HAL
open science

Neural Empirical Bayes

Saeed Saremi, Aapo Hyvärinen

To cite this version:

Saeed Saremi, Aapo Hyvärinen. Neural Empirical Bayes. Journal of Machine Learning Research, 2019, 20, pp.1 - 23. hal-02419496

HAL Id: hal-02419496

https://inria.hal.science/hal-02419496v1

Submitted on 19 Dec 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Journal of Machine Learning Research 20 (2019) 1-23 Submitted 3/19; Revised 9/19; Published 11/19

Neural Empirical Bayes

Saeed Saremi SAEED@BERKELEY.EDU
Redwood Center for Theoretical Neuroscience
University of California
Berkeley, CA 94720-3198, USA
NNAISENSE Inc., Austin, TX

Aapo Hyvärinen AAPO.HYVARINEN@HELSINKI.FI
University College London, UK
Université Paris-Saclay, Inria, France
University of Helsinki, Finland

Editor: Yoshua Bengio

Abstract

We unify *kernel density estimation* and *empirical Bayes* and address a set of problems in unsupervised machine learning with a geometric interpretation of those methods, rooted in the *concentration of measure* phenomenon. Kernel density is viewed symbolically as $X \rightharpoonup Y$ where the random variable X is smoothed to $Y = X + N(0, \sigma^2 I_d)$, and empirical Bayes is the machinery to denoise in a *least-squares* sense, which we express as $X \leftharpoondown Y$. A learning objective is derived by combining these two, symbolically captured by $X \rightleftharpoons Y$. Crucially, instead of using the original nonparametric estimators, we parametrize *the energy function* with a neural network denoted by ϕ; at optimality, $\nabla\phi \approx -\nabla \log f$ where f is the density of Y. The optimization problem is abstracted as interactions of high-dimensional spheres which emerge due to the concentration of isotropic Gaussians. We introduce two algorithmic frameworks based on this machinery: (i) a "walk-jump" sampling scheme that combines Langevin MCMC (walks) and empirical Bayes (jumps), and (ii) a probabilistic framework for *associative memory*, called NEBULA, defined à la Hopfield by the *gradient flow* of the learned energy to a set of attractors. We finish the paper by reporting the *emergence* of very rich "creative memories" as attractors of NEBULA for highly-overlapping spheres.

Keywords: empirical Bayes, unnormalized densities, concentration of measure, Langevin MCMC, associative memory

1. Introduction

Let $X_1, \ldots, X_n$ be an *i.i.d.* sequence in $\mathbb{R}^d$ from a probability density function f_X. A well-known *nonparametric estimator* of f_X is the *kernel density estimator* $\widehat{f}$ that spreads the *Dirac masses* of the empirical distribution with a (gaussian) kernel f_N (Parzen, 1962). Consider $\widehat{f}$ and its standard definition, but take the kernel bandwidth to be a *hyperparameter*. Now, we can consider $\widehat{f}$ to be an estimator of the smoothed density $f_Y = f_X * f_N$, which is the probability density function of the noisy random variable

$$Y = X + N(0, \sigma^2 I_d).$$

©2019 Saeed Saremi and Aapo Hyvärinen.

License: CC-BY 4.0, see https://creativecommons.org/licenses/by/4.0/. Attribution requirements are provided at http://jmlr.org/papers/v20/19-216.html.

This setup is denoted symbolically as $X \rightharpoonup Y$.

Remark 1 (X and Y) *The distinction between X and Y is crucial, and in this work we constantly go "back and forth" between them. Regarding their probability density functions, we adopt a "clean notation": $f(y) = f_Y(y)$ and $f(x) = f_X(x)$, where the subscripts in f_X and f_Y are understood from their arguments x and y. In the absence of arguments, the subscripts are brought back, e.g.*

$$f_Y = f_X * f_N.$$

In addition, as it will become clear shortly, we are only concerned with approximating $\nabla \log f_Y$ in this paper, therefore the subscript Y becomes the default; it is understood that $f = f_Y$.

Next, we discuss the random variable Y in $\mathbb{R}^d$ from the perspective of *concentration of measure*. Assume that the random variable X in high dimensions is concentrated on a low-dimensional manifold $\mathscr{M}$, and denote the manifold where the random variable $Y = X + N(0, \sigma^2 I_d)$ is concentrated as $\mathscr{N}$.[1] We are interested in formalizing the hypothesis that as $d \to \infty$, the convolution of f_X and f_N (for any σ) would "disintegrate" $\mathscr{M}$ such that $\dim(\mathscr{N}) \gg \dim(\mathscr{M})$. We study an example of a "gaussian manifold", defined by $X \sim N(0, \Sigma_\sharp)$ where

$$\Sigma_\sharp = \begin{bmatrix} I_{d_\sharp} & 0 \\ 0 & \epsilon^2 I_{d-d_\sharp} \end{bmatrix},$$

with $\epsilon \ll 1$ and $d_\sharp \ll d$. We revisit textbook calculations on concentration of measure and show that in high dimensions, the smoothed manifold $\mathscr{N}$ is approximately a sphere of dimension $d-1$:

$$\mathscr{N} \approx \sqrt{(\epsilon^2 + \sigma^2)(d - d_\sharp)} S^{d-1}.$$

This analysis paints a picture of "manifold disintegration-expansion" as a general phenomenon.

(C1) The first conceptual contribution of this paper is to expose a notion of manifold disintegration-expansion, with a general takeaway that in (very) high dimensions, gaussian smoothing pushes away (all) the probability masses near $\mathscr{M}$. This is due to the fact that in high dimensions, the Gaussian random variable

$$N(0, \sigma^2 I_d) \approx \mathrm{Unif}(\sigma\sqrt{d} S^{d-1})$$

is not necessarily concentrated on $\mathscr{M}$, and there are many directions (asymptotically infinite) where samples from Y "escape" the manifold. The thesis is that, in convolving f_X and f_N, $\mathscr{M}$ would be mapped to a (much) higher dimensional manifold.

A seemingly unrelated theory is *empirical Bayes*, as formulated by Robbins (1956), which we use for pulling the probability mass back towards $\mathscr{M}$. In 1956, Robbins considered a scenario of a random variable Y that depends "in a known way" on an "unknown" random variable

1. Unfortunately, the concept of "manifold" for a probability distribution, let alone its dimension, is quite a fuzzy concept. But it should also be stated that this is a very important problem; the *assumption* on the *existence* of $\mathscr{M}$ is a pillar of machine learning (Saul and Roweis, 2003; Bengio et al., 2013a).

X.[2] He found that, given an observation $Y = y$, the *least squares estimator* of X is the *Bayes estimator*, and quite remarkably, can be expressed *purely* in terms of the distribution of Y (he showed this for Poisson, geometric, and Laplacian kernels). Robbins' results were later extended to *gaussian kernels* by (Miyasawa, 1961) who derived the estimator

$$\widehat{x}(y) = y + \sigma^2 \nabla \log f(y).$$

The least-squares estimator of X above is derived for *any* σ, not necessarily infinitesimal. To signify this important fact, we also refer to this estimation as "Robbins jump" or *jump* for short. The empirical Bayes machinery is denoted symbolically as

$$X \leftharpoonup Y.$$

The smoothing of kernel density estimation, $X \rightharpoonup Y$, and the denoising mechanism of empirical Bayes, $X \leftharpoonup Y$, come together to define the learning objective

$$\mathcal{L} = \mathbb{E}\, \|X - \widehat{x}(Y)\|^2.$$

(The squared ℓ_2 norm $\|\cdot\|^2$ in the definition of $\mathcal{L}$ is due to the fact that the estimator of X is a *least-squares estimator*.) In this setup, we take the two *nonparametric estimators* that defined $\mathcal{L}$ and parametrize the *energy function* of the random variable Y with a neural network $\phi : \mathbb{R}^d \to \mathbb{R}$ with parameters θ. It should be stated that ϕ is defined modulo an additive constant. The learning objective, expressed in terms of θ, is therefore given by

$$\mathcal{L}(\theta) = \mathbb{E}\, \|X - Y + \sigma^2 \nabla \phi(Y, \theta)\|^2. \tag{1}$$

(Throughout the paper, ∇ is the gradient taken with respect to the inputs y, not parameters.)

(C2) The second conceptual contribution of this paper is this unification of kernel density estimation and empirical Bayes. The unification, denoted symbolically as $X \rightleftharpoons Y$, is encapsulated in the expression for the learning objective above, $\mathcal{L} = \mathbb{E}\, \|X - \widehat{x}(Y)\|^2$. We thus combine two principles of nonparametric estimation into a single learning objective. In optimizing the objective, we choose to parametrize the energy function with a (overparametrized) neural network. The growing understanding on the representational power of deep (and wide) neural networks and the effectiveness of SGD for the "problem of learning" (Vapnik, 1995) is behind this choice.

Remark 2 (DEEN) *For gaussian noise, the objective $\mathcal{L} = \mathbb{E}\, \|X - \widehat{x}(Y)\|^2$ is the same as the learning objective in "deep energy estimator networks" (DEEN) (Saremi et al., 2018) which itself was based on denoising score matching (Hyvärinen, 2005; Vincent, 2011). However, this equivalence breaks down beyond gaussian kernels—see (Raphan and Simoncelli, 2011) for a comprehensive survey of empirical Bayes least squares estimators. From this angle, empirical Bayes appears as a more fundamental framework to formulate the problem of unnormalized density estimation for noisy random variables.*

2. The starting point in (Robbins, 1956) was the existence of a noisy random variable that was denoted by X. In that setup, one can *only* observe values of X. Here, we started with the *"clean"* i.i.d. sequence $X_1, \ldots, X_n$ and *artifically created* $Y = X + N(0, \sigma^2 I_d)$. This departure in starting points is related to our new take on empirical Bayes which will become clear in Section 3.

When analyzing a finite i.i.d. sequence, $X_1, \ldots, X_n$, i.i.d. samples from $Y = X + N(0, \sigma^2 I_d)$ are generated as

$$Y_{ij} = X_i + \varepsilon_j, \ \varepsilon_j \sim N(0, \sigma^2 I_d),$$

and the learning objective $\mathcal{L}(\theta)$ is then approximated as

$$\mathcal{L}(\theta) \approx \sum \left\| X_i - Y_{ij} + \sigma^2 \nabla \phi(Y_{ij}, \theta) \right\|^2,$$

where $\sum$ is a shorthand for $\frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m}$. In high dimensions, the samples Y_{ij} are (approximately) uniformly distributed on a "thin spherical shell" around the sphere $\sigma\sqrt{d}S_i^{d-1}$ with its center at X_i which leads to the following definition.

Definition 3 (i-sphere) *The sphere $\sigma\sqrt{d}S_i^{d-1}$ centered at X_i is a useful abstraction and it is referred to as "i-sphere". The i in i-sphere refers to the index i in X_i, and the index j is reserved for the gaussian noise ε_j. Therefore, Y_{ij} is the jth sample on the i-sphere. Fixing the index i, the samples Y_{ij} are approximately uniformly distributed on a thin spherical shell around the i-sphere (see Figure 1).*

In fact, the learning based on $\mathcal{L}$ can be interpreted as shaping the global energy function such that locally, for samples Y_{ij} from a single i-sphere, the denoised versions $\widehat{x}(Y_{ij})$ come as close as possible (in squared ℓ_2 norm) to X_i at the center (see Figure 1c). This "shaping of the energy function" is programmed in a simple algorithm,

DEEN: minimize $\mathcal{L}(\theta)$ with stochastic gradient descent and return θ^*.

Remark 4 *In most of the paper, we are interested in studying ϕ which is already at optimality with parameteres $\theta^* = \operatorname{argmin} \mathcal{L}(\theta)$. In these contexts, θ^* is dropped, and its presence is understood, e.g. at optimality, $\phi \approx -\log f$ (modulo a constant).*

Next, we define a notion of interactions between i-spheres that we find useful in thinking about the goal of approximating the *score function* through optimizing $\mathcal{L}(\theta)$. However, this can be skipped until our discussions of the associative memory in Sections 6 and 7.

Definition 5 (i-sphere interactions) *The interaction between i-spheres is defined as the competition that emerge due to the presence of $\nabla\phi$ in the problem of minimizing $\mathcal{L}$. In other words, it refers to "the set of constraints" that $\nabla\phi$, the gradient of the globally defined energy function, has to satisfy locally on i-spheres to arrive at the optimality such that $\nabla\phi \approx -\nabla \log f$. It is indeed an abstract notion in this work, but it is a useful abstraction to have which we come back to in thinking about the gradient flow $y' = -\nabla\phi(y)$. It should be stated that there is also a notion of interaction that already exists for the kernel density in the sense of "mass aggregation" (see Section 7 for references), but in this framework, i-spheres interact even when they do not overlap in that they "communicate" via $\nabla\phi$.*

(C3) The third conceptual contribution of this paper is introducing the physical picture of interactions between *i.i.d.* samples. This is a useful abstraction in this work in thinking about the problem of approximating the *score function* with the gradient of an *energy function*. The interaction is a code for "the set of constraints" that $\nabla\phi$ (evaluated at i.i.d. samples Y_{ij}) has to satisfy to arrive at optimality, $\nabla\phi \approx -\nabla \log f$. These interactions could also give rise to some *collective phenomena* (Anderson, 1972).

Next, we outline the technical contributions of this paper. Approximating $\nabla \log f$ in the empirical Bayes setup leads to a novel sampling algorithm, a new notion of associative memory and the emergence of "creative memories". First, we define the matrix χ that was introduced to quantify i-sphere overlaps and was used for designing experiments.

(i) Concentration of measure leads to a geometric picture for the kernel density as a *"mixture of i-spheres"*, and we define the matrix χ to quantify the extent to which the spheres in the mixture overlap; the entries in χ are essentially pairwise distances, scaled by $2\sqrt{d}$ (where the scaling is related to the concentration of isotropic Gaussians in high dimensions):

$$\chi_{ii'} = \frac{\|X_i - X_{i'}\|}{2\sqrt{d}}. \tag{2}$$

The i-sphere and the i'-sphere do not overlap if $\sigma < \chi_{ii'}$, and they overlap if $\sigma > \chi_{ii'}$ (see Figure 2). Of special interest is σ_c, defined as

$$\sigma_c = \max_{ii'} \chi_{ii'}.$$

We define "extreme noise" as the regime $\sigma > \sigma_c$; it is the regime where all i-spheres have some degree of overlap.

(ii) "Walk-jump sampling" is an approximative sampling algorithm that first draws exact samples from the density $\exp(-\phi)/Z$ (Z is the unknown normalizing constant) by *Langevin MCMC*:

$$y_{t+1} = y_t - \delta^2 \nabla \phi(y_t) + \sqrt{2}\delta\varepsilon, \quad \varepsilon \sim N(0, I_d),$$

where $\delta \ll 1$ is the step size and here t is discrete time. At an arbitrary time τ, approximative samples "close" to $\mathscr{M}$ are generated with the least-squares estimator of X—*the jumps*:

$$\widehat{x}(y_\tau) = y_\tau - \sigma^2 \nabla \phi(y_\tau).$$

A signature of walk-jump sampling is that the walks and the jumps are decoupled in that the jumps can be made at arbitrary times.

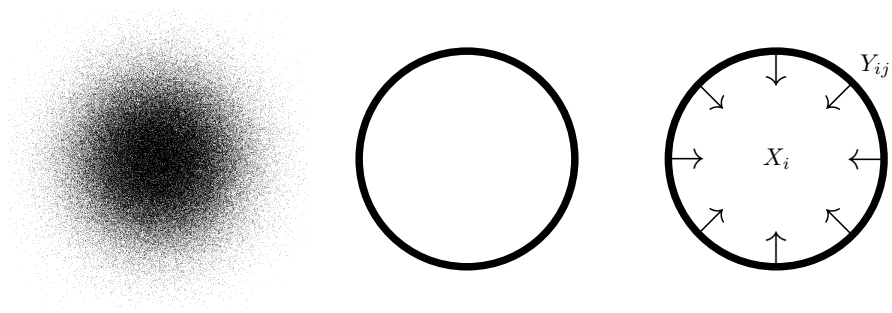
(iii) "Neural empirical Bayes associative memory"—named NEBULA, where the "LA" refers to *à la* Hopfield (1982)—is defined as the flow to strict local minima of the energy function ϕ. In continuous time, the memory dynamics is governed by the *gradient flow*:

$$y'(t) = -\nabla \phi(y(t)).$$

In retrieving a "memory", one flows deterministically to an *attractor*. NEBULA is an intriguing construct as the deterministic dynamics is governed by a *probability density function*, since $\nabla \phi \approx -\nabla \log f$—this is very far from true for Hopfield networks.

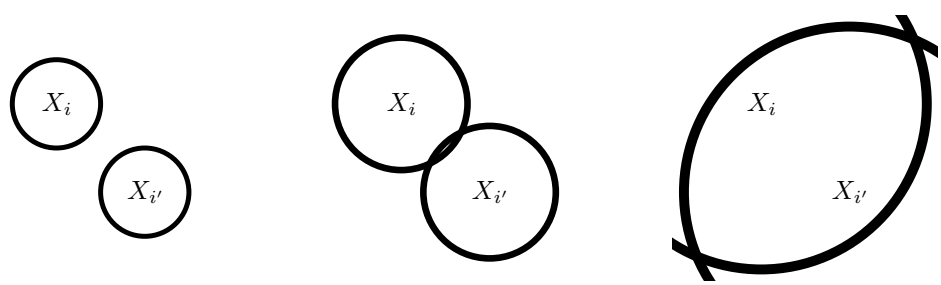
(iv) Memories in NEBULA are believed to be formed due to *i-sphere interactions* (Definition 3). We provide evidence for the presence of this abstract notion of interaction in this problem by reporting the emergence of very structured memories in a regime of highly overlapping i-spheres. They are named "creative memories" because they have intuitively appealing structures, while clearly being new instances, in fact quite different from the training data.

This paper is organized as follows. *Manifold disintegration-expansion* is discussed in Section 2. In Section 3, we review empirical Bayes and give a derivation of the least squares estimator of X for gaussian kernels. We then talk about a *Gedankenexperiment*—in the school of Robbins (1956) and Robbins and Monro (1951)—to bring home the *unification scheme* $X \rightleftharpoons Y$ and shed some light on the inner-working of the learning objective. In Section 4, we define the notion of *extreme noise* and demonstrate two extreme denoising experiments. In Section 5, we present the *walk-jump sampling* algorithm. *NEBULA* is defined in Section 6, where the abstract notion of *i-sphere interactions* is grounded to some extent; we present two sets of experiments with qualitatively different behaviors. In Section 7, we report the emergence of *creative memories* for two values of σ close to σ_c. We finish with a summary.



(a) $d = 2$ gaussian (b) $d \gg 1$ gaussian (c) $X \rightleftharpoons Y$

Figure 1: (a) Samples from a 2D isotropic gaussian, obtained and rendered in the programming language Processing. (b) Schematic of an isotropic gaussian in high dimensions, where the concentration of norm is illustarted. (c) Schematic of the i-sphere, with samples $Y_{ij} = X_i + \varepsilon_j,\ \varepsilon_j \sim N(0, \sigma^2 I_d)$. The arrows represent $-\nabla\phi$, evaluated on the sphere. The learning objective is encapsulated by $X \rightleftharpoons Y$, where the squared ℓ_2 norm $\|X_i - \widehat{x}(Y_{ij})\|^2$ is the learning signal and minimized in expectation. Ignoring the other spheres, the learning objective is constructed such that $-\nabla\phi$ evaluated at Y_{ij} points to X_i.



(a) $\sigma < \chi_{ii'}$ (b) $\sigma \approx \chi_{ii'}$ (c) $\sigma > 2\chi_{ii'}$

Figure 2: (*overlapping i-spheres*) The extent of the overlap between i-sphere and i'-sphere is tuned by σ in relation to $\chi_{ii'} = \|X_i - X_{i'}\|/(2\sqrt{d})$. The scaling $(2\sqrt{d})$ is due to the fact that $N(0, \sigma^2 I_d) \approx \mathrm{Unif}(\sigma\sqrt{d}S^{d-1})$ in high dimensions.

2. Manifold disintegration-expansion in high dimensions

Our low-dimensional intuitions break down spectacularly in high dimensions. This is in large part due to the exponential growth of space in high dimensions (this abundance of space also underlies the *curse of dimensionality*, well known in statistics and machine learning). Our focus in this section is to develop some intuitions on the effect of gaussian smoothing, $X \rightharpoonup Y$, on the data manifold $\mathscr{M}$. We start by a summary of results on *concentration of measure*. The textbooks (Ledoux, 2001; Tao, 2012; Vershynin, 2018) should be consulted to fill in details. We then extend the textbook calculations for a "gaussian manifold" and make a case for *manifold disintegration-expansion*.

Start with the isotropic gaussian. In 2D, samples from the gaussian form a "cloud of points", centered around its mode as illustrated in Figure 1a. Next, take the isotropic Gaussian in d dimensions, $(X_1, \ldots, X_d) = N(0, I_d)$, and ask where the random variable is likely to be located. Before stating a well-known result on the concentration of measure, we can build intuition by observing the following identities for the expectation and the variance of the squared norm:

$$\begin{aligned} \mathbb{E}\,\|X\|^2 &= \sum_{i=1}^{d} \mathbb{E}\,X_i^2 = d, \\ \mathbb{V}\|X\|^2 &= \sum_{i=1}^{d} \mathbb{E}(X_i^2 - 1)^2 = 2d. \end{aligned}$$

Since the components of $X = (X_1, \ldots, X_d)$ are jointly independent, and since $d \gg 1$, the following holds with high probability,

$$\begin{aligned} \|X\|^2 &= d \pm O(\sqrt{d}), \\ \|X\| &= \sqrt{d} \pm O(1). \end{aligned}$$

The concentration of norm is visualized in Figure 1b; also see Figure 3.2 in (Vershynin, 2018). In contradiction to our low-dimensional intuition, in high dimensions, the Gaussian random variable $N(0, I_d)$ is not concentrated close to the mode of its density f_N (at the origin here), but in a "thin spherical shell" of width $O(1)$ around the sphere of radius $\sqrt{d}$. The concentration of norm suggests an even deeper phenomenon, the concentration of measure, captured by a non-asymptotic result for the deviation of the norm $\|X\|$ from $\sqrt{d}$, where the deviation has a *sub-gaussian* tail (Tao, 2012; Vershynin, 2018):

$$\mathbb{P}\left(\left|\|X\| - \sqrt{d}\right| \geq t\right) \leq 2\exp(-ct^2), \quad \text{for all } t \geq 0.$$

A related result states that

$$\mathbb{P}(\|X\| \leq \epsilon\sqrt{d}) \leq (C\epsilon)^d, \quad \text{for all } \epsilon \geq 0.$$

In above expressions, C and c are absolute constants. In high dimensions, the Gaussian random variable is thus approximated with the uniform distribution on the sphere $\sigma\sqrt{d}S^{d-1}$,

$$N(0, \sigma^2 I_d) \approx \mathrm{Unif}(\sigma\sqrt{d}S^{d-1}).$$

The approximation becomes equality in distribution, as $d \to \infty$.

Remark 6 *In high dimensions, our intuitions for the uniform distribution itself breaks down, where the probability mass no longer has a "uniform geometry". For example,* $\mathrm{Unif}([0,1]^d)$ *is concentrated near the hyperplane* $X_1 + \cdots + X_d = d/2$*, which is easy to see, starting from* $\mathbb{E}\, X = 1/2$ *for the univariate* X *(Ledoux, 2001). The same goes for* $\mathrm{Unif}(\sqrt{d}S^{d-1})$*, which is a prime example of "concentration without independence", with counterintuitive results like the "blow-up" phenomenon (Vershynin, 2018).*

We finish this section with a new analysis. Consider $X \sim N(0, \Sigma_\sharp)$ in $\mathbb{R}^d$. A simple low-dimensional "manifold structure" with the dimension $d_\sharp \ll d$ is imposed by considering

$$\Sigma_\sharp = \begin{bmatrix} I_{d_\sharp} & 0 \\ 0 & \epsilon^2 I_{d-d_\sharp} \end{bmatrix},$$

where $\epsilon \ll 1$. The manifold $\mathscr{M}$ is viewed as a "gaussian manifold" where, in an abuse of notation, $\dim(\mathscr{M}) \approx d_\sharp$. Now consider $Y = X + N(0, \sigma^2 I_d)$ which means

$$Y = N(0, \Sigma_\sharp + \sigma^2 I_d).$$

We repeat the concentration of norm calculations:

$$\begin{aligned} \mathbb{E}\,\|Y\|^2 &= d_\sharp(1+\sigma^2) + (d-d_\sharp)(\epsilon^2+\sigma^2), \\ \mathbb{V}\|Y\|^2 &= 2d_\sharp(1+\sigma^2)^2 + 2(d-d_\sharp)(\epsilon^2+\sigma^2)^2. \end{aligned}$$

In high dimensions, assuming

$$(d/d_\sharp - 1) \gg \max(\Delta, \Delta^2), \;\; \Delta = (1+\sigma^2)/(\epsilon^2+\sigma^2),$$

the "manifold terms", $d_\sharp(1+\sigma^2)$ and $d_\sharp(1+\sigma^2)^2$, become negligible, and the following holds with high probability:

$$\begin{aligned} \|Y\|^2 &= (\epsilon^2+\sigma^2)(d - d_\sharp \pm O(\sqrt{d-d_\sharp})), \\ \|Y\| &= \sqrt{\epsilon^2+\sigma^2}(\sqrt{d-d_\sharp} \pm O(1)). \end{aligned}$$

The calculations in the previous page are reproduced by setting $\sigma = 1$, $\epsilon = 0$, $d_\sharp = 0$.

In summary, under the convolution $f_Y = f_X * f_N$, the "gaussian manifold" $\mathscr{M}$ is transformed/mapped to

$$\mathscr{N} \approx \sqrt{(\epsilon^2+\sigma^2)(d-d_\sharp)}S^{d-1},$$

therefore $\dim(\mathscr{N}) \approx d-1$. In our terminology, $\mathscr{M}$ has been *disintegrated-expanded* to $\mathscr{N}$. We expect this phenomenon to hold for any $\mathscr{M}$ asymptotically, i.e. as $d \to \infty$, $\mathscr{N} = \sigma\sqrt{d}S^{d-1}$, but (unfortunately) the limit $d \to \infty$ is not practical. However, the picture should stay: in high dimensions, the convolution of f_X and f_N has severe side effects on the manifold $\mathscr{M}$ itself. Smoothing is indeed desired, but also a mechanism to restore $\mathscr{M}$. In the next section, we discuss such a mechanism using *empirical Bayes*, where the "restoration" is in *least squares*.

3. Neural empirical Bayes

In a seminal work from 1956, titled *an empirical Bayes approach to statistics*, Robbins considered an intriguing scenario of a random variable Y having a probability distribution that depends "in a known way" on an "unknown" random variable X (Robbins, 1956). Observing $Y = y$, the *least-squares estimator* of X is the *Bayes estimator* (see Remark 1):

$$\widehat{x}(y) = \frac{\int x f(y|x) f(x) dx}{\int f(y|x) f(x) dx} = \int x f(x|y) dx = \mathbb{E}(X|Y = y).$$

If f_X is known to "the experimenter", $\widehat{x}$ is a computable function, but what if the prior f_X is unknown? It is quite remarkable that for a large class of kernels, the least-squares estimator can be derived in closed form purely in terms of the distribution of Y. Informally speaking, there is an "abstraction barrier" (Abelson and Sussman, 1985) between X and Y where the knowledge of f_X is not needed to estimate X. This *functional* dependence of $\widehat{x}$ on f_Y was achieved for the Poisson, geometric, and Laplacian kernels in (Robbins, 1956), and it was later extended to the gaussian kernel by (Miyasawa, 1961).

This work builds on Miyasawa's result, and we repeat the calculation here in our notations. Take X to be a random variable in $\mathbb{R}^d$ and Y a noisy observation of X with a known gaussian kernel with symmetric positive-definite $\Sigma \succ 0$:

$$f(y|x) = \frac{1}{(2\pi)^{d/2} |\det(\Sigma)|^{1/2}} \exp\left(-(y-x)^\top \Sigma^{-1} (y-x)/2\right).$$

It follows,

$$\Sigma \ \nabla_y f(y|x) = f(y|x)(x - y).$$

Multiply the expression above by $f(x)$ and integrate,

$$\Sigma \ \nabla f(y) = \int (x - y) f(y|x) f(x) dx = f(y)(\widehat{x}(y) - y),$$

which then leads to the expression

$$\widehat{x}(y) = y + \Sigma \ \nabla \log f(y).$$

For the isotropic case $\Sigma = \sigma^2 I_d$, the estimator takes the form

$$\widehat{x}(y) = y + \sigma^2 \nabla \log f(y). \tag{3}$$

To sum up, the estimator above is obtained in a setup where only the corrupted data (the random variable Y) are observed, with the knowledge of the measurement (gaussian) kernel *alone* and without *any knowledge* of the prior f_X. The remarkable result is the fact the least-squares estimator of X is written in closed form purely as a functional of $\nabla \log f$, also known as the *score function* (Hyvärinen, 2005). The expression above for the *least squares* estimator of X is the basis for an algorithm to approximate $\nabla \log f$ which is discussed next.

In *empirical Bayes*, the random variable X is estimated in least squares from the noisy observations Y, without *any* knowledge of the *prior* f_X. But how did we get here? We started the paper with the i.i.d. sequence $X_1, \ldots, X_n$, where Y did not even exist!

The idea is a Gedankenexperiment of sort, where we first construct $Y = X + N(0, \sigma^2 I_d)$ by taking samples $Y_{ij} \sim f_X * f_N$; in turn, the experimenter (in the school of empirical Bayes) "observes" Y_{ij} and estimates X. But in this artificially supervised setup (the "target" is X_i), the error signal $\|X_i - \widehat{x}(Y_{ij})\|^2$ can also be measured—this is not the case in (Robbins, 1956)—and it drives the learning. The learning is achieved with stochastic gradient descent as the experimenter has parameterized the energy function with a neural network and can compute the stochastic gradients (Robbins and Monro, 1951),

$$\nabla_\theta \|X_i - Y_{ij} + \nabla_y \phi(Y_{ij}, \theta)\|^2.$$

Approximating $\nabla \log f$ in this manner is the essence of neural empirical Bayes.

For *finite* samples, the best we can do is to *approximate* $\nabla \log f$. Given $X_1, \ldots, X_n$, the i.i.d. samples $Y_{ij} \sim f_X * f_N$ are given by $Y_{ij} = X_i + \varepsilon_j,\ \varepsilon_j \sim N(0, \sigma^2 I_d)$. These are indeed i.i.d. samples from density estimated by the kernel density estimator

$$\widehat{f}_Y(y) = \frac{1}{n} \sum_{i=1}^{n} f_N(y - X_i),$$

where the subscript in f_N is a short for $N(0, \sigma^2 I_d)$. Here, σ is a *hyperparameter* and the kernel estimator above is considered an estimator of f_Y. Note that the *optimal bandwidth* in estimating f_X depends on n (van der Vaart, 2000).

The problem is that, in high dimensions, the kernel density estimator $\widehat{f}_Y$ (or any other *nonparametric estimator*) suffers from a severe *curse of dimensionality* (Wainwright, 2019). This leads naturally to *deep neural architectures* which has been viewed as an alternative for "breaking the curse of dimensionality" (Bengio, 2009). In this work, the *energy function* is parametrized with a deep neural network $\phi : \mathbb{R}^d \to \mathbb{R}$ with parameters θ. The objective $\mathcal{L}(\theta)$ is then approximated,

$$\mathcal{L}(\theta) \approx \sum \left\| X_i - Y_{ij} + \sigma^2 \nabla \phi(Y_{ij}, \theta) \right\|^2,$$

At optimality (achieved with *stochastic gradient descent*), $\theta^* = \operatorname{argmin} \mathcal{L}(\theta)$,

$$-\nabla \phi(\cdot, \theta^*) \approx \nabla \log f(\cdot).$$

Remark 7 (implicit vs. explicit parameterization) *As it is made clear, our goal is to approximate $\nabla \log f : \mathbb{R}^d \to \mathbb{R}^d$. In doing so, another choice could have been the explicit parametrization of the score function as $\psi : \mathbb{R}^d \to \mathbb{R}^d$, where one avoids the computation $\nabla \phi$ in the optimization, but there, $\partial_j \psi_i = \partial_i \psi_j$ must hold pointwise. The parametrization of the energy function as $\phi : \mathbb{R}^d \to \mathbb{R}$ is essentially an implicit parametrization of $\nabla \log f$, since the computation $\nabla \phi$ is not a symbolic one, but achieved with automatic differentiation (Baydin et al., 2018). Also, the equivalence between the two strategies for one-hidden-layer networks (Vincent, 2011) breaks down for two hidden layers and beyond. See (Saremi, 2019) for a proof and further discussions.*

	min	median	mean	max $\approx \sigma_c$
χ	0.0186	0.1822	0.1812	0.2884

Table 1: Some statistics of the matrix χ (see Equation 2) estimated from 10^7 pairs from the *handwritten digit database* (LeCun et al., 1998). The σ in each experiment must be viewed in relation to these numbers and our review of the *concentration of measure* in Section 2 (see Figure 2).

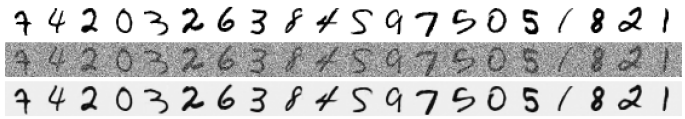


Figure 3: ($\sigma \approx \sigma_c$) Denoising performance of DEEN with a single jump for $\sigma = 0.3$. The noisy pixel values are in the range $[-1.200, 1.995]$, and the denoised ones are in $[-0.0749, 1.0539]$.

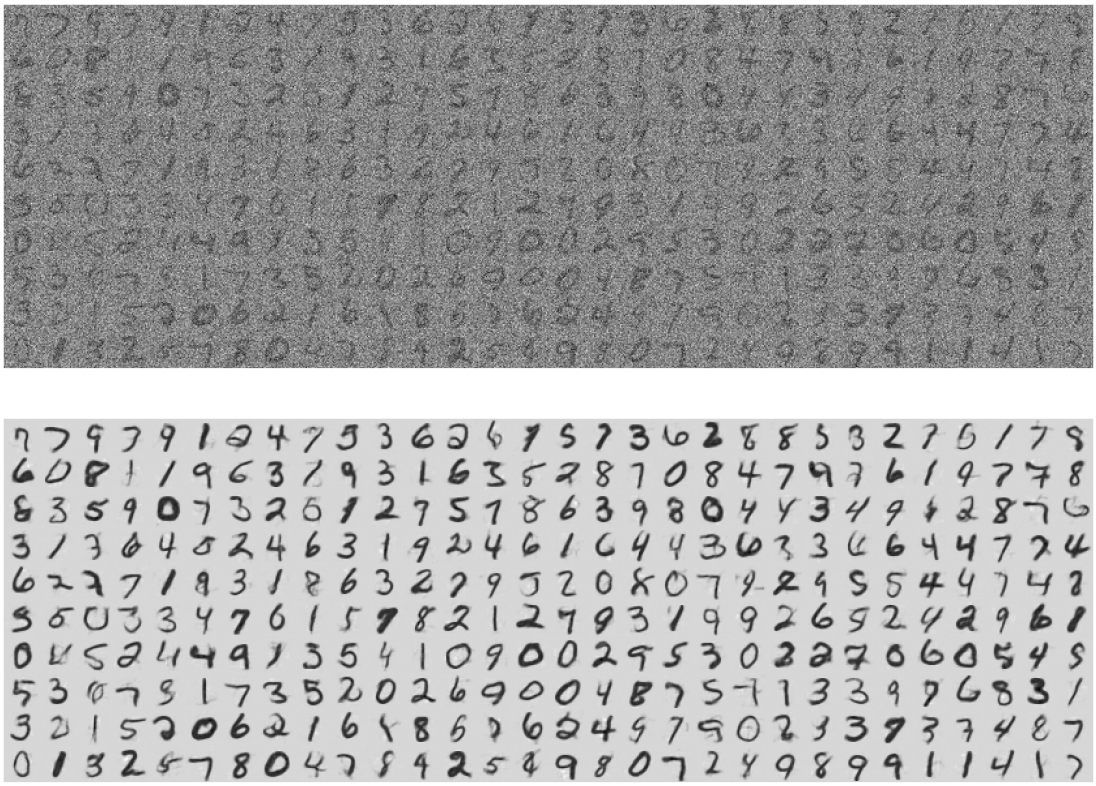


Figure 4: ($\sigma > 2\sigma_c$) Here, $\sigma = 0.7$. The noisy pixel values are in the range $[-3.314, 3.683]$, and the denoised ones are in $[-0.2405, 1.2686]$. In this regime, the whole database is inside each i-sphere (see Figure 2c).

4. Extreme noise/denoising

In empirical Bayes, the least-squares estimator of X,

$$\widehat{x}(y) = y + \sigma^2 \nabla \log f(y),$$

is derived for any σ, which inspires us to call this a "jump" to capture the fact σ may in fact be "large". Denoising is not our actual goal, but the expression above can be viewed as "denoising y to $\widehat{x}(y)$". The denoising performance is important since in the machinery of neural empirical Bayes, the goal of approximating the score function is formulated by the "denoising objective"

$$\mathcal{L}(\theta) = \mathbb{E}\,\|X - Y + \sigma^2 \nabla \phi(Y, \theta)\|^2.$$

As the least-squares estimator of X is derived for any σ, the denoising performance of DEEN may also be tested to its limits. Here, we report such experiments for MNIST, where $d = 784$ and X is in the hypercube $[0,1]^d$ (see Remark 8). Next, we elaborate on the geometric meaning of the noise levels, where in addition we define a notion of "extreme noise".

- ($\sigma > \sigma_c$). The value $\sigma_c = \max_{ii'} \chi_{ii'}$ is defined as the onset of *extreme noise*. Called "extreme", because for $\sigma > \sigma_c$, all i-spheres in the dataset overlap (see Figure 2b). For MNIST, $\sigma_c \approx 0.2884$ was estimated from 10^7 handwritten digit pairs (see Table 1). The results for $\sigma = 0.3$ on the *test set* is presented in Figure 3. It should be stated that, this value of "extreme noise", just above σ_c, is not visually *perceived* as extreme.
- ($\sigma > 2\sigma_c$). In this regime, due to geometry, the whole dataset is in *inside* of each i-sphere (see Figure 2c). This is very extreme! For MNIST, we experimented with $\sigma = 0.7$ which is in this regime. The results on the *test set* are presented in Figure 4.

Remark 8 *Note that the least-squares estimator of X can take values anywhere in $\mathbb{R}^d$, not restricted to be in $[0,1]^d$. This is the reason for the "gray background" in the experiments, in relation to which, in the figure captions, we report the min/max of all the pixel values. This also holds for the jump samples presented in the next section.*

4.1. The network architecture

The following comments are in order regarding the *network architecture* we used in our experiments.

- (*activation function*) The denoising results are a significant improvement over (Saremi et al., 2018). This was due to the use of ConvNets, instead of a fully connected network (more on that below) and the use of a *"smooth ReLU"* activation function

  $$\sigma(z, \beta) = z/(1 + \exp(-\beta z)),$$

  where the default was $\beta = 1$. The activation function above converges uniformly to ReLU in the limit $\beta \to \infty$. It has been studied independently from different angles (Elfwing et al., 2018; Ramachandran et al., 2017). ReLU consistently gave a higher loss, which we believe is due to the term $\nabla_y \phi$ in the learning objective, which must be computed first before computing $\nabla_\theta \mathcal{L}$ to update the parameters $\theta \leftarrow \theta - \epsilon \nabla_\theta \mathcal{L}$.

- *(wide convnets)* All experiments reported in the paper were performed in a *fixed* wide ConvNet architecture with the expanding channels $= (256, 512, 1024)$, *without pooling*, and with a *bottleneck layer* of size 10. All hidden layers were activated with the activation function $\sigma(\cdot, \beta = 1)$, and the readout layer was linear.
- *(readout overparametrization)* We observed a slightly faster convergence to lower loss (especially, very early in the training) by *overparameterizing* the linear output layer. These experiments were inspired by the "acceleration effects" studied for *linear neural networks* (Arora et al., 2018), but we did not study it thoroughly.
- We used the *Adam optimizer* (Kingma and Ba, 2014). The optimization was stable over a wide range of *mini-batch sizes* and *learning rates*, and as expected, we did not observe the validation/test loss going up in the experiments. This stability is due to the fact that the inputs to ϕ are noisy samples. The *automatic differentiation* (Baydin et al., 2018) was implemented in PyTorch (Paszke et al., 2017). DEEN is "memory hungry" but the computational costs are not addressed here in detail.

5. Walk-jump sampling: Langevin walk, Robbins jump

Sampling "complex distributions" in high dimensions is an intractable problem due to the fact that their probability mass is concentrated in *sharp ridges* that are separated by *large regions of low probability*, therefore making MCMC methods ineffective *(the "low-dimensional manifold" $\mathscr{M}$ is in fact quite complex when viewed in the ambient space $\mathbb{R}^d$)*. These problems are well known, but empirically they have mostly been studied for sampling f_X. Regarding the random variable Y, it is clear that the problem of sampling $f = f_X * f_N$ is in fact easier. The idea is that MCMC mixes faster as $\mathscr{N}$ is less complex. The "sharp ridges" are smoothed out and (as we argued in Section 2) $\mathscr{M}$ itself is expanded in dimensions, $\dim(\mathscr{N}) \gg \dim(\mathscr{M})$, therefore the "large regions of low probability" themselves are smaller in size. In summary, *Langevin MCMC* is believed to be more effective in sampling

$$\exp(-\phi)/Z \approx f$$

than f_X (putting aside the harder problem of learning $\nabla \log f_X$). At any time, the estimator $\widehat{x}$ can be used to *jump* near $\mathscr{M}$. "Near" is intuitive, and it should be stated that neither $f_{X|Y}$ nor f_X is exactly sampled during jumps; the jump samples must be seen as heuristic.

In what follows, it is understood that ϕ is at optimality. We first give a description of the algorithm and then express the equations.

- *(walks)* The Langevin MCMC is used to draw statistical samples $y_t \sim \exp(-\phi)/Z$. Langevin MCMC is based on discretizing the *Langevin diffusion* (van Kampen, 1992) and the updates are based solely on $\nabla\phi$ (the equations are coming next), therefore not knowing the *normalizing constant* Z is of no concern.
- *(jumps)* Given y_τ, ideally an *exact sample* (MacKay, 2003) from $\exp(-\phi)/Z$, the jumps are made with the *Bayes estimator* of X. The *spirit of empirical Bayes* is fully present here: the jump samples are generated without knowing ∇f_X (or its approximation), and without running Langevin MCMC on its "rougher terrain".

What emerges is a sampling algorithm in two parts. The Langevin MCMC is the *engine* that should run continously. By contrast, the jumps can be made at any time. That is:

(i) Sample from $\exp(-\phi)/Z$ with Langevin MCMC,

$$y_{t+1} = y_t - \delta^2 \nabla \phi(y_t) + \sqrt{2}\delta\varepsilon,\ \varepsilon \sim N(0, I_d),$$

where $\delta \ll 1$ is the step size and here t is discrete time.

(ii) At any time τ, use the Bayes estimator of X to jump from y_τ to $\widehat{x}(y_\tau)$,

$$\widehat{x}(y_\tau) = y_\tau - \sigma^2 \nabla \phi(y_\tau).$$

Two types of approximations have been made in walk-jump sampling. The first is that the Langevin MCMC does not sample f but the density $\exp(-\phi)/Z \approx f$; this approximation is unavoidable. The approximation involved in jumps are less understood. Stepping back, $\widehat{x}(y_\tau)$ *can* be thought of as approximating the *posterior* $f_{X|y_\tau}$ with a single Dirac mass at $\mathbb{E}(X|Y = y_\tau)$ but this is a very poor approximation to the posterior. Regarding f_X itself, the intuitive idea behind Robbins jump is that the *Bayes estimator* of X "lands" near $\mathscr{M}$. But this intuition must be validated by computing the covariance of the estimator, and this is indeed the next step to better understand the walk-jump sampling.

We tested the algorithm on the *handwritten digit database* for $\sigma = 0.3$ and $\sigma = 0.15$. The results are shown in Figure 5 and 6 respectively. For $\sigma = 0.3$ (in the regime of highly overlapping spheres), there was mixing between styles/classes. For $\sigma = 0.15$ (in the regime of mostly non-overlapping spheres), samples stayed within a class while the styles changed.

Some side-by-side comparisons with *denoising autoencoders* are in order.

- In walk-jump sampling, the key step is sampling from the density $\exp(-\phi)/Z$. The least-squares estimation of X—the jumps—are decoupled from Langevin MCMC. This is in contrast to the chain $X_t \to Y_t \to X_{t+1}$ constructed in *generalized denoising autoencoders* (Bengio et al., 2013b), which does not suffer from the problems we mentioned for the jumps.

- Here, the learning objective is derived for *any* σ. By contrast, *denoising autoencoders* approximate the score function only in the limit $\sigma \to 0$ (Alain and Bengio, 2014). *Generalized denoising autoencoders* (Bengio et al., 2013b) and *generative stochastic networks* (Alain et al., 2016) were devised as a remedy for those $\sigma \to 0$ limitations. This is avoided altogether here.

- In this work, there is a clear *geometrical* notion for large and small σ, which is based on the concentration of measure phenomenon. This picture is lacking in the references cited, and it is not clear which ranges of noise values should be considered there.

- Most importantly, denoising autoencoders suffer from *"limited parameterization"* as expressed by Alain and Bengio (2014), and independently observed in (Saremi et al., 2018). To summarize, in denoising autoencoders, one must learn a curl-free encoder-decoder to be able to properly approximate the *score function* and this is problematic beyond *one-hidden-layer* architectures. In our work, this problem is avoided due to the *energy parameterization* (see Remark 7).

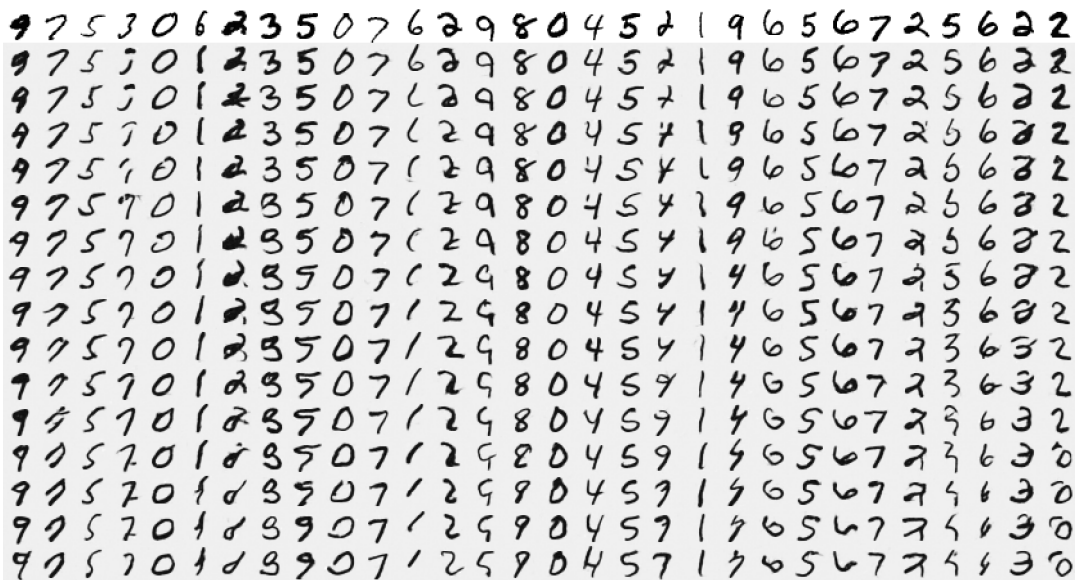


Figure 5: *($\sigma \approx \sigma_c$)* Top row is $x_0 \sim f_X$, sampled from the handwritten digit database which DEEN was trained on. The *Langevin MCMC* was intialized at $y_0 = x_0 + \varepsilon$, $\varepsilon \sim N(0, \sigma^2 I_d)$, where σ was the same value of σ which DEEN had been trained on. The samples y_t are are not shown. The *jumps* are shown in multiples of $\tau_0 = 10^4$, and the step size was $\delta = \sigma/100$. Here, $\sigma = 0.3$. The pixel values are in the range `[-0.07, 1.10]`.

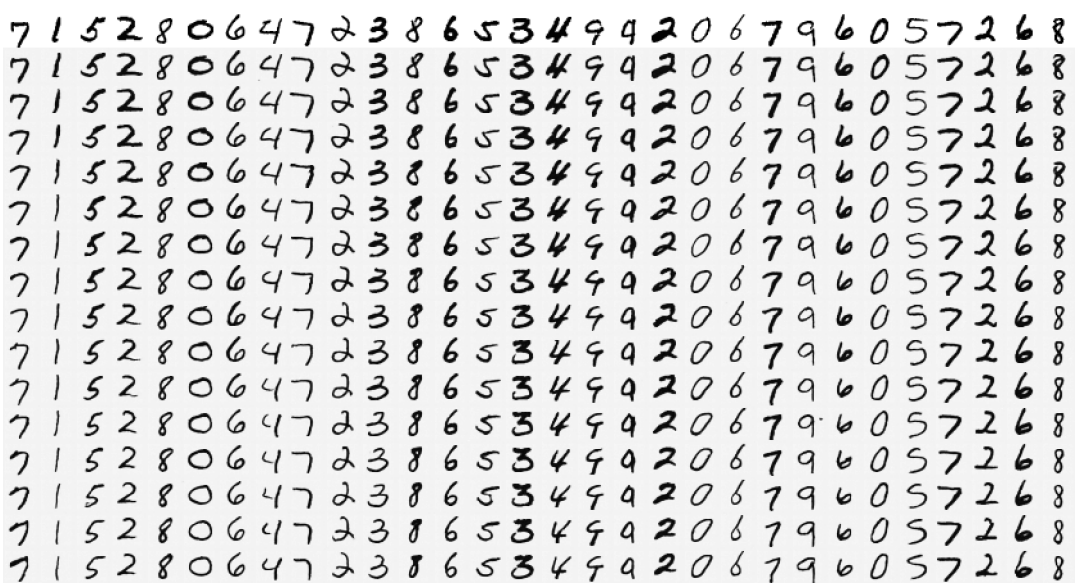


Figure 6: ($\sigma < \text{median}(\chi)$) Here, $\sigma = 0.15$, which is in the regime of mostly non-overlapping spheres. The Langevin MCMC parameters are set as above. The pixel values are in the range `[-0.05, 1.04]`.

6. Neural empirical Bayes associative memory (NEBULA)

In this section, the neural empirical Bayes machinery is used to define a new notion of associative memory. *Associative memory* (also known as *content-addressable memory*) is a deep subject in neural networks with a rich history and with roots in psychology and neuroscience (long-term potentiation and Hebb's rule). The depiction of this computation is even present in the arts and literature, championed by Marcel Proust, in the form of "stream of consciousness", and the constant back and forth between *perception* and *memory*.

In 1982, Hopfield brought together earlier attempts to formulate *associative memory*, and showed that the collective computation of a system of neurons with symmetric weights, in the form of asynchronous updates of *McCulloch-Pitts neurons* (McCulloch and Pitts, 1943), minimize an energy function (Hopfield, 1982). The energy function was constructed in a Hebbian fashion. The associative memory was then formulated as the "flow" (the neurons were binary) to the local minima of the energy function.

However, Hopfield's energy function is not *the* energy function[3]: it does not approximate (learn) the negative log probability density function of its stored memories. The *Boltzmann machine* (Hinton and Sejnowski, 1986) was developed in fact inspired by that observation, in which learning *the* energy function was achieved by introducing *hidden units*. Regarding the associative memory and the *phase space flow*, the problem is that in Boltzmann machines, hidden units need to be inferred first before having any notion of flow for the *visible units*. And inference is indeed computationally very expensive—*"the curse of inference"* (but not nearly as fundamental as the *curse of dimensionality*)—in *probabilistic graphical models* (Wainwright and Jordan, 2008; Koller and Friedman, 2009).

What we have achieved so far is to learn a function ϕ which approximates *the energy function* of Y, $\phi \approx -\log f$ (modulo a constant). The key here is that ϕ is a *function* (a "computer") that *computes* $f(y)$ for any y; the "hidden units" in ϕ are not inferred. In other words, the hidden units (and the parameters) in ϕ are there solely for *universal approximation*, $-\nabla\phi \approx \nabla \log f$. In *Boltzmann machines*, the hidden units are there to think! And this is the big difference. The definition below should be viewed against this backdrop.

Definition 9 *We define the neural empirical Bayes associative memory (NEBULA), à la Hopfield (1982), as the flow to strict local minima of the energy function ϕ. In continuous time, the memory dynamics is governed by the gradient flow:*

$$y'(t) = -\nabla\phi(y(t)). \tag{4}$$

NEBULA is identified by its attractors, the set of all the strict local minima of ϕ:

$$\mathcal{X}^* = \{X_1^*, \ldots, X_A^*\}.$$

The mapping $\mathscr{A} : \mathbb{R}^d \to \mathcal{X}^*$ denotes the convergence to an attractor under the gradient flow. The basin of attraction for the attractor X_a^* is denoted by $\mathcal{A}(X_a^*)$ and defined (intuitively) as the largest subset of $\mathbb{R}^d$ such that (in a slight abuse of notation) $\mathscr{A}(\mathcal{A}(X_a^*)) = \{X_a^*\}$.

3. In addition, Hopfield networks have severe limitations in *memory capacity*, addressed most recently in (Hillar and Tran, 2014; Krotov and Hopfield, 2016; Chaudhuri and Fiete, 2017).

What is so special about the local minima of ϕ? Start with a *single sample* X_1,

$$\phi(y) = \|y - X_1\|^2/(2\sigma^2),\ \mathcal{X}^* = \{X_1^*\},\quad X_1^* = X_1,\ \mathscr{A}(\cdot) = X_1,\ \mathcal{A}(X_1) = \mathbb{R}^d.$$

For *many samples*, the learning objective is *optimized* such that $-\nabla\phi$ evaluated on i-spheres point to X_i (see Figure 1c), but there are *conflicts of interests*, and these "conflicts" are the essence of i-sphere interactions, stated in Definition 3; in its practical summary, larger σ means more i-sphere overlaps, and therefore larger "interaction couplings". *For NEBULA, the presence of i-sphere interactions implies that X_a^* are not statistical samples from f_X.* In other words, given $X_i \sim f_X$, $\mathscr{A}(X_i) \not\sim f_X$. But what is the "right metric" $d(X_i, \mathscr{A}(X_i))$, to measure the distance between $X_i \sim f_X$ and $\mathscr{A}(X_i)$? The problem is that the flow is the gradient flow of $\phi \approx -\log f$ but the attractors are roughly speaking "close to $\mathscr{M}$". The total length of the path taken by the gradient flow is a natural choice but we leave that for furture studies. Our expectation (based on i-sphere interactions) is that $d(X_i, \mathscr{A}(X_i))$ will be larger for larger σ, visualized in Figures 7 and 8, with pronounced qualitative differences.

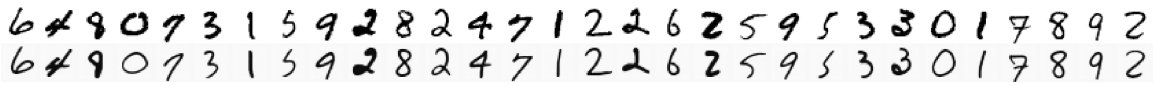


Figure 7: *(σ=0.15)* The top row are X_i from the MNIST test set. The bottom row are the attractors $\mathscr{A}(X_i)$; *they are not statistical samples from f_X: $\mathscr{A}(X_i) \not\sim f_X$.*

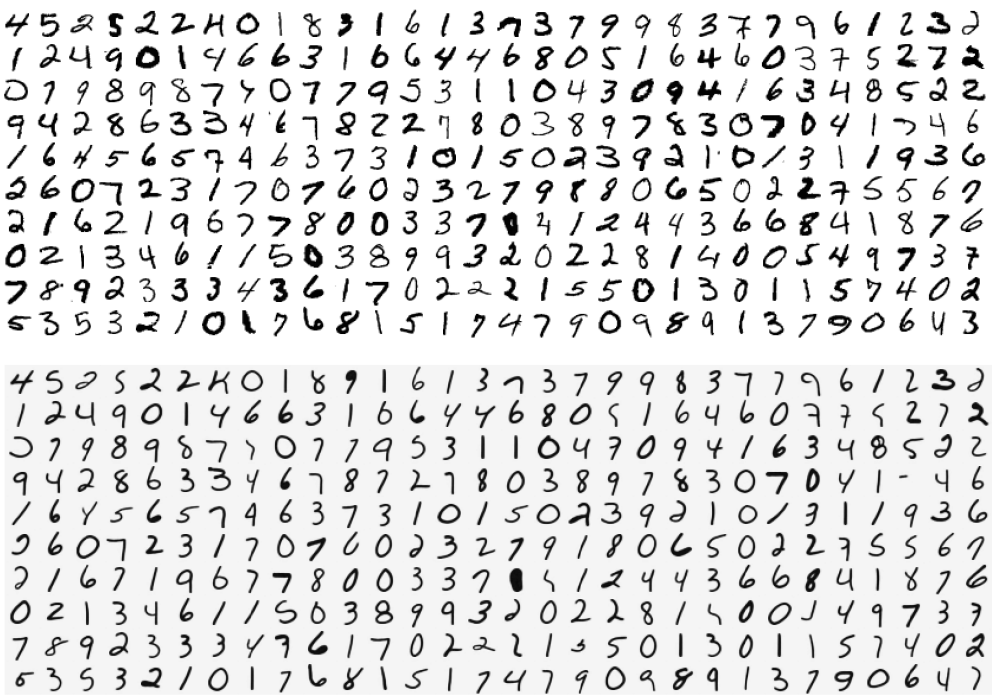


Figure 8: *(σ=0.3)* As expected, i-sphere interactions are stronger compared to $\sigma = 0.15$. Note the bottom right digit, flowing from a "3" to a "7".

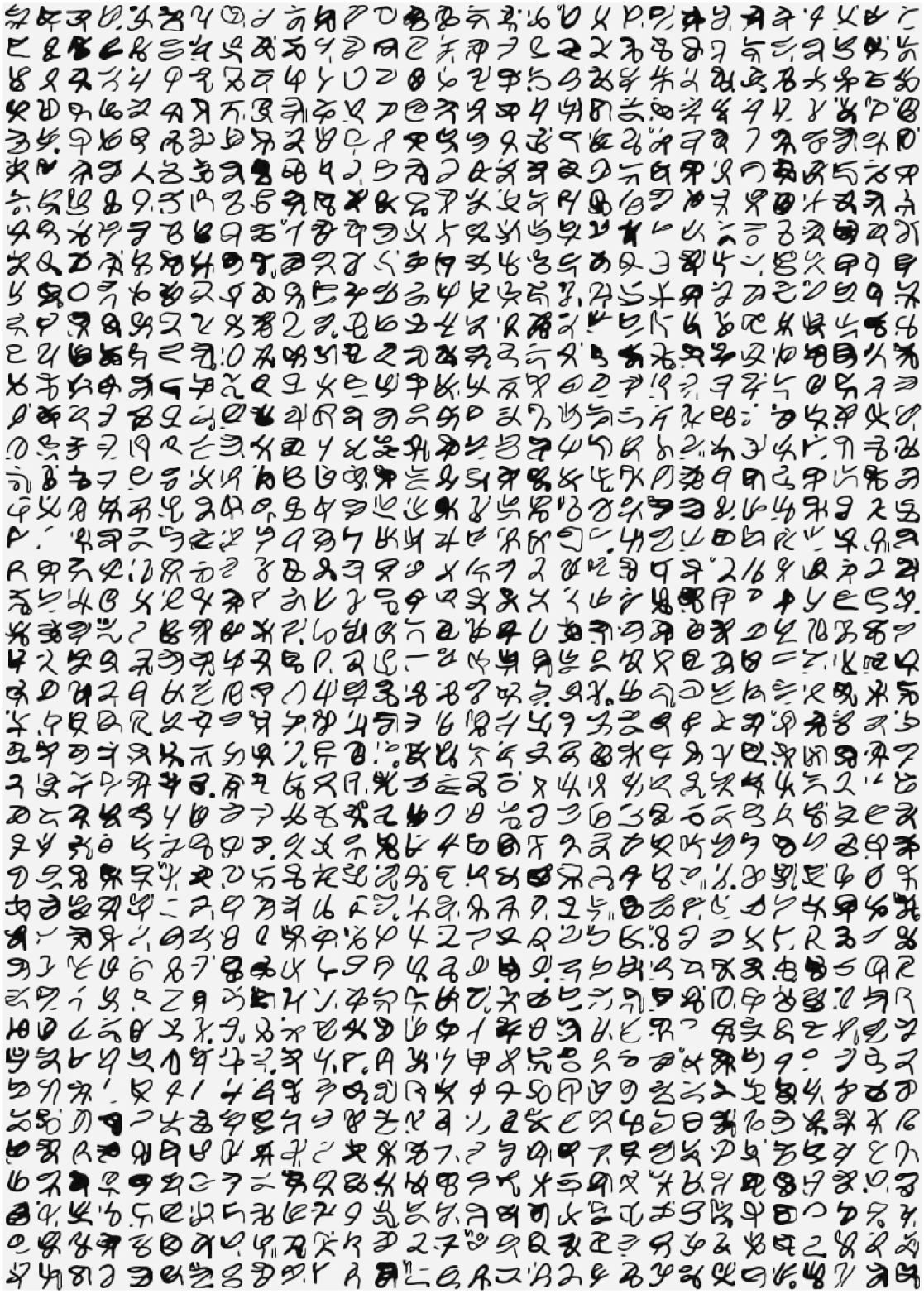


Figure 9: Attractors of NEBULA for $\sigma = 0.3$ ($\sigma_c \approx 0.29$).

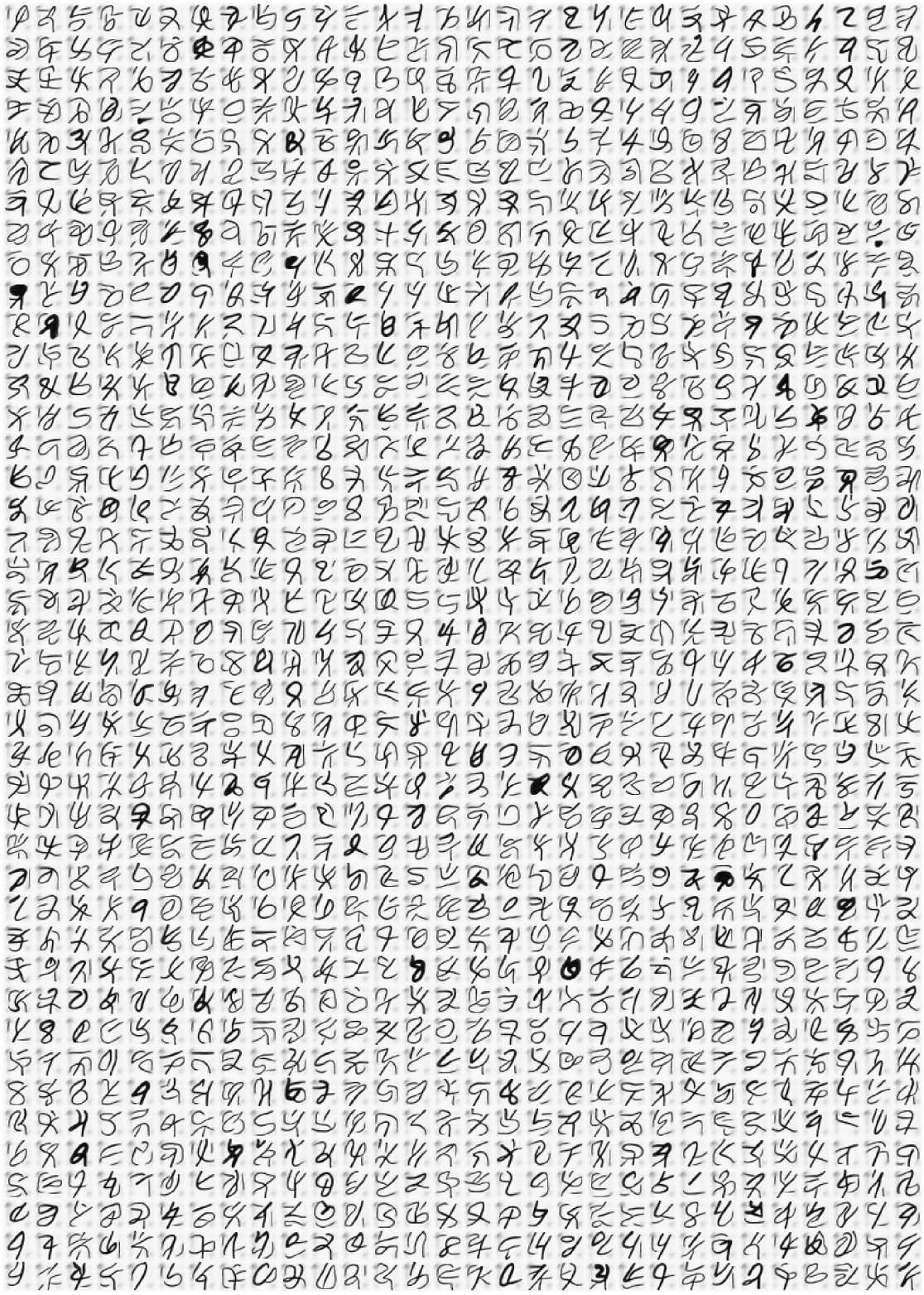


Figure 10: Attractors of NEBULA for $\sigma = 0.25$ ($\sigma_c \approx 0.29$).

7. Emergence of "creative memories"

We finish the paper with some surprising results on the emergence of highly-structured non-digit memories, named *"creative memories"*. The experiments designed in this section are a natural continuation of the experiments from the previous section. In its summary, NEBULA is "well behaved" when it is initialized at $X_i \sim f_X$. *But what if it is not initialized as such?* Here, we explore *i-sphere interactions* (see Definition 5) from a different angle by initializing NEBULA at a random point,

$$y_0 \sim \mathrm{Unif}([0,1]^d).$$

The results in Figures 9 and 10 are the strongest evidence for *i-sphere interactions* as being a good *abstraction*. In viewing Figures 9 and 10, consider Figure 2, but with *60000* highly overlapping i-spheres, and $-\nabla\phi$ on each i-sphere that should "point towards" X_i *in expectation*, where in addition, there are other complexities (touched upon in Remark 6) regarding the i-spheres themselves.

Several remarks are in order.

- The images shown in Figures 9 and 10 are *not* statistical samples from f_X nor f_Y.
- They are not the result of mass aggregation of the *kernel density estimator* $\widehat{f}_Y$. *Mode seeking* is a deep topic in the kernel density literature around the *mean shift algorithm* (Fukunaga and Hostetler, 1975; Cheng, 1995; Comaniciu and Meer, 2002), also around the intriguing topic of "ghost modes" (Carreira-Perpiñán and Williams, 2003). Of course, the *attractors* reported here has not been reported in the kernel density literature (see (Carreira-Perpiñán, 2015) and the examples therein for MNIST).
- The results presented are from all the random initializations in the *unit hypercube* from which we ran the algorithm, without any hand-picking!

8. Summary

In *neural empirical Bayes*, the smoothing of a random variable X to Y ($f = f_X * f_N$), denoising, and the (unnormalized) density estimation of the smoothed density f were unified in a single machinery. Its inner workings were captured symbolically by $X \rightleftharpoons Y$, as well as by a *Gedankenexperiment* with an "experimenter" in the school of Robbins (1956) and Robbins and Monro (1951).[4] In this machinery, the energy function is parametrized with a neural network and SGD becomes the engine for learning, whose end result is captured by $\nabla\phi(\cdot,\theta^*) \approx -\nabla \log f(\cdot)$. We proposed an approximative walk-jump sampling scheme which produces samples which are particularly appealing visually due to the denoising jump used. The energy function can further be used as an associative memory (NEBULA), which even has some "creative" capacities.

4. In our presentation, the smoothing part was *viewed* from the angle of *kernel density estimation*, with an important distinction that the kernel bandwidth was a *hyperparameter*. But on reflection, this is not necessary. Fundamentally, setting aside the "*neural* energy function", one can go on with our program by just knowing the (deep) machineries of *empirical Bayes* (Robbins, 1956) and *stochastic gradients* (Robbins and Monro, 1951).

Acknowledgement

This work was supported by the Canadian Institute for Advanced Research, the Gatsby Charitable Foundation, and the National Science Foundation through grant IIS-1718991. We especially thank Bruno Olshausen, Eero Simoncelli, and Francis Bach for discussions.

References

Harold Abelson and Gerald Jay Sussman. Structure and interpretation of computer programs. *MIT Press*, 1985.

Guillaume Alain and Yoshua Bengio. What regularized auto-encoders learn from the data-generating distribution. *Journal of Machine Learning Research*, 15(1):3563–3593, 2014.

Guillaume Alain, Yoshua Bengio, Li Yao, Jason Yosinski, Eric Thibodeau-Laufer, Saizheng Zhang, and Pascal Vincent. GSNs: generative stochastic networks. *Information and Inference: A Journal of the IMA*, 5(2):210–249, 2016.

Philip W Anderson. More is different. *Science*, 177(4047):393–396, 1972.

Sanjeev Arora, Nadav Cohen, and Elad Hazan. On the optimization of deep networks: Implicit acceleration by overparameterization. *arXiv preprint arXiv:1802.06509*, 2018.

Atilim Gunes Baydin, Barak A Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *Journal of Machine Learning Research*, 18(153), 2018.

Yoshua Bengio. Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, 2(1):1–127, 2009.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013a.

Yoshua Bengio, Li Yao, Guillaume Alain, and Pascal Vincent. Generalized denoising auto-encoders as generative models. In *Advances in Neural Information Processing Systems*, pages 899–907, 2013b.

Miguel Á Carreira-Perpiñán. A review of mean-shift algorithms for clustering. *arXiv preprint arXiv:1503.00687*, 2015.

Miguel Á Carreira-Perpiñán and Christopher KI Williams. On the number of modes of a gaussian mixture. In *International Conference on Scale-Space Theories in Computer Vision*, pages 625–640. Springer, 2003.

Rishidev Chaudhuri and Ila Fiete. Associative content-addressable networks with exponentially many robust stable states. *arXiv preprint arXiv:1704.02019*, 2017.

Yizong Cheng. Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):790–799, 1995.

Dorin Comaniciu and Peter Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 24(5):603–619, 2002.

Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*, 2018.

Keinosuke Fukunaga and Larry Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on information theory*, 21(1):32–40, 1975.

Christopher Hillar and Ngoc M Tran. Robust exponential memory in Hopfield networks. *arXiv preprint arXiv:1411.4625*, 2014.

Geoffrey E Hinton and Terrence J Sejnowski. Learning and relearning in Boltzmann machines. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, 1:282–317, 1986.

John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, 1982.

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(Apr):695–709, 2005.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT Press, 2009.

Dmitry Krotov and John J Hopfield. Dense associative memory for pattern recognition. In *Advances in Neural Information Processing Systems*, pages 1172–1180, 2016.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Michel Ledoux. *The concentration of measure phenomenon*. Number 89. American Mathematical Society, 2001.

David MacKay. *Information theory, inference and learning algorithms*. Cambridge University Press, 2003.

Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4):115–133, 1943.

Koichi Miyasawa. An empirical Bayes estimator of the mean of a normal population. *Bulletin of the International Statistical Institute*, 38(4):181–188, 1961.

Emanuel Parzen. On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, 1962.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. 2017.

Prajit Ramachandran, Barret Zoph, and Quoc V Le. Swish: a self-gated activation function. *arXiv preprint arXiv:1710.05941*, 7, 2017.

Martin Raphan and Eero P Simoncelli. Least squares estimation without priors or supervision. *Neural Computation*, 23(2):374–420, 2011.

Herbert Robbins. An empirical Bayes approach to statistics. In *Proc. Third Berkeley Symp.*, volume 1, pages 157–163, 1956.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, pages 400–407, 1951.

Saeed Saremi. On approximating ∇f with neural networks. *arXiv preprint arXiv:1910.12744*, 2019.

Saeed Saremi, Arash Mehrjou, Bernhard Schölkopf, and Aapo Hyvärinen. Deep energy estimator networks. *arXiv preprint arXiv:1805.08306*, 2018.

Lawrence K Saul and Sam T Roweis. Think globally, fit locally: unsupervised learning of low dimensional manifolds. *Journal of Machine Learning Research*, 4(Jun):119–155, 2003.

Terence Tao. *Topics in random matrix theory.* American Mathematical Society, 2012.

Aad van der Vaart. *Asymptotic statistics.* Cambridge University Press, 2000.

Nicolaas Godfried van Kampen. *Stochastic processes in physics and chemistry*, volume 1. Elsevier, 1992.

Vladimir Vapnik. *The nature of statistical learning theory.* Springer, 1995.

Roman Vershynin. *High-dimensional probability: An introduction with applications in data science.* Cambridge University Press, 2018.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011.

Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press, 2019.

Martin J Wainwright and Michael I Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305, 2008.