An Introduction to Natural Gradient with Applications in Optimization and Neural Networks

Luigi Malagò

Transylvanian Institute of Neuroscience
Quaesta AI

University of Genova, DIMA

March 6, 2023

Outline of Part 1

- Motivations from Statistics and Machine Learning
- A Gentle Introduction to Riemannian Optimization
- Amari's Natural Gradient
- Applications in Stochastic Optimization
- Applications in Training Neural Networks

Outline of Part 1

- Motivations from Statistics and Machine Learning
- A Gentle Introduction to Riemannian Optimization
- Amari's Natural Gradient
- Applications in Stochastic Optimization
- Applications in Training Neural Networks

Standard monographs

S.-I. Amari, H. Nagaoka
Methods of Information Geometry. Oxford University Press, 2000

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

`https://press.princeton.edu/absil`

Motivations from Statistics and Machine Learning

Consider the following optimization problems

- log-likelihood estimation of the parameters of a statistical model

$$\max \log p(x;\xi)$$

- stochastic relaxation of a function $f(x) : \Omega \to \mathbb{R}$

$$\min \mathbb{E}_{p(x;\xi)}[f(x)]$$

(cf. stochastic optimization)

- minimization (or maximization) of the polarization measure

$$\mathrm{POL}(p) = \sum_i p_i^2 (1 - p_i)$$

(Pino and Vidal-Robert, 2013)

Optimization over Statistical Models

The previous examples fit the general case of the optimization of a function whose variables are the parameters of a statistical model

Optimization over Statistical Models

The previous examples fit the general case of the optimization of a function whose variables are the parameters of a statistical model

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

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

We want to solve the following optimization problem

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

Optimization over Statistical Models

The previous examples fit the general case of the optimization of a function whose variables are the parameters of a statistical model

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

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

We want to solve the following optimization problem

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

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

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

the problem can be reformulated in a parametric form

$$\inf_{\xi \in \Xi} F(\xi) = \inf_{\xi \in \Xi} (\mathcal{F} \circ p)(\xi)$$

Gradient Descent Over Statistical Models

Optimizing $F(\boldsymbol{\xi})$ over Ξ may be a non-trivial task, e.g., for non-convex functions and in absence of closed-form solutions

A naive but still powerfull approach is gradient descent

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

where $\lambda > 0$ is the step size, and ∇ stands for $\frac{\partial}{\partial \boldsymbol{\xi}}$

However a series of issues may arise:

- dependence on the parameterization ξ
- slow convergence over plateaux
- a projection of the gradient is required on the boundary of Ξ

$\Rightarrow$ Most of these problem can be overcome by choosing a more convenient geometry for $\mathcal{M}$

A Toy Example from Stochastic Relaxation

Let $n = 2$, $\Omega = \{-1, +1\}^2$, we want to minimize $\mathcal{F}(p) = \mathbb{E}_p[f]$

$$f(\boldsymbol{x}) = x_1 + 2x_2 + 3x_1x_2$$

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

A Toy Example from Stochastic Relaxation

Let $n = 2$, $\Omega = \{-1, +1\}^2$, we want to minimize $\mathcal{F}(p) = \mathbb{E}_p[f]$

$$f(\boldsymbol{x}) = x_1 + 2x_2 + 3x_1 x_2$$

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

The Euclidean gradient flow is the solution of the following differential equation

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

We are interested in studying gradient flows for different parameterization over the independence model for X_1, X_2

Gradient Flows on the Independence Model

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

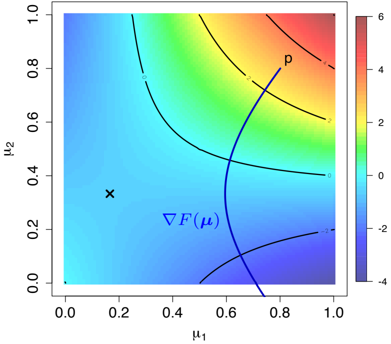
$$\nabla F(\boldsymbol{\mu}) = \left(-4 + 12\mu_2, -2 + 12\mu_1\right)^{\mathrm{T}}$$

Gradient Flows on the Independence Model

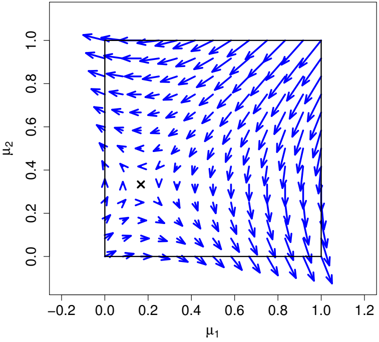
$$F(\boldsymbol{\mu}) = \sum_{\boldsymbol{x}\in\Omega} f(\boldsymbol{x}) p_1(x_1) p_2(x_2) = -4\mu_1 - 2\mu_2 + 12\mu_1\mu_2$$

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

Gradient flow in $\boldsymbol{\mu}$



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

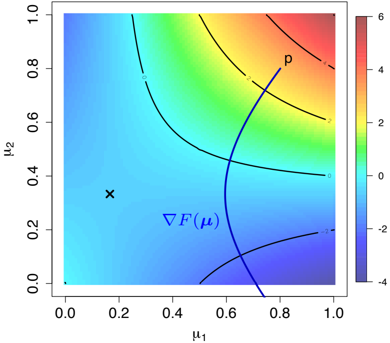


Gradient Flows on the Independence Model

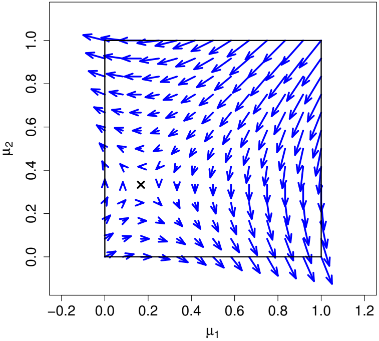
$$F(\boldsymbol{\mu}) = \sum_{\boldsymbol{x}\in\Omega} f(\boldsymbol{x})p_1(x_1)p_2(x_2) = -4\mu_1 - 2\mu_2 + 12\mu_1\mu_2$$

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

Gradient flow in $\boldsymbol{\mu}$



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



$\nabla F(\boldsymbol{\eta})$ does not convergence to local optima, projection is 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

$$p(\boldsymbol{x};\boldsymbol{\theta}) = \exp\left\{\sum_{i=1}^{n} \theta_i x_i - \psi(\boldsymbol{\theta})\right\}$$

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

The natural parameters of the independence model $\mathcal{M}_1$ represented by an exponential family are $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_n) \in \mathbb{R}^n$, 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

$$p(\boldsymbol{x};\boldsymbol{\theta}) = \exp\left\{\sum_{i=1}^{n} \theta_i x_i - \psi(\boldsymbol{\theta})\right\}$$

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

The natural parameters of the independence model $\mathcal{M}_1$ represented by an exponential family are $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_n) \in \mathbb{R}^n$, 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 $p > 0$

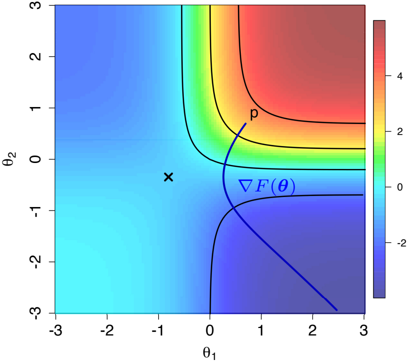
$$\theta_i = \left(\ln(\mu_i) - \ln(1-\mu_i)\right)/2 \qquad\qquad \mu_i = \frac{e^{\theta_i}}{e^{\theta_i} + e^{-\theta_i}}$$

Gradient Flows on the Independence Model

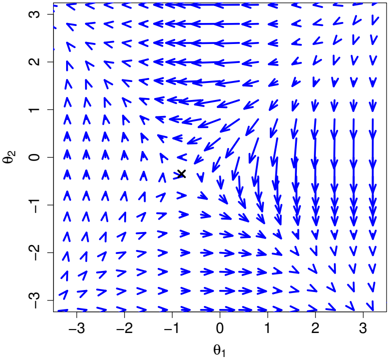
$$F(\boldsymbol{\theta}) = (-4e^{\theta_1-\theta_2} - 2e^{-\theta_1+\theta_2} + 6e^{\theta_1+\theta_2})/Z(\boldsymbol{\theta})$$

$$\nabla F(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{\theta}}[f(\boldsymbol{X} - \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{X}])] = \mathrm{Cov}_{\boldsymbol{\theta}}(f, \boldsymbol{X})$$

Gradient flow in $\boldsymbol{\theta}$



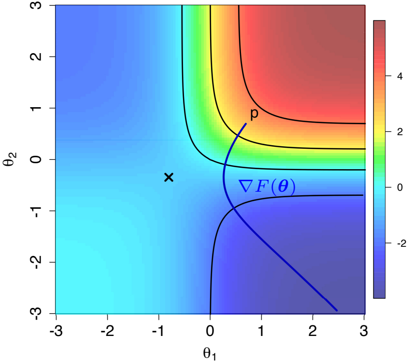
Gradient vectors in $\boldsymbol{\theta}$, $\lambda = 0.15$



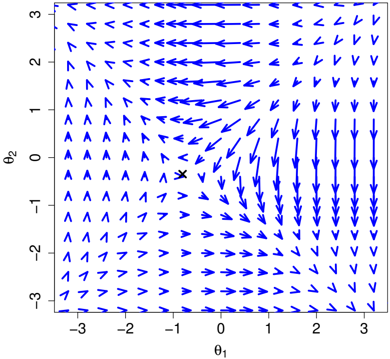
Gradient Flows on the Independence Model

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

Gradient flow in $\boldsymbol{\theta}$

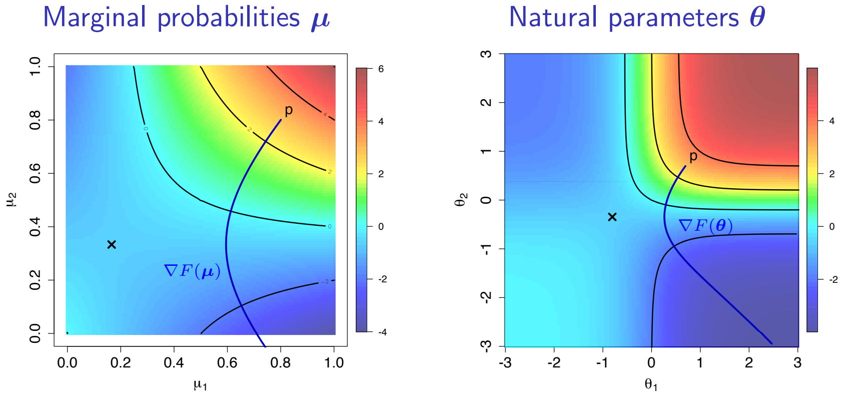


Gradient vectors in $\boldsymbol{\theta}$, $\lambda = 0.15$



In the $\boldsymbol{\theta}$ parameters, $\nabla F(\boldsymbol{\theta})$ vanishes over the plateaus

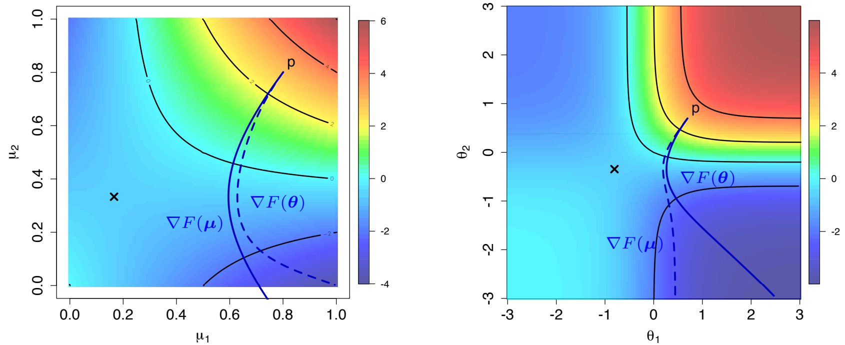
Gradient Flows on the Independence Model



Marginal probabilities $\boldsymbol{\mu}$

Natural parameters $\boldsymbol{\theta}$

Gradient Flows on the Independence Model



Marginal probabilities $\boldsymbol{\mu}$

Natural parameters $\boldsymbol{\theta}$

Gradient flows $\nabla F(\boldsymbol{\xi})$ depend on the parameterization

In the $\boldsymbol{\eta}$ parameters, $\nabla F(\boldsymbol{\eta})$ does not convergence to the expected distribution $\delta_{\boldsymbol{x}^*}$ over an optimum

Riemannian Optimization

Riemannian Optimization

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

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

Manifold structures appear in presence of symmetries in the space, invariance properties of the cost function, or in the constraints

Applications in linear algebra, signal processing, robotics, machine learning, statistics, physics, . . .

Riemannian Optimization

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

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

Manifold structures appear in presence of symmetries in the space, invariance properties of the cost function, or in the constraints

Applications in linear algebra, signal processing, robotics, machine learning, statistics, physics, . . .

Advantages of a Riemannian approach to optimization:

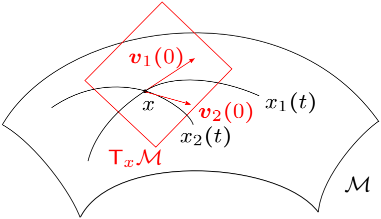
- by taking into account the structure of the problem, more effective algorithms can be developed
- a mathematical framework which provides the basis for convergence analysis of the algorithms

The Tangent Space

Suppose we have a manifold structure with coordinate charts

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

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



The tangent space can be identified by the linear space spanned by the velocity vectors to all smooth curves passing through x

Riemannian Metric

Over the tangent space is a vector space we can define an inner product called Riemannian metric

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

The inner product induces a norm

$$\|\boldsymbol{v}\|_x = \sqrt{\langle \boldsymbol{v}, \boldsymbol{v} \rangle_x}$$

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

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

A geodesic is a length-minimizing curve between two points

Riemannian Gradient

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

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

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

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

Riemannian Gradient

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

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

$$g(\mathrm{grad}\, f, \boldsymbol{v}) = \mathrm{D}_{\boldsymbol{v}}\, f,$$

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

Given a parameterization $\psi^{-1}(\boldsymbol{\theta}) \mapsto x$ for $\mathcal{M}$, let $\tilde{f} = f \circ \psi^{-1}$

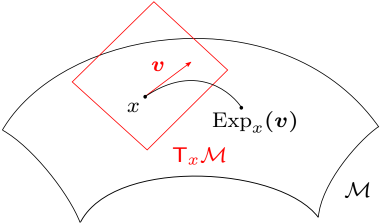
$$\mathrm{grad}\, \tilde{f}(\boldsymbol{\theta}) = \sum_{i,j} g^{ij} \frac{\partial \tilde{f}}{\partial \theta_i} \frac{\partial}{\partial \theta_j}$$

with components $\tilde{G}(\boldsymbol{\theta})^{-1} \nabla \tilde{f}(\boldsymbol{\theta})$

The Riemannian gradient depends on the metric g trough $G = [g_{ij}]$, with $G^{-1} = [g^{ij}]$

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 geodesic from x to $\mathrm{Exp}_{\boldsymbol{\theta}_t} v$



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

Riemannian Gradient Descent

Consider the naïve implementation of gradient descent on $\mathbb{R}^n$

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

This cannot be directly applied to manifolds, since we cannot sum $x \in \mathcal{M}$ and $\boldsymbol{v} \in \mathsf{T}_x\mathcal{M}$

Moreover, the gradient depends on the parameterization

Riemannian Gradient Descent

Consider the naïve implementation of gradient descent on $\mathbb{R}^n$

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

This cannot be directly applied to manifolds, since we cannot sum $x \in \mathcal{M}$ and $\boldsymbol{v} \in \mathsf{T}_x\mathcal{M}$

Moreover, the gradient depends on the parameterization

Such problem can be addressed by computing the Riemannian gradient and applying the exponential map

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

Relaxations, can be obtained using a retraction $R_{\boldsymbol{\theta}}$, a map from tangent space to the manifold

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

which requires weaker conditions compared to the exponential map

Geometry Derived by the KL Divergence

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

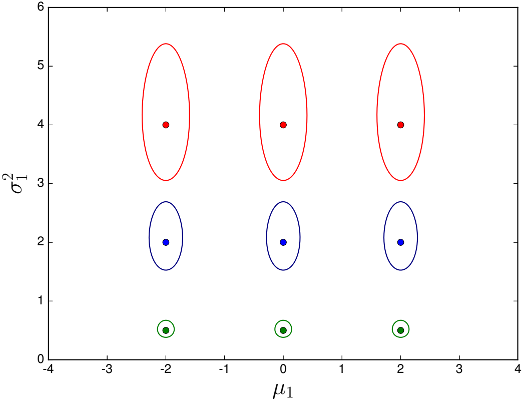
The direction of steepest ascent $\Delta\boldsymbol{\xi}$ in a Euclidean space for F can then be evaluated by minimizing $F(\boldsymbol{\xi} + \Delta\boldsymbol{\xi})$ with $\|\Delta\boldsymbol{\xi}\| = 1$

Amari replaces this contraint with the KL divergence

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

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

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

Amari's Natural Gradient (1998) 2/2

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

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

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

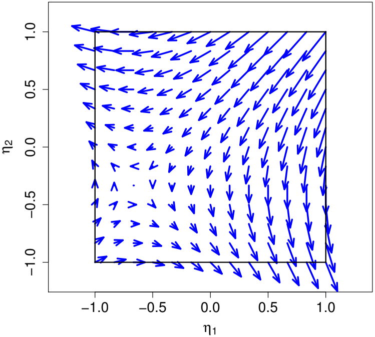
$$\begin{aligned}
&\nabla_{\Delta\boldsymbol{\xi}}\left(F(\boldsymbol{\xi}) + \nabla F(\boldsymbol{\xi})^{\mathrm{T}}\Delta\boldsymbol{\xi} - \lambda\frac{1}{2}\Delta\boldsymbol{\xi}^{\mathrm{T}} I(\boldsymbol{\xi})\Delta\boldsymbol{\xi}\right) = 0\\
&\nabla F(\boldsymbol{\xi}) - \lambda I(\boldsymbol{\xi})\Delta\boldsymbol{\xi}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

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

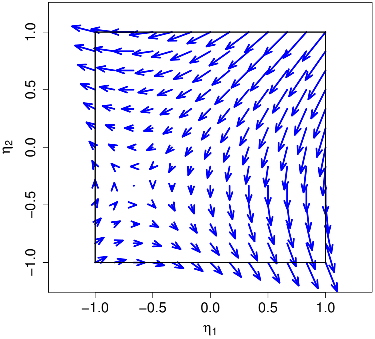
$$\widetilde{\nabla} F(\boldsymbol{\xi}) = I(\boldsymbol{\xi})^{-1}\nabla F(\boldsymbol{\xi})$$

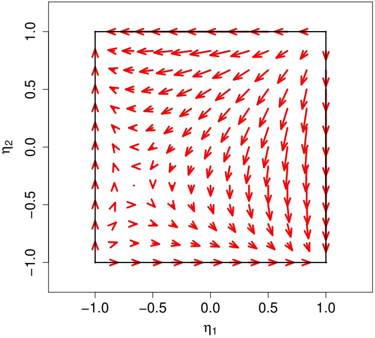
Vanilla vs Natural Gradient: $\boldsymbol{\eta}$

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



Vanilla vs Natural Gradient: $\boldsymbol{\eta}$

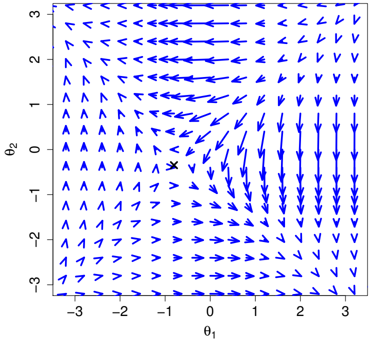

Vanilla gradient $\nabla F(\boldsymbol{\eta})$


Natural gradient $\widetilde{\nabla} F(\boldsymbol{\eta})$

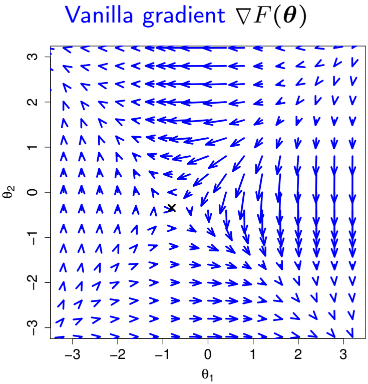
In both cases there exist two basins of attraction, however $\widetilde{\nabla} F(\boldsymbol{\eta})$ convergences to $\delta_{\boldsymbol{x}}$ distributions, which are local optima for $F(\boldsymbol{\eta})$ and where $\widetilde{\nabla} F(\delta_{\boldsymbol{x}}) = 0$

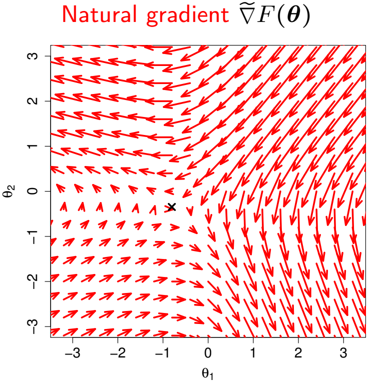
Euclidean vs Natural Gradient: $\boldsymbol{\theta}$

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



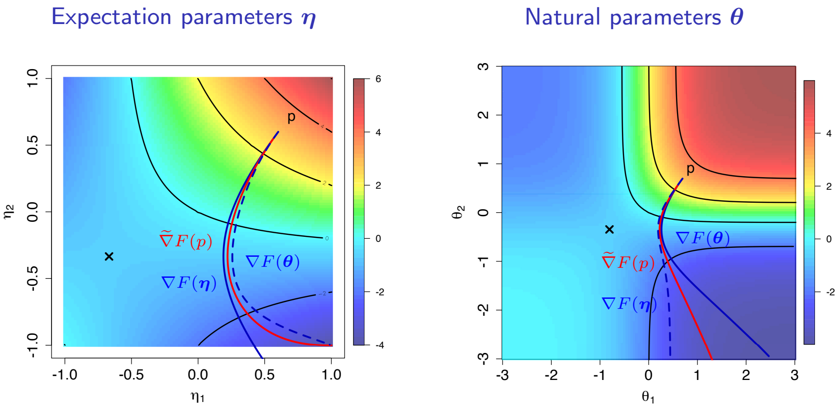
Euclidean vs Natural Gradient: $\boldsymbol{\theta}$





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

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

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

Riemannian Natural Gradient

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

Riemannian Natural Gradient

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

For each vector field U over $\mathcal{M}$, the natural gradient of $\mathcal{F}(p)$, i.e., the direction of steepest ascent is the unique vector that satisfies

$$g(\operatorname{grad} \mathcal{F}, U) = \mathrm{D}_U \mathcal{F},$$

where $\mathrm{D}_U \mathcal{F}$ is the directional derivative of $\mathcal{F}$ in the direction U

Riemannian Natural Gradient

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

For each vector field U over $\mathcal{M}$, the natural gradient of $\mathcal{F}(p)$, i.e., the direction of steepest ascent is the unique vector that satisfies

$$g(\operatorname{grad} \mathcal{F}, U) = \mathrm{D}_U \mathcal{F},$$

where $\mathrm{D}_U \mathcal{F}$ is the directional derivative of $\mathcal{F}$ in the direction U

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

$$\widetilde{\nabla} F(\boldsymbol{\xi}) = \sum_{i,j=1}^{d} g^{ij} \frac{\partial F}{\partial \xi_i} \frac{\partial}{\partial \xi_j} = I(\boldsymbol{\xi})^{-1} \nabla F(\boldsymbol{\xi})$$

Riemannian Natural Gradient

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

For each vector field U over $\mathcal{M}$, the natural gradient of $\mathcal{F}(p)$, i.e., the direction of steepest ascent is the unique vector that satisfies

$$g(\operatorname{grad}\mathcal{F}, U) = \mathrm{D}_U\,\mathcal{F},$$

where $\mathrm{D}_U\,\mathcal{F}$ is the directional derivative of $\mathcal{F}$ in the direction U

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

$$\widetilde{\nabla} F(\boldsymbol{\xi}) = \sum_{i,j=1}^{d} g^{ij} \frac{\partial F}{\partial \xi_i} \frac{\partial}{\partial \xi_j} = I(\boldsymbol{\xi})^{-1} \nabla F(\boldsymbol{\xi})$$

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

We use $\widetilde{\nabla} F(\boldsymbol{\xi})$ to distinguish the natural gradient from the vanilla gradient $\nabla F(\boldsymbol{\xi})$, i.e., the vector of partial derivatives of F w. r. t. $\boldsymbol{\xi}$

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)
- 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

Information Geometry of the Gaussian Distribution in View of Stochastic Optimization

References

L. Malagò, M. Matteucci, and G. Pistone
Towards the geometry of estimation of distribution algorithms based on the exponential family
In FOGA '11, pages 230–242, ACM, New York, NY, USA, 2011

Context and Motivation

In this section we present an application of Information Geometry to the context Random Search optimization

Context and Motivation

In this section we present an application of Information Geometry to the context Random Search optimization

We focus on algorithms where the search for the optimum is guided by some statistical model

Context and Motivation

In this section we present an application of Information Geometry to the context Random Search optimization

We focus on algorithms where the search for the optimum is guided by some statistical model

We consider algorithm based on the minimization of the expected value of a function by gradient descent techniques

Context and Motivation

In this section we present an application of Information Geometry to the context Random Search optimization

We focus on algorithms where the search for the optimum is guided by some statistical model

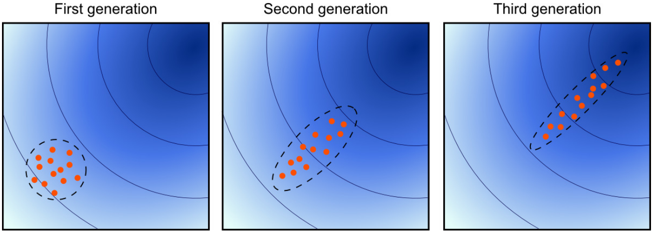
We consider algorithm based on the minimization of the expected value of a function by gradient descent techniques

We study the case of continuous domains, where statistical model are Gaussian distributions

A Small Example

A naïve model-based optimization algorithm is an iterative procedure based on

1. Sampling from a statistical model
2. Evaluation of the function over the sample
3. Updating of the parameters of the distribution



Source: Wikipedia, CMA-ES

Optimization over Statistical Models

Algorithms and heuristics in model-based optimization

- Sampling, Selection and Estimation paradigm (EDAs, Larrañaga and Lozano, 2002, CE method, Rubinstein, 1997)
- Fitness Modeling (DEUM, Shakya et al., 2005)
- Covariance Matrix Adaptation (CMA-ES, Hansen et al., 2001), Natural Evolutionary Strategies (NES, Wierstra et al., 2008)
- Boltzmann distribution and Gibbs sampler (Geman and Geman, 1984), Simulated Annealing and Boltzmann Machines (Aarts and Korst, 1989)

Optimization over Statistical Models

Algorithms and heuristics in model-based optimization

- Sampling, Selection and Estimation paradigm (EDAs, Larrañaga and Lozano, 2002, CE method, Rubinstein, 1997)
- Fitness Modeling (DEUM, Shakya et al., 2005)
- Covariance Matrix Adaptation (CMA-ES, Hansen et al., 2001), Natural Evolutionary Strategies (NES, Wierstra et al., 2008)
- Boltzmann distribution and Gibbs sampler (Geman and Geman, 1984), Simulated Annealing and Boltzmann Machines (Aarts and Korst, 1989)

Many different fields of applications

- Random Search and Stochastic Optimization
- Policy Learning in Reinforcement Learning
- Neural Networks training
- Parameter estimation in Statistics

Stochastic Relaxation in $\mathbb{R}^n$

Consider the unconstrained minimization problem of $f(\boldsymbol{x}) : \mathbb{R}^n \mapsto \mathbb{R}$

$$\text{(P)} \qquad \min_{x \in \mathbb{R}^n} f(x)$$

Stochastic Relaxation in $\mathbb{R}^n$

Consider the unconstrained minimization problem of $f(\boldsymbol{x}) : \mathbb{R}^n \mapsto \mathbb{R}$

$$\text{(P)} \qquad \min_{x \in \mathbb{R}^n} f(x)$$

Let $\mathcal{M} = \{p(\boldsymbol{x}; \boldsymbol{\xi})\}$ be a statistical model parametrized by some parameter vector $\boldsymbol{\xi} \in \Xi$

Define $F(p) = \mathbb{E}_p[f] : \mathcal{M} \mapsto \mathbb{R}$ and its parametric representation $F_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = \mathbb{E}_{\boldsymbol{\xi}}[f]$

Stochastic Relaxation in $\mathbb{R}^n$

Consider the unconstrained minimization problem of $f(\boldsymbol{x}) : \mathbb{R}^n \mapsto \mathbb{R}$

$$\text{(P)} \qquad \min_{x \in \mathbb{R}^n} f(x)$$

Let $\mathcal{M} = \{p(\boldsymbol{x};\boldsymbol{\xi})\}$ be a statistical model parametrized by some parameter vector $\boldsymbol{\xi} \in \Xi$

Define $F(p) = \mathbb{E}_p[f] : \mathcal{M} \mapsto \mathbb{R}$ and its parametric representation $F_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = \mathbb{E}_{\boldsymbol{\xi}}[f]$

We look for the minimum of f by optimizing its stochastic relaxation

$$\text{(SR)} \qquad \min_{\boldsymbol{\xi} \in \Xi} F(\boldsymbol{\xi})$$

Some hypothesis on $\mathcal{M}$ are required for (R) and (SR) to be equivalent

Gradient Descent

We need a statistical model for the relaxation of f defined over $\mathbb{R}^n$

Gradient Descent

We need a statistical model for the relaxation of f defined over $\mathbb{R}^n$

→ A natural choice is the multivariate Gaussian Distribution

$$\mathcal{M} = \mathcal{N}(\boldsymbol{\mu}, \Sigma) = \{p(\boldsymbol{x}; \boldsymbol{\xi})\} \qquad \boldsymbol{\xi} = (\boldsymbol{\mu}, \Sigma)$$

Gradient Descent

We need a statistical model for the relaxation of f defined over $\mathbb{R}^n$

→ A natural choice is the multivariate Gaussian Distribution

$$\mathcal{M} = \mathcal{N}(\boldsymbol{\mu}, \Sigma) = \{p(\boldsymbol{x}; \boldsymbol{\xi})\} \qquad \boldsymbol{\xi} = (\boldsymbol{\mu}, \Sigma)$$

We need a policy to search for densities in the model

Gradient Descent

We need a statistical model for the relaxation of f defined over $\mathbb{R}^n$

→ A natural choice is the multivariate Gaussian Distribution

$$\mathcal{M} = \mathcal{N}(\boldsymbol{\mu}, \Sigma) = \{p(\boldsymbol{x}; \boldsymbol{\xi})\} \qquad \boldsymbol{\xi} = (\boldsymbol{\mu}, \Sigma)$$

We need a policy to search for densities in the model

→ A standard approach is gradient descent

$$\boldsymbol{\xi}^{t+1} = \boldsymbol{\xi}^{t} - \lambda \widetilde{\nabla}_{\boldsymbol{\xi}} F(\boldsymbol{\xi})$$

- $\widetilde{\nabla}_{\boldsymbol{\xi}} F(\boldsymbol{\xi})$ is the natural gradient vector
- $\lambda > 0$ is step size

Gradient Descent

We need a statistical model for the relaxation of f defined over $\mathbb{R}^n$

→ A natural choice is the multivariate Gaussian Distribution

$$\mathcal{M} = \mathcal{N}(\boldsymbol{\mu}, \Sigma) = \{p(\boldsymbol{x}; \boldsymbol{\xi})\} \qquad \boldsymbol{\xi} = (\boldsymbol{\mu}, \Sigma)$$

We need a policy to search for densities in the model

→ A standard approach is gradient descent

$$\boldsymbol{\xi}^{t+1} = \boldsymbol{\xi}^{t} - \lambda \widetilde{\nabla}_{\boldsymbol{\xi}} F(\boldsymbol{\xi})$$

- $\widetilde{\nabla}_{\boldsymbol{\xi}} F(\boldsymbol{\xi})$ is the natural gradient vector
- $\lambda > 0$ is step size

In black-box contexts f is unknown and can only be evaluated

Amari's Natural Gradient

Why $\widetilde{\nabla}_{\boldsymbol{\xi}} F(\boldsymbol{\xi})$ and not just $\nabla_{\boldsymbol{\xi}} F(\boldsymbol{\xi})$?

Amari's Natural Gradient

Why $\widetilde{\nabla}_{\boldsymbol{\xi}} F(\boldsymbol{\xi})$ and not just $\nabla_{\boldsymbol{\xi}} F(\boldsymbol{\xi})$?

At this point we should know that the geometry of $\mathcal{M}$ is not Euclidean, Euclidean gradients are not invariant to reparametrization

$\widetilde{\nabla}_{\boldsymbol{\xi}} F(\boldsymbol{\xi})$ is the natural gradient, i.e., the direction of steepest descent evaluated over a statistical model

In general $\widetilde{\nabla}_{\boldsymbol{\xi}} F(\boldsymbol{\xi})$ does not coincide with the vector of partial derivatives with respect to $\boldsymbol{\xi}$ denoted by $\nabla_{\boldsymbol{\xi}} F(\boldsymbol{\xi})$

The Gaussian Distribution

The multivariate Gaussian Distribution

$$p(\boldsymbol{x}; \boldsymbol{\mu}, \Sigma) = (2\pi)^{-d/2} |\Sigma|^{-1/2} \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^{\mathrm{T}} \Sigma^{-1} (\boldsymbol{x} - \boldsymbol{\mu})\right)$$

- $\boldsymbol{\mu}$ is the mean vector
- Σ is the covariance matrix

The Gaussian Distribution

The multivariate Gaussian Distribution

$$p(\boldsymbol{x};\boldsymbol{\mu},\Sigma) = (2\pi)^{-d/2}|\Sigma|^{-1/2}\exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\mathrm{T}}\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$$

- $\boldsymbol{\mu}$ is the mean vector
- Σ is the covariance matrix

The Gaussian distribution belongs to the exponential family

$$p(\boldsymbol{x};\boldsymbol{\theta}) = \exp\left(\sum_{i=1}^{k}\boldsymbol{\theta}_i T_i(\boldsymbol{x}) - \psi(\boldsymbol{\theta})\right)$$

- $\boldsymbol{\theta}$ natural parameters
- $\{T_i\}$ sufficient statistics
- $\psi(\boldsymbol{\theta})$ log of the partition function

Change of Parameters: from $(\boldsymbol{\mu}, \Sigma)$ and $\boldsymbol{\theta}$

By writing the Gaussian distribution as an exponential family

$$p(\boldsymbol{x}; \boldsymbol{\mu}, \Sigma) = (2\pi)^{-d/2} |\Sigma|^{-1/2} \exp\left(-\frac{1}{2} (\boldsymbol{x} - \boldsymbol{\mu})^{\mathrm{T}} \Sigma^{-1} (\boldsymbol{x} - \boldsymbol{\mu}) \right)$$

we have

$$\begin{aligned} \boldsymbol{T} &= \left((X_i), (X_i^2), (2X_iX_j)_{i<j} \right)^{\mathrm{T}} \\ &= ((T_i), (T_{ii}), (T_{ij})_{i<j})^{\mathrm{T}} \end{aligned}$$

$$\begin{aligned} \boldsymbol{\theta} &= \left(\Sigma^{-1}\boldsymbol{\mu}, (-\frac{1}{2}\sigma^{ii}), (-\frac{1}{2}\sigma^{ij})_{i<j} \right)^{\mathrm{T}} \\ &= ((\theta_i), (\theta_{ii}), (\theta_{ij})_{i<j})^{\mathrm{T}} \end{aligned}$$

$$\psi(\boldsymbol{\theta}) = \frac{n}{2} \log(2\pi) - \frac{1}{4} \theta^{\mathrm{T}} \Theta^{-1} \theta - \frac{1}{2} \log|-2\Theta|$$

Change of Parameters: $(\boldsymbol{\mu}, \Sigma)$ and $\boldsymbol{\theta}$

Equivalently we represent $\boldsymbol{\theta} = (\theta; \Theta)$

$$
\begin{aligned}
\theta &= (\theta_i) = \Sigma^{-1}\boldsymbol{\mu} \\
\Theta &= \sum_i \theta_{ii} \boldsymbol{e}_i \boldsymbol{e}_i^{\mathrm{T}} + \sum_{i<j} \theta_{ij} (\boldsymbol{e}_i \boldsymbol{e}_j^{\mathrm{T}} + \boldsymbol{e}_j \boldsymbol{e}_i^{\mathrm{T}}) = -\frac{1}{2}\Sigma^{-1}
\end{aligned}
$$

so that

$$
p_{\boldsymbol{\theta}}(\boldsymbol{x}; \boldsymbol{\theta}) = \exp\left(\theta^{\mathrm{T}} \boldsymbol{x} + \boldsymbol{x}^{\mathrm{T}} \Theta \boldsymbol{x} - \psi(\boldsymbol{\theta})\right)
$$

Change of Parameters: $(\boldsymbol{\mu}, \Sigma)$ and $\boldsymbol{\theta}$

Equivalently we represent $\boldsymbol{\theta} = (\theta; \Theta)$

$$\theta = (\theta_i) = \Sigma^{-1}\boldsymbol{\mu}$$

$$\Theta = \sum_i \theta_{ii}\boldsymbol{e}_i\boldsymbol{e}_i^{\mathrm{T}} + \sum_{i<j}\theta_{ij}(\boldsymbol{e}_i\boldsymbol{e}_j^{\mathrm{T}} + \boldsymbol{e}_j\boldsymbol{e}_i^{\mathrm{T}}) = -\frac{1}{2}\Sigma^{-1}$$

so that

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}; \boldsymbol{\theta}) = \exp\left(\theta^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{x}^{\mathrm{T}}\Theta\boldsymbol{x} - \psi(\boldsymbol{\theta})\right)$$

$$\theta = (\theta_i)^{\mathrm{T}} = \Sigma^{-1}\boldsymbol{\mu}$$

$$\Theta = [\theta_{ij}] = -\frac{1}{2}\Sigma^{-1}$$

$$\boldsymbol{\mu} = -\frac{1}{2}\Theta^{-1}\theta$$

$$\Sigma = -\frac{1}{2}\Theta^{-1}$$

Dual Parameterization

Exponential families admit a dual parametrization given by the expectation parameters $\boldsymbol{\eta} = \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{T}]$

$$\begin{aligned}\boldsymbol{\eta} &= \left((\mu_i), (\sigma_{ii} - \mu_i^2), (2\sigma_{ij} - 2\mu_i\mu_j)_{i<j}\right)^{\mathrm{T}} \\ &= ((\mu_i), (\eta_{ii}), (\eta_{ij})_{i<j})^{\mathrm{T}}\end{aligned}$$

where $\varphi(\boldsymbol{\eta})$ is the negative entropy of p

$$\varphi(\boldsymbol{\eta}) = -\frac{n}{2}\left(\log(2\pi) + 1\right) - \frac{1}{2}\log|E - \eta\eta^{\mathrm{T}}|$$

Dual Parameterization

Exponential families admit a dual parametrization given by the expectation parameters $\boldsymbol{\eta} = \mathbb{E}_{\boldsymbol{\theta}}[\boldsymbol{T}]$

$$\begin{aligned}\boldsymbol{\eta} &= \left((\mu_i), (\sigma_{ii} - \mu_i^2), (2\sigma_{ij} - 2\mu_i\mu_j)_{i<j}\right)^{\mathrm{T}} \\ &= \left((\mu_i), (\eta_{ii}), (\eta_{ij})_{i<j}\right)^{\mathrm{T}}\end{aligned}$$

where $\varphi(\boldsymbol{\eta})$ is the negative entropy of p

$$\varphi(\boldsymbol{\eta}) = -\frac{n}{2}\left(\log(2\pi) + 1\right) - \frac{1}{2}\log|E - \eta\eta^{\mathrm{T}}|$$

The Gaussian distribution in the $\boldsymbol{\eta}$ parameters becomes

$$\begin{aligned}p(\boldsymbol{x}; \boldsymbol{\eta}) &= \exp\left(-\frac{1}{2}(\boldsymbol{x} - \eta)^{\mathrm{T}}(E - \eta\eta^{\mathrm{T}})^{-1}(\boldsymbol{x} - \eta) + \varphi(\boldsymbol{\eta}) + \frac{n}{2}\right) \\ &= \exp\left(\sum_{i=1}^{k}(T_i - \eta_i)\partial_i\varphi(\boldsymbol{\eta}) + \varphi(\boldsymbol{\eta})\right)\end{aligned}$$

Change of Parameters: $(\boldsymbol{\mu}, \Sigma)$ and $\boldsymbol{\theta}$

Equivalently we represent $\boldsymbol{\eta} = (\eta; E)$

$$\begin{aligned} \eta &= (\eta_i) = \boldsymbol{\mu} \ , \\ E &= \sum_i \eta_{ii} \boldsymbol{e}_i \boldsymbol{e}_i^{\mathrm{T}} + \sum_{i<j} \eta_{ij} \frac{\boldsymbol{e}_i \boldsymbol{e}_j^{\mathrm{T}} + \boldsymbol{e}_j \boldsymbol{e}_i^{\mathrm{T}}}{2} = \Sigma + \boldsymbol{\mu}\boldsymbol{\mu}^{\mathrm{T}} \end{aligned}$$

and

$$\begin{aligned} \boldsymbol{\mu} &= \eta = \mathbb{E}_{\boldsymbol{\eta}}[\boldsymbol{X}] \\ \Sigma &= E - \eta\eta^{\mathrm{T}} = \mathrm{Cov}_{\boldsymbol{\eta}}(\boldsymbol{X}, \boldsymbol{X}) \end{aligned}$$

Change of Parameters: $(\boldsymbol{\mu}, \Sigma)$, $\boldsymbol{\eta}$ and $\boldsymbol{\theta}$

$$\begin{tikzcd}
 & & \boldsymbol{\theta} \arrow[lld, bend right] \arrow[d, shift left, "\nabla_{\boldsymbol{\theta}}\psi"] \\
p & (\boldsymbol{\mu}, \Sigma) \arrow[l] \arrow[ru, "\omega"] \arrow[rd, "\chi"'] & \\
 & & \boldsymbol{\eta} \arrow[llu, bend left] \arrow[u, shift left, "\nabla_{\boldsymbol{\eta}}\varphi"]
\end{tikzcd}$$

Variable transformations are given by

$$\boldsymbol{\eta} = \nabla_{\boldsymbol{\theta}}\psi(\boldsymbol{\theta}) = (\nabla_{\boldsymbol{\eta}}\varphi)^{-1}(\boldsymbol{\theta})$$

$$\boldsymbol{\theta} = \nabla_{\boldsymbol{\eta}}\varphi(\boldsymbol{\eta}) = (\nabla_{\boldsymbol{\theta}}\psi)^{-1}(\boldsymbol{\eta})$$

Change of Parameters: $(\boldsymbol{\mu}, \Sigma)$, $\boldsymbol{\eta}$ and $\boldsymbol{\theta}$

$$\begin{tikzcd}
 & & \boldsymbol{\theta} \arrow[dll, bend right] \arrow[dd, shift left, "\nabla_{\boldsymbol{\theta}}\psi"] \\
p & (\boldsymbol{\mu}, \Sigma) \arrow[l] \arrow[ur, "\omega"] \arrow[dr, "\chi"'] & \\
 & & \boldsymbol{\eta} \arrow[ull, bend left] \arrow[uu, shift left, "\nabla_{\boldsymbol{\eta}}\varphi"]
\end{tikzcd}$$

Variable transformations are given by

$$\boldsymbol{\eta} = \nabla_{\boldsymbol{\theta}}\psi(\boldsymbol{\theta}) = (\nabla_{\boldsymbol{\eta}}\varphi)^{-1}(\boldsymbol{\theta})$$
$$\boldsymbol{\theta} = \nabla_{\boldsymbol{\eta}}\varphi(\boldsymbol{\eta}) = (\nabla_{\boldsymbol{\theta}}\psi)^{-1}(\boldsymbol{\eta})$$

The $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$ are connected by the Legendre transform

$$\psi(\boldsymbol{\theta}) + \varphi(\boldsymbol{\eta}) - \langle \boldsymbol{\theta}, \boldsymbol{\eta} \rangle = 0$$

Fisher Information Matrix

The geometry of $\mathcal{M}$ is not Euclidean and the metric tensor is given by the Fisher information matrix

$$I_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = \mathbb{E}_{\boldsymbol{\xi}}\left[(\partial_i \log p(\boldsymbol{x};\boldsymbol{\xi}))(\partial_j \log p(\boldsymbol{x};\boldsymbol{\xi}))^{\mathrm{T}}\right]$$

with $\partial_i = \partial / \partial \xi_i$

Fisher Information Matrix

The geometry of $\mathcal{M}$ is not Euclidean and the metric tensor is given by the Fisher information matrix

$$I_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = \mathbb{E}_{\boldsymbol{\xi}}\left[(\partial_i \log p(\boldsymbol{x};\boldsymbol{\xi}))(\partial_j \log p(\boldsymbol{x};\boldsymbol{\xi}))^{\mathrm{T}}\right]$$

with $\partial_i = \partial/\partial\xi_i$

Under certain regularity conditions, an equivalent formulation is given by

$$I_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = -\mathbb{E}_{\boldsymbol{\xi}}\left[\partial_i\partial_j \log p(\boldsymbol{x};\boldsymbol{\xi})\right]$$

Fisher Information Matrix

The geometry of $\mathcal{M}$ is not Euclidean and the metric tensor is given by the Fisher information matrix

$$I_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = \mathbb{E}_{\boldsymbol{\xi}}\left[(\partial_i \log p(\boldsymbol{x};\boldsymbol{\xi}))(\partial_j \log p(\boldsymbol{x};\boldsymbol{\xi}))^{\mathrm{T}}\right]$$

with $\partial_i = \partial/\partial\xi_i$

Under certain regularity conditions, an equivalent formulation is given by

$$I_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = -\mathbb{E}_{\boldsymbol{\xi}}\left[\partial_i\partial_j \log p(\boldsymbol{x};\boldsymbol{\xi})\right]$$

In the exponential family, the Fisher information in $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$ can be evaluated as the Hessian of the dual functions

$$I_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \mathrm{Hess}\,\psi(\boldsymbol{\theta})$$
$$I_{\boldsymbol{\eta}}(\boldsymbol{\eta}) = \mathrm{Hess}\,\varphi(\boldsymbol{\eta})$$

Fisher Information Matrix

In the Gaussian distribution the Fisher information matrix admits a closed formula (Miller, 1974)

Fisher Information Matrix

In the Gaussian distribution the Fisher information matrix admits a closed formula (Miller, 1974)

Let $\boldsymbol{\mu}$ and Σ be a function of the parameter vector $\boldsymbol{\xi}$

$$I_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = \left[(\partial_i \boldsymbol{\mu})^{\mathrm{T}} \Sigma^{-1} (\partial_j \boldsymbol{\mu}) + \frac{1}{2} \operatorname{Tr}\left(\Sigma^{-1} (\partial_i \Sigma) \Sigma^{-1} (\partial_j \Sigma) \right) \right]_{ij}$$

Fisher Information Matrix

In the Gaussian distribution the Fisher information matrix admits a closed formula (Miller, 1974)

Let $\boldsymbol{\mu}$ and Σ be a function of the parameter vector $\boldsymbol{\xi}$

$$I_{\boldsymbol{\xi}}(\boldsymbol{\xi}) = \left[(\partial_i \boldsymbol{\mu})^{\mathrm{T}} \Sigma^{-1} (\partial_j \boldsymbol{\mu}) + \frac{1}{2} \operatorname{Tr}\left(\Sigma^{-1} (\partial_i \Sigma) \Sigma^{-1} (\partial_j \Sigma) \right) \right]_{ij}$$

When $\boldsymbol{\mu}$ and Σ depend on disjoint sets of parameters, such as in the mean and covariance parameterization, $I_{\boldsymbol{\xi}}(\boldsymbol{\xi})$ becomes block diagonal

Fisher Information Matrix in $(\boldsymbol{\mu}; \Sigma)$

$$I_{\boldsymbol{\mu},\Sigma}(\boldsymbol{\mu}; \Sigma) = \begin{array}{c|c|c} & i & kl \\ j & \Sigma^{-1} & 0 \\ \hline mn & 0 & \alpha_{klmn} \end{array}$$

$$\alpha_{klmn} = \begin{cases} \frac{1}{2}(\sigma^{kk})^2 \ , & \text{if } k = l = m = n \\ \sigma^{km}\sigma^{ln} \ , & \text{if } k = l \veebar m = n \\ \sigma^{km}\sigma^{ln} + \sigma^{lm}\sigma^{kn} \ , & \text{otherwise} \end{cases}$$

Fisher Information Matrix in $\boldsymbol{\theta}$

$$I_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \frac{1}{2} \times \begin{array}{cc} & \begin{array}{cc} \quad i \quad & \qquad kl \qquad \end{array} \\ \begin{array}{c} j \\ mn \end{array} & \left[\begin{array}{c|c} -\Theta^{-1} & \Lambda_{kl}\theta \\ \hline \theta^{\mathrm{T}}\Lambda_{mn} & \lambda_{klmn} - \theta^{\mathrm{T}}\Lambda_{klmn}\theta \end{array}\right] \end{array}$$

$$\Lambda_{kl} = \begin{cases} [\Theta^{-1}]_{\cdot k}[\Theta^{-1}]_{k\cdot}\ , & \text{if } k = l \\ [\Theta^{-1}]_{\cdot k}[\Theta^{-1}]_{l\cdot} + [\Theta^{-1}]_{\cdot l}[\Theta^{-1}]_{k\cdot} & \text{otherwise} \end{cases}$$

$$\lambda_{klmn} = \begin{cases} [\Theta^{-1}]_{kk}[\Theta^{-1}]_{kk}\ , & \text{if } k = l = m = n \\ [\Theta^{-1}]_{km}[\Theta^{-1}]_{ln} + [\Theta^{-1}]_{lm}[\Theta^{-1}]_{kn}\ , & \text{if } k = l \veebar m = n \\ 2\left([\Theta^{-1}]_{km}[\Theta^{-1}]_{ln} + [\Theta^{-1}]_{lm}[\Theta^{-1}]_{kn}\right) & \text{otherwise} \end{cases}$$

$$\Lambda_{klmn} = \begin{cases} [\Lambda_{kk}]_{\cdot m}[\Theta^{-1}]_{n\cdot}\ , & \text{if } k = l \\ [\Lambda_{kl}]_{\cdot m}[\Theta^{-1}]_{n\cdot} + [\Lambda_{kl}]_{\cdot n}[\Theta^{-1}]_{m\cdot}\ , & \text{otherwise} \end{cases}$$

Fisher Information Matrix in $\boldsymbol{\eta}$ 1/2

$$I_{\boldsymbol{\eta}}(\boldsymbol{\eta}) = \begin{array}{c} \\ j \\ mn \end{array}\left[\begin{array}{c|c} i & kl \\ \Gamma & -K_{kl}\eta \\ \hline -\eta^{\mathrm{T}}K_{mn} & \kappa_{klmn} \end{array}\right]$$

$$\begin{aligned}\Gamma = (E-\eta\eta^{\mathrm{T}})^{-1} + (E-\eta\eta^{\mathrm{T}})^{-1}\eta^{\mathrm{T}}(E-\eta\eta^{\mathrm{T}})^{-1}\eta + \\ + (E-\eta\eta^{\mathrm{T}})^{-1}\eta\eta^{\mathrm{T}}(E-\eta\eta^{\mathrm{T}})^{-1}\end{aligned}$$

$$K_{kl} = \begin{cases} [(E-\eta\eta^{\mathrm{T}})^{-1}]_{\cdot k}[(E-\eta\eta^{\mathrm{T}})^{-1}]_{k\cdot}\ , & \text{if } k=l \\ \frac{1}{2}\Big([(E-\eta\eta^{\mathrm{T}})^{-1}]_{\cdot k}[(E-\eta\eta^{\mathrm{T}})^{-1}]_{l\cdot} + & \\ \qquad + [(E-\eta\eta^{\mathrm{T}})^{-1}]_{\cdot l}[(E-\eta\eta^{\mathrm{T}})^{-1}]_{k\cdot}\Big) & \text{otherwise} \end{cases}$$

Fisher Information Matrix in $\boldsymbol{\eta}$ 2/2

$$I_{\boldsymbol{\eta}}(\boldsymbol{\eta}) = \begin{array}{cc} & \begin{array}{cc} \qquad i \qquad & \qquad kl \qquad \end{array} \\ \begin{array}{c} j \\ \\ mn \end{array} & \left[\begin{array}{c|c} \Gamma & -K_{kl}\eta \\ \hline -\eta^{\mathrm{T}}K_{mn} & \kappa_{klmn} \end{array}\right] \end{array}$$

$$\kappa_{klmn} = \begin{cases} \frac{1}{2}[(E-\eta\eta^{\mathrm{T}})^{-1}]_{kk}[(E-\eta\eta^{\mathrm{T}})^{-1}]_{kk}\ , & \text{if } k = l = m = n \\ \frac{1}{2}[(E-\eta\eta^{\mathrm{T}})^{-1}]_{km}[(E-\eta\eta^{\mathrm{T}})^{-1}]_{ln}\ , & \text{if } k = l \veebar m = n \\ \frac{1}{4}\Big([(E-\eta\eta^{\mathrm{T}})^{-1}]_{km}[(E-\eta\eta^{\mathrm{T}})^{-1}]_{ln} + & \\ \qquad + [(E-\eta\eta^{\mathrm{T}})^{-1}]_{lm}[(E-\eta\eta^{\mathrm{T}})^{-1}]_{kn}\Big) & \text{otherwise.} \end{cases}$$

Reparameterization of $I_{\boldsymbol{\xi}}$

- The Fisher information matrix can be reparametrized using the chain rule for differentiation

$$I_{\boldsymbol{\eta}}(\boldsymbol{\eta}) = J(\nabla\varphi)(\boldsymbol{\eta})^{\mathrm{T}}\ (I_{\boldsymbol{\theta}} \circ \nabla\varphi)(\boldsymbol{\eta})\ J(\nabla\varphi)(\boldsymbol{\eta})$$

- Let J be the Jacobian of the variable transformation

$$\begin{aligned} J(\nabla\psi)(\boldsymbol{\theta}) &= (I_{\boldsymbol{\eta}} \circ \nabla\psi)(\boldsymbol{\theta})^{-1} \\ J(\nabla\varphi)(\boldsymbol{\eta}) &= (I_{\boldsymbol{\theta}} \circ \nabla\varphi)(\boldsymbol{\eta})^{-1} \end{aligned}$$

Reparameterization of $I_{\boldsymbol{\xi}}$

- The Fisher information matrix can be reparametrized using the chain rule for differentiation

$$I_{\boldsymbol{\eta}}(\boldsymbol{\eta}) = J(\nabla\varphi)(\boldsymbol{\eta})^{\mathrm{T}}\ (I_{\boldsymbol{\theta}} \circ \nabla\varphi)(\boldsymbol{\eta})\ J(\nabla\varphi)(\boldsymbol{\eta})$$

- Let J be the Jacobian of the variable transformation

$$J(\nabla\psi)(\boldsymbol{\theta}) = (I_{\boldsymbol{\eta}} \circ \nabla\psi)(\boldsymbol{\theta})^{-1}$$
$$J(\nabla\varphi)(\boldsymbol{\eta}) = (I_{\boldsymbol{\theta}} \circ \nabla\varphi)(\boldsymbol{\eta})^{-1}$$

- By applying a transformation between one parameterization and the other we obtain

$$I_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = (I_{\boldsymbol{\eta}} \circ \nabla\psi)(\boldsymbol{\theta})^{-1} = I_{\boldsymbol{\eta}}(\boldsymbol{\eta})^{-1}$$
$$I_{\boldsymbol{\eta}}(\boldsymbol{\eta}) = (I_{\boldsymbol{\theta}} \circ \nabla\varphi)(\boldsymbol{\eta})^{-1} = I_{\boldsymbol{\theta}}(\boldsymbol{\theta})^{-1}$$

Reparameterization of $I_{\boldsymbol{\xi}}$

- The Fisher information matrix can be reparametrized using the chain rule for differentiation

$$I_{\boldsymbol{\eta}}(\boldsymbol{\eta}) = J(\nabla\varphi)(\boldsymbol{\eta})^{\mathrm{T}}\ (I_{\boldsymbol{\theta}} \circ \nabla\varphi)(\boldsymbol{\eta})\ J(\nabla\varphi)(\boldsymbol{\eta})$$

- Let J be the Jacobian of the variable transformation

$$J(\nabla\psi)(\boldsymbol{\theta}) = (I_{\boldsymbol{\eta}} \circ \nabla\psi)(\boldsymbol{\theta})^{-1}$$
$$J(\nabla\varphi)(\boldsymbol{\eta}) = (I_{\boldsymbol{\theta}} \circ \nabla\varphi)(\boldsymbol{\eta})^{-1}$$

- By applying a transformation between one parameterization and the other we obtain

$$\begin{tikzcd}
\boldsymbol{\theta} \arrow[r, "\operatorname{Hess}\psi"] \arrow[d, shift left, "\nabla_{\boldsymbol{\theta}}\psi"] & I_{\boldsymbol{\theta}} \arrow[d, shift left, "(\mathbf{1}_{\mathcal{I}})^{-1}"] \\
\boldsymbol{\eta} \arrow[r, "\operatorname{Hess}\varphi"'] \arrow[u, shift left, "\nabla_{\boldsymbol{\eta}}\varphi"] & I_{\boldsymbol{\eta}} \arrow[u, shift left, "(\mathbf{1}_{\mathcal{I}})^{-1}"]
\end{tikzcd}$$

Duality Between Natural and Vanilla Gradients

- Form previous relationships we obtain

$$\widetilde{\nabla}_{\boldsymbol{\eta}} F(\boldsymbol{\eta}) = (\nabla_{\boldsymbol{\theta}} F \circ \nabla \varphi)(\boldsymbol{\eta}) = \nabla_{\boldsymbol{\theta}} F(\boldsymbol{\theta})$$
$$\widetilde{\nabla}_{\boldsymbol{\theta}} F(\boldsymbol{\theta}) = (\nabla_{\boldsymbol{\eta}} F \circ \nabla \psi)(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\eta}} F(\boldsymbol{\eta})$$

Duality Between Natural and Vanilla Gradients

- Form previous relationships we obtain

$$\widetilde{\nabla}_{\boldsymbol{\eta}} F(\boldsymbol{\eta}) = (\nabla_{\boldsymbol{\theta}} F \circ \nabla \varphi)(\boldsymbol{\eta}) = \nabla_{\boldsymbol{\theta}} F(\boldsymbol{\theta})$$
$$\widetilde{\nabla}_{\boldsymbol{\theta}} F(\boldsymbol{\theta}) = (\nabla_{\boldsymbol{\eta}} F \circ \nabla \psi)(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\eta}} F(\boldsymbol{\eta})$$

Indeed, we have

$$\widetilde{\nabla}_{\boldsymbol{\eta}} F(\boldsymbol{\eta}) = I_{\boldsymbol{\eta}}(\boldsymbol{\eta})^{-1} \nabla_{\boldsymbol{\eta}} F(\boldsymbol{\eta})$$
$$\widetilde{\nabla}_{\boldsymbol{\eta}} F(\boldsymbol{\eta}) = (\nabla_{\boldsymbol{\theta}} F \circ \nabla \varphi)(\boldsymbol{\eta})$$

Duality Between Natural and Vanilla Gradients

- Form previous relationships we obtain

$$\widetilde{\nabla}_{\boldsymbol{\eta}} F(\boldsymbol{\eta}) = (\nabla_{\boldsymbol{\theta}} F \circ \nabla \varphi)(\boldsymbol{\eta}) = \nabla_{\boldsymbol{\theta}} F(\boldsymbol{\theta})$$
$$\widetilde{\nabla}_{\boldsymbol{\theta}} F(\boldsymbol{\theta}) = (\nabla_{\boldsymbol{\eta}} F \circ \nabla \psi)(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\eta}} F(\boldsymbol{\eta})$$

Indeed, we have

$$\widetilde{\nabla}_{\boldsymbol{\eta}} F(\boldsymbol{\eta}) = I_{\boldsymbol{\eta}}(\boldsymbol{\eta})^{-1} \nabla_{\boldsymbol{\eta}} F(\boldsymbol{\eta})$$
$$\widetilde{\nabla}_{\boldsymbol{\eta}} F(\boldsymbol{\eta}) = (\nabla_{\boldsymbol{\theta}} F \circ \nabla \varphi)(\boldsymbol{\eta})$$

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

$$\widetilde{\nabla}_{\boldsymbol{\eta}} F(\boldsymbol{\eta}) = \operatorname{Cov}_{\boldsymbol{\eta}}(f, \boldsymbol{T}) = \mathbb{E}_{\boldsymbol{\eta}}[f(\boldsymbol{T} - \boldsymbol{\eta})]$$

Stochastic Natural Gradient Descent

Due to the properties of the exponential family

$$\begin{aligned}\operatorname{Hess}\psi(\boldsymbol{\theta}) &= I_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \operatorname{Cov}_{\boldsymbol{\theta}}(\boldsymbol{T},\boldsymbol{T}) \\ \nabla F(\boldsymbol{\theta}) &= \operatorname{Cov}(f,\boldsymbol{T}) \\ \widetilde{\nabla} F(\boldsymbol{\theta}) &= \operatorname{Cov}_{\boldsymbol{\theta}}(\boldsymbol{T},\boldsymbol{T})^{-1}\operatorname{Cov}(f,\boldsymbol{T})\end{aligned}$$

For the Gaussian distribution in particular

$$\widetilde{\nabla} F(\boldsymbol{\theta}) = I_{\boldsymbol{\eta}}(\boldsymbol{\theta})\operatorname{Cov}(f,\boldsymbol{T})$$

Stochastic Natural Gradient Descent

Due to the properties of the exponential family

$$\begin{aligned} \operatorname{Hess}\psi(\boldsymbol{\theta}) &= I_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \operatorname{Cov}_{\boldsymbol{\theta}}(\boldsymbol{T},\boldsymbol{T}) \\ \nabla F(\boldsymbol{\theta}) &= \operatorname{Cov}(f,\boldsymbol{T}) \\ \widetilde{\nabla} F(\boldsymbol{\theta}) &= \operatorname{Cov}_{\boldsymbol{\theta}}(\boldsymbol{T},\boldsymbol{T})^{-1}\operatorname{Cov}(f,\boldsymbol{T}) \end{aligned}$$

For the Gaussian distribution in particular

$$\widetilde{\nabla} F(\boldsymbol{\theta}) = I_{\boldsymbol{\eta}}(\boldsymbol{\theta})\operatorname{Cov}(f,\boldsymbol{T})$$

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

Stochastic Natural Gradient Descent

Due to the properties of the exponential family

$$\begin{aligned}
\operatorname{Hess}\psi(\boldsymbol{\theta}) &= I_{\boldsymbol{\theta}}(\boldsymbol{\theta}) = \operatorname{Cov}_{\boldsymbol{\theta}}(\boldsymbol{T},\boldsymbol{T}) \\
\nabla F(\boldsymbol{\theta}) &= \operatorname{Cov}(f,\boldsymbol{T}) \\
\widetilde{\nabla} F(\boldsymbol{\theta}) &= \operatorname{Cov}_{\boldsymbol{\theta}}(\boldsymbol{T},\boldsymbol{T})^{-1}\operatorname{Cov}(f,\boldsymbol{T})
\end{aligned}$$

For the Gaussian distribution in particular

$$\widetilde{\nabla} F(\boldsymbol{\theta}) = I_{\boldsymbol{\eta}}(\boldsymbol{\theta})\operatorname{Cov}(f,\boldsymbol{T})$$

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

Gradients can be estimated from a sample by means of Monte Carlo methods

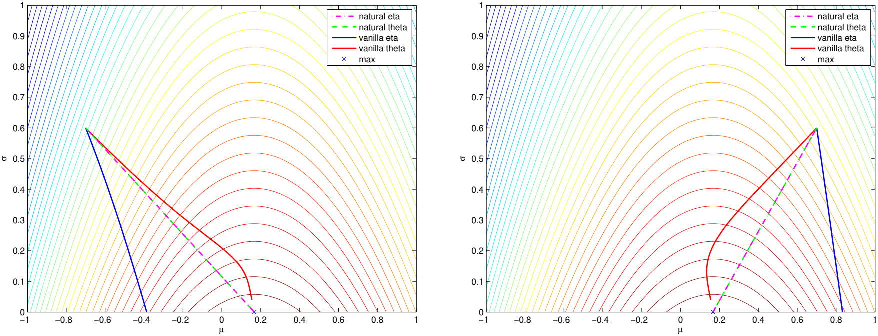
Vanilla and Natural Gradient

- Vanilla and Natural gradient flows are the solutions of the following differential equations given an initial condition

$$\dot{\boldsymbol{\xi}}(t) = \nabla_{\xi} F(\boldsymbol{\xi}(t)) \qquad\qquad \dot{\boldsymbol{\xi}}(t) = \widetilde{\nabla}_{\xi} F(\boldsymbol{\xi}(t)$$

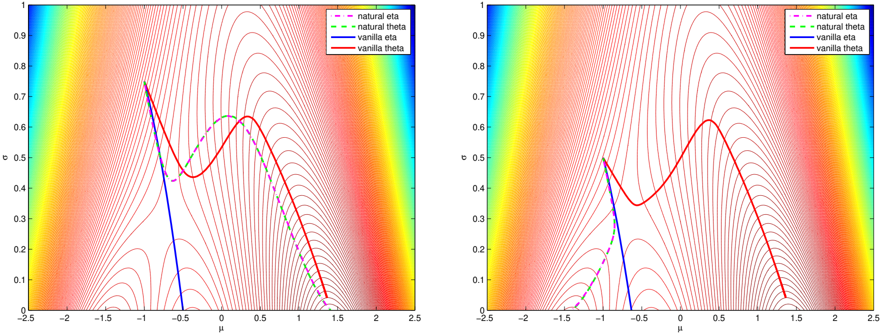
- Notice that flows represent the expected behavior of an algorithm for infinite sample size, when the step size is infinitesimal

Quadratic functions in $\mathbb{R}$



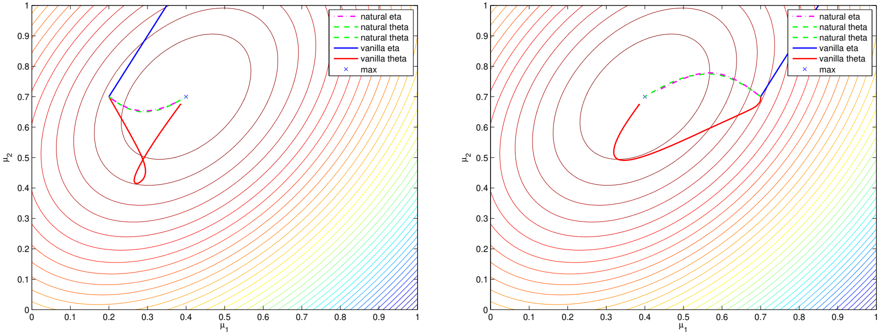
Vanilla and natural flows for $f = x - 3x^2$, represented in the parameter space (μ, σ). The level lines are associated to $\mathbb{E}_{\mu,\sigma}[f]$.

Quartic functions in $\mathbb{R}$



Vanilla and natural flows for $f = 6x + 8x^2 - x^3 - 2x^4$, represented in the parameter space (μ, σ). The level lines are associated to $\mathbb{E}_{\mu,\sigma}[f]$.

Quadratic functions in $\mathbb{R}^2$



Vanilla and natural flows for $f = x_1 + 2x_2 - 3x_1^2 - 2x_1x_2 - 2x_2^2$ projected onto (μ_1, μ_2), with $\sigma_{11} = 1, \sigma_{12} = -0.5, \sigma_{22} = 2$. Level lines associated to f.

Optimization in the $\boldsymbol{\theta}$ parameters

Solving the Stochastic Relaxation in the $\boldsymbol{\theta}$ parameters provides a natural way to identify sub-models

Optimization in the $\boldsymbol{\theta}$ parameters

Solving the Stochastic Relaxation in the $\boldsymbol{\theta}$ parameters provides a natural way to identify sub-models

The inverse covariance matrix $\Sigma^{-1} = -2\Theta = -2[\theta_{ij}]$ is the precision or concentration matrix

Optimization in the $\boldsymbol{\theta}$ parameters

Solving the Stochastic Relaxation in the $\boldsymbol{\theta}$ parameters provides a natural way to identify sub-models

The inverse covariance matrix $\Sigma^{-1} = -2\Theta = -2[\theta_{ij}]$ is the precision or concentration matrix

By fixing some $\theta_{ij} = 0$ we are identifying a lower-dimensional exponential model in the Gaussian distribution

Optimization in the $\boldsymbol{\theta}$ parameters

Solving the Stochastic Relaxation in the $\boldsymbol{\theta}$ parameters provides a natural way to identify sub-models

The inverse covariance matrix $\Sigma^{-1} = -2\Theta = -2[\theta_{ij}]$ is the precision or concentration matrix

By fixing some $\theta_{ij} = 0$ we are identifying a lower-dimensional exponential model in the Gaussian distribution

A zero entry $\theta_{ij} = 0$ implies conditional independence among X_i and X_j given all the other variables, so that the sub-model has a statistical interpretation

Convergence of Markov Random Fields

[Theorem FOGA'15] For lower-bounded, lower semicontinuous f, with compact level sets, such that $f \in \mathrm{Span}\{T_1, \dots, T_k\}$, i.e.,

$$f = \sum_{i=1}^{k} c_i T_i + c_0$$

then the limits of natural gradient flows over the exponential family with sufficient statistics $\{T_i\}$ weakly converge and are supported by the closed set where f reaches $\operatorname{ess\,inf} f$. If the minimum is unique, we have global convergence to the delta mass at the minimum

Convergence of Markov Random Fields

[Theorem FOGA'15] For lower-bounded, lower semicontinuous f, with compact level sets, such that $f \in \mathrm{Span}\{T_1, \ldots, T_k\}$, i.e.,

$$f = \sum_{i=1}^{k} c_i T_i + c_0$$

then the limits of natural gradient flows over the exponential family with sufficient statistics $\{T_i\}$ weakly converge and are supported by the closed set where f reaches $\operatorname{ess\,inf} f$. If the minimum is unique, we have global convergence to the delta mass at the minimum

The result applies to the Gaussian distribution, and suggests to choose lower-dimensional models when f has sparse interactions.

Applications to the training of Neural Networks

References

C. Varady, R. Volpi, L. Malagò, and N. Ay.
Natural Wake-Sleep Algorithm

Neural Networks, 155 (2022)

Sigmoid Belief Networks

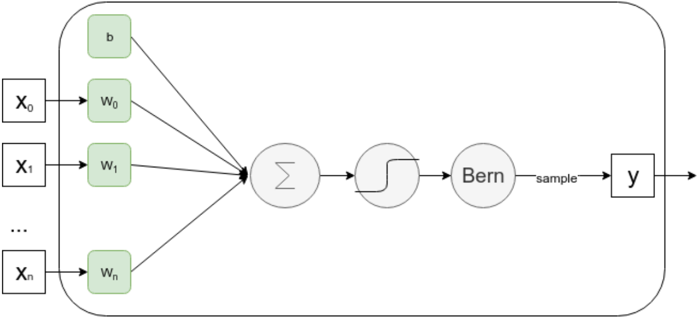


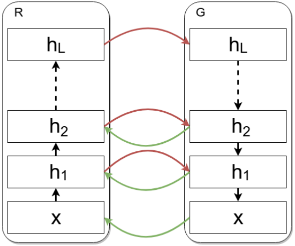
Figure: Node of a Sigmoid Belief Network

$$p(y|x) = \rho^{y}(1-\rho)^{1-y}$$

$$\rho = \sigma(b + xW^{T})$$

Wake-Sleep Algorithm

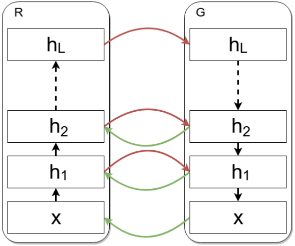
The Helmholtz [Machine Dayan et al., 1995] is a Sigmoid Belief Network constructed from a Generative and Recognition network to optimize the Helmholtz Free Energy. The Wake-Sleep algorithm [Hinton et al., 1995]



Helmholtz Machine and the Wake-Sleep Algorithm

Wake-Sleep Algorithm

The Helmholtz [Machine Dayan et al., 1995] is a Sigmoid Belief Network constructed from a Generative and Recognition network to optimize the Helmholtz Free Energy. The Wake-Sleep algorithm [Hinton et al., 1995]

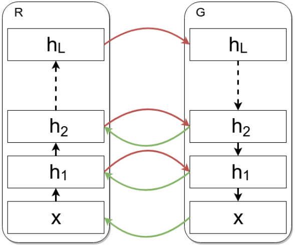


Helmholtz Machine and the Wake-Sleep Algorithm

- Alternatively optimizes the Generation network p and Recognition network q through Wake and Sleep cycles.

Wake-Sleep Algorithm

The Helmholtz [Machine Dayan et al., 1995] is a Sigmoid Belief Network constructed from a Generative and Recognition network to optimize the Helmholtz Free Energy. The Wake-Sleep algorithm [Hinton et al., 1995]



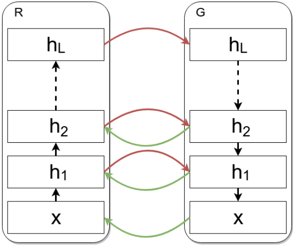
Helmholtz Machine and the Wake-Sleep Algorithm

- Alternatively optimizes the Generation network p and Recognition network q through Wake and Sleep cycles.
- Wake: Updates the weights of the Generation Network, optimizes:

$$\mathcal{L}_p(\theta, x \sim \mathcal{D}) = \underset{h,x\sim q_\phi(h|x)}{-\mathbb{E}}\Big[\ln p(x,h)\Big]$$

Wake-Sleep Algorithm

The Helmholtz [Machine Dayan et al., 1995] is a Sigmoid Belief Network constructed from a Generative and Recognition network to optimize the Helmholtz Free Energy. The Wake-Sleep algorithm [Hinton et al., 1995]



Helmholtz Machine and the Wake-Sleep Algorithm

- Alternatively optimizes the Generation network p and Recognition network q through Wake and Sleep cycles.
- Wake: Updates the weights of the Generation Network, optimizes:

$$\mathcal{L}_p(\theta, x \sim \mathcal{D}) = -\underset{h,x \sim q_\phi(h|x)}{\mathbb{E}}\left[\ln p(x,h)\right]$$

- Sleep: Updates the weights of the Recognition Network, optimizes:

$$\mathcal{L}_q(\phi, (x,h)) = -\underset{h,x \sim p_\theta(x,h)}{\mathbb{E}}\left[\ln q(h|x)\right]$$

- Convergence properties were studied by [Ikeda et al., 1999]

The Fisher Matrix of a Helmholtz Machine

Natural Gradient follows the steepest descent by computing the inverse of the Fisher Information Matrix

$$\mathbf{F} = -\mathbb{E}_{p_\theta(x,h)}\left[\nabla_\theta^2 \ln p_\theta(x,h)\right] = -\mathbb{E}_{p_\theta(x,h)}\left[\sum_{\substack{r,s \in N_i \\ i \in [0,L]}} \delta_{rs} \nabla_{\theta_r} \nabla_{\theta_s} \ln p(h_r^i | h^{i+1}; \theta_r)\right]$$

The Fisher Matrix of a Helmholtz Machine

Natural Gradient follows the steepest descent by computing the inverse of the Fisher Information Matrix

$$\mathbf{F} = -\mathbb{E}_{p_\theta(x,h)}\left[\nabla^2_\theta \ln p_\theta(x,h)\right] = -\mathbb{E}_{p_\theta(x,h)}\left[\sum_{\substack{r,s\in N_i\\ i\in[0,L]}} \delta_{rs}\nabla_{\theta_r}\nabla_{\theta_s}\ln p(h^i_r|h^{i+1};\theta_r)\right]$$

The Fisher Matrix has been shown to be block-diagonal for specific architectures [Ay, 2002], in particular for the HM we can demonstrate

$$\mathbf{F}^i_{p,j} = \mathbb{E}_{p(x,h)}\left[\sigma'\left(W^i_j h^{i+1}\right) h^{i+1}{h^{i+1}}^{\mathrm{T}}\right],$$

$$\mathbf{F}^i_{q,j} = \mathbb{E}_{q(x,h)}\left[\sigma'\left(V^i_j h^{i-1}\right) h^{i-1}{h^{i-1}}^{\mathrm{T}}\right].$$

The Fisher Matrix of a Helmholtz Machine

Natural Gradient follows the steepest descent by computing the inverse of the Fisher Information Matrix

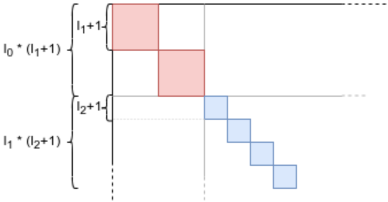
$$\mathbf{F} = -\mathbb{E}_{p_\theta(x,h)}\left[\nabla_\theta^2 \ln p_\theta(x,h)\right] = -\mathbb{E}_{p_\theta(x,h)}\Bigg[\sum_{\substack{r,s\in N_i\\ i\in[0,L]}} \delta_{rs}\nabla_{\theta_r}\nabla_{\theta_s}\ln p(h_r^i|h^{i+1};\theta_r)\Bigg]$$

The Fisher Matrix has been shown to be block-diagonal for specific architectures [Ay, 2002], in particular for the HM we can demonstrate

$$\mathbf{F}^i_{p,j} = \mathbb{E}_{p(x,h)}\Bigg[\sigma^{'}\left(W_j^i h^{i+1}\right) h^{i+1}{h^{i+1}}^{\mathrm{T}}\Bigg],$$

$$\mathbf{F}^i_{q,j} = \mathbb{E}_{q(x,h)}\Bigg[\sigma^{'}\left(V_j^i h^{i-1}\right) h^{i-1}{h^{i-1}}^{\mathrm{T}}\Bigg].$$

The fine-grained block diagonal structure of the Fisher Matrix, where $l_0, l_1, \dots$ are sizes of the layers:



The Fisher Matrices of both p and q are block-diagonal, largest block is of size $l_0 \times l_0$.

Natural Reweighted Wake-Sleep

Extension of the Reweighted Wake-Sleep [Bornschein and Bengio, 2014a]:

Natural Reweighted Wake-Sleep

Extension of the Reweighted Wake-Sleep [Bornschein and Bengio, 2014a]:

- Wake phase: update the $\nabla_G \mathcal{L}_p$ with the inverse Fisher Matrix of the Generation Network

$$\widetilde{\nabla}_G \mathcal{L}_p(\theta, x \sim \mathcal{D}) = \mathbf{F}_G^{-1}(\theta)\mathbb{E}_{q(h|x)}[\nabla_G \ln p(x,h)]$$

- Wake phase q update: $\nabla_R \mathcal{L}_q$ with the inverse Fisher Matrix of the Recognition Network and samples from the dataset:

$$\widetilde{\nabla}_R \mathcal{L}_q(\phi, x \sim \mathcal{D}) = \mathbf{F}_R^{-1}(\phi)\mathbb{E}_{q(h|x)}[\nabla_R \ln q(h|x)]$$

- Sleep phase: $\nabla_R \mathcal{L}_q$ with the inverse Fisher Matrix of the Recognition Network

$$\widetilde{\nabla}_R \mathcal{L}_q(\phi, (x,h)) = \mathbf{F}_R^{-1}(\phi)\mathbb{E}_{p(h|x)}[\nabla_R \ln q(h|x)]$$

Natural Reweighted Wake-Sleep

Extension of the Reweighted Wake-Sleep [Bornschein and Bengio, 2014a]:

- Wake phase: update the $\nabla_G \mathcal{L}_p$ with the inverse Fisher Matrix of the Generation Network

$$\widetilde{\nabla}_G \mathcal{L}_p(\theta, x \sim \mathcal{D}) = \mathbf{F}_G^{-1}(\theta)\mathbb{E}_{q(h|x)}[\nabla_G \ln p(x,h)]$$

- Wake phase q update: $\nabla_R \mathcal{L}_q$ with the inverse Fisher Matrix of the Recognition Network and samples from the dataset:

$$\widetilde{\nabla}_R \mathcal{L}_q(\phi, x \sim \mathcal{D}) = \mathbf{F}_R^{-1}(\phi)\mathbb{E}_{q(h|x)}[\nabla_R \ln q(h|x)]$$

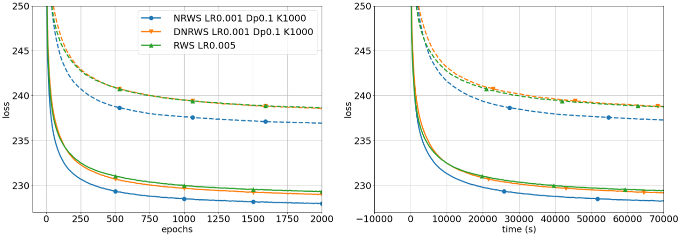
- Sleep phase: $\nabla_R \mathcal{L}_q$ with the inverse Fisher Matrix of the Recognition Network

$$\widetilde{\nabla}_R \mathcal{L}_q(\phi, (x,h)) = \mathbf{F}_R^{-1}(\phi)\mathbb{E}_{p(h|x)}[\nabla_R \ln q(h|x)]$$

Computation

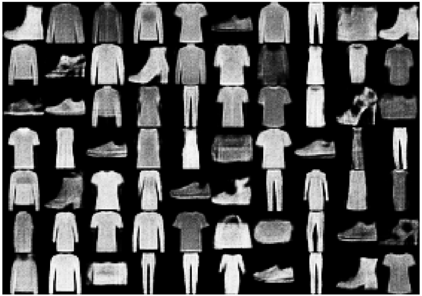
- The estimation of the FIMs is done by Monte-Carlo sampling
- The inverse of FIM is stabilized by Tikhonov Regularization and sped up by using the Sherman-Morrison formula.

Results on the FashionMNIST Dataset

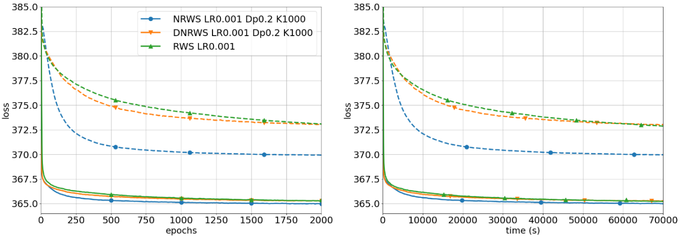


(left) Loss in epochs (left), in seconds (right).

Generated Images - FashionMNIST

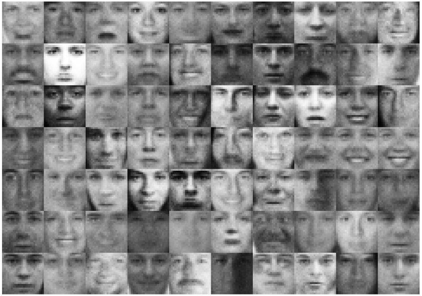


Results on the TFD Dataset



(left) Loss in epochs (left), in seconds (right).

Generated Images - TFD Dataset



Take Home Message

The geometry of statistical models is rich, Riemannian Fisher-Rao geometry is just one component

Dually-flat geometries play a key role in the computation of the natural gradient, since they allow to obtain simplified formula

Natural gradient finds multiple applications in optimization: the key aspect is the choice of models and parametrization which allow computation in large dimensions

Thanks for Your Attention!

For any question feel free to get in contact `malago@tins.ro`