Minimum Probability Flow Learning

Jascha Sohl-Dickstein[ad1*], **Peter Battaglino**[bd2*] **and Michael R. DeWeese**[bcd3]
[a]Biophysics Graduate Group, [b]Department of Physics, [c]Helen Wills Neuroscience Institute
[d]Redwood Center for Theoretical Neuroscience
University of California, Berkeley, 94720
[1]jascha@berkeley.edu, [2]pbb@berkeley.edu,
[3]deweese@berkeley.edu, [*]*These authors contributed equally.*

Abstract

Learning in probabilistic models is often hampered by the general intractability of the normalization factor and its derivatives. Here we propose a new learning technique that obviates the need to compute an intractable normalization factor or sample from the equilibrium distribution of the model. This is achieved by establishing dynamics that would transform the observed data distribution into the model distribution, and then setting as the objective the minimization of the initial flow of probability away from the data distribution. Score matching, minimum velocity learning, and certain forms of contrastive divergence are shown to be special cases of this learning technique. We demonstrate the application of minimum probability flow learning to parameter estimation in Ising models, deep belief networks, multivariate Gaussian distributions and a continuous model with a highly general energy function defined as a power series. In the Ising model case, minimum probability flow learning outperforms current state of the art techniques by approximately two orders of magnitude in learning time, with comparable error in recovered parameters. It is our hope that this technique will alleviate existing restrictions on the classes of probabilistic models that are practical for use.

1 Introduction

Estimating parameters for probabilistic models is a fundamental problem in many scientific and engineering disciplines. Unfortunately, most probabilistic learning techniques require calculating the normalization factor, or partition function, of the probabilistic model in question, or at least calculating its gradient. For the overwhelming majority of models there are no known analytic solutions, confining us to the highly restrictive subset of probabilistic models that *can* be analytically solved, or those that can be made tractable using known approximate learning techniques. Thus, development of new techniques for parameter estimation in currently intractable probabilistic models has the potential to be of great benefit, lifting near ubiquitous restrictions on how we are able to model the world.

Many approaches exist for approximate learning, including mean field theory and its expansions, variational Bayes techniques and a plethora of sampling or numerical integration based methods [22, 10, 9, 5]. Of particular interest are contrastive divergence (CD), developed by Welling, Hinton and Carreira-Perpiñán [23, 4], Hyvärinen's score matching (SM) [7], and the minimum velocity learning framework proposed by Movellan [14, 13, 15].

Contrastive divergence [23, 4] is a variation on steepest gradient descent of the maximum (log) likelihood (ML) objective function. Rather than integrating over the full model model distribution, CD approximates the partition function term in the gradient by averaging over the distribution real-

ized after taking a few Markov chain Monte Carlo (MCMC) steps away from the data distribution[1]. Qualitatively, one can imagine that the data distribution is contrasted against a distribution which has evolved a small distance towards the model distribution, whereas it would usually be contrasted against the true model distribution. Although CD is not guaranteed to converge to the right answer, or even to a fixed point, it has proven to be an effective and fast heuristic for parameter estimation [11, 24].

Score matching, developed by Aapo Hyvärinen [7], is a method that learns parameters in a probabilistic model using only derivatives of the energy function evaluated over the data distribution (see Equation (12)). This sidesteps the need to explicitly sample or integrate over the model distribution. In score matching one minimizes the expected square distance of the score function with respect to spatial coordinates given by the data distribution from the similar score function given by the model distribution. It can be seen as an integration of the contrastive divergence gradient for infinitesimal Langevin dynamics [8], as the limit of approximating the model distribution by patching together cutouts of the model distribution around each data point [21], and finally as equivalent to minimum velocity learning [14].

Minimum velocity learning is an approach recently proposed by Movellan [14] that recasts a number of the ideas behind CD, treating the minimization of the initial dynamics away from the data distribution as the goal itself rather than a surrogate for it. Movellan's proposal is that rather than directly minimize the difference between the data and the model, one introduces system dynamics that have the model as their equilibrium distribution, and minimizes the initial flow of probability away from the data under those dynamics. If the model looks exactly like the data there will be no flow of probability, and if model and data are similar the flow of probability will tend to be minimal. Movellan applies this intuition to the specific case of distributions over continuous state spaces evolving via diffusion dynamics. The velocity in minimum velocity learning is the difference in average drift velocities between particles diffusing under the model distribution and particles diffusing under the data distribution.

Here we provide a framework, applicable to *any* parametric model, of which minimum velocity, certain forms of CD, and SM are all special cases, and which is in many situations more powerful than any of these algorithms. This framework extends the ideas behind minimum velocity learning to arbitrary state spaces and a far broader class of dynamics. We show that learning under this framework is effective and fast in a number of cases: Ising models, deep belief networks (DBN), multidimensional Gaussian distributions, and a complicated two-dimensional continuous distribution.

2 Minimum probability flow

Our goal is to find the parameters that cause a probabilistic model to best agree with a set of (assumed iid) observations of the state of a system. We will do this by proposing dynamics that guarantee the transformation of the data distribution into the model distribution, and then minimizing the magnitude of the initial flow of probability away from the data distribution.

2.1 Distributions

The data distribution is represented by a vector $\mathbf{p}^{(0)}$, with $p_i^{(0)}$ the probability of observing the system in a state i. The superscript (0) represents time $t = 0$ under the system dynamics, as will

[1] The update rule for gradient descent of the negative log likelihood, or maximum likelihood objective function, is

$$\Delta\theta \propto \frac{\partial \left[\sum_i p_i^{(0)} \log p_i^{(\infty)}(\theta)\right]}{\partial\theta} = -\sum_i \frac{\partial E_i(\theta)}{\partial\theta} p_i^{(0)} + \sum_i \frac{\partial E_i(\theta)}{\partial\theta} p_i^{(\infty)}(\theta),$$

where $\mathbf{p}^{(0)}$ and $\mathbf{p}^{(\infty)}(\theta)$ represent the data distribution and model distribution, respectively, $\mathbf{E}(\theta)$ is the energy function associated with the model distribution and i indexes the states of the system (see Section 2.1). The second term in this gradient can be extremely difficult to compute (costing in general an amount of time exponential in the dimensionality of the system). Under contrastive divergence $p_i^{(\infty)}(\theta)$ is replaced by samples only a few Monte Carlo steps away from the data.

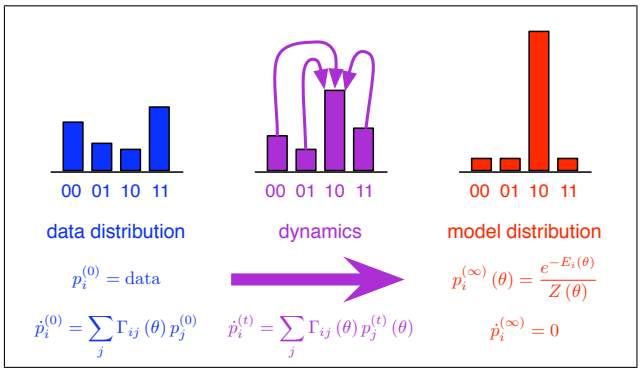

Figure 1: Dynamics of minimum probability flow learning. Model dynamics represented by the probability flow matrix $\mathbf{\Gamma}$ (*middle*) determine how probability flows from the empirical histogram of the sample data points (*left*) to the equilibrium distribution of the model (*right*) after a sufficiently long time. In this example there are only four possible states for the system, which consists of a pair of binary variables, and the particular model parameters favor state 10 whereas the data falls mostly on other states.

be described in more detail in Section 2.2. If the observations were of a two variable binary system, then $\mathbf{p}^{(0)}$ would have four entries representing the probabilities of observing states 00, 01, 10 and 11.

Our goal is to find the parameters θ that cause a model distribution $\mathbf{p}^{(\infty)}(\theta)$ to best match the data distribution $\mathbf{p}^{(0)}$. Without loss of generality, we assume the model distribution to be of the form

$$p_i^{(\infty)}(\theta) = \frac{\exp\left(-E_i(\theta)\right)}{Z(\theta)}, \tag{1}$$

where $\mathbf{E}(\theta)$ is referred to as the energy function, and the normalizing factor $Z(\theta)$ is called the partition function,

$$Z(\theta) = \sum_i \exp\left(-E_i(\theta)\right) \tag{2}$$

(here we have set the "temperature" of the system to 1). The superscript (∞) indicates that this is the equilibrium distribution reached after running the dynamics for infinite time.

2.2 Dynamics

We wish to generalize Movellan's diffusion dynamics to arbitrary state spaces. To accomplish this, we observe that diffusion dynamics are a special case of dynamics governed by a master equation that enforces conservation of probability [16]:

$$\dot{p}_i^{(t)} = \sum_{j \neq i} \Gamma_{ij}(\theta)\, p_j^{(t)} - \sum_{j \neq i} \Gamma_{ji}(\theta)\, p_i^{(t)}, \tag{3}$$

where $\dot{p}_i^{(t)} = \frac{\partial p_i^{(t)}}{\partial t}$ is the rate of change of probability of state i with time. Transition rates $\Gamma_{ij}(\theta)$ give the rate at which probability will flow from a state j into a state i. The first term of Equation (3) represents flow of probability out of other states j into the state i, and the second represents flow out of i into other states j. The dependence on θ results from the requirement that the dynamics we choose cause $\mathbf{p}^{(t)}$ to flow to the equilibrium distribution $\mathbf{p}^{(\infty)}(\theta)$. For readability, explicit dependence on θ will be dropped except where specifically relevant. If we choose the diagonal of $\mathbf{\Gamma}$ to obey $\Gamma_{ii} = -\sum_{j \neq i} \Gamma_{ji}$, then we can write the dynamics as

$$\dot{\mathbf{p}}^{(t)} = \mathbf{\Gamma}\mathbf{p}^{(t)} \tag{4}$$

(see Figure 1). The unique solution for $\mathbf{p}^{(t)}$ is[2]

$$\mathbf{p}^{(t)} = \exp\left(\mathbf{\Gamma} t\right) \mathbf{p}^{(0)}. \tag{5}$$

2.3 Detailed Balance

$\mathbf{\Gamma}$ must be chosen such that the dynamics in Equation (4) converge to the model distribution. One way to guarantee this is by choosing $\mathbf{\Gamma}$ such that it satisfies detailed balance for the model distribution $\mathbf{p}^{(\infty)}$, and such that there is a path through $\mathbf{\Gamma}$ allowing mixing between any two states. Note that there is no need to restrict the dynamics defined by $\mathbf{\Gamma}$ to those of any real physical process, such as diffusion. Detailed balance requires that at equilibrium the probability flow from state i into state j equals the probability flow from j into i,

$$\Gamma_{ji}\, p_i^{(\infty)}\left(\theta\right) = \Gamma_{ij}\, p_j^{(\infty)}\left(\theta\right), \tag{6}$$

which can be rewritten as

$$\frac{\Gamma_{ij}}{\Gamma_{ji}} = \frac{p_i^{(\infty)}\left(\theta\right)}{p_j^{(\infty)}\left(\theta\right)} = \exp\left[E_j\left(\theta\right) - E_i\left(\theta\right)\right]. \tag{7}$$

$\mathbf{\Gamma}$ is underconstrained by the above equation. Motivated by symmetry and aesthetics, we choose as the form for the (non-zero, non-diagonal) entries in $\mathbf{\Gamma}$

$$\Gamma_{ij} = \exp\left[\frac{1}{2}\left(E_j\left(\theta\right) - E_i\left(\theta\right)\right)\right] \qquad\qquad (i \neq j). \tag{8}$$

The choice $\Gamma_{ij} = \Gamma_{ji} = 0$ also satisfies Equation (6), allowing a sparse population of $\mathbf{\Gamma}$ for purposes of computational tractability. Theoretically, to guarantee convergence to the model distribution, the non-zero elements of $\mathbf{\Gamma}$ must be chosen such that, given sufficient time, probability can flow between any pair of states. In practice, we will only need to consider a small fraction of the non-zero elements in $\mathbf{\Gamma}$ (see Section 2.5).

2.4 Objective Function

The goal is to minimize the initial flow of probability away from the data distribution (Figure 2). Although other objective functions are possible for a minimum probability flow approach, we have found the L1 norm to be particularly effective:

$$\hat{\theta} \;=\; \arg\min_{\theta} K\left(\theta\right), \tag{9}$$

$$K\left(\theta\right) \;=\; \left\|\dot{\mathbf{p}}^{(0)}\left(\theta\right)\right\|_1 = \sum_i \left|\dot{p}_i^{(0)}\left(\theta\right)\right|. \tag{10}$$

This objective function is uniquely zero when $\mathbf{p}^{(0)}$ and $\mathbf{p}^{(\infty)}\left(\theta\right)$ are exactly equal (although in general the relationship of $\hat{\theta}$ to the maximum likelihood solution is less clear). Some algebra gives the learning gradient with respect to θ:

$$\frac{\partial K}{\partial \theta} \;=\; \frac{1}{2}\sum_{i,j} p_j^{(0)} \Gamma_{ij}\left(\theta\right)\left[\frac{\partial E_j\left(\theta\right)}{\partial \theta} - \frac{\partial E_i\left(\theta\right)}{\partial \theta}\right]\left[\operatorname{sgn}\left(\dot{p}_i^{(0)}\left(\theta\right)\right) - \operatorname{sgn}\left(\dot{p}_j^{(0)}\left(\theta\right)\right)\right]. \tag{11}$$

Note that Equations (9) through (11) do not depend on the partition function $Z\left(\theta\right)$ or its derivatives.

Under the constraint that $\mathbf{\Gamma}$ does not allow probability to flow directly from one state with data to another - nearly always satisfied when the number of system states is much larger than the number of states with data - Equation (9) is equivalent to minimizing the initial rate of growth of the KL divergence between $\mathbf{p}^{(0)}$ and $\mathbf{p}^{(t)}$, $\left.\frac{\partial D_{KL}\left(\mathbf{p}^{(0)} || \mathbf{p}^{(t)}\right)}{\partial t}\right|_{t=0}$ (see Appendix A).

Under the same constraint, the minimum probability flow objective function $K\left(\theta\right)$ is convex for all models $\mathbf{p}^{(\infty)}\left(\theta\right)$ in the exponential family - that is, models who's energy function $\mathbf{E}\left(\theta\right)$ is linear in their parameters θ [12] (see Appendix B).

[2] The form chosen for $\mathbf{\Gamma}$ in Equation (4), coupled with the satisfaction of detailed balance and ergodicity introduced in section 2.3, guarantees that there is a unique eigenvector $\mathbf{p}^{(\infty)}$ of $\mathbf{\Gamma}$ with eigenvalue zero, and that all other eigenvalues of $\mathbf{\Gamma}$ have negative real parts.

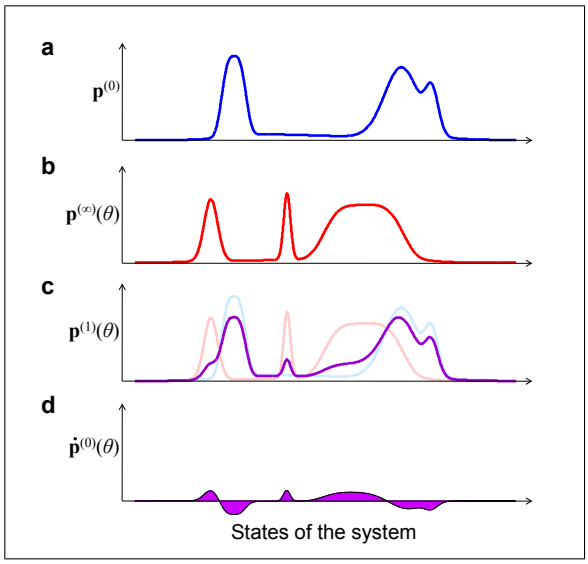

Figure 2: An illustration of the minimum probability flow objective function, which minimizes the initial flow of probability away from the data. **a**. Empirical histogram of the observed data over all possible states of the system. **b**. Model distribution that the dynamics would converge to if allowed to run for a sufficiently long time. The dynamics and model distribution are both functions of the model parameters (θ). Our goal is to make the model, or equilibrium, distribution as much like the data distribution as possible. **c**. The distribution after starting at the data and running the dynamics for a short time period. **d**. The temporal derivative of the probability distribution, or probability flow, at $t = 0$. Learning is achieved by changing the model parameters so as to minimize the shaded region of this graph.

2.5 Tractability

The vector $\mathbf{p}^{(0)}$ is typically huge, as is $\mathbf{\Gamma}$ (*e.g.*, 2^N and $2^N \times 2^N$, respectively, for an N-bit binary system). Naïvely, this would seem to prohibit evaluation and minimization of the objective function. Fortunately, all the elements in $\mathbf{p}^{(0)}$ not corresponding to observations are zero. Since our objective function is only evaluated at time $t = 0$ this allows us to ignore all those Γ_{ij} for which no data point exists at state j. Additionally, there is a great deal of flexibility as far as which elements of $\mathbf{\Gamma}$ can be set to zero. By populating $\mathbf{\Gamma}$ so as to connect each state to a small fixed number of additional states, the cost of the algorithm in both memory and time is $\mathcal{O}(M)$, where M is the number of observed data points, and does not depend on the number of system states.

2.6 Continuous Systems

Although we have motivated this technique using systems with a large, but finite, number of states, it generalizes in a straightforward manner to continuous systems. The flow matrix $\mathbf{\Gamma}$ and distribution vectors $\mathbf{p}^{(t)}$ transition from being very large to being infinite in size. $\mathbf{\Gamma}$ can still be chosen to connect each state to a small, finite, number of additional states however, and only outgoing probability flow from states with data contributes to the objective function, so the cost of learning remains largely unchanged.

In addition, for a particular pattern of connectivity in $\mathbf{\Gamma}$ this objective function, like Movellan's [14], reduces to score matching [7] (other connectivity patterns reduce to alternate forms). Taking the limit of connections between all states within a small distance ϵ of each other, and then Taylor expanding in ϵ, one can show that, up to an overall constant and scaling factor

$$K \approx K_{\mathrm{SM}} = \sum_{i \in \{\mathrm{samples}\}} \left[\frac{1}{2} \nabla E(x_i) \cdot \nabla E(x_i) - \nabla^2 E(x_i) \right]. \tag{12}$$

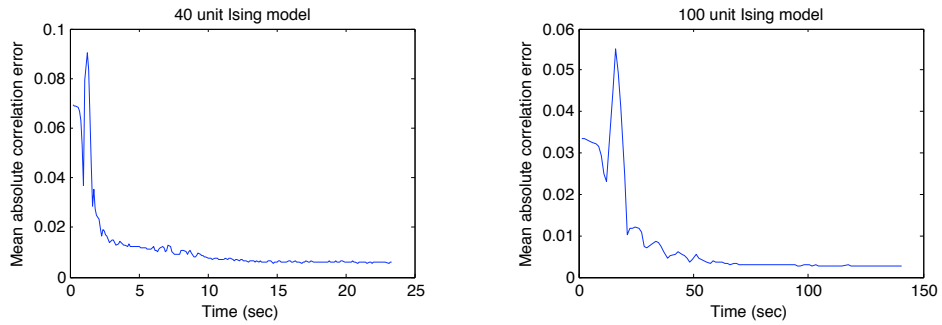

Figure 3: A demonstration of rapid fitting of the Ising model by minimum probability flow learning. The mean absolute error in the learned model's correlation matrix is shown as a functions of learning time for 40 and 100 unit fully connected Ising models. Convergence is reached in about 15 seconds for $20,000$ samples from the 40 unit model *(left)* and in about 1 minute for $100,000$ samples from the 100 unit model *(right)*. Details of the 100 unit model can be seen in Figure 4.

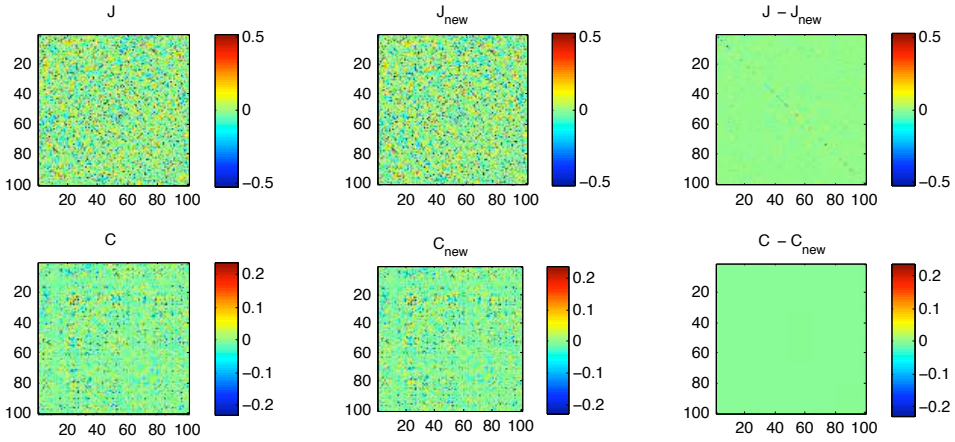

Figure 4: An example 100 unit Ising model fit using minimum probability flow learning. *(left)* Randomly chosen Gaussian coupling matrix J (top) with variance 0.04 and associated correlation matrix C (bottom) for a 100 unit, fully-connected Ising model. The diagonal has been removed from the correlation matrix C for increased visibility. *(center)* The recovered coupling and correlation matrices after minimum probability flow learning on $100,000$ samples from the model in the left panels. *(right)* The error in recovery of the coupling and correlation matrices.

This reproduces the link discovered by Movellan [14] between diffusion dynamics over continuous spaces and score matching.

3 Experimental Results

Matlab code implementing minimum probability flow learning for each of the following cases is available upon request. A public toolkit is under construction.

All minimization was performed using Mark Schmidt's remarkably effective minFunc [17].

3.1 Ising model

The Ising model has a long and storied history in physics [3] and machine learning [1] and it has recently been found to be a surprisingly useful model for networks of neurons in the retina [18, 20]. The ability to fit Ising models to the activity of large groups of simultaneously recorded neurons is

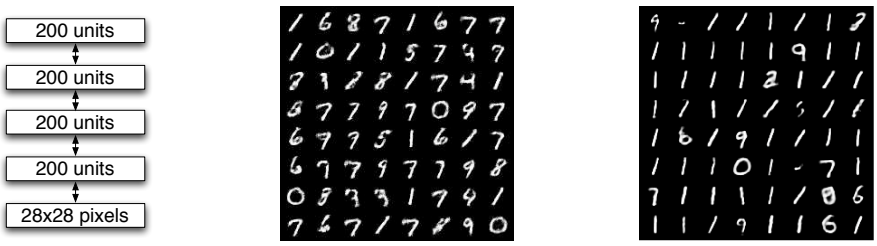

Figure 5: A deep belief network trained using minimum probability flow learning and contrastive divergence. *(left)* A four layer deep belief network was trained on the MNIST postal hand written digits dataset. *(center)* Confabulations after training via minimum probability flow learning. A reasonable probabilistic model for handwritten digits has been learned. *(right)* Confabulations after training via single step CD. Note the uneven distribution of digit occurrences.

of current interest given the increasing number of these types of data sets from the retina, cortex and other brain structures.

We fit an Ising model (fully visible Boltzmann machine) of the form

$$p^{(\infty)}(\mathbf{x};\mathbf{J}) = \frac{1}{Z(\mathbf{J})} \exp\left[-\sum_{i,j} J_{ij} x_i x_j\right] \tag{13}$$

to a set of N d-element iid data samples $\left\{x^{(i)} | i = 1...N\right\}$ generated via Gibbs sampling from an Ising model as described below, where each of the d elements of $\mathbf{x}$ is either 0 or 1. Because each $x_i \in \{0, 1\}$, $x_i^2 = x_i$, we can write the energy function as

$$E(\mathbf{x};\mathbf{J}) = \sum_{i,j \neq i} J_{ij} x_i x_j + \sum_i J_{ii} x_i. \tag{14}$$

The probability flow matrix $\mathbf{\Gamma}$ has $2^N \times 2^N$ elements, but we allow only elements corresponding to transitions into states a single bit-flip away to be non-zero.

Figure 3 shows the average error in predicted correlations as a function of learning time for $20,000$ samples from a 40 unit, fully connected Ising model. The J_{ij} used were graciously provided by Broderick and coauthors, and were identical to those used for synthetic data generation in the 2008 paper "Faster solutions of the inverse pairwise Ising problem" [2]. Training was performed on $20,000$ samples so as to match the number of samples used in section III.A. of Broderick et al. Note that given sufficient samples, the minimum probability flow algorithm would converge exactly to the right answer, as learning in the Ising model is convex (Appendix B), and has its global minimum at the true solution. On an 8 core 2.33 GHz Intel Xeon, the learning converges in about 15 seconds. Broderick et al. perform a similar learning task on a 100-CPU grid computing cluster, with a convergence time of approximately 200 seconds.

Similar learning was performed for $100,000$ samples from a 100 unit, fully connected, Ising model. A coupling matrix was chosen with elements randomly drawn from a Gaussian with mean 0 and variance 0.04. Using the minimum probability flow learning technique, learning took approximately 1 minute, compared to roughly 12 hours for a 100 unit (nearest neighbor coupling only) model of retinal data [19] (personal communication, J. Shlens). Figure 4 demonstrates the recovery of the coupling and correlation matrices for our fully connected Ising model, while Figure 3 shows the time course for learning.

3.2 Deep Belief Network

As a demonstration of learning on a more complex discrete valued model, we trained a 4 layer deep belief network (DBN) [6] on MNIST handwritten digits. A DBN consists of stacked restricted Boltzmann machines (RBMs), such that the hidden layer of one RBM forms the visible layer of the

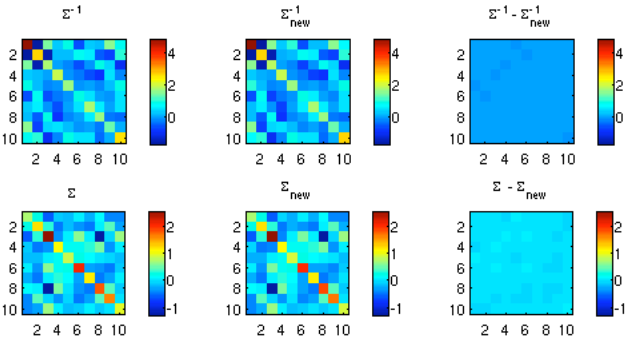

Figure 6: A continuous state space model fit using minimum probability flow learning. *(left)* Randomly chosen coupling matrix $\mathbf{\Sigma}^{-1}$ and associated covariance matrix $\mathbf{\Sigma}$ for a 10 dimensional Gaussian distribution. *(center)* The recovered coupling matrix $\mathbf{\Sigma}_{\text{new}}^{-1}$ and associated covariance matrix $\mathbf{\Sigma}_{\text{new}}$ after minimum probability flow learning on $10,000$ samples from the model in (left). *(right)* The error in recovery of the coupling and covariance matrices.

next. Each RBM has the form:

$$p^{(\infty)}(\mathbf{x}_{\text{vis}}, \mathbf{x}_{\text{hid}}; \mathbf{W}) = \frac{1}{Z(\mathbf{W})} \exp\left[-\sum_{i,j} W_{ij} x_{\text{vis},i} x_{\text{hid},j}\right], \tag{15}$$

$$p^{(\infty)}(\mathbf{x}_{\text{vis}}; \mathbf{W}) = \frac{1}{Z(\mathbf{W})} \exp\left[\sum_{j} \log\left(1 + \exp\left[-\sum_{i} W_{ij} x_{\text{vis},i}\right]\right)\right]. \tag{16}$$

Note that sampling-free application of the minimum probability flow algorithm requires analytically marginalizing over the hidden units. RBMs were trained in sequence, starting at the bottom layer, on 10,000 samples from the MNIST postal hand written digits data set. As in the Ising case, the probability flow matrix $\mathbf{\Gamma}$ was populated so as to connect every state to all states which differed by only a single bit flip. Training was performed by both minimum probability flow and single step CD to allow a simple comparison of the two techniques (note that CD turns into full ML learning as the number of steps is increased, and that the quality of the CD answer can thus be improved at the cost of computational time by using many-step CD).

Confabulations were performed by Gibbs sampling from the top layer RBM, then propagating each sample back down to the pixel layer by way of the conditional distribution $p^{(\infty)}(\mathbf{x}_{\text{vis}}|\mathbf{x}_{\text{hid}}; \mathbf{W}^k)$ for each of the intermediary RBMs, where k indexes the layer in the stack. As shown in Figure 5, minimum probability flow learned a good model of handwritten digits.

3.3 Gaussian

As an example of minimum probability flow learning applied to continuous models, we fit a multivariate Gaussian distribution to synthetic data. The model distribution has the form

$$p^{(\infty)}(\mathbf{x}; \mathbf{\Sigma}^{-1}) = \frac{1}{Z\left(\mathbf{\Sigma}^{-1}\right)} \exp\left[-\frac{1}{2}\mathbf{x}^T \mathbf{\Sigma}^{-1} \mathbf{x}\right], \tag{17}$$

with vector $\mathbf{x}$ and coupling matrix $\mathbf{\Sigma}^{-1}$. We fit to $10,000$ iid samples from a 10-dimensional Gaussian distribution. The probability flow matrix $\mathbf{\Gamma}$ was populated so as to connect every state to 20 additional states, chosen from a Gaussian distribution with variance 0.01 centered on the state. Results are shown in Figure 6.

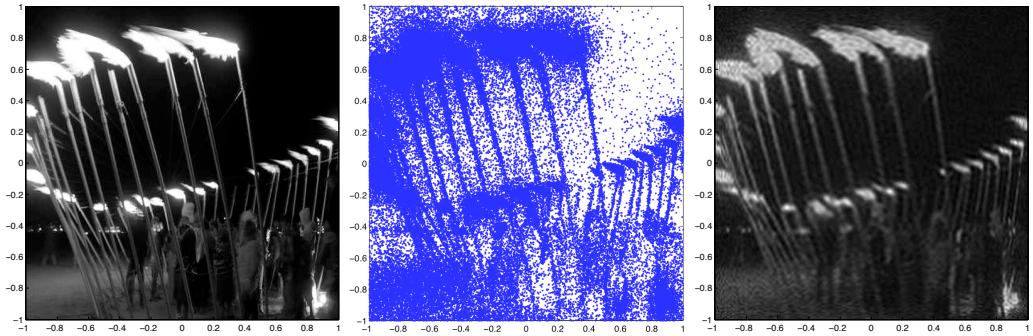

Figure 7: A highly unconstrained, difficult to normalize, model fit using minimum probability flow learning. *(left)* Histogram of a complicated two-dimensional continuous distribution $p^{(0)}(x,y)$, $(x,y) \in [-1,1]^2$. The probability of observing a sample (x,y) is proportional to the pixel value at location (x,y) in the histogram. Note, the image represents a distribution over (x,y) values, not a sample from a distribution. *(center)* Scatter plot of 100,000 samples drawn from the distribution in (left) *(right)* Histogram of learned distribution $p^{(\infty)}(x,y;\theta)$ trained in batch mode on groups of 100,000 samples from the distribution in (left), where $p^{(\infty)}(x,y;\theta) \propto \exp\left(-\sum_{m=0}^{128}\sum_{n=0}^{128}\theta_{mn}L_m(x)L_n(y)\right)$ and $L_m(x)$ is the mth Legendre polynomial in x.

3.4 Power Series Energy Function

To demonstrate minimum probability flow's effectiveness in an extremely flexible, difficult to normalize, model, we learned parameters θ for a two-dimensional continuous distribution of the form

$$p^{(\infty)}(x,y;\theta) = \frac{1}{Z(\theta)} \exp\left[-\sum_{m,n=0}^{M}\theta_{mn}L_m(x)L_n(y)\right], \tag{18}$$

where $L_m(x)$ is the mth order Legendre polynomial in x, $(x,y) \in [-1,1]^2$, M is the maximum polynomial order, and $Z(\theta)$ is the normalization factor. We fit an $M = 128$ distribution using consecutive line searches on batches of $100,000$ iid samples from the distribution shown on the left of Figure 7. The probability flow matrix $\mathbf{\Gamma}$ was populated so as to connect every state with 20 other states, chosen from a uniform distribution in the range $[-1,1]^2$. Figure 7 shows a histogram of the data distribution $p^{(0)}(x,y;\theta)$ compared to a histogram of the learned Legendre function expansion $p^{(\infty)}(x,y;\theta)$.

4 Summary

We have presented a novel framework for efficient learning in the context of any parametric model. This method was inspired by the minimum velocity approach developed by Movellan, and it reduces to that technique as well as to score matching and some forms of contrastive divergence under suitable choices for the dynamics and state space. By decoupling the dynamics from any specific physical process, such as diffusion, and focusing on the initial flow of probability from the data to a subset of other states chosen in part for their utility and convenience, we have arrived at a framework that is not only more general than previous approaches, but also potentially much more powerful. We expect that this framework will render some previously intractible models more amenable to estimation.

Acknowledgments

We would like to thank Javier Movellan for sharing a work in progress; Tamara Broderick, Miroslav Dudík, Gašper Tkačik, Robert E. Schapire and William Bialek for use of their Ising model coupling parameters; Jonathon Shlens for useful discussion and ground truth for his Ising model convergence times; Bruno Olshausen, Anthony Bell, Christopher Hillar, Charles Cadieu, Kilian Koepsell and the

rest of the Redwood Center for many useful discussions and for comments on earlier versions of the manuscript; Ashvin Vishwanath for useful discussion; and the Canadian Institute for Advanced Research - Neural Computation and Perception Program for their financial support (JSD).

APPENDICES

A Connection to KL Divergence

We want to measure the rate of growth of the KL divergence at time 0, $\frac{\partial D_{KL}}{\partial t}\big|_{t=0}$.

The KL divergence between the data distribution, and the distribution resulting after running the dynamics for a time t, is

$$D_{KL} = \sum_i p_i^{(0)} \log p_i^{(0)} - \sum_i p_i^{(0)} \log p_i^{(t)}. \tag{A-1}$$

Note that the terms for which $p_i^{(0)} = 0$ will *never* contribute to this sum. To make this explicit, we rewrite the sum as being over the set of states which are non-zero in $p_i^{(0)}$. This set is

$$D = \left\{ i : p_i^{(0)} \neq 0 \right\}. \tag{A-2}$$

We also note the complement of this set,

$$D^C = \left\{ i : p_i^{(0)} = 0 \right\}. \tag{A-3}$$

This makes the KL divergence

$$D_{KL} = \sum_{i \in D} p_i^{(0)} \log p_i^{(0)} - \sum_{i \in D} p_i^{(0)} \log p_i^{(t)}. \tag{A-4}$$

The derivative is

$$\begin{aligned}
\frac{\partial D_{KL}}{\partial t} &= -\sum_{i \in D} p_i^{(0)} \frac{1}{p_i^{(t)}} \frac{\partial p_i^{(t)}}{\partial t} && \text{(A-5)} \\
&= -\sum_{i \in D} \frac{p_i^{(0)}}{p_i^{(t)}} \sum_{j \neq i} \left[\Gamma_{ij} p_j^{(t)} - \Gamma_{ji} p_i^{(t)} \right] && \text{(A-6)} \\
&= \sum_{i \in D} \sum_j \delta_{i \neq j} \frac{p_i^{(0)}}{p_i^{(t)}} \left[\Gamma_{ji} p_i^{(t)} - \Gamma_{ij} p_j^{(t)} \right] && \text{(A-7)} \\
&= \sum_{i \in D} \sum_{j \in D} \delta_{i \neq j} \frac{p_i^{(0)}}{p_i^{(t)}} \Gamma_{ji} p_i^{(t)} + \sum_{i \in D} \sum_{j \in D^C} \frac{p_i^{(0)}}{p_i^{(t)}} \Gamma_{ji} p_i^{(t)} && \text{(A-8)} \\
&\quad - \sum_{i \in D} \sum_{j \in D} \delta_{i \neq j} \frac{p_i^{(0)}}{p_i^{(t)}} \Gamma_{ij} p_j^{(t)} - \sum_{i \in D} \sum_{j \in D^C} \frac{p_i^{(0)}}{p_i^{(t)}} \Gamma_{ij} p_j^{(t)}.
\end{aligned}$$

In the last line the sum over all j has been broken into a sum over D and its complement D^C. We evaluate the derivative at $t = 0$

$$\begin{aligned}
\frac{\partial D_{KL}}{\partial t}\Big|_{t=0} &= \sum_{i \in D} \sum_{j \in D} \delta_{i \neq j} \Gamma_{ji} p_i^{(0)} + \sum_{i \in D} \sum_{j \in D^C} \Gamma_{ji} p_i^{(0)} && \text{(A-9)} \\
&\quad - \sum_{i \in D} \sum_{j \in D} \delta_{i \neq j} \Gamma_{ij} p_j^{(0)} - \sum_{i \in D} \sum_{j \in D^C} \Gamma_{ij} p_j^{(0)}.
\end{aligned}$$

We can simplify this by noting that the following terms are 0:

$$\begin{aligned}
\sum_{i \in D} \sum_{j \in D} \delta_{i \neq j} \Gamma_{ji} p_i^{(0)} - \sum_{i \in D} \sum_{j \in D} \delta_{i \neq j} \Gamma_{ij} p_j^{(0)} &= 0 && \text{(A-10)} \\
\sum_{i \in D} \sum_{j \in D^C} \Gamma_{ij} p_j^{(0)} &= 0 && \text{(A-11)}
\end{aligned}$$

This means that the rate of growth of the KL divergence at the data distribution, $t = 0$, is

$$\left.\frac{\partial D_{KL}}{\partial t}\right|_{t=0} = \sum_{i \in D} \sum_{j \in D^C} \Gamma_{ji} p_i^{(0)}. \tag{A-12}$$

That is, the rate of growth of the KL divergence is equal to the rate of probability flow from states with data to those without.

This is equivalent to the minimum probability flow L1 objective function in the usual case that $\mathbf{\Gamma}$ does not allow probability to flow directly from one state with data to another.

B Convexity

As observed by Macke and Gerwinn [12], Equation (A-12) is convex for models in the exponential family.

We wish to minimize

$$K = \sum_{i \in D} \sum_{j \in D^C} \Gamma_{ji} p_i^{(0)}. \tag{B-1}$$

K has derivative

$$\frac{\partial K}{\partial \theta_m} = \sum_{i \in D} \sum_{j \in D^c} \left(\frac{\partial \Gamma_{ij}}{\partial \theta_m} \right) p_i^{(0)} \tag{B-2}$$

$$= \frac{1}{2} \sum_{i \in D} \sum_{j \in D^c} \Gamma_{ij} \left(\frac{\partial E_j}{\partial \theta_m} - \frac{\partial E_i}{\partial \theta_m} \right) p_i^{(0)}, \tag{B-3}$$

and Hessian

$$\frac{\partial^2 K}{\partial \theta_m \partial \theta_n} = \frac{1}{4} \sum_{i \in D} \sum_{j \in D^c} \Gamma_{ij} \left(\frac{\partial E_j}{\partial \theta_m} - \frac{\partial E_i}{\partial \theta_m} \right) \left(\frac{\partial E_j}{\partial \theta_n} - \frac{\partial E_i}{\partial \theta_n} \right) p_i^{(0)} \tag{B-4}$$

$$+ \frac{1}{2} \sum_{i \in D} \sum_{j \in D^c} \Gamma_{ij} \left(\frac{\partial^2 E_j}{\partial \theta_m \partial \theta_n} - \frac{\partial^2 E_i}{\partial \theta_m \partial \theta_n} \right) p_i^{(0)}. \tag{B-5}$$

The first term is a weighted sum of outer products, with non-negative weights $\frac{1}{4} \Gamma_{ij} p_i^{(0)}$, and is thus positive semidefinite. The second term is 0 for models in the exponential family (those with energy functions linear in their parameters).

Parameter estimation for models in the exponential family is therefore convex using minimum probability flow learning, in the commonly satisfied limit that $\mathbf{\Gamma}$ does not directly connect any two data points.

References

[1] D H Ackley, G E Hinton, and T J Sejnowski. A learning algorithm for Boltzmann machines. *Cognitive Science*, 9(2):147–169, 1985.

[2] T Broderick, M Dudík, G Tkačik, R Schapire, and W Bialek. Faster solutions of the inverse pairwise Ising problem. *E-print arXiv*, Jan 2007.

[3] S G Brush. History of the Lenz-Ising model. *Reviews of Modern Physics*, 39(4):883–893, Oct 1967.

[4] M A Carreira-Perpiñán and G E Hinton. On contrastive divergence (CD) learning. *Technical report, Dept. of Computere Science, University of Toronto*, 2004.

[5] S Haykin. *Neural networks and learning machines; 3rd edition*. Prentice Hall, 2008.

[6] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554, Jul 2006.

[7] A Hyvärinen. Estimation of non-normalized statistical models using score matching. *Journal of Machine Learning Research*, 6:695–709, 2005.

[8] A Hyvärinen. Connections between score matching, contrastive divergence, and pseudolikelihood for continuous-valued variables. *IEEE Transactions on Neural Networks*, Jan 2007.

[9] T Jaakkola and M Jordan. A variational approach to Bayesian logistic regression models and their extensions. *Proceedings of the Sixth International Workshop on Artificial Intelligence and Statistics*, Jan 1997.

[10] H Kappen and F Rodríguez. Mean field approach to learning in Boltzmann machines. *Pattern Recognition Letters*, Jan 1997.

[11] D MacKay. Failures of the one-step learning algorithm. *http://www.inference.phy.cam.ac.uk/mackay/gbm.pdf*, Jan 2001.

[12] J Macke and S Gerwinn. Personal communication. 2009.

[13] J R Movellan. Contrastive divergence in Gaussian diffusions. *Neural Computation*, 20(9):2238–2252, 2008.

[14] J R Movellan. A minimum velocity approach to learning. *unpublished draft*, Jan 2008.

[15] J R Movellan and J L McClelland. Learning continuous probability distributions with symmetric diffusion networks. *Cognitive Science*, 17:463–496, 1993.

[16] R Pathria. *Statistical Mechanics*. Butterworth Heinemann, Jan 1972.

[17] M Schmidt. minfunc. *http://www.cs.ubc.ca/ schmidtm/Software/minFunc.html*, 2005.

[18] E Schneidman, M J Berry 2nd, R Segev, and W Bialek. Weak pairwise correlations imply strongly correlated network states in a neural population. *Nature*, 440(7087):1007–12, 2006.

[19] J Shlens, G D Field, J L Gauthier, M Greschner, A Sher, A M Litke, and E J Chichilnisky. The structure of large-scale synchronized firing in primate retina. *Journal of Neuroscience*, 29(15):5022–5031, Apr 2009.

[20] J Shlens, G D Field, J L Gauthier, M I Grivich, D Petrusca, A Sher, A M Litke, and E J Chichilnisky. The structure of multi-neuron firing patterns in primate retina. *J. Neurosci.*, 26(32):8254–66, 2006.

[21] J Sohl-Dickstein and B Olshausen. A spatial derivation of score matching. *Redwood Center Technical Report*, 2009.

[22] T Tanaka. Mean-field theory of Boltzmann machine learning. *Physical Review Letters E*, Jan 1998.

[23] M Welling and G Hinton. A new learning algorithm for mean field Boltzmann machines. *Lecture Notes in Computer Science*, Jan 2002.

[24] A Yuille. The convergence of contrastive divergences. *Department of Statistics, UCLA. Department of Statistics Papers.*, 2005.