Information-Geometric Optimization Algorithms: A Unifying Picture via Invariance Principles

Yann Ollivier, Ludovic Arnold, Anne Auger, Nikolaus Hansen

Journal of Machine Learning Research 18 (2017) 1-65 Submitted 11/14; Revised 10/16; Published 4/17

Information-Geometric Optimization Algorithms: A Unifying Picture via Invariance Principles

Yann Ollivier YANN.OLLIVIER@LRI.FR
CNRS & LRI (UMR 8623), Université Paris-Saclay
91405 Orsay, France

Ludovic Arnold ARNOLD@LRI.FR
Univ. Paris-Sud, LRI
91405 Orsay, France

Anne Auger ANNE.AUGER@INRIA.FR
Nikolaus Hansen NIKOLAUS.HANSEN@INRIA.FR
Inria & CMAP, Ecole polytechnique
91128 Palaiseau, France

Editor: Una-May O'Reilly

Abstract

We present a canonical way to turn any smooth parametric family of probability distributions on an arbitrary search space X into a continuous-time black-box optimization method on X, the *information-geometric optimization* (IGO) method. Invariance as a major design principle keeps the number of arbitrary choices to a minimum. The resulting *IGO flow* is the flow of an ordinary differential equation conducting the natural gradient ascent of an adaptive, time-dependent transformation of the objective function. It makes no particular assumptions on the objective function to be optimized.

The IGO method produces explicit IGO algorithms through time discretization. It naturally recovers versions of known algorithms and offers a systematic way to derive new ones. In continuous search spaces, IGO algorithms take a form related to natural evolution strategies (NES). The cross-entropy method is recovered in a particular case with a large time step, and can be extended into a smoothed, parametrization-independent maximum likelihood update (IGO-ML). When applied to the family of Gaussian distributions on $\mathbb{R}^d$, the IGO framework recovers a version of the well-known CMA-ES algorithm and of xNES. For the family of Bernoulli distributions on $\{0,1\}^d$, we recover the seminal PBIL algorithm and cGA. For the distributions of restricted Boltzmann machines, we naturally obtain a novel algorithm for discrete optimization on $\{0,1\}^d$. All these algorithms are natural instances of, and unified under, the single information-geometric optimization framework.

The IGO method achieves, thanks to its intrinsic formulation, maximal invariance properties: invariance under reparametrization of the search space X, under a change of parameters of the probability distribution, and under increasing transformation of the function to be optimized. The latter is achieved through an adaptive, quantile-based formulation of the objective.

Theoretical considerations strongly suggest that IGO algorithms are essentially characterized by a minimal change of the distribution over time. Therefore they have minimal loss in diversity through the course of optimization, provided the initial diversity is high. First experiments using restricted Boltzmann machines confirm this insight. As a simple conse-

quence, IGO seems to provide, from information theory, an elegant way to simultaneously explore several valleys of a fitness landscape in a single run.

Keywords: black-box optimization, stochastic optimization, randomized optimization, natural gradient, invariance, evolution strategy, information-geometric optimization

1. Introduction

Optimization problems are at the core of many disciplines. Given an objective function $f : X \to \mathbb{R}$, to be optimized on some space X, the goal of black-box optimization is to find solutions $x \in X$ with small (in the case of minimization) value $f(x)$, using the least number of calls to the function f. In a *black-box* scenario, knowledge about the function f is restricted to the handling of a device (e.g., a simulation code) that delivers the value $f(x)$ for any input $x \in X$. The search space X may be finite, discrete infinite, or continuous. However, optimization algorithms are often designed for a specific type of search space, exploiting its specific structure.

One major design principle in general and in optimization in particular is related to *invariance*. Invariance extends performance observed on a single function to an entire associated invariance class, that is, it generalizes from a single problem to a class of problems. Thus it hopefully provides better robustness w.r.t. changes in the presentation of a problem. For continuous search spaces, invariance under translation of the coordinate system is standard in optimization. Invariance under general affine-linear changes of the coordinates has been—we believe—one of the keys to the success of the Nelder-Mead *Downhill Simplex method* (Nelder and Mead, 1965), of quasi-Newton methods (Deuflhard, 2011) and of the *covariance matrix adaptation evolution strategy*, CMA-ES (Hansen and Ostermeier, 2001; Hansen and Auger, 2014). While these relate to transformations in the search space, another important invariance concerns the application of monotonically increasing transformations to f, so that it is indifferent whether the function f, f^3 or $f \times |f|^{-2/3}$ is minimized. This way some non-convex or non-smooth functions can be as "easily" optimised as convex ones. Invariance under f-transformation is not uncommon, e.g., for evolution strategies (Schwefel, 1995) or pattern search methods (Hooke and Jeeves, 1961; Torczon, 1997; Nelder and Mead, 1965); however it has not always been recognized as an attractive feature.

Many stochastic optimization methods have been proposed to tackle black-box optimization. The underlying (often hidden) principle of these stochastic methods is to iteratively update a probability distribution P_θ defined on X, parametrized by a set of parameters θ. At a given iteration, the distribution P_θ represents, loosely speaking, the current belief about where solutions with the smallest values of the function f may lie. Over time, P_θ is expected to concentrate around the minima of f. The update of the distribution involves querying the function with points sampled from the current probability distribution P_θ. Although implicit in the presentation of many stochastic optimization algorithms, this is the explicit setting for the wide family of *estimation of distribution algorithms* (EDA) (Larranaga and Lozano, 2002; Baluja and Caruana, 1995; Pelikan et al., 2002). Updates of the probability distribution often rely on heuristics (nevertheless in Toussaint 2004 the possible interest of information geometry to exploit the structure of probability distributions for designing better grounded heuristics is pointed out). In addition, in the EDA setting we can distinguish two theoretically founded approaches to update P_θ.

First, the *cross-entropy* method consists in taking θ minimizing the Kullback–Leibler divergence between P_θ and the indicator of the best points according to f (de Boer et al., 2005).

Second, one can transfer the objective function f to the space of parameters θ by taking the average of f under P_θ, seen as a function of θ. This average is a new function from an Euclidian space to $\mathbb{R}$ and is minimal when P_θ is concentrated on minima of f. Consequently, θ can be updated by following a gradient descent of this function with respect to θ. This has been done in various situations such as $X = \{0,1\}^d$ and the family of Bernoulli measures (Berny, 2000a) or of Boltzmann machines (Berny, 2002), or on $X = \mathbb{R}^d$ for the family of Gaussian distributions (Berny, 2000b; Gallagher and Frean, 2005) or the exponential family also using second order information (Zhou and Hu, 2014; E. Zhou, 2016).

However, taking the ordinary gradient with respect to θ depends on the precise way a parameter θ is chosen to represent the distribution P_θ, and does not take advantage of the Riemannian metric structure of families of probability distributions. In the context of machine learning, Amari noted the shortcomings of the ordinary gradient for families of probability distributions (Amari, 1998) and proposed instead to use the natural gradient with respect to the Fisher metric (Rao, 1945; Jeffreys, 1946; Amari and Nagaoka, 2000). In the context of optimization, this natural gradient has been used for exponential families on $X = \{0,1\}^d$ (Malagò et al., 2008, 2011) and for the family of Gaussian distributions on $X = \mathbb{R}^d$ with so-called natural evolution strategies (NES, Wierstra et al. 2008; Sun et al. 2009; Glasmachers et al. 2010; Wierstra et al. 2014).

However, none of the previous attempts using gradient updates captures the invariance under increasing transformations of the objective function, which is instead, in some cases, enforced *a posteriori* with heuristic arguments.

Building on these ideas, this paper overcomes the invariance problem of previous attempts and provides a consistent, unified picture of optimization on arbitrary search spaces via invariance principles. More specifically, we consider an arbitrary search space X, either discrete or continuous, and a black-box optimization problem on X, that is, a function $f : X \to \mathbb{R}$ to be minimized. We assume that a family of probability distributions P_θ on X depending on a continuous multicomponent parameter $\theta \in \Theta$ has been chosen. A classical example is to take $X = \mathbb{R}^d$ and to consider the family of all Gaussian distributions P_θ on $\mathbb{R}^d$, with $\theta = (m, C)$ the mean and covariance matrix. Another simple example is $X = \{0,1\}^d$ equipped with the family of Bernoulli measures: $\theta = (\theta_i)_{1 \leqslant i \leqslant d}$ and $P_\theta(x) = \prod \theta_i^{x_i}(1-\theta_i)^{1-x_i}$ for $x = (x_i) \in X$.

From this setting, *information-geometric optimization* (IGO) can be defined in a natural way. At each (continuous) time t, we maintain a value θ^t of the parameter of the distribution. The function f to be optimized is transferred to the parameter space Θ by means of a suitable time-dependent transformation based on the P_{θ^t}*-levels* of f (Definition 3). The *IGO flow*, introduced in Definition 4, follows the natural gradient of the expected value of this function of θ^t in the parameter space Θ, where the natural gradient derives from the Fisher information metric. The IGO flow is thus the flow of an ordinary differential equation in space Θ. This continuous-time gradient flow is turned into a family of explicit *IGO algorithms* by taking an Euler time discretization of the differential equation and approximating the distribution P_{θ^t} by using samples. From the start, the IGO flow is invariant under strictly increasing transformations of f (Proposition 17); we also prove that the sampling procedure

is consistent (Theorem 6). IGO algorithms share their final algebraic form with the *natural evolution strategies* (NES) introduced in the Gaussian setting (Wierstra et al., 2008; Sun et al., 2009; Glasmachers et al., 2010; Wierstra et al., 2014); the latter are thus recovered in the IGO framework as an Euler approximation to a well-defined flow, without heuristic arguments.

The IGO method also has an equivalent description as an *infinitesimal maximum likelihood update* (Theorem 10); this reveals a new property of the natural gradient and does not require a smooth parametrization by θ anymore. This also establishes a specific link (Theorem 12) between IGO, the natural gradient, and the *cross-entropy method* (de Boer et al., 2005).

When we instantiate IGO using the family of Gaussian distributions on $\mathbb{R}^d$, we naturally obtain versions of the well-known *covariance matrix adaptation evolution strategy* (CMA-ES, Hansen and Ostermeier 2001; Hansen and Kern 2004; Jastrebski and Arnold 2006) and of *natural evolution strategies*. With Bernoulli measures on the discrete cube $\{0,1\}^d$, we recover (Proposition 14) the well-known *population-based incremental learning* (PBIL, Baluja and Caruana 1995; Baluja 1994) and the *compact genetic algorithm* (cGA, Harik et al. 1999); this derivation of PBIL or cGA as a natural gradient ascent appears to be new, and emphasizes the common ground between continuous and discrete optimization.

From the IGO framework, it is (theoretically) immediate to build new optimization algorithms using more complex families of distributions than Gaussian or Bernoulli. As an illustration, distributions associated with restricted Boltzmann machines (RBMs) provide a new but natural algorithm for discrete optimization on $\{0,1\}^d$ which is able to handle dependencies between the bits (see also Berny 2002). The probability distributions associated with RBMs are multimodal; combined with the specific information-theoretic properties of IGO that guarantee minimal change in diversity over time, this allows IGO to reach multiple optima at once in a natural way, at least in a simple experimental setup (Appendix B).

The IGO framework is built to achieve maximal *invariance properties*. First, the IGO flow is invariant under reparametrization of the family of distributions P_θ, that is, it only depends on P_θ and not on the way we write the parameter θ (Proposition 18). This invariance under θ-reparametrization is the main idea behind *information geometry* (Amari and Nagaoka, 2000). For instance, for Gaussian measures it should not matter whether we use the covariance matrix or its inverse or a Cholesky factor as the parameter. This limits the influence of encoding choices on the behavior of the algorithm. Second, the IGO flow is invariant under a change of coordinates in the search space X, provided that this change of coordinates globally preserves the family of distributions P_θ (Proposition 19). For instance, for Gaussian distributions on $\mathbb{R}^d$, this includes all affine changes of coordinates. This means that the algorithm, apart from initialization, does not depend on the precise way the data is presented. Last, the IGO flow and IGO algorithms are invariant under applying a strictly increasing function to f (Proposition 17). Contrary to previous formulations using natural gradients (Wierstra et al., 2008; Glasmachers et al., 2010; Akimoto et al., 2010), this invariance is achieved from the start. Such invariance properties mean that we deal with *intrinsic* properties of the objects themselves, and not with the way we encode them as collections of numbers in $\mathbb{R}^d$. It also means, most importantly, that we make a minimal number of arbitrary choices.

In Section 2, we define the IGO flow and the IGO algorithm. We begin with standard facts about the definition and basic properties of the natural gradient, and its connection with Kullback–Leibler divergence and diversity. We then proceed to the detailed description of the algorithm.

In Section 3, we state some first mathematical properties of IGO. These include monotone improvement of the objective function, invariance properties, and the form of IGO for exponential families of probability distributions.

In Section 4 we explain the theoretical relationships between IGO, maximum likelihood estimates and the cross-entropy method. In particular, IGO is uniquely characterized by a weighted log-likelihood maximization property (Theorem 10).

In Section 5, we apply the IGO framework to explicit families of probability distributions. Several well-known optimization algorithms are recovered this way. These include PBIL (Sec. 5.1) using Bernoulli distributions, and versions of CMA-ES and other evolutionary algorithms such as EMNA and xNES (Sec. 5.2) using Gaussian distributions.

Appendix A discusses further aspects, perspectives and implementation matters of the IGO framework. In Appendix B, we illustrate how IGO can be used to design new optimization algorithms, by deriving the IGO algorithm associated with restricted Boltzmann machines. We illustrate experimentally, on a simple bimodal example, the specific influence of the Fisher information matrix on the optimization trajectories, and in particular on the diversity of the optima obtained. Appendix C details a number of further mathematical properties of IGO (such as its invariance properties or the case of noisy objective functions). Appendix D contains the previously omitted proofs of the mathematical statements.

2. Algorithm Description

We now present the outline of the algorithm. Each step is described in more detail in the sections below.

The IGO flow can be seen as an *estimation of distribution algorithm*: at each time t, we maintain a probability distribution P_{θ^t} on the search space X, where $\theta^t \in \Theta$. The value of θ^t will evolve so that, over time, P_{θ^t} gives more weight to points x with better values of the function $f(x)$ to optimize.

A straightforward way to proceed is to transfer f from x-space to θ-space: define a function $F(\theta)$ as the P_θ-average of f and then do a gradient descent for $F(\theta)$ in space Θ (Berny, 2000a, 2002, 2000b; Gallagher and Frean, 2005). This way, θ will converge to a point such that P_θ yields a good average value of f. We depart from this approach in two ways:

- At each time, we replace f with an adaptive transformation of f representing how good or bad observed values of f are *relative to other observations*. This provides invariance under all monotone transformations of f.

- Instead of the vanilla gradient for θ, we use the so-called *natural gradient* given by the Fisher information matrix. This reflects the intrinsic geometry of the space of probability distributions, as introduced by Rao and Jeffreys (Rao, 1945; Jeffreys, 1946) and later elaborated upon by Amari and others (Amari and Nagaoka, 2000). This provides invariance under reparametrization of θ and, importantly, minimizes the change of diversity of P_θ.

The algorithm is constructed in two steps: we first give an "ideal" version, namely, a version in which time t is continuous so that the evolution of θ^t is given by an ordinary differential equation in Θ. Second, the actual algorithm is a time discretization using a finite time step and Monte Carlo sampling instead of exact P_θ-averages.

2.1 The Natural Gradient on Parameter Space

We recall suitable definitions of the vanilla and the natural gradient and motivate using the natural gradient in the context of optimization.

2.1.1 About Gradients and the Shortest Path Uphill

Let g be a smooth function from $\mathbb{R}^d$ to $\mathbb{R}$, to be maximized. We first recall the interpretation of gradient ascent as "the shortest path uphill".

Let $y \in \mathbb{R}^d$. Define the vector z by

$$z = \lim_{\varepsilon \to 0} \operatorname*{arg\,max}_{z,\, \|z\| \leqslant 1} g(y + \varepsilon z) \ . \tag{1}$$

Then one can check that z is the normalized gradient of g at y: $z_i = \frac{\partial g/\partial y_i}{\|\partial g/\partial y\|}$. (This holds only at points y where the gradient of g does not vanish.)

This shows that, for small δt, the well-known gradient ascent of g given by

$$y_i^{t+\delta t} = y_i^t + \delta t \, \frac{\partial g}{\partial y_i}$$

realizes the largest increase of the value of g, for a given step size $\|y^{t+\delta t} - y^t\|$.

The relation (1) depends on the choice of a norm $\|\cdot\|$ (the gradient of g is given by $\partial g/\partial y_i$ only in an orthonormal basis). If we use, instead of the standard metric $\|y - y'\| = \sqrt{\sum (y_i - y_i')^2}$ on $\mathbb{R}^d$, a metric $\|y - y'\|_A = \sqrt{\sum A_{ij}(y_i - y_i')(y_j - y_j')}$ defined by a positive definite matrix A_{ij}, then the gradient of g with respect to this metric is given by $\sum_j A_{ij}^{-1} \frac{\partial g}{\partial y_i}$. This follows from the textbook definition of gradients by $g(y+\varepsilon z) = g(y) + \varepsilon \langle \nabla g, z \rangle_A + O(\varepsilon^2)$ with $\langle \cdot, \cdot \rangle_A$ the scalar product associated with the matrix A_{ij} (Schwartz, 1992).

It is possible to write the analogue of (1) using the A-norm. We then find that the gradient ascent associated with metric A is given by

$$y^{t+\delta t} = y^t + \delta t \, A^{-1} \, \frac{\partial g}{\partial y}$$

for small δt and maximizes the increment of g for a given A-distance $\|y^{t+\delta t} - y^t\|_A$—it realizes the steepest A-ascent. Maybe this viewpoint clarifies the relationship between gradient and metric: this steepest ascent property can actually be used as a definition of gradients.

In our setting we want to use a gradient ascent in the parameter space Θ of our distributions P_θ. The "vanilla" gradient $\frac{\partial}{\partial \theta_i}$ is associated with the metric $\|\theta - \theta'\| = \sqrt{\sum (\theta_i - \theta_i')^2}$ and clearly depends on the choice of parametrization θ. Thus this metric, and the direction pointed by this gradient, are not intrinsic, in the sense that they do not depend only on the *distribution* P_θ. A metric depending on θ only through the distributions P_θ can be defined as follows.

2.1.2 Fisher Information and the Natural Gradient on Parameter Space

Let $\theta, \theta' \in \Theta$ be two values of the distribution parameter. A widely used way to define a "distance" between two generic distributions[1] P_θ and $P_{\theta'}$ is the *Kullback–Leibler divergence* from information theory, defined (Kullback, 1997) as

$$\mathrm{KL}(P_{\theta'} \,\|\, P_\theta) = \int_x \ln \frac{P_{\theta'}(\mathrm{d}x)}{P_\theta(\mathrm{d}x)}\, P_{\theta'}(\mathrm{d}x) \ .$$

When $\theta' = \theta + \delta\theta$ is close to θ, under mild smoothness assumptions we can expand the Kullback–Leibler divergence at second order in $\delta\theta$. This expansion defines the Fisher information matrix I at θ (Kullback, 1997):

$$\mathrm{KL}(P_{\theta+\delta\theta} \,\|\, P_\theta) = \frac{1}{2} \sum I_{ij}(\theta)\, \delta\theta_i \delta\theta_j + O(\delta\theta^3) \ . \tag{2}$$

An equivalent definition of the Fisher information matrix is by the usual formulas (Cover and Thomas, 2006)

$$I_{ij}(\theta) = \int_x \frac{\partial \ln P_\theta(x)}{\partial \theta_i} \frac{\partial \ln P_\theta(x)}{\partial \theta_j}\, P_\theta(\mathrm{d}x) = -\int_x \frac{\partial^2 \ln P_\theta(x)}{\partial \theta_i\, \partial \theta_j}\, P_\theta(\mathrm{d}x) \ .$$

The Fisher information matrix defines a (Riemannian) metric on Θ: the distance, in this metric, between two very close values of θ is given by the square root of twice the Kullback–Leibler divergence. Since the Kullback–Leibler divergence depends only on P_θ and not on the parametrization of θ, this metric is intrinsic.

If $g : \Theta \to \mathbb{R}$ is a smooth function on the parameter space, its *natural gradient* (Amari, 1998) at θ is defined in accordance with the Fisher metric as

$$(\widetilde{\nabla}_\theta\, g)_i = \sum_j I^{-1}_{ij}(\theta)\, \frac{\partial g(\theta)}{\partial \theta_j}$$

or more synthetically

$$\widetilde{\nabla}_\theta\, g = I^{-1}\, \frac{\partial g}{\partial \theta} \ .$$

From now on, we will use $\widetilde{\nabla}_\theta$ to denote the natural gradient and $\frac{\partial}{\partial \theta}$ to denote the vanilla gradient.

By construction, the natural gradient descent is intrinsic: it does not depend on the chosen parametrization θ of P_θ, so that it makes sense to speak of the natural gradient ascent of a function $g(P_\theta)$. The Fisher metric is essentially the only way to obtain this property (Amari and Nagaoka, 2000, Section 2.4).

Given that the Fisher metric comes from the Kullback–Leibler divergence, the "shortest path uphill" property of gradients mentioned above translates as follows (see also Amari 1998, Theorem 1):

1. Throughout the text we do not distinguish a probability distribution P, seen as a measure, and its density with respect to some unspecified reference measure $\mathrm{d}x$, and so will write indifferently $P(\mathrm{d}x)$ or $P(x)\mathrm{d}x$. The measure-theoretic viewpoint allows for a unified treatment of the discrete and continuous case.

Proposition 1 *The natural gradient ascent points in the direction $\delta\theta$ achieving the largest change of the objective function, for a given distance between P_θ and $P_{\theta+\delta\theta}$ in Kullback–Leibler divergence. More precisely, let g be a smooth function on the parameter space Θ. Let $\theta \in \Theta$ be a point where $\widetilde{\nabla} g(\theta)$ does not vanish. Then, if*

$$\delta\theta = \frac{\widetilde{\nabla} g(\theta)}{\|\widetilde{\nabla} g(\theta)\|}$$

is the direction of the natural gradient of g (with $\|\cdot\|$ the Fisher norm), we have

$$\delta\theta = \lim_{\varepsilon \to 0} \frac{1}{\varepsilon} \underset{\substack{\delta\theta' \text{ such that} \\ \mathrm{KL}\left(P_{\theta+\delta\theta'} \,\|\, P_\theta\right) \leqslant \varepsilon^2/2}}{\arg\max} g(\theta + \delta\theta').$$

Here we have implicitly assumed that the parameter space Θ is such that no two points $\theta \in \Theta$ define the same probability distribution, and the mapping $P_\theta \mapsto \theta$ is smooth.

2.1.3 Why Use the Fisher Metric Gradient for Optimization? Relationship to Diversity

The first reason for using the natural gradient is its reparametrization invariance, which makes it the only gradient available in a general abstract setting (Amari and Nagaoka, 2000). Practically, this invariance also limits the influence of encoding choices on the behavior of the algorithm (Appendix C.1). The Fisher matrix can be also seen as an *adaptive learning rate* for different components of the parameter vector θ_i: components i with a high impact on P_θ will be updated more cautiously.

Another advantage comes from the relationship with Kullback–Leibler distance in view of the "shortest path uphill" (see also Amari 1998). To minimize the value of some function $g(\theta)$ defined on the parameter space Θ, the naive approach follows a gradient descent for g using the "vanilla" gradient

$$\theta_i^{t+\delta t} = \theta_i^t + \delta t \frac{\partial g}{\partial \theta_i}$$

and, as explained above, this maximizes the increment of g for a given increment $\|\theta^{t+\delta t} - \theta^t\|$. On the other hand, the Fisher gradient

$$\theta_i^{t+\delta t} = \theta_i^t + \delta t I^{-1} \frac{\partial g}{\partial \theta_i}$$

maximizes the increment of g for a given Kullback–Leibler distance $\mathrm{KL}(P_{\theta^{t+\delta t}} \,\|\, P_{\theta^t})$.

In particular, if we choose an initial value θ^0 such that P_{θ^0} covers the whole space X uniformly (or a wide portion, in case X is unbounded), the Kullback–Leibler divergence between P_{θ^t} and P_{θ^0} is the Shannon entropy of the uniform distribution minus the Shannon entropy of P_{θ^t}, and so this divergence measures the loss of diversity of P_{θ^t} with respect to the uniform distribution.

Proposition 2 *Let $g : \Theta \to \mathbb{R}$ be a regular function of θ and let θ^0 such that P_{θ^0} is the uniform distribution on a finite space X. Let $(\theta^t)_{t \geqslant 0}$ be the trajectory of the gradient ascent of g using the natural gradient. Then for small t we have*

$$\theta^t = \arg\max_\theta \{t \cdot g(\theta) + \mathrm{Ent}(P_\theta)\} + o(t) \tag{3}$$

where Ent *is the Shannon entropy.*

(We have stated the proposition over a finite space to have a well-defined uniform distribution. A short proof, together with the regularity conditions on g, is given in Appendix D.)

So following the natural gradient of a function g, starting at or close to the uniform distribution, amounts to optimizing the function g with *minimal loss of diversity, provided the initial diversity is large.* (This is valid, of course, only at the beginning; once one gets too far from uniform, a better interpretation is minimal *change* of diversity.) On the other hand, the vanilla gradient descent does not satisfy Proposition 2: it optimizes g with minimal change in the numerical values of the parameter θ, which is of little interest.

So arguably this method realizes the best trade-off between optimization and loss of diversity. (Though, as can be seen from the detailed algorithm description below, maximization of diversity occurs only greedily at each step, and so there is no guarantee that after a given time, IGO will provide the highest possible diversity for a given objective function value.)

An experimental confirmation of the positive influence of the Fisher matrix on diversity is given in Appendix B below.

2.2 IGO: Information-Geometric Optimization

We now introduce a quantile-based rewriting of the objective function. From applying the natural gradient on the rewritten objective we derive *information-geometric optimization.*

2.2.1 Quantile Rewriting of f

Our original problem is to minimize a function $f : X \to \mathbb{R}$. A simple way to turn f into a function on Θ is to use the expected value $-\mathbb{E}_{P_\theta} f$ (Berny, 2000a; Wierstra et al., 2008), but expected values can be unduly influenced by extreme values and using them can be rather unstable (Whitley, 1989); moreover $-\mathbb{E}_{P_\theta} f$ is not invariant under increasing transformation of f (this invariance implies we can only *compare* f-values, not sum them up).

Instead, we take an adaptive, quantile-based approach by first replacing the function f with a monotone rewriting $W^f_{\theta^t}$, depending on the current parameter value θ^t, and then following the gradient of $\mathbb{E}_{P_\theta} W^f_{\theta^t}$, seen as a function of θ. A due choice of $W^f_{\theta^t}$ allows to control the range of the resulting values and achieves the desired invariance. Because the rewriting $W^f_{\theta^t}$ depends on θ^t, it might be viewed as an *adaptive* f-transformation.

The monotone rewriting entails that if $f(x)$ is "small" then $W^f_\theta(x) \in \mathbb{R}$ is "large" and vice versa. The quantitative meaning of "small" or "large" depends on $\theta \in \Theta$. To obtain the value of $W^f_\theta(x)$ we compare $f(x)$ to the quantiles of f under the current distribution, as measured by the P_θ-level fraction in which the value of $f(x)$ lies.

Definition 3 *The lower and upper P_θ-f-levels of $x \in X$ are defined as*

$$\begin{aligned} q^<_\theta(x) &= \Pr_{x' \sim P_\theta}(f(x') < f(x)) \\ q^\leqslant_\theta(x) &= \Pr_{x' \sim P_\theta}(f(x') \leqslant f(x)) \ . \end{aligned} \tag{4}$$

Let $w : [0; 1] \to \mathbb{R}$ be a non-increasing function, the selection scheme*.*

The transform $W_\theta^f(x)$ of an objective function $f : X \to \mathbb{R}$ is defined as a function of the P_θ-f-level of x as

$$W_\theta^f(x) = \begin{cases} w(q_\theta^{\leqslant}(x)) & \text{if } q_\theta^{\leqslant}(x) = q_\theta^{<}(x), \\ \frac{1}{q_\theta^{\leqslant}(x) - q_\theta^{<}(x)} \int_{q=q_\theta^{<}(x)}^{q=q_\theta^{\leqslant}(x)} w(q)\,\mathrm{d}q & \text{otherwise.} \end{cases} \tag{5}$$

The level functions $q : X \to [0, 1]$ reflect the probability to sample a better value than $f(x)$. They are monotone in f (if $f(x_1) \leqslant f(x_2)$ then $q_\theta^{<}(x_1) \leqslant q_\theta^{<}(x_2)$, and likewise for $q^{\leqslant}$) and invariant under strictly increasing transformations of f.

A typical choice for w is $w(q) = \mathbb{1}_{q \leqslant q_0}$ for some fixed value q_0, the *selection quantile*. In what follows, we suppose that a selection scheme (weighting scheme) w has been chosen once and for all.

As desired, the definition of W_θ^f is invariant under a strictly increasing transformation of f. For instance, the P_θ-median of f gets remapped to $w(\frac{1}{2})$.

Note that $\mathbb{E}_{x\sim P_\theta} W_\theta^f(x)$ is always equal to $\int_0^1 w$, independently of f and θ: indeed, by definition, the P_θ inverse-quantile of a random point under P_θ is uniformly distributed in $[0; 1]$. In the following, our objective will be to maximize the expected value of $W_{\theta^t}^f$ over θ, that is, to maximize

$$\mathbb{E}_{P_\theta} W_{\theta^t}^f = \int W_{\theta^t}^f(x)\, P_\theta(\mathrm{d}x) \tag{6}$$

over θ, where θ^t is fixed at a given step but will adapt over time.

Importantly, $W_\theta^f(x)$ can be estimated in practice: indeed, the P_θ-f-levels $\Pr_{x'\sim P_\theta}(f(x') < f(x))$ can be estimated by taking samples of P_θ and ordering the samples according to the value of f (see below). The estimate remains invariant under strictly increasing f-transformations.

2.2.2 The IGO Gradient Flow

At the most abstract level, IGO is a continuous-time gradient flow in the parameter space Θ, which we define now. In practice, discrete time steps (a.k.a. iterations) are used, and P_θ-integrals are approximated through sampling, as described in the next section.

Let θ^t be the current value of the parameter at time t, and let $\delta t \ll 1$. We define $\theta^{t+\delta t}$ in such a way as to increase the P_θ-weight of points where f is small, while not going too far from P_{θ^t} in Kullback–Leibler divergence. We use the adaptive weights $W_{\theta^t}^f$ as a way to measure which points have large or small values. In accordance with (6), this suggests taking the gradient ascent

$$\theta^{t+\delta t} = \theta^t + \delta t\, \widetilde{\nabla}_\theta \int W_{\theta^t}^f(x)\, P_\theta(\mathrm{d}x) \tag{7}$$

where the natural gradient is suggested by Proposition 1.

Note again that we use $W_{\theta^t}^f$ and not W_θ^f in the integral. So the gradient $\widetilde{\nabla}_\theta$ does not act on the adaptive objective $W_{\theta^t}^f$. If we used W_θ^f instead, we would face a paradox: right after a move, previously good points do not seem so good any more since the distribution has improved. More precisely, $\int W_\theta^f(x)\, P_\theta(\mathrm{d}x)$ is constant and always equal to the average weight $\int_0^1 w$, and so the gradient would always vanish.

Using the log-likelihood trick $\widetilde{\nabla} P_\theta = P_\theta \, \widetilde{\nabla} \ln P_\theta$ (assuming P_θ is smooth), we get an equivalent expression of the update above as an integral under the current distribution P_{θ^t}; this is important for practical implementation. This leads to the following definition.

Definition 4 (IGO flow) *The IGO flow is the set of continuous-time trajectories in space Θ, defined by the ordinary differential equation*

$$\frac{\mathrm{d}\theta^t}{\mathrm{d}t} = \widetilde{\nabla}_\theta \int W^f_{\theta^t}(x) \, P_\theta(x) \mathrm{d}x \tag{8}$$

$$= \int W^f_{\theta^t}(x) \, \frac{\widetilde{\nabla}_\theta P_\theta(x)}{P_{\theta^t}(x)} P_{\theta^t}(x) \mathrm{d}x$$

$$= \int W^f_{\theta^t}(x) \, \widetilde{\nabla}_\theta \ln P_\theta(x) \, P_{\theta^t}(\mathrm{d}x) \tag{9}$$

$$= I^{-1}(\theta^t) \int W^f_{\theta^t}(x) \, \frac{\partial \ln P_\theta(x)}{\partial \theta} \, P_{\theta^t}(\mathrm{d}x) \ . \tag{10}$$

where the gradients are taken at point $\theta = \theta^t$, and I is the Fisher information matrix.

Natural evolution strategies (NES, Wierstra et al. 2008; Glasmachers et al. 2010; Sun et al. 2009; Wierstra et al. 2014) feature a related gradient *descent* with $f(x)$ instead of $W^f_{\theta^t}(x)$. The associated flow would read

$$\frac{\mathrm{d}\theta^t}{\mathrm{d}t} = -\widetilde{\nabla}_\theta \int f(x) \, P_\theta(\mathrm{d}x) \ , \tag{11}$$

where the gradient is taken at θ^t (in the sequel when not explicitly stated, gradients in θ are taken at $\theta = \theta^t$). However, in the end NESs always implement algorithms using sample quantiles (via "nonlinear *fitness shaping*"), as if derived from the gradient ascent of $W^f_{\theta^t}(x)$.

The update (9) is a weighted average of "intrinsic moves" increasing the log-likelihood of some points. We can slightly rearrange the update as

$$\frac{\mathrm{d}\theta^t}{\mathrm{d}t} = \int \overbrace{W^f_{\theta^t}(x)}^{\text{preference weight}} \underbrace{\widetilde{\nabla}_\theta \ln P_\theta(x)}_{\text{intrinsic move to reinforce } x} \overbrace{P_{\theta^t}(\mathrm{d}x)}^{\text{current sample distribution}} \tag{12}$$

$$= \widetilde{\nabla}_\theta \int \underbrace{W^f_{\theta^t}(x) \ln P_\theta(x)}_{\text{weighted log-likelihood}} P_{\theta^t}(\mathrm{d}x) \ , \tag{13}$$

which provides an interpretation for the IGO gradient flow as a gradient ascent optimization of the weighted log-likelihood of the "good points" of the current distribution. In the sense of Theorem 10 below, IGO is in fact the "best" way to increase this log-likelihood.

For exponential families of probability distributions, we will see later that the IGO flow rewrites as a nice derivative-free expression (21).

2.2.3 IGO Algorithms: Time Discretization and Sampling

The above is a mathematically well-defined continuous-time flow in parameter space. Its practical implementation involves three approximations depending on two parameters N and δt:

- the integral under P_{θ^t} is approximated using N samples taken from P_{θ^t};
- the value $W_{\theta^t}^f$ is approximated for each sample taken from P_{θ^t};
- the time derivative $\frac{\mathrm{d}\theta^t}{\mathrm{d}t}$ is approximated by a δt time increment: instead of the continuous-time IGO flow (8) we use its Euler approximation scheme $\theta^{t+\delta t} \approx \theta^t + \delta t \frac{\mathrm{d}\theta^t}{\mathrm{d}t}$, so that the time t of the flow is discretized with a step size δt, which thus becomes the learning rate of the algorithm. (See Corollary 21 for an interpretation of δt as a number of bits of information introduced in the distribution P_{θ^t} at each step.)

We also assume that the Fisher information matrix $I(\theta)$ and $\frac{\partial \ln P_\theta(x)}{\partial \theta}$ can be computed (see discussion below if $I(\theta)$ is unknown).

At each step, we draw N samples $x_1, \dots, x_N$ under P_{θ^t}. To approximate $W_{\theta^t}^f$, we rank the samples according to the value of f. Define $\mathrm{rk}(x_i) = \#\{j \mid f(x_j) < f(x_i)\}$ and let the estimated weight of sample x_i be

$$\widehat{w}_i = \frac{1}{N} w\left(\frac{\mathrm{rk}(x_i) + 1/2}{N}\right), \tag{14}$$

using the non-increasing selection scheme function w introduced in Definition 3 above. (This is assuming there are no ties in our sample; in case several sample points have the same value of f, we define $\widehat{w}_i$ by averaging the above over all possible rankings of the ties[2].)

Then we can approximate the IGO flow as follows.

Definition 5 (IGO algorithms) *The* IGO algorithm *associated with parametrization θ, sample size N and step size δt is the following update rule for the parameter θ^t. At each step, N sample points $x_1, \dots, x_N$ are drawn according to the distribution P_{θ^t}. The parameter is updated according to*

$$\theta^{t+\delta t} = \theta^t + \delta t \sum_{i=1}^N \widehat{w}_i \left.\widetilde{\nabla}_\theta \ln P_\theta(x_i)\right|_{\theta=\theta^t} \tag{16}$$

$$= \theta^t + \delta t\, I^{-1}(\theta^t) \sum_{i=1}^N \widehat{w}_i \left.\frac{\partial \ln P_\theta(x_i)}{\partial \theta}\right|_{\theta=\theta^t} \tag{17}$$

where $\widehat{w}_i$ is the weight (14) *obtained from the ranked values of the objective function f.*

Equivalently one can fix the weights $w_i = \frac{1}{N} w\left(\frac{i-1/2}{N}\right)$ once and for all and rewrite the update as

$$\theta^{t+\delta t} = \theta^t + \delta t\, I^{-1}(\theta^t) \sum_{i=1}^N w_i \left.\frac{\partial \ln P_\theta(x_{i:N})}{\partial \theta}\right|_{\theta=\theta^t} \tag{18}$$

2. A mathematically neater but less intuitive version would be

$$\widehat{w}_i = \frac{1}{\mathrm{rk}^{\leqslant}(x_i) - \mathrm{rk}^{<}(x_i)} \int_{u=\mathrm{rk}^{<}(x_i)/N}^{u=\mathrm{rk}^{\leqslant}(x_i)/N} w(u)\mathrm{d}u \tag{15}$$

with $\mathrm{rk}^{<}(x_i) = \#\{j \mid f(x_j) < f(x_i)\}$ and $\mathrm{rk}^{\leqslant}(x_i) = \#\{j \mid f(x_j) \leqslant f(x_i)\}$.

where $x_{i:N}$ denotes the sampled point ranked i^{th} according to f, i.e. $f(x_{1:N}) < \ldots < f(x_{N:N})$ (assuming again there are no ties). Note that $\{x_{i:N}\} = \{x_i\}$ and $\{w_i\} = \{\widehat{w}_i\}$.

As will be discussed in Section 5, this update applied to multivariate normal distributions or Bernoulli measures allows us to neatly recover versions of some well-established algorithms, in particular CMA-ES and PBIL. Actually, in the Gaussian context updates of the form (17) have already been introduced (Glasmachers et al., 2010; Akimoto et al., 2010), though not formally derived from a continuous-time flow with inverse quantiles.

When $N \to \infty$, the IGO algorithm using samples approximates the continuous-time IGO gradient flow, see Theorem 6 below. Indeed, the IGO algorithm, with $N = \infty$, is simply the Euler approximation scheme for the ordinary differential equation defining the IGO flow (8). The latter result thus provides a sound mathematical basis for currently used rank-based updates.

2.2.4 IGO Flow versus IGO Algorithms

The IGO *flow* (8) is a well-defined continuous-time set of trajectories in the space of probability distributions P_θ, depending only on the objective function f and the chosen family of distributions. It does not depend on the chosen parametrization for θ (Proposition 18).

On the other hand, there are several IGO *algorithms* associated with this flow. Each IGO algorithm approximates the IGO flow in a slightly different way. An IGO algorithm depends on three further choices: a sample size N, a time discretization step size δt, and a choice of parametrization for θ in which to implement (17).

If δt is small enough, and N large enough, the influence of the parametrization θ disappears and all IGO algorithms are approximations of the "ideal" IGO flow trajectory. However, the larger δt, the poorer the approximation gets.

So for large δt, different IGO algorithms for the same IGO flow may exhibit different behaviors. We will see an instance of this phenomenon for Gaussian distributions: both CMA-ES and the maximum likelihood update (EMNA) can be seen as IGO algorithms, but the latter with $\delta t = 1$ is known to exhibit premature loss of diversity (Section 5.2).

Still, if δt is sufficiently small, two IGO algorithms for the same IGO flow will differ less from each other than from a non-IGO algorithm: at each step the difference is only $O(\delta t^2)$ (Appendix C.1). On the other hand, for instance, the difference between an IGO algorithm and the vanilla gradient ascent is, generally, not smaller than $O(\delta t)$ at each step, i.e., it can be roughly as big as the steps themselves.

2.2.5 Unknown Fisher Matrix

The algorithm presented so far assumes that the Fisher matrix $I(\theta)$ is known as a function of θ. This is the case for Gaussian or Bernoulli distributions. However, for restricted Boltzmann machines as considered below, no analytical form is known. Yet, provided the quantity $\frac{\partial}{\partial\theta} \ln P_\theta(x)$ can be computed or approximated, it is possible to approximate the integral

$$I_{ij}(\theta) = \int_x \frac{\partial \ln P_\theta(x)}{\partial \theta_i} \frac{\partial \ln P_\theta(x)}{\partial \theta_j} \, P_\theta(\mathrm{d}x)$$

using P_θ-Monte Carlo samples for x. These samples may or may not be the same as those used in the IGO update (17): in particular, it is possible to use as many Monte Carlo

samples as necessary to approximate I_{ij}, at no additional cost in terms of the number of calls to the black-box function f to optimize.

Note that each Monte Carlo sample x will contribute $\frac{\partial \ln P_\theta(x)}{\partial \theta_i} \frac{\partial \ln P_\theta(x)}{\partial \theta_j}$ to the Fisher matrix approximation. This is a rank-1 non-negative matrix[3]. So, for the approximated Fisher matrix to be invertible, the number of (distinct) samples x needs to be at least equal to, and ideally much larger than, the number of components of the parameter θ: $N_{\text{Fisher}} \geqslant \dim \Theta$.

For exponential families of distributions, the IGO update has a particular form (21) which simplifies this matter somewhat (Section 3.3). In Appendix B this is examplified using restricted Boltzmann machines.

3. First Properties of IGO

In this section we derive some basic properties of IGO and present the IGO flow for exponential families.

3.1 Consistency of Sampling

The first property to check is that when $N \to \infty$, the update rule using N samples converges to the IGO update rule. This is *not* a straightforward application of the law of large numbers, because the estimated weights $\widehat{w}_i$ depend (non-continuously) on the *whole* sample $x_1, \dots, x_N$, and not only on x_i.

Theorem 6 (Consistency) *When $N \to \infty$, the N-sample IGO update rule* (17)*:*

$$\theta^{t+\delta t} = \theta^t + \delta t \, I^{-1}(\theta^t) \sum_{i=1}^{N} \widehat{w}_i \left. \frac{\partial \ln P_\theta(x_i)}{\partial \theta} \right|_{\theta=\theta^t}$$

converges with probability 1 *to the update rule* (7)*:*

$$\theta^{t+\delta t} = \theta^t + \delta t \, \widetilde{\nabla}_\theta \int W_{\theta^t}^f(x) \, P_\theta(\mathrm{d}x) \quad .$$

The proof is given in Appendix D, under mild regularity assumptions. In particular we do not require that w is continuous. Unfortunately, the proof does not provide an explicit sample size above which the IGO algorithm would be guaranteed to stay close to the IGO flow with high probability; presumably such a size would be larger than the typical sample sizes used in practice.

This theorem may clarify previous claims (Wierstra et al., 2008; Akimoto et al., 2010) where rank-based updates similar to (7), such as in NES or CMA-ES, were derived from optimizing the expected value $-\mathbb{E}_{P_\theta} f$. The rank-based weights $\widehat{w}_i$ were then introduced *after* the derivation as a useful heuristic to improve stability. Theorem 6 shows that, for large N, CMA-ES and NES actually follow the gradient flow of the quantity $\mathbb{E}_{P_\theta} W_{\theta^t}^f$: the update can be rigorously derived from optimizing the expected value of the inverse-quantile-rewriting $W_{\theta^t}^f$.

3. The alternative, equivalent formula $I_{ij}(\theta) = -\int_x \frac{\partial^2 \ln P_\theta(x)}{\partial \theta_i \, \partial \theta_j} \, P_\theta(\mathrm{d}x)$ for the Fisher matrix will not necessarily yield non-negative matrices through Monte Carlo sampling.

3.2 Monotonicity: Quantile Improvement

Gradient descents come with a guarantee that the fitness value decreases over time. Here, since we work with probability distributions on X, we need to define a "fitness" of the distribution P_{θ^t}. An obvious choice is the expectation $\mathbb{E}_{P_{\theta^t}} f$, but it is not invariant under f-transformation and moreover may be sensitive to extreme values.

It turns out that the monotonicity properties of the IGO gradient flow depend on the choice of the selection scheme w. For instance, if $w(u) = \mathbb{1}_{u \leqslant 1/2}$, then the median of f under P_{θ^t} improves over time.

Proposition 7 (Quantile improvement) *Consider the IGO flow* (8)*, with the weight* $w(u) = \mathbb{1}_{u \leqslant q}$ *where* $0 < q < 1$ *is fixed. Then the value of the* q*-quantile of* f *improves over time: if* $t_1 \leqslant t_2$ *then* $Q^q_{P_{\theta^{t_2}}}(f) \leqslant Q^q_{P_{\theta^{t_1}}}(f)$*. Here the* q*-quantile value* $Q^q_P(f)$ *of* f *under a probability distribution* P *is defined as the largest number* m *such that* $\Pr_{x\sim P}(f(x) \geqslant m) \geqslant 1 - q$*.*

Assume moreover that the objective function f *has no plateau, i.e. for any* $v \in \mathbb{R}$ *and any* $\theta \in \Theta$ *we have* $\Pr_{x\sim P_\theta}(f(x) = v) = 0$*. Then for* $t_1 < t_2$ *either* $\theta^{t_1} = \theta^{t_2}$ *or* $Q^q_{P_{\theta^{t_2}}}(f) < Q^q_{P_{\theta^{t_1}}}(f)$*.*

The proof is given in Appendix D, together with the necessary regularity assumptions. Note that on a discrete search space, the objective function has only plateaus, and the q-quantile will evolve by successive jumps even as θ evolves continuously.

This property is proved here only for the IGO gradient flow (8) with $N = \infty$ and $\delta t \to 0$. For an IGO algorithm with finite N, the dynamics is random and one cannot expect monotonicity. Still, Theorem 6 ensures that, with high probability, trajectories of a large enough finite population stay close to the infinite-population limit trajectory.

In Akimoto and Ollivier (2013) this result was extended to finite time steps instead of infinitesimal δt, using the IGO-ML framework from Section 4 below.

3.3 The IGO Flow for Exponential Families

The expressions for the IGO update simplify somewhat if the family P_θ happens to be an exponential family of probability distributions (see also Malagò et al. 2008, 2011 for optimization using the natural gradient for exponential families). This covers, for instance, Gaussian or Bernoulli distributions.

Suppose that P_θ can be written as

$$P_\theta(\mathrm{d}x) = \frac{1}{Z(\theta)} \exp\left(\sum \theta_i T_i(x)\right) H(\mathrm{d}x)$$

where $T_1, \dots, T_k$ is a finite family of functions on X, $H(\mathrm{d}x)$ is an arbitrary reference measure on X, and $Z(\theta)$ is the normalization constant. It is well-known (Amari and Nagaoka, 2000, (2.33)) that

$$\frac{\partial \ln P_\theta(x)}{\partial \theta_i} = T_i(x) - \mathbb{E}_{P_\theta} T_i \tag{19}$$

so that (Amari and Nagaoka, 2000, (3.59))

$$I_{ij}(\theta) = \mathrm{Cov}_{P_\theta}(T_i, T_j) \ . \tag{20}$$

By plugging this into the definition of the IGO flow (10) we find:

Proposition 8 *Let P_θ be an exponential family parametrized by the natural parameters θ as above. Then the IGO flow is given by*

$$\frac{\mathrm{d}\theta}{\mathrm{d}t} = \mathrm{Cov}_{P_\theta}(T,T)^{-1}\,\mathrm{Cov}_{P_\theta}(T, W_\theta^f) \tag{21}$$

where $\mathrm{Cov}_{P_\theta}(T, W_\theta^f)$ denotes the vector $(\mathrm{Cov}_{P_\theta}(T_i, W_\theta^f))_i$, and $\mathrm{Cov}_{P_\theta}(T,T)$ the matrix $(\mathrm{Cov}_{P_\theta}(T_i,T_j))_{ij}$.

Note that the right-hand side does not involve derivatives w.r.t. θ any more. This result makes it easy to simulate the IGO flow using, e.g., a Gibbs sampler for P_θ: both covariances in (21) may be approximated by sampling, so that neither the Fisher matrix nor the gradient term need to be known in advance, and no derivatives are involved.

The values of the variables $\bar{T}_i = \mathbb{E}T_i$, namely the expected value of T_i under the current distribution, can often be used as an alternative parametrization for an exponential family (e.g. for a one-dimensional Gaussian, these are the mean μ and the second moment $\mu^2+\sigma^2$). The IGO flow (9) may be rewritten using these variables, using the relation $\widetilde{\nabla}_{\theta_i} = \frac{\partial}{\partial \bar{T}_i}$ for the natural gradient of exponential families (Proposition 29 in Appendix D). One finds:

Proposition 9 *With the same setting as in Proposition 8, the expectation variables $\bar{T}_i = \mathbb{E}_{P_\theta} T_i$ satisfy the following evolution equation under the IGO flow*

$$\frac{\mathrm{d}\bar{T}_i}{\mathrm{d}t} = \mathrm{Cov}(T_i, W_\theta^f) = \mathbb{E}(T_i\, W_\theta^f) - \bar{T}_i\, \mathbb{E}W_\theta^f \quad . \tag{22}$$

The proof is given in Appendix D, in the proof of Theorem 12. We shall further exploit this fact in Section 4.

3.3.1 Exponential Families with Latent Variables.

Similar formulas hold when the distribution $P_\theta(x)$ is the marginal of an exponential distribution $P_\theta(x,h)$ over a "hidden" or "latent" variable h, such as the restricted Boltzmann machines of Appendix B.

Namely, with $P_\theta(x) = \frac{1}{Z(\theta)} \sum_h \exp(\sum_i \theta_i T_i(x,h))\, H(\mathrm{d}x, \mathrm{d}h)$ we have

$$\frac{\partial \ln P_\theta(x)}{\partial \theta_i} = U_i(x) - \mathbb{E}_{P_\theta} U_i \tag{23}$$

where

$$U_i(x) = \mathbb{E}_{P_\theta}(T_i(x,h)|x)$$

is the expectation of $T_i(x,h)$ knowing x. Then the Fisher matrix is

$$I_{ij}(\theta) = \mathrm{Cov}_{P_\theta}(U_i, U_j) \tag{24}$$

and consequently, the IGO flow takes the form

$$\frac{\mathrm{d}\theta}{\mathrm{d}t} = \mathrm{Cov}_{P_\theta}(U,U)^{-1}\,\mathrm{Cov}_{P_\theta}(U, W_\theta^f) \quad . \tag{25}$$

3.4 Further Mathematical Properties of IGO

IGO enjoys a number of other mathematical properties that are expanded upon in Appendix C.

- By its very construction, the IGO flow is invariant under a number of transformations of the original problem. First, replacing the objective function f with a strictly increasing function of f does not change the IGO flow (Proposition 17 in Appendix C.1).

  Second, changing the parameterization θ used for the family P_θ (e.g., letting θ be a variance or a standard deviation) results in unchanged trajectories for the distributions P_θ under the IGO flow (Proposition 18 in Appendix C.1).

  Finally, the IGO flow is insensitive to transformations of the original problem by a change of variable in the search space X itself, provided this transformation can be reflected in the family of distributions P_θ (Proposition 19 in Appendix C.1); this covers, for instance, optimizing $f(Ax)$ instead of $f(x)$ in $\mathbb{R}^d$ using Gaussian distributions, where A is any invertible matrix.

  These latter two invariances are specifically due to the natural gradient and are not satisfied by a vanilla gradient descent. f-invariance and X-invariance are directly inherited from the IGO flow by IGO algorithms, but this is only approximately true of θ-invariance, as discussed in Appendix C.1.

- The speed of the IGO flow is bounded by the variance of the weight function w on $[0; 1]$ (Appendix C.2, Proposition 20). This implies that the parameter δt in the IGO flow is not a meaningless variable but is related to the maximal number of bits introduced in P_θ at each step (Appendix C.2, Corollary 21).

- IGO algorithms still make sense when the objective function f is noisy, that is, when each call to f returns a non-deterministic value. This can be accounted for without changing the framework: we prove in Proposition 22 (Appendix C.3) that IGO for noisy f is equivalent to IGO for a non-noisy $\tilde{f}$ defined on a larger space $X \times \Omega$ with distributions $\tilde{P}_\theta$ that are uniform over Ω. Consequently, theorems such as consistency of sampling immediately transfer to the noisy case.

- The IGO flow can be computed explicitly in the simple case of linear functions on $\mathbb{R}^d$ using Gaussian distributions, and its convergence can be proven for linear functions on $\{0, 1\}^d$ with Bernoulli distributions (Appendix C.4).

3.5 Implementation Remarks

We conclude this section by a few practical remarks when implementing IGO algorithms.

3.5.1 Influence of the Selection Scheme w

The selection scheme w directly affects the update rule (18).

A natural choice is $w(u) = \mathbb{1}_{u \leqslant q}$. This results, as shown in Proposition 7, in an improvement of the q-quantile over the course of optimization. Taking $q = 1/2$ springs to mind (giving positive weights to the better half of the samples); however, this is often

not selective enough, and both theory and experiments confirm that for the Gaussian case, efficient optimization requires $q < 1/2$ (see Section 5.2). According to Beyer (2001), on the sphere function $f(x) = \sum_i x_i^2$, the optimal q is about 0.27 if sample size N is not larger than the search space dimension d, and even smaller otherwise (Jebalia and Auger, 2010).

Second, replacing w with $w + c$ for some constant c clearly has no influence on the IGO continuous-time flow (7), since the gradient will cancel out the constant. However, this is not the case for the update rule (18) with a finite sample of size N.

Indeed, adding a constant c to w adds a quantity $c\frac{1}{N}\sum \widetilde{\nabla}_\theta \ln P_\theta(x_i)$ to the update. In expectation, this quantity vanishes because the P_θ-expected value of $\widetilde{\nabla}_\theta \ln P_\theta$ is 0 (because $\int (\widetilde{\nabla}_\theta \ln P_\theta)\, P_\theta = \int \widetilde{\nabla} P_\theta = \widetilde{\nabla} 1 = 0$). So adding a constant to w does not change the expected value of the update, but it may change, e.g., its variance. The empirical average of $\widetilde{\nabla}_\theta \ln P_\theta(x_i)$ in the sample will be $O(1/\sqrt{N})$. So translating the weights results in a $O(1/\sqrt{N})$ change in the update. See also Section 4 in Sun et al. (2009).

Thus, one may be tempted to introduce a well chosen value of c so as to reduce the variance of the update. However, determining an optimal value for c is difficult: the optimal value minimizing the variance actually depends on possible correlations between $\widetilde{\nabla}_\theta \ln P_\theta$ and the function f. The only general result is that one should shift w such that 0 lies within its range. Assuming independence, or dependence with enough symmetry, the optimal shift is when the weights average to 0.

3.5.2 Complexity

The complexity of the IGO algorithm depends much on the computational cost model. In optimization, it is fairly common to assume that the objective function f is very costly compared to any other calculations performed by the algorithm (Moré et al., 1981; Dolan and Moré, 2002). Then the cost of IGO in terms of number of f-calls is N per iteration, and the cost of using inverse quantiles and computing the natural gradient is negligible.

Setting the cost of f aside, the complexity of the IGO algorithm depends mainly on the computation of the (inverse) Fisher matrix. Assume an analytical expression for this matrix is known. Then, with $p = \dim \Theta$ the number of parameters, the cost of storage of the Fisher matrix is $O(p^2)$ per iteration, and its inversion typically costs $O(p^3)$ per iteration. However, depending on the situation and on possible algebraic simplifications, strategies exist to reduce this cost (e.g., Le Roux et al. 2007 in a learning context). For instance, for CMA-ES the cost is $O(Np)$ (Suttorp et al., 2009). More generally, parametrization by expectation parameters (see above), when available, may reduce the cost to $O(Np)$ as well.

If no analytical form of the Fisher matrix is known and Monte Carlo estimation is required, then complexity depends on the particular situation at hand and is related to the best sampling strategies available for a particular family of distributions. For Boltzmann machines, for instance, a host of such strategies are available (Ackley et al., 1985; Salakhutdinov and Murray, 2008; Salakhutdinov, 2009; Desjardins et al., 2010). Still, in such a situation, IGO may be competitive if the objective function f is costly.

3.5.3 Recycling Old Samples

To compute the ranks of samples in (14), it might be advisable to re-use samples from previous iterations, so that a smaller number of samples is necessary, see e.g. Sun et al. (2009).

For $N = 1$, this is indispensable. In order to preserve sampling consistency (Theorem 6) the old samples need to be reweighted using the ratio of their likelihood under the current versus old distribution, as in importance sampling.

In evolutionary computation, elitist selection (also called plus-selection) is a common approach where the all-time best samples are taken into account in each iteration. Elitist selection can be modelled in the IGO framework by using the current all-time best samples in addition to samples from P_θ. Specifically, in the $(\mu + \lambda)$-selection scheme, we set $N = \mu + \lambda$ and let $x_1, \ldots, x_\mu$ be the current all-time μ best points. Then we sample λ new points, $x_{\mu+1}, \ldots, x_N$, from the current distribution P_θ and apply (16) with $w(q) = (N/\mu)\mathbb{1}_{q \leq \mu/N}$.

3.5.4 Initialization

As with other distribution-based optimization algorithms, it is usually a good idea to initialize in such a way as to cover a wide portion of the search space, i.e. θ^0 should be chosen so that P_{θ^0} has large diversity. For IGO algorithms this is particularly effective, since, as explained above, the natural gradient provides minimal change of diversity (greedily at each step) for a given change in the objective function.

4. IGO, Maximum Likelihood, and the Cross-Entropy Method

In this section we generalize the IGO update for settings where the natural gradient may not exist. This generalization reveals a unique IGO algorithm for finite step-sizes δt and a natural link to the cross-entropy method.

4.1 IGO as a Smooth-time Maximum Likelihood Estimate

The IGO flow turns out to be the only way to maximize a *weighted* log-likelihood, where points of the current distribution are slightly reweighted according to f-preferences.

This relies on the following interpretation of the natural gradient as a weighted maximum likelihood update with infinitesimal learning rate. This result singles out, in yet another way, the *natural* gradient among all possible gradients. The proof is given in Appendix D.

Theorem 10 (Natural gradient as ML with infinitesimal weights) *Let $\varepsilon > 0$ and $\theta_0 \in \Theta$. Let $W(x)$ be a function of x and let θ be the solution of*

$$\theta = \arg\max_\theta \left\{ (1-\varepsilon) \underbrace{\int \ln P_\theta(x)\, P_{\theta_0}(\mathrm{d}x)}_{= \text{const} - \mathrm{KL}(P_{\theta_0} \| P_\theta),\ \textit{maximal for } \theta = \theta_0} + \varepsilon \int \ln P_\theta(x)\, \overbrace{W(x)}^{\textit{preference weight biasing } P_{\theta_0}} P_{\theta_0}(\mathrm{d}x) \right\} . \tag{26}$$

Then, when $\varepsilon \to 0$ we have

$$\theta = \theta_0 + \varepsilon \int \widetilde{\nabla}_\theta \ln P_\theta(x)\ W(x)\, P_{\theta_0}(\mathrm{d}x)\ + O(\varepsilon^2)\ . \tag{27}$$

Likewise for discrete samples: with $x_1, \ldots, x_N \in X$, let θ be the solution of

$$\theta = \arg\max_\theta \left\{ (1-\varepsilon) \int \ln P_\theta(x)\, P_{\theta_0}(\mathrm{d}x) + \varepsilon \sum_i W(x_i)\ \ln P_\theta(x_i) \right\} . \tag{28}$$

Then when $\varepsilon \to 0$ *we have*

$$\theta = \theta_0 + \varepsilon \sum_i W(x_i)\ \widetilde{\nabla}_\theta \ln P_\theta(x_i) + O(\varepsilon^2)\ . \tag{29}$$

So if $W(x) = W^f_{\theta_0}(x)$ is the weight of the points according to quantilized f-preferences, the weighted maximum log-likelihood necessarily is the IGO flow (9) using the natural gradient—or the IGO update (17) when using samples.

Thus the IGO flow is the unique flow that, continuously in time, slightly changes the distribution to maximize the log-likelihood of points with good values of f. (In addition, IGO continuously updates the weight $W^f_{\theta^t}(x)$ depending on f and on the current distribution, so that we keep optimizing.)

This theorem suggests a way to approximate the IGO flow by enforcing this interpretation for a given non-infinitesimal step size δt, as follows.

Definition 11 (IGO-ML algorithm) *The* IGO-ML algorithm *with step size* δt *updates the value of the parameter* θ^t *according to*

$$\theta^{t+\delta t} = \arg\max_\theta \left\{ (1 - \delta t \sum_i \widehat{w}_i) \int \ln P_\theta(x)\, P_{\theta^t}(\mathrm{d}x) \ +\ \delta t \sum_i \widehat{w}_i \ln P_\theta(x_i) \right\} \tag{30}$$

where $x_1, \dots, x_N$ *are sample points drawn according to the distribution* P_{θ^t}*, and* $\widehat{w}_i$ *is the weight* (14) *obtained from the ranked values of the objective function* f*.*

The IGO-ML algorithm is obviously independent of the parametrization θ: indeed it only depends on P_θ itself. Furthermore, the IGO-ML update (30) does not even require a smooth parametrization of the distribution anymore (though in this case, a small δt will likely result in stalling: $\theta^{t+\delta t} = \theta^t$ if the set of possible values for θ is discrete).

Like the cross-entropy method below, the IGO-ML algorithm can be applied only when the argmax can be computed.

It turns out that for exponential families, IGO-ML is just the IGO algorithm in a particular parametrization (see Theorem 12).

4.2 The Cross-Entropy Method

Taking $\delta t = 1$ in (30) above corresponds to a full maximum likelihood update; when using the truncation selection scheme w, this is the *cross-entropy method* (CEM). The cross-entropy method can be defined in an optimization setting as follows (de Boer et al., 2005). Like IGO, it depends on a family of probability distributions P_θ parametrized by $\theta \in \Theta$, and a number of samples N at each iteration. Let also $N_e = \lceil qN \rceil$ $(0 < q < 1)$ be a number of *elite* samples.

At each step, the cross-entropy method for optimization samples N points $x_1, \dots, x_N$ from the current distribution P_{θ^t}. Let $\widehat{w}_i$ be $1/N_e$ if x_i belongs to the N_e samples with the best value of the objective function f, and $\widehat{w}_i = 0$ otherwise. Then the *cross-entropy method* or *maximum likelihoood* update (CEM/ML) for optimization is (de Boer et al., 2005, Algorithm 3.1)

$$\theta^{t+1} = \arg\max_\theta \sum \widehat{w}_i \ln P_\theta(x_i) \tag{31}$$

(assuming the argmax is tractable). This corresponds to $\delta t = 1$ in (30).

A commonly used version of CEM with a smoother update depends on a step size parameter $0 < \alpha \leqslant 1$ and is given (de Boer et al., 2005) by

$$\theta^{t+1} = (1-\alpha)\theta^t + \alpha \arg\max_\theta \sum \widehat{w}_i \ln P_\theta(x_i). \tag{32}$$

The standard CEM/ML update is $\alpha = 1$. For $\alpha = 1$, the standard cross-entropy method is independent of the parametrization θ, whereas for $\alpha < 1$ this is not the case.

Note the difference between the IGO-ML algorithm (30) and the smoothed CEM update (32) with step size $\alpha = \delta t$: the smoothed CEM update performs a weighted average of the parameter value *after* taking the maximum likelihood estimate, whereas IGO-ML uses a weighted average of current and previous likelihoods, *then* takes a maximum likelihood estimate. In general, these two rules can greatly differ, as they do for Gaussian distributions (Section 5.2).

This swapping of averaging makes IGO-ML parametrization-independent whereas the smoothed CEM update is not.

Yet, for exponential families of probability distributions, there exists one particular parametrization θ in which the IGO algorithm and the smoothed CEM update coincide. We now proceed to this construction.

4.3 IGO for Expectation Parameters and Maximum Likelihood

The particular form of IGO for exponential families has an interesting consequence if the parametrization chosen for the exponential family is the set of *expectation parameters*. Let $P_\theta(x) = \frac{1}{Z(\theta)} \exp\left(\sum \theta_j T_j(x)\right) H(\mathrm{d}x)$ be an exponential family as above. The *expectation parameters* are $\bar{T}_j = \bar{T}_j(\theta) = \mathbb{E}_{P_\theta} T_j$, (denoted η_j in Amari and Nagaoka 2000, Eq. 3.56). The notation $\bar{T}$ will denote the collection $(\bar{T}_j)$. We shall use the notation $P_{\bar{T}}$ to denote the probability distribution P parametrized by the expectation parameters.

It is well-known that, in this parametrization, the maximum likelihood estimate for a sample of points $x_1, \ldots, x_N$ is just the empirical average of the expectation parameters over that sample:

$$\arg\max_{\bar{T}} \frac{1}{N} \sum_{i=1}^N \ln P_{\bar{T}}(x_i) = \frac{1}{N} \sum_{i=1}^N T(x_i) \ . \tag{33}$$

In the discussion above, one main difference between IGO and smoothed CEM was whether we took averages before or after taking the maximum log-likelihood estimate. For the expectation parameters $\bar{T}_i$, we see that these operations commute. (One can say that these expectation parameters "linearize maximum likelihood estimates".) After some work we get the following result.

Theorem 12 (IGO, CEM and maximum likelihood) *Let*

$$P_\theta(\mathrm{d}x) = \frac{1}{Z(\theta)} \exp\Big(\sum \theta_j T_j(x)\Big) \ H(\mathrm{d}x)$$

be an exponential family of probability distributions, where the T_j are functions of x and H is some reference measure. Let us parametrize this family by the expected values $\bar{T}_j = \mathbb{E} T_j$.

Let us assume the chosen weights $\widehat{w}_i$ sum to 1. For a sample $x_1, \dots, x_N$, let

$$T_j^* = \sum_i \widehat{w}_i \, T_j(x_i).$$

Then the IGO update (17) *in this parametrization reads*

$$\bar{T}_j^{t+\delta t} = (1 - \delta t) \, \bar{T}_j^t + \delta t \, T_j^*. \tag{34}$$

Moreover these three algorithms coincide:

- *The IGO-ML algorithm* (30)*.*
- *The IGO algorithm* (17) *written in the parametrization $\bar{T}_j$* (34)*.*
- *The smoothed CEM algorithm* (32) *written in the parametrization $\bar{T}_j$, with $\alpha = \delta t$.*

Corollary 13 *For exponential families, the standard CEM/ML update* (31) *coincides with the IGO algorithm in parametrization $\bar{T}_j$ with $\delta t = 1$.*

Beware that the expectation parameters $\bar{T}_j$ are not always the most obvious parameters (Amari and Nagaoka, 2000, Section 3.5). For example, for 1-dimensional Gaussian distributions, the expectation parameters are the mean μ and the second moment $\mu^2 + \sigma^2$, not the mean and variance. When expressed back in terms of mean and variance, the update (34) boils down to $\mu \leftarrow (1 - \delta t)\mu + \delta t \mu^*$ and $\sigma^2 \leftarrow (1 - \delta t)\sigma^2 + \delta t (\sigma^*)^2 + \delta t (1 - \delta t)(\mu^* - \mu)^2$, where μ^* and σ^* denote the mean and standard deviation of the samples x_i.

On the other hand, when using smoothed CEM with mean and variance as parameters, the new variance is $(1-\delta t)\sigma^2 + \delta t (\sigma^*)^2$, which can be significantly smaller for $\delta t \in (0, 1)$. This proves, in passing, that the smoothed CEM update in other parametrizations is generally *not* an IGO algorithm (because it can differ at first order in δt).

The case of Gaussian distributions is further exemplified in Section 5.2 below: in particular, smoothed CEM in the (μ, σ) parametrization almost invariably exhibits a reduction of variance, often leading to premature convergence.

For these reasons we think that the IGO-ML algorithm is the sensible way to define an interpolated ML estimate for $\delta t < 1$ in a parametrization-independent way (see however the analysis of a critical δt in Section 5.2). In Appendix A we further discuss IGO and CEM and sum up the differences and relative advantages.

Taking $\delta t = 1$ is a bold approximation choice: the "ideal" continuous-time IGO flow itself, after time 1, does not coincide with the maximum likelihood update of the best points in the sample. Since the maximum likelihood algorithm is known to converge prematurely in some instances (Section 5.2), using the parametrization by expectation parameters with large δt may not be desirable.

The considerable simplification of the IGO update in these coordinates reflects the duality of coordinates $\bar{T}_i$ and θ_i. More precisely, the natural gradient ascent w.r.t. the parameters $\bar{T}_i$ is given by the vanilla gradient w.r.t. the parameters θ_i:

$$\widetilde{\nabla}_{\bar{T}_i} = \frac{\partial}{\partial \theta_i}$$

(Proposition 29 in Appendix D).

5. CMA-ES, NES, EDAs and PBIL from the IGO Framework

In this section we investigate the IGO algorithms for Bernoulli measures and for multivariate normal distributions, and show the correspondence to well-known algorithms. Restricted Boltzmann machines are given as a third, novel example. In addition, we discuss the influence of the parametrization of the distributions.

5.1 PBIL and cGA as IGO Algorithms for Bernoulli Measures

Let us consider on $X = \{0,1\}^d$ a family of Bernoulli measures $P_\theta(x) = p_{\theta_1}(x_1) \times \ldots \times p_{\theta_d}(x_d)$ with $p_{\theta_i}(x_i) = \theta_i^{x_i}(1-\theta_i)^{1-x_i}$, with each $\theta_i \in [0;1]$. As this family is a product of probability measures $p_{\theta_i}(x_i)$, the different components of a random vector y following P_θ are independent and all off-diagonal terms of the Fisher information matrix are zero. Diagonal terms are given by $\frac{1}{\theta_i(1-\theta_i)}$. Therefore the inverse of the Fisher matrix is a diagonal matrix with diagonal entries equal to $\theta_i(1-\theta_i)$. In addition, the partial derivative of $\ln P_\theta(x)$ w.r.t. θ_i is computed in a straightforward manner resulting in

$$\frac{\partial \ln P_\theta(x)}{\partial \theta_i} = \frac{x_i}{\theta_i} - \frac{1-x_i}{1-\theta_i} \ .$$

Let $x_1, \ldots, x_N$ be N samples at step t with distribution P_{θ^t} and let $x_{1:N}, \ldots, x_{N:N}$ be the samples ranked according to f value. The natural gradient update (18) with Bernoulli measures is then

$$\theta_i^{t+\delta t} = \theta_i^t + \delta t \, \theta_i^t(1-\theta_i^t) \sum_{j=1}^N w_j \left(\frac{[x_{j:N}]_i}{\theta_i^t} - \frac{1-[x_{j:N}]_i}{1-\theta_i^t} \right) \tag{35}$$

where $w_j = w((j-1/2)/N)/N$ and $[y]_i$ denotes the i^{th} coordinate of $y \in X$. The previous equation simplifies to

$$\theta_i^{t+\delta t} = \theta_i^t + \delta t \sum_{j=1}^N w_j \left([x_{j:N}]_i - \theta_i^t \right) \ , \tag{36}$$

or, denoting $\bar{w}$ the sum of the weights $\sum_{j=1}^N w_j$,

$$\theta_i^{t+\delta t} = (1 - \bar{w}\delta t) \, \theta_i^t + \delta t \sum_{j=1}^N w_j \, [x_{j:N}]_i \ . \tag{37}$$

The algorithm so obtained coincides with the so-called *population-based incremental learning* algorithm (PBIL, Baluja and Caruana 1995) for $N =$ `NUMBER_SAMPLES` and the appropriate (usually non-negative) weights w_j, as well as with the *compact genetic algorithm* (cGA, Harik et al. 1999) for $N = 2$ and $w_1 = -w_2$. Different variants of PBIL correspond to different choices of the selection scheme w. In cGA, components for which both samples have the same value are unchanged, because $w_1 + w_2 = 0$. We have thus proved the following.

Proposition 14 *The IGO algorithm on $\{0,1\}^d$ using Bernoulli measures parametrized by θ as above, coincides with the* compact Genetic Algorithm *(cGA) when $N = 2$ and $w_1 = -w_2$.*

Moreover, it coincides with Population-Based Incremental Learning *(PBIL) with the following correspondence of parameters. The PBIL algorithm using the μ best solutions, see Baluja and Caruana (1995, Figure 4), is recovered*[4] *using $\delta t = \text{LR}$, $w_j = (1-\text{LR})^{j-1}$ for $j = 1, \dots, \mu$, and $w_j = 0$ for $j = \mu+1, \dots, N$.*

If the selection scheme of IGO is chosen as $w_1 = 1$, $w_j = 0$ for $j = 2, \dots, N$, IGO recovers the PBIL/EGA algorithm with update rule towards the best solution (Baluja, 1994, Figure 4), with $\delta t = \text{LR}$ (the learning rate of PBIL) and $\text{MUT\_PROBABILITY} = 0$ (no random mutation of θ).

Interestingly, the parameters θ_i are the expectation parameters described in Section 4: indeed, the expectation of x_i is θ_i. So the formulas above are particular cases of (34). Thus, by Theorem 12, PBIL is both a smoothed CEM in these parameters and an IGO-ML algorithm.

Let us now consider another, so-called "logit" representation, given by the logistic function $P(x_i = 1) = \frac{1}{1+\exp(-\tilde{\theta}_i)}$. This $\tilde{\theta}$ is the exponential parametrization of Section 3.3. We find that

$$\frac{\partial \ln P_{\tilde{\theta}}(x)}{\partial \tilde{\theta}_i} = (x_i - 1) + \frac{\exp(-\tilde{\theta}_i)}{1+\exp(-\tilde{\theta}_i)} = x_i - \mathbb{E} x_i \tag{38}$$

(cf. Eq. 19) and that the diagonal elements of the Fisher information matrix are given by $\exp(-\tilde{\theta}_i)/(1+\exp(-\tilde{\theta}_i))^2 = \operatorname{Var} x_i$ (as per Eq. 20). So the natural gradient update (18) with Bernoulli measures in parametrization $\tilde{\theta}$ reads

$$\tilde{\theta}_i^{t+\delta t} = \tilde{\theta}_i^t + \delta t (1+\exp(\tilde{\theta}_i^t)) \left(-\bar{w} + (1+\exp(-\tilde{\theta}_i^t)) \sum_{j=1}^{N} w_j [x_{j:N}]_i \right) \quad . \tag{39}$$

To better compare the update with the previous representation, note that $\theta_i = \frac{1}{1+\exp(-\tilde{\theta}_i)}$ and thus we can rewrite

$$\tilde{\theta}_i^{t+\delta t} = \tilde{\theta}_i^t + \frac{\delta t}{\theta_i^t (1-\theta_i^t)} \sum_{j=1}^{N} w_j \left([x_{j:N}]_i - \theta_i^t \right) \quad . \tag{40}$$

So the direction of the update is the same as before and is given by the proportion of bits set to 0 or 1 in the best samples, compared to its expected value under the current distribution. The magnitude of the update is different since the parameter $\tilde{\theta}$ ranges from $-\infty$ to $+\infty$ instead of from 0 to 1. We did not find this algorithm in the literature.

These updates also illustrate the influence of setting the sum of weights to 0 or not (Section 3.5). If, at some time, the first bit is equal to 1 both for a majority of good points and for a majority of bad points, then the original PBIL will increase the probability of setting the first bit to 1, which is counterintuitive. If the weights w_i are chosen to sum to 0 this noise effect disappears; otherwise, it disappears only on average.

4. Note that the pseudocode for the algorithm in Baluja and Caruana (1995, Figure 4) is slightly erroneous since it gives smaller weights to better individuals. The error can be fixed by updating the probability in reversed order, looping from `NUMBER_OF_VECTORS_TO_UPDATE_FROM` to 1. This was confirmed by S. Baluja in personal communication. We consider here the corrected version of the algorithm.

5.2 Multivariate Normal Distributions (Gaussians)

Evolution strategies (Rechenberg, 1973; Schwefel, 1995; Beyer and Schwefel, 2002) are black-box optimization algorithms for the continuous search domain, $X \subseteq \mathbb{R}^d$ (for simplicity we assume $X = \mathbb{R}^d$ in the following), which use multivariate normal distributions to sample new solutions. In the context of continuous black-box optimization, *Natural Evolution Strategies* (NES) introduced the idea of using a natural gradient update of the distribution parameters (Wierstra et al., 2008; Sun et al., 2009; Glasmachers et al., 2010; Wierstra et al., 2014). Surprisingly, the well-known *Covariance Matrix Adaption Evolution Strategy* (CMA-ES, Hansen and Ostermeier 1996, 2001; Hansen et al. 2003; Hansen and Kern 2004; Jastrebski and Arnold 2006) also turns out to conduct a natural gradient update of distribution parameters (Akimoto et al., 2010; Glasmachers et al., 2010).

Let $x \in \mathbb{R}^d$. As the most prominent example, we use mean vector $m = \mathbb{E}x$ and covariance matrix $C = \mathbb{E}(x-m)(x-m)^\mathsf{T} = \mathbb{E}(xx^\mathsf{T}) - mm^\mathsf{T}$ to parametrize a normal distribution via $\theta = (m, C)$. The IGO update in (17) or (18) in this parametrization can now be entirely formulated without the (inverse) Fisher matrix, similarly to (34) or (22). The complexity of the update is linear in the number of parameters (size of $\theta = (m, C)$, where $(d^2 - d)/2$ parameters are redundant).

Let us discuss known algorithms that implement updates of this kind.

5.2.1 CMA-ES.

The rank-μ-update CMA-ES implements the equations[5]

$$m^{t+1} = m^t + \eta_\mathrm{m} \sum_{i=1}^{N} \widehat{w}_i (x_i - m^t) \tag{41}$$

$$C^{t+1} = C^t + \eta_\mathrm{c} \sum_{i=1}^{N} \widehat{w}_i ((x_i - m^t)(x_i - m^t)^\mathsf{T} - C^t) \tag{42}$$

where $\widehat{w}_i$ are the weights based on ranked f-values, see (14) and (17).

Proposition 15 *The IGO update* (17) *for Gaussian distributions in the parametrization by mean and covariance matrix* (m, C)*, coincides with the CMA-ES update equations* (41) *and* (42) *with* $\eta_\mathrm{c} = \eta_\mathrm{m}$.

This result is essentially due to Akimoto et al. (2010) and Glasmachers et al. (2010), who showed that the CMA-ES update with $\eta_\mathrm{c} = \eta_\mathrm{m}$ is a natural gradient update[6].

However, in deviation from the IGO algorithm, the learning rates η_m and η_c are assigned different values if $N \ll \dim \Theta$ in CMA-ES[7]. Note that the Fisher information matrix is block-diagonal in m and C (Akimoto et al., 2010), so that application of the different

5. The CMA-ES implements these equations given the parameter setting $c_1 = 0$ and $c_\sigma = 0$ (or $d_\sigma = \infty$, see e.g. Hansen 2009) that disengages the effect of the rank-one update and of step size control and therefore of both so-called evolution paths.

6. In these articles the result has been derived for $\theta \leftarrow \theta + \eta \widetilde{\nabla}_\theta \mathbb{E}_{P_\theta} f$, see (11), leading to $f(x_i)$ in place of $\widehat{w}_i$. No assumptions on f have been used besides that it does not depend on θ. Consequently, by replacing f with $W^f_{\theta^t}$, where θ^t is fixed, the derivation holds equally well for $\theta \leftarrow \theta + \eta \widetilde{\nabla}_\theta \mathbb{E}_{P_\theta} W^f_{\theta^t}$.

7. Specifically, let $\sum |\widehat{w}_i| = 1$, then the settings are $\eta_\mathrm{m} = 1$ and $\eta_\mathrm{c} \approx 1/(d^2 \sum \widehat{w}_i^2)$ (Hansen, 2006b).

learning rates and of the inverse Fisher matrix commute. Moreover, CMA-ES uses a *path cumulation* method to adjust the step sizes, which is not covered by the IGO framework (see Appendix A).

5.2.2 Convenient Reparametrizations Over Time.

For practical purposes, at each step it is convenient to work in a representation of θ in which the diagonal Fisher matrix $I(\theta^t)$ has a simple form, e.g., diagonal with simple diagonal entries. It is generally not possible to obtain such a representation for all θ simultaneously. Still it is always possible to find a transformation achieving a diagonal Fisher matrix at a single parameter θ^t, in multiple ways (it amounts to choosing a basis of parameter space which is orthogonal in the Fisher metric). Such a representation is never unique and not intrinsic, yet it still provides a convenient way to write the algorithms.

For CMA-ES, one such representation can be found by sending the current covariant matrix C^t to the identity, e.g., by representing the mean and covariance matrix by $((C^t)^{-1/2}m, (C^t)^{-1/2}C(C^t)^{-1/2})$ instead of (m, C). Then the Fisher matrix $I(\theta^t)$ at (m^t, C^t) becomes diagonal. The next algorithm we discuss, xNES (Glasmachers et al., 2010), exploits this possibility in a logarithmic representation of the covariance matrix.

5.2.3 Natural Evolution Strategies.

Natural evolution strategies (NES, Wierstra et al. 2008; Sun et al. 2009) implement (41) as well, while using a Cholesky decomposition of C as the parametrization for the update of the variance parameters. The resulting update that replaces (42) is neither particularly elegant nor numerically efficient. The more recent xNES (Glasmachers et al., 2010) chooses an "exponential" parametrization that naturally depends on the current parameters. This leads to an elegant formulation where the additive update in exponential parametrization becomes a multiplicative update for C. With $C = AA^\mathsf{T}$, the matrix update reads

$$A \leftarrow A \times \exp\left(\frac{\eta_\mathrm{c}}{2}\sum_{i=1}^{N} \widehat{w}_i \times (z_i z_i^\mathsf{T} - \mathrm{I}_d)\right) \tag{43}$$

where $z_i = A^{-1}(x_i - m)$ and I_d is the identity matrix. From (43) the updated covariance matrix is $C \leftarrow A \times \exp\left(\eta_\mathrm{c} \sum_{i=1}^{N} \widehat{w}_i \times (z_i z_i^\mathsf{T} - \mathrm{I}_d)\right) \times A^\mathsf{T}$.

Compared to (42), the update has the advantage that also negative weights, $\widehat{w}_i < 0$, always lead to a feasible covariance matrix. By default, xNES sets $\eta_\mathrm{m} \neq \eta_\mathrm{c}$ in the same circumstances as in CMA-ES, but contrary to CMA-ES the past evolution path is not taken into account (Glasmachers et al., 2010).

When $\eta_\mathrm{c} = \eta_\mathrm{m}$, xNES is consistent with the IGO flow (8), and implements an IGO algorithm (17) slightly generalized in that it uses a θ^t-dependent parametrization, which represents the current covariance matrix by 0. Namely, we have:

Proposition 16 (exponential IGO update of Gaussians) *Let (m^t, C^t) be the current mean and covariance matrix. Let $C^t = AA^\mathsf{T}$. Let θ be the time-dependent parametrization of the space of Gaussian distributions, which parametrizes the Gaussian distribution (m, C) by*

$$\theta = (m, R), \qquad R = \ln(A^{-1}C(A^\mathsf{T})^{-1})$$

where ln *is the logarithm of positive matrices. (Note that the current value* C^t *of* C *is represented as* $R = 0$*.)*

Then the IGO update (17) *in the parametrization* θ *is as follows: the mean* m *is updated as in CMA-ES* (41)*, and the parameter* R *is updated as*

$$R \leftarrow \delta t \sum_{i=1}^{N} \widehat{w}_i \times (A^{-1}(x_i - m)(x_i - m)^{\mathsf{T}}(A^{\mathsf{T}})^{-1} - \mathrm{I}_d) \tag{44}$$

thus resulting in the same update as (43) *(with* $\eta_{\mathrm{c}} = \delta t$*) for the covariance matrix:* $C \leftarrow A \exp(R) A^{\mathsf{T}}$*.*

Proof Indeed, by basic differential geometry, if parametrization $\theta' = \varphi(\theta)$ is used, the IGO update for θ' is $D\varphi(\theta^t)$ applied to the IGO update for θ, where $D\varphi$ is the differential of φ. Here, given the update (42) for C, to find the update for R we have to compute the differential of the map $C \mapsto \ln(A^{-1}C(A^{\mathsf{T}})^{-1})$ taken at $C = AA^{\mathsf{T}}$: for any matrix M we have $\ln(A^{-1}(AA^{\mathsf{T}} + \varepsilon M)(A^{\mathsf{T}})^{-1}) = \varepsilon\, A^{-1}M(A^{\mathsf{T}})^{-1} + O(\varepsilon^2)$. So to find the update for the variable R we have to apply $A^{-1} \dots (A^{\mathsf{T}})^{-1}$ to the update (42) for C. ■

5.2.4 Cross-Entropy Method and EMNA

Estimation of distribution algorithms (EDA) and the *cross-entropy method* (CEM, Rubinstein 1999; Rubinstein and Kroese 2004) estimate a new distribution from a censored sample. Generally, the new parameter value can be written as

$$\begin{aligned} \theta_{\mathrm{maxLL}} &= \arg\max_{\theta} \sum_{i=1}^{N} \widehat{w}_i \ln P_\theta(x_i) \\ &\longrightarrow_{N\to\infty} \arg\max_{\theta} \mathbb{E}_{P_{\theta^t}} W^f_{\theta^t} \ln P_\theta \end{aligned} \tag{45}$$

by Theorem 6. Here, the weights $\widehat{w}_i$ are equal to $1/\mu$ for the μ best points (censored or elitist sample) and 0 otherwise. This θ_{maxLL} maximizes the weighted log-likelihood of $x_1, \dots, x_N$; equivalently, it minimizes the cross-entropy and the Kullback–Leibler divergence to the distribution of the best μ samples[8].

For Gaussian distributions, Equation (45) can be explicitly written in the form

$$m^{t+1} = m^* = \sum_{i=1}^{N} \widehat{w}_i x_i \tag{46}$$

$$C^{t+1} = C^* = \sum_{i=1}^{N} \widehat{w}_i (x_i - m^*)(x_i - m^*)^{\mathsf{T}} \tag{47}$$

the empirical mean and variance of the elite sample.

8. Let P_w denote the distribution of the weighted samples: $\Pr(x = x_i) = \widehat{w}_i$ and $\sum_i \widehat{w}_i = 1$. Then the cross-entropy between P_w and P_θ reads $\sum_i P_w(x_i) \ln 1/P_\theta(x_i)$ and the KL divergence reads $\mathrm{KL}(P_w \,\|\, P_\theta) = \sum_i P_w(x_i) \ln 1/P_\theta(x_i) - \sum_i P_w(x_i) \ln 1/P_w(x_i)$. Minimization of both terms in θ result in θ_{maxLL}.

Equations (46) and (47) also define the simplest continuous domain EDA, the *estimation of multivariate normal algorithm* ($\text{EMNA}_{\text{global}}$, Larranaga and Lozano 2002). Interestingly, (46) and (47) only differ from (41) and (42) (with $\eta_{\text{m}} = \eta_{\text{c}} = 1$) in that the new mean m^{t+1} is used instead of m^t in the covariance matrix update (Hansen, 2006b).

The smoothed CEM (32) in this parametrization thus writes

$$m^{t+\delta t} = (1 - \delta t)m^t + \delta t m^* \tag{48}$$

$$C^{t+\delta t} = (1 - \delta t)C^t + \delta t C^* \ . \tag{49}$$

Note that *this is not an IGO algorithm* (i.e., there is no parametrization of the family of Gaussian distributions in which the IGO algorithm coincides with update Eq. 49): indeed, all IGO algorithms coincide at first order in δt when $\delta t \to 0$ (because they recover the IGO flow), while this update for $C^{t+\delta t}$ does not coincide with (42) in this limit, due to the use of m^* instead of m^t. This does not contradict Theorem 12: smoothed CEM is an IGO algorithm only if smoothed CEM is written in the expectation parametrization, which (m, C) is not.

5.2.5 CMA-ES, Smoothed CEM, and IGO-ML

Let us compare IGO-ML (30), rank-μ CMA (41)–(42), and smoothed CEM (48)–(49) in the parametrization with mean and covariance matrix. These algorithms all update the distribution mean in the same way, while the update of the covariance matrix depends on the algorithm. With learning rate δt, these updates are computed to be

$$\begin{aligned} m^{t+1} &= (1 - \delta t)\, m^t + \delta t\, m^* \\ C^{t+1} &= (1 - \delta t)\, C^t + \delta t\, C^* + \delta t (1 - \delta t)^j\, (m^* - m^t)(m^* - m^t)^\mathsf{T} \ , \end{aligned} \tag{50}$$

for different values of j, where m^* and C^* are the mean and covariance matrix computed over the elite sample (with positive weights $\widehat{w}_i$ summing to one) as above. The rightmost term of (50) is reminiscent of the so-called rank-one update in CMA-ES (not included in Eq. 42).

For $j = 0$ we recover the rank-μ CMA-ES update (42), for $j = 1$ we recover IGO-ML, and for $j = \infty$ we recover smoothed CEM (the rightmost term is absent). The case $j = 2$ corresponds to an update that uses m^{t+1} instead of m^t in (42) (with $\eta_{\text{m}} = \eta_{\text{c}} = \delta t$). For $0 < \delta t < 1$, the larger j, the smaller C^{t+1}. For $\delta t = 1$, IGO-ML and smoothed CEM/EMNA realize θ_{maxLL} from (45)–(47).

For $\delta t \to 0$, the update is independent of j at first order in δt if $j < \infty$: this reflects compatibility with the IGO flow of CMA-ES and of IGO-ML, but not of smoothed CEM.

In the default (full) CMA-ES (as opposed to rank-μ CMA-ES), the coefficient preceeding $(m^* - m^t)(m^* - m^t)^\mathsf{T}$ in (50) reads approximately $3\delta t$, where the additional $2\delta t$ originate from the so-called rank-one update and are moreover modulated by a "cumulated path" up to a factor of about $\sqrt{d}$ (Hansen and Auger, 2014).

5.2.6 Critical δt

Let us assume that $\mu < N$ weights are set to $\widehat{w}_i = 1/\mu$ and the remaining weights to zero, so that the selection ratio is $q = \mu/N$.

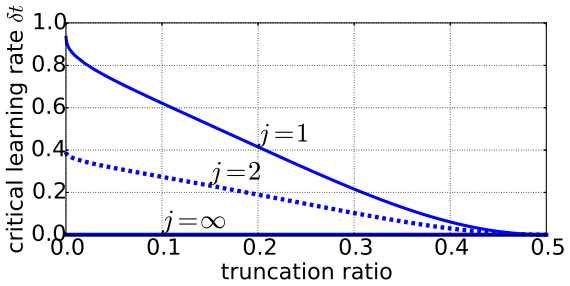

Figure 1: Critical δt versus selection truncation ratio q for three values of j in (50). With δt above the critical δt, the variance decreases systematically when optimizing a linear function, indicating failure of the algorithm. For CMA-ES/NES where $j = 0$, the critical δt for $q < 0.5$ is infinite.

Then there is a critical value of δt depending on this ratio q, such that above this critical δt the algorithms given by IGO-ML and smoothed CEM are prone to premature convergence. Indeed, let f be a linear function on $\mathbb{R}^d$, and consider the variance in the direction of the gradient of f. Assuming further $N \to \infty$ and $q \leq 1/2$, then the variance C^* of the elite sample is smaller than the current variance C^t, by a constant factor. Depending on the precise update for C^{t+1}, if δt is too large, the variance C^{t+1} is going to be smaller than C^t by a constant factor as well. This implies that the algorithm is going to stall, i.e., the variance will go to 0 before the optimum is reached. (On the other hand, the continuous-time IGO flow corresponding to $\delta t \to 0$ does not stall, see Appendix C.4.)

We now study the critical δt (in the limit $N \to \infty$) below which the algorithm does not stall (this is done similarly to the analysis in Appendix C.4 for linear functions). For IGO-ML, ($j = 1$ in Eq. 50, or equivalently for the smoothed CEM in the expectation parameters $(m, C + mm^\mathsf{T})$, see Section 4), the variance increases if and only if δt is smaller than the critical value $\delta t_{\text{crit}} = qb\sqrt{2\pi}e^{b^2/2}$ where b is the percentile function of q, i.e. b is such that $q = \int_b^\infty e^{-x^2/2}/\sqrt{2\pi}$. This value δt_{crit} is plotted as a solid line in Fig. 1. For $j = 2$, δt_{crit} is smaller, related to the above by $\delta t_{\text{crit}} \leftarrow \sqrt{1 + \delta t_{\text{crit}}} - 1$ and plotted as a dashed line in Fig. 1. For CEM ($j = \infty$), the critical δt is zero (reflecting the non-IGO behavior of CEM in this parametrization). For CMA-ES ($j = 0$), the critical δt is infinite for $q < 1/2$. When the selection ratio q is above $1/2$, for all algorithms the critical δt becomes zero.

5.2.7 Gaussian Distributions with Restricted Parametrization

When considering a restricted parametrization of multivariate normal distributions, IGO recovers other known algorithms. In particular for sep-CMA-ES (Ros and Hansen, 2008) and SNES (Schaul et al., 2011), the update has been restricted to the diagonal of the covariance matrix.

5.3 IGO for Restricted Boltzmann Machines, and Multimodal Optimization

As a third example after Bernoulli and Gaussian distributions, we have applied IGO to restricted Boltzmann machines (RBMs, Smolensky 1986; Ackley et al. 1985), which are

families of distributions on the search space $X = \{0,1\}^d$ extending Bernoulli distributions to allow for dependencies between the bits (see, e.g., Berny 2002 for Boltzmann machines used in an optimization setting). The details are given in Appendix B. We chose this example for two reasons.

First, it illustrates how to obtain a novel algorithm from a family of probability distributions in the IGO framework, in a case when there is no fully explicit expression for the Fisher information matrix.

Second, it illustrates the influence of the natural gradient on diversity and its relevance to multimodal optimization. The RBM probability distributions on $\{0,1\}^d$ are multimodal, contrary to Bernoulli distributions. Thus, in principle, IGO on such distributions could reach several optima simultaneously. This allows for a simple testing of the entropy-maximizing property of the IGO flow (Proposition 2). Accordingly, for a given improvement on the objective function, we would expect IGO to favor preserving the diversity of P_θ. Indeed on a simple objective function with two optima, we find that the IGO flow converges to a distribution putting weight on both optima, while a descent using the vanilla gradient always only concentrates around one optimum. This might be relevant for multimodal optimization, where simultaneous handling of multiple hypotheses is seen as useful for optimization (Sareni and Krähenbühl, 1998; Das et al., 2011), or in situations in which several valleys appear equally good at first but only one of them contains the true optimum.

We refer to Appendix B for a reminder about Boltzmann machines, for how IGO rolls out in this case, and for the detailed experimental setup for studying the influence of IGO in a multimodal situation.

6. Summary and Conclusion

We sum up:

- The information-geometric optimization (IGO) framework derives from invariance principles the uniquely defined IGO flow (Definition 4), and allows, via discretization in time and space, to build optimization algorithms from any family of distributions on any search space. In some instances (Gaussian distributions on $\mathbb{R}^d$ or Bernoulli distributions on $\{0,1\}^d$) it recovers versions of known algorithms (PBIL, CMA-ES, cGA, NES); in other instances (restricted Boltzmann machine distributions) it produces new, hopefully efficient optimization algorithms.

- The use of a quantile-based, time-dependent transform of the objective function, Equation (5), provides a rigorous derivation of rank-based update rules, frequently used in current optimization algorithms. Theorem 6 uniquely identifies the infinite-population limit of these update rules.

- The IGO flow is singled out by its equivalent description as an infinitesimal weighted maximum log-likelihood update (Theorem 10). In a particular parametrization and with a step size of 1, IGO recovers the cross-entropy method (Corollary 13). This reveals a connection between CEM and the natural gradient, and allows to define a new, fully parametrization-invariant smoothed maximum likelihood update, the IGO-ML.

- Theoretical arguments suggest that the IGO flow minimizes the change of diversity in the course of optimization. As diversity usually decreases in the course of optimization, IGO algorithms tend to exhibit minimal diversity loss for the observed improvement. In particular, starting with high diversity and using multimodal distributions may allow simultaneous exploration of multiple optima of the objective function. Preliminary experiments with restricted Boltzmann machines confirm this effect in a simple situation.

Thus, the IGO framework provides sound theoretical foundations to optimization algorithms based on probability distributions. In particular, this viewpoint helps to bridge the gap between continuous and discrete optimization.

The invariance properties, which reduce the number of arbitrary choices, together with the relationship between natural gradient and diversity, may contribute to a theoretical explanation of the good practical performance of those currently used algorithms, such as CMA-ES, which can be interpreted as instantiations of IGO.

We hope that invariance properties will acquire in computer science the importance they have in mathematics, where intrinsic thinking is the first step for abstract linear algebra or differential geometry, and in modern physics, where the notions of invariance w.r.t. the coordinate system and so-called gauge invariance play a central role.

Acknowledgments

The authors would like to thank Michèle Sebag for the acronym and for helpful comments. We also thank Youhei Akimoto for helpful feedback and inspiration. Y. O. would like to thank Cédric Villani and Bruno Sévennec for helpful discussions on the Fisher metric. A. A. and N. H. would like to acknowledge the Dagstuhl Seminar No 10361 on the Theory of Evolutionary Computation (`http://www.dagstuhl.de/10361`) for crucially inspiring this paper, their work on natural gradients and beyond. This work was partially supported by the ANR-2010-COSI-002 grant (SIMINOLE) of the French National Research Agency.

Appendix A. Further Discussion and Perspectives

This appendix touches upon further aspects and perspectives of the IGO framework and its implementations.

A.1 A Single Framework for Optimization on Arbitrary Spaces

A strength of the IGO viewpoint is to automatically provide a distinct and arguably in some sense optimal optimization algorithm from any family of probability distributions on any given space, discrete or continuous. IGO algorithms feature desired invariance properties and thereby make fewer arbitrary choices.

In particular, IGO describes several well-known optimization algorithms within a single framework. For Bernoulli distributions, to the best of our knowledge, PBIL or cGA have never been identified as a natural gradient ascent in the literature[9]. For Gaussian distributions, algorithms of the same form (17) had been developed previously (Hansen and Ostermeier, 2001; Wierstra et al., 2008) and their close relationship with a natural gradient ascent had been recognized (Akimoto et al., 2010; Glasmachers et al., 2010). These works, however, also strongly suggest that IGO algorithms may need to be complemented with further heuristics to achieve efficient optimization algorithms. In particular, learning rate settings and diversity control (step-size) deviate from the original framework. These deviations mostly stem from time discretization and the choice of a finite sample size, both necessary to derive an IGO *algorithm* from the IGO *flow*. Further work is needed to fully understand this from a theoretical viewpoint.

The wide applicability of natural gradient approaches seems not to be widely known in the optimization community (though see Malagò et al. 2008).

A.2 About Invariance

The role of invariance is two-fold. First, invariance breaks otherwise arbitrary choices in the algorithm design. Second, and more importantly, invariance implies generalization of behavior from single problem instances to entire problem classes (with the caveat to choose the initial state of the algorithm accordingly), thereby making the outcome of optimization more predictable.

Optimization problems faced in reality—and algorithms used in practice—are often far too complex to be amenable to a rigorous mathematical analysis. Consequently, the judgement of algorithms in practice is to a large extent based in empirical observations, either on artificial benchmarks or on the experience with real-world problems. Invariance, as a guarantee of generalization, has an immediate impact on the relevance of any such empirical observations and, in the same line of reasoning, can be interpreted as a notion of robustness.

A.3 About Quantiles

The IGO flow in (8) has, to the best of our knowledge, never been defined before. The introduction of the quantile-rewriting (5) of the objective function provides the first rigorous

9. Thanks to Jonathan Shapiro for giving an early argument confirming this property (personal communication).

derivation of quantile- or rank- or comparison-based optimization from a gradient ascent in θ-space.

NES and CMA-ES have been claimed to maximize $-\mathbb{E}_{P_\theta} f$ via natural gradient ascent (Wierstra et al., 2008; Akimoto et al., 2010). However, we have proved that the NES and CMA-ES updates actually converge to the IGO flow, not to the similar flow with the gradient of $\mathbb{E}_{P_\theta} f$ (Theorem 6). So we find that in reality these algorithms maximize $\mathbb{E}_{P_\theta} W^f_{\theta^t}$, where $W^f_{\theta^t}$ is a decreasing transformation of the f-quantiles under the current sample distribution.

Moreover, in practice, maximizing $-\mathbb{E}_{P_\theta} f$ tends to be a rather unstable procedure and has been discouraged, see for example Whitley (1989) and Sun et al. (2009).

A.4 About Choice of P_θ: Learning a Model of Good Points

The choice of the family of probability distributions P_θ plays a double role.

First, it is analogous to choosing the variation operators (namely *mutation* or *recombination*) as seen in evolutionary algorithms: indeed, P_θ encodes possible moves according to which new sample points are explored.

Second, optimization algorithms using distributions can be interpreted as learning a probabilistic model of where the points with good values lie in the search space. With this point of view, P_θ describes *richness of this model*: for instance, restricted Boltzmann machines with h hidden units can describe distributions with up to 2^h modes, whereas the Bernoulli distribution is unimodal. This influences, for instance, the ability to explore several valleys and optimize multimodal functions in a single run.

More generally, the IGO framework makes it tempting to use more complex models of where good points lie, inspired, e.g., from machine learning, and adapt them for optimization. The restricted Boltzmann machines of Appendix B are a first step in this direction. The initial idea behind these machines is that each hidden unit controls a block of coordinates of the search space (a block of features), so that the optimization algorithm hopefully builds a good model of which features must be activated or de-activated together to obtain good values of f. This is somewhat reminiscent of a crossover operator: if observation of good points shows that a block of features go together, this information is stored in the RBM structure and this block may be later activated as a whole, thus effectively transferring blocks of features from one good solution to another. Inspired by models of deep learning (Bengio et al., 2012), one might be tempted to stack such models on top of each other, so that optimization would operate on a more and more abstract representation of the problem. IGO and the natural gradient might help in exploiting the added expressivity that comes with richer models: in our simple experiment, the vanilla gradient ignores the additional expressivity of RBMs with respect to Bernoulli distributions (Appendix B). The downside of a richer model is that either the sample size N must be increased or the learning rate δt must be decreased to obtain a stable algorithm (a situation analogous to overfitting in machine learning: richer models will be quicker to concentrate too much around a few observed good samples).

A.5 Natural Gradient and Parametrization Invariance

Central to IGO is the use of the natural gradient, which follows from θ-invariance (parametrization invariance) and makes sense on any search space, discrete or continuous.

While the IGO flow is exactly θ-invariant, for any practical implementation of an IGO algorithm, a parametrization choice has to be made. Different parametrizations lead to different algorithms and larger values of δt are likely to result in more differing algorithms. Still, since all IGO algorithms approximate the IGO flow, two parametrizations in combination with IGO will differ less than the same two parametrizations in combination with another algorithm (such as the vanilla gradient or the smoothed CEM method)—at least if the learning rate δt is not too large. The chosen parametrization becomes more relevant as the step size δt increases.

On the other hand, natural evolution strategies have not emphasized θ-invariance: the chosen parametrization (Cholesky, exponential) has been considered as a defining feature. We believe the term "natural evolution strategy" should rather be used independently of the chosen parameterization, thereby referring to the usage of the natural gradient as the main principle for the update of distribution parameters.

A.6 IGO, Maximum Likelihood and Cross-Entropy

The cross-entropy method (CEM, de Boer et al. 2005) can be used to produce optimization algorithms given a family of probability distributions on an arbitrary space, by performing a jump to a maximum likelihood estimate of the parameters.

We have seen (Corollary 13) that the standard CEM is an IGO algorithm *in a particular parametrization*, with a learning rate δt equal to 1. However, it is well-known, both theoretically and experimentally (Branke et al., 2007; Hansen, 2006b; Wagner et al., 2004), that standard CEM loses diversity too fast in many situations. The usual solution (de Boer et al., 2005) is to reduce the learning rate (smoothed CEM, Equation 32), but this breaks the reparametrization invariance of non-smoothed CEM.

On the other hand, the IGO flow can be seen as a *maximum likelihood update with infinitesimal learning rate* (Theorem 10). This interpretation allows to define a particular IGO algorithm, the IGO-ML (Definition 11): it performs a maximum likelihood update with an arbitrary learning rate, and keeps the reparametrization invariance. It coincides with CEM when the learning rate is set to 1, but it differs from smoothed CEM by the exchange of the order of argmax and averaging (compare Equations 30 and 32), and coincides with the IGO flow for small learning rates. We argue that this new, fully invariant algorithm is the conceptually better way to reduce the learning rate and achieve smoothing in CEM.

Standard CEM with rate 1 can lose diversity, yet is a particular case of an IGO algorithm: this illustrates the fact that reasonable values of the learning rate δt depend on the parametrization. We have studied this phenomenon in detail for various Gaussian IGO algorithms (Section 5.2).

Why would a smaller learning rate perform better than a large one in an optimization setting? It might seem more efficient to jump directly to the maximum likelihood estimate of currently known good points, instead of performing an iterated gradient ascent towards this maximum. Due to having only a limited number of samples, however, optimization faces a "moving target", contrary to a learning setting in which the example distribution is often stationary. Currently known good points heavily depend on the current distribution and are likely not to indicate the *position* at which the optimum lies, but, rather, the broad

direction in which the optimum is to be found. After each update, the next elite sample points are going to be located somewhere new.

A.7 Diversity and Multiple Optima

The IGO framework emphasizes the relation of natural gradient and diversity: we argued that IGO provides minimal diversity change for a given objective function improvement. In particular, provided the initial diversity is large, diversity is kept at a maximum after the upate. This theoretical relationship can be observed experimentally for restricted Boltzmann machines (Appendix B).

On the other hand, using the vanilla gradient does not lead to a balanced distribution between the two optima in our experiments. Using the vanilla gradient introduces hidden arbitrary choices between those points (more exactly between moves in Θ-space). This results in unnecessary and unwelcome loss of diversity, and might also be detrimental at later stages in the optimization. This may reflect the fact that the Euclidean metric on the space of parameters, implicitly used in the vanilla gradient, becomes less and less meaningful for gradient descent on complex distributions.

IGO and the natural gradient are certainly relevant to the well-known problem of exploration-exploitation balance: as we have seen, arguably the natural gradient realizes the largest improvement in the objective with the least possible change of diversity in the distribution.

More generally, like other distribution-based optimization algorithms, IGO tries to learn a model of where the good points are. This is typical of machine learning, one of the contexts for which the natural gradient was studied. The conceptual relationship of IGO and IGO-like optimization algorithms with machine learning is still to be explored and exploited.

We now present some ideas which we believe would be worth exploring.

A.8 Adaptive Learning Rate

Comparing consecutive updates to evaluate a learning rate or step size is an effective measure. For example, in back-propagation, the update sign has been used to adapt the learning rate of each single weight in an artificial neural network (Silva and Almeida, 1990). In CMA-ES, a step size is adapted depending on whether recent steps tended to move in a consistent direction or to backtrack. This is measured by considering the changes of the mean m of the Gaussian distribution.

For a probability distribution P_θ on an arbitrary search space, in general no notion of mean may be defined. However, it is still possible to define "backtracking" in the evolution of θ as follows.

Consider two successive updates $\delta\theta^t = \theta^t - \theta^{t-\delta t}$ and $\delta\theta^{t+\delta t} = \theta^{t+\delta t} - \theta^t$. Their scalar product in the Fisher metric $I(\theta^t)$ is

$$\langle \delta\theta^t, \delta\theta^{t+\delta t} \rangle = \sum_{ij} I_{ij}(\theta^t)\, \delta\theta_i^t\, \delta\theta_j^{t+\delta t} \quad .$$

Dividing by the associated norms will yield the cosine $\cos\alpha$ of the angle between $\delta\theta^t$ and $\delta\theta^{t+\delta t}$.

If this cosine is positive, the learning rate δt may be increased. If the cosine is negative, the learning rate probably needs to be decreased. Various schemes for the change of δt can be devised; for instance, inspired by step size updates commonly used in evolution strategies, one can multiply δt by $\exp(\beta(\cos\alpha))$ or $\exp(\beta\,\mathrm{sign}(\cos\alpha))$, where $\beta \approx \min(N/\dim\Theta, 1/2)$. As this cosine can be quite noisy, cumulation over several time steps might be advisable.

As before, this scheme is constructed to be robust w.r.t. reparametrization of θ, thanks to the use of the Fisher metric. However, for large learning rates δt, in practice the parametrization might well become relevant.

A.9 Geodesic Parametrization

While the IGO flow is fully invariant under θ-reparametrization, an IGO algorithm does depend on the choice of parametrization for θ, even if for small δt the difference between two IGO algorithms is $O(\delta t^2)$, one order of magnitude smaller than between IGO and vanilla gradient in general.

So one can wonder how to discretize time in the IGO flow in a fully intrinsic way, not depending at all on a parametrization for θ. A first possibility is given by the IGO-ML algorithm (Definition 11)—this means, for exponential families, that we can decide to single out the parametrization by expectation parameters.

Another, more geometric solution is to use *geodesics* on the statistical manifold. This means we approximate the trajectories of the IGO flow by successive geodesic segments of length δt in the Fisher metric, where the initial direction of each segment is given by the direction of the IGO flow (16). This defines an approximation to the IGO flow that depends on the step size δt and sample size N, but *not* on any choice of parametrization. This approach is fully developed in Bensadon (2015).

A.10 Finite Sample Size and Noisy IGO Flow

The IGO flow is an ideal model of the IGO algorithms. But the IGO flow is deterministic while IGO algorithms are stochastic, depending on a finite number N of random samples. This might result in important differences in their behavior and one can wonder if there is a way to reflect stochasticity directly in the definition of the IGO flow.

The IGO update (16) is a stochastic update

$$\theta^{t+\delta t} = \theta^t + \delta t \sum_{i=1}^{N} \widehat{w}_i \;\; \widetilde{\nabla}_\theta \ln P_\theta(x_i)\Big|_{\theta=\theta^t}$$

because the term $\sum_{i=1}^{N} \widehat{w}_i \;\; \widetilde{\nabla}_\theta \ln P_\theta(x_i)\Big|_{\theta=\theta^t}$ involves a random sample. As such, this term has an expectation and a variance. So for a fixed N and δt, this random update is a weak approximation with step size δt (Kloeden and Platen, 1992, Chapter 9.7) of a stochastic differential equation on θ, whose drift is the expectation of the IGO update (which tends to the IGO flow when $N \to \infty$), and whose noise term is $\sqrt{\delta t}$ times the square root of the covariance matrix of the update applied to a normal random vector.

Such a stochastic differential equation, defining a *noisy IGO flow*, might be a better theoretical object with which to compare the actual behavior of IGO algorithms, than the ideal noiseless IGO flow.

For instance, this strongly suggests that if we have $\delta t \to 0$ while N is kept fixed in an IGO algorithm, noise will disappear (compare Remark 2 in Akimoto et al. 2012).

Second, for large N, one expects the variance of the IGO update to scale like $1/N$, so that the noise term will scale like $\sqrt{\delta t/N}$. This formally suggests that, within reasonable bounds, multiplying or dividing both N and δt by the same factor should result in similar behavior of the algorithm, so that for instance it should be reasonable to reset δt to $10\delta t/N$ and N to 10. (Note that the cost in terms of f-calls of these two algorithms is similar.)

This dependency is reflected in evolution strategies in several ways, provided N is smaller than the search space dimension. First, theoretical results for large search space dimension on the sphere function, $f(x) = \|x\|^2$, indicate that the optimal step size δt for the mean vector is proportional to N, provided the weighting function w is either truncation selection with a fixed truncation ratio (Beyer, 2001) or optimal weights (Arnold, 2006). Second, the learning rate δt of the covariance matrix in CMA-ES is chosen proportional to $\left(\sum_{i=1}^N \widehat{w}_i\right)^2 / \sum_{i=1}^N \widehat{w}_i^2$ which is again proportional to N (Hansen and Kern, 2004). For small enough N, the progress per f-call is then in both cases rather independent of the choice of N.

These results suggest that in implementations, N can be chosen rather freely, whereas δt will be set to $\text{const} \cdot N$. The constant is chosen small enough to ensure stability, but as large as possible to maximize speed; still, $\delta t \leq 1$ is another constraint for (very) large N.

A.11 Influence of the Fisher Geometry of the Statistical Manifold

The global Riemannian geometry of the statistical manifold P_θ might have a bearing on the behavior of stochastic algorithms exploring this manifold. For instance, the Fisher metric identifies the set of 1-dimensional normal distributions $\mathcal{N}(m, \sigma^2)$ with the two-dimensional hyperbolic plane. The latter has negative curvature. The sign of curvature has a strong influence on the behavior of random walks in a Riemannian manifold: in particular, in negative curvature, successive random errors tend to not compensate as much as in the Euclidean case (because geodesics diverge more quickly); this might be relevant to the settings of a stochastic optimization algorithm, suggesting to use larger sample size or smaller steps when curvature is negative; Bensadon (2015) provides first observations in this direction, associated with negative curvature in the space of Gaussians. This is speculative and remains to be explored.

Appendix B. Implementing an IGO Algorithm with a New Family of Probability Distributions: Restricted Boltzmann Machines

In this appendix we show how to apply IGO to restricted Boltzmann machines (RBMs, Smolensky 1986; Ackley et al. 1985), a family of probability distributions on the discrete hypercube that extends Bernoulli distributions and can represent correlations between variables.

This first illustrates how to set up an IGO algorithm from a family of probability distributions for which no fully explicit expression for the Fisher information matrix is available.

Second, we test the relationship between IGO and diversity of p_θ, in view of the entropy-maximizing property of the IGO flow (Proposition 2). We consider a simple function with

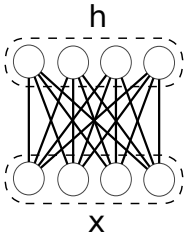

Figure 2: The RBM architecture with the observed ($\mathbf{x}$) and latent ($\mathbf{h}$) variables. In our experiments, a single hidden unit was used.

two optima, and compare the behavior of the IGO flow with that of a vanilla gradient flow. The latter always converges to a single optimum, even though RBMs allow for multimodal distributions. On the other hand, IGO seems to always converge to both optima at once, taking advantage of RBM multimodality.

Finally we interpret this observations by pointing out a non-obvious breach of symmetry between 0 and 1 on $\{0,1\}^d$ for the vanilla gradient of RBMs; the natural gradient automatically compensates for this.

B.1 Restricted Boltzmann Machines.

RBMs first define a joint distribution on $\mathbf{x} \in \{0,1\}^d$ together with a *hidden* or *latent* variable $\mathbf{h} \in \{0,1\}^{d_h}$ (Ghahramani, 2004). Summation over $\mathbf{h}$ provides the distribution over $\mathbf{x}$. The probability associated with an observation $\mathbf{x} = (x_i) \in \{0,1\}^d$ and latent variable $\mathbf{h} = (h_j) \in \{0,1\}^{d_h}$ is

$$P_\theta(\mathbf{x},\mathbf{h}) = \frac{e^{-E(\mathbf{x},\mathbf{h})}}{\sum_{\mathbf{x}',\mathbf{h}'} e^{-E(\mathbf{x}',\mathbf{h}')}}\,, \qquad P_\theta(\mathbf{x}) = \sum_{\mathbf{h}} P_\theta(\mathbf{x},\mathbf{h}), \tag{51}$$

where the *energy function* E is

$$E(\mathbf{x},\mathbf{h}) = -\sum_i a_i x_i - \sum_j b_j h_j - \sum_{i,j} w_{ij} x_i h_j \tag{52}$$

(compare Section 3.3). The distribution is parametrized by $\theta = (\mathbf{a},\mathbf{b},\mathbf{w})$. The "biases" $\mathbf{a}$ and $\mathbf{b}$ can be subsumed into the weights $\mathbf{w}$ by introducing variables x_0 and h_0 always equal to one, which we implicitly do from now on.

The hidden variable $\mathbf{h}$ introduces correlations between the components of $\mathbf{x}$ (see Figure 2). For instance, with $d_h = 1$, the distribution on $\mathbf{x}$ is the sum of two Bernoulli distributions with $\mathbf{h}$ acting as a "switch".

B.2 IGO for RBMs.

RBMs constitute an exponential family with latent variables (with the statistics $T(\mathbf{x},\mathbf{h})$ being all the $x_i h_j$). So the IGO flow (10) is given by (25) where the first contribution is the Fisher matrix (24) and the second contribution is the log-probability derivatives weighted

by the objective function values. However, these expressions are not explicit due to the expectations under P_θ. Still, these expectations can be replaced with Monte Carlo sampling; this is what we will do.

Actually, when applying IGO to such distributions to optimize an objective function $f(\mathbf{x})$, we have two choices. The first is to see the objective function $f(\mathbf{x})$ as a function of $(\mathbf{x}, \mathbf{h})$ where $\mathbf{h}$ is a dummy variable; then we can use the IGO algorithm to optimize over $(\mathbf{x}, \mathbf{h})$ using the distributions $P_\theta(\mathbf{x}, \mathbf{h})$. A second possibility is to marginalize $P_\theta(\mathbf{x}, \mathbf{h})$ over the hidden units $\mathbf{h}$ as in (51), to define the distribution $P_\theta(\mathbf{x})$; then we can use the IGO algorithm to optimize f over $\mathbf{x}$ using $P_\theta(\mathbf{x})$.

These two approaches yield slightly different algorithms. The Fisher matrix for the distributions $P_\theta(\mathbf{x}, \mathbf{h})$ is given by (20) (exponential families) whereas the one for the distributions $P_\theta(\mathbf{x})$ is given by (24) (exponential families with latent variables). For instance, with $I_{w_{ij}w_{i'j'}}$ denoting the entry of the Fisher matrix corresponding to the components w_{ij} and $w_{i'j'}$ of the parameter θ, from (20) we get

$$I_{w_{ij}w_{i'j'}}(\theta) = \mathrm{Cov}_{P_\theta}(x_i h_j, x_{i'} h_{j'}) = \mathbb{E}_{P_\theta}[x_i h_j x_{i'} h_{j'}] - \mathbb{E}_{P_\theta}[x_i h_j]\, \mathbb{E}_{P_\theta}[x_{i'} h_{j'}] \tag{53}$$

whereas from (24) we get the same expression in which each h_j is replaced with its expectation $\bar{h}_j$ knowing $\mathbf{x}$ namely $\bar{h}_j = \mathbb{E}_{P_\theta}[h_j|\mathbf{x}] = \left(1 + e^{-b_j - \sum_i x_i w_{ij}}\right)^{-1}$ and likewise for $h_{j'}$.

Both versions were tested on a small instance and found to be viable. However the version using $(\mathbf{x}, \mathbf{h})$ is numerically more stable and requires fewer samples to get a reliable (in particular, invertible) estimate of the Fisher matrix, both in practice and theory[10]. For this reason we selected the first approach: we optimize f as a function of $(\mathbf{x}, \mathbf{h})$ using IGO for the probability distributions $P_\theta(\mathbf{x}, \mathbf{h})$.

A practical IGO update is thus obtained from (21) by replacing the expectations with Monte Carlo samples $(\mathbf{x}, \mathbf{h})$ from P_θ. A host of strategies are available to sample from RBMs (Ackley et al., 1985; Salakhutdinov and Murray, 2008; Salakhutdinov, 2009; Desjardins et al., 2010); we simply used Gibbs sampling (Hinton., 2002). So the IGO update reads

$$\theta^{t+\delta t} = \theta^t + \delta t\, \widehat{\mathrm{Cov}}(T, T)^{-1}\, \widehat{\mathrm{Cov}}(T, W^f_{\theta^t}) \tag{54}$$

where T denotes the vector of all statistics $T_{ij}(\mathbf{x}, \mathbf{h}) = x_i h_j$ corresponding to the component w_{ij} of the parameter θ, and where $\widehat{\mathrm{Cov}}$ denotes the empirical covariance over a set of Monte Carlo samples $(\mathbf{x}, \mathbf{h})$ taken from $P_{\theta^t}(\mathbf{x}, \mathbf{h})$. Thus $\hat{I} = \widehat{\mathrm{Cov}}(T, T)$ is the estimated Fisher matrix, a matrix of size $\dim(\theta)^2$ as in (53). $\widehat{\mathrm{Cov}}(T, W^f_{\theta^t})$ is a vector of size $\dim(\theta)$ giving the correlation, in the Monte Carlo sample, between the statistics T_{ij} and the ranked values $W^f_{\theta^t}$ of the objective function of the points $(\mathbf{x}, \mathbf{h})$ in the sample: this vector is the sum of weighted log-probability derivatives in the IGO update (17), thanks to (19).

Different sample sizes N_{Fish} and N can be used to evaluate $\widehat{\mathrm{Cov}}(T, T)$ and $\widehat{\mathrm{Cov}}(T, W^f_{\theta^t})$. The sample size for $\widehat{\mathrm{Cov}}(T, W^f_{\theta^t})$ is just the IGO parameter N, the number of points on

10. Indeed, if $I_1(\theta)$ is the Fisher matrix at θ in the first approach and $I_2(\theta)$ in the second approach, we always have $I_1(\theta) \geqslant I_2(\theta)$ in the sense of positive-definite matrices. This is because probability distributions on the pair $(\mathbf{x}, \mathbf{h})$ carry more information than their projections on $\mathbf{x}$ only, and so the Kullback–Leibler distances will always be larger. In particular, there exist values of θ for which the Fisher matrix I_2 is not invertible whereas I_1 is.

which the objective function has to be evaluated; it is typically kept small especially if f is costly. On the other hand it is important to obtain a reliable (in particular, invertible) estimate of the Fisher matrix: invertibility requires a number of samples at least, and ideally much larger than, $\dim(\theta)$, because each point in the sample contributes a rank-1 term to the empirical covariance matrix $\widehat{\mathrm{Cov}}(T,T)$. Increasing N_{Fish} does not require additional f-calls.

B.3 An Experiment with Two Optima: IGO, Diversity, and Multimodal Optimization.

We tested the resulting IGO trajectories on a simple objective function with two optima on $\{0,1\}^d$, namely, the *two-min function based at* $\mathbf{y}$ defined as

$$f_{\mathbf{y}}(\mathbf{x}) = \min\left(\sum_i |x_i - y_i|\,, \sum_i |(1-x_i) - y_i)|\right) \tag{55}$$

which, as a function of $\mathbf{x}$, has two optima, one at $\mathbf{x} = \mathbf{y}$ and the other at its binary complement $\mathbf{x} = \bar{\mathbf{y}}$. The value of the base point $\mathbf{y}$ was randomized for each independent run.

We ran both the IGO algorithm as described above, and a version using the vanilla gradient instead of the natural gradient (that is, omitting the Fisher matrix in the IGO update).

The dimension was $d = 40$ and we used an RBM with only one latent variable ($d_h = 1$). Therefore $\dim(\theta) = 81$. We used a large sample size of $N = 10,000$ for Monte Carlo sampling, so as to be close to the theoretical IGO flow behavior. We also tested a smaller, more realistic sample size of $N = 10$ (still keeping $N_{\mathrm{Fish}} = 10,000$), with similar but noisier results. The selection scheme (Section 2.2) was $w(q) = \mathbb{1}_{q \leqslant 1/5}$ (cf. Rechenberg 1994) so that the best 20% points in the sample are given weight 1 for the update.

The RBM was initialized so that at startup, the distribution P_{θ^0} is close to uniform on $(\mathbf{x}, \mathbf{h})$, in line with Proposition 2. Explicitly we set $w_{ij} \leftarrow \mathcal{N}(0, \frac{1}{d.d_h})$ and then $b_j \leftarrow -\sum_i \frac{w_{ij}}{2}$ and $a_i \leftarrow -\sum_j \frac{w_{ij}}{2} + \mathcal{N}(0, \frac{0.01}{d^2})$ which ensure a close-to-uniform initial distribution.

Full experimental details, including detailed setup and additional results, can be found in a previous version of this article (Ollivier et al., 2011, Section 5). (In particular, IGO runs are frozen when the estimated Fisher matrix becomes singular or unreliable.) The code used for these experiments can be found at `http://www.ludovicarnold.com/projects:igocode` .

The results show that, most of the time, with IGO the distribution P_{θ^t} converges to a distribution giving positive mass to both optima; on the other hand, over 300 independent runs, the same algorithm using the vanilla gradient only converges to one optimum at the expense at the other, so that the vanilla gradient *never* exploited the possibility offered by the RBM to create a bimodal probability distribution on $\{0,1\}^d$. Figure 3 shows ten random runs (out of 300 in our experiments) of the two algorithms: for each of the two optima we plot its distance to the nearest of the points drawn from P_{θ^t}, as a function of time t.[11]

11. Note that the value of δt is not directly comparable between the natural and vanilla gradients. Theory suggests that at startup the IGO trajectory with δt is most comparable to the vanilla gradient trajectory with $4\delta t$, because from (53) most of the diagonal terms of the Fisher matrix are equal to 1/4 and most off-diagonal terms are 0 at startup. The experiments confirm that this yields roughly comparable convergence speeds.

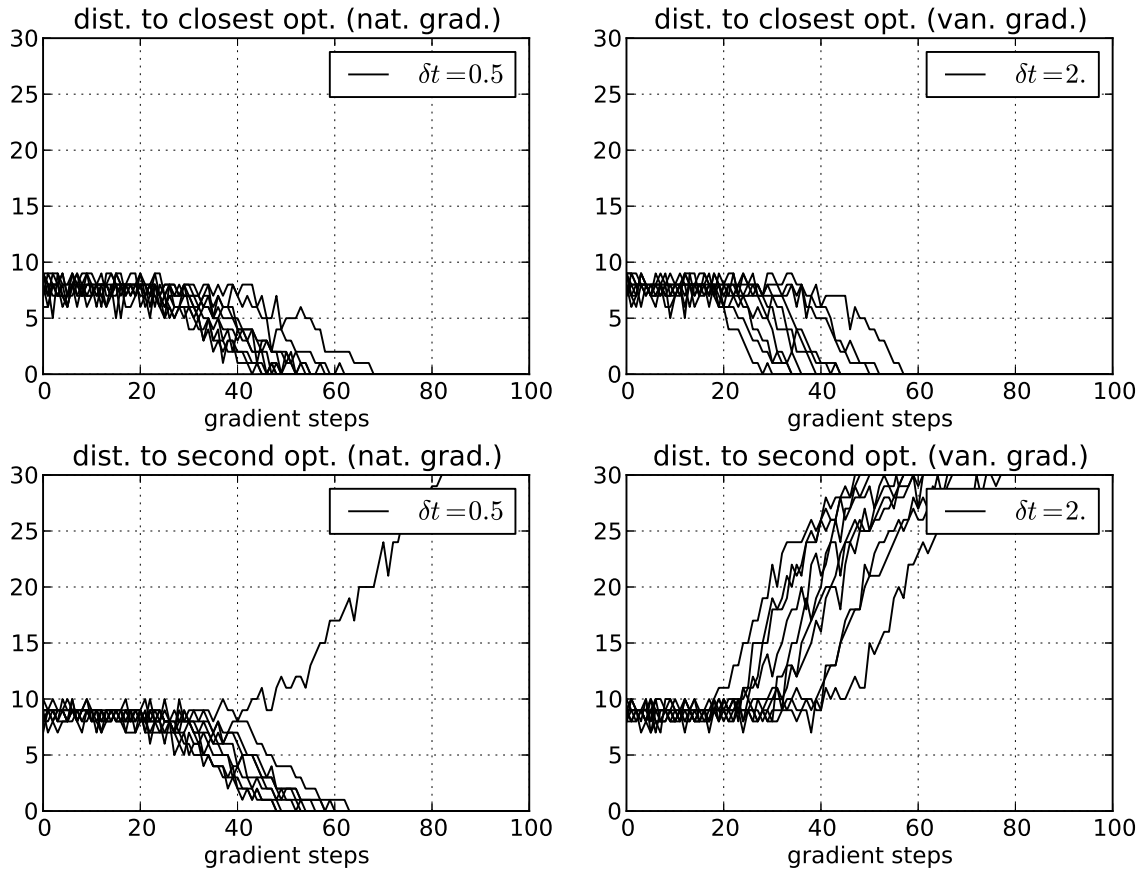

Figure 3: Distance to the two optima during 10 IGO optimization runs and 10 vanilla gradient runs. For each optimum, we plot its distance to the nearest point among the samples from P_{θ^t} found at each step.

This is even clearer when looking at the hidden variable $\mathbf{h}$. With $d_h = 1$ the two possible values $h = 0$ and $h = 1$ can create a sum of two Bernoulli distributions on $\mathbf{x}$. Figure 4 plots the average value of h in the sample for IGO and for the vanilla gradient. As can be seen, over IGO optimization the balance between the two modes is preserved, whereas using vanilla gradient optimization the zero mode is lost.

We interpret this as an illustration of Proposition 2: for a given improvement on the objective function, IGO will favor preserving the diversity (entropy) of P_θ. Using more richly multimodal distributions (e.g., RBMs with $d_h > 1$), this might be useful for multimodal optimization or for optimization situations in which there are several almost equally deep valleys only one of which contains the true optimum.

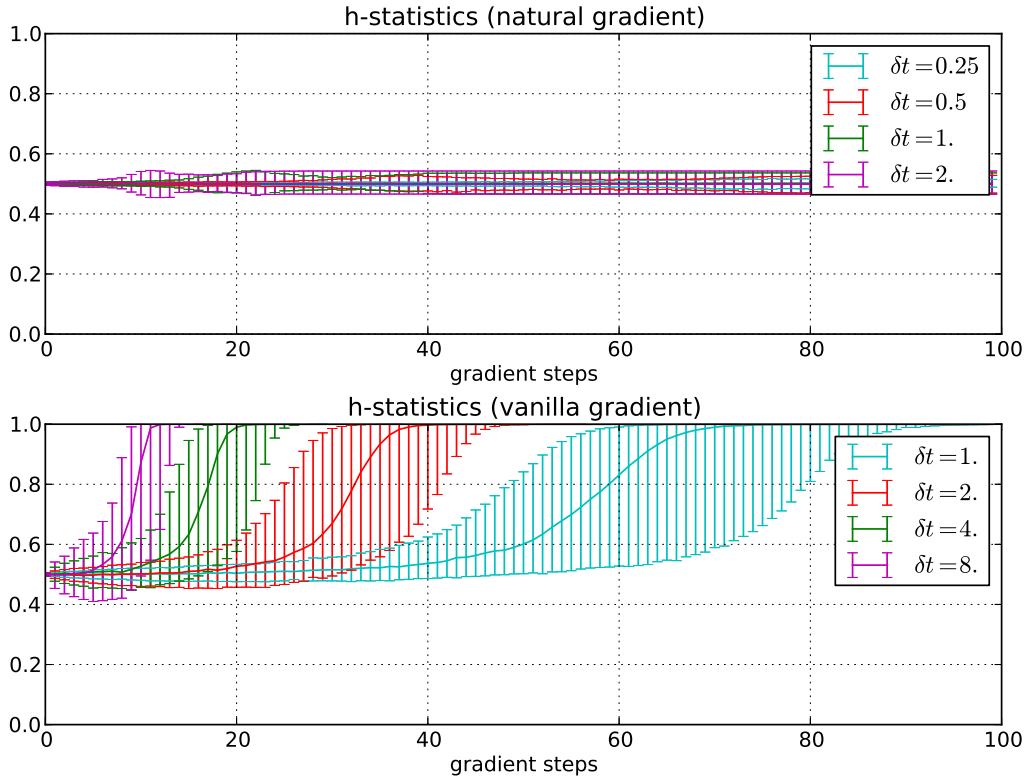

Figure 4: Median over 300 runs of the average of h in the sample, for IGO and vanilla gradient optimization. Error bars indicate the 16th and 84th quantile over the runs.

B.4 Breach of Symmetry by the Vanilla Gradient.

This experiment reveals that, curiously, the vanilla gradient loses multimodality by always setting the hidden variable h to 1, not to 0 (Figure 4). So the vanilla gradient for RBMs seems to favor $h = 1$.

Of course, exchanging the values 0 and 1 for the hidden variables in a restricted Boltzmann machine still gives a distribution of another Boltzmann machine. Changing h_j into $1 - h_j$ is equivalent to resetting $a_i \leftarrow a_i + w_{ij}$, $b_j \leftarrow -b_j$, and $w_{ij} \leftarrow -w_{ij}$.

IGO and the natural gradient are impervious to such a change by Proposition 19. But the vanilla gradient implicitly relies on the Euclidean norm on parameter space, as explained in Section 2.1. For this norm, the distance between the RBM distributions (a_i, b_j, w_{ij}) and (a'_i, b'_j, w'_{ij}) is $\sum_i |a_i - a'_i|^2 + \sum_j \left|b_j - b'_j\right|^2 + \sum_{ij} \left|w_{ij} - w'_{ij}\right|^2$. However, the change of variables $a_i \leftarrow a_i + w_{ij}$, $b_j \leftarrow -b_j$, $w_{ij} \leftarrow -w_{ij}$ does *not* preserve this Euclidean metric. Thus, exchanging 0 and 1 for the hidden variables will result in two different vanilla gradient ascents. The observed asymmetry on h is a consequence of this implicit asymmetry.

Of course it is possible to use parametrizations for which the vanilla gradient will be more symmetric: for instance, using $-1/1$ instead of $0/1$ for the variables, or rewriting the

energy as

$$E(\mathbf{x}, \mathbf{h}) = -\sum_i A_i(x_i - \tfrac{1}{2}) - \sum_j B_j(h_j - \tfrac{1}{2}) - \sum_{i,j} W_{ij}(x_i - \tfrac{1}{2})(h_j - \tfrac{1}{2}) \tag{56}$$

with "bias-free" parameters A_i, B_j, W_{ij} related to the usual parametrization by $w_{ij} = W_{ij}$, $a_i = A_i - \frac{1}{2}\sum_j w_{ij}$, and $b_j = B_j - \frac{1}{2}\sum_i w_{ij}$. The vanilla gradient might perform better in this parametrization.

However, we adopted the approach of using a family of probability distributions found in the literature, with the parametrization commonly found in the literature. We then used the vanilla gradient and the natural gradient on these distributions—and indeed the vanilla gradient, or an approximation thereof, is routinely applied to RBMs in the literature to optimize the log-likelihood of data (Hinton., 2002; Hinton et al., 2006; Bengio et al., 2007). It was not obvious a priori (at least for us) that the vanilla gradient ascent favors $h = 1$. Since the first version of this article was written, this phenomenon has been recognized for Boltzmann machines (Montavon and Müller, 2012).

This directly illustrates the specific influence of the chosen gradient (the two implementations only differ by the inclusion of the Fisher matrix): the natural gradient offers a systematic way to recover symmetry from a non-symmetric gradient update.

Symmetry alone does not explain why IGO reaches the two optima simultaneously: a symmetry-preserving stochastic algorithm could as well end up on either single optimum with 50% probability in each run. The diversity-preserving property of IGO (Proposition 2) offers a reasonable interpretation of why this does not happen.

Appendix C. Further Mathematical Properties of the IGO Flow

This appendix provides further mathematical properties of the IGO flow in general and in specific scenarios.

C.1 Invariance Properties

Here we formally state the invariance properties of the IGO flow under various reparametrizations. Since these results follow from the very construction of the algorithm, the proofs are omitted.

Proposition 17 (f-invariance) *Let $\varphi : \mathbb{R} \to \mathbb{R}$ be a strictly increasing function. Then the trajectories of the IGO flow when optimizing the functions f and $\varphi(f)$ are the same.*

The same is true for the discretized algorithm with population size N and step size $\delta t > 0$.

Proposition 18 (θ-invariance) *Let $\theta' = \varphi(\theta)$ be a smooth bijective function of θ and let $P'_{\theta'} = P_{\varphi^{-1}(\theta')}$. Let θ^t be the trajectory of the IGO flow when optimizing a function f using the distributions P_θ, initialized at θ^0. Then the IGO flow trajectory $(\theta')^t$ obtained from the optimization of the function f using the distributions $P'_{\theta'}$, initialized at $(\theta')^0 = \varphi(\theta^0)$, is the same, namely $(\theta')^t = \varphi(\theta^t)$.*

For the algorithm with finite N and $\delta t > 0$, invariance under reparametrization of θ is only true approximately, in the limit when $\delta t \to 0$. As mentioned above, the IGO update

(17), with $N = \infty$, is simply the Euler approximation scheme for the ordinary differential equation (8) defining the IGO flow. At each step, the Euler scheme is known to make an error $O(\delta t^2)$ with respect to the true flow. This error actually depends on the parametrization of θ.

So the IGO updates for different parametrizations coincide at first order in δt, and may, in general, differ by $O(\delta t^2)$. For instance the difference between the CMA-ES and xNES updates is indeed $O(\delta t^2)$, see Section 5.2.

For comparison, using the vanilla gradient results in a divergence of $O(\delta t)$ at each step between different parametrizations, so this divergence could be of the same magnitude as the steps themselves.

In that sense, one can say that IGO algorithms are "more parametrization-invariant" than other algorithms. This stems from their origin as a discretization of the IGO flow.

However, if the map φ is affine then this phenomenon disappears: parametrizations that differ by an affine map on θ yield the same IGO algorithm.

The next proposition states that, for example, if one uses a family of distributions on $\mathbb{R}^d$ which is invariant under affine transformations, then IGO algorithms optimize equally well a function and its image under any affine transformation (up to an obvious change in the initialization). This proposition generalizes the well-known corresponding property of CMA-ES (Hansen and Auger, 2014, Proposition 9).

This invariance under X-transformations only holds provided the X-transformation preserves the "shape" of the family of probability distributions P_θ, as follows.

Let us define, as usual, the image (*pushforward*) of a probability distribution P by a transformation $\varphi : X \to X$ as the probability distribution P' such that $P'(Y) = P(\varphi^{-1}(Y))$ for any subset $Y \subset X$ (Schilling, 2005, Chapter 7). In the continuous domain, the density of the new distribution P' is obtained by the usual change of variable formula involving the Jacobian of φ (Schilling, 2005, Chapter 15).

We say that a transformation $\varphi : X \to X$ *globally preserves* a family of probability distributions (P_θ), if the image of any P_θ by φ is equal to some distribution $P_{\theta'}$ in the same family, and if moreover the correspondence $\theta \mapsto \theta'$ is locally a diffeomorphism.

Proposition 19 (X-invariance) *Let $\varphi : X \to X$ be a one-to-one transformation of the search space which globally preserves the family of measures P_θ. Let θ^t be the IGO flow trajectory for the optimization of function f, initialized at P_{θ^0}. Let $(\theta')^t$ be the IGO flow trajectory for optimization of $f \circ \varphi^{-1}$, initialized at the image of P_{θ^0} by φ. Then $P_{(\theta')^t}$ is the image of P_{θ^t} by φ.*

For the discretized algorithm with population size N and step size $\delta t > 0$, the same is true up to an error of $O(\delta t^2)$ per iteration. This error disappears if the action of the map φ on Θ by pushforward is affine.

The latter case of affine transforms is well exemplified by CMA-ES: here, using the variance and mean as the parametrization of Gaussians, the new mean and variance after an affine transform of the search space are an affine function of the old mean and variance; specifically, for the affine transformation $A : x \mapsto Ax + b$ we have $(m, C) \mapsto (Am + b, ACA^\mathsf{T})$. Another example, on the discrete search space $X = \{0, 1\}^d$, is the exchange of 0 and 1: for reasonable choices of the family P_θ, the IGO flow and IGO algorithms will be invariant under such a change in the way the data is presented.

C.2 Speed of the IGO Flow

Proposition 20 *The speed of the IGO flow, i.e. the norm of* $\frac{\mathrm{d}\theta^t}{\mathrm{d}t}$ *in the Fisher metric, is at most* $\sqrt{\int_0^1 w^2 - (\int_0^1 w)^2}$ *where* w *is the selection scheme.*

The proof is given in Appendix D.

A bounded speed means that the IGO flow will not explode in finite time, or go out-of-domain if the Fisher metric on the statistical manifold Θ is complete (for instance, the IGO flow on Gaussian distributions will not yield non-positive or degenerate covariance matrices). Due to the approximation terms $O(\delta t^2)$, this may not be true of IGO algorithms.

This speed can be monitored in practice in at least two ways. The first is just to compute the Fisher norm of the increment $\theta^{t+\delta t} - \theta^t$ using the Fisher matrix; for small δt this is close to $\delta t \| \frac{\mathrm{d}\theta}{\mathrm{d}t} \|$ with $\| \cdot \|$ the Fisher metric. The second is as follows: since the Fisher metric coincides with the Kullback–Leibler divergence up to a factor $1/2$, we have $\mathrm{KL}(P_{\theta^{t+\delta t}} \,\|\, P_{\theta^t}) \approx \frac{1}{2} \delta t^2 \| \frac{\mathrm{d}\theta}{\mathrm{d}t} \|^2$ at least for small δt. Since it is relatively easy to estimate $\mathrm{KL}(P_{\theta^{t+\delta t}} \,\|\, P_{\theta^t})$ by comparing the new and old log-likelihoods of points in a Monte Carlo sample, one can obtain an estimate of $\| \frac{\mathrm{d}\theta}{\mathrm{d}t} \|$.

Corollary 21 *Consider an IGO algorithm with selection scheme* w*, step size* δt *and sample size* N*. Then, with probability* 1*,*

$$\frac{\mathrm{KL}(P_{\theta^{t+\delta t}} \,\|\, P_{\theta^t})}{\delta t^2} \leqslant \frac{1}{2} \operatorname{Var}_{[0,1]} w + o(1)_{(\delta t, N) \to (0,\infty)} \ .$$

For instance, with $w(q) = \mathbb{1}_{q \leqslant q_0}$ and neglecting the error terms, an IGO algorithm introduces at most $\frac{1}{2} \delta t^2 \, q_0(1 - q_0)$ bits of information (in base e) per iteration into the probability distribution P_θ. The proof is given in Appendix D.

Thus, the time discretization parameter δt is not just an arbitrary variable: it has an intrinsic interpretation related to a number of bits introduced at each step of the algorithm. This kind of relationship suggests, more generally, to use the Kullback–Leibler divergence as an external and objective way to measure learning rates in those optimization algorithms which use probability distributions.

The result above is only an upper bound. Maximal speed can be achieved only if all "good" points point in the same direction. If the various good points in the sample suggest moves in inconsistent directions, then the IGO update will be much smaller. While non-consistent moves are generally to be expected if $N < \dim \Theta$, it may also be a sign that the signal is noisy, or that the family of distributions P_θ is not well suited to the problem at hand and should be enriched.

As an example, using a family of Gaussian distributions with unkown mean and fixed identity variance on $\mathbb{R}^d$, one checks that for the optimization of a linear function on $\mathbb{R}^d$, with the weight $w(u) = \mathbb{1}_{u<1/2}$, the IGO flow moves at constant speed $1/\sqrt{2\pi} \approx 0.4$, whatever the dimension d. On a rapidly varying sinusoidal function, the moving speed will be much slower because there are "good" and "bad" points in all directions.

This may suggest ways to design the selection scheme w to achieve maximal speed in some instances. Indeed, looking at the proof of the proposition, which involves a Cauchy–Schwarz inequality, one can see that the maximal speed is achieved only if there is a linear relationship

between the weights $W_\theta^f(x)$ and the gradient $\widetilde{\nabla}_\theta \ln P_\theta(x)$. For instance, for the optimization of a linear function on $\mathbb{R}^d$ using Gaussian measures of known variance, the maximal speed will be achieved when the selection scheme $w(u)$ is the inverse of the Gaussian cumulative distribution function. (In particular, $w(u)$ tends to $+\infty$ when $u \to 0$ and to $-\infty$ when $u \to 1$.) This is in accordance with known results: the expected value of the i-th order statistic of N standard Gaussian variates is the optimal $\widehat{w}_i$ value for Gaussians on the sphere function, $f(x) = \sum_i x_i^2$, where $d \to \infty$ (Beyer, 2001; Arnold, 2006). For $N \to \infty$, this order statistic converges to the inverse Gaussian cumulative distribution function.

C.3 Noisy Objective Functions

Suppose that the objective function f is non-deterministic: each time we ask for the value of f at a point $x \in X$, we get a random result. In this setting we may write the random value $f(x)$ as $f(x) = \tilde{f}(x, \omega)$ where ω is an unseen random seed, and $\tilde{f}$ is a deterministic function of x and ω. Without loss of generality, up to a change of variables we can assume that ω is uniformly distributed in $[0, 1]$.

We can still use the IGO algorithm without modification in this context. One might wonder which properties (consistency of sampling, etc.) still apply when f is not deterministic. Actually, IGO algorithms for noisy functions fit very nicely into the IGO framework: the following proposition allows to transfer any property of IGO to the case of noisy functions.

Proposition 22 (Noisy IGO) *Let f be a random function of $x \in X$, namely, $f(x) = \tilde{f}(x, \omega)$ where ω is a random variable uniformly distributed in $[0, 1]$, and $\tilde{f}$ is a deterministic function of x and ω. Then the two following algorithms coincide:*

- *The IGO algorithm* (16)*, using a family of distributions P_θ on space X, applied to the noisy function f, and where the samples are ranked according to the random observed value of f (here we assume that, for each sample, the noise ω is independent from everything else);*

- *The IGO algorithm on space $X \times [0, 1]$, using the family of distributions $\tilde{P}_\theta = P_\theta \otimes U_{[0,1]}$, applied to the deterministic function $\tilde{f}$. Here $U_{[0,1]}$ denotes the uniform law on $[0, 1]$.*

The (easy) proof is given in the Appendix D.

This proposition states that noisy optimization can be modeled as ordinary distribution-based optimization with a component of the distribution being independent of the control parameter θ. Conversely, any component of the search space in which a distribution-based optimization algorithm cannot perform selection or specialization will effectively act as a random noise on the objective function.

As a consequence of this result, all properties of IGO can be transferred to the noisy case. Consider, for instance, consistency of sampling (Theorem 6). The N-sample IGO update rule for the noisy case is identical to the non-noisy case (17):

$$\theta^{t+\delta t} = \theta^t + \delta t\, I^{-1}(\theta^t) \sum_{i=1}^{N} \widehat{w}_i \left.\frac{\partial \ln P_\theta(x_i)}{\partial \theta}\right|_{\theta=\theta^t}$$

where each weight $\widehat{w}_i$ computed from (14) now incorporates noise from the objective function because the rank of x_i is computed on the random function, or equivalently on the deterministic function $\tilde{f}$: $\mathrm{rk}(x_i) = \#\{j, \tilde{f}(x_j, \omega_j) < \tilde{f}(x_i, \omega_i)\}$.

Consistency of sampling (Theorem 6) thus takes the following form through Proposition 22: When $N \to \infty$, the N-sample IGO update rule on the noisy function f converges with probability 1 to the update rule

$$\begin{aligned}\theta^{t+\delta t} &= \theta^t + \delta t \, \widetilde{\nabla}_\theta \int_0^1 \int W^{\tilde{f}}_{\theta^t}(x, \omega) \, P_\theta(\mathrm{d}x) \, \mathrm{d}\omega \\ &= \theta^t + \delta t \, \widetilde{\nabla}_\theta \int \bar{W}^f_{\theta^t}(x) \, P_\theta(\mathrm{d}x) \end{aligned} \tag{57}$$

where $\bar{W}^f_\theta(x) = \mathbb{E}_\omega W^{\tilde{f}}_\theta(x, \omega)$. Thus when $N \to \infty$ the update becomes deterministic as the effects of noise get averaged, as could be expected.

Thus, applying the IGO algorithm to noisy objective functions as in Proposition 22 requires defining the IGO flow for noisy objective functions as

$$\frac{\mathrm{d}\theta^t}{\mathrm{d}t} = \widetilde{\nabla}_\theta \int \bar{W}^f_{\theta^t}(x) \, P_\theta(\mathrm{d}x) \tag{58}$$

where $\bar{W}^f_\theta(x) = \mathbb{E}_\omega W^{\tilde{f}}_\theta(x, \omega)$ is the *average* weight of x over the values $f(x)$. Thus, the effects of noise remain visible even for $N \to \infty$, as $\bar{W}$ is in general flatter than W.

Note that there is another possible way to define the IGO flow for noisy objective functions. The flow (58) amounts to taking a point x, computing a (random) value $f(x)$, computing the level $q^<_\theta(x) = \Pr_{x' \sim P_\theta}(f(x') < f(x))$ of this random value $f(x)$, applying the selection scheme w, and then averaging. In this approach the value $q^<_\theta(x)$ is a random variable depending on the value of $f(x)$. Alternatively, we could define the value $q_\theta(x)$ by applying Definition 3 unchanged, namely, $q^<_\theta(x) = \Pr_{x' \sim P_\theta}(f(x') < f(x))$ in which $f(x)$ itself is treated as a random variable under the probability Pr, so that $q^<_\theta(x)$ is deterministic and takes into account all possible values for $f(x)$. Then we could apply the selection scheme w to this value $q^<_\theta(x)$ as in Definition 3 and define the IGO flow accordingly. The value of $q^<_\theta(x)$ in this second version is the expected value of $q^<_\theta(x)$ in the first version.

When the selection scheme w is affine, these two approaches coincide; however this is not the case in general, as the second version averages the q-values while the first version averages the weights $w(q)$. The second version would be the $N \to \infty$ limit of a slightly more complex algorithm using several evaluations of f for each sample x_i in order to compute noise-free average ranks and quantiles. The first version has the advantage of leaving the algorithm unchanged and of inheriting properties from the deterministic case via Proposition 22.

C.4 The IGO Flow for Linear Functions on $\{0,1\}^d$ and $\mathbb{R}^d$

In this section we take a closer look at the IGO differential equation solutions of (8) for some simple examples of objective functions, for which it is possible to obtain exact information about these IGO trajectories.

We start with the discrete search space $X = \{0,1\}^d$ and linear functions (to be minimized) defined as $f(x) = c - \sum_{i=1}^d \alpha_i x_i$ with $\alpha_i > 0$. (So maximization of the classical onemax function $f_{\mathrm{onemax}}(x) = \sum_{i=1}^d x_i$ is covered by setting $\alpha_i = 1$.) The differential equation of

the IGO flow (8) for the Bernoulli measures $P_\theta(x) = p_{\theta_1}(x_1) \dots p_{\theta_d}(x_d)$ defined on X is the $\delta t \to 0$ limit of the IGO-PBIL update (36):

$$\frac{\mathrm{d}\theta_i^t}{\mathrm{d}t} = \int W_{\theta^t}^f(x)(x_i - \theta_i^t) P_{\theta^t}(\mathrm{d}x) =: g_i(\theta^t) \ . \tag{59}$$

Although finding the analytical solution of the differential equation (59) for any initial condition seems a bit intricate, we show that the solution of (59) converges to $(1, \dots, 1)$ starting from any initial $\theta \in (0,1]^d$. Note that starting with $\theta_i = 0$ for some i prevents IGO (and PBIL) from sampling any 1 for component i, so that the components of θ that are equal to 0 at startup will always stay so; in that case the IGO flow effectively works as in a smaller-dimensional space.

To prove convergence to $(1, \dots, 1)$, we establish the following result:

Lemma 23 *Assume that the selection scheme $w : [0,1] \to \mathbb{R}$ is bounded, that w is a nonincreasing function and that there exists $q_0, q_1 \in (0,1)$ such that $w(q_0) \neq w(q_1)$. On $f(x) = c - \sum_{i=1}^d \alpha_i x_i$, the solution of (59) satisfies $\sum_{i=1}^d \alpha_i \frac{\mathrm{d}\theta_i^t}{\mathrm{d}t} \geqslant 0$; moreover $\sum \alpha_i \frac{\mathrm{d}\theta_i^t}{\mathrm{d}t} = 0$ if and only $\theta \in \{0,1\}^d$.*

Proof We compute $\sum_{i=1}^d \alpha_i g_i(\theta^t)$ and find that

$$\begin{aligned}
\sum_{i=1}^d \alpha_i \frac{\mathrm{d}\theta_i^t}{\mathrm{d}t} &= \int W_{\theta^t}^f(x) \left(\sum_{i=1}^d \alpha_i x_i - \sum_{i=1}^d \alpha_i \theta_i^t \right) P_{\theta^t}(\mathrm{d}x) \\
&= \int W_{\theta^t}^f(x)(f(\theta^t) - f(x)) P_{\theta^t}(\mathrm{d}x) \\
&= \mathbb{E}[W_{\theta^t}^f(x)]\, \mathbb{E}[f(x)] - \mathbb{E}[W_{\theta^t}^f(x) f(x)]
\end{aligned}$$

where the expectations are taken under P_{θ^t}. Using Lemma 24 below, we have that $\mathbb{E}[W_{\theta^t}^f(x)]\, \mathbb{E}[f(x)] - \mathbb{E}[W_{\theta^t}^f(x) f(x)] \geqslant 0$ and in addition $\mathbb{E}[W_{\theta^t}^f(x)]\, \mathbb{E}[f(x)] - \mathbb{E}[W_{\theta^t}^f(x) f(x)] = 0$ if and only if $\theta^t \in \{0,1\}^d$. ■

Lemma 24 *Under the assumptions of Lemma 23*

$$\mathbb{E}[-W_{\theta^t}^f(x) f(x)] \geqslant \mathbb{E}[-W_{\theta^t}^f(x)] \mathbb{E}[f(x)] \tag{60}$$

with equality if and only if $\theta^t \in \{0,1\}^d$. Here the expectations are taken under $x \sim P_{\theta^t}$.

Proof We want to prove that $\mathrm{Cov}[-W_{\theta^t}^f(x), f(x)] \geqslant 0$ with equality if and only if $\theta^t \in \{0,1\}^d$. Let us define the random variable $Z = f(x)$ when $x \sim P_{\theta^t}$. Remark that $W_{\theta^t}^f(x) = \mathscr{G}(Z)$ where

$$\mathscr{G}(z) = \frac{1}{F_{\theta^t}^{\leqslant}(z) - F_{\theta^t}^{<}(z)} \int_{F_{\theta^t}^{<}(z)}^{F_{\theta^t}^{\leqslant}(z)} w(q) dq \ , \tag{61}$$

where $F_{\theta^t}^{\leqslant}(z) = \Pr_{x' \sim P_{\theta^t}}(f(x') \leqslant z)$ and $F_{\theta^t}^{<}(z) = \Pr_{x' \sim P_{\theta^t}}(f(x') < z)$. That is $\mathscr{G}$ is the average of w on the interval $(F_{\theta^t}^{<}(z), F_{\theta^t}^{\leqslant}(z))$. Since w is nonincreasing, the function $\mathscr{G}$ is also nonincreasing. We have the following equality

$$\mathrm{Cov}[-W_{\theta^t}^f(x), f(x)] = \mathrm{Cov}[-\mathscr{G}(Z), Z]$$

where $-\mathscr{G}$ is nondecreasing. Following (Thorisson, 2000, Chapter 1), we write

$$\mathrm{Cov}[-\mathscr{G}(Z), Z] = \frac{1}{2}\,\mathrm{Cov}[-\mathscr{G}(Z) + \mathscr{G}(Z'), Z - Z']$$

where Z and Z' are independent following the distribution P_{θ^t}. Given that both the mean of $-\mathscr{G}(Z) + \mathscr{G}(Z')$ and $Z - Z'$ are zero

$$\mathrm{Cov}[-\mathscr{G}(Z), Z] = \frac{1}{2}E[(-\mathscr{G}(Z) + \mathscr{G}(Z'))(Z - Z')] = \frac{1}{2}E[|-\mathscr{G}(Z) + \mathscr{G}(Z')||Z - Z'|] \geqslant 0 \ ,$$

where the last equality holds because $-\mathscr{G}$ is nondecreasing. This proves the main statement of the lemma.

We will now show that if $\theta^t \notin \{0,1\}^d$, then $E[|-\mathscr{G}(Z) + \mathscr{G}(Z')||Z - Z'|] > 0$ and thus consequently $\mathrm{Cov}[-W^f_{\theta^t}(x), f(x)] > 0$. We simply need to show that Z can take with strictly positive probabilities p_1 and p_2 two distinct values z_1 and z_2 such that $\mathscr{G}(z_1)$ and $\mathscr{G}(z_2)$ are distinct. We will then have

$$E[|-\mathscr{G}(Z) + \mathscr{G}(Z')||Z - Z'|] \geqslant (|-\mathscr{G}(z_1) + \mathscr{G}(z_2)||z_1 - z_2|)\, p_1 p_2 > 0 \ .$$

Let us assume that one single θ_i belongs to $(0,1)$ and all the others are either 0 or 1. We can assume without loss of generality that $\theta_1 \in (0,1)$. Then $f(x)$ takes (only) two distinct values with positive probability; let z_1 be the one associated with $x_1 = 1$ and z_2 with $x_1 = 0$. Then $z_1 < z_2$ thanks to the definition of f and because $\alpha_i > 0$. Moreover, unravelling the definitions we find $\mathscr{G}(z_1) = \frac{1}{\theta_1}\int_0^{\theta_1} w(q)dq$ and $\mathscr{G}(z_2) = \frac{1}{1-\theta_1}\int_{\theta_1}^1 w(q)dq$.

The assumption states that there exists $q_0, q_1 \in (0,1)$ such that $w(q_0) \neq w(q_1)$. Assume without loss of generality that $q_0 < q_1$ and $w(q_0) > w(q_1)$. Either $w(q_1) \geqslant w(\theta_1)$ or $w(q_1) < w(\theta_1)$. In the first case, then $\theta_1 > q_0$ and

$$\frac{1}{\theta_1}\int_0^{\theta_1} w(q)dq = \frac{1}{\theta_1}\left(\int_0^{q_0} w(q)dq + \int_{q_0}^{\theta_1} w(q)dq\right) > \frac{1}{\theta_1}\left(w(q_0)q_0 + w(\theta_1)(\theta_1 - q_0)\right) > w(\theta_1) \ ,$$

Meanwhile, $\mathscr{G}(z_2) = \frac{1}{1-\theta_1}\int_{\theta_1}^1 w(q)dq \leqslant w(\theta_1)$ and thus

$$\mathscr{G}(z_1) > \mathscr{G}(z_2) \ .$$

If $w(q_1) < w(\theta_1)$, then $q_1 > \theta_1$ and

$$\begin{aligned}\mathscr{G}(z_2) = \frac{1}{1-\theta_1}\int_{\theta_1}^1 w(q)dq &= \frac{1}{1-\theta_1}\left(\int_{\theta_1}^{q_1} w(q)dq + \int_{q_1}^1 w(q)dq\right) \\ &\leqslant \frac{1}{1-\theta_1}(w(\theta_1)(q_1 - \theta_1) + w(q_1)(1 - q_1)) < w(\theta_1) \quad (62)\end{aligned}$$

and thus $\mathscr{G}(z_2) < w(\theta_1) \leqslant \mathscr{G}(z_1)$ as well.

In the case where we have not only two but l distinct fitnesses that can be sampled with strictly positive probabilities, say $z_1 < \ldots < z_l$, we can show similarly that $\mathscr{G}(z_1) > \mathscr{G}(z_l)$. ■

We are now ready to state the convergence of the trajectories of IGO-PBIL (and cGA) update in the limit of $\delta t \to 0$.

Proposition 25 *Assume that the selection scheme $w : [0,1] \to \mathbb{R}$ is bounded, that w is a nonincreasing function and that there exists $q_0, q_1 \in (0,1)$ such that $w(q_0) \neq w(q_1)$. On the linear functions $f(x) = c - \sum_{i=1}^d \alpha_i x_i$, the critical point $\theta = (1, \dots, 1)$ of the IGO-PBIL differential equation* (59) *is stable and for any initial condition $\theta \in (0;1]^d$, the continuous-time trajectory solving* (59) *converges to* $(1, \dots, 1)$.

Proof Remark first that if we start at $\theta^0 \in [\varepsilon, 1]^d$, then the solution of (59) stays in $[\varepsilon, 1]^d$. Indeed, the trajectory cannot go out of $[0,1]^d$ and, in addition, for $\theta \in [\varepsilon, 1)^d$ we have $g_i(\theta) \geqslant 0$ so that the trajectory of (59) cannot go out of $[\varepsilon, 1]^d$. The proof that $g_i(\theta) \geqslant 0$ for $\theta \in [\varepsilon, 1)^d$ is similar to the proof that $\mathrm{Cov}[-W^f_{\theta^t}(x), f(x)] \geqslant 0$ in Lemma 24: namely, we can write $g_i(\theta) = \int \mathbb{E}[W^f_{\theta^t}(x)|x_i](x_i - \theta_i) p_{\theta_i}(x_i) dx_i = \theta_i(1-\theta_i)[h(1) - h(0)]$ where $h(x_i) = \mathbb{E}[W^f_{\theta^t}(x)|x_i] = \mathbb{E}[\mathscr{G}(f(x))|x_i]$ where $\mathscr{G}$ is defined in (61). Then $h(1) \geqslant h(0)$ comes from the increase of $\mathscr{G}$ when going to better function values.

Consider now on $[\varepsilon, 1]^d$, the non-negative function $V(\theta) = \sum_{i=1}^d \alpha_i - \sum_{i=1}^d \alpha_i \theta_i$. Then $V^*(\theta) := \nabla V(\theta) \cdot g(\theta) \leqslant 0$, and $V^*(\theta)$ equals zero only for $\theta \in \{0,1\}^d$ according to Lemma 23. Hence for $\theta \in [\varepsilon, 1]^d$, the function V is a Lyapunov function (Khalil, 1996; Agarwal and O'Regan, 2008), which is minimal at $(1, \dots, 1)$ and such that $V^*(\theta) < 0$ except at $(1, \dots, 1)$. Therefore the trajectory of (59) will converge to $(1, \dots, 1)$ as t goes to infinity (the proof is similar to that of (Khalil, 1996, Theorem 4.1)). Given that this holds for any $\varepsilon > 0$, we can conclude that the trajectory of (59) converges to $(1, \dots, 1)$ starting from any $\theta \in (0,1]^d$. ■

We now consider on $\mathbb{R}^d$ the family of multivariate normal distributions $P_\theta = \mathcal{N}(m, \sigma^2 I_d)$ with covariance matrix equal to $\sigma^2 I_d$. The parameter θ thus has $d+1$ components $\theta = (m, \sigma) \in \mathbb{R}^d \times \mathbb{R}$. The natural gradient update using this family was derived in Glasmachers et al. (2010); from this we can derive the IGO differential equation which reads:

$$\frac{\mathrm{d}m^t}{\mathrm{d}t} = \int_{\mathbb{R}^d} W^f_{\theta^t}(x)(x - m^t) P_{\mathcal{N}(m^t, (\sigma^t)^2 I_d)}(x) \mathrm{d}x \tag{63}$$

$$\frac{\mathrm{d}\tilde{\sigma}^t}{\mathrm{d}t} = \int_{\mathbb{R}^d} \frac{1}{2d} \left\{ \sum_{i=1}^d \left(\frac{x_i - m_i^t}{\sigma^t} \right)^2 - 1 \right\} W^f_{\theta^t}(x) P_{\mathcal{N}(m^t, (\sigma^t)^2 I_d)}(x) \mathrm{d}x \tag{64}$$

where σ^t and $\tilde{\sigma}^t$ are linked via $\sigma^t = \exp(\tilde{\sigma}^t)$ or $\tilde{\sigma}^t = \ln(\sigma^t)$. Denoting $\mathcal{N}$ a random vector following a centered multivariate normal distribution with identity covariance matrix we can rewrite the gradient flow as

$$\frac{\mathrm{d}m^t}{\mathrm{d}t} = \sigma^t \mathbb{E}\left[W^f_{\theta^t}(m^t + \sigma^t \mathcal{N}) \mathcal{N} \right] \tag{65}$$

$$\frac{\mathrm{d}\tilde{\sigma}^t}{\mathrm{d}t} = \mathbb{E}\left[\frac{1}{2} \left(\frac{\|\mathcal{N}\|^2}{d} - 1 \right) W^f_{\theta^t}(m^t + \sigma^t \mathcal{N}) \right] \ . \tag{66}$$

Let us analyze the solution of the previous system on linear functions. Without loss of generality (thanks to invariance) we can consider the linear function $f(x) = x_1$. We have

$$W^f_{\theta^t}(x) = w(\Pr(m_1^t + \sigma^t Z_1 < x_1))$$

where Z_1 follows a standard one-dimensional normal distribution and thus

$$W^f_{\theta^t}(m^t + \sigma^t \mathcal{N}) = w(\Pr_{Z_1 \sim \mathcal{N}(0,1)}(Z_1 < \mathcal{N}_1)) \tag{67}$$
$$= w(\mathcal{F}(\mathcal{N}_1)) \tag{68}$$

with $\mathcal{F}$ the cumulative distribution of a standard normal distribution, and $\mathcal{N}_1$ the first component of $\mathcal{N}$. The differential equation thus simplifies into

$$\frac{\mathrm{d}m^t}{\mathrm{d}t} = \sigma^t \begin{pmatrix} \mathbb{E}\left[w(\mathcal{F}(\mathcal{N}_1))\mathcal{N}_1\right] \\ 0 \\ \vdots \\ 0 \end{pmatrix} \tag{69}$$

$$\frac{\mathrm{d}\tilde{\sigma}^t}{\mathrm{d}t} = \frac{1}{2d}\mathbb{E}\left[(|\mathcal{N}_1|^2 - 1)w(\mathcal{F}(\mathcal{N}_1))\right] \ . \tag{70}$$

Consider, for example, the truncation selection function, i.e. $w(q) = 1_{q \leqslant q_0}$ where $q_0 \in (0, 1)$. (Within the IGO algorithm, this results in so-called *intermediate recombination* of the $q_0 \times N$ best samples.) We find that

$$\frac{\mathrm{d}m^t_1}{\mathrm{d}t} = \sigma^t \mathbb{E}[\mathcal{N}_1 1_{\{\mathcal{N}_1 \leqslant \mathcal{F}^{-1}(q_0)\}}] =: \sigma^t \beta \tag{71}$$

$$\frac{\mathrm{d}\tilde{\sigma}^t}{\mathrm{d}t} = \frac{1}{2d}\left(\int_0^{q_0} \mathcal{F}^{-1}(u)^2 \mathrm{d}u - q_0\right) =: \alpha \ . \tag{72}$$

The solution of the IGO flow for the linear function $f(x) = x_1$ is thus given by

$$m^t_1 = m^0_1 + \frac{\sigma^0 \beta}{\alpha}\exp(\alpha t) \tag{73}$$

$$\sigma^t = \sigma^0 \exp(\alpha t) \ . \tag{74}$$

The coefficient β is negative for any $q_0 < 1$. The coefficient α is positive if and only if $q_0 < 1/2$ by a simple calculus argument[12]; this corresponds to selecting less than half of the sampled points in an ES. In this case the step size σ^t grows exponentially fast to infinity and the mean vector moves along the gradient direction towards minus ∞ at the same rate. If more than half of the points are selected, $q_0 \geqslant 1/2$, the step size will decrease to zero exponentially fast and the mean vector will get stuck (compare also Grahl et al. 2005; Hansen 2006a; Pošík 2008).

For an analysis of the solutions of the system of differential equations (65) and (66) on more complex functions, namely convex-quadratic functions and twice continuously differentiable functions, we refer to Akimoto et al. (2012).

12. Indeed $\alpha = \frac{1}{2d\sqrt{2\pi}} \int_{-\infty}^{\mathcal{F}^{-1}(q_0)} (x^2 - 1)\exp(-x^2/2)\mathrm{d}x = \frac{1}{2d\sqrt{2\pi}} g(\mathcal{F}^{-1}(q_0))$ where $g(y) = \int_{-\infty}^{y} (x^2 - 1)\exp(-x^2/2)\mathrm{d}x$. Using $g(0) = 0$ and $\lim_{y \to \pm\infty} g(y) = 0$, and studying the sign of $g'(y)$, we find that g is positive for $y < 0$ and negative for $y > 0$. Since $\mathcal{F}^{-1}(q_0) < 0$ if and only if $q_0 < 1/2$, we find that $\alpha = \frac{1}{2d\sqrt{2\pi}} g(\mathcal{F}^{-1}(q_0))$ is positive if and only if $q_0 < 1/2$.

Appendix D. Proofs

This final appendix provides longer proofs of propositions and theorems of the paper.

D.1 Proof of Proposition 2

We begin with a calculus lemma which will also be used for the proof of Theorem 10. The proof is omitted and amounts to writing the maximum of a quadratic function obtained by the second-order Taylor expansion of f.

Lemma 26 *Let f be real-valued function on a finite-dimensional vector space E equipped with a positive definite quadratic form $\|\cdot\|^2$. Assume f is smooth and has at most quadratic growth at infinity. Then, for any $x \in E$, we have*

$$\nabla f(x) = \lim_{\varepsilon \to 0} \frac{1}{\varepsilon} \arg\max_h \left\{ f(x+h) - \frac{1}{2\varepsilon}\|h\|^2 \right\}$$

where ∇ is the gradient associated with the norm $\|\cdot\|$. Equivalently,

$$\arg\max_y \left\{ f(y) - \frac{1}{2\varepsilon}\|y-x\|^2 \right\} = x + \varepsilon \nabla f(x) + O(\varepsilon^2)$$

when $\varepsilon \to 0$.

Proposition 2 follows by taking the Fisher information metric at θ^0 for the metric $\|\cdot\|^2$, and using the relation $\mathrm{KL}(P \,\|\, Q) = -\operatorname{Ent}(P) + \log \# X$ if Q is the uniform distribution on a finite space X.

D.2 Proof of Theorem 6 (Convergence of Empirical Means and Quantiles)

Let us give a more precise statement including the necessary regularity conditions.

Proposition 27 *Let $\theta \in \Theta$. Assume that the derivative $\frac{\partial \ln P_\theta(x)}{\partial \theta}$ exists for P_θ-almost all $x \in X$ and that $\mathbb{E}_{P_\theta} \left| \frac{\partial \ln P_\theta(x)}{\partial \theta} \right|^2 < +\infty$. Assume that the function w is non-decreasing and bounded.*

Let $(x_i)_{i \in \mathbb{N}}$ be a sequence of independent samples of P_θ. Then with probability 1, as $N \to \infty$ we have

$$\frac{1}{N} \sum_{i=1}^{N} \widehat{W}^f(x_i) \frac{\partial \ln P_\theta(x_i)}{\partial \theta} \to \int W_\theta^f(x) \, \frac{\partial \ln P_\theta(x)}{\partial \theta} \, P_\theta(\mathrm{d}x)$$

where

$$\widehat{W}^f(x_i) = w\left(\frac{\mathrm{rk}_N(x_i) + 1/2}{N} \right)$$

with $\mathrm{rk}_N(x_i) = \#\{1 \leqslant j \leqslant N,\, f(x_j) < f(x_i)\}$. (When there are f-ties in the sample, $\widehat{W}^f(x_i)$ is defined as the average of $w((r+1/2)/N)$ over the possible rankings r of x_i.)

Proof Let $g : X \to \mathbb{R}$ be any function with $\mathbb{E}_{P_\theta} g^2 < \infty$. We will show that $\frac{1}{N}\sum \widehat{W}^f(x_i)g(x_i) \to \int W_\theta^f(x)g(x)\,P_\theta(\mathrm{d}x)$. Applying this with g equal to the components of $\frac{\partial \ln P_\theta(x)}{\partial\theta}$ will yield the result.

Let us decompose

$$\frac{1}{N}\sum \widehat{W}^f(x_i)g(x_i) = \frac{1}{N}\sum W_\theta^f(x_i)g(x_i) + \frac{1}{N}\sum(\widehat{W}^f(x_i) - W_\theta^f(x_i))g(x_i).$$

Each summand in the first term involves only one sample x_i (contrary to $\widehat{W}^f(x_i)$ which depends on the whole sample). So by the strong law of large numbers, almost surely $\frac{1}{N}\sum W_\theta^f(x_i)g(x_i)$ converges to $\int W_\theta^f(x)g(x)\,P_\theta(\mathrm{d}x)$. So we have to show that the second term converges to 0 almost surely.

By the Cauchy–Schwarz inequality, we have

$$\left|\frac{1}{N}\sum(\widehat{W}^f(x_i) - W_\theta^f(x_i))g(x_i)\right|^2 \leqslant \left(\frac{1}{N}\sum(\widehat{W}^f(x_i) - W_\theta^f(x_i))^2\right)\left(\frac{1}{N}\sum g(x_i)^2\right)$$

By the strong law of large numbers, the second term $\frac{1}{N}\sum g(x_i)^2$ converges to $\mathbb{E}_{P_\theta} g^2$ almost surely. So we have to prove that the first term $\frac{1}{N}\sum(\widehat{W}^f(x_i) - W_\theta^f(x_i))^2$ converges to 0 almost surely.

Since w is bounded by assumption, we can write

$$\begin{aligned}(\widehat{W}^f(x_i) - W_\theta^f(x_i))^2 &\leqslant 2B\left|\widehat{W}^f(x_i) - W_\theta^f(x_i)\right| \\ &= 2B\left|\widehat{W}^f(x_i) - W_\theta^f(x_i)\right|_+ + 2B\left|\widehat{W}^f(x_i) - W_\theta^f(x_i)\right|_-\end{aligned}$$

where B is the bound on $|w|$. We will bound each of these terms.

Let us abbreviate $q_i^< = \Pr_{x'\sim P_\theta}(f(x') < f(x_i))$, $q_i^\leqslant = \Pr_{x'\sim P_\theta}(f(x') \leqslant f(x_i))$, $r_i^< = \#\{j \leqslant N,\, f(x_j) < f(x_i)\}$, $r_i^\leqslant = \#\{j \leqslant N,\, f(x_j) \leqslant f(x_i)\}$.

By definition of $\widehat{W}^f$ we have

$$\widehat{W}^f(x_i) = \frac{1}{r_i^\leqslant - r_i^<}\sum_{k=r_i^<}^{r_i^\leqslant - 1} w((k+1/2)/N)$$

and moreover $W_\theta^f(x_i) = w(q_i^<)$ if $q_i^< = q_i^\leqslant$ or $W_\theta^f(x_i) = \frac{1}{q_i^\leqslant - q_i^<}\int_{q_i^<}^{q_i^\leqslant} w$ otherwise.

The Glivenko–Cantelli theorem (Billingsley, 1995, Theorem 20.6) implies that $\sup_i \left|q_i^\leqslant - r_i^\leqslant/N\right|$ tends to 0 almost surely, and likewise for $\sup_i |q_i^< - r_i^</N|$. So let N be large enough so that these errors are bounded by ε.

Since w is non-increasing, we have $w(q_i^<) \leqslant w(r_i^</N - \varepsilon)$. In the case $q_i^< \neq q_i^\leqslant$, we decompose the interval $[q_i^<; q_i^\leqslant]$ into $(r_i^\leqslant - r_i^<)$ subintervals. The average of w over each such subinterval is compared to a term in the sum defining $w^N(x_i)$: since w is non-increasing, the average of w over the k^{th} subinterval is at most $w((r_i^< + k)/N - \varepsilon)$. So we get

$$W_\theta^f(x_i) \leqslant \frac{1}{r_i^\leqslant - r_i^<}\sum_{k=r_i^<}^{r_i^\leqslant - 1} w(k/N - \varepsilon)$$

so that

$$W_\theta^f(x_i) - \widehat{W}^f(x_i) \leqslant \frac{1}{r_i^{\leqslant} - r_i^{<}} \sum_{k=r_i^{<}}^{r_i^{\leqslant}-1} (w(k/N - \varepsilon) - w((k+1/2)/N)).$$

Let us sum over i, remembering that there are $(r_i^{\leqslant} - r_i^{<})$ values of j for which $f(x_j) = f(x_i)$. Taking the positive part, we get

$$\frac{1}{N}\sum_{i=1}^{N} \left| W_\theta^f(x_i) - \widehat{W}^f(x_i) \right|_+ \leqslant \frac{1}{N} \sum_{k=0}^{N-1} (w(k/N - \varepsilon) - w((k+1/2)/N)).$$

Since w is non-increasing we have

$$\frac{1}{N}\sum_{k=0}^{N-1} w(k/N - \varepsilon) \leqslant \int_{-\varepsilon - 1/N}^{1-\varepsilon-1/N} w$$

and

$$\frac{1}{N}\sum_{k=0}^{N-1} w((k+1/2)/N) \geqslant \int_{1/2N}^{1+1/2N} w$$

(we implicitly extend the range of w so that $w(q) = w(0)$ for $q < 0$ and likewise for $q > 1$). So we have

$$\frac{1}{N}\sum_{i=1}^{N} \left| W_\theta^f(x_i) - \widehat{W}^f(x_i) \right|_+ \leqslant \int_{-\varepsilon-1/N}^{1/2N} w - \int_{1-\varepsilon-1/N}^{1+1/2N} w \leqslant (2\varepsilon + 3/N)B$$

where B is the bound on $|w|$.

Reasoning symmetrically with $w(k/N + \varepsilon)$ and the inequalities reversed, we get a similar bound for $\frac{1}{N}\sum \left| W_\theta^f(x_i) - \widehat{W}^f(x_i) \right|_-$. This ends the proof. ■

D.3 Proof of Proposition 7 (Quantile Improvement)

Let us use the weight $w(u) = \mathbb{1}_{u \leqslant q}$. Let m be the value of the q-quantile of f under P_{θ^t}. We want to show that the value of the q-quantile of f under $P_{\theta^{t+\delta t}}$ is less than m, unless the gradient vanishes and the IGO flow is stationary.

Let $p_- = \Pr_{x\sim P_{\theta^t}}(f(x) < m)$, $p_m = \Pr_{x\sim P_{\theta^t}}(f(x) = m)$ and $p_+ = \Pr_{x\sim P_{\theta^t}}(f(x) > m)$. By definition of the quantile value we have $p_- + p_m \geqslant q$ and $p_+ + p_m \geqslant 1 - q$. Let us assume that we are in the more complicated case $p_m \neq 0$ (for the case $p_m = 0$, simply remove the corresponding terms).

We have $W_{\theta_t}^f(x) = 1$ if $f(x) < m$, $W_{\theta_t}^f(x) = 0$ if $f(x) > m$ and $W_{\theta_t}^f(x) = \frac{1}{p_m}\int_{p_-}^{p_- + p_m} w(u)\mathrm{d}u = \frac{q - p_-}{p_m}$ if $f(x) = m$.

Using the same notation as above, let $g_t(\theta) = \int W_{\theta^t}^f(x)\, P_\theta(\mathrm{d}x)$. Decomposing this integral on the three sets $f(x) < m$, $f(x) = m$ and $f(x) > m$, we get that $g_t(\theta) = \Pr_{x\sim P_\theta}(f(x) < m) + \Pr_{x\sim P_\theta}(f(x) = m)\frac{q-p_-}{p_m}$. In particular, $g_t(\theta^t) = q$.

Since we follow a gradient ascent of g_t, for δt small enough we have $g_t(\theta^{t+\delta t}) > g_t(\theta^t)$ unless the gradient vanishes. If the gradient vanishes we have $\theta^{t+\delta t} = \theta^t$ and the quantiles are the same. Otherwise we get $g_t(\theta^{t+\delta t}) > g_t(\theta^t) = q$.

Since $\frac{q-p_-}{p_m} \leqslant \frac{(p_-+p_m)-p_-}{p_m} = 1$, we have $g_t(\theta) \leqslant \Pr_{x\sim P_\theta}(f(x) < m) + \Pr_{x\sim P_\theta}(f(x) = m) = \Pr_{x\sim P_\theta}(f(x) \leqslant m)$.

So $\Pr_{x\sim P_{\theta^{t+\delta t}}}(f(x) \leqslant m) \geqslant g_t(\theta^{t+\delta t}) > q$. This implies, by definition, that the q-quantile value of $P_{\theta^{t+\delta t}}$ is at most m. Moreover, if the objective function has no plateau then $\Pr_{x\sim P_{\theta^{t+\delta t}}}(f(x) = m) = 0$ and so $\Pr_{x\sim P_{\theta^{t+\delta t}}}(f(x) < m) > q$ which implies that the q-quantile of $P_{\theta^{t+\delta t}}$ is stricly less than m.

D.4 Proof of Theorem 10 (Natural Gradient as ML with Infinitesimal Weights)

The proof of Theorem 10 will use Lemma 26. Let W be a function of x, and fix some θ_0 in Θ.

We need some regularity assumptions: we assume that no two points $\theta \in \Theta$ define the same probability distribution and that the map $P_\theta \mapsto \theta$ is continuous. We also assume that the map $\theta \mapsto P_\theta$ is smooth enough, so that $\int \ln P_\theta(x)\, W(x)\, P_{\theta_0}(\mathrm{d}x)$ is a smooth function of θ. (These are restrictions on θ-regularity: this does not mean that W has to be continuous as a function of x.)

The two statements of Theorem 10 using a sum and an integral have similar proofs, so we only include the first. For $\varepsilon > 0$, let θ be the solution of

$$\theta = \arg\max \left\{ (1-\varepsilon) \int \ln P_\theta(x)\, P_{\theta_0}(\mathrm{d}x) + \varepsilon \int \ln P_\theta(x)\, W(x)\, P_{\theta_0}(\mathrm{d}x) \right\}.$$

Then we have

$$\theta = \arg\max \left\{ \int \ln P_\theta(x)\, P_{\theta_0}(\mathrm{d}x) + \varepsilon \int \ln P_\theta(x)\, (W(x)-1)\, P_{\theta_0}(\mathrm{d}x) \right\}$$

$$= \arg\max \left\{ \int \ln P_\theta(x)\, P_{\theta_0}(\mathrm{d}x) - \int \ln P_{\theta_0}(x)\, P_{\theta_0}(\mathrm{d}x) + \varepsilon \int \ln P_\theta(x)\, (W(x)-1)\, P_{\theta_0}(\mathrm{d}x) \right\}$$

(because the added term does not depend on θ)

$$= \arg\max \left\{ -\mathrm{KL}(P_{\theta_0} \,\|\, P_\theta) + \varepsilon \int \ln P_\theta(x)\, (W(x)-1)\, P_{\theta_0}(\mathrm{d}x) \right\}$$

$$= \arg\max \left\{ -\frac{1}{\varepsilon}\mathrm{KL}(P_{\theta_0} \,\|\, P_\theta) + \int \ln P_\theta(x)\, (W(x)-1)\, P_{\theta_0}(\mathrm{d}x) \right\}$$

When $\varepsilon \to 0$, the first term exceeds the second one if $\mathrm{KL}(P_{\theta_0} \,\|\, P_\theta)$ is too large (because W is bounded), and so $\mathrm{KL}(P_{\theta_0} \,\|\, P_\theta)$ tends to 0. So we can assume that θ is close to θ_0.

When $\theta = \theta_0 + \delta\theta$ is close to θ_0, we have

$$\mathrm{KL}(P_{\theta_0} \,\|\, P_\theta) = \frac{1}{2} \sum I_{ij}(\theta_0)\, \delta\theta_i\, \delta\theta_j + O(\delta\theta^3)$$

with $I_{ij}(\theta_0)$ the Fisher matrix at θ_0. (This actually holds both for $\mathrm{KL}(P_{\theta_0} \,\|\, P_\theta)$ and $\mathrm{KL}(P_\theta \,\|\, P_{\theta_0})$.)

Thus, we can apply Lemma 26 using the Fisher metric $\sum I_{ij}(\theta_0)\,\delta\theta_i\,\delta\theta_j$, and working on a small neighborhood of θ_0 in θ-space (which can be identified with $\mathbb{R}^{\dim\Theta}$). The lemma states that the argmax above is attained at

$$\theta = \theta_0 + \varepsilon\widetilde{\nabla}_\theta \int \ln P_\theta(x)\,(W(x)-1)\,P_{\theta_0}(\mathrm{d}x)$$

up to $O(\varepsilon^2)$, with $\widetilde{\nabla}$ the natural gradient.

Finally, the gradient cancels the constant -1 because $\int(\widetilde{\nabla}\ln P_\theta)\,P_{\theta_0} = 0$ at $\theta=\theta_0$. This proves Theorem 10.

D.5 Proof of Theorem 12 (IGO, CEM and IGO-ML)

Let P_θ be a family of probability distributions of the form

$$P_\theta(\mathrm{d}x) = \frac{1}{Z(\theta)}\exp\Big(\sum\theta_i T_i(x)\Big)\;H(\mathrm{d}x)$$

where $T_1,\dots,T_k$ is a finite family of functions on X and $H(\mathrm{d}x)$ is some reference measure on X. We assume that the family of functions $(T_i)_i$ together with the constant function $T_0(x)=1$, are linearly independent. This prevents redundant parametrizations where two values of θ describe the same distribution; this also ensures, by elementary linear algebra, that the Fisher matrix $\mathrm{Cov}(T_i,T_j)$ is invertible.

The IGO update (17) in the parametrization $\bar{T}_i$ is a sum of terms of the form

$$\widetilde{\nabla}_{\bar{T}_i}\ln P(x).$$

So we will compute the natural gradient $\widetilde{\nabla}_{\bar{T}_i}$ in those coordinates.

Let us start with a proposition giving an expression for the Fisher scalar product between two tangent vectors δP and $\delta' P$ of a statistical manifold of exponential distributions. It is one way to express the duality between the coordinates θ_i and $\bar{T}_i$ (compare (Amari and Nagaoka, 2000, (3.30) and Section 3.5)).

Proposition 28 *Let $\delta\theta_i$ and $\delta'\theta_i$ be two small variations of the parameters θ_i. Let $\delta P(x)$ and $\delta' P(x)$ be the resulting variations of the probability distribution P, and $\delta\bar{T}_i$ and $\delta'\bar{T}_i$ the resulting variations of $\bar{T}_i$. Then the scalar product, in Fisher information metric, between the tangent vectors δP and $\delta' P$, is*

$$\langle \delta P, \delta' P\rangle = \sum_i \delta\theta_i\,\delta'\bar{T}_i = \sum_i \delta'\theta_i\,\delta\bar{T}_i.$$

Proof

First, let us prove that $\frac{\partial\bar{T}_i}{\partial\theta_j} = I_{ij}$, the Fisher matrix for θ. Indeed, $\frac{\partial\bar{T}_i}{\partial\theta_j} = \int_x T_i(x)\frac{\partial P(x)}{\partial\theta_j} = \int_x T_i(x)P(x)\frac{\partial\ln P(x)}{\partial\theta_j} = \int_x T_i(x)P(x)(T_j(x)-\bar{T}_j)$ by (19); this is equal to $\mathrm{Cov}(T_i,T_j)$ which is I_{ij} by (20).

Then, by definition of the Fisher metric we have $\langle \delta P, \delta' P \rangle = \sum_{i,j} I_{ij}\, \delta\theta_i\, \delta'\theta_j$ but $\sum_j I_{ij}\delta'\theta_j$ is equal to $\delta'\bar{T}_i$ by the above, because $\delta'\bar{T}_i = \sum_j \frac{\partial \bar{T}_i}{\partial \theta_j}\delta'\theta_j$. Thus we find $\langle \delta P, \delta' P \rangle = \sum_i \delta\theta_i \delta'\bar{T}_i$ as needed. ■

Proposition 29 *Let f be a function on the statistical manifold of an exponential family as above. Then the components of the natural gradient w.r.t. the expectation parameters are given by the vanilla gradient w.r.t. the natural parameters:*

$$\widetilde{\nabla}_{\bar{T}_i} f = \frac{\partial f}{\partial \theta_i} \tag{75}$$

and conversely

$$\widetilde{\nabla}_{\theta_i} f = \frac{\partial f}{\partial \bar{T}_i} \ . \tag{76}$$

(Beware this does *not* mean that the gradient ascent in any of those parametrizations is the vanilla gradient ascent.)

We could not find a reference for this result, though we think it is known (as a consequence of Amari and Nagaoka 2000, Eq. 3.32).

Proof We saw above that $\frac{\partial \bar{T}}{\partial \theta} = I$. Since $\widetilde{\nabla}_\theta f = I^{-1}\frac{\partial f}{\partial \theta}$, this proves (76) by substituting $\frac{\partial f}{\partial \theta} = \left(\frac{\partial \bar{T}}{\partial \theta}\right)^\mathsf{T} \frac{\partial f}{\partial \bar{T}}$.

For the first statement (75) (the one needed for Theorem 12) we have to derive the Fisher matrix for the variables $\bar{T}$. It follows from Proposition 28 that the Fisher matrix in these variables is I^{-1}, by considering the Fisher metric $\sum \delta\theta_i . \delta'\bar{T}_i$ and substituting $I^{-1}\delta\bar{T}$ for $\delta\theta$. Then (75) is proved along the same lines as (76). ■

Back to the proof of Theorem 12. We can now compute the desired terms:

$$\widetilde{\nabla}_{\bar{T}_i} \ln P(x) = \frac{\partial \ln P(x)}{\partial \theta_i} = T_i(x) - \bar{T}_i$$

by (19). This proves the first statement (34) in Theorem 12 about the form of the IGO update in these parameters.

The other statements follow easily from this together with the additional fact (33) that, for any set of (positive or negative) weights a_i with $\sum a_i = 1$, the value $T^* = \sum_i a_i T(x_i)$ maximizes $\sum_i a_i \ln P(x_i)$.

D.6 Proof of Proposition 20 and Corollary 21 (Speed of IGO)

Lemma 30 *Let X be a centered L^2 random variable with values in $\mathbb{R}^d$ and let A be a real-valued L^2 random variable. Then*

$$\|\mathbb{E}(AX)\| \leqslant \sqrt{\lambda \operatorname{Var} A}$$

where λ is the largest eigenvalue of the covariance matrix of X expressed in an orthonormal basis.

Proof Let v be any vector in $\mathbb{R}^d$; its norm satisfies

$$\|v\| = \sup_{w,\, \|w\| \leqslant 1} (v \cdot w)$$

and in particular

$$\begin{aligned}
\|\mathbb{E}(AX)\| &= \sup_{w,\, \|w\| \leqslant 1} (w \cdot \mathbb{E}(AX)) \\
&= \sup_{w,\, \|w\| \leqslant 1} \mathbb{E}(A\,(w \cdot X)) \\
&= \sup_{w,\, \|w\| \leqslant 1} \mathbb{E}((A - \mathbb{E}A)\,(w \cdot X)) \quad \text{since } (w \cdot X) \text{ is centered} \\
&\leqslant \sup_{w,\, \|w\| \leqslant 1} \sqrt{\operatorname{Var} A}\, \sqrt{\mathbb{E}((w \cdot X)^2)}
\end{aligned}$$

by the Cauchy–Schwarz inequality.

Now, in an orthonormal basis we have

$$(w \cdot X) = \sum_i w_i X_i$$

so that

$$\begin{aligned}
\mathbb{E}((w \cdot X)^2) &= \mathbb{E}\left((\textstyle\sum_i w_i X_i)(\sum_j w_j X_j) \right) \\
&= \textstyle\sum_i \sum_j \mathbb{E}(w_i X_i w_j X_j) \\
&= \textstyle\sum_i \sum_j w_i w_j \mathbb{E}(X_i X_j) \\
&= \textstyle\sum_i \sum_j w_i w_j C_{ij}
\end{aligned}$$

with C_{ij} the covariance matrix of X. The latter expression is the scalar product $(w \cdot Cw)$. Since C is a symmetric positive-semidefinite matrix, $(w \cdot Cw)$ is at most $\lambda \|w\|^2$ with λ the largest eigenvalue of C. ■

For the IGO flow we have $\frac{\mathrm{d}\theta^t}{\mathrm{d}t} = \mathbb{E}_{x \sim P_\theta} W_\theta^f(x) \tilde{\nabla}_\theta \ln P_\theta(x)$.

So applying the lemma, we get that the norm of $\frac{\mathrm{d}\theta}{\mathrm{d}t}$ is at most $\sqrt{\lambda \operatorname{Var}_{x \sim P_\theta} W_\theta^f(x)}$ where λ is the largest eivengalue of the covariance matrix of $\tilde{\nabla}_\theta \ln P_\theta(x)$ (expressed in a coordinate system where the Fisher matrix at the current point θ is the identity).

By construction of the quantiles, we have $\operatorname{Var}_{x \sim P_\theta} W_\theta^f(x) \leqslant \operatorname{Var}_{[0,1]} w$ (with equality unless there are ties). Indeed, for a given x, let $\mathcal{U}$ be a uniform random variable in $[0,1]$ independent from x and define the random variable $Q = q^<(x) + (q^\leqslant(x) - q^<(x))\mathcal{U}$. Then Q is uniformly distributed between the upper and lower quantiles $q^\leqslant(x)$ and $q^<(x)$ and thus we can rewrite $W_\theta^f(x)$ as $\mathbb{E}(w(Q)|x)$. By the Jensen inequality we have $\operatorname{Var} W_\theta^f(x) = \operatorname{Var} \mathbb{E}(w(Q)|x) \leqslant \operatorname{Var} w(Q)$. In addition when x is taken under P_θ, Q is uniformly distributed in $[0,1]$ and thus $\operatorname{Var} w(Q) = \operatorname{Var}_{[0,1]} w$, i.e., $\operatorname{Var}_{x \sim P_\theta} W_\theta^f(x) \leqslant \operatorname{Var}_{[0,1]} w$.

Besides, consider the tangent space in Θ-space at point θ^t, and let us choose an orthonormal basis in this tangent space for the Fisher metric. Then, in this basis we have $\tilde{\nabla}_i \ln P_\theta(x) =$

$\partial_i \ln P_\theta(x)$. So the covariance matrix of $\tilde{\nabla} \ln P_\theta(x)$ is $\mathbb{E}_{x\sim P_\theta}(\partial_i \ln P_\theta(x) \partial_j \ln P_\theta(x))$, which is equal to the Fisher matrix by definition. So this covariance matrix is the identity, whose largest eigenvalue is 1. This proves Proposition 20.

For Corollary 21, by the relationship (2) between Fisher matrix and Kullback–Leibler divergence, if v is the speed of the IGO flow then the Kullback–Leibler divergence between P_{θ^t} and $P_{\theta^{t+\delta t}}$ (where $P_{\theta^{t+\delta t}}$ is the trajectory of the IGO flow after a time δt) is equal to the square norm of $\delta t.v$ in Fisher metric up to an $O(\|\delta t.v\|^3)$ term. Now if $P_{\theta^{t+\delta t}}$ is obtained by a finite-population IGO *algorithm*, by Theorem 6 the actual v from the IGO algorithm differs from the speed of the IGO flow by an $o(1)_{N\to\infty}$ term. Collecting terms, we find the expression in Corollary 21.

D.7 Proof of Proposition 22 (Noisy IGO)

On the one hand, let P_θ be a family of distributions on X. The IGO algorithm (16) applied to a random function $f(x) = \tilde{f}(x, \omega)$ where ω is a random variable uniformly distributed in $[0, 1]$ reads

$$\theta^{t+\delta t} = \theta^t + \delta t \sum_{i=1}^{N} \hat{w}_i \widetilde{\nabla}_\theta \ln P_\theta(x_i) \tag{77}$$

where $x_i \sim P_\theta$ and $\hat{w}_i$ is according to (14) where ranking is applied to the values $\tilde{f}(x_i, \omega_i)$, with ω_i uniform variables in $[0, 1]$ independent from x_i and from each other.

On the other hand, for the IGO algorithm using $P_\theta \otimes U_{[0,1]}$ and applied to the deterministic function $\tilde{f}$, $\hat{w}_i$ is computed using the ranking according to the $\tilde{f}$ values of the sampled points $\tilde{x}_i = (x_i, \omega_i)$, and thus coincides with the one in (77).

Besides,

$$\partial_\theta \ln P_{\theta\otimes U_{[0,1]}}(\tilde{x}_i) = \partial_\theta \ln P_\theta(x_i) + \underbrace{\partial_\theta \ln U_{[0,1]}(\omega_i)}_{=0}$$

and thus, both the vanilla gradients and the Fisher matrix I (given by the tensor square of the vanilla gradients) coincide. This proves that the IGO algorithm update on space $X \times [0, 1]$, using the family of distributions $\tilde{P}_\theta = P_\theta \otimes U_{[0,1]}$, applied to the deterministic function $\tilde{f}$, coincides with (77).

References

D.H. Ackley, G.E. Hinton, and T.J. Sejnowski. A learning algorithm for Boltzmann machines. *Cognitive Science*, 9(1):147–169, 1985.

R.P. Agarwal and D. O'Regan. *An Introduction to Ordinary Differential Equations*. Springer, 2008.

Y. Akimoto and Y. Ollivier. Objective improvement in information-geometric optimization. In F. Neumann and K. DeJong, editors, *Foundations of Genetic Algorithms XII (FOGA 2013)*, Adelaide, Australia, 2013.

Y. Akimoto, Y. Nagata, I. Ono, and S. Kobayashi. Bidirectional relation between CMA evolution strategies and natural evolution strategies. In *Proceedings of Parallel Problem*

Solving from Nature - PPSN XI, volume 6238 of *Lecture Notes in Computer Science*, pages 154–163. Springer, 2010.

Y. Akimoto, A. Auger, and N. Hansen. Convergence of the continuous time trajectories of isotropic evolution strategies on monotonic $\mathcal{C}^2$-composite functions. In C.A. Coello Coello, V. Cutello, K. Deb, S. Forrest, G. Nicosia, and M. Pavone, editors, *PPSN (1)*, volume 7491 of *Lecture Notes in Computer Science*, pages 42–51. Springer, 2012. ISBN 978-3-642-32936-4.

S.-I. Amari. Natural gradient works efficiently in learning. *Neural Comput.*, 10:251–276, February 1998. ISSN 0899-7667. doi: 10.1162/089976698300017746. URL `http://portal.acm.org/citation.cfm?id=287476.287477`.

S.-I. Amari and H. Nagaoka. *Methods of information geometry*, volume 191 of *Translations of Mathematical Monographs*. American Mathematical Society, Providence, RI, 2000. ISBN 0-8218-0531-2. Translated from the 1993 Japanese original by Daishi Harada.

D.V. Arnold. Weighted multirecombination evolution strategies. *Theoretical computer science*, 361(1):18–37, 2006.

S. Baluja. Population based incremental learning: A method for integrating genetic search based function optimization and competitve learning. Technical Report CMU-CS-94-163, Carnegie Mellon Report, 1994.

S. Baluja and R. Caruana. Removing the genetics from the standard genetic algorithm. In *Proceedings of ICML'95*, pages 38–46, 1995.

Y. Bengio, P. Lamblin, V. Popovici, and H. Larochelle. Greedy layer-wise training of deep networks. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 153–160. MIT Press, Cambridge, MA, 2007.

Y. Bengio, A.C. Courville, and P. Vincent. Unsupervised feature learning and deep learning: A review and new perspectives. *ArXiv preprints*, arXiv:1206.5538, 2012.

J. Bensadon. Black-box optimization using geodesics in statistical manifolds. *Entropy*, 17 (1):304–345, 2015.

A. Berny. Selection and reinforcement learning for combinatorial optimization. In M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J. Merelo, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature PPSN VI*, volume 1917 of *Lecture Notes in Computer Science*, pages 601–610. Springer Berlin Heidelberg, 2000a.

A. Berny. An adaptive scheme for real function optimization acting as a selection operator. In *Combinations of Evolutionary Computation and Neural Networks, 2000 IEEE Symposium on*, pages 140 –149, 2000b. doi: 10.1109/ECNN.2000.886229.

A. Berny. Boltzmann machine for population-based incremental learning. In *ECAI*, pages 198–202, 2002.

H.-G. Beyer. *The Theory of Evolution Strategies*. Natural Computing Series. Springer-Verlag, 2001.

H.-G. Beyer and H.-P. Schwefel. Evolution strategies—a comprehensive introduction. *Natural computing*, 1(1):3–52, 2002. ISSN 1567-7818.

P. Billingsley. *Probability and measure*. Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons Inc., New York, third edition, 1995. ISBN 0-471-00710-2. A Wiley-Interscience Publication.

J. Branke, C. Lode, and J.L. Shapiro. Addressing sampling errors and diversity loss in UMDA. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, pages 508–515. ACM, 2007.

T.M. Cover and J.A. Thomas. *Elements of information theory*. Wiley-Interscience [John Wiley & Sons], Hoboken, NJ, second edition, 2006. ISBN 978-0-471-24195-9; 0-471-24195-4.

S. Das, S. Maity, B.-Y. Qu, and P.N. Suganthan. Real-parameter evolutionary multimodal optimization - a survey of the state-of-the-art. *Swarm and Evolutionary Computation*, 1 (2):71–88, 2011.

P.-T. de Boer, D.P. Kroese, S. Mannor, and R.Y. Rubinstein. A tutorial on the cross-entropy method. *Annals OR*, 134(1):19–67, 2005.

G. Desjardins, A. Courville, Y. Bengio, P. Vincent, and O. Dellaleau. Parallel tempering for training of restricted Boltzmann machines. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010.

P. Deuflhard. *Newton methods for nonlinear problems: affine invariance and adaptive algorithms*, volume 35. Springer, 2011.

E.D. Dolan and J.J. Moré. Benchmarking optimization software with performance profiles. *Mathematical programming*, 91(2):201–213, 2002.

S. Bhatnagar E. Zhou. Gradient-based adaptive stochastic search for simulation optimization over continuous space. *ArXiv preprints*, arXiv:1608.00663, 2016.

M. Gallagher and M. Frean. Population-based continuous optimization, probabilistic modelling and mean shift. *Evol. Comput.*, 13(1):29–42, January 2005. ISSN 1063-6560. doi: 10.1162/1063656053583478. URL `http://dx.doi.org/10.1162/1063656053583478`.

Z. Ghahramani. Unsupervised learning. In O. Bousquet, U. von Luxburg, and G. Rätsch, editors, *Advanced Lectures on Machine Learning*, volume 3176 of *Lecture Notes in Computer Science*, pages 72–112. Springer Berlin / Heidelberg, 2004.

T. Glasmachers, T. Schaul, Y. Sun, D. Wierstra, and J. Schmidhuber. Exponential natural evolution strategies. In *Proceedings of the 12th annual conference on Genetic and evolutionary computation GECCO'10*, pages 393–400. ACM, 2010.

J. Grahl, S. Minner, and F. Rothlauf. Behaviour of umda c with truncation selection on monotonous functions. In *Evolutionary Computation, 2005. The 2005 IEEE Congress on*, volume 3, pages 2553–2559. IEEE, 2005.

N. Hansen. An analysis of mutative σ-self-adaptation on linear fitness functions. *Evolutionary Computation*, 14(3):255–275, 2006a.

N. Hansen. The CMA evolution strategy: a comparing review. In J.A. Lozano, P. Larranaga, I. Inza, and E. Bengoetxea, editors, *Towards a new evolutionary computation. Advances on estimation of distribution algorithms*, pages 75–102. Springer, 2006b.

N. Hansen. Benchmarking a BI-population CMA-ES on the BBOB-2009 function testbed. In *Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers*, GECCO '09, pages 2389–2396, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-505-5. doi: http://doi.acm.org/10.1145/1570256.1570333. URL `http://doi.acm.org/10.1145/1570256.1570333`.

N. Hansen and A. Auger. Principled design of continuous stochastic search: From theory to practice. In Y. Borenstein and A. Moraglio, editors, *Theory and Principled Methods for the Design of Metaheuristics*, Natural Computing Series, pages 145–180. Springer Berlin Heidelberg, 2014. ISBN 978-3-642-33205-0. doi: 10.1007/978-3-642-33206-7_8. URL `http://dx.doi.org/10.1007/978-3-642-33206-7_8`.

N. Hansen and S. Kern. Evaluating the CMA evolution strategy on multimodal test functions. In X. Yao et al., editors, *Parallel Problem Solving from Nature PPSN VIII*, volume 3242 of *LNCS*, pages 282–291. Springer, 2004.

N. Hansen and A. Ostermeier. Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation. In *ICEC96*, pages 312–317. IEEE Press, 1996.

N. Hansen and A. Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 9(2):159–195, 2001.

N. Hansen, S.D. Müller, and P. Koumoutsakos. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). *Evolutionary Computation*, 11(1):1–18, 2003. ISSN 1063-6560.

G.R. Harik, F.G. Lobo, and D.E. Goldberg. The compact genetic algorithm. *Evolutionary Computation, IEEE Transactions on*, 3(4):287–297, 1999.

G.E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14:1771–1800, 2002.

G.E. Hinton, S. Osindero, and Y.-W. Teh. A fast learning algorithm for deep belief nets. *Neural Conputation*, 18:1527–1554, 2006.

R. Hooke and T.A. Jeeves. "Direct search" solution of numerical and statistical problems. *Journal of the ACM*, 8:212–229, 1961.

G.A. Jastrebski and D.V. Arnold. Improving evolution strategies through active covariance matrix adaptation. In *Evolutionary Computation, 2006. CEC 2006. IEEE Congress on*, pages 2814–2821. IEEE, 2006. ISBN 0780394879.

M. Jebalia and A. Auger. Log-linear convergence of the scale-invariant $(\mu/\mu_w, \lambda)$-ES and optimal μ for intermediate recombination for large population sizes. In R. Schaefer et al., editor, *Parallel Problem Solving from Nature (PPSN XI)*, volume 6239, pages 52–61. Springer, 2010. URL `http://hal.inria.fr/docs/00/49/44/78/PDF/ppsn2010JebaliaAuger.pdf`.

H. Jeffreys. An invariant form for the prior probability in estimation problems. *Proc. Roy. Soc. London. Ser. A.*, 186:453–461, 1946. ISSN 0962-8444.

H.K. Khalil. *Nonlinear Systems*. Nonlinear Systems. Prentice-Hall, Inc., second edition, 1996.

P.E. Kloeden and E. Platen. *Numerical solution of stochastic differential equations*, volume 23 of *Applications of Mathematics (New York)*. Springer-Verlag, Berlin, 1992. ISBN 3-540-54062-8.

S. Kullback. *Information theory and statistics*. Dover Publications Inc., Mineola, NY, 1997. ISBN 0-486-69684-7. Reprint of the second (1968) edition.

P. Larranaga and J.A. Lozano. *Estimation of distribution algorithms: A new tool for evolutionary computation*. Springer Netherlands, 2002. ISBN 0792374665.

N. Le Roux, P.-A. Manzagol, and Y. Bengio. Topmoumoute online natural gradient algorithm. In *NIPS*, 2007.

L. Malagò, M. Matteucci, and B. Dal Seno. An information geometry perspective on estimation of distribution algorithms: boundary analysis. In *GECCO (Companion)*, pages 2081–2088, 2008.

L. Malagò, M. Matteucci, and G. Pistone. Towards the geometry of estimation of distribution algorithms based on the exponential family. In H.-G. Beyer and W.B. Langdon, editors, *FOGA, Proceedings*, pages 230–242. ACM, 2011. ISBN 978-1-4503-0633-1.

G. Montavon and K.-R. Müller. Deep boltzmann machines and the centering trick. In G. Montavon, G.B. Orr, and K.-R. Müller, editors, *Neural Networks: Tricks of the Trade (2nd ed.)*, volume 7700 of *Lecture Notes in Computer Science*, pages 621–637. Springer, 2012. ISBN 978-3-642-35288-1.

J.J. Moré, B.S. Garbow, and K.E. Hillstrom. Testing unconstrained optimization software. *ACM Transactions on Mathematical Software (TOMS)*, 7(1):17–41, 1981.

J.A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, pages 308–313, 1965.

Y. Ollivier, L. Arnold, A. Auger, and N. Hansen. Information-geometric optimization: A unifying picture via invariance principles. *ArXiv preprints*, arXiv:1106.3708v2, 2011.

M. Pelikan, D.E. Goldberg, and F.G. Lobo. A survey of optimization by building and using probabilistic models. *Computational optimization and applications*, 21(1):5–20, 2002. ISSN 0926-6003.

P. Pošík. Preventing premature convergence in a simple eda via global step size setting. In *Parallel Problem Solving from Nature–PPSN X*, pages 549–558. Springer, 2008.

C.R. Rao. Information and the accuracy attainable in the estimation of statistical parameters. *Bull. Calcutta Math. Soc.*, 37:81–91, 1945. ISSN 0008-0659.

I. Rechenberg. *Evolutionstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Frommann-Holzboog Verlag, Stuttgart, 1973.

I. Rechenberg. *Evolutionsstrategie '94*. Frommann-Holzboog Verlag, 1994.

R. Ros and N. Hansen. A simple modification in CMA-ES achieving linear time and space complexity. In G. Rudolph, T. Jansen, S. Lucas, C. Polini, and N. Beume, editors, *Proceedings of Parallel Problem Solving from Nature (PPSN X)*, volume 5199 of *Lecture Notes in Computer Science*, pages 296–305. Springer, 2008.

R.Y. Rubinstein. The cross-entropy method for combinatorial and continuous optimization. *Methodology and Computing in Applied Probability*, 1:127–190, 1999. ISSN 1387-5841. URL `http://dx.doi.org/10.1023/A:1010091220143`. 10.1023/A:1010091220143.

R.Y. Rubinstein and D.P. Kroese. *The cross-entropy method: a unified approach to combinatorial optimization, Monte-Carlo simulation, and machine learning*. Springer-Verlag New York Inc, 2004. ISBN 038721240X.

R. Salakhutdinov. Learning in Markov random fields using tempered transitions. In Y. Bengio, D. Schuurmans, J. Lafferty, C.K.I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 1598–1606. MIT Press, 2009.

R. Salakhutdinov and I. Murray. On the quantitative analysis of deep belief networks. In *Proceedings of the 25th international conference on Machine learning*, ICML '08, pages 872–879, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-205-4. URL `http://doi.acm.org/10.1145/1390156.1390266`.

B. Sareni and L. Krähenbühl. Fitness sharing and niching methods revisited. *IEEE Trans. Evolutionary Computation*, 2(3):97–106, 1998.

T. Schaul, T. Glasmachers, and J. Schmidhuber. High dimensions and heavy tails for natural evolution strategies. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, GECCO '11, pages 845–852, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0557-0. doi: 10.1145/2001576.2001692. URL `http://doi.acm.org/10.1145/2001576.2001692`.

R.L. Schilling. *Measures, integrals and martingales*. Cambridge University Press, New York, 2005. ISBN 978-0-521-61525-9; 0-521-61525-9. doi: 10.1017/CBO9780511810886. URL `http://dx.doi.org/10.1017/CBO9780511810886`.

L. Schwartz. *Analyse. II*, volume 43 of *Collection Enseignement des Sciences [Collection: The Teaching of Science]*. Hermann, Paris, 1992. Calcul différentiel et équations différentielles, With the collaboration of K. Zizi.

H.-P. Schwefel. *Evolution and Optimum Seeking*. Sixth-generation computer technology series. John Wiley & Sons, Inc. New York, NY, USA, 1995. ISBN 0471571482.

F. Silva and L. Almeida. Acceleration techniques for the backpropagation algorithm. *Neural Networks*, pages 110–119, 1990.

P. Smolensky. Information processing in dynamical systems: foundations of harmony theory. In D. Rumelhart and J. McClelland, editors, *Parallel Distributed Processing*, volume 1, chapter 6, pages 194–281. MIT Press, Cambridge, MA, USA, 1986. ISBN 0-262-68053-X.

Y. Sun, D. Wierstra, T. Schaul, and J. Schmidhuber. Efficient natural evolution strategies. In *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, GECCO '09, pages 539–546, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-325-9. doi: http://doi.acm.org/10.1145/1569901.1569976. URL `http://doi.acm.org/10.1145/1569901.1569976`.

T. Suttorp, N. Hansen, and C. Igel. Efficient covariance matrix update for variable metric evolution strategies. *Machine Learning*, 75(2):167–197, 2009.

H. Thorisson. *Coupling, Stationarity, and Regeneration*. Springer, 2000.

V. Torczon. On the convergence of pattern search algorithms. *SIAM Journal on optimization*, 7(1):1–25, 1997.

M. Toussaint. Notes on information geometry and evolutionary processes. *ArXiv preprints*, arXiv:nlin/0408040, 2004.

M. Wagner, A. Auger, and M. Schoenauer. EEDA : A new robust estimation of distribution algorithms. Research Report RR-5190, INRIA, 2004. URL `http://hal.inria.fr/inria-00070802/en/`.

D. Whitley. The genitor algorithm and selection pressure: Why rank-based allocation of reproductive trials is best. In *Proceedings of the third international conference on Genetic algorithms*, pages 116–121, 1989.

D. Wierstra, T. Schaul, J. Peters, and J. Schmidhuber. Natural evolution strategies. In *IEEE Congress on Evolutionary Computation*, pages 3381–3387, 2008.

D. Wierstra, T. Schaul, T. Glasmachers, Y. Sun, J. Peters, and J. Schmidhuber. Natural evolution strategies. *Journal of Machine Learning Research*, 15:949–980, 2014. URL `http://jmlr.org/papers/v15/wierstra14a.html`.

E. Zhou and J. Hu. Gradient-based adaptive stochastic search for non-differentiable optimization. *IEEE Transactions on Automatic Control*, 59(7):1818–1832, July 2014. ISSN 0018-9286. doi: 10.1109/TAC.2014.2310052.