On the locality of the natural gradient for learning in deep Bayesian networks

Nihat Ay^{1,2,3}

Received: 29 June 2020 / Revised: 29 October 2020 / Accepted: 31 October 2020
© The Author(s) 2020

Abstract

We study the natural gradient method for learning in deep Bayesian networks, including neural networks. There are two natural geometries associated with such learning systems consisting of visible and hidden units. One geometry is related to the full system, the other one to the visible sub-system. These two geometries imply different natural gradients. In a first step, we demonstrate a great simplification of the natural gradient with respect to the first geometry, due to locality properties of the Fisher information matrix. This simplification does not directly translate to a corresponding simplification with respect to the second geometry. We develop the theory for studying the relation between the two versions of the natural gradient and outline a method for the simplification of the natural gradient with respect to the second geometry based on the first one. This method suggests to incorporate a recognition model as an auxiliary model for the efficient application of the natural gradient method in deep networks.

Keywords Natural gradient · Fisher–Rao metric · Deep learning · Helmholtz machines · Wake–sleep algorithm

Contents

1	Introduction	
2	Locality of deep learning in Bayesian and neural networks	
3	Gradients on full versus coarse grained models	
4	Conclusions: A natural gradient perspective of the wake–sleep algorithm	
	Appendix	
	References	

✉ Nihat Ay
nay@mis.mpg.de

¹ Max Planck Institute for Mathematics in the Sciences, 04103 Leipzig, Germany

² Leipzig University, 04109 Leipzig, Germany

³ Santa Fe Institute, Santa Fe, NM 87501, USA

1 Introduction

1.1 The natural gradient method

Within the last decade, deep artificial neural networks have led to unexpected successes of machine learning in a large number of applications [15]. One important direction of research within the field of deep learning is based on the natural gradient method from information geometry [3,4,8]. It has been proposed by Amari [2] as a gradient method that is invariant with respect to coordinate transformations. This method turns out to be extremely efficient within various fields of artificial intelligence and machine learning, including neural networks [2], reinforcement learning [7,19], and robotics [27]. It is known to overcome several problems of traditional gradient methods. Most importantly, the natural gradient method avoids the so-called plateau problem, and it is less sensitive to singularities (for a detailed discussion, see Section 12.2 of the book [3]; the subject of singularities is treated in [32]). On the other hand, there are significant challenges and limitations concerning the applicability of the natural gradient method [22]. Without further assumptions this method becomes intractable in the context of deep neural networks with a large number of parameters. Various approximate methods have been proposed and studied as alternatives to the original method [20,23,26]. In this article, we highlight information-geometric structures of deep Bayesian and, in particular, neural networks that allow for a simplification of the natural gradient. The guiding scheme of this simplification is *locality* with respect to the underlying network structure [5]. There are several aspects of learning that can be addressed from this perspective:

1. *Objective function:* Typically, learning is based on the optimisation of some global objective function related to the overall performance of the network, which, in the most general context, is evaluated in some behaviour space. On the other hand, if we assume that individual units access information only from their local neighbourhood, then we are naturally led to the following problem. Is it possible to decompose the objective function into individual local objective functions that can be evaluated by the corresponding units?
2. *Learning I:* Assuming that learning is based on the gradient of a global objective function, does the above-mentioned decomposition into local functions imply a corresponding locality of the gradient with respect to the parametrisation? In that case, the individual units would adjust their parameter values, such as the synaptic connection strengths in the case of neural networks, based on local information. This is a typical implicit assumption within the field of neural networks, most prominently realised in terms of Hebbian learning [16], which implies that a connection between two neurons is modified based on their joint activity.
3. *Learning II:* When computing the *natural* gradient of an objective function, we have to evaluate (the inverse of) the Fisher information matrix. Even if locality of learning is guaranteed for the Euclidean gradient, this matrix might reintroduce non-locality. We will analyse to what extent the natural gradient preserves locality? One instance of this property corresponds to a block diagonal structure of the Fisher information matrix which simplifies its inversion [5,28].

We are now going to introduce the required formalism and outline the problem setting in more detail.

1.2 Preliminaries and the main problem

We first introduce the notation used in this article. Let S be a non-empty finite set. We denote the canonical basis of the vector space $\mathbb{R}^S$ by $e_s, s \in S$. The corresponding dual vectors $\delta^s \in (\mathbb{R}^S)^*$, $s \in S$, defined by

$$\delta^s(e_{s'}) := \begin{cases} 1, & \text{if } s = s' \\ 0, & \text{otherwise} \end{cases},$$

can be identified with the Dirac measures on S . Each linear form $l \in (\mathbb{R}^S)^*$ can be written as $\sum_s l(s) \delta^s$, where $l(s) := l(e_s)$. We denote the open simplex of strictly positive probability vectors on S by

$$\mathcal{P}(S) := \left\{ p = \sum_s p(s) \delta^s : p(s) > 0 \text{ for all } s, \text{ and } \sum_s p(s) = 1 \right\}.$$

For each point $p \in \mathcal{P}(S)$, the tangent space in p can be naturally identified with

$$\mathcal{T}(S) := \left\{ A = \sum_s A(s) \delta^s : \sum_s A(s) = 0 \right\}.$$

The *Fisher–Rao metric* on $\mathcal{P}(S)$ in $p = \sum_s p(s) \delta^s$ is defined by

$$\langle A, B \rangle_p := \sum_s \frac{1}{p(s)} A(s) B(s), \quad A, B \in \mathcal{T}(S). \quad (1)$$

Let us now consider a model $\mathcal{M} \subseteq \mathcal{P}(S)$ which we assume to be a d -dimensional smooth manifold with local coordinates $\xi = (\xi_1, \dots, \xi_d) \mapsto p_\xi$, where ξ is from an open domain $\mathcal{E}$ in $\mathbb{R}^d$. Below, we will treat more general models, but starting with manifolds allows us to outline more clearly the challenges we face in the context of the natural gradient method. With $p(s; \xi) := p_\xi(s)$, we define the vectors $\partial_i(\xi) := \frac{\partial}{\partial \xi_i} p_\xi$, $i = 1, \dots, d$, which span the tangent space $T_\xi \mathcal{M}$. (Throughout this article, we use the subscript ξ , as in $T_\xi \mathcal{M}$, to denote the point p_ξ whenever this simplifies the notation.) From (1) we then obtain the Fisher information matrix $G(\xi) = (g_{ij}(\xi))_{ij}$, defined by

$$g_{ij}(\xi) := \langle \partial_i(\xi), \partial_j(\xi) \rangle_\xi = \sum_s p(s; \xi) \frac{\partial \ln p(s; \cdot)}{\partial \xi_i}(\xi) \frac{\partial \ln p(s; \cdot)}{\partial \xi_j}(\xi). \quad (2)$$

Given a smooth function $\mathcal{L} : \mathcal{M} \rightarrow \mathbb{R}$, its gradient $\text{grad}_\xi^\mathcal{M} \mathcal{L} \in T_\xi \mathcal{M}$ in p_ξ is the direction of steepest ascent. It has the following usual representation in the local coordinates ξ :

$$\text{grad}_{\xi}^{\mathcal{M}} \mathcal{L} = G^{-1}(\xi) \nabla_{\xi} \mathcal{L}. \quad (3)$$

As we can see, the gradient depends on the Fisher–Rao metric in p_{ξ} . It is this dependence that makes it *natural* and the reason for calling it the *natural gradient*. Let us clarify how to read the equation (3). The LHS is a vector in $T_{\xi} \mathcal{M}$, whereas the RHS is a vector in $\mathbb{R}^d$, which appears somewhat inconsistent. The way to read this is the following: As a vector in the tangent space in p_{ξ} , the gradient has a representation $\sum_{i=1}^d x^i \partial_i(\xi)$. The coordinates x^i are then given by the RHS of (3) (see also “Moore–Penrose inverse and gradients” of the Appendix).

In this article, the set S will typically be a Cartesian product of state sets of units, for instance binary neurons. More precisely, we consider a non-empty and finite set N of units consisting of n visible units V and m hidden units H , that is $N = V \uplus H$. The state sets of the units are denoted by X_i , $i \in N$, and assumed to be non-empty and finite. For any subset $M \subseteq N$, we have the corresponding configuration or state set $X_M := \times_{i \in M} X_i$, the set $\mathcal{P}_M := \mathcal{P}(X_M)$ of strictly positive probability vectors on X_M , and the tangent space $\mathcal{T}_M := \mathcal{T}(X_M)$. Consider now the restriction $X_M : X_N \rightarrow X_M$, $x = (x_M, x_{N \setminus M}) \mapsto x_M$, and its push-forward map

$$\pi_M : \mathcal{P}_N \rightarrow \mathcal{P}_M, \quad p \mapsto \pi_M(p) := \sum_{x_M \in X_M} p(x_M) \delta^{x_M},$$

where $p(x_M) := \sum_{x_{N \setminus M} \in X_{N \setminus M}} p(x_M, x_{N \setminus M})$. This is simply the marginalisation map where $\pi_M(p)$ is the M -marginal of p . We will primarily deal with the case where the subset M is given by the visible units V .

Given a model $\mathcal{M}$ in $\mathcal{P}_{V,H} := \mathcal{P}_N$, we consider the *marginal* or *projected model* $\mathcal{M}_V := \pi_V(\mathcal{M})$ in $\mathcal{P}_V$ which will play a major role in this article. With a parametrisation $\xi \mapsto p_{\xi}$ of $\mathcal{M}$, we also have a parametrisation of $\mathcal{M}_V$, simply by mapping ξ to the point $\pi_V(p_{\xi}) \in \mathcal{M}_V$. For ξ , we then have the tangent vectors $\partial_i(\xi) = \frac{\partial}{\partial \xi_i} p_{\xi}$, $i = 1, \dots, d$, in $T_{\xi} \mathcal{M}$ and the tangent vectors $\bar{\partial}_i(\xi) := \frac{\partial}{\partial \xi_i} \pi_V(p_{\xi})$, $i = 1, \dots, d$, in $T_{\xi} \mathcal{M}_V$. The definition (2) of the Fisher information matrix in ξ can now be applied to both models, $\mathcal{M}$ and $\mathcal{M}_V$. In order to distinguish them from each other, we write $g_{ij}(\xi) = \langle \partial_i(\xi), \partial_j(\xi) \rangle_{\xi}$ for the components of the Fisher information matrix $G(\xi)$ in $p_{\xi} \in \mathcal{M}$, and correspondingly $\bar{g}_{ij}(\xi) := \langle \bar{\partial}_i(\xi), \bar{\partial}_j(\xi) \rangle_{\xi}$ for the components of the Fisher information matrix $\bar{G}(\xi)$ in $\pi_V(p_{\xi}) \in \mathcal{M}_V$.

Notice that we face a number of difficulties already at this point.

1. Even if we choose $\mathcal{M}$ to be a smooth manifold, its projection $\mathcal{M}_V$ is typically a much more complicated geometric object with various kinds of singularities (to be formally defined in Section 3.1). Therefore, we will allow for more general models without assuming $\mathcal{M}$ to be a smooth manifold in the first place. However, we will restrict attention to non-singular points only.
2. In addition to having a general model $\mathcal{M}$, we also drop the assumption that the parametrisation $\xi = (\xi_1, \dots, \xi_d) \mapsto p_{\xi}$ is given by a (diffeomorphic) coordinate system. This has consequences on the definition of the Fisher–Rao metric in a non-singular point p_{ξ} :

- (a) In order to interpret the Fisher–Rao metric as a Riemannian metric, the derivatives $\partial_i(\xi) = \frac{\partial}{\partial \xi_i} p_\xi, i = 1, \dots, d$, have to span the whole tangent space $T_\xi \mathcal{M}$ in p_ξ . (This is often implicitly assumed but not explicitly stated.) Otherwise, the Fisher–Rao metric defined by (2) will not be positive definite. We will refer to a parametrisation that satisfies this condition as a *proper parametrisation*. Note that for a proper parametrisation $\xi \mapsto p_\xi$ of $\mathcal{M}$, the composition $\xi \mapsto \pi_V(p_\xi)$ is not necessarily a proper parametrisation of $\mathcal{M}_V$.
- (b) Another consequence of not having a coordinate system as a parametrisation is the fact that the number d of parameters may exceed the dimension of the model. Even if we assume $\mathcal{M}$ to be a smooth manifold and its parametrisation given by a coordinate system, such that d equals the dimension of $\mathcal{M}$, the corresponding projected model $\mathcal{M}_V$ can have a much lower dimension. In that case, we say that the model is overparametrised. Such models play an important role within the field of deep learning. The Fisher–Rao metric for such models is well defined in non-singular points. However the Fisher information matrix (2) will be degenerate so that the representation of a gradient in terms of the parameters is not unique anymore. Below, we will come back to this problem.

We use the natural gradient method in order to minimise (or maximise) a function $\mathcal{L} : \mathcal{M}_V \rightarrow \mathbb{R}$ which is usually obtained as a restriction of a smooth function defined on $\mathcal{P}_V$. Therefore, it is natural to use the Fisher–Rao metric on $\mathcal{M}_V$ which is inherited from $\mathcal{P}_V$. Assuming that all required quantities are well defined, we can express this natural gradient in terms of the parametrisation as

$$\text{grad}_\xi^{\mathcal{M}_V} \mathcal{L} = \overline{G}^+(\xi) \nabla_\xi \mathcal{L}, \quad (4)$$

where $\overline{G}^+(\xi)$ is the Moore–Penrose inverse of the Fisher information matrix $\overline{G}(\xi)$ (for details on the Moore–Penrose inverse, see “Moore–Penrose inverse and gradients” of the Appendix). If the parametrisation is given by a coordinate system then this reduces to the ordinary matrix inverse. The general difficulty that we face with Eq. (4) is the inversion of the Fisher information matrix, especially in deep networks with many parameters. On the other hand, the model $\mathcal{M}_V$ is obtained as the image of the model $\mathcal{M}$ which can be easier to handle, despite the fact that it “lives” in the larger space $\mathcal{P}_{V,H}$. Instead of optimising the function $\mathcal{L}$ on $\mathcal{M}_V$ we can try to optimise the pull-back of $\mathcal{L}$, that is $\mathcal{L} \circ \pi_V$, which is defined on $\mathcal{M}$. But this creates a conceptual problem related to the very nature of the natural gradient method. As $\mathcal{M}$ inherits the Fisher–Rao metric from $\mathcal{P}_{V,H}$, we can express the corresponding gradient as

$$\text{grad}_\xi^{\mathcal{M}} (\mathcal{L} \circ \pi_V) = G^+(\xi) \nabla_\xi \mathcal{L}. \quad (5)$$

This can simplify the problem in various ways. As already outlined, $\mathcal{M}_V$ typically has singularities, even if $\mathcal{M}$ is a smooth manifold. In that case, the gradient (5) is well defined for all ξ , whereas the gradient (4) is not. A further simplification comes from the fact that $\mathcal{M}$ is typically associated with some network, which implies a block structure of the Fisher information matrix $G(\xi)$ in $p_\xi \in \mathcal{M}$. In Section 2, we will demonstrate this simplification for models that are associated with directed acyclic graphs, where

the elements of $\mathcal{M}$ factorise accordingly. With this simplification, the inversion of $G(\xi)$ can become much easier than the inversion of $\overline{G}(\xi)$ (when the latter is defined). On the other hand, if we consider the model $\mathcal{M}_V$ to be the prime model, where the hidden units play the role of auxiliary units, then we *have* to use the information geometry of $\mathcal{M}_V$ for learning. Therefore, it is important to relate the corresponding natural gradients, that is (4) and (5), to each other. This is done in a second step, presented in Sect. 3. In particular, we will identify conditions for the equivalence of the two gradients, leading to a new interpretation of Chentsov's classical characterisation of the Fisher–Rao metric in terms of its invariance with respect to congruent Markov morphisms [12]. (A general version of this characterisation is provided in [8].) Based on the comparison of the gradients (4) and (5), we will investigate how to extend locality properties of learning that hold for $\mathcal{M}$ to the model $\mathcal{M}_V$. This is closely related to the above-mentioned approximate methods as alternatives to the natural gradient method. Of particular relevance in this context is the replacement of the Fisher information matrix by the unitwise Fisher information matrices as studied in [20, 26]. Note, however, that we are not aiming at *approximating* the natural gradient on $\mathcal{M}_V$ by the unitwise natural gradient. In this article, we aim at identifying conditions for their *equivalence*. Furthermore, in order to satisfy these conditions we propose an extension $\tilde{\mathcal{M}}$ of $\mathcal{M}$ which corresponds to an interesting extension of the underlying network. This will lead us to a new interpretation of so-called recognition models, which are used in the context of Helmholtz machines and the wake-sleep algorithm [13, 17, 25]. Information-geometric works on the wake-sleep algorithm and its close relation to the *em*-algorithm are classical [1, 14, 18]. More recent contributions to the information geometry of the wake-sleep algorithm are provided by [10] and [31]. Directions of related research in view of this article are outlined in the conclusions, Sect. 4.

2 Locality of deep learning in Bayesian and neural networks

2.1 Locality of the Euclidean gradient

We now define a sub-manifold of $\mathcal{P}_{V,H}$ in terms of a directed acyclic graph $G = (N, E)$ where E is the set of directed edges. For a node s , we define the set $pa(s) := \{r \in N : (r, s) \in E\}$ of its *parents* and the set $ch(s) := \{t \in N : (s, t) \in E\}$ of its *children*. The latter will be only used in “Gibbs sampling” of the Appendix.

With each node s we associate a local *Markov kernel*, that is a map

$$k^s : X_{pa(s)} \times X_s \rightarrow (0, 1), \quad (x_{pa(s)}, x_s) \mapsto k^s(x_s | x_{pa(s)}),$$

with $\sum_{x_s} k^s(x_s | x_{pa(s)}) = 1$ for all $x_{pa(s)} \in X_{pa(s)}$. Note that for $pa(s) = \emptyset$, the configuration set $X_{pa(s)}$ consists of one element, the empty configuration. In this case, a Markov kernel reduces to a probability vector over X_s . (We will revisit Markov kernels from a geometric perspective in Section 3.2.) Given such a family of Markov kernels, we define the joint distribution

$$p(x_N) = \prod_{s \in N} k^s(x_s | x_{pa(s)}). \quad (6)$$

The distributions of the product structure (6) form a (statistical) model that plays an important role within the field of graphical models, in particular Bayesian networks [21]. A natural sub-model is given by the product distributions, that is those distributions of the form

$$p(x_N) = \prod_{s \in N} p(x_s).$$

In order to treat a more general sub-model, in particular one that is given by a neural network, a so-called *neuro-manifold*, we consider for each unit $s \in N$ a parametrisation $\mathbb{R}^{d_s} \ni \xi_s = (\xi_{(s;1)}, \dots, \xi_{(s;d_s)}) \mapsto \kappa_{\xi_s}^s$. This defines a model $\mathcal{M}$ as the image of the map

$$\xi := (\xi_s)_{s \in N} \mapsto p_\xi(x_N) = \prod_{s \in N} k^s(x_s | x_{pa(s)}; \xi_s), \quad (7)$$

where $k^s(x_s | x_{pa(s)}; \xi_s) := \kappa_{\xi_s}^s(x_s | x_{pa(s)})$. In order to use vector and matrix notation, we consider a numbering of the units, that is $N = \{s_1, \dots, s_{n+m}\}$, with $i \leq j$ whenever $s_i \in pa(s_j)$. To simplify notation, we can alternatively assume, without loss of generality, $N = \{1, 2, \dots, n+m\}$ such that $r \leq s$ whenever $r \in pa(s)$. This allows us to write the parametrisation (7) as a mapping from a domain in $\mathbb{R}^d$, $d = \sum_{s=1}^{n+m} d_s$, to $\mathcal{M}$, where

$$\xi = (\underbrace{\xi_{(1;1)}, \dots, \xi_{(1;d_1)}}_{=\xi_1}, \underbrace{\xi_{(2;1)}, \dots, \xi_{(2;d_2)}}_{=\xi_2}, \dots, \underbrace{\xi_{(n+m;1)}, \dots, \xi_{(n+m;d_{n+m})}}_{=\xi_{n+m}}) \quad (8)$$

is mapped to $p_\xi \in \mathcal{M}$.

Now we come to the main objective of learning as studied in this article. Given a target probability vector $p^* \in \mathcal{P}_V$ on the state set of visible units, the aim is to represent, or at least approximate, it by an appropriate element $\hat{p}$ of the model $\mathcal{M}_V = \pi_V(\mathcal{M}) \subseteq \mathcal{P}_V$. Such an approximation requires a measure of proximity, a *divergence*, between probability vectors. Information geometry provides ways to identify a natural choice of such a divergence, referred to as *canonical divergence* [6]. In the present context, the relative entropy or Kullback–Leibler divergence (abbreviated as KL-divergence)

$$D(p \| q) = \sum_x p(x) \ln \frac{p(x)}{q(x)}$$

between two probability vectors p and q is the most commonly used divergence. This leads to the search of a probability vector $\hat{p} \in \mathcal{M}_V$ that satisfies

$$D(p^* \| \mathcal{M}_V) := \inf_{p \in \mathcal{M}_V} D(p^* \| p) = D(p^* \| \hat{p}). \quad (9)$$

For this search we use the parametrisation (7) of the elements of $\mathcal{M}$ and define the function

$$\begin{aligned}\mathcal{L}(\xi) &:= D(p^* \parallel \pi_V(p_\xi)) \\ &= \sum_{x_V} p^*(x_V) \ln \frac{p^*(x_V)}{p(x_V; \xi)} \\ &= \sum_{x_V} p^*(x_V) \ln \frac{p^*(x_V)}{\sum_{x_H} p(x_V, x_H; \xi)}.\end{aligned}\quad (10)$$

Minimisation of $\mathcal{L}$ can be realised in terms of the gradient method. In this section we begin with the Euclidean gradient which is determined by the partial derivatives of $\mathcal{L}$. It is remarkable that, even though the network can be large, with many hidden units, the resulting derivatives are local in a very useful way (see a similar derivation in the context of sigmoid belief networks in [24]):

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \xi_{(r;i)}}(\xi) &= \frac{\partial}{\partial \xi_{(r;i)}} \sum_{x_V} p^*(x_V) \ln \frac{p^*(x_V)}{p(x_V; \xi)} \\ &= - \sum_{x_V} p^*(x_V) \frac{\partial \ln p(x_V; \cdot)}{\partial \xi_{(r;i)}}(\xi) \\ &= - \sum_{x_V} \frac{p^*(x_V)}{p_\xi(x_V)} \sum_{x_H} \frac{\partial p(x_V, x_H; \cdot)}{\partial \xi_{(r;i)}}(\xi) \\ &= - \sum_{x_V} \frac{p^*(x_V)}{p_\xi(x_V)} \sum_{x_H} \frac{\partial}{\partial \xi_{(r;i)}} \prod_{s \in N} k^s(x_s | x_{pa(s)}; \xi_s) \\ &= - \sum_{x_V} \frac{p^*(x_V)}{p_\xi(x_V)} \sum_{x_H} \prod_{\substack{s \in N \\ s \neq r}} k^s(x_s | x_{pa(s)}; \xi_s) \frac{\partial k^r(x_r | x_{pa(r)}; \cdot)}{\partial \xi_{(r;i)}}(\xi_r) \\ &= - \sum_{x_V} \frac{p^*(x_V)}{p_\xi(x_V)} \sum_{x_H} \prod_{s \in N} k^s(x_s | x_{pa(s)}; \xi_s) \frac{\partial \ln k^r(x_r | x_{pa(r)}; \cdot)}{\partial \xi_{(r;i)}}(\xi_r) \\ &= - \sum_{x_V, x_H} \frac{p^*(x_V)}{p_\xi(x_V)} p_\xi(x_V, x_H) \frac{\partial \ln k^r(x_r | x_{pa(r)}; \cdot)}{\partial \xi_{(r;i)}}(\xi_r) \\ &= - \sum_{x_V, x_H} p^*(x_V) p_\xi(x_H | x_V) \frac{\partial}{\partial \xi_{(r;i)}} \ln k^r(x_r | x_{pa(r)}; \xi_r).\end{aligned}$$

With $p^*(x_V, x_H; \xi) := p^*(x_V) p(x_H | x_V; \xi)$, we finally obtain

$$\begin{aligned}
 & \frac{\partial \mathcal{L}}{\partial \xi_{(r;i)}}(\xi) \\
 &= - \sum_{x_V, x_H} p^*(x_V, x_H; \xi) \frac{\partial}{\partial \xi_{(r;i)}} \ln k^r(x_r | x_{pa(r)}; \xi_r) \\
 &= - \sum_{x_{pa(r)}} p^*(x_{pa(r)}; \xi) \sum_{x_r} p^*(x_r | x_{pa(r)}; \xi) \frac{\partial}{\partial \xi_{(r;i)}} \ln k^r(x_r | x_{pa(r)}; \xi_r). \quad (11)
 \end{aligned}$$

We have an expectation value of a function, $\frac{\partial}{\partial \xi_{(r;i)}} \ln k^r(x_r | x_{pa(r)}; \xi_r)$, that is local in two ways: All arguments of this function, the states *and* the parameters, are local with respect to the node r . However, the distribution p_ξ^* , used for the evaluation of the expectation value, depends on the full set of parameters ξ . On the other hand, due to the locality of $\frac{\partial}{\partial \xi_{(r;i)}} \ln k^r(x_r | x_{pa(r)}; \xi_r)$ with respect to the states $x_{pa(r)}$ and x_r , this expectation value depends only on the marginal $p^*(x_{pa(r)}, x_r)$. One natural way to approximate (11) is by sampling from this distribution. This is typically difficult, compared to the sampling from p_ξ which factorises according to the underlying directed acyclic graph G . “One-shot sampling” from p_ξ is possible by simply using p_ξ as a generative model, which here simply means recursive application of the local kernels $k_{\xi_r}^r$ according to the underlying directed acyclic graph. This kind of sampling is also referred to as *ancestral sampling* [15]. As p_ξ^* incorporates the target distribution p^* on X_V and does not necessarily factorise according to G , sampling from it has to run much longer. For completeness, the Gibbs sampling method is outlined in more detail in “Gibbs sampling” of the Appendix.

We exemplify the derivative (11) in the context of binary neurons where it leads to a natural learning rule.

Example 1 (Neural networks (I)) We assume that the units $r \in N$, referred to as *neurons* in this context, are binary with state sets $\{-1, +1\}$. For each neuron r , we consider a vector $w_r = (w_{ir})_{i \in pa(r)}$ of *synaptic connection strengths* and a *threshold value* ϑ_r . For a synaptic strength w_{ir} , i is referred to as the *pre-synaptic* and r the *post-synaptic neuron*, respectively. We set $\xi_{(r;i)} := w_{ir}$, $i = 1, \dots, d_r - 1$, and $\xi_{(r;d_r)} := \vartheta_r$, that is $\xi_r = (w_r, \vartheta_r)$. In order to update its state, the neuron first evaluates the local function

$$h_r(x_{pa(r)}) := \sum_{i \in pa(r)} w_{ir} x_i - \vartheta_r$$

and then generates a state $x_r \in \{-1, +1\}$ with probability

$$k^r(x_r | x_{pa(r)}; w_r, \vartheta_r) := \frac{1}{1 + e^{-x_r h_r(x_{pa(r)})}}. \quad (12)$$

We calculate the derivatives

$$\frac{\partial}{\partial w_{ir}} \ln k^r(x_r | x_{pa(r)}; w_r, \vartheta_r) = \frac{x_i x_r}{1 + e^{x_r h_r(x_{pa(r)})}}, \quad (13)$$

$$\frac{\partial}{\partial \vartheta_r} \ln k^r(x_r | x_{pa(r)}; w_r, \vartheta_r) = -\frac{x_r}{1 + e^{x_r h_r(x_{pa(r)})}}, \quad (14)$$

and, with (11), we obtain

$$\frac{\partial \mathcal{L}}{\partial w_{ir}}(w, \vartheta) = - \sum_{x_{pa(r)}} \sum_{x_r} p_{\xi}^*(x_{pa(r)}, x_r) \frac{x_i x_r}{1 + e^{x_r h_r(x_{pa(r)})}}, \quad (15)$$

$$\frac{\partial \mathcal{L}}{\partial \vartheta_r}(w, \vartheta) = \sum_{x_{pa(r)}} \sum_{x_r} p_{\xi}^*(x_{pa(r)}, x_r) \frac{x_r}{1 + e^{x_r h_r(x_{pa(r)})}}. \quad (16)$$

Equation (15) is one instance of the *Hebb rule* which is based on the learning paradigm phrased as “cells that fire together wire together” [16]. Note, however, that the causal interpretation of the underlying directed acyclic graph ensures that the pre-synaptic activity x_i is measured *before* the post-synaptic activity x_r . This causally consistent version of the Hebb rule has been experimentally studied in the context of *spike-timing-dependent plasticity* of real neurons (e.g., [9]). In order to use the derivatives (15) and (16) for learning, we have to sample from p_{ξ}^* . An outline of Gibbs sampling in this context is provided in Example 7 of the Appendix.

2.2 The wake–sleep algorithm

We now highlight an important alternative to sampling from p_{ξ}^* for the computation of the derivative (11). This alternative is based on the idea that we have, in addition to the generative model $\mathcal{M}$ of distributions p_{ξ} , a so-called *recognition model* $\mathcal{Q}_{H|V}$ of conditional distributions $q(x_H | x_V; \eta)$ with which we can approximate $p(x_H | x_V; \xi)$. As a consequence, such a recognition model allows us to approximate (11) where we replace $p^*(x_V, x_H; \xi) = p^*(x_V) p(x_H | x_V; \xi)$ by $q^*(x_V, x_H; \eta) := p^*(x_V) q(x_H | x_V; \eta)$, and correspondingly the marginals on $pa(r) \cup \{r\}$. We obtain

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \xi_{(r;i)}}(\xi) &\approx - \sum_{x_{pa(r)}} q^*(x_{pa(r)}; \eta) \sum_{x_r} q^*(x_r | x_{pa(r)}; \eta) \frac{\partial}{\partial \xi_{(r;i)}} \ln k^r(x_r | x_{pa(r)}; \xi_r) \quad (17) \\ &= - \frac{\partial}{\partial \xi_{(r;i)}} \sum_{x_{pa(r)}} q^*(x_{pa(r)}; \eta) \sum_{x_r} q^*(x_r | x_{pa(r)}; \eta) \ln k^r(x_r | x_{pa(r)}; \xi_r) \quad (18) \\ &= \frac{\partial}{\partial \xi_{(r;i)}} \sum_{x_{pa(r)}} q^*(x_{pa(r)}; \eta) \sum_{x_r} q^*(x_r | x_{pa(r)}; \eta) \ln \frac{q^*(x_r | x_{pa(r)}; \eta)}{k^r(x_r | x_{pa(r)}; \xi_r)}. \quad (19) \end{aligned}$$

For the evaluation of the gradient of $\mathcal{L}$ with respect to the ξ -parameters we can now use the recognition model, instead of the generative model. This approximation will

be the more accurate the smaller the following relative entropy is:

$$D(\xi \parallel \eta) := \sum_{x_V} p(x_V; \xi) \sum_{x_H} p(x_H | x_V; \xi) \ln \frac{p(x_H | x_V; \xi)}{q(x_H | x_V; \eta)}. \quad (20)$$

Ideally, we would like the recognition model to be rich enough to represent the conditional distributions of the generative model. More precisely, we assume that for all ξ , there is an $\eta = \eta(\xi)$ so that $q(x_H | x_V; \eta) = p(x_H | x_V; \xi)$. Furthermore, for (20) to be tractable, we assume that $q(x_H | x_V; \eta)$ also factorises according to some directed acyclic graph G' , so that

$$q(x_H | x_V; \eta) = \prod_{r \in H} l^r(x_r | x_{pa'(r)}; \eta_r), \quad (21)$$

where $pa'(r)$ denotes the parent set of the node r with respect to the graph G' . With the assumption (21), the expression (20) simplifies considerably, and we obtain

$$\begin{aligned} & \frac{\partial D(\xi \parallel \cdot)}{\partial \eta_{(r; j)}}(\eta) \\ &= -\frac{\partial}{\partial \eta_{(r; j)}} \sum_{x_{pa'(r)}} p(x_{pa'(r)}; \xi) \sum_{x_r} p(x_r | x_{pa'(r)}; \xi) \ln l^r(x_r | x_{pa'(r)}; \eta_r) \quad (22) \\ &= \frac{\partial}{\partial \eta_{(r; j)}} \sum_{x_{pa'(r)}} p(x_{pa'(r)}; \xi) \sum_{x_r} p(x_r | x_{pa'(r)}; \xi) \ln \frac{p(x_r | x_{pa'(r)}; \xi)}{l^r(x_r | x_{pa'(r)}; \eta_r)}. \quad (23) \end{aligned}$$

Note that, while p_ξ factorises according to G so that the conditional distribution $p(x_r | x_{pa(r)}; \xi)$ coincides with the kernel $k^r(x_r | x_{pa'(r)}; \xi)$, the conditional distribution $p(x_r | x_{pa'(r)}; \xi)$ with respect to G' does not have a correspondingly simple structure. On the other hand, we can easily sample from p_ξ , and thereby also from $p(x_{pa'(r)}; \xi)$ and $p(x_r | x_{pa'(r)}; \xi)$, using the product structure with respect to G .

Let us now come back to the original problem of minimising $\mathcal{L}$ with respect to ξ based on the gradient descent method. If the parameter η of the recognition model is such that $q(x_H | x_V; \eta) = p(x_H | x_V; \xi)$ then the approximation (17) is exact, and we can evaluate the partial derivatives $\partial/\partial \xi_{(r; i)}$ by sampling from $q^*(x_V, x_H; \eta) = p^*(x_V) q(x_H | x_V; \eta)$. This can then be used for updating the parameter ξ , say from ξ to $\xi + \Delta \xi$ where $\Delta \xi$ is proportional to the Euclidean gradient. As this update is based on sampling from the target distribution $p^*(x_V)$ and the recognition model $q(x_H | x_V; \eta)$, it is referred to as the *wake phase*. After this update, we typically have $q(x_H | x_V; \eta) \neq p(x_H | x_V; \xi + \Delta \xi)$. In order to use (17) for the next update of ξ , we therefore have to readjust η , say from η to $\eta + \Delta \eta$, so that we recover the identity $q(x_H | x_V; \eta + \Delta \eta) = p(x_H | x_V; \xi + \Delta \xi)$. This can be achieved by choosing $\Delta \eta$ to be proportional to the Euclidean gradient (22) with respect to η . The evaluation of the partial derivatives $\partial/\partial \eta_{(r; j)}$ requires sampling from the generative model $p(x_V, x_H; \xi)$, with no involvement of the target distribution $p^*(x_V)$. This is the reason why the η -update is referred to as the *sleep phase*. Alternating application of the wake

phase and the sleep phase yields the so-called *wake-sleep algorithm*, which has been introduced and studied in the context of neural networks in [13,17,25]. It has been pointed out that this algorithm cannot be interpreted as a gradient decent algorithm of a potential function on both variables ξ and η . On the other hand, here we derived the wake-sleep algorithm as a gradient decent algorithm for the optimisation of the objective function $\mathcal{L}$ which only depends on the variable ξ . The auxiliary variable η is used for the approximation of the gradient of $\mathcal{L}$ with respect to ξ . In order to have a good approximation of this gradient, we have to apply the sleep phase update more often, until convergence of η . Only then, we can update ξ within the next wake phase. With this asymmetry of time scale for the two phases, the wake-sleep algorithm is a gradient decent algorithm for ξ , which has been pointed out in the context of the *em*-algorithm in [18].

We have introduced the parameters η for sampling and thereby evaluating the derivative (11). However, there is another remarkable feature of the corresponding extended optimisation problem. While the original optimisation function $\mathcal{L}$, defined by (10), does not appear to be local in any sense, the extended optimisation in terms of a generalised wake-sleep algorithm, which is equivalent to the original problem, is based on a set of local functions associated with the respective units. More precisely, the expressions (18) and (22) are derivatives of local cross entropies, whereas the expressions (19) and (23) are derivatives of local KL-divergences.

We conclude with the important note that a recognition model which, on the one hand, is rich enough to represent all distributions $p(x_H|x_V; \xi)$ and, on the other hand, factorises according to (21) might require a large graph G' and a correspondingly large number of parameters $\eta_{(r,j)}$ which constitute the vector η . In practice, the recognition model is typically chosen to be of the same dimensionality as the generative model and does not necessarily satisfy the above conditions.

Example 2 Figure 1a depicts a typical generative network G , which underlies the model $p(x_V|x_H; \xi)$. It is a directed acyclic network, and we assume that the model is simply given by the set of all joint distributions on $X_H \times X_V$ that factorise according to G . In Fig. 1b, we have a typical recognition network G' obtained from the generative network G of Fig. 1a by reverting the directions of all arrows. The corresponding recognition model is given by the set of all conditional distributions $q(x_H|x_V)$ that factorise according to G' . However, this model is not large enough to ensure that for all ξ , there is an η such that $q(x_H|x_V, \eta) = p(x_H|x_V, \xi)$. Adding further lateral connections, as shown in Fig. 1c, enlarges G' , and we obtain a correspondingly enlarged recognition model which now has that property.

2.3 Locality of the natural gradient

In the previous section, we have computed the partial derivatives (11) which turn out to be local and allow us to apply the gradient method for learning. However, from the information-geometric point of view, we have to use the Fisher–Rao metric for the gradient, which leads us to the natural gradient method. In general, the natural gradient is difficult to evaluate because the Fisher information matrix has to be inverted (see Eqs.

Fig. 1 Comparison of generative and recognition networks: **a** a typical generative network G ; **b** a typical recognition network G' obtained from the generative network G by reverting the directions; **c** recognition network G' extended by lateral connections

(4) and (5)). In our context of a model that is associated with a directed acyclic graph G , however, the Fisher information matrix simplifies considerably. More precisely, we consider a model $\mathcal{M}$ with the parametrisation (7). The tangent space of $\mathcal{M}$ in p_ξ is spanned by the vectors

$$\partial_{(r;i)}(\xi) := \frac{\partial}{\partial \xi_{(r;i)}} p_\xi, \quad r = 1, \dots, n+m, \quad i = 1, \dots, d_r. \quad (24)$$

The following theorem specifies the structure of the Fisher information matrix $G(\xi)$ with the entries $g_{(r;i)(s;j)}(\xi) = \langle \partial_{(r;i)}(\xi), \partial_{(s;j)}(\xi) \rangle_\xi$.

Theorem 1 *Let $\mathcal{M}$ be a model with the parametrisation (7). Then the Fisher information matrix $G(\xi) := (g_{(r;i)(s;j)}(\xi))_{(r;i)(s;j)}$ decomposes into “local” $d_r \times d_r$ matrices $G_r(\xi) := (g_{(r;i)(j)}(\xi))_{i,j}$, $r \in N$, with*

$$g_{(r;i)(j)}(\xi) := \sum_{x_{pa(r)}} p(x_{pa(r)}; \xi) C(x_{pa(r)}; \xi_r), \quad (25)$$

where

$$C(x_{pa(r)}; \xi_r) := \sum_{x_r} k^r(x_r | x_{pa(r)}; \xi_r) \times \frac{\partial \ln k^r(x_r | x_{pa(r)}; \cdot)}{\partial \xi_{(r;i)}}(\xi_r) \frac{\partial \ln k^r(x_r | x_{pa(r)}; \cdot)}{\partial \xi_{(r;j)}}(\xi_r). \quad (26)$$

With this, the entries of the Fisher information matrix $G(\xi)$ are given by $g_{(r;i)(s;j)}(\xi) = g_{(r;i)(j)}(\xi)$ whenever $r = s$, and $g_{(r;i)(s;j)}(\xi) = 0$ otherwise. Using matrix notation, we have

$$G(\xi) = \begin{pmatrix} G_1(\xi) & & 0 \\ & \ddots & \\ 0 & & G_{m+n}(\xi) \end{pmatrix}.$$

Proof The parametrisation (7) yields

$$\ln p(x; \xi) = \sum_{r \in N} \ln k^r(x_r | x_{pa(r)}; \xi_r)$$

and therefore

$$\frac{\partial \ln p(x; \cdot)}{\partial \xi_{(s;j)}}(\xi) = \frac{\partial \ln k^s(x_s | x_{pa(s)}; \cdot)}{\partial \xi_{(s;j)}}(\xi_s). \quad (27)$$

In what follows, we use the shorthand notation $x_{<s}$ and $x_{>s}$ for $x_{\{i \in N : i < s\}}$ and $x_{\{i \in N : i > s\}}$, respectively. With (27) we obtain for $r \leq s$:

$$\begin{aligned} & g(r;i)(s;j)(\xi) \\ &= \sum_x p(x; \xi) \frac{\partial \ln p(x; \cdot)}{\partial \xi_{(r;i)}}(\xi) \frac{\partial \ln p(x; \cdot)}{\partial \xi_{(s;j)}}(\xi) \\ &= \sum_x p(x; \xi) \frac{\partial \ln k^r(x_r | x_{pa(r)}; \cdot)}{\partial \xi_{(r;i)}}(\xi_r) \frac{\partial \ln k^s(x_s | x_{pa(s)}; \cdot)}{\partial \xi_{(s;j)}}(\xi_s) \quad (\text{by (27)}) \\ &= \sum_{x_{<s}} \sum_{x_s} \sum_{x_{>s}} \left\{ \prod_{i < s} k^i(x_i | x_{pa(i)}; \xi_i) \right\} k^s(x_s | x_{pa(s)}; \xi_s) \left\{ \prod_{i > s} k^i(x_i | x_{pa(i)}; \xi_i) \right\} \\ &\quad \times \frac{\partial \ln k^r(x_r | x_{pa(r)}; \cdot)}{\partial \xi_{(r;i)}}(\xi_r) \frac{\partial \ln k^s(x_s | x_{pa(s)}; \cdot)}{\partial \xi_{(s;j)}}(\xi_s) \\ &= \sum_{x_{<s}} \left\{ \prod_{i < s} k^i(x_i | x_{pa(i)}; \xi_i) \right\} \sum_{x_s} k^s(x_s | x_{pa(s)}; \xi_s) \underbrace{\left\{ \sum_{x_{>s}} \prod_{i > s} k^i(x_i | x_{pa(i)}; \xi_i) \right\}}_{=1} \\ &\quad \times \frac{\partial \ln k^r(x_r | x_{pa(r)}; \cdot)}{\partial \xi_{(r;i)}}(\xi_r) \frac{\partial \ln k^s(x_s | x_{pa(s)}; \cdot)}{\partial \xi_{(s;j)}}(\xi_s) \\ &= \sum_{x_{pa(s)}} p_\xi(x_{pa(s)}) \sum_{x_s} k^s(x_s | x_{pa(s)}; \xi_s) \\ &\quad \times \frac{\partial \ln k^r(x_r | x_{pa(r)}; \cdot)}{\partial \xi_{(r;i)}}(\xi_r) \frac{\partial \ln k^s(x_s | x_{pa(s)}; \cdot)}{\partial \xi_{(s;j)}}(\xi_s). \end{aligned}$$

If $r \neq s$, this expression reduces to

$$\begin{aligned} & \sum_{x_{pa(s)}} p_{\xi}(x_{pa(s)}) \frac{\partial \ln k^r(x_r | x_{pa(r)}; \cdot)}{\partial \xi_{(r;i)}}(\xi_r) \\ & \quad \times \sum_{x_s} k^s(x_s | x_{pa(s)}; \xi_s) \frac{\partial \ln k^s(x_s | x_{pa(s)}; \cdot)}{\partial \xi_{(s;j)}}(\xi_s) \\ &= \sum_{x_{pa(s)}} p_{\xi}(x_{pa(s)}) \frac{\partial \ln k^r(x_r | x_{pa(r)}; \cdot)}{\partial \xi_{(r;i)}}(\xi_r) \underbrace{\sum_{x_s} \frac{\partial k^s(x_s | x_{pa(s)}; \cdot)}{\partial \xi_{(s;j)}}(\xi_s)}_{=0} \\ &= 0. \end{aligned}$$

This concludes the proof. $\square$

Theorem 1 highlights a number of simplifications of the Fisher information matrix as result of the particular parametrisation of the model in terms of a directed acyclic graph. The presented proof is adapted from [5] (see also the related work [28]):

1. The Fisher information matrix $G(\xi)$ has a block structure, reflecting the structure of the underlying graph (see Fig. 3). Each block $G_r(\xi)$ corresponds to a node r and has $d_r \times d_r$ components. Outside of these blocks the matrix is filled with zeros. The natural gradient method requires the inversion of $G(\xi)$ (the usual inverse $G^{-1}(\xi)$, if it exists, or, more generally, the Moore–Penrose inverse $G^+(\xi)$). With the block structure of $G(\xi)$, this inversion reduces to the inversion of the individual matrices $G_r(\xi)$. The corresponding simplification of the natural gradient is summarised in Corollary 1.
2. The terms $g_{(r;i,j)}(\xi)$, defined by (25), are expectation values of the functions $C(x_{pa(r)}; \xi_r)$. These functions are local in two ways. On the one hand, they depend only on local states $x_{pa(r)}$ and, on the other hand, only local parameters ξ_r are involved (see the definition (26)). This kind of locality is very useful in applications of the natural gradient method. Especially in the context of neural networks, locality of learning is considered to be essential. Note, however, that the terms $g_{(r;i,j)}(\xi)$ are not completely local. This is because the expectation value (25) is evaluated with respect to p_{ξ} , where ξ is the full parameter vector. (As only the distribution of $X_{pa(r)}$ appears, parameters of non-ancestors of r do not play a role in the definition of $g_{(r;i,j)}(\xi)$, which simplifies the situation to some extent.) In order to evaluate the Fisher information matrix in applications, we have to overcome this non-locality by sampling from $p(x_{pa(r)}; \xi)$. As we are dealing with directed acyclic graphs, this can be simply done by recursive application of the local kernels $k_{\xi_r}^r$.

To highlight the relevance of Theorem 1, let us consider a few simple examples.

Example 3 (*Exponential families*) Consider the model given by local kernels of the exponential form

$$k^r(x_r | x_{pa(r)}; \xi_r) = \frac{\exp\left(\sum_{i=1}^{d_r} \xi_{(r;i)} \phi^{(r;i)}(x_{pa(r)}, x_r)\right)}{\sum_{x'_r} \exp\left(\sum_{i=1}^{d_r} \xi_{(r;i)} \phi^{(r;i)}(x_{pa(r)}, x'_r)\right)}. \quad (28)$$

In this case, the expression (25) yields

$$g_{(r;i,j)}(\xi) = \sum_{x_{pa(r)}} p(x_{pa(r)}; \xi) \text{Cov}\left(\phi^{(r;i)}(x_{pa(r)}, \cdot), \phi^{(r;j)}(x_{pa(r)}, \cdot) \mid x_{pa(r)}; \xi_r\right), \quad (29)$$

where the conditional covariance on the RHS of (29) is evaluated with respect to $k^r(\cdot | x_{pa(r)}; \xi_r)$.

Example 4 (*Neural networks (II)*) Neural networks, which we introduced in Example 1, can be considered as a special case of the models of Example 3. This can be seen by rewriting the transition probability (12) as follows:

$$\begin{aligned} k^r(x_r | x_{pa(r)}; w_r, \vartheta_r) &= \frac{1}{1 + e^{-x_r h_r(x_{pa(r)})}} \\ &= \frac{e^{\frac{1}{2} x_r h_r(x_{pa(r)})}}{e^{\frac{1}{2} x_r h_r(x_{pa(r)})} + e^{-\frac{1}{2} x_r h_r(x_{pa(r)})}} \\ &= \frac{\exp\left(\frac{1}{2} \sum_{j \in pa(r)} w_{jr} x_j x_r - \frac{1}{2} \vartheta_r x_r\right)}{\sum_{x'_r} \exp\left(\frac{1}{2} \sum_{j \in pa(r)} w_{jr} x_j x'_r - \frac{1}{2} \vartheta_r x'_r\right)}. \end{aligned}$$

This is a special case of (28) which only involves pairwise interactions. In order to evaluate the terms (26) we need the derivatives

$$\frac{\partial}{\partial w_{ir}} \ln k^r(x_r | x_{pa(r)}; w_r, \vartheta_r) = \frac{x_i x_r}{1 + e^{x_r h_r(x_{pa(r)})}}, \quad (30)$$

$$\frac{\partial}{\partial \vartheta_r} \ln k^r(x_r | x_{pa(r)}; w_r, \vartheta_r) = -\frac{x_r}{1 + e^{x_r h_r(x_{pa(r)})}}. \quad (31)$$

According to Theorem 1, we can evaluate the Fisher information matrix in a local way. More explicitly, we have

$$g_{(r;i,j)}(\xi) = \sum_{x_{pa(r)}} p_\xi(x_{pa(r)}) \frac{f(x_i, x_j)}{(1 + e^{h_r(x_{pa(r)})})(1 + e^{-h_r(x_{pa(r)})})},$$

Fig. 2 Two architectures with the same number of parameters but different complexity of the Fisher information matrix

where

$$f(x_i, x_j) := \begin{cases} x_i x_j & \text{if } 1 \leq i, j \leq d_r - 1, \\ -x_i & \text{if } 1 \leq i \leq d_r - 1, j = d_r, \\ -x_j & \text{if } i = d_r, 1 \leq j \leq d_r - 1, \\ 1 & \text{if } i = d_r, j = d_r. \end{cases}$$

Example 5 (Shallow versus deep networks) In this example, we demonstrate the difference in sparsity of the Fisher information matrix for architectures of varying depth. Fig. 2 shows two networks with three visible and nine hidden neurons each.

The number of synaptic connections is 27 in both cases, thereby assuming the neuronal model of Examples 1 and 4 (for simplicity, we do not consider the threshold values). If we associate one parameter with each edge, the synaptic strength, then we have 27 parameters in the system, and the Fisher information matrices have $27 \times 27 = 729$ entries. Theorem 1 implies the block diagonal structure of the Fisher information matrices shown in Fig. 3. As we can see, depth is associated with higher sparsity of these matrices. We have at least 486 zeros in the shallow case and at least 648 zeros in the deep case.

This example can be generalised to a network with n visible and $m = l \cdot n$ hidden neurons. As in Fig. 2, in the one case we arrange all m hidden neurons in one layer of width $l \cdot n$ and, in the other case, we arrange the hidden neurons in l layers of width n . In both cases, we have $n \cdot m = n(l \cdot n) = l \cdot n^2$ edges, corresponding to the number of parameters, and therefore the Fisher information matrix has $l^2 n^4$ entries. With the shallow architecture, we have at most $n(l \cdot n)^2 = l^2 n^3$ non-zero entries, whereas in the deep architecture there are at most $l \cdot n \cdot n^2 = l \cdot n^3$ non-zero entries. The difference is $l^2 \cdot n^3 - l \cdot n^3 = l \cdot n^3(l - 1)$. For $n = l = 3$, we recover the difference of the above numbers, $648 - 486 = 162$.

Example 6 (Restricted Boltzmann machine) If we deal with models that are associated with undirected graphs we cannot expect the Fisher information matrix to have a block diagonal structure. Consider, for instance, a restricted Boltzmann machine, as shown

Fig. 3 The structure of the Fisher information matrix of size $27 \times 27 = 729$: **a** the shallow case where we have 486 zeros; **b** the deep case where we have 648 zeros

Fig. 4 The architecture of a restricted Boltzmann machine

in Fig. 4. With each edge $(i, j) \in V \times H$ we associate a weight w_{ij} and denote the full weight matrix by W . The family of all weight matrices parametrises the model $\mathcal{M}$ consisting of distributions

$$p(x_V, x_H; W) = \frac{e^{\sum_{i \in V, j \in H} w_{ij} x_i x_j}}{\sum_{x'_V, x'_H} e^{\sum_{i \in V, j \in H} w_{ij} x'_i x'_j}}.$$

Note that this deviates somewhat from the setting of a restricted Boltzmann machine as we ignore the threshold values for simplicity.

The Fisher information matrix on $\mathcal{M}$ is given by

$$g_{ij,kl}(W) = \text{Cov}_{p(\cdot; W)}(X_i X_j, X_k X_l)$$

which has no zeros imposed by the architecture.

The simplification of the Fisher information matrix, stated in Theorem 1, has several important consequences. As an immediate consequence we obtain a corresponding simplification of the natural gradient of a smooth real-valued function $\mathcal{L}$ on $\mathcal{M}$, mainly referring to the function (10), in terms of the parametrisation (7). In order to express

this simplification we consider the vectors (24) which span the tangent space of $\mathcal{M}$ in p_ξ . In particular, they allow us to represent the gradient of $\mathcal{L}$ as a linear combination

$$\text{grad}_\xi^{\mathcal{M}} \mathcal{L} = \sum_{r=1}^{n+m} \sum_{i=1}^{d_r} x^{(r;i)}(\xi) \partial_{(r;i)}(\xi). \quad (32)$$

Corollary 1 *Consider the situation of Theorem 1 and a real-valued smooth function $\mathcal{L}$ on $\mathcal{M}$. With*

$$\nabla_{\xi_r} \mathcal{L} := \begin{pmatrix} \frac{\partial \mathcal{L}}{\partial \xi_{(r;1)}}(\xi) \\ \vdots \\ \frac{\partial \mathcal{L}}{\partial \xi_{(r;d_r)}}(\xi) \end{pmatrix},$$

we have the following coordinates of the natural gradient of $\mathcal{L}$ in the representation (32):

$$x^r(\xi) := \begin{pmatrix} x^{(r;1)}(\xi) \\ \vdots \\ x^{(r;d_r)}(\xi) \end{pmatrix} = G_r^+(\xi) \nabla_{\xi_r} \mathcal{L}, \quad r = 1, \dots, n+m. \quad (33)$$

Here, $G_r^+(\xi)$ denotes the Moore–Penrose inverse of the matrix $G_r(\xi)$ defined by (25) and (26). (It reduces to the usual matrix inverse whenever $G_r(\xi)$ has maximal rank.)

Note that Theorem 1 as well as its Corollary 1 can equally be applied to the recognition model $\mathcal{Q}_{H|V}$ defined by (21). In Section 2.2 we have studied natural objective functions that involve both, the generative as well as the recognition model, and highlighted their locality properties. Together with the locality of the corresponding Fisher information matrices, these properties allow us to evaluate a natural gradient version of the wake-sleep algorithm, referred to as *natural wake-sleep algorithm* in [31].

The prime objective function to be optimised is typically defined on the projected model $\mathcal{M}_V$. It naturally carries the Fisher–Rao metric of $\mathcal{P}_V$ so that we can define the natural gradient of the given objective function directly on $\mathcal{M}_V$. On the other hand, we have seen that the Fisher information matrix on the full model $\mathcal{M} \subseteq \mathcal{P}_{V,H}$ has a block structure associated with the underlying network. This implies useful locality properties of the natural gradient and thereby makes the method applicable within the context of deep learning. The main problem that we are now going to study is the following: Can we extend the locality of the natural gradient on the full model $\mathcal{M}$, as stated in Corollary 1, to the natural gradient on the projected model $\mathcal{M}_V$? In the following section we first study this problem in a more general setting of Riemannian manifolds.

3 Gradients on full versus coarse grained models

3.1 The general problem

We now develop a more general perspective, which we motivate by analogy to the context of the previous sections. Assume that we have two Riemannian manifolds $(\mathcal{Z}, g^{\mathcal{Z}})$ and $(\mathcal{X}, g^{\mathcal{X}})$ with dimensions $d_{\mathcal{Z}}$ and $d_{\mathcal{X}}$, respectively, and a differentiable map $\pi : \mathcal{Z} \rightarrow \mathcal{X}$, with its differential $d\pi_p : T_p\mathcal{Z} \rightarrow T_{\pi(p)}\mathcal{X}$ in p . The manifold $\mathcal{Z}$ corresponds to the manifold of (strictly positive) distributions on the full set of units, the visible and the hidden units. The map π plays the role of the marginalisation map which marginalises out the hidden units and which we will interpret in Sect. 3.2 as one instance of a more general coarse graining procedure. Typically, we have a model $\mathcal{M} \subseteq \mathcal{Z}$ which corresponds to a model consisting of the joint distributions on the full system that can be represented by a network. It is obtained in terms of a parametrisation $\varphi : \mathcal{E} \rightarrow \mathcal{Z}$, $\xi \mapsto p_{\xi}$, where $\mathcal{E}$ is a differentiable manifold, usually an open subset of $\mathbb{R}^d$. In general, $\mathcal{M}$ will not be a sub-manifold of $\mathcal{Z}$ and can contain various kinds of singularities. We restrict attention to the non-singular points of $\mathcal{M}$. A point p in $\mathcal{M} \subseteq \mathcal{Z}$ is said to be a *non-singular* point of $\mathcal{M}$ if there exists a smooth chart $\psi : \mathcal{U} \rightarrow \mathcal{U}'$ with an open set $\mathcal{U}$ in $\mathcal{Z}$ and an open set $\mathcal{U}'$ in $\mathbb{R}^{d_{\mathcal{Z}}}$ such that $p \in \mathcal{U}$ and, for some k ,

$$\begin{aligned} \psi(\mathcal{M} \cap \mathcal{U}) \\ = \{(x_1, \dots, x_k, x_{k+1}, \dots, x_{d_{\mathcal{Z}}}) \in \mathcal{U}' : x_{k+1} = \dots = x_{d_{\mathcal{Z}}} = 0\}. \end{aligned} \quad (34)$$

Note that k is a local dimension of $\mathcal{M}$ in p , which is upper bounded by the dimension d of $\mathcal{E}$. We denote the set of non-singular points of $\mathcal{M}$ by $\text{Smooth}(\mathcal{M})$. If a point $p \in \mathcal{M}$ is not non-singular, it is called a *singularity* or a *singular point* of $\mathcal{M}$ (for more details see [32]). In a non-singular point p , the tangent space $T_p\mathcal{M}$ is well defined. Throughout this article, we will assume that the parametrisation φ of $\mathcal{M}$ is a *proper parametrisation* in the sense that for all $p \in \text{Smooth}(\mathcal{M})$ and all $\xi \in \mathcal{E}$ with $\varphi(\xi) = p$, the image of the differential $d\varphi_{\xi}$ coincides with the full tangent space $T_p\mathcal{M}$. This assumption is required, but often not explicitly stated, when dealing with the natural gradient method for optimisation on parametrised models. More precisely, when we interpret the Fisher information matrix (2) as a “coordinate representation” of the Fisher–Rao metric, we implicitly assume that the vectors $\partial_i(\xi) = \frac{\partial}{\partial \xi_i} p_{\xi}$, $i = 1, \dots, d$, span the tangent space of the model in p_{ξ} . Note that linear independence, which ensures the non-degeneracy of the Fisher information matrix, is not required and would in fact be too restrictive given that overparametrised models play an important role within the field of deep learning.

We now consider a smooth function $\mathcal{L} : \mathcal{X} \rightarrow \mathbb{R}$ and study its gradient on $\mathcal{X}$ (with respect to $g^{\mathcal{X}}$) in relation to the corresponding gradient of $\mathcal{L} \circ \pi : \mathcal{Z} \rightarrow \mathbb{R}$ on $\text{Smooth}(\mathcal{M})$ (with respect to $g^{\mathcal{Z}}$). For a non-singular point of $\mathcal{M}$, we decompose the tangent space $T_p\mathcal{M}$ into a “vertical component” $T_p^{\mathcal{V}}\mathcal{M} := T_p\mathcal{M} \cap \ker d\pi_p$ and its orthogonal complement $T_p^{\mathcal{H}}\mathcal{M}$ in $T_p\mathcal{M}$, the corresponding “horizontal component”. (The symbols $\mathcal{V}$ and $\mathcal{H}$ should not be confused with the symbols V and H , denoting the

visible and the hidden units, respectively.) We have the following proposition where we use the somewhat simpler notation “ $\langle \cdot, \cdot \rangle$ ” for both metrics, $g^{\mathcal{Z}}$ and $g^{\mathcal{X}}$.

Proposition 1 *Consider a model $\mathcal{M}$ in $\mathcal{Z}$ and a differentiable map $\pi : \mathcal{Z} \rightarrow \mathcal{X}$ and let p be a non-singular point of $\mathcal{M}$. Assume that the following compatibility condition is satisfied:*

$$A, B \in T_p^{\mathcal{H}}\mathcal{M} \Rightarrow \langle A, B \rangle_p = \langle d\pi_p(A), d\pi_p(B) \rangle_{\pi(p)}. \quad (35)$$

Then, for all smooth functions $\mathcal{L} : \mathcal{X} \rightarrow \mathbb{R}$, we have

$$d\pi_p \left(\text{grad}_p^{\mathcal{M}}(\mathcal{L} \circ \pi) \right) = \Pi \left(\text{grad}_{\pi(p)}^{\mathcal{X}} \mathcal{L} \right), \quad (36)$$

where Π denotes the projection of tangent vectors in $T_{\pi(p)}\mathcal{X}$ onto $d\pi_p(T_p\mathcal{M})$.

Proof First observe that $\text{grad}_p^{\mathcal{M}}(\mathcal{L} \circ \pi) \in T_p^{\mathcal{H}}\mathcal{M}$. Indeed, for all $B \in T_p^{\mathcal{V}}\mathcal{M}$ we have

$$\langle \text{grad}_p^{\mathcal{M}}(\mathcal{L} \circ \pi), B \rangle_p = d(\mathcal{L} \circ \pi)_p(B) = d\mathcal{L}_{\pi(p)}(\underbrace{d\pi_p(B)}_{=0}) = 0. \quad (37)$$

Let $A' \in d\pi_p(T_p\mathcal{M}) \subseteq T_{\pi(p)}\mathcal{X}$. There exists $A \in T_p\mathcal{M}$ such that $d\pi_p(A) = A'$. We can decompose A orthogonally into a component A_1 contained in $T_p^{\mathcal{V}}\mathcal{M}$ and a component A_2 contained in $T_p^{\mathcal{H}}\mathcal{M}$. With this decomposition we have $A' = d\pi_p(A) = d\pi_p(A_1 + A_2) = d\pi_p(A_2)$. This implies

$$\begin{aligned} \langle d\pi_p(\text{grad}_p^{\mathcal{M}}(\mathcal{L} \circ \pi)), A' \rangle_{\pi(p)} &= \langle d\pi_p(\text{grad}_p^{\mathcal{M}}(\mathcal{L} \circ \pi)), d\pi_p(A_2) \rangle_{\pi(p)} \\ &= \langle \text{grad}_p^{\mathcal{M}}(\mathcal{L} \circ \pi), A_2 \rangle_p \quad (\text{by (37) and (35)}) \\ &= d(\mathcal{L} \circ \pi)_p(A_2) \\ &= d\mathcal{L}_{\pi(p)}(d\pi_p(A_2)) \\ &= \langle \text{grad}_{\pi(p)}^{\mathcal{X}} \mathcal{L}, d\pi_p(A_2) \rangle_{\pi(p)} \\ &= \langle \text{grad}_{\pi(p)}^{\mathcal{X}} \mathcal{L}, A' \rangle_{\pi(p)}. \end{aligned}$$

This proves Eq. (36). $\square$

As stated above, the parametrised model $\mathcal{M}$ plays the role of the distributions on the full network, consisting of the visible and hidden units. We want to relate this model to the projected model $\mathcal{S} := \pi(\mathcal{M})$. The composition of the parametrisation φ and the projection π serves as a parametrisation $\xi \mapsto \pi(p_\xi)$ of $\mathcal{S}$ as shown in the following diagram.

$$\begin{array}{ccccc} & & \pi \circ \varphi & & \\ & \curvearrowright & & \curvearrowleft & \\ \mathcal{E} & \xrightarrow{\varphi} & \mathcal{M} & \xrightarrow{\pi} & \mathcal{S} \end{array}$$

The map $\pi \circ \varphi$ is a proper parametrisation of $\mathcal{S}$ if for all $q \in \text{Smooth}(\mathcal{S})$ and all ξ with $\pi(p_\xi) = q$, the image of the differential $d(\pi \circ \varphi)_\xi$ coincides with the full tangent space $T_q\mathcal{S}$. Obviously, this does not follow from the assumption that φ is a proper parametrisation of $\mathcal{M}$ and requires further assumptions. One necessary, but not sufficient, condition is the following: Assume that $\pi \circ \varphi$ is a proper parametrisation of $\mathcal{S}$ and consider a point $p \in \text{Smooth}(\mathcal{M})$ with $\pi(p) \in \text{Smooth}(\mathcal{S})$. With $\xi \in \mathcal{E}$, $\varphi(\xi) = p$, we have

$$\begin{aligned} T_{\pi(p)}\mathcal{S} &= d(\pi \circ \varphi)_\xi (T_\xi \mathcal{E}) && (\pi \circ \varphi \text{ proper parametrisation}) \\ &= d\pi_{\varphi(\xi)} (d\varphi_\xi (T_\xi \mathcal{E})) && (\text{by the chain rule}) \\ &= d\pi_{\varphi(\xi)} (T_{\varphi(\xi)}\mathcal{M}) && (\varphi \text{ proper parametrisation}) \\ &= d\pi_p (T_p\mathcal{M}). \end{aligned} \quad (38)$$

The condition (38) is sufficient for $\pi \circ \varphi$ being a proper parametrisation of $\mathcal{S}$ if $\pi^{-1}(\text{Smooth}(\mathcal{S})) \subseteq \text{Smooth}(\mathcal{M})$, which is clearly satisfied if $\mathcal{M}$ is a smooth submanifold of $\mathcal{Z}$ and therefore has no singularities. In any case, the condition (38) is required when dealing with properly parametrised models. We call a point $p \in \mathcal{M}$ *admissible* if $p \in \text{Smooth}(\mathcal{M})$, $\pi(p) \in \text{Smooth}(\mathcal{S})$, and (38) is satisfied in p .

We have the following implication of Proposition 1.

Theorem 2 *Consider a model $\mathcal{M}$ in $\mathcal{Z}$ and a differentiable map $\pi : \mathcal{Z} \rightarrow \mathcal{X}$ with image $\mathcal{S} = \pi(\mathcal{M})$. Furthermore, assume that the compatibility condition (35) is satisfied in an admissible point $p \in \mathcal{M}$. Then for all smooth functions $\mathcal{L} : \mathcal{X} \rightarrow \mathbb{R}$, we have*

$$d\pi_p \left(\text{grad}_p^{\mathcal{M}}(\mathcal{L} \circ \pi) \right) = \text{grad}_{\pi(p)}^{\mathcal{S}} \mathcal{L}. \quad (39)$$

Proof This follows directly from (36). For an admissible point p we have $d\pi_p (T_p\mathcal{M}) = T_{\pi(p)}\mathcal{S}$. Therefore, the RHS of (36) reduces to the orthogonal projection of $\text{grad}_{\pi(p)}^{\mathcal{X}} \mathcal{L}$ onto $T_{\pi(p)}\mathcal{S}$ which equals the $\text{grad}_{\pi(p)}^{\mathcal{S}} \mathcal{L}$. $\square$

Note that, if we do not assume (38), we have to replace the RHS of (39) by $\Pi \left(\text{grad}_{\pi(p)}^{\mathcal{S}} \mathcal{L} \right)$, where Π denotes the projection of tangent vectors in $T_{\pi(p)}\mathcal{S}$ onto $d\pi_p (T_p\mathcal{M})$. Therefore, it can well be the case that the gradient on $\mathcal{M}$ vanishes in a point p while the corresponding gradient on $\mathcal{S}$, that is $\text{grad}_{\pi(p)}^{\mathcal{S}} \mathcal{L}$, does not. Such a point p is referred to as *spurious critical point* (see [30]). In addition to the problem of having singularities of $\mathcal{M}$ and $\mathcal{S} = \pi(\mathcal{M})$, this represents another problem with gradient methods for the optimisation of smooth functions on parametrised models. However, the problem of spurious critical points does not appear if we are dealing with a proper parametrisation φ of $\mathcal{M}$ for which $\pi \circ \varphi$ is also a proper parametrisation of $\mathcal{S}$.

We conclude this section by addressing the following problem: If we assume that the compatibility condition (35) is satisfied for a model $\mathcal{M}$ in $\mathcal{Z}$, what can we say about the corresponding compatibility for a sub-model $\mathcal{M}'$ of $\mathcal{M}$? In general we

cannot expect that (35) also holds for $\mathcal{M}'$. The following theorem characterises those sub-models $\mathcal{M}'$ of $\mathcal{M}$ for which this is satisfied, so that Theorem 2 will also hold for them.

Theorem 3 *Assume that (35) holds for a model $\mathcal{M}$ in $\mathcal{Z}$ and consider a sub-model $\mathcal{M}' \subseteq \mathcal{M}$. Then (35) also holds for $\mathcal{M}'$ if and only if for each point $p \in \text{Smooth}(\mathcal{M}')$ the tangent space $T_p\mathcal{M}'$ satisfies*

$$T_p\mathcal{M}' = \left(T_p\mathcal{M}' \cap T_p^{\mathcal{H}}\mathcal{M}\right) + \left(T_p\mathcal{M}' \cap T_p^{\mathcal{V}}\mathcal{M}\right). \quad (40)$$

This theorem is a direct implication of Lemma 1 below which reduces the problem to the simple setting of linear algebra. Let $(\mathcal{F}, \langle \cdot, \cdot \rangle_{\mathcal{F}})$, $(\mathcal{G}, \langle \cdot, \cdot \rangle_{\mathcal{G}})$ be two finite-dimensional real Hilbert spaces, and let $T : \mathcal{F} \rightarrow \mathcal{G}$ be a linear map. We can decompose $\mathcal{F}$ into a “vertical component” $\mathcal{F}^{\mathcal{V}} := \ker T$ and its orthogonal complement $\mathcal{F}^{\mathcal{H}}$ in $\mathcal{F}$, the corresponding “horizontal component”. Now let $\mathcal{E}$ be a linear subspace of $\mathcal{F}$, equipped with the induced inner product $\langle \cdot, \cdot \rangle_{\mathcal{E}}$, and consider the restriction $T_{\mathcal{E}} : \mathcal{E} \rightarrow \mathcal{G}$ of T to $\mathcal{E}$. Denoting by $\perp_{\mathcal{E}}$ and $\perp_{\mathcal{F}}$ the orthogonal complements in $\mathcal{E}$ and $\mathcal{F}$, respectively, we can decompose $\mathcal{E}$ into

$$\begin{aligned} \mathcal{E}^{\mathcal{V}} &:= \ker T_{\mathcal{E}} \\ &= \mathcal{E} \cap \ker T \\ &= \mathcal{E} \cap \mathcal{F}^{\mathcal{V}}, \end{aligned}$$

and

$$\begin{aligned} \mathcal{E}^{\mathcal{H}} &:= \left(\mathcal{E}^{\mathcal{V}}\right)^{\perp_{\mathcal{E}}} \\ &= \mathcal{E} \cap \left(\mathcal{E}^{\mathcal{V}}\right)^{\perp_{\mathcal{F}}} \\ &= \mathcal{E} \cap \left(\mathcal{E} \cap \mathcal{F}^{\mathcal{V}}\right)^{\perp_{\mathcal{F}}} \\ &= \mathcal{E} \cap \left(\mathcal{E}^{\perp_{\mathcal{F}}} + \mathcal{F}^{\mathcal{H}}\right) \\ &\supseteq \mathcal{E} \cap \mathcal{F}^{\mathcal{H}}. \end{aligned}$$

Note that, while we always have $\mathcal{E}^{\mathcal{V}} \subseteq \mathcal{F}^{\mathcal{V}}$, in general $\mathcal{E}^{\mathcal{H}} \not\subseteq \mathcal{F}^{\mathcal{H}}$.

Lemma 1 *Assume:*

$$A, B \in \mathcal{F}^{\mathcal{H}} \Rightarrow \langle A, B \rangle_{\mathcal{F}} = \langle T(A), T(B) \rangle_{\mathcal{G}}. \quad (41)$$

Then the following two statements about a subspace $\mathcal{E}$ of $\mathcal{F}$ are equivalent:

$$(i) \ A, B \in \mathcal{E}^{\mathcal{H}} \Rightarrow \langle A, B \rangle_{\mathcal{E}} = \langle T_{\mathcal{E}}(A), T_{\mathcal{E}}(B) \rangle_{\mathcal{G}}. \quad (42)$$

$$(ii) \quad \mathcal{E} = (\mathcal{E} \cap \mathcal{F}^{\mathcal{H}}) + (\mathcal{E} \cap \mathcal{F}^{\mathcal{V}}). \quad (43)$$

Proof Let us first assume that (43) holds true. This implies

$$\mathcal{E}^{\mathcal{H}} = (\mathcal{E}^{\mathcal{V}})^{\perp_{\mathcal{E}}} = (\mathcal{E} \cap \mathcal{F}^{\mathcal{V}})^{\perp_{\mathcal{E}}} = \mathcal{E} \cap \mathcal{F}^{\mathcal{H}} \subseteq \mathcal{F}^{\mathcal{H}}. \quad (44)$$

For all $A, B \in \mathcal{E}^{\mathcal{H}} \subseteq \mathcal{F}^{\mathcal{H}}$, (41) then takes the form

$$\langle A, B \rangle_{\mathcal{E}} = \langle A, B \rangle_{\mathcal{F}} = \langle T(A), T(B) \rangle_{\mathcal{G}} = \langle T_{\mathcal{E}}(A), T_{\mathcal{E}}(B) \rangle_{\mathcal{G}}. \quad (45)$$

In order to prove the opposite implication, we assume that (43) does not hold for $\mathcal{E}$. This means that

$$\mathcal{Q} := (\mathcal{E} \cap \mathcal{F}^{\mathcal{H}}) + (\mathcal{E} \cap \mathcal{F}^{\mathcal{V}}) \quad (46)$$

is a proper subspace of $\mathcal{E}$. We denote the orthogonal complement of $\mathcal{Q}$ in $\mathcal{E}$ by $\mathcal{R}$ and choose a non-trivial vector A in $\mathcal{R}$. Such a vector can be uniquely decomposed as a sum of two non-trivial vectors $A_1 \in \mathcal{F}^{\mathcal{H}}$ and $A_2 \in \mathcal{F}^{\mathcal{V}}$. This implies

$$\begin{aligned} \|A\|_{\mathcal{E}} &= \|A\|_{\mathcal{F}} \\ &= \|A_1 + A_2\|_{\mathcal{F}} \\ &> \|A_1\|_{\mathcal{F}} \\ &= \|T(A_1)\|_{\mathcal{G}} \\ &= \|T(A_1) + T(A_2)\|_{\mathcal{G}} \\ &= \|T(A)\|_{\mathcal{G}} \\ &= \|T_{\mathcal{E}}(A)\|_{\mathcal{G}}. \end{aligned}$$

This means that (42) does not hold for the subspace $\mathcal{E}$. □

3.2 A new interpretation of Chentsov's theorem

We now come back to the context of probability distributions but take a slightly more general perspective than in Section 1.2. We interpret X_V as a coarse graining of the set $X_V \times X_H$ which lumps together all pairs $(v, h), (v', h')$ with $v = v'$. Replacing the Cartesian product $X_V \times X_H$ by a general set Z , a coarse graining of Z is an onto mapping $X : Z \rightarrow X$, which partitions Z into the atoms $Z_x := X^{-1}(x)$. The corresponding push-forward map is given by

$$X_* : \mathcal{P}(Z) \rightarrow \mathcal{P}(X), \quad p = \sum_z p(z) \delta^z \mapsto X_*(p) = \sum_x \underbrace{\left(\sum_{z \in Z_x} p(z) \right)}_{=: p(x)} \delta^x,$$

with the differential

$$dX_* : \mathcal{T}(\mathcal{Z}) \rightarrow \mathcal{T}(\mathcal{X}), \quad A = \sum_z A(z) \delta^z \mapsto \sum_x \left(\sum_{z \in \mathcal{Z}_x} A(z) \right) \delta^x.$$

Obviously, we have

$$\mathcal{V}_p := \ker dX_* = \left\{ \sum_z A(z) \delta^z : \sum_{z \in \mathcal{Z}_x} A(z) = 0 \text{ for all } x \right\}, \quad (47)$$

and the orthogonal complement

$$\mathcal{H}_p := \mathcal{V}_p^\perp = \left\{ \tilde{A} = \sum_x \frac{A(x)}{p(x)} \sum_{z \in \mathcal{Z}_x} p(z) \delta^z : \sum_x A(x) = 0 \right\} \quad (48)$$

with respect to the Fisher–Rao metric in p (note that $\mathcal{V}_p$ is independent of p). Given a vector $A = \sum_z A(z) \delta^z \in \mathcal{T}(\mathcal{Z})$, we can decompose it uniquely as

$$A = A^{\mathcal{H}} + A^{\mathcal{V}},$$

with $A^{\mathcal{H}} \in \mathcal{H}_p$ and $A^{\mathcal{V}} \in \mathcal{V}_p$. More precisely,

$$A^{\mathcal{H}} = \sum_x \sum_{z \in \mathcal{Z}_x} \left(\frac{p(z)}{p(x)} \sum_{z' \in \mathcal{Z}_x} A(z') \right) \delta^z, \quad (49)$$

$$A^{\mathcal{V}} = \sum_x \sum_{z \in \mathcal{Z}_x} \left(A(z) - \frac{p(z)}{p(x)} \sum_{z' \in \mathcal{Z}_x} A(z') \right) \delta^z. \quad (50)$$

For a vector

$$\tilde{A} = \sum_x \frac{A(x)}{p(x)} \sum_{z \in \mathcal{Z}_x} p(z) \delta^z \in \mathcal{H}_p,$$

we have

$$dX_*(\tilde{A}) = A. \quad (51)$$

We now examine the inner product of two such vectors $\tilde{A}, \tilde{B} \in \mathcal{H}_p$:

$$\begin{aligned}
 \langle \tilde{A}, \tilde{B} \rangle_p &= \sum_z \frac{1}{p(z)} \tilde{A}(z) \tilde{B}(z) \\
 &= \sum_x \sum_{z \in Z_x} \frac{1}{p(z)} \left(\frac{A(x)}{p(x)} p(z) \right) \left(\frac{B(x)}{p(x)} p(z) \right) \\
 &= \sum_x \frac{A(x)}{p(x)} \frac{B(x)}{p(x)} \sum_{z \in Z_x} p(z) \\
 &= \sum_x \frac{1}{p(x)} A(x) B(x) \\
 &= \langle A, B \rangle_{X_*(p)} \\
 &= \langle dX_*(\tilde{A}), dX_*(\tilde{B}) \rangle_{X_*(p)}. \quad (\text{by (51)})
 \end{aligned} \tag{52}$$

Thus, the compatibility condition (35) is satisfied. Theorem 2 implies that for all smooth functions $\mathcal{L} : \mathcal{P}(X) \rightarrow \mathbb{R}$ and all $p \in \mathcal{P}(Z)$, the following equality of gradients holds (note that all points $p \in \mathcal{P}(Z)$ are admissible):

$$dX_* \left(\text{grad}_p^{\mathcal{P}(Z)} (\mathcal{L} \circ X_*) \right) = \text{grad}_{X_*(p)}^{\mathcal{P}(X)} \mathcal{L}, \tag{53}$$

where $\mathcal{P}(Z)$ and $\mathcal{P}(X)$ are equipped with the respective Fisher–Rao metrics. Even though this is a simple observation, it highlights an important point here. A coarse graining is generally associated with a loss of information, which is expressed by the monotonicity of the Fisher–Rao metric. This information loss is maximal when we project from the full space $\mathcal{P}(Z)$ onto $\mathcal{P}(X)$. Nevertheless, the gradient of any function $\mathcal{L}$ that is defined on $\mathcal{P}(X)$ is not sensitive to this information loss. In order to study models $\mathcal{M}$ in $\mathcal{P}(Z)$ with the same invariance of gradients, we have to impose the condition (40), which takes the form

$$T_p \mathcal{M} = (T_p \mathcal{M} \cap \mathcal{H}_p) + (T_p \mathcal{M} \cap \mathcal{V}_p), \quad p \in \text{Smooth}(\mathcal{M}). \tag{54}$$

Definition 1 If a model $\mathcal{M} \subseteq \mathcal{P}(Z)$ satisfies the condition (54) in p , we say that it is *cylindrical in p* . If it is cylindrical in all non-singular points, we say that it is *(pointwise) cylindrical*.

Of particular interest are cylindrical models with a trivial vertical component. These are the models, for which the coarse graining X is a minimal sufficient statistic. They have been used by Chentsov [12] in order to characterise the Fisher–Rao metric (see Theorem 4). To be more precise, we need to revisit Markov kernels from a geometric perspective. We consider the space of linear maps from $\mathbb{R}^Z$ to $\mathbb{R}^X$, which is canonically isomorphic to $(\mathbb{R}^Z)^* \otimes \mathbb{R}^X$, and define the polytope of Markov kernels as

$$\mathcal{K}(Z|X) := \left\{ K = \sum_{x,z} k(z|x) \delta^z \otimes e_x \in (\mathbb{R}^Z)^* \otimes \mathbb{R}^X : \right. \\ \left. k(z|x) \geq 0 \text{ for all } x, z, \text{ and } \sum_z k(z|x) = 1 \text{ for all } x \right\}.$$

The set $\mathcal{P}(Z)$ of probability vectors is a subset where each vector p is identified with $k(z|x) := p(z)$. We now consider a Markov kernel K that is coupled with the coarse graining $X : Z \rightarrow X$ in terms of $k(z|x) > 0$ if and only if $z \in Z_x$. Such a Markov kernel is called *X-congruent*. It defines an embedding $K_* : \mathcal{P}(X) \rightarrow \mathcal{P}(Z)$,

$$p = \sum_x p(x) \delta^x \mapsto \sum_z \left(\sum_x p(x) k(z|x) \right) \delta^z = \sum_x p(x) \left(\sum_{z \in Z_x} k(z|x) \delta^z \right),$$

referred to as an *X-congruent Markov morphism*. We obviously have $X_* \circ K_* = \text{id}_{\mathcal{P}(X)}$. The image of K_* , which we denote by $\mathcal{M}(K)$, is the relative interior of the simplex with the extreme points

$$\sum_{z \in Z_x} k(z|x) \delta^z, \quad x \in X.$$

It is easy to see that $\mathcal{M}(K)$ is cylindrical. (For $p \in \mathcal{M}(K)$, we have $T_p \mathcal{M}(K) = \mathcal{H}_p$. This implies $T_p \mathcal{M}(K) \cap \mathcal{H}_p = \mathcal{H}_p$ and $T_p \mathcal{M}(K) \cap \mathcal{V}_p = \{0\}$, which verifies (54).) Therefore, we have for all smooth functions $\mathcal{L} : \mathcal{P}(X) \rightarrow \mathbb{R}$ and all $p \in \mathcal{M}(K)$,

$$dX_* \left(\text{grad}_p^{\mathcal{M}(K)} (\mathcal{L} \circ X_*) \right) = \text{grad}_{X_*(p)}^{\mathcal{P}(X)} \mathcal{L}. \quad (55)$$

Comparing the Eq. (55) with (53), we observe that the gradient on the LHS is now evaluated on $\mathcal{M}(K)$, with respect to the induced Fisher–Rao metric. The gradient on the RHS remains as it is.

The differential of K_* is given by

$$dK_* : \mathcal{T}(X) \rightarrow \mathcal{T}(Z), \quad A = \sum_x A(x) \delta^x \mapsto \sum_x A(x) \left(\sum_{z \in Z_x} k(z|x) \delta^z \right),$$

with the image

$$\text{im } dK_* = \left\{ \sum_x A(x) \left(\sum_{z \in Z_x} k(z|x) \delta^z \right) : \sum_x A(x) = 0 \right\}.$$

Fig. 5 X -congruent Markov morphism with the following coarse graining $X: z_1 \mapsto x_1, z_2 \mapsto x_2, z_3 \mapsto x_3, z_4 \mapsto x_3$. The inner product between A and B equals the inner product of $dK_*(A)$ and $dK_*(B)$ (see (56))

The following simple calculation shows that K_* is an isometric embedding (see Fig. 5):

$$\begin{aligned}
 & \langle dK_*(A), dK_*(B) \rangle_{K_*(p)} \\
 &= \sum_x \sum_{z \in Z_x} \frac{1}{\sum_{x'} p(x') k(z|x')} \left(\sum_{x'} A(x') k(z|x') \right) \left(\sum_{x'} B(x') k(z|x') \right) \\
 &= \sum_x \sum_{z \in Z_x} \frac{1}{p(x) k(z|x)} A(x) k(z|x) B(x) k(z|x) \\
 &= \sum_x \frac{1}{p(x)} A(x) B(x) \sum_{z \in Z_x} k(z|x) \\
 &= \sum_x \frac{1}{p(x)} A(x) B(x) \\
 &= \langle A, B \rangle_p.
 \end{aligned} \tag{56}$$

The invariance (55) of gradients follows also directly from the invariance (56) of inner products. In fact, K_* being an isometric embedding is equivalent to (55) (for details, see the proof of Theorem 5). A fundamental result of Chentsov [12] characterises the Fisher–Rao metric as the only invariant metric (see also [8]).

Theorem 4 (Chentsov’s theorem) *Assume that for any non-empty finite set S , $\mathcal{P}(S)$ is equipped with a Riemannian metric $g^{(S)}$, such that the following is satisfied: Whenever we have a coarse graining $X: Z \rightarrow X$ and an X -congruent Markov morphism $K_*: \mathcal{P}(X) \rightarrow \mathcal{P}(Z)$, the invariance (56) holds, interpreted as a condition for $g^{(X)}$ and $g^{(Z)}$. Then there exists a positive real number α such that for all S , the metric $g^{(S)}$ coincides with the Fisher–Rao metric multiplied by α .*

In order to compute the gradient of a function on an extended space that is equivalent to the actual gradient, we want to use Eq. (39) of Theorem 2. Instances of this equivalence are given by the Eqs. (53) and (55) where we considered two extreme cases, the full model $\mathcal{P}(Z)$ and the model $\mathcal{M}(K)$, respectively, which both project onto $\mathcal{P}(X)$. We know that Theorem 2 also holds for all cylindrical models $\mathcal{M}$, including, but not restricted to, intermediate cases where $\mathcal{M}(K) \subseteq \mathcal{M} \subseteq \mathcal{P}(Z)$. How flexible are we here with the choice of the metric? In fact, a reformulation of Chentsov's uniqueness result, Theorem 4, identifies the Fisher–Rao metric as the only metric for which Eq. (39) holds.

Theorem 5 *Assume that for any non-empty finite set S , $\mathcal{P}(S)$ is equipped with a Riemannian metric $g^{(S)}$. Then the following properties are equivalent:*

1. *Let $X : Z \rightarrow X$ be a coarse graining, $\mathcal{M}$ a cylindrical model in $\mathcal{P}(Z)$, and $\mathcal{M}_X := X_*(\mathcal{M})$ its image. Then for all smooth functions $\mathcal{L} : \mathcal{P}(X) \rightarrow \mathbb{R}$ and all admissible points $p \in \mathcal{M}$, we have*

$$dX_* \left(\text{grad}_p^{\mathcal{M}}(\mathcal{L} \circ X_*) \right) = \text{grad}_{X_*(p)}^{\mathcal{M}_X} \mathcal{L}, \quad (57)$$

where the gradient on the LHS is evaluated with respect to the restriction of $g^{(Z)}$ and the RHS is evaluated with respect to the restriction of $g^{(X)}$.

2. *There exists a positive real number α such that for all S , the metric $g^{(S)}$ coincides with the Fisher–Rao metric multiplied by α .*

Proof “(1) $\Rightarrow$ (2):” Let $X : Z \rightarrow X$ be a coarse graining, and let $K_* : \mathcal{P}(X) \rightarrow \mathcal{P}(Z)$ be an X -congruent Markov morphism. We consider the model $\mathcal{M}(K)$, as a special instance of a cylindrical model $\mathcal{M}$ in $\mathcal{P}(Z)$. In that case, (57) is equivalent to

$$\text{grad}_{K_*(p)}^{\mathcal{M}(K)}(\mathcal{L} \circ X_*) = dK_* \left(\text{grad}_p^{\mathcal{P}(X)} \mathcal{L} \right). \quad (58)$$

We choose $p \in \mathcal{P}(X)$ and $A, B \in \mathcal{T}(X)$. We can represent A as a gradient of a function $\mathcal{L}$. More precisely, with

$$\mathcal{L} : \mathcal{P}(X) \rightarrow \mathbb{R}, \quad p \mapsto \mathcal{L}(p) := \sum_x A(x) \cdot \ln p(x), \quad (59)$$

we have

$$\text{grad}_p^{\mathcal{P}(X)} \mathcal{L} = A. \quad (60)$$

This implies

$$\begin{aligned}
 \langle A, B \rangle_p &= \left\langle \text{grad}_p^{\mathcal{P}(X)} \mathcal{L}, B \right\rangle_p && \text{(by (60))} \\
 &= d\mathcal{L}_p(B) \\
 &= d\mathcal{L}_p((dX_* \circ dK_*)(B)) \\
 &= d(\mathcal{L} \circ X_*)_{K_*(p)}(dK_*(B)) && \text{(by the chain rule)} \\
 &= \left\langle \text{grad}_{K_*(p)}^{\mathcal{M}(K)} (\mathcal{L} \circ X_*), dK_*(B) \right\rangle_{K_*(p)} \\
 &= \left\langle dK_* \left(\text{grad}_p^{\mathcal{P}(X)} \mathcal{L} \right), dK_*(B) \right\rangle_{K_*(p)} && \text{(by (58))} \\
 &= \langle dK_*(A), dK_*(B) \rangle_{K_*(p)}.
 \end{aligned}$$

This proves the invariance (56). According to Chentsov's uniqueness result, Theorem 4, this invariance characterises the Fisher–Rao metric up to a constant $\alpha > 0$.

“(2) $\Rightarrow$ (1):” This follows from the compatibility (52), which holds for the Fisher–Rao metric, and Theorems 2 and 3. $\square$

3.3 Cylindrical extensions of a model

Throughout this section, we consider a model $\mathcal{M}$, together with a proper parametrisation $\mathbb{R}^d \supseteq \mathcal{E} \rightarrow \mathcal{P}(Z)$, $\xi \mapsto p_\xi \in \mathcal{M}$, satisfying that the composition $\xi \mapsto X_*(p_\xi)$ is a proper parametrisation of $\mathcal{M}_X := X_*(\mathcal{M})$. This ensures that all tangent spaces in non-singular points of $\mathcal{M}$ and $\mathcal{M}_X$, respectively, can be generated in terms of partial derivatives with respect to the parameters ξ_i , $i = 1, \dots, d$.

We can easily construct a model $\widetilde{\mathcal{M}} \subseteq \mathcal{P}(Z)$ that satisfies the conditions

$$(a) \ \mathcal{M} \subseteq \widetilde{\mathcal{M}}, \quad (b) \ X_*(\mathcal{M}) = X_*(\widetilde{\mathcal{M}}), \quad \text{and} \quad (c) \ \widetilde{\mathcal{M}} \text{ is cylindrical.} \quad (61)$$

We refer to such a model as a *cylindrical extension* of $\mathcal{M}$. Before we come to the explicit construction of cylindrical extensions, let us first demonstrate their direct use for relating the respective natural gradients to each other. Given an admissible point $p \in \mathcal{M}$ that is also admissible in $\widetilde{\mathcal{M}}$, we can decompose the tangent space $T_p \widetilde{\mathcal{M}}$ into the sum $T_p \mathcal{M} \oplus T_p^\perp \mathcal{M}$, where the second summand is the orthogonal complement of the first one in $T_p \widetilde{\mathcal{M}}$. We can use this decomposition in order to relate the natural gradient of a smooth function $\mathcal{L}$ defined on the projected model $\mathcal{M}_X$ to the natural gradient of $\mathcal{L} \circ X_*$:

$$\begin{aligned}
 \text{grad}_{X_*(p)}^{\mathcal{M}_X} \mathcal{L} &\stackrel{dX_*}{\longleftarrow} \text{grad}_p^{\widetilde{\mathcal{M}}} (\mathcal{L} \circ X_*) \\
 &= \text{grad}_p^\top (\mathcal{L} \circ X_*) + \text{grad}_p^\perp (\mathcal{L} \circ X_*) \\
 &= \text{grad}_p^{\mathcal{M}} (\mathcal{L} \circ X_*) + \text{grad}_p^\perp (\mathcal{L} \circ X_*).
 \end{aligned} \quad (62)$$

(Here “ $\top$ ” stands for the projection onto $T_p \mathcal{M}$ and “ $\perp$ ” stands for the projection onto the corresponding orthogonal complement in $T_p \widetilde{\mathcal{M}}$.) The difference between the

natural gradient on the full model $\mathcal{M}$ and the natural gradient on the coarse grained model $\mathcal{M}_X$ is given by $\text{grad}_p^\perp(\mathcal{L} \circ X_*)$ which vanishes when $\mathcal{M}$ itself is already cylindrical. Thus, the equality (62) generalises (57).

The product extension I Given a non-singular point $p_\xi = \sum_z p(z; \xi) \delta^z$ of $\mathcal{M}$, the tangent space in p_ξ is spanned by

$$\begin{aligned} \partial_i(\xi) &:= \sum_{z \in Z} \frac{\partial p(z; \cdot)}{\partial \xi_i}(\xi) \delta^z \\ &= \sum_{z \in Z} p(z; \xi) \frac{\partial \ln p(z; \cdot)}{\partial \xi_i}(\xi) \delta^z, \quad i = 1, \dots, d. \end{aligned} \quad (63)$$

Now, consider the projection of p_ξ onto $\mathcal{P}(X)$ in terms of X_* , that is $X_*(p_\xi) = \sum_{x \in X} p(x; \xi) \delta^x$ where $p(x; \xi) = \sum_{z \in Z_x} p(z; \xi)$. Assuming that this projected point is a non-singular point of $\mathcal{M}_X = X_*(\mathcal{M})$, the corresponding tangent space $T_{X_*(p_\xi)} \mathcal{M}_X$ is spanned by

$$\begin{aligned} \bar{\partial}_i^{\mathcal{H}}(\xi) &:= dX_*(\partial_i(\xi)) \\ &= \sum_{x \in X} \frac{\partial p(x; \cdot)}{\partial \xi_i}(\xi) \delta^x \\ &= \sum_{x \in X} p(x; \xi) \frac{\partial \ln p(x; \cdot)}{\partial \xi_i}(\xi) \delta^x, \quad i = 1, \dots, d. \end{aligned} \quad (64)$$

In addition to the described projection of p_ξ onto the “horizontal” space, leading to $\mathcal{M}_X$, we can also project it onto the “vertical” space. In order to do so, we define a Markov kernel $K_\xi = \sum_{x,z} p(z|x; \xi) \delta^z \otimes e_x$:

$$p(z|x; \xi) := \begin{cases} \frac{p(z; \xi)}{p(x; \xi)}, & \text{if } X(z) = x \\ 0, & \text{otherwise.} \end{cases} \quad (65)$$

We denote the image of the map $\xi \mapsto K_\xi$ by $\mathcal{M}_{Z|X} \subseteq \mathcal{K}(Z|X)$, and assume that K_ξ is a non-singular point of $\mathcal{M}_{Z|X}$. The corresponding tangent vectors in K_ξ are given by

$$\begin{aligned} \bar{\partial}_i^{\mathcal{V}}(\xi) &:= \sum_{x,z} \frac{\partial p(z|x; \cdot)}{\partial \xi_i}(\xi) \delta^z \otimes e_x \\ &= \sum_{x,z} p(z|x; \xi) \frac{\partial \ln p(z|x; \cdot)}{\partial \xi_i}(\xi) \delta^z \otimes e_x, \quad i = 1, \dots, d. \end{aligned} \quad (66)$$

Note that for all three sets of vectors, $\partial_i(\xi)$, $\bar{\partial}_i^{\mathcal{H}}(\xi)$, and $\bar{\partial}_i^{\mathcal{V}}(\xi)$, $i = 1, \dots, d$, linear independence is not required. In fact, it is important to include overparametrised systems into the analysis, where linear independence is not given.

Now, we can define the *product extension* $\widetilde{\mathcal{M}}^I$ of $\mathcal{M}$ as follows: For each pair $(\xi, \xi') \in \Xi \times \Xi$, we define $p_{\xi, \xi'} = p(\cdot; \xi, \xi')$ as

$$\begin{aligned} \sum_z p(z; \xi, \xi') \delta^z &:= \sum_x \sum_{z \in Z_x} [p(z; \xi) + p(x; \xi)(p(z|x; \xi') - p(z|x; \xi))] \delta^z \\ &= \sum_x \sum_{z \in Z_x} p(x; \xi) p(z|x; \xi') \delta^z. \end{aligned} \quad (67)$$

The product extension is then simply the set of all points that can be obtained in this way. Obviously, $\mathcal{M}$ consists of those points in $\widetilde{\mathcal{M}}^I$ that are given by identical parameters, that is $\xi = \xi'$, which proves (61) (a). Furthermore, $X_*(p_{\xi, \xi'}) = X_*(p_\xi)$, and therefore this extension has the same projection as the original model $\mathcal{M}$ so that (61) (b) is satisfied. The last requirement for $\widetilde{\mathcal{M}}^I$ to be a cylindrical extension of $\mathcal{M}$, (61) (c), will be proven below in Proposition 2. We obtain the tangent space of $\widetilde{\mathcal{M}}^I$ in $p_{\xi, \xi'}$ by taking the derivatives with respect to $\xi_1, \dots, \xi_d$ and $\xi'_1, \dots, \xi'_d$, respectively:

$$\begin{aligned} \partial_i^{\mathcal{H}}(\xi, \xi') &:= \frac{\partial}{\partial \xi_i} \sum_z p(z; \xi, \xi') \delta^z \\ &= \sum_z p(z; \xi, \xi') \frac{\partial \ln p(z; \cdot, \xi')}{\partial \xi_i}(\xi) \delta^z \\ &= \sum_x \sum_{z \in Z_x} p(z; \xi, \xi') \frac{\partial \ln p(x; \cdot)}{\partial \xi_i}(\xi) \delta^z \\ &= \sum_x \sum_{z \in Z_x} p(x; \xi) p(z|x; \xi') \frac{\partial \ln p(x; \cdot)}{\partial \xi_i}(\xi) \delta^z \\ &= \sum_x p(x; \xi) \frac{\partial \ln p(x; \cdot)}{\partial \xi_i}(\xi) \left(\sum_{z \in Z_x} p(z|x; \xi') \delta^z \right), \quad i = 1, \dots, d. \end{aligned} \quad (68)$$

A comparison with (64) shows that we have a natural isometric correspondence

$$\partial_i^{\mathcal{H}}(\xi, \xi') \longleftrightarrow \bar{\partial}_i^{\mathcal{H}}(\xi), \quad i = 1, \dots, d, \quad (69)$$

by mapping δ^x to $\sum_{z \in Z_x} p(z|x; \xi') \delta^z$ (this map is given by the X -congruent Markov morphism discussed above; see also Fig. 5). Now we consider the vertical directions:

Fig. 6 Extension of $\mathcal{M}$ to the cylindrical model $\widetilde{\mathcal{M}}^I$, with the corresponding tangent vectors

$$\begin{aligned}
 & \partial_i^V(\xi, \xi') \\
 &:= \frac{\partial}{\partial \xi'_i} \sum_z p(z; \xi, \xi') \delta^z \\
 &= \sum_z p(z; \xi, \xi') \frac{\partial \ln p(z; \xi, \cdot)}{\partial \xi'_i}(\xi') \delta^z \\
 &= \sum_x \sum_{z \in Z_x} p(z; \xi, \xi') \frac{\partial \ln p(z|x; \cdot)}{\partial \xi'_i}(\xi') \delta^z \\
 &= \sum_x \sum_{z \in Z_x} p(x; \xi) p(z|x; \xi') \frac{\partial \ln p(z|x; \cdot)}{\partial \xi'_i}(\xi') \delta^z, \quad i = 1, \dots, d. \quad (70)
 \end{aligned}$$

A comparison with (66) shows that we also have a natural correspondence

$$\partial_i^V(\xi, \xi') \longleftrightarrow \bar{\partial}_i^V(\xi'), \quad i = 1, \dots, d, \quad (71)$$

by mapping $\delta^z \otimes e_x$ to $p(x; \xi) \delta^z$, in addition to the above-mentioned correspondence (69). This proves that $(\xi, \xi') \mapsto p_{\xi, \xi'}$ is a proper parametrisation of $\widetilde{\mathcal{M}}^I$. The situation is illustrated in Fig. 6.

Now we consider the natural Fisher–Rao metric on $\widetilde{\mathcal{M}}^I \subseteq \mathcal{P}(Z)$ in $p_{\xi, \xi'}$, assuming that all points associated with (ξ, ξ') are non-singular. It follows from Proposition 2 below that $\langle \partial_i^H(\xi, \xi'), \partial_j^V(\xi, \xi') \rangle_{\xi, \xi'} = 0$ for all i, j , where $\langle \cdot, \cdot \rangle_{\xi, \xi'}$ denotes the Fisher–Rao metric in $p_{\xi, \xi'}$. For the inner products of the horizontal vectors we obtain

$$\begin{aligned}
 g_{ij}^{\mathcal{H}}(\xi, \xi') &:= \left\langle \partial_i^{\mathcal{H}}(\xi, \xi'), \partial_j^{\mathcal{H}}(\xi, \xi') \right\rangle_{\xi, \xi'} \\
 &= \sum_x p(x; \xi) \sum_{z \in \mathbb{Z}_x} p(z|x; \xi') \frac{\partial \ln p(x; \cdot)}{\partial \xi_i}(\xi) \frac{\partial \ln p(x; \cdot)}{\partial \xi_j}(\xi) \\
 &= \sum_x p(x; \xi) \frac{\partial \ln p(x; \cdot)}{\partial \xi_i}(\xi) \frac{\partial \ln p(x; \cdot)}{\partial \xi_j}(\xi) \\
 &= \left\langle \bar{\partial}_i^{\mathcal{H}}(\xi), \bar{\partial}_j^{\mathcal{H}}(\xi) \right\rangle_{\xi} \\
 &=: \bar{g}_{ij}^{\mathcal{H}}(\xi).
 \end{aligned} \tag{72}$$

In particular, these inner products do not depend on ξ' . The inner products of the vertical vectors are given by

$$\begin{aligned}
 g_{ij}^{\mathcal{V}}(\xi, \xi') &:= \left\langle \partial_i^{\mathcal{V}}(\xi, \xi'), \partial_j^{\mathcal{V}}(\xi, \xi') \right\rangle_{\xi, \xi'} \\
 &= \sum_x p(x; \xi) \sum_{z \in \mathbb{Z}_x} p(z|x; \xi') \frac{\partial \ln p(z|x; \cdot)}{\partial \xi'_i}(\xi') \frac{\partial \ln p(z|x; \cdot)}{\partial \xi'_j}(\xi').
 \end{aligned} \tag{73}$$

This defines two matrices,

$$\begin{aligned}
 \bar{G}^{\mathcal{H}}(\xi) &= (\bar{g}_{ij}^{\mathcal{H}}(\xi))_{1 \leq i, j \leq d}, \quad \text{and} \\
 G^{\mathcal{V}}(\xi, \xi') &= (g_{ij}^{\mathcal{V}}(\xi, \xi'))_{1 \leq i, j \leq d},
 \end{aligned}$$

and the Fisher information matrix $\tilde{G}(\xi, \xi')$ with respect to the product coordinate system is a block matrix,

$$\tilde{G}(\xi, \xi') = \begin{pmatrix} \bar{G}^{\mathcal{H}}(\xi) & 0 \\ 0 & G^{\mathcal{V}}(\xi, \xi') \end{pmatrix}.$$

In order to compute the gradient of a function $\tilde{\mathcal{L}} : \tilde{\mathcal{M}}^I \rightarrow \mathbb{R}$, we have to consider the pseudoinverse of $\tilde{G}(\xi, \xi')$, and, with the Euclidean gradient $\nabla_{\xi, \xi'} \tilde{\mathcal{L}} = (\nabla_{\xi} \tilde{\mathcal{L}}, \nabla_{\xi'} \tilde{\mathcal{L}})$, we have

$$\text{grad}_{\xi, \xi'}^{\tilde{\mathcal{M}}^I} \tilde{\mathcal{L}} = \begin{pmatrix} \bar{G}^{\mathcal{H}^+}(\xi) & 0 \\ 0 & G^{\mathcal{V}^+}(\xi, \xi') \end{pmatrix} \begin{pmatrix} \nabla_{\xi} \tilde{\mathcal{L}} \\ \nabla_{\xi'} \tilde{\mathcal{L}} \end{pmatrix}. \tag{74}$$

Now we assume $\tilde{\mathcal{L}} = \mathcal{L} \circ X_*$, where $\mathcal{L}$ is a function defined on the model $X_*(\tilde{\mathcal{M}}) = X_*(\mathcal{M}) = \mathcal{M}_X$. This implies that it only depends on the horizontal variable ξ :

$$\tilde{\mathcal{L}}(p_{\xi, \xi'}) = (\mathcal{L} \circ X_*)(p_{\xi, \xi'}) = \mathcal{L}(X_*(p_{\xi, \xi'})) = \mathcal{L}(p_\xi).$$

This implies $\nabla_{\xi'} \tilde{\mathcal{L}} = 0$ and $\nabla_\xi \tilde{\mathcal{L}} = \nabla_\xi \mathcal{L}$. With (74), we obtain

$$\text{grad}_{\xi, \xi'}^{\tilde{\mathcal{M}}^I}(\mathcal{L} \circ X_*) = \begin{pmatrix} \bar{G}^{\mathcal{H}^+}(\xi) \nabla_\xi \mathcal{L} \\ 0 \end{pmatrix}. \quad (75)$$

This is a confirmation of our more general result that the natural gradient on a cylindrical model, here $\tilde{\mathcal{M}}^I$, is equivalent to the natural gradient on the projected model $X_*(\tilde{\mathcal{M}}^I) = \mathcal{M}_X$ (see Theorem 5). However, Eq. (75) does not imply any simplification of the problem, because $\bar{G}^{\mathcal{H}^+}(\xi)$ equals the original Fisher information matrix defined on the projected model $\mathcal{M}_X$ which does not necessarily have a block structure (see Eq. (72)). Assuming that the Fisher information matrix $G(\xi)$ on the full model $\mathcal{M}$ has a block structure, we can try to exploit this structure within its product extension $\tilde{\mathcal{M}}^I$. For this, note that the tangent vectors (63) of $\mathcal{M}$ in p_ξ can be expressed as

$$\partial_i(\xi) = \partial_i^{\mathcal{H}}(\xi, \xi) + \partial_i^{\mathcal{V}}(\xi, \xi) \in T_\xi \mathcal{M}, \quad i = 1, \dots, d. \quad (76)$$

This implies $G(\xi) = \bar{G}^{\mathcal{H}}(\xi) + G^{\mathcal{V}}(\xi, \xi)$, and therefore, according to (75), we have to invert $\bar{G}^{\mathcal{H}}(\xi) = G(\xi) - G^{\mathcal{V}}(\xi, \xi)$, a difference of two matrices where the first one has a block structure and the second one does not. This shows that the block structure of $G(\xi)$ is not sufficient for the simplification of the problem. In what follows, we modify the product extension $\tilde{\mathcal{M}}^I$ and open up the possibility for simplification. The main idea here parallels the idea of introducing a recognition model, in addition to the generative model, as we did in the context of the wake-sleep algorithm in Sect. 2.2.

The product extension II We now generalise the first product extension and replace (67) by $p_{\xi, \eta} = p(\cdot; \xi, \eta)$ where

$$\begin{aligned} \sum_z p(z; \xi, \eta) \delta^z &:= \sum_x \sum_{z \in \mathcal{Z}_x} [p(z; \xi) + p(x; \xi)(q(z|x; \eta) - p(z|x; \xi))] \delta^z \\ &= \sum_x \sum_{z \in \mathcal{Z}_x} p(x; \xi) q(z|x; \eta) \delta^z, \end{aligned}$$

denoting by q the elements of a model $\mathcal{Q}_{Z|X}$ that is properly parametrised by $\eta = (\eta_1, \dots, \eta_{d'}) \in \mathcal{H} \subseteq \mathbb{R}^{d'}$ and contains the model $\mathcal{M}_{Z|X}$. That is, for each ξ there is an $\eta = \eta(\xi)$ such that $p(z|x; \xi) = q(z|x; \eta)$. This is closely related to the recognition model discussed in Sect. 2.2. For $\eta \in \mathcal{H}$, the tangent space in $T_\eta \mathcal{Q}_{Z|X}$ is spanned by

$$\bar{\partial}_i^{\mathcal{V}}(\eta) := \sum_{x, z} q(z|x; \eta) \frac{\partial \ln q(z|x; \cdot)}{\partial \eta_i}(\eta) \delta^z \otimes e_x, \quad i = 1, \dots, d'. \quad (77)$$

Consider a pair $(\xi, \eta) \in \mathcal{E} \times \mathcal{H}$ so that all points associated with it are non-singular points of the respective models. For the horizontal and vertical vectors we obtain, analogous to (68) and (70),

$$\partial_i^{\mathcal{H}}(\xi, \eta) = \sum_x p(x; \xi) \frac{\partial \ln p(x; \cdot)}{\partial \xi_i}(\xi) \left(\sum_{z \in \mathcal{Z}_x} q(z|x; \eta) \delta^z \right), \quad i = 1, \dots, d, \quad (78)$$

and

$$\partial_i^{\mathcal{V}}(\xi, \eta) = \sum_x \sum_{z \in \mathcal{Z}_x} p(x; \xi) q(z|x; \eta) \frac{\partial \ln q(z|x; \cdot)}{\partial \eta_i}(\eta) \delta^z, \quad i = 1, \dots, d'. \quad (79)$$

The correspondence (69) of horizontal vectors translates to

$$\partial_i^{\mathcal{H}}(\xi, \eta) \longleftrightarrow \bar{\partial}_i^{\mathcal{H}}(\xi), \quad i = 1, \dots, d, \quad (80)$$

this time by mapping δ^x to $\sum_{z \in \mathcal{Z}_x} q(z|x; \eta) \delta^z$. Furthermore, we obtain the generalisation of (71) as

$$\partial_i^{\mathcal{V}}(\xi, \eta) \longleftrightarrow \bar{\partial}_i^{\mathcal{V}}(\eta), \quad i = 1, \dots, d', \quad (81)$$

by mapping $\delta^z \otimes e_x$ to $p(x; \xi) \delta^z$. The situation is illustrated in Fig. 7.

We now consider the Fisher–Rao metric on $\widetilde{\mathcal{M}}^{II} \subseteq \mathcal{P}(\mathcal{Z})$ in $p_{\xi, \eta}$. It follows again from Proposition 2 below that $\langle \partial_i^{\mathcal{H}}(\xi, \eta), \partial_j^{\mathcal{V}}(\xi, \eta) \rangle_{\xi, \eta} = 0$ for all i, j , where $\langle \cdot, \cdot \rangle_{\xi, \eta}$ denotes the Fisher–Rao metric in $p_{\xi, \eta}$. For the inner products of the horizontal and the vertical vectors, respectively, we obtain

$$\begin{aligned} g_{ij}^{\mathcal{H}}(\xi, \eta) &= \left\langle \partial_i^{\mathcal{H}}(\xi, \eta), \partial_j^{\mathcal{H}}(\xi, \eta) \right\rangle_{\xi, \eta} \\ &= \sum_x p(x; \xi) \frac{\partial \ln p(x; \cdot)}{\partial \xi_i}(\xi) \frac{\partial \ln p(x; \cdot)}{\partial \xi_j}(\xi) = \bar{g}_{ij}^{\mathcal{H}}(\xi), \end{aligned} \quad (82)$$

and

$$\begin{aligned} g_{ij}^{\mathcal{V}}(\xi, \eta) &= \left\langle \partial_i^{\mathcal{V}}(\xi, \eta), \partial_j^{\mathcal{V}}(\xi, \eta) \right\rangle_{\xi, \eta} \\ &= \sum_x p(x; \xi) \sum_{z \in \mathcal{Z}_x} \frac{\partial \ln q(z|x; \cdot)}{\partial \eta_i}(\eta) \frac{\partial \ln q(z|x; \cdot)}{\partial \eta_j}(\eta). \end{aligned} \quad (83)$$

The gradient of a function $\mathcal{L}$ on $\mathcal{M}_X$ is given in terms of (75), the formula that we already obtained for the previous product extension, where we have to replace ξ' by

Fig. 7 Extension of $\mathcal{M}$ to the cylindrical model $\widetilde{\mathcal{M}}^{II}$, with the corresponding tangent vectors

η and $\widetilde{\mathcal{M}}^I$ by $\widetilde{\mathcal{M}}^{II}$. However, with the second product extension we can choose the model $\mathcal{Q}_{Z|X}$ to be larger than $\mathcal{M}_{Z|X}$. This provides a way to simplify $\overline{\mathbf{G}}^{\mathcal{H}}(\xi)$ in (75). In order to be more explicit, consider the parametrisation

$$\xi \mapsto \sum_z p(z; \xi, \eta(\xi)) \delta^z \quad (84)$$

of $\mathcal{M}$ which is naturally embedded in $\widetilde{\mathcal{M}}^{II}$. For the tangent vectors we now obtain

$$\begin{aligned} \partial_i(\xi) &= \sum_z p(z; \xi, \eta(\xi)) \frac{\partial \ln p(z; \cdot, \eta(\cdot))}{\partial \xi_i}(\xi) \delta^z \\ &= \sum_x \sum_{z \in Z_x} p(x; \xi) q(z|x; \eta(\xi)) \left[\frac{\partial \ln p(x; \cdot)}{\partial \xi_i}(\xi) + \frac{\partial \ln q(z|x; \eta(\cdot))}{\partial \xi_i}(\xi) \right] \delta^z \\ &= \sum_x \sum_{z \in Z_x} p(x; \xi) q(z|x; \eta(\xi)) \frac{\partial \ln p(x; \cdot)}{\partial \xi_i}(\xi) \delta^z \\ &\quad + \sum_x \sum_{z \in Z_x} p(x; \xi) q(z|x; \eta(\xi)) \sum_k \frac{\partial \ln q(z|x; \cdot)}{\partial \eta_k}(\eta(\xi)) \frac{\partial \eta_k}{\partial \xi_i}(\xi) \delta^z \\ &\quad \text{(by the chain rule)} \\ &= \partial_i^{\mathcal{H}}(\xi, \eta(\xi)) + \sum_k \frac{\partial \eta_k}{\partial \xi_i}(\xi) \partial_k^{\mathcal{V}}(\xi, \eta(\xi)). \end{aligned}$$

This derivation generalises the Eq. (76). For the Fisher information matrix $G(\xi) = (g_{ij}(\xi))_{1 \leq i, j \leq d}$ we obtain

$$\begin{aligned} g_{ij}(\xi) &= \langle \partial_i(\xi), \partial_j(\xi) \rangle_{\xi} \\ &= \underbrace{\left\langle \partial_i^{\mathcal{H}}(\xi, \eta(\xi)), \partial_j^{\mathcal{H}}(\xi, \eta(\xi)) \right\rangle_{\xi, \eta(\xi)}}_{= \bar{g}_{ij}^{\mathcal{H}}(\xi)} \\ &\quad + \sum_{k,l} \frac{\partial \eta_k}{\partial \xi_i}(\xi) \frac{\partial \eta_l}{\partial \xi_j}(\xi) \underbrace{\left\langle \partial_k^{\mathcal{V}}(\xi, \eta(\xi)), \partial_l^{\mathcal{V}}(\xi, \eta(\xi)) \right\rangle_{\xi, \eta(\xi)}}_{= g_{kl}^{\mathcal{V}}(\xi, \eta(\xi))} \\ &= \bar{g}_{ij}^{\mathcal{H}}(\xi) + \sum_{k,l} \frac{\partial \eta_k}{\partial \xi_i}(\xi) \frac{\partial \eta_l}{\partial \xi_j}(\xi) g_{kl}^{\mathcal{V}}(\xi, \eta(\xi)). \end{aligned}$$

Thus, we can insert

$$\bar{g}_{ij}^{\mathcal{H}}(\xi) = g_{ij}(\xi) - \sum_{k,l} \frac{\partial \eta_k}{\partial \xi_i}(\xi) \frac{\partial \eta_l}{\partial \xi_j}(\xi) g_{kl}^{\mathcal{V}}(\xi, \eta(\xi)) \quad (85)$$

into Eq. (75). At first sight, this does not appear to simplify the problem. However, as we will outline in the next section, it suggests conditions for both, the generative model as well as the recognition model, that would be sufficient for a simplification of $\bar{G}^{\mathcal{H}}(\xi)$. These conditions involve locality properties, as we studied in Sect. 2, but also an appropriate coupling between the two models.

We now prove that the second product extension, and thereby also the first one, are indeed cylindrical extensions of $\mathcal{M}$.

Proposition 2 *The product extensions $\widetilde{\mathcal{M}}^{II}$ and, as a special case, $\widetilde{\mathcal{M}}^I$ are cylindrical extensions of $\mathcal{M}$. More precisely, we have*

$$T_{\xi, \eta} \widetilde{\mathcal{M}}^{II} \cap \mathcal{H}_{\xi, \eta} = \text{span} \left\{ \partial_i^{\mathcal{H}}(\xi, \eta) : i = 1, \dots, d \right\} \quad (86)$$

$$T_{\xi, \eta} \widetilde{\mathcal{M}}^{II} \cap \mathcal{V}_{\xi, \eta} = \text{span} \left\{ \partial_i^{\mathcal{V}}(\xi, \eta) : i = 1, \dots, d' \right\} \quad (87)$$

$$T_{\xi, \eta} \widetilde{\mathcal{M}}^{II} = \left(T_{\xi, \eta} \widetilde{\mathcal{M}}^{II} \cap \mathcal{H}_{\xi, \eta} \right) + \left(T_{\xi, \eta} \widetilde{\mathcal{M}}^{II} \cap \mathcal{V}_{\xi, \eta} \right). \quad (88)$$

Proof We have to verify the properties (a), (b), and (c) in (61).

- (a) We have assumed that for each ξ there exists an $\eta = \eta(\xi)$ such that $p(z|x; \xi) = q(z|x; \eta(\xi))$. This implies that each distribution $p_{\xi} \in \mathcal{M}$ is also contained in $\widetilde{\mathcal{M}}^{II}$:

$$p(z; \xi) = p(x; \xi) p(z|x; \xi) = p(x; \xi) q(z|x; \eta(\xi)) = p(z; \xi, \eta(\xi)).$$

- (b) Clearly, from (a) we obtain $X_*(\mathcal{M}) \subseteq X_*(\widetilde{\mathcal{M}}^{II})$. To prove the opposite inclusion, we consider a point $p_{\xi, \eta} \in \widetilde{\mathcal{M}}^{II}$ and show that the point $p_\xi \in \mathcal{M}$ has the same X_* -projection:

$$\begin{aligned} X_*(p_{\xi, \eta}) &= X_* \left(\sum_x \sum_{z \in Z_x} p(x; \xi) q(z|x; \eta) \delta^z \right) \\ &= \sum_x \left(\sum_{z \in Z_x} p(x; \xi) q(z|x; \eta) \right) \delta^x \\ &= \sum_x p(x; \xi) \delta^x \\ &= \sum_x \left(\sum_{z \in Z_x} p(x; \xi) q(z|x; \eta(\xi)) \right) \delta^x \\ &= \sum_x \left(\sum_{z \in Z_x} p(x; \xi) p(z|x; \xi) \right) \delta^x \\ &= \sum_x \left(\sum_{z \in Z_x} p(z; \xi) \right) \delta^x = X_* \left(\sum_z p(z; \xi) \delta^z \right) = X_*(p_\xi). \end{aligned}$$

- (c) We have

$$\mathcal{H}_{\xi, \eta} := \left\{ \tilde{A} = \sum_x A(x) \sum_{z \in Z_x} q(z|x; \eta) \delta^z : \sum_x A(x) = 0 \right\}$$

with the orthogonal complement

$$\mathcal{V}_{\xi, \eta} := \left\{ \sum_z A(z) \delta^z : \sum_{z \in Z_x} A(z) = 0 \text{ for all } x \right\}.$$

We first show that the horizontal vectors

$$\partial_i^{\mathcal{H}}(\xi, \eta) = \sum_x p(x; \xi) \frac{\partial \ln p(x; \cdot)}{\partial \xi_i}(\xi) \sum_{z \in Z_x} q(z|x; \eta) \delta^z$$

are contained in $\mathcal{H}_{\xi, \eta}$. To this end, we set $A(x) = p(x; \xi) \frac{\partial \ln p(x; \cdot)}{\partial \xi_i}(\xi)$ and verify

$$\begin{aligned} \sum_x A(x) &= \sum_x p(x; \xi) \frac{\partial \ln p(x; \cdot)}{\partial \xi_i}(\xi) \\ &= \sum_x \frac{\partial p(x; \cdot)}{\partial \xi_i}(\xi) = \frac{\partial}{\partial \xi_i} \sum_x p(x; \cdot) \Big|_{\xi} \\ &= 0. \end{aligned}$$

Now we show that the vertical vectors

$$\partial_i^{\mathcal{V}}(\xi, \eta) = \sum_x \sum_{z \in \mathcal{Z}_x} p(x; \xi) q(z|x; \eta) \frac{\partial \ln q(z|x; \cdot)}{\partial \eta_i}(\eta) \delta^z$$

are contained in $\mathcal{V}_{\xi, \eta}$. We set $A(z) := p(X(z); \xi) q(z|X(z); \eta) \frac{\partial \ln q(z|X(z); \cdot)}{\partial \eta_i}(\eta)$ and verify

$$\begin{aligned} \sum_z A(z) &= \sum_x \sum_{z \in \mathcal{Z}_x} p(x; \xi) q(z|x; \eta) \frac{\partial \ln q(z|x; \cdot)}{\partial \eta_i}(\eta) \\ &= \sum_x \sum_{z \in \mathcal{Z}_x} p(x; \xi) \frac{\partial q(z|x; \cdot)}{\partial \eta_i}(\eta) \\ &= \sum_x p(x; \xi) \frac{\partial}{\partial \eta_i} \sum_{z \in \mathcal{Z}_x} q(z|x; \cdot) \Big|_{\eta} \\ &= 0. \end{aligned}$$

In conclusion, we have

$$\begin{aligned} T_{\xi, \eta} \widetilde{\mathcal{M}}^{II} &= \text{span} \left\{ \partial_i^{\mathcal{H}}(\xi, \eta) : i = 1, \dots, d \right\} + \text{span} \left\{ \partial_i^{\mathcal{V}}(\xi, \eta) : i = 1, \dots, d' \right\} \\ &\subseteq \left(T_{\xi, \eta} \widetilde{\mathcal{M}}^{II} \cap \mathcal{H}_{\xi, \eta} \right) + \left(T_{\xi, \eta} \widetilde{\mathcal{M}}^{II} \cap \mathcal{V}_{\xi, \eta} \right) \\ &\subseteq T_{\xi, \eta} \widetilde{\mathcal{M}}^{II}, \end{aligned}$$

which proves the equalities (86), (87), and (88). $\square$

4 Conclusions: A natural gradient perspective of the wake–sleep algorithm

Information geometry provides two natural geometries associated with a learning system that has visible units V and hidden units H . Typically, the system is given in terms of a model $\mathcal{M}$ of probability distributions of global states of the full system,

$\mathcal{P}_{V,H}$, but the objective function $\mathcal{L}$ only depends on the probability distribution of the visible states, giving rise to a projected model $\mathcal{M}_V \subseteq \mathcal{P}_V$. Both geometric objects, $\mathcal{M}$ and $\mathcal{M}_V$, carry a natural geometry inherited from the respective ambient space. In Sect. 2 we studied various locality properties of the natural gradient based on the first geometry, thereby assuming a factorisation of the elements of $\mathcal{M}$ according to a directed acyclic graph. These properties simplify the Fisher information matrix for $\mathcal{M}$ and allow us to apply the natural gradient method to deep networks. The second geometry, the geometry of $\mathcal{M}_V$, was studied in Sect. 3 where we took a somewhat more general perspective. In what follows, we restate the general problem of comparing the two mentioned geometries within that perspective and summarise the corresponding results.

Consider a model $\mathcal{S}$ in the set $\mathcal{P}(X)$ of probability distributions on a finite set X , and a smooth function $\mathcal{L} : \mathcal{P}(X) \rightarrow \mathbb{R}$. The task is to optimise $\mathcal{L}$ on $\mathcal{S}$ in terms of the natural gradient $\text{grad}^{\mathcal{S}} \mathcal{L}$. With no further assumptions this can be a very difficult problem. Typically, however, $\mathcal{S}$ is obtained as the image of a simpler model $\mathcal{M}$ of probability distributions on a larger set Z , $\mathcal{P}(Z)$. More precisely, we consider a surjective map $X : Z \rightarrow X$, and the corresponding push-forward map $X_* : \mathcal{P}(Z) \rightarrow \mathcal{P}(X)$ of probability measures. The model $\mathcal{S}$ is then nothing but the X_* -image of $\mathcal{M}$, that is $\mathcal{S} = \mathcal{M}_X = X_*(\mathcal{M})$. Now, instead of optimising $\mathcal{L}$ on $\mathcal{M}_X$, we can optimise $\mathcal{L} \circ X_*$ on $\mathcal{M}$ and aim to simplify the problem by exploiting the structure of $\mathcal{M}$. This works to some extent. Even though the two problems are closely related, the corresponding gradient fields, $dX_*(\text{grad}^{\mathcal{M}}(\mathcal{L} \circ X_*))$ and $\text{grad}^{\mathcal{M}_X} \mathcal{L}$, typically differ from each other. Thus, the optimisation of $\mathcal{L}$ on $\mathcal{M}_X$, based on the Fisher–Rao metric on $\mathcal{P}(X)$, and the optimisation of $\mathcal{L} \circ X_*$ on $\mathcal{M}$, based on the Fisher–Rao metric on $\mathcal{P}(Z)$, are not equivalent. We can try to improve the situation by replacing the Fisher–Rao metric on $\mathcal{M}$ and $\mathcal{M}_X$, respectively, by different Riemannian metrics. While this might be a reasonable approach for the simplification of the problem, from the information-geometric perspective, the Fisher–Rao metric is the most natural one, which is the reason for referring to the Fisher–Rao gradient as the *natural gradient*. This is directly linked to the invariance of gradients, as we have highlighted in this article. If we request invariance of the gradients for all coarse grainings $X : Z \rightarrow X$, all models $\mathcal{M} \subseteq \mathcal{P}(Z)$ from a particular class, and all smooth functions $\mathcal{L} : \mathcal{M}_X \rightarrow \mathbb{R}$, by Chentsov’s classical characterisation theorem, we *have* to impose the Fisher–Rao metric on the individual models (see Theorem 5). Even then, the invariance of gradients is satisfied only if the model is cylindrical in the sense of Definition 1. Given a model $\mathcal{M}$ that is not cylindrical, we have proposed cylindrical extensions $\widetilde{\mathcal{M}}$ which contain $\mathcal{M}$. The natural gradient of $\mathcal{L}$ on $\mathcal{M}_X$ is then equivalent to the natural gradient of $\mathcal{L} \circ X_*$ on such an extension $\widetilde{\mathcal{M}}$.

As an outlook, we want to touch upon the following two related problems:

1. Can we exploit the simplicity of the original model $\mathcal{M}$ in order to simplify the optimisation on $\widetilde{\mathcal{M}}$?
2. The original model $\mathcal{M}$ is associated with some network. What kind of network can we associate with the extended model $\widetilde{\mathcal{M}}$?

We want to briefly address these problems within the context of Section 2, where $X = X_V$, $Z = X_V \times X_H$, and $X = X_V : (v, h) \mapsto v$. As the cylindrical extension $\widetilde{\mathcal{M}}^{II}$

suggests, it can be associated with the addition of a recognition model $\mathcal{Q}_{H|V}$, assuming that $\mathcal{M}$ is a generative model. If both models are parametrised by (7) and (21), respectively, then the corresponding Fisher information matrices simplify as stated in Theorem 1. They both have a block structure where each block corresponds to one unit. Outside of these blocks, the matrices are filled with zeros. Being more precise, we consider all parameters that correspond to unit r , the parameters $\xi_r = (\xi_{(r;1)}, \dots, \xi_{(r;d_r)})$ of the generative model $\mathcal{M}$, and the parameters $\eta_r = (\eta_{(r;1)}, \dots, \eta_{(r;d'_r)})$ of the recognition model $\mathcal{Q}_{H|V}$. With (85) we then obtain

$$\begin{aligned} \bar{g}_{(r;i)(s;j)}^{\mathcal{H}}(\xi) \\ = g_{(r;i)(s;j)}(\xi) - \sum_{t,u} \sum_{(t;k),(u;l)} \frac{\partial \eta_{(t;k)}}{\partial \xi_{(r;i)}}(\xi) \frac{\partial \eta_{(u;l)}}{\partial \xi_{(s;j)}}(\xi) g_{(t;k)(u;l)}^{\mathcal{V}}(\xi, \eta(\xi)). \end{aligned} \quad (89)$$

We know that $g_{(r;i)(s;j)}(\xi) = 0$ if $r \neq s$ and $g_{(t;k)(u;l)}^{\mathcal{V}}(\xi, \eta(\xi)) = 0$ if $t \neq u$. With the latter property, the sum on the RHS of (89) reduces to

$$\sum_t \sum_{(t;k),(t;l)} \frac{\partial \eta_{(t;k)}}{\partial \xi_{(r;i)}}(\xi) \frac{\partial \eta_{(t;l)}}{\partial \xi_{(s;j)}}(\xi) g_{(t;k)(t;l)}^{\mathcal{V}}(\xi, \eta(\xi)). \quad (90)$$

If all partial derivatives $\partial \eta_{(t;k)}/\partial \xi_{(r;i)}(\xi)$ are local in the sense that they vanish whenever $t \neq r$, then the matrix $\bar{G}^{\mathcal{H}}(\xi)$ inherits the block structure of the matrices $G(\xi)$ and $G^{\mathcal{V}}(\xi, \eta(\xi))$. However, this is typically not the case and suggests an additional coupling between the generative model and the recognition model. Without that coupling, the partial derivatives in (90) will “overwrite” the block structure of the matrix $G(\xi)$, leading to a non-local matrix $\bar{G}^{\mathcal{H}}(\xi)$ with $\bar{g}_{(r;i)(s;j)}^{\mathcal{H}}(\xi) \neq 0$ even if $r \neq s$. The degree of non-locality will depend on the specific properties of the partial derivatives $\partial \eta_{(t;k)}/\partial \xi_{(r;i)}(\xi)$.

We conclude this article by revisiting the wake–sleep algorithm of Sect. 2.2. Let us assume that (89) and (90) imply a sufficient simplification so that a natural gradient step in $\tilde{\mathcal{M}}^{II}$ can be made. This will update the generation parameters, say from ξ to $\xi + \Delta\xi$, and leave the recognition parameters η unchanged. (The situation is illustrated in Fig. 8.) Such an update corresponds to a natural gradient version of the wake step. The resulting point $(\xi + \Delta\xi, \eta)$ in $\tilde{\mathcal{M}}^{II}$ will typically be outside of $\mathcal{M}$. As the simplification through (89) and (90) only holds on $\mathcal{M}$, we have to update the recognition parameters, say from η to $\eta + \Delta\eta$, so that the resulting point $(\xi + \Delta\xi, \eta + \Delta\eta)$ is again in $\mathcal{M}$. This sleep step will ensure that the next update of the generation parameters benefits from the simplicity of the Fisher information matrix.

Note that it is irrelevant how we get back to $\mathcal{M}$ within the sleep step, as far as we do not change the generation parameters. Also, it might be required to apply several sleep steps until we get back to $\mathcal{M}$, which highlights the asymmetry of time scales of the two phases. This asymmetric version has been outlined and discussed in the context of the *em*-algorithm in [18]. The overall wake–sleep step will typically not follow the gradient of an objective function on $\tilde{\mathcal{M}}^{II}$. However, this is not the aim here. The prime process is the process in ξ which parametrises $\mathcal{M}_V$. Effectively, the outlined version

Fig. 8 Illustration of the generalised wake–sleep algorithm, taking place on the cylindrical extension $\widetilde{\mathcal{M}}^{II}$ of $\mathcal{M}$

of the wake-sleep algorithm will follow the natural gradient of the objective function with respect to the geometry of $\mathcal{M}_V$. The natural wake-sleep algorithm with respect to the geometry of $\mathcal{M}$ has been recently studied in [31].

In Sect. 2.2 we introduced the recognition model as an auxiliary model for the evaluation of the gradient with respect to ξ . This work reveals another role of the recognition model in the context of the natural gradient method. It allows us to define an extension of the original model $\mathcal{M}$ so that we can effectively apply the natural gradient method on the projected model $\mathcal{M}_V$ within the context of deep learning. The presented results suggest criteria for the coupling between the generative model and recognition model that would ensure the locality of the natural gradient on $\mathcal{M}_V$. These criteria involve, on the one hand, the structure of the underlying networks (see Example 2) and, on the other hand, the parametrisation of the models. However, in this article they are formulated as sufficient conditions for the simplification of the natural gradient, based on theoretical results. The derivation of explicit design principles for correspondingly coupled models requires further studies.

Acknowledgements The author is grateful for valuable discussions with Luigi Malagò, Riccardo Volpi, and Csongor-Huba Várady. He acknowledges the support of the Deutsche Forschungsgemeinschaft Priority Programme “The Active Self” (SPP 2134).

Funding Open Access funding enabled and organized by Projekt DEAL.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence,

and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Appendix

Moore–Penrose inverse and gradients

We consider a model $\mathcal{M}$, with a parametrisation $\mathbb{R}^d \supseteq \mathcal{E} \rightarrow \mathcal{M}, \xi \mapsto p_\xi$. For a non-singular point $p_\xi \in \mathcal{M}$, we assume that the tangent space in $p_\xi, T_\xi \mathcal{M}$, is spanned by the vectors $\partial_i(\xi) := \frac{\partial}{\partial \xi_i} p_\xi, i = 1, \dots, d$. Note that we do not assume that these vectors are independent. Now consider a function $\mathcal{L} : \mathcal{M} \rightarrow \mathbb{R}$ that is smooth in p_ξ , and its differential

$$d\mathcal{L}_\xi : T_\xi \mathcal{M} \rightarrow \mathbb{R}, \quad A \mapsto d\mathcal{L}_\xi(A) = \frac{\partial \mathcal{L}}{\partial A}(\xi).$$

This is a linear form on $T_\xi \mathcal{M}$. With an inner product g_ξ on $T_\xi \mathcal{M}$ we can identify $d\mathcal{L}_\xi$ with a vector $\text{grad}_\xi^\mathcal{M} \mathcal{L} \in T_\xi \mathcal{M}$, which points in the direction of maximal infinitesimal increase of $\mathcal{L}$ in ξ . It is uniquely characterised by the equation

$$g_\xi(\text{grad}_\xi^\mathcal{M} \mathcal{L}, A) = d\mathcal{L}_\xi(A), \quad A \in T_\xi \mathcal{M}. \quad (91)$$

Let us now express the gradient in terms of the parameters ξ . Clearly, it has a representation

$$\text{grad}_\xi^\mathcal{M} \mathcal{L} = \sum_{i=1}^d x^i(\xi) \partial_i(\xi). \quad (92)$$

Note that this representation of the gradient in terms of the coefficients $x(\xi) = (x^1(\xi), \dots, x^d(\xi))$ is not necessarily unique (due to the fact that the vectors $\partial_i(\xi), i = 1, \dots, d$, need not be independent). We insert the RHS of (92) and $A = \partial_j(\xi)$ into (91) and obtain

$$\sum_{i=1}^d x^i(\xi) g_{ij}(\xi) = \frac{\partial \mathcal{L}}{\partial \xi_j}(\xi), \quad j = 1, \dots, d, \quad (93)$$

where $g_{ij}(\xi) = g_\xi(\partial_i(\xi), \partial_j(\xi))$. This can be written in matrix notation as

$$G(\xi) x(\xi) = \nabla_\xi \mathcal{L}. \quad (94)$$

Any coefficient vector $x(\xi) \in \mathbb{R}^d$ that satisfies (94) will provide an equally valid representation of the gradient in terms of the tangent vectors $\partial_i(\xi)$. Furthermore, we

know that there is at least one solution $x(\xi)$ that represents the gradient. In the case where $G(\xi)$ is of maximal rank this solution is unique and we can simply apply the inverse of $G(\xi)$ in order to obtain the coefficients of the gradient as $x(\xi) = G^{-1}(\xi) \nabla_{\xi} \mathcal{L}$. This is the usual case when we have a local (diffeomorphic) coordinate system around the point p_{ξ} . Even though we interpret a parametrisation of a model as a coordinate system, the number of parameters often exceeds the dimension of the model. In these cases, the matrix $G(\xi)$ will not be of maximal rank so that we have a non-trivial kernel $\ker G(\xi)$. We can always add to a solution $x(\xi)$ of (94) a vector $y(\xi)$ from that kernel and obtain another solution $x(\xi) + y(\xi)$. The affine space $\mathcal{A}(\xi) = x(\xi) + \ker G(\xi) \subseteq \mathbb{R}^d$ of solutions describes all possible representations of the gradient in terms of $\partial_1(\xi), \dots, \partial_d(\xi)$. They are all equally adequate for describing a learning process that takes place in $\mathcal{M}$. However, from the perspective of linear algebra there is a natural choice, the element in the affine solution space $\mathcal{A}(\xi)$ that is orthogonal to $\ker G(\xi)$ (with respect to the canonical inner product in $\mathbb{R}^d$). This defines the Moore–Penrose inverse $G^+(\xi)$, also called pseudoinverse. In this paper, we were concerned with a number of simplifications of the natural gradient. One simplification was expressed in terms of a block diagonal structure of the Fisher information matrix. For the representation of the natural gradient, we evaluated the pseudoinverse of that block diagonal matrix based on the following simple observation (see, e.g., [11] for more general results related to the pseudoinverse of a block matrix):

$$\begin{pmatrix} G_1 & & 0 \\ & \ddots & \\ 0 & & G_{|N|} \end{pmatrix}^+ = \begin{pmatrix} G_1^+ & & 0 \\ & \ddots & \\ 0 & & G_{|N|}^+ \end{pmatrix}. \quad (95)$$

How natural is the Moore–Penrose inverse? There are two perspectives here. On the one hand, $G^+(\xi) \nabla_{\xi} \mathcal{L}$ is natural in the sense that it represents an object, $\text{grad}_{\xi}^{\mathcal{M}} \mathcal{L}$, that is independent of the parametrisation. On the other hand, the inner product used for the definition of $G^+(\xi)$ is the canonical inner product in $\mathbb{R}^d$ which does not have to be at all related to the metric g_{ξ} . In this article, we have chosen the Moore–Penrose inverse as one possible extension of the usual inverse to overparametrised models, following previous proposals (see, e.g., [29]). Note, however, that there are also other possibilities for such an extension. We have some flexibility here which might allow us to further simplify the representation of the natural gradient in terms of a particular choice of the parametrisation.

Gibbs sampling

In this section we outline the Gibbs sampling method for the approximation of the derivative (11). By holding the configuration x_V constant, we can sample from $p(x_H | x_V; \xi)$ by randomly selecting a node $s \in H$, and then updating the state of that node according to $p(x_s | x_{H \setminus s}, x_V; \xi)$. After this update we repeat choosing a node and updating its state. This will generate, after many repetitions, p_{ξ}^* -typical patterns. The conditional distribution is simple because, due to the local Markov property, it satisfies $p(x_s | x_{H \setminus s}, x_V; \xi) = p(x_s | x_{bl(s)}; \xi)$, where $bl(s)$ denotes the Markov blanket

Fig. 9 Illustration of the Markov blanket $bl(s)$ of a node s . Its elements are highlighted by thick circles

of s (see Fig. 9). It is defined as

$$bl(s) := pa(s) \cup ch(s) \cup \bigcup_{j \in ch(s)} (pa(j) \setminus s).$$

Note that, in general, the Markov blanket is larger than the parent set and $p(x_s | x_{bl(s)}; \xi)$ differs from $k^s(x_s | x_{pa(s)}; \xi_s)$. More precisely, for $s \in H$, we have

$$\begin{aligned} p(x_s | x_{bl(s)}; \xi) &= p(x_s | x_V, x_{H \setminus s}; \xi) \\ &= \frac{p(x_V, x_{H \setminus s}, x_s; \xi)}{\sum_{x'_s} p(x_V, x_{H \setminus s}, x'_s; \xi)} \\ &= \frac{\prod_{i \in N} k^i(x_i | x_{pa(i)}; \xi_i)}{\sum_{x'_s} \prod_{i \in N} k^i(x'_i | x_{pa(i)}; \xi_i)} \\ &= \frac{k^s(x_s | x_{pa(s)}; \xi_s) \prod_{i \in ch(s)} k^i(x_i | x_{pa(i)}; \xi_i)}{\sum_{x'_s} k^s(x'_s | x_{pa(s)}; \xi_s) \prod_{i \in ch(s)} k^i(x_i | x_{pa(i) \setminus s}, x'_s; \xi_i)}. \quad (96) \end{aligned}$$

Even though we cannot use the local kernels $k^s(x_s | x_{pa(s)}; \xi_s)$ as a generative model for sampling from p_ξ^* , the kernels $p(x_s | x_{bl(s)}; \xi)$, which are used for Gibbs sampling, are still local in the sense that they only depend on the Markov blanket of s (see Fig. 9).

We exemplify the derivations of this section in the context of binary neural networks, thereby extending Example 1.

Example 7 (*Neural networks (III)*) In order to evaluate the derivatives (15) and (16) in Example 1, we have to sample from p_{ξ}^* . We use Gibbs sampling based on the expression (96) which, for binary state sets $\{-1, +1\}$, reduces to

$$p(x_s | x_V, x_{H \setminus s}; \xi) = \frac{1}{1 + \frac{k^s(-x_s | x_{pa(s)}; \xi_s)}{k^s(x_s | x_{pa(s)}; \xi_s)} \prod_{i \in ch(s)} \frac{k^i(x_i | x_{pa(i) \setminus s}, -x_s; \xi_i)}{k^i(x_i | x_{pa(i) \setminus s}, x_s; \xi_i)}}.$$

Let us analyse the term in the denominator. Using the update rule (12), simple calculations yield

$$\begin{aligned} & \frac{k^s(-x_s | x_{pa(s)}; \xi_s)}{k^s(x_s | x_{pa(s)}; \xi_s)} \prod_{i \in ch(s)} \frac{k^i(x_i | x_{pa(i) \setminus s}, -x_s; \xi_i)}{k^i(x_i | x_{pa(i) \setminus s}, x_s; \xi_i)} \\ &= \frac{e^{-\frac{1}{2}x_s h_s(x_{pa(s)})}}{e^{\frac{1}{2}x_s h_s(x_{pa(s)})}} \\ & \quad \times \prod_{i \in ch(s)} \frac{e^{\frac{1}{2}x_i(h_i(x_{pa(i)}) - 2w_{si}x_s)}}{e^{\frac{1}{2}x_i(h_i(x_{pa(i)}) - 2w_{si}x_s)} + e^{-\frac{1}{2}x_i(h_i(x_{pa(i)}) - 2w_{si}x_s)}} \\ & \quad \times \frac{e^{\frac{1}{2}x_i h_i(x_{pa(i)})} + e^{-\frac{1}{2}x_i h_i(x_{pa(i)})}}{e^{\frac{1}{2}x_i h_i(x_{pa(i)})}} \\ &= e^{-x_s h_s(x_{pa(s)})} \prod_{i \in ch(s)} \frac{e^{-w_{si}x_s x_i} \left(e^{\frac{1}{2}x_i h_i(x_{pa(i)})} + e^{-\frac{1}{2}x_i h_i(x_{pa(i)})} \right)}{e^{-w_{si}x_s x_i} e^{\frac{1}{2}x_i h_i(x_{pa(i)})} + e^{w_{si}x_s x_i} e^{-\frac{1}{2}x_i h_i(x_{pa(i)})}} \\ &= e^{-x_s h_s(x_{pa(s)})} \prod_{i \in ch(s)} \frac{e^{\frac{1}{2}x_i h_i(x_{pa(i)})} + e^{-\frac{1}{2}x_i h_i(x_{pa(i)})}}{e^{\frac{1}{2}x_i h_i(x_{pa(i)})} + e^{2w_{si}x_s x_i} e^{-\frac{1}{2}x_i h_i(x_{pa(i)})}} \\ &= e^{-x_s h_s(x_{pa(s)})} \prod_{i \in ch(s)} \underbrace{\frac{1 + e^{-x_i h_i(x_{pa(i)})}}{1 + e^{2w_{si}x_s x_i} e^{-x_i h_i(x_{pa(i)})}}}_{=: a_s(x_{bl(s)})}. \end{aligned}$$

This finally implies

$$p_{\xi}(x_s | x_{bl(s)}) = \frac{1}{1 + e^{-x_s h_s(x_{pa(s)})} a_s(x_{bl(s)})}. \quad (97)$$

Comparing this with the update rule (12), we observe that the full Markov blanket is involved in terms of the modulation function a_s .

References

1. Amari, Shun-ichi: Information geometry of the EM and em algorithms for neural networks. *Neural Netw.* **8**(9), 1379–1408 (1995)
2. Amari, S.: Natural gradient works efficiently in learning. *Neural Comput.* **10**(2), 251–276 (1998)

3. Amari, S.: Information Geometry and Its Applications, vol. 194. Springer, New York (2016)
4. Amari, S., Nagaoka, H.: Methods of Information Geometry. Oxford University Press, Oxford (2000)
5. Ay, N.: Locality of global stochastic interaction in directed acyclic networks. *Neural Comput.* **14**(12), 2959–2980 (2002)
6. Ay, N., Amari, S.: A novel approach to canonical divergences within information geometry. *Entropy* **17**(12), 8111–8129 (2015)
7. Ay, N., Montúfar, G., Rauh, J.: Selection criteria for neuromanifolds of stochastic dynamics. In: *Advances in Cognitive Neurodynamics (III)*, pp. 147–154. Springer, New York (2012)
8. Ay, N., Jost, J., Lê, H.V., Schwachhöfer, L.: Information Geometry. Springer, New York (2017)
9. Bi, G., Poo, M.: Synaptic modification of correlated activity: Hebb's postulate revisited. *Annu. Rev. Neurosci.* **24**(1), 139–166 (2001)
10. Bornschein, J., Shabanian, S., Fischer, A., Bengio, A.: Bidirectional Helmholtz machines. In: *International Conference on Machine Learning*, pp. 2511–2519 (2016)
11. Castro-González, N., Martínez-Serrano, M.F., Robles, J.: Expressions for the Moore-Penrose inverse of block matrices involving the schur complement. *Linear Algebra Appl.* 471:353–368 (2015)
12. Chentsov, N.N.: Statistical decision rules and optimal inference. In: *Mathematics Monographs*, vol. 53. American Mathematics Society, Providence (1982)
13. Dayan, P., Hinton, G.E., Neal, R.M., Zemel, R.S.: The Helmholtz machine. *Neural Comput.* **7**(5):889–904 (1995)
14. Fujiwara, A., Amari, S.: Gradient systems in view of information geometry. *Phys. D Nonlinear Phenomena* **80**(3), 317–327 (1995)
15. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016). <http://www.deeplearningbook.org>
16. Hebb, D.O.: The Organization of Behaviour - A Neuropsychological Theory. Wiley, New York (1949)
17. Hinton, G.E., Dayan, P., Frey, B.J., Neal, R.M.: The “wake-sleep” algorithm for unsupervised neural networks. *Science* **268**(5214), 1158–1161 (1995)
18. Ikeda, S., Amari, S., Nakahara, H.: Convergence of the wake-sleep algorithm. In: *Proceedings of the 11th International Conference on Neural Information Processing Systems, NIPS 1998*, pp. 239–245, MIT Press, Cambridge (1998)
19. Kakade, S.: A natural policy gradient. In: *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic, NIPS 2001*, pp. 1531–1538. MIT Press, Cambridge (2001)
20. Kurita, T.: Iterative weighted least squares algorithms for neural networks classifiers. *New Gener. Comput.* **12**(4), 375–394 (1994)
21. Lauritzen, S.: Graphical Models. Oxford University Press, Oxford (1996)
22. Martens, J.: New perspectives on the natural gradient method. [arXiv:1412.1193](https://arxiv.org/abs/1412.1193) (2015)
23. Martens, J., Grosse, R.: Optimizing neural networks with Kronecker-factored approximate curvature. [arXiv:1503.05671](https://arxiv.org/abs/1503.05671) (2015)
24. Neal, R.M.: Connectionist learning of belief networks. *Artif. Intell.* **56**(1), 71–113 (1992)
25. Neal, R.M., Dayan, P.: Factor analysis using delta-rule wake-sleep learning. *Neural Comput.* **9**(8), 1781–1803 (1997)
26. Ollivier, Y.: Riemannian metrics for neural networks I: feedforward networks. *Inf. Inference A J. IMA* **4**(2), 108–153 (2015)
27. Peters, J., Vijayakumar, S., Schaal, S.: Natural actor-critic. In: *Machine Learning: ECML 2005*, pp. 280–291. Springer, Berlin (2005)
28. Sun, K., Nielsen, F.: Relative Fisher information and natural gradient for learning large modular models. In: *Proceedings of the 34th International Conference on Machine Learning*, pp. 3289–3298 (2017)
29. Thomas, P.S.: GeNGA: A generalization of natural gradient ascent with positive and negative convergence results. In: *31st International Conference on Machine Learning, ICML 2014*, vol. 5, 3533–3541, 01 (2014)
30. Trager, M., Kohn, K., Bruna, J.: Pure and spurious critical points: a geometric study of linear networks. In: *International Conference on Learning Representations* (2020). <https://openreview.net/forum?id=rkgOICVYvB>
31. Várady, C.-H., Volpi, R., Malagò, L., Ay, N.: Natural wake-sleep algorithm. [arXiv:2008.06687](https://arxiv.org/abs/2008.06687) (2020)
32. Watanabe, S.: Algebraic Geometry and Statistical Learning Theory. Cambridge University Press, Cambridge (2009)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.