mathrm{D}_v\, f$ is the directional derivative of f in the direction v

Given a coordinate system ξ for $\mathcal{M}$ we have

$$\operatorname{grad} f(\xi) = \sum_{i,j=1}^{d} g^{ij} \frac{\partial f_\xi}{\partial \theta_i} \frac{\partial}{\partial \theta_j} = G_\xi(\xi)^{-1} \nabla f_\xi(\xi)$$

The Riemannian gradient depends on the metric g trough $G = [g_{ij}]$

First-order Optimization: Riemannian Gradient Descent

Consider the Euclidean naïve implementation of gradient descent over a manifold

$$x_{t+1} = x_t - \lambda \operatorname{grad} f(x_t)$$

In principle x_{t+1} may not be a point in $\mathcal{M}$ for a given λ

Moreover, for finite λ, the invariance w.r.t. the parameterization is lost, due to the discretization of the gradient flow

First-order Optimization: Riemannian Gradient Descent

Consider the Euclidean naïve implementation of gradient descent over a manifold

$$x_{t+1} = x_t - \lambda \operatorname{grad} f(x_t)$$

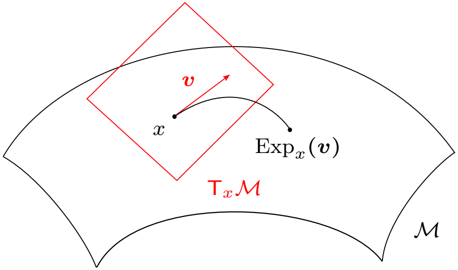
In principle x_{t+1} may not be a point in $\mathcal{M}$ for a given λ

Moreover, for finite λ, the invariance w.r.t. the parameterization is lost, due to the discretization of the gradient flow

Such problem is addressed in Riemannian optimization using the exponential map Exp_p

Exponential Map

The exponential map is a map from the tangent space $\mathsf{T}_x\mathcal{M}$ to the manifold $\mathcal{M}$, such that v is the tangent vector to the geodetic from x to $\mathrm{Exp}_{\boldsymbol{\theta}_t} v$



The exponential map can be used to implement gradient descent

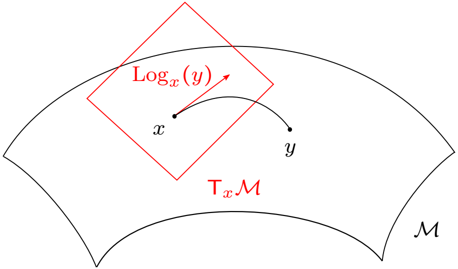
$$x_{t+1} = \mathrm{Exp}_{x_t}(-\lambda \operatorname{grad} f(x_t))$$

The exponential map may be hard to be computed, since it requires the evaluation of the geodetic $\gamma(t)$, with $\gamma(0) = p$ for a given $\dot{\gamma}(0)$

Log Map

The exponential map is a smooth map and it can be inverted to map points to the tangent space

The inverse exponential map is the log map, defined over $\mathcal{M}$ with values in $\mathsf{T}\mathcal{M}$



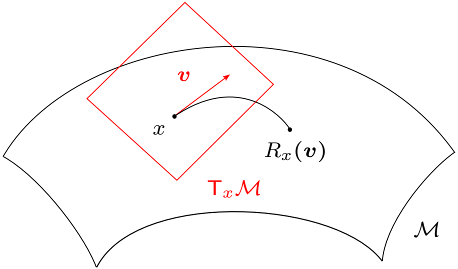
Retraction Map

Exponential maps can be hard to be computed since they require the computation of the geodetic, which is a hard task in general

Instead it is possible to consider retractions, i.e., maps from tangent space to the manifold

$$R_x(\boldsymbol{v}) : \mathsf{T}_x\mathcal{M} \to \mathcal{M}$$

with weaker conditions compared to the exponential maps, but yet strong a enough first-order constraint which ensures convergence properties



A Motivation from Optimization

Let $\mathcal{M}$ be a statistical model, i.e., a set of probability distributions over a sample space Ω, for instance,

- $\Omega = \mathbb{R}^d$, $\mathcal{M} =$ Gaussian distribution
- Ω finite, $\mathcal{M} =$ multinomial distribution

A Motivation from Optimization

Let $\mathcal{M}$ be a statistical model, i.e., a set of probability distributions over a sample space Ω, for instance,

- $\Omega = \mathbb{R}^d$, $\mathcal{M} =$ Gaussian distribution
- Ω finite, $\mathcal{M} =$ multinomial distribution

Let F be a real-valued function defined over $\mathcal{M}$, for instance,

- the log-likelihood of a sample x,
- for a $f : \Omega \to \mathbb{R}$, the stochastic relaxation of f, i.e., $F(p) = \mathbb{E}_p[f]$

A Motivation from Optimization

Let $\mathcal{M}$ be a statistical model, i.e., a set of probability distributions over a sample space Ω, for instance,

- $\Omega = \mathbb{R}^d$, $\mathcal{M}$ = Gaussian distribution
- Ω finite, $\mathcal{M}$ = multinomial distribution

Let F be a real-valued function defined over $\mathcal{M}$, for instance,

- the log-likelihood of a sample x,
- for a $f : \Omega \to \mathbb{R}$, the stochastic relaxation of f, i.e., $F(p) = \mathbb{E}_p[f]$

We are interested in the following optimization problem

$$\min_{p \in \mathcal{M}} F(p)$$

Parametric Statistical Models

Given a parameterization $\boldsymbol{\xi}$ for $\mathcal{M}$, i.e.,

$$\mathcal{M} = \{p_{\boldsymbol{\xi}}(x; \boldsymbol{\xi}) : \boldsymbol{\xi} \in \Xi\},$$

we can reformulate the previous optimization problem in a parametric form

$$\min_{\boldsymbol{\xi} \in \Xi} F_{\boldsymbol{\xi}}(\boldsymbol{\xi})$$

Parametric Statistical Models

Given a parameterization $\boldsymbol{\xi}$ for $\mathcal{M}$, i.e.,

$$\mathcal{M} = \{p_{\boldsymbol{\xi}}(x; \boldsymbol{\xi}) : \boldsymbol{\xi} \in \Xi\},$$

we can reformulate the previous optimization problem in a parametric form

$$\min_{\boldsymbol{\xi} \in \Xi} F_{\boldsymbol{\xi}}(\boldsymbol{\xi})$$

then $F_{\boldsymbol{\xi}}$ is a real-valued function defined over Ξ, e.g.,

- log-likelihood: $F_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = \mathcal{L}(\boldsymbol{\xi}|x) = \log p_{\boldsymbol{\xi}}(x; \boldsymbol{\xi})$
- stochastic relaxation of f: $F_{\boldsymbol{\xi}} = \mathbb{E}_{\boldsymbol{\xi}}[f]$

Parametric Statistical Models

Given a parameterization $\boldsymbol{\xi}$ for $\mathcal{M}$, i.e.,

$$\mathcal{M} = \{p_{\boldsymbol{\xi}}(x; \boldsymbol{\xi}) : \boldsymbol{\xi} \in \Xi\},$$

we can reformulate the previous optimization problem in a parametric form

$$\min_{\boldsymbol{\xi} \in \Xi} F_{\boldsymbol{\xi}}(\boldsymbol{\xi})$$

then $F_{\boldsymbol{\xi}}$ is a real-valued function defined over Ξ, e.g.,

- log-likelihood: $F_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = \mathcal{L}(\boldsymbol{\xi}|x) = \log p_{\boldsymbol{\xi}}(x; \boldsymbol{\xi})$
- stochastic relaxation of f: $F_{\boldsymbol{\xi}} = \mathbb{E}_{\boldsymbol{\xi}}[f]$

Independently from the nature of Ω and f, under some regularity conditions over $\mathcal{M}$, $\mathbb{E}_p[f]$ is smooth

Parametric Statistical Models

Given a parameterization $\boldsymbol{\xi}$ for $\mathcal{M}$, i.e.,

$$\mathcal{M} = \{p_{\boldsymbol{\xi}}(x;\boldsymbol{\xi}) : \boldsymbol{\xi} \in \Xi\},$$

we can reformulate the previous optimization problem in a parametric form

$$\min_{\boldsymbol{\xi}\in\Xi} F_{\boldsymbol{\xi}}(\boldsymbol{\xi})$$

then $F_{\boldsymbol{\xi}}$ is a real-valued function defined over Ξ, e.g.,

- log-likelihood: $F_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = \mathcal{L}(\boldsymbol{\xi}|x) = \log p_{\boldsymbol{\xi}}(x;\boldsymbol{\xi})$
- stochastic relaxation of f: $F_{\boldsymbol{\xi}} = \mathbb{E}_{\boldsymbol{\xi}}[f]$

Independently from the nature of Ω and f, under some regularity conditions over $\mathcal{M}$, $\mathbb{E}_p[f]$ is smooth

For smooth F, it is natural to study gradient descent methods

Gradient Descent Over Statistical Models

A natural approach consists in computing the derivative of $F(\boldsymbol{\xi})$, and implement a naive gradient descent

$$\boldsymbol{\xi}_{t+1} = \boldsymbol{\xi}_t - \lambda \nabla F_{\boldsymbol{\xi}}(\boldsymbol{\xi}_t)$$

- ∇ is shorthand for $\frac{\partial}{\partial \boldsymbol{\xi}}$
- $\lambda > 0$ step size

Gradient Descent Over Statistical Models

A natural approach consists in computing the derivative of $F(\boldsymbol{\xi})$, and implement a naive gradient descent

$$\boldsymbol{\xi}_{t+1} = \boldsymbol{\xi}_t - \lambda \nabla F_{\boldsymbol{\xi}}(\boldsymbol{\xi}_t)$$

- ∇ is shorthand for $\frac{\partial}{\partial \boldsymbol{\xi}}$
- $\lambda > 0$ step size

However a series of issues may arise:

- dependence on the parameterization
- gradient may point outside of the domain of Ξ
- target distribution may not be a critical point
- slow convergence over plateaux

Gradient Descent Over Statistical Models

A natural approach consists in computing the derivative of $F(\boldsymbol{\xi})$, and implement a naive gradient descent

$$\boldsymbol{\xi}_{t+1} = \boldsymbol{\xi}_t - \lambda \nabla F_{\boldsymbol{\xi}}(\boldsymbol{\xi}_t)$$

- ∇ is shorthand for $\frac{\partial}{\partial \boldsymbol{\xi}}$
- $\lambda > 0$ step size

However a series of issues may arise:

- dependence on the parameterization
- gradient may point outside of the domain of Ξ
- target distribution may not be a critical point
- slow convergence over plateaux

Many of these issues are consequence of the choice of a Euclidean geometry for $\mathcal{M}$

A Toy Example of Stochastic Relaxation

Let $n = 2$, $\boldsymbol{x} \in \Omega = \{-1, +1\}^2$, suppose $f(\boldsymbol{x}) = x_1 + 2x_2 + 3x_1x_2$

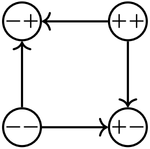
x_1	x_2	$f(\boldsymbol{x})$
$+1$	$+1$	6
$+1$	-1	-4
-1	$+1$	-2
-1	-1	0

$$\begin{tikzcd} -+ & ++ \arrow[l] \arrow[d] \\ -- \arrow[u] \arrow[r] & +- \end{tikzcd}$$

A Toy Example of Stochastic Relaxation

Let $n = 2$, $\boldsymbol{x} \in \Omega = \{-1, +1\}^2$, suppose $f(\boldsymbol{x}) = x_1 + 2x_2 + 3x_1x_2$

x_1	x_2	$f(\boldsymbol{x})$
$+1$	$+1$	6
$+1$	-1	-4
-1	$+1$	-2
-1	-1	0

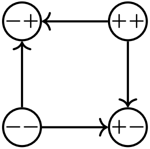


We want to minimize the stochastic relaxation $F(p) = \mathbb{E}_p[f]$ for p in the independence model for $\boldsymbol{x}$

A Toy Example of Stochastic Relaxation

Let $n = 2$, $\boldsymbol{x} \in \Omega = \{-1, +1\}^2$, suppose $f(\boldsymbol{x}) = x_1 + 2x_2 + 3x_1x_2$

x_1	x_2	$f(\boldsymbol{x})$
$+1$	$+1$	6
$+1$	-1	-4
-1	$+1$	-2
-1	-1	0



We want to minimize the stochastic relaxation $F(p) = \mathbb{E}_p[f]$ for p in the independence model for $\boldsymbol{x}$

In the following, we will study the Euclidean gradient flow for different parameterizations, i.e., the solution of the differential equation

$$\dot{\boldsymbol{\xi}} = \nabla F_{\boldsymbol{\xi}}(\boldsymbol{\xi})$$

Gradient Flows on the Independence Model

Let $\mu_i = \mathbb{P}(X_i = 1)$, then $\boldsymbol{\mu} = (\mu_1, \mu_2) \in [0, 1]$

$$F_{\boldsymbol{\mu}}(\boldsymbol{\mu}) = \sum_{\boldsymbol{x} \in \Omega} f(\boldsymbol{x}) p_1(x_1) p_2(x_2) = -4\mu_1 - 2\mu_2 + 12\mu_1\mu_2$$

$$\nabla F_{\boldsymbol{\mu}}(\boldsymbol{\mu}) = (-4 + 12\mu_2, -2 + 12\mu_1)^{\mathrm{T}}$$

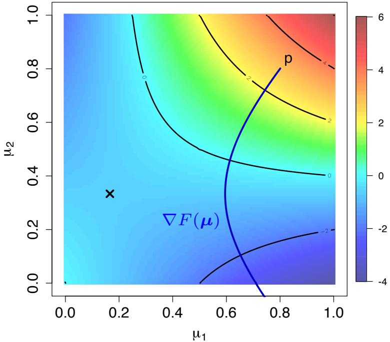
Gradient Flows on the Independence Model

Let $\mu_i = \mathbb{P}(X_i = 1)$, then $\boldsymbol{\mu} = (\mu_1, \mu_2) \in [0,1]$

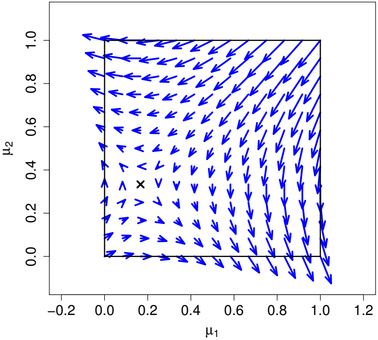
$$F_{\boldsymbol{\mu}}(\boldsymbol{\mu}) = \sum_{\boldsymbol{x}\in\Omega} f(\boldsymbol{x}) p_1(x_1) p_2(x_2) = -4\mu_1 - 2\mu_2 + 12\mu_1\mu_2$$

$$\nabla F_{\boldsymbol{\mu}}(\boldsymbol{\mu}) = (-4 + 12\mu_2, -2 + 12\mu_1)^{\mathrm{T}}$$

Gradient flow over $\boldsymbol{\mu}$



Gradient vector over $\boldsymbol{\mu}$, $\lambda = 0.025$



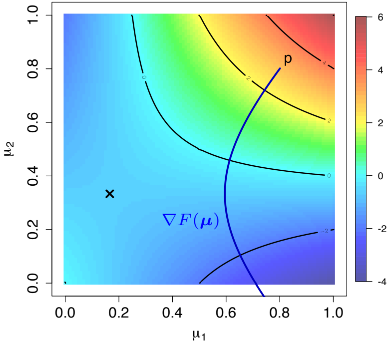
Gradient Flows on the Independence Model

Let $\mu_i = \mathbb{P}(X_i = 1)$, then $\boldsymbol{\mu} = (\mu_1, \mu_2) \in [0,1]$

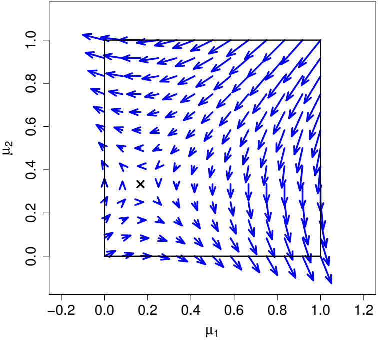
$$F_{\boldsymbol{\mu}}(\boldsymbol{\mu}) = \sum_{\boldsymbol{x}\in\Omega} f(\boldsymbol{x}) p_1(x_1) p_2(x_2) = -4\mu_1 - 2\mu_2 + 12\mu_1\mu_2$$

$$\nabla F_{\boldsymbol{\mu}}(\boldsymbol{\mu}) = (-4 + 12\mu_2, -2 + 12\mu_1)^{\mathrm{T}}$$

Gradient flow over $\boldsymbol{\mu}$



Gradient vector over $\boldsymbol{\mu}$, $\lambda = 0.025$



$\nabla F_{\boldsymbol{\mu}}(\boldsymbol{\mu})$ does not vanish on local optima, projections are required

Natural Parameters for the Independence Model

If we restrict to positive probabilities $p > 0$, we can represent the interior of the independence model as the exponential family

$$p(\boldsymbol{x};\boldsymbol{\theta}) = \exp\{\theta_1 x_1 + \theta_2 x_2 - \psi(\boldsymbol{\theta})\}$$

where $\psi(\boldsymbol{\theta}) = \ln Z(\boldsymbol{\theta})$ is the log-partition function

The natural parameters of the independence model represented as an exponential family are $\boldsymbol{\theta} = (\theta_1, \theta_2) \in \mathbb{R}^2$, with

$$p_i(x_i) = \frac{e^{\theta_i x_i}}{e^{\theta_i} + e^{-\theta_i}}$$

Natural Parameters for the Independence Model

If we restrict to positive probabilities $p > 0$, we can represent the interior of the independence model as the exponential family

$$p(\boldsymbol{x}; \boldsymbol{\theta}) = \exp\{\theta_1 x_1 + \theta_2 x_2 - \psi(\boldsymbol{\theta})\}$$

where $\psi(\boldsymbol{\theta}) = \ln Z(\boldsymbol{\theta})$ is the log-partition function

The natural parameters of the independence model represented as an exponential family are $\boldsymbol{\theta} = (\theta_1, \theta_2) \in \mathbb{R}^2$, with

$$p_i(x_i) = \frac{e^{\theta_i x_i}}{e^{\theta_i} + e^{-\theta_i}}$$

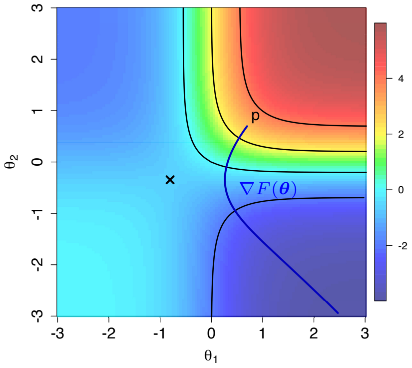
The mapping between marginal probabilities and natural parameters is one-to-one for $p > 0$

$$\theta_i = \frac{1}{2}\left(\ln(\mu_i) - \ln(1 - \mu_i)\right) \qquad \mu_i = \frac{e^{\theta_i}}{e^{\theta_i} + e^{-\theta_i}}$$

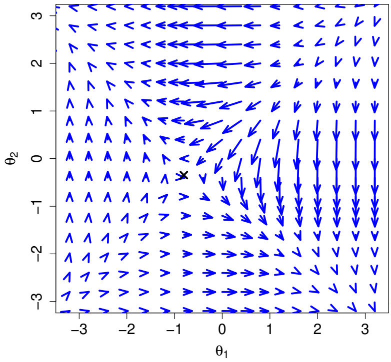
Gradient Flows on the Independence Model

$$F_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = (-4e^{\theta_1-\theta_2} - 2e^{-\theta_1+\theta_2} + 6e^{\theta_1+\theta_2})/Z(\boldsymbol{\theta})$$
$$\nabla F_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{\theta}}[f(\boldsymbol{X} - \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{X}])] = \mathrm{Cov}_{\boldsymbol{\theta}}(f, \boldsymbol{X})$$

Gradient flow over $\boldsymbol{\theta}$



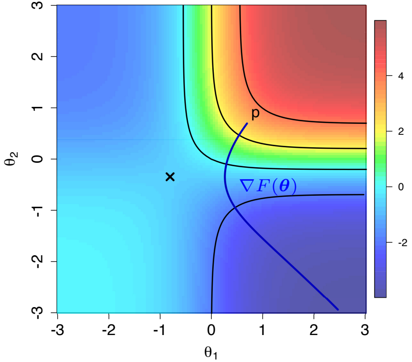
Gradient vectors over $\boldsymbol{\theta}$, $\lambda = 0.15$



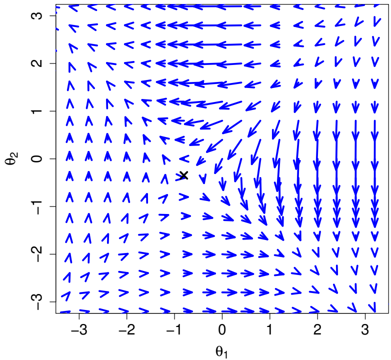
Gradient Flows on the Independence Model

$$F_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = (-4e^{\theta_1-\theta_2} - 2e^{-\theta_1+\theta_2} + 6e^{\theta_1+\theta_2})/Z(\boldsymbol{\theta})$$
$$\nabla F_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{\theta}}[f(\boldsymbol{X} - \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{X}])] = \mathrm{Cov}_{\boldsymbol{\theta}}(f, \boldsymbol{X})$$

Gradient flow over $\boldsymbol{\theta}$

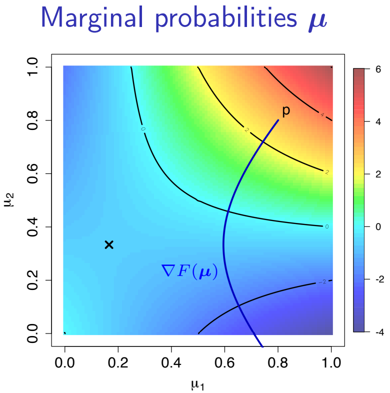


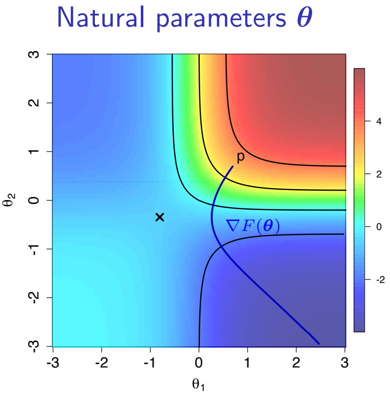
Gradient vectors over $\boldsymbol{\theta}$, $\lambda = 0.15$



In the natural parameters, $\nabla F_{\boldsymbol{\theta}}(\boldsymbol{\theta})$ vanishes over the plateaux

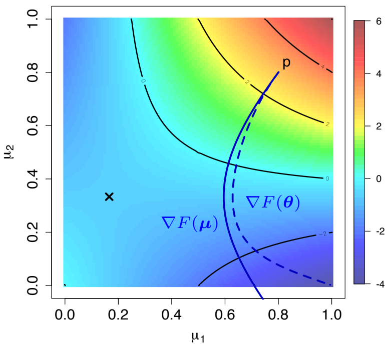
Gradient Flows on the Independence Model



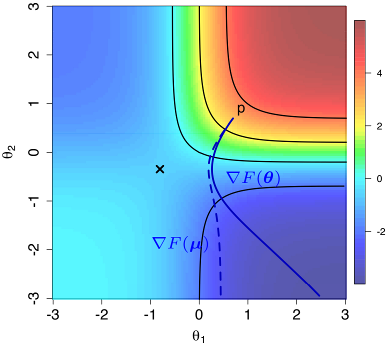


Gradient Flows on the Independence Model

Marginal probabilities $\boldsymbol{\mu}$



Natural parameters $\boldsymbol{\theta}$



Gradient flows of $F_{\boldsymbol{\xi}}(\boldsymbol{\xi})$ depend on the parameterization $\boldsymbol{\xi}$

The trajectories associated to $\nabla F_{\boldsymbol{\xi}}(\boldsymbol{\xi})$ may not convergence to the expected distribution unless a projection is computed

Information Geometry

Euclidean geometry is not the most convenient geometry for statistical models, as (probably) first remarked by Hotelling (1930) and Rao (1945)

Information Geometry

Euclidean geometry is not the most convenient geometry for statistical models, as (probably) first remarked by Hotelling (1930) and Rao (1945)

Information Geometry follows a different geometric approach, given by the representation of statistical models as Riemannian statistical manifolds, endowed with the Fisher information metric

Besides the Riemannian one, Information Geometry also studies other non-Euclidean geometries for statistical models, based on the notion of dual affine manifolds

Information Geometry

Euclidean geometry is not the most convenient geometry for statistical models, as (probably) first remarked by Hotelling (1930) and Rao (1945)

Information Geometry follows a different geometric approach, given by the representation of statistical models as Riemannian statistical manifolds, endowed with the Fisher information metric

Besides the Riemannian one, Information Geometry also studies other non-Euclidean geometries for statistical models, based on the notion of dual affine manifolds

The research in Information Geometry has started in the 80's, with the pioneer work of Amari (1982,1985), Barndorff-Nielsen (1978), Cencov (1982), Lauritzen (1987), Pistone and Sempi (1995) and colleagues

Standard References

Three monographs by Amari, who is considered the founder of Information Geometry

- S.-I. Amari. *Differential-geometrical methods in statistics.* Lecture notes in statistics, Springer-Verlag, Berlin, 1985.
- S.-I. Amari and Hiroshi Nagaoka. *Methods of Information Geometry*. AMS, Oxford University Press, 2000. Translated from the 1993 Japanese original by Daishi Harada.
- S.-I. Amari. *Information Geometry and Its Applications.* Springer, 2016.

Other standard references are

- M. Murray and J. Rice. *Differential geometry and statistics.* Monographs on Statistics and Applied Probability 48. Chapman and Hall, 1993.
- R. E. Kass and P. W. Vos. *Geometrical Foundations of Asymptotic Inference.* Series in Probability and Statistics, Wiley, 1997.

Geometry Derived by the KL Divergence

An alternative geometry for a statistical model can be defined by measuring infinitesimal distances using the Kullback-Leibler divergence

$$D_{\mathsf{KL}}(p||q) = \int_\Omega p(x) \log \frac{p(x)}{q(x)} \, \mathrm{d}x$$

Geometry Derived by the KL Divergence

An alternative geometry for a statistical model can be defined by measuring infinitesimal distances using the Kullback-Leibler divergence

$$D_{\mathsf{KL}}(p\|q) = \int_\Omega p(x) \log \frac{p(x)}{q(x)} \, \mathrm{d}x$$

It can be proved that such choice determines a Riemannian structure for $\mathcal{M}$, where the Fisher Information matrix plays the role of metric tensor

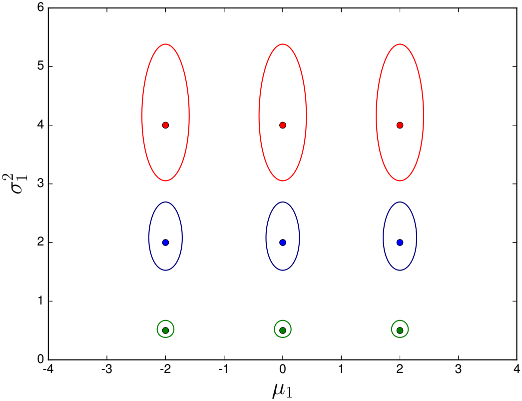
The direction of steepest ascent $\Delta\boldsymbol{\theta}$ for a function $F_{\boldsymbol{\theta}}$ can then be evaluated by solving

$$\begin{aligned} &\arg\min_{\Delta\boldsymbol{\theta}} \; F_{\boldsymbol{\theta}}(\boldsymbol{\theta} + \Delta\boldsymbol{\theta}) \\ &\text{s.t. } D_{\mathsf{KL}}(p_{\boldsymbol{\theta}} \| p_{\boldsymbol{\theta}+\Delta\boldsymbol{\theta}}) = \epsilon \end{aligned}$$

where the constraints replaces $\|\Delta\boldsymbol{\theta}\| = 1$ in the Euclidean case

Example: The Gaussian Distribution

ϵ–ball of constant KL divergence, $\epsilon = 0.02$



Let $p_0 \sim \mathcal{N}(\mu_0, \sigma_0^2)$, and $p_1 \sim \mathcal{N}(\mu_1, \sigma_1^2)$,

$$D_{\mathsf{KL}}(p_0 \| p_1) = \log \frac{\sigma_1}{\sigma_0} + \frac{\sigma_0^2 + (\mu_0 - \mu_1)^2}{2\sigma_1^2} - \frac{1}{2}$$

Amari's Natural Gradient (1998) 1/2

By taking the second-order Taylor approximation of the KL divergence in $\boldsymbol{\xi}$ we get

$$\begin{aligned} D_{\mathsf{KL}}(p_{\boldsymbol{\xi}} \| p_{\boldsymbol{\xi}+\Delta\boldsymbol{\xi}}) &= \mathbb{E}_{\boldsymbol{\xi}}[\log p_{\boldsymbol{\xi}}] - \mathbb{E}_{\boldsymbol{\xi}}[\log p_{\boldsymbol{\xi}+\Delta\boldsymbol{\xi}}] \\ &\approx \mathbb{E}_{\boldsymbol{\xi}}[\log p_{\boldsymbol{\xi}}] - \mathbb{E}_{\boldsymbol{\xi}}[\log p_{\boldsymbol{\xi}}] - \mathbb{E}_{\boldsymbol{\xi}}[\nabla \log p_{\boldsymbol{\xi}}]^{\mathrm{T}} \Delta\boldsymbol{\xi} + \\ &\quad - \frac{1}{2} \Delta\boldsymbol{\xi}^{\mathrm{T}} \mathbb{E}_{\boldsymbol{\xi}} \left[\nabla^2 \log p_{\boldsymbol{\xi}} \right] \Delta\boldsymbol{\xi} \\ &= \frac{1}{2} \Delta\boldsymbol{\xi}^{\mathrm{T}} I(\boldsymbol{\xi}) \Delta\boldsymbol{\xi}, \end{aligned}$$

where $I_{\boldsymbol{\xi}}(\boldsymbol{\xi})$ is the Fisher Information matrix

$$\begin{aligned} I_{\boldsymbol{\xi}}(\boldsymbol{\xi}) &= -\mathbb{E}_{\boldsymbol{\xi}} \left[\nabla^2 \log p_{\boldsymbol{\xi}+\Delta\boldsymbol{\xi}} \right] \\ &= \mathbb{E}_{\boldsymbol{\xi}} \left[\nabla \log p(\boldsymbol{\xi}) \nabla \log p(\boldsymbol{\xi})^{\mathrm{T}} \right] \end{aligned}$$

Amari's Natural Gradient (1998) 2/2

We proceed by taking the first-order approximation of $F_{\boldsymbol{\xi}}(\boldsymbol{\xi} + \Delta\boldsymbol{\xi})$

$$\begin{aligned} \arg\min_{\Delta\boldsymbol{\xi}} \; & F_{\boldsymbol{\xi}}(\boldsymbol{\xi}) + \nabla F_{\boldsymbol{\xi}}(\boldsymbol{\xi})^{\mathrm{T}} \Delta\boldsymbol{\xi} \\ \text{s.t. } & \frac{1}{2} \Delta\boldsymbol{\xi}^{\mathrm{T}} I_{\boldsymbol{\xi}}(\boldsymbol{\xi}) \Delta\boldsymbol{\xi} = \epsilon \end{aligned}$$

We apply the Lagrangian method, and solve for $\Delta\boldsymbol{\xi}$

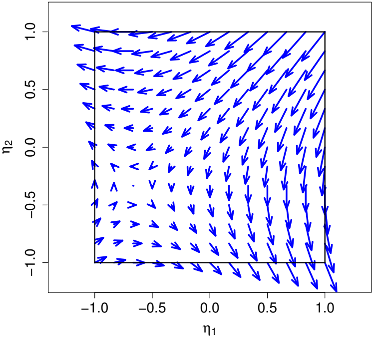
$$\begin{aligned} & \nabla_{\Delta\boldsymbol{\xi}} \left(F_{\boldsymbol{\xi}}(\boldsymbol{\xi}) + \nabla F_{\boldsymbol{\xi}}(\boldsymbol{\xi})^{\mathrm{T}} \Delta\boldsymbol{\xi} - \lambda \frac{1}{2} \Delta\boldsymbol{\xi}^{\mathrm{T}} I_{\boldsymbol{\xi}}(\boldsymbol{\xi}) \Delta\boldsymbol{\xi} \right) = 0 \\ & \nabla F_{\boldsymbol{\xi}}(\boldsymbol{\xi}) - \lambda I_{\boldsymbol{\xi}}(\boldsymbol{\xi}) \Delta\boldsymbol{\xi} = 0 \\ & \Delta\boldsymbol{\xi} = \lambda I_{\boldsymbol{\xi}}(\boldsymbol{\xi})^{-1} \nabla F_{\boldsymbol{\xi}}(\boldsymbol{\xi}) \end{aligned}$$

Such derivations lead to the natural gradient (Amari, 1998)

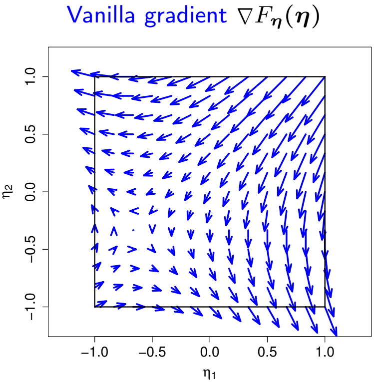
$$\widetilde{\nabla} F_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = I_{\boldsymbol{\xi}}(\boldsymbol{\xi})^{-1} \nabla F_{\boldsymbol{\xi}}(\boldsymbol{\xi})$$

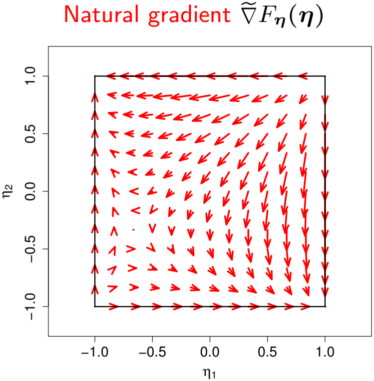
Vanilla vs Natural Gradient: $\boldsymbol{\eta}, \lambda = 0.05$

Vanilla gradient $\nabla F_{\boldsymbol{\eta}}(\boldsymbol{\eta})$



Vanilla vs Natural Gradient: $\boldsymbol{\eta}, \lambda = 0.05$

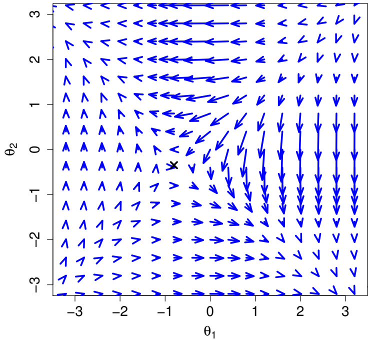




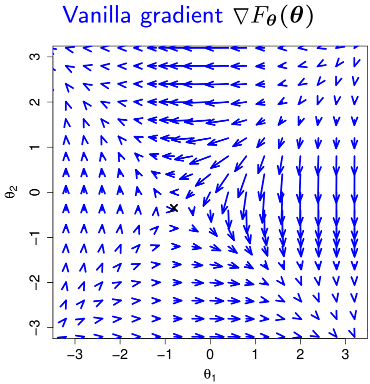
In both cases there exist two basins of attraction, however $\widetilde{\nabla} F_{\boldsymbol{\eta}}(\boldsymbol{\eta})$ convergences to $\delta_{\boldsymbol{x}}$ distributions, which are local optima for $F_{\boldsymbol{\eta}}(\boldsymbol{\eta})$ and where $\widetilde{\nabla} F_{\boldsymbol{\eta}}(\delta_{\boldsymbol{x}}) = 0$

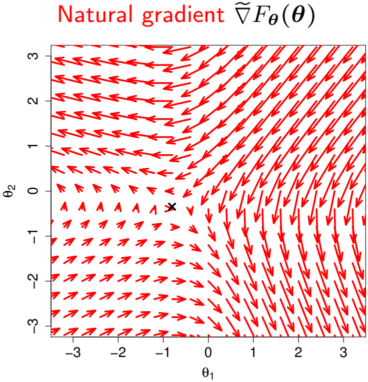
Euclidean vs Natural Gradient: $\boldsymbol{\theta}, \lambda = 0.15$

Vanilla gradient $\nabla F_{\boldsymbol{\theta}}(\boldsymbol{\theta})$



Euclidean vs Natural Gradient: $\boldsymbol{\theta}, \lambda = 0.15$

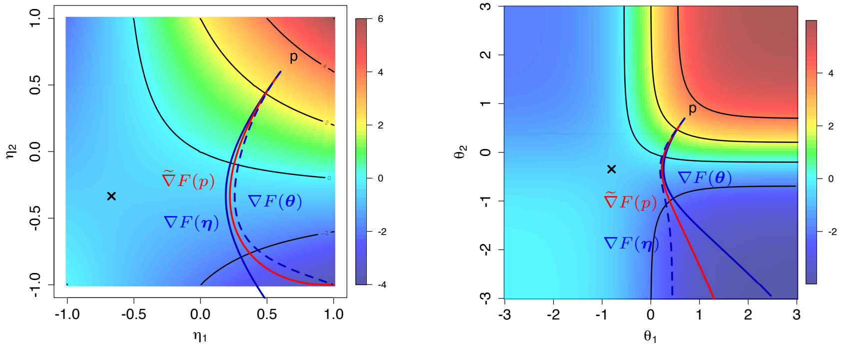




In both cases there exist two basins of attraction, however in the natural parameters $\widetilde{\nabla} F_{\boldsymbol{\theta}}(\boldsymbol{\theta})$ "speeds up" over the plateaux

Euclidean vs Natural Gradient

Expectation parameters $\boldsymbol{\eta}$ | Natural parameters $\boldsymbol{\theta}$



Vanilla gradient ∇F vs Natural gradient $\widetilde{\nabla} F$

The natural gradient flow is invariant to parameterization

Riemannian Geometry of Statistical Manifolds

In the previous slide the natural gradient has been derived by imposing a constant KL divergence

From a differential geometry point of view, the natural gradient corresponds to the Riemannian gradient over a statistical manifolds endowed with the Fisher information metric

The Exponential Family

In the following, we consider models in the exponential family $\mathcal{E}$

$$p(\boldsymbol{x}, \boldsymbol{\theta}) = \exp\left(\sum_{i=1}^{m} \theta_i T_i(\boldsymbol{x}) - \psi(\boldsymbol{\theta})\right)$$

- sufficient statistics $\boldsymbol{T} = (T_1(\boldsymbol{x}), \ldots, T_m(\boldsymbol{x}))$
- natural parameters $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_m) \in \Theta \subset \mathbb{R}^m$
- log-partition function $\psi(\boldsymbol{\theta})$

Fisher Information Metric

The tangent space at each point p is defined by

$$\mathsf{T}_p\mathcal{M} = \{U(\boldsymbol{x}) : \mathbb{E}_p[U(\boldsymbol{x})] = 0\}$$

Fisher Information Metric

The tangent space at each point p is defined by

$$\mathsf{T}_p\mathcal{M} = \{U(\boldsymbol{x}) : \mathbb{E}_p[U(\boldsymbol{x})] = 0\}$$

Consider a curve $p(\theta)$ such that $p(0) = p$, then $\frac{\dot{p}}{p} \in \mathsf{T}_p$

In a moving coordinate system, tangent (velocity) vectors in $\mathsf{T}_{p(\theta)}$ to the curve are given by logarithmic derivative

$$\frac{\dot{p}(\theta)}{p(\theta)} = \frac{d}{d\theta}\log p(\theta) \qquad \mathsf{T}_p\mathcal{M} = \mathrm{Span}\{T_i(\boldsymbol{x}) - E_p[T_i(\boldsymbol{x})]\}$$

Fisher Information Metric

The tangent space at each point p is defined by

$$\mathsf{T}_p\mathcal{M} = \{U(\boldsymbol{x}) : \mathbb{E}_p[U(\boldsymbol{x})] = 0\}$$

Consider a curve $p(\theta)$ such that $p(0) = p$, then $\frac{\dot{p}}{p} \in \mathsf{T}_p$

In a moving coordinate system, tangent (velocity) vectors in $\mathsf{T}_{p(\theta)}$ to the curve are given by logarithmic derivative

$$\frac{\dot{p}(\theta)}{p(\theta)} = \frac{d}{d\theta}\log p(\theta) \qquad \mathsf{T}_p\mathcal{M} = \mathrm{Span}\{T_i(\boldsymbol{x}) - E_p[T_i(\boldsymbol{x})]\}$$

The tangent space is provided with an inner product $\langle U, V\rangle_p = \mathbb{E}_p[UV] = \boldsymbol{u}^{\mathrm{T}} I(p)\boldsymbol{v}$ defined by the Fisher information matrix

$$I_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = [g_{ij}] = \mathbb{E}_{\boldsymbol{\theta}}\left[\frac{d}{d\theta_i}\log p(\boldsymbol{\theta})\frac{d}{d\theta_j}\log p(\boldsymbol{\theta})\right]$$

Black-Box Optimization in $\mathbb{R}^n$

The stochastic relaxation of a continuos function with respect to the Gaussian distribution is a design principle for popular model-based algorithms in Evolutionary Computation

- Covariance Matrix Adaptation CMA-ES (Hansen and Ostermeier, 2001; Akimoto et. al., 2012)
- Natural Evolutionary Strategies - NES (Wiestra et. al., 2008-14)

See also

- Malagò et. al., 2011, for the stochastic relaxation of pseudo-Boolean functions with respect to the exponential family
- Information Geometry Optimization - IGO (Ollivier et. al., 2011) for a general framework for stochastic relaxation

Gaussian distribution and Natural Gradient

CMA and NES explicitly implement a gradient descent paradigm for $F = \mathbb{E}_p[f]$ with respect to the multivariate Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$

$$\mu_{t+1} = \mu_t - \lambda \operatorname{grad} F(\mu_t, \Sigma_t)$$
$$\Sigma_{t+1} = \Sigma_t - \lambda \operatorname{grad} F(\mu_t, \Sigma_t)$$

Implementing natural gradient by itself is not sufficient, indeed parameter tuning and other adaptation mechanisms play a fundamental role for efficient algorithms

Estimation of $\widetilde{\nabla} F$ by Monte Carlo Methods

Due to the properties of the exponential family

$$I_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \operatorname{Hess}\psi(\boldsymbol{\theta}) = \operatorname{Cov}_{\boldsymbol{\theta}}(\boldsymbol{T}, \boldsymbol{T})$$

Moreover, for $F = \mathbb{E}_{\boldsymbol{\theta}}[f]$, we have

$$\nabla F_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \operatorname{Cov}(f, \boldsymbol{T}),$$

this implies

$$\widetilde{\nabla} F_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \operatorname{Cov}_{\boldsymbol{\theta}}(\boldsymbol{T}, \boldsymbol{T})^{-1} \operatorname{Cov}(f, \boldsymbol{T})$$

It follows that vanilla and natural gradient in $\boldsymbol{\theta}$ can be expressed in terms of covariances that only depend on the evaluation of f

Estimation of $\widetilde{\nabla} F$ by Monte Carlo Methods

Due to the properties of the exponential family

$$I_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \mathrm{Hess}\,\psi(\boldsymbol{\theta}) = \mathrm{Cov}_{\boldsymbol{\theta}}(\boldsymbol{T}, \boldsymbol{T})$$

Moreover, for $F = \mathbb{E}_{\boldsymbol{\theta}}[f]$, we have

$$\nabla F_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \mathrm{Cov}(f, \boldsymbol{T}),$$

this implies

$$\widetilde{\nabla} F_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \mathrm{Cov}_{\boldsymbol{\theta}}(\boldsymbol{T}, \boldsymbol{T})^{-1}\,\mathrm{Cov}(f, \boldsymbol{T})$$

It follows that vanilla and natural gradient in $\boldsymbol{\theta}$ can be expressed in terms of covariances that only depend on the evaluation of f

Thus Monte Carlo methods can be used in the estimation

Natural Gradient in Machine Learning

Natural gradient (Amari, 1998) methods are becoming constantly popular in machine learning, e.g.,

- Training of Neural Networks (Amari, 1997) and recently Deep Learning (Ollivier et. al., 2014; Pascanu and Bengio, 2014; Martens et. al 2015; Desjardins et. al., 2014)
- Reinforcement learning and Markov Decision Processes (Kakade, 2001; Peters and Schaal, 2008)
- Stochastic Relaxation and Evolutionary Optimization (i.e., black-box derivative-free methods) (Wiestra et. al., 2008-14; Malagò et. al., 2011; Ollivier et. al., 2011; Akimoto et. al., 2012)
- Bayesian variational inference (Honkela et. al., 2008)
- Bayesian optimization
- and many others

Manifold Optimization in Evolutionary Computation

Manifold optimization is an active and expanding research field

In the last 10 years the number of algorithms and applications increased, with a focus on first- and second-order algorithms

Manifold Optimization in Evolutionary Computation

Manifold optimization is an active and expanding research field

In the last 10 years the number of algorithms and applications increased, with a focus on first- and second-order algorithms

More recently, notions from optimization over manifolds are starting to appear also in the design of meta-heuristics, e.g.,

- Modified Particle Swarm Optimization for multilinear rank approximations (Borckmans et. al., 2010)
- Oriented Bounding Box Computation Using Particle Swarm Optimization (Borckmans and Absil, 2010)
- Manifold distance-based particle swarm optimization for classification (Liu et. al., 2013)
- Fuzzy Adaptive Simulated Annealing in Evolutionary Global Optimization, Manifolds and Applications (Aguiar e Oliveira Junior, 2016)

Derivative-free Optimization and Tangent Vectors

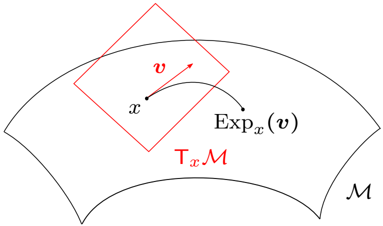
Black-box optimization doesn't rely on exact first-order information, thus the Riemannian gradient cannot be evaluated

However, the presence of a manifold structure for the domain of the cost function has an impact on the evaluation of tangent vectors

Derivative-free Optimization and Tangent Vectors

Black-box optimization doesn't rely on exact first-order information, thus the Riemannian gradient cannot be evaluated

However, the presence of a manifold structure for the domain of the cost function has an impact on the evaluation of tangent vectors



Differently from an Euclidean space, where $\boldsymbol{v} = y - x$, the tangent vector $\boldsymbol{v}$ to the geodetic from x to y is obtained using the Log map

(Euclidean) Particle Swart Optimization

PSO (Eberhart and Kennedy, 1995) is a popular population-based algorithm where particles are evolving in the search space guided by velocity vectors, i.e.,

$$\boldsymbol{v}_i(t+1) = w(t)\boldsymbol{v}_i(t) + c\alpha_i(t)(y_i(t) - x_i(t)) + s\beta_i(t)(\hat{y}_i(t) - x_i(t))$$
$$x_i(t+1) = x_i(t) + v_i(t+1)$$

where

- y is the best personal position so far
- $\hat{y}$ the best global position found by the swarm

Riemannian Particle Swart Optimization

Log and exponential maps (or retractions) are the tools required to generalize population based-algorithms, such as PSO, to search spaces which admit a manifold structure

$$\boldsymbol{v}_i(t+1) = w(t)\boldsymbol{v}_i(t) + c\alpha_i(t)\mathrm{Log}_{x_i(t)}\, y_i(t) + s\beta_i(t)\mathrm{Log}_{x_i(t)}\, \hat{y}_i(t)$$

$$x_i(t+1) = \mathrm{Exp}_{x_i(t+1)}\, v_i(t+1)$$

Riemannian Particle Swart Optimization

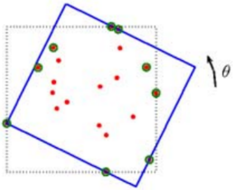
Log and exponential maps (or retractions) are the tools required to generalize population based-algorithms, such as PSO, to search spaces which admit a manifold structure

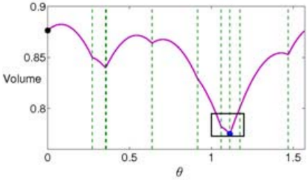
$$\boldsymbol{v}_i(t+1) = w(t)\boldsymbol{v}_i(t) + c\alpha_i(t)\mathrm{Log}_{x_i(t)}\, y_i(t) + s\beta_i(t)\mathrm{Log}_{x_i(t)}\, \hat{y}_i(t)$$
$$x_i(t+1) = \mathrm{Exp}_{x_i(t+1)}\, v_i(t+1)$$

- The formulæ for the log and the exponential maps depend on the manifold structure associated to the search space
- By exploiting the manifold structure we have in general better convergence properties
- However taking into account the manifold structure may result in a higher computational cost for the algorithm

Oriented Bounding Box Computation Using PSO

In the following we refer to Borckmans and Absil (2010), where oriented bounding box are computed in $\mathbb{R}^3$ using PSO





data set	method	time (s)	relative error (%)
set1 132 nodes	O'Rourke	$29 \cdot 10^{0}$	0
	PCA	$20 \cdot 10^{-5}$	28.04
	PSO	$55 \cdot 10^{-1}$	min: $21 \cdot 10^{-14}$ max: $10 \cdot 10^{-2}$ mean: $17 \cdot 10^{-2}$ var: $33 \cdot 10^{-3}$
set2 6479 nodes	O'Rourke	$14 \cdot 10^{3}$	0
	PCA	$38 \cdot 10^{-2}$	114.9
	PSO	$87 \cdot 10^{-1}$	min: $15 \cdot 10^{-12}$ max: $56 \cdot 10^{-2}$ mean: $17 \cdot 10^{-2}$ var: $46 \cdot 10^{-3}$
set3 1560 nodes	O'Rourke	$22 \cdot 10^{2}$	0
	PCA	$49 \cdot 10^{-4}$	83.8
	PSO	$67 \cdot 10^{-1}$	min: $15 \cdot 10^{-12}$ max: $56 \cdot 10^{-2}$ mean: $17 \cdot 10^{-2}$ var: $46 \cdot 10^{-3}$

Take Home Messages

- The geometry of the search space plays an important role in optimization
- Optimization over manifolds offers a formal framework for design and analysis of algorithms

Take Home Messages

- The geometry of the search space plays an important role in optimization
- Optimization over manifolds offers a formal framework for design and analysis of algorithms

- The geometry of statistical models is much richer than one could expect
- Information Geometry provides a unifying geometric framework for the analysis of model-based optimization

Take Home Messages

- The geometry of the search space plays an important role in optimization
- Optimization over manifolds offers a formal framework for design and analysis of algorithms

- The geometry of statistical models is much richer than one could expect
- Information Geometry provides a unifying geometric framework for the analysis of model-based optimization

- Riemannian optimization only recently started to play a role in Evolutionary Computation
- There is a lot of room for further developments and cross-fertilization between the two fields

Open Postdocs Positions at RIST

The Romanian Institute of Science and Technology has multiple postdoc positions on Information Geometry, Riemannian Optimization and Deep Learning, funded by two 4-year EU Projects

Cluj-Napoca is the capital of Transylvania and the 2nd largest city in Romania

12 universities and over 70k students

IT hub (9% growth per year) - The Silicon Valley of Transylvania

RIST is a private and non profit research institute founded in 2008

Talk to me for more information or write me at `malago@rist.ro`