Noise-contrastive estimation: A new estimation principle for unnormalized statistical models

Michael Gutmann
Dept of Computer Science
and HIIT, University of Helsinki
michael.gutmann@helsinki.fi

Aapo Hyvärinen
Dept of Mathematics & Statistics, Dept of Computer
Science and HIIT, University of Helsinki
aapo.hyvarinen@helsinki.fi

Abstract

We present a new estimation principle for parameterized statistical models. The idea is to perform nonlinear logistic regression to discriminate between the observed data and some artificially generated noise, using the model log-density function in the regression nonlinearity. We show that this leads to a consistent (convergent) estimator of the parameters, and analyze the asymptotic variance. In particular, the method is shown to directly work for unnormalized models, i.e. models where the density function does not integrate to one. The normalization constant can be estimated just like any other parameter. For a tractable ICA model, we compare the method with other estimation methods that can be used to learn unnormalized models, including score matching, contrastive divergence, and maximum-likelihood where the normalization constant is estimated with importance sampling. Simulations show that noise-contrastive estimation offers the best trade-off between computational and statistical efficiency. The method is then applied to the modeling of natural images: We show that the method can successfully estimate a large-scale two-layer model and a Markov random field.

1 Introduction

Estimation of unnormalized parameterized statistical models is a computationally difficult problem. Here, we propose a new principle for estimating such models.

Appearing in Proceedings of the 13^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2010, Chia Laguna Resort, Sardinia, Italy. Volume 9 of JMLR: W&CP 9.

Our method provides, at the same time, an interesting theoretical connection between unsupervised learning and supervised learning.

The basic estimation problem . Assume a sample of a rv $\mathbf{x} \in \mathbb{R}^n$ is observed which follows an unknown pdf $p_d(.)$. The data pdf $p_d(.)$ is modeled by a parameterized family of functions $\{p_m(.;\alpha)\}_\alpha$. We assume that $p_d(.)$ belongs to this family . In other words , $p_d(.) = p_m(.;\alpha^\star)$ for some parameter $\alpha^\star$. The problem we consider here is how to estimate α from the observed sample by maximizing some objective function.
Any solution $\hat{\alpha}$ to this estimation problem must yield a properly normalized density $p_m(.;\hat{\alpha})$ with

$$\int p_m(\mathbf{u};\hat{\alpha})\mathrm{d}\mathbf{u} = 1. \quad (1)$$

This defines essentially a constraint in the optimization problem.[1] In principle, the constraint can always be fulfilled by redefining the pdf as

$$p_m(.;\alpha) = \frac{p_m^0(.;\alpha)}{Z(\alpha)}, \quad Z(\alpha) = \int p_m^0(\mathbf{u};\alpha)\mathrm{d}\mathbf{u}, \quad (2)$$

where $p_m^0(.;\alpha)$ specifies the functional form of the pdf and does not need to integrate to one. The calculation of the normalization constant (partition function) $Z(\alpha)$ is, however, very problematic: The integral is rarely analytically tractable, and if the data is high-dimensional, numerical integration is difficult. Examples of statistical models where the normalization constraint poses a problem can be found in Markov random fields (Roth & Black, 2009; Köster et al., 2009), energy-based models (Hinton, 2002; Teh et al., 2004), and multilayer networks (Osindero et al., 2006; Köster & Hyvärinen, 2007).

[1] Often, this constraint is imposed on $p_m(.;\alpha)$ for all α, but we will see in this paper that it is actually enough to impose it on the solution obtained.

A conceptually simple way to deal with the normalization constraint would be to consider the normalization constant $Z(\alpha)$ as an additional parameter of the model. This approach is, however, not possible for Maximum Likelihood Estimation (MLE). The reason is that the likelihood can be made arbitrarily large by making $Z(\alpha)$ go to zero. Therefore, methods have been proposed which estimate the model directly using $p_m^0(.;\alpha)$ without computation of the integral which defines the normalization constant; the most recent ones are contrastive divergence (Hinton, 2002) and score matching (Hyvärinen, 2005).

Here, we present a new estimation principle for unnormalized models which shows advantages over contrastive divergence or score matching. Both the parameter α in the unnormalized pdf $p_m^0(.;\alpha)$ and the normalization constant can be estimated by maximization of the same objective function. The basic idea is to estimate the parameters by learning to discriminate between the data $\mathbf{x}$ and some artificially generated noise $\mathbf{y}$. The estimation principle thus relies on noise with which the data is contrasted, so that we will refer to the new method as "noise-contrastive estimation".

In Section 2, we formally define noise-contrastive estimation, establish fundamental statistical properties, and make the connection to supervised learning explicit. In Section 3, we first illustrate the theory with the estimation of an ICA model, and compare the performance to other estimation methods. Then, we apply noise-contrastive estimation to the learning of a two-layer model and a Markov random field model of natural images. Section 4 concludes the paper.

2 NCE

2.1 Definition of the estimator

For a statistical model which is specified through an unnormalized pdf $p_m^0(.;\alpha)$, we include the normalization constant as another parameter c of the model. That is, we define $\ln p_m(.;\theta) = \ln p_m^0(.;\alpha) + c$, where $\theta = \{\alpha, c\}$. Parameter c is an estimate of the negative logarithm of the normalization constant $Z(\alpha)$. Note that $p_m(.;\theta)$ will only integrate to one for some specific choice of the parameter c.

Denote by $X = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$ the observed data set, consisting of T observations of the data $\mathbf{x}$, and by $Y = (\mathbf{y}_1, \ldots, \mathbf{y}_T)$ an artificially generated data set of noise $\mathbf{y}$ with distribution $p_n(.)$. The estimator $\hat{\theta}_T$ is defined to be the θ which maximizes the objective function

$$J_T(\theta) = \frac{1}{2T} \sum_t \ln\left[h(\mathbf{x}_t;\theta)\right] + \ln\left[1 - h(\mathbf{y}_t;\theta)\right], \quad (3)$$

where

$$h(\mathbf{u};\theta) = \frac{1}{1+\exp\left[-G(\mathbf{u};\theta)\right]}, \quad (4)$$

$$G(\mathbf{u};\theta) = \ln p_m(\mathbf{u};\theta) - \ln p_n(\mathbf{u}). \quad (5)$$

Below, we will denote the logistic function by $r(.)$ so that $h(\mathbf{u};\theta) = r(G(\mathbf{u};\theta))$.

2.2 Connection to supervised learning

The objective function in Eq. (3) occurs also in supervised learning. It is the log-likelihood in a logistic regression model which discriminates the observed data X from the noise Y. This connection to supervised learning, namely logistic regression and classification, provides us with intuition of how the proposed estimator works: By discriminating, or comparing, between data and noise, we are able to learn properties of the data in the form of a statistical model. In less mathematical terms, the idea behind noise-contrastive estimation is "learning by comparison".

To make the connection explicit, we show now how the objective function in Eq. (3) is obtained in the setting of supervised learning. Denote by $U = (\mathbf{u}_1, \ldots, \mathbf{u}_{2T})$ the union of the two sets X and Y, and assign to each data point $\mathbf{u}_t$ a binary class label C_t: $C_t = 1$ if $\mathbf{u}_t \in X$ and $C_t = 0$ if $\mathbf{u}_t \in Y$. In logistic regression, the posterior probabilities of the classes given the data $\mathbf{u}_t$ are estimated. As the pdf $p_d(.)$ of the data $\mathbf{x}$ is unknown, the class-conditional probability $p(.|C=1)$ is modeled with $p_m(.;\theta)$.[2] The class-conditional probability densities are thus

$$p(\mathbf{u}|C=1;\theta) = p_m(\mathbf{u};\theta) \quad p(\mathbf{u}|C=0) = p_n(\mathbf{u}). \quad (6)$$

Since we have equal probabilities for the two class labels, i.e. $P(C=1) = P(C=0) = 1/2$, we obtain the following posterior probabilities

$$P(C=1|\mathbf{u};\theta) = \frac{p_m(\mathbf{u};\theta)}{p_m(\mathbf{u};\theta) + p_n(\mathbf{u})} = h(\mathbf{u};\theta) \quad (8)$$

$$P(C=0|\mathbf{u};\theta) = 1 - h(\mathbf{u};\theta). \quad (9)$$

The class labels C_t are Bernoulli-distributed so that the log-likelihood of the parameters θ becomes

$$\ell(\theta) = \sum_t C_t \ln P(C_t=1|\mathbf{u}_t;\theta) + (1-C_t)\ln P(C_t=0|\mathbf{u}_t;\theta) \quad (10)$$

$$= \sum_t \ln\left[h(\mathbf{x}_t;\theta)\right] + \ln\left[1 - h(\mathbf{y}_t;\theta)\right], \quad (11)$$

which is, up to the factor $1/2T$, the same as our objective function in Eq. (3).

[2] Classically, $p_m(.;\theta)$ would in the context of this section be a normalized pdf. In our paper, however, the normalization constant may also be part of the parameters.

2.3 Properties of the estimator

We characterize here the behavior of the estimator $\hat{\theta}_T$ when the sample size T becomes arbitrarily large. The weak law of large numbers shows that in that case, the objective function $J_T(\theta)$ converges in probability to J,

$$J(\theta) = \frac{1}{2}\,\mathrm{E}\,\ln\left[h(\mathbf{x};\theta)\right] + \ln\left[1 - h(\mathbf{y};\theta)\right]. \qquad (12)$$

Let us denote by $\tilde{J}$ the objective J seen as a function of $f(.) = \ln p_m(.;\theta)$, i.e.

$$\begin{aligned}\tilde{J}(f) \quad = \quad & \frac{1}{2}\,\mathrm{E}\,\ln\left[r(f(\mathbf{x}) - \ln p_n(\mathbf{x}))\right] + \\ & \ln\left[1 - r(f(\mathbf{y}) - \ln p_n(\mathbf{y}))\right]. \qquad (13)\end{aligned}$$

We start the characterization of the estimator $\hat{\theta}_T$ with a description of the optimization landscape of $\tilde{J}$. The following theorem[3] shows that the data pdf $p_d(.)$ can be found by maximization of $\tilde{J}$, i.e. by learning a classifier under the ideal situation of infinite amount of data.

Theorem 1 (Nonparametric estimation)**.** *$\tilde{J}$ attains a maximum at $f(.) = \ln p_d(.)$. There are no other extrema if the noise density $p_n(.)$ is chosen such it is nonzero whenever $p_d(.)$ is nonzero.*

A fundamental point in the theorem is that the maximization is performed without any normalization constraint for $f(.)$. This is in stark contrast to MLE, where $\exp(f)$ must integrate to one. With our objective function, no such constraints are necessary. The maximizing pdf is found to have unit integral automatically. The positivity condition for $p_n(.)$ in the theorem tells us that the data pdf $p_d(.)$ cannot be inferred where there are no contrastive noise samples for some relevant regions in the data space. This situation can be easily avoided by taking, for example, a Gaussian distribution for the contrastive noise.

In practice, the amount of data is limited and a finite number of parameters $\theta \in \mathbb{R}^m$ specify $p_m(.;\theta)$. This has in general two consequences: First, it restricts the space where the data pdf $p_d(.)$ is searched for. Second, it may introduce local maxima into the optimization landscape. For the characterization of the estimator in this situation, it is normally assumed that $p_d(.)$ follows the model, i.e. there is a $\theta^\star$ such that $p_d(.) = p_m(.;\theta^\star)$.

Our second theorem tells us that $\hat{\theta}_T$, the value of θ which (globally) maximizes J_T, converges to $\theta^\star$ and leads thus to the correct estimate of $p_d(.)$ as the sample size T increases. For unnormalized models, the log-normalization constant is part of the parameters. This means that the maximization of our objective function leads to the correct estimates for both the parameters α in the unnormalized pdf $p^0_m(.;\alpha)$ and the log-normalization constant c, which is impossible when using likelihood.

Theorem 2 (Consistency)**.** *If conditions (a) to (c) are fulfilled then $\hat{\theta}_T$ converges in probability to $\theta^\star$, i.e. $\hat{\theta}_T \xrightarrow{P} \theta^\star$.*

(a) $p_n(.)$ is nonzero whenever $p_d(.)$ is nonzero

(b) $\sup_\theta |J_T(\theta) - J(\theta)| \xrightarrow{P} 0$

(c) $\mathcal{I} = \int \mathbf{g}(\mathbf{x})\mathbf{g}(\mathbf{x})^T P(\mathbf{x}) p_d(\mathbf{x})\mathrm{d}\mathbf{x}$ has full rank, where

$$P(\mathbf{x}) = \frac{p_n(\mathbf{x})}{p_d(\mathbf{x}) + p_n(\mathbf{x})}, \quad \mathbf{g}(\mathbf{x}) = \nabla_\theta \ln p_m(\mathbf{x};\theta)|_{\theta^\star}$$

Condition (a) is inherited from Theorem 1, and is easily fulfilled by choosing, for example, the noise to be Gaussian. Conditions (b) and (c) have their counterparts in MLE, see e.g.(Wasserman, 2004). We need in (b) uniform convergence in probability of J_T to J; in MLE, uniform convergence of the log-likelihood to the Kullback-Leibler distance is required likewise. Condition (c) assures that for large sample sizes, the objective function J_T becomes peaky enough around the true value $\theta^\star$. This imposes through the vector $\mathbf{g}$ a condition on the model $p_m(.;\theta)$. A similar constraint is required in MLE. For the estimation of normalized models $p_m(.;\alpha)$, where the normalization constant is not part of the parameters, the vector $\mathbf{g}(\mathbf{x})$ is the score function as in MLE. Furthermore, if $P(\mathbf{x})$ were a constant, $\mathcal{I}$ would be proportional to the Fisher information matrix.

The following theorem describes the distribution of the estimation error $(\hat{\theta}_T - \theta^\star)$ for large sample sizes.

Theorem 3 (Asymptotic normality)**.** *$\sqrt{T}(\hat{\theta}_T - \theta^\star)$ is asymptotically normal with mean zero and covariance matrix Σ,*

$$\begin{aligned}\Sigma \quad = \quad & \mathcal{I}^{-1} - 2\mathcal{I}^{-1}\left[\int \mathbf{g}(\mathbf{x})P(\mathbf{x})p_d(\mathbf{x})\mathrm{d}\mathbf{x}\right] \times \qquad (14) \\ & \left[\int \mathbf{g}(\mathbf{x})^T P(\mathbf{x})p_d(\mathbf{x})\mathrm{d}\mathbf{x}\right]\mathcal{I}^{-1}.\end{aligned}$$

When we are estimating a normalized model $p_m(.;\alpha)$, we observe here again some similarities to MLE by considering the hypothetical case that $P(\mathbf{x})$ is a constant: The integral in the brackets is then zero because it is proportional to the expectation of the score function, which is zero. The covariance matrix Σ is thus up to a scaling constant equal to the Fisher information matrix.

Theorem 3 leads to the following corollary:

Corollary 1. *For large sample sizes T, the mean squared error $\mathrm{E}\,||\hat{\theta}_T - \theta^\star||^2$ behaves like $\mathrm{tr}(\Sigma)/T$.*

[3] Proofs are omitted due to a lack of space.

2.4 Choice of the contrastive noise distribution

The noise distribution $p_n(.)$, which is used for contrast, is a design parameter. In practice, we would like to have a noise distribution which fulfills the following:

(1) It is easy to sample from, since the method relies on a set of samples Y from the noise distribution.
(2) It allows for an analytical expression for the log-pdf, so that we can evaluate the objective function in Eq. (3) without any problems.
(3) It leads to a small mean squared error $\mathrm{E}\,||\hat{\theta}_T - \theta^\star||^2$.

Our result on consistency (Theorem 2) also includes some technical constraints on $p_n(.)$ but they are so mild that, given an estimation problem at hand, many distributions will verify them. In principle, one could minimize the MSE in Corollary 1 wrt the noise distribution $p_n(.)$. However, this turns out to be quite difficult, and sampling from such a distribution might not be straightforward either. In practice, a well-known noise distribution which satisfies points (1) and (2) above seems to be a good choice. Some examples are a Gaussian or uniform distribution, a Gaussian mixture distribution, or an ICA distribution.

Intuitively, the noise distribution should be close to the data distribution, because otherwise, the classification problem might be too easy and would not require the system to learn much about the structure of the data. This intuition is partly justified by the following theoretical result: If the noise is equal to the data distribution, then Σ in Theorem 3 equals two times the Cramér-Rao bound. Thus, for a noise distribution that is close to the data distribution, we have some guarantee that the MSE is reasonably close to the theoretical optimum.[4] As a consequence, one could choose a noise distribution by first estimating a preliminary model of the data, and then use this preliminary model as the noise distribution.

3 Simulations

3.1 Simulations with artificial data

We illustrate NCE with the estimation of an ICA model (Hyvarinen et al., 2001), and compare its performance with other estimation methods, namely MLE, MLE where the normalization (partition function) is calculated with importance sampling (see e.g. (Wasserman, 2004) for an introduction to IS), CD (Hinton, 2002), and score matching (Hyvarinen, 2005). MLE gives the performance baseline. It can, however, only be used if an analytical expression for the partition function is available. The other methods can all be used to learn unnormalized models.

3.1.1 Data and unnormalized model

Data $\mathbf{x} \in \mathbb{R}^4$ is generated via the ICA model

$$\mathbf{x} = A\mathbf{s}, \tag{15}$$

where $A = (\mathbf{a}_1, \ldots, \mathbf{a}_4)$ is a 4×4 mixing matrix. All four independent sources in $\mathbf{s}$ follow a Laplacian density of unit variance and zero mean. The data log-pdf $\ln p_d(.)$ is thus

$$\ln p_d(\mathbf{x}) = -\sum_{i=1}^{4} \sqrt{2}\,|\mathbf{b}_i^\star \mathbf{x}| + (\ln|\det B^\star| - \ln 4), \tag{16}$$

where $\mathbf{b}_i^\star$ is the i-th row of the matrix $B^\star = A^{-1}$. The unnormalized model is

$$\ln p_m^0(\mathbf{x};\alpha) = -\sum_{i=1}^{4} \sqrt{2}|\mathbf{b}_i \mathbf{x}|. \tag{17}$$

The parameters $\alpha \in \mathbb{R}^{16}$ are the row vectors $\mathbf{b}_i$. For noise-contrastive estimation, we consider also the normalization constant to be a parameter and work with

$$\ln p_m(\mathbf{x};\theta) = \ln p_m^0(\mathbf{x};\alpha) + c. \tag{18}$$

The scalar c is an estimate for the negative log-partition function. The total set of the parameters for noise-contrastive estimation is thus $\theta = \{\alpha, c\}$ while for the other methods, the parameters are given by α. The true values of the parameters are the vectors $\mathbf{b}_i^\star$ for α and $c^\star = \ln|\det B^\star| - \ln 4$ for c.

3.1.2 Estimation methods

For noise-contrastive estimation, we choose the contrastive noise $\mathbf{y}$ to be Gaussian with the same mean and covariance matrix as $\mathbf{x}$. The parameters θ are then estimated by learning to discriminate between the data $\mathbf{x}$ and the noise $\mathbf{y}$, i.e. by maximizing J_T in Eq. (3). The optimization is done with a conjugate gradient algorithm (Rasmussen, 2006).

We give now a short overview of the estimation methods that we used for comparison and comment on our implementation:
In MLE, the parameters α are chosen such that the probability for the observed data is maximized, i.e.

$$J_{\mathrm{MLE}}(\alpha) = \frac{1}{T}\sum_t \ln p_m^0(\mathbf{x}(t);\alpha) - \ln Z(\alpha) \tag{19}$$

[4] At a first glance, this might be counterintuitive. In the setting of logistic regression, however, we will then have to learn that the two distributions are equal and that the posterior probability for any point belonging to any of the two classes is 50%, which is a well defined problem.

is maximized. Calculation of the gradient gives

$$\nabla_\alpha J_{\mathrm{MLE}} = \frac{1}{T}\sum_t \nabla_\alpha \ln p^0_m(\mathbf{x}(t);\alpha) - \frac{\nabla_\alpha Z(\alpha)}{Z(\alpha)}, \quad (20)$$

so that a steepest ascent algorithm can be used for the optimization. In our implementation for the estimation of the ICA model, we used the faster natural gradient, see e.g. (Hyvarinen et al., 2001).

IS:

$$Z(\alpha) \approx \frac{1}{T}\sum_t \frac{p^0_m(\mathbf{n}_t;\alpha)}{p_{\mathrm{IS}}(\mathbf{n}_t)}. \quad (21)$$

The derivative $\nabla_\alpha Z(\alpha)$ is calculated in the same way. The samples $\mathbf{n}_t$ are i.i.d. and follow the distribution $p_{\mathrm{IS}}(.)$. The ratio $\nabla_\alpha Z(\alpha)/Z(\alpha)$, and thus the gradient of J_{MLE} in Eq. (20) becomes available for the optimization of J_{MLE}. In our implementation, we made for $p_{\mathrm{IS}}(.)$ and the number of samples T the same choice as for NCE.

CD: the gradient of the log partition function in Eq. (20) can be rewritten as

$$\frac{\nabla_\alpha Z(\alpha)}{Z(\alpha)} = \int p_m(\mathbf{n};\alpha)\nabla_\alpha p^0_m(\mathbf{n};\alpha)\mathrm{d}\mathbf{n}. \quad (22)$$

If we had data $\mathbf{n}_t$ at hand which follows the model density $p_m(.;\alpha)$, we could evaluate the last equation by taking the sample average. In contrastive divergence, data $\mathbf{n}_t$ is created which follows approximately $p_m(.;\alpha)$ by means of MCMC. In our implementation, we used one step of HMC (MacKay, 2002) with three leapfrog steps.

SM: the cost function

$$J_{\mathrm{sm}}(\alpha) = \frac{1}{T}\sum_{t,n}\frac{1}{2}\Psi^2_n(\mathbf{x}(t);\alpha) + \Psi'_n(\mathbf{x}(t);\alpha) \quad (23)$$

must be minimized where $\Psi_n(\mathbf{x};\alpha)$ is the derivative of the unnormalized model with respect to the n-th element of $\mathbf{x}$, i.e. $\Psi_n(\mathbf{x};\alpha) = \partial_{\mathbf{x}(n)} \ln p^0_m(\mathbf{x};\alpha)$. For the ICA model with Laplacian sources, we obtain

$$\Psi_n(\mathbf{x};\alpha) = \sum_i g(\mathbf{b}_i\mathbf{x})B_{in}, \quad (24)$$

$$\Psi'_n(\mathbf{x};\alpha) = \sum_i g'(\mathbf{b}_i\mathbf{x})B^2_{in}, \quad (25)$$

where $g(u) = -\sqrt{2}\mathrm{sign}(u)$. We can see here that the sign-function is not smooth enough to be used in score matching. We use therefore the approximation $\mathrm{sign}(u) \approx \tanh(10u)$. This corresponds to assuming a logistic density for the sources. The optimization is then done by conjugate gradient (Rasmussen, 2006).

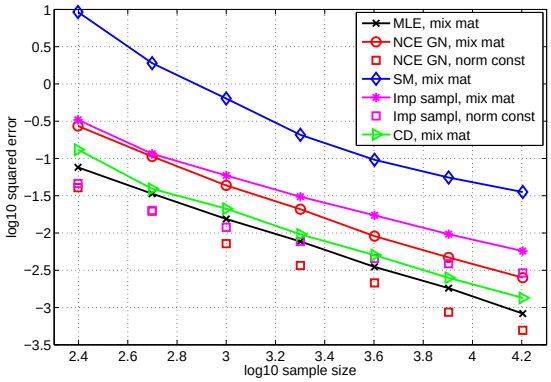
3.1.3 Results

Figure 1 (a) shows the MSE $\mathrm{E}\,||\hat{\theta}_T - \theta^\star||^2$ for the different estimation methods. MLE gives the reference performance (black crosses). In red, the performance of NCE is shown . We can see that the error for the demixing matrix B decreases with increasing sample size T (red circles). The same holds for the error in the log-normalization constant c (red squares). This illustrates the consistency of the estimator as convergence in quadratic mean implies convergence in probability . Figure (a) shows also that NCE performs better than MLE where the normalization constant is calculated with IS (magenta asterisks). In particular , the estimate of the log -normalization constant c is more accurate . CD (green triangles) yields, for fixed sample sizes, the best results after MLE. It should be pointed out, however, that the distribution of the squared error has a much higher variance for contrastive divergence than for the other methods (around 50 times higher than NCE). The figure shows also that SM (blue diamonds) is outperformed by the other methods . The reason is that we had to resort to an approximation of the Laplacian density for the estimation with SM.

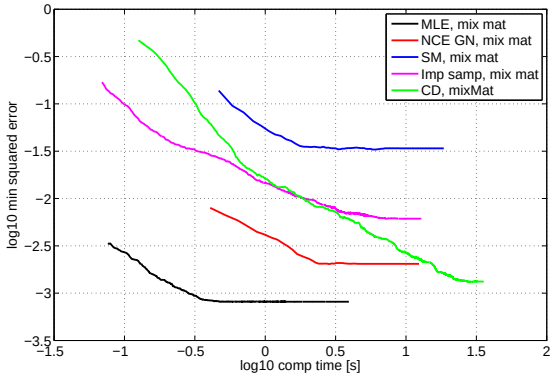
Figure 1 (b) investigates the trade-off between statistical and computational efficiency. MLE needs the shortest computation time for a given precision in the estimate. This reflects the fact that MLE, unlike the other methods, works with properly normalized densities. Among the methods for unnormalized models, noise-contrastive estimation requires the least computation time to reach a required level of precision. Compared to contrastive divergence, it is at least three times faster.

Figure 1 (c) illustrates the idea of using a noise distribution which is as close to the data distribution as possible. In a first step, the inverse B of the mixing matrix A was estimated by contrasting the data with Gaussian noise as in Figure 1 (a). In a second step, contrastive noise was generated by mixing Laplacian sources of unit variance with the matrix $\hat{A} = \hat{B}^{-1}$. The performance for this kind of noise is shown in blue. We see that fine-tuning the parameters with the Laplacian contrastive noise results in estimates of smaller MSE. Testing for the significance of the difference of the MSE before and after fine-tuning gives p-values $< 10^{-12}$.

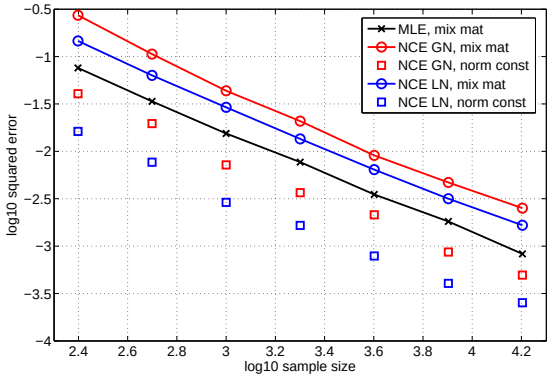
Figure 1 (d) shows that, for large sample sizes T, Corollary 1 predicts correctly the MSE of the parameters: The MSE decays as $\mathrm{tr}(\Sigma)/T$, where Σ was defined in Theorem 3. Note that here, the set of parameters includes the parameter for the normalization constant.



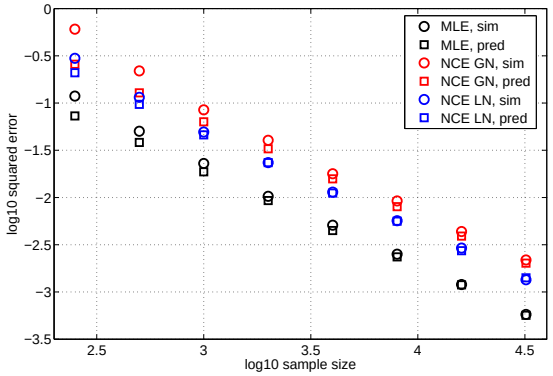
(a) Estimation accuracy



(b) Estimation accuracy versus computation time



(c) Fine tuning



(d) Asymptotic behavior

Figure 1: NCE of an ICA model and comparison to other estimation methods. Figure (a) shows the MSE for the estimation methods in function of the sample size. Figure (b) shows the estimation error in function of the computation time. Among the methods for unnormalized models, NCE requires the least computation time to reach a required level of precision. Figure (c) shows that NCE with Laplacian contrastive noise leads to a better estimate than NCE with Gaussian noise. Figure (d) shows that Corollary 1 describes the behavior of the MSE for large sample sizes correctly. *Simulation and plotting details:* Figures (a)-(c) show the median of the simulation results. In (d), we took the average. For each sample size T, averaging is based on 500 random mixing matrices A with condition numbers less than 10. In figures (a),(c) and (d), we started for each mixing matrix the optimization at 5 different initializations in order to avoid local optima. This was not possible for CD as it does not have a proper objective function. This might be the reason for the higher error variance of CD that we have pointed out in the main text. For NCE and SM, we relied on the built-in criteria of (Rasmussen, 2006) to determine convergence. For the other methods, we did not use a particular convergence criterion: They were all given a sufficiently long running time to assure that the algorithms had converged. In figure (b), we performed only one optimization run per mixing matrix to make the comparison fair. Note that, while the other figures show the MSE at time of convergence of the algorithms, this figure shows the behavior of the error during the runs. In more detail, the curves in the figure show the, on average, minimal possible estimation error at a given time. For any given method at hand, the curve was created as follows: We monitored for each mixing matrix the estimation error and the elapsed time during the runs. This was done for all the sample sizes that we used in figure (a). For any fixed time t_0, we obtained in that way a set of estimation errors, one for each sample size.
We retained then the smallest error. This gives the minimal possible estimation error that can be achieved by time t_0. Taking the median over all 500 mixing matrices yielded, for a given method, the curve shown in the figure. Note that, by construction, the curves in the figure do not depend on the stopping criterion. Comparing figure (b) with figure (a) shows furthermore that the curves in (b) flatten out at error levels that correspond to the MSE in figure (a) for the sample size $T = 16000$, i.e. $\log 10(T) \approx 4.2$. This was the largest sample size used in the simulations. For larger sample sizes, the curves would flatten out at lower error levels.

3.2 Simulations with natural images

We use here noise-contrastive estimation to learn the statistical structure of natural images. Current models of natural images can be broadly divided into patch-based models and Markov Random Field (MRF) based models. Patch-based models are mostly two-layer models (Osindero et al., 2006; Köster & Hyvärinen, 2007; Karklin & Lewicki, 2005), although in (Osindero & Hinton, 2008) a three-layer model is presented. Most of these models are unnormalized. Score matching and contrastive divergence have typically been used to estimate them. For the learning of MRF from natural images, contrastive divergence has been used in (Roth & Black, 2009), while (Köster et al., 2009) employs score matching.

3.2.1 Patch-model

Natural image data was obtained by sampling patches of size 30×30 pixel from images of van Hateren's database which depict wild-life scenes only. As preprocessing, we removed the DC component of each patch, whitened the data, and reduced the dimensions from 900 to 225. The dimension reduction implied that we retained 92 % of the variance of the data. As a novel preprocessing step, we then further normalized each image patch so that it had zero DC value and unit variance. The whitened data was thus projected onto a sphere. Projection onto a sphere can be considered as a form of divisive normalization (Lyu & Simoncelli, 2009). For the contrastive noise, we used a uniform distribution on the sphere.

Our model for a patch $\mathbf{x}$ is

$$\log p_m(\mathbf{x};\theta) = \sum_n f_{\text{th}}\left(\ln\left[\mathbf{v}_n\left(W\mathbf{x}\right)^2 + 1\right] + b_n\right) + c,$$

where the squaring operation is applied to every element of the vector $W\mathbf{x}$, and $f_{\text{th}}(.)$ is a smooth thresholding function.[5] The parameters θ of the model are the matrix $W \in \mathbb{R}^{225\times225}$, the 225 row vectors $\mathbf{v}_n \in \mathbb{R}^{225}$ and the equal number of bias terms b_n which define the thresholds, as well as c for the normalization of the pdf. The only constraint we are imposing is that the vectors $\mathbf{v}_n$ are limited to be non-negative.

We learned the model in three steps: First, we learned all the parameters but keeping the second layer (the matrix V with row vectors $\mathbf{v}_n$), fixed to identity. The second step was learning of V with the other parameters held fixed. Initializing V randomly to small values proved helpful. When the objective function reached again the level it had at the end of the first step, we switched, as the third step, to concurrent learning of all parameters. For the optimization, we used a conjugate gradient algorithm (Rasmussen, 2006).

[5] $f_{\text{th}}(u) = 0.25\ln(\cosh(2u)) + 0.5u + 0.17$

Figure 2 (a) shows the estimation results. The first layer features $\mathbf{w}_i$ (rows of W) are Gabor-like ("simple cells"). The second layer weights $\mathbf{v}_i$ pool together features of similar orientation and frequency, which are not necessarily centered at the same location ("complex cells"). The results correspond to those reported in (Köster & Hyvärinen, 2007) and (Osindero et al., 2006).

3.2.2 MRF

We used basically the same data, preprocessing and contrastive noise as for the patch based model. In order to train a MRF with clique size 15 pixels, we used, however, image patches of size 45×45 pixel.[6] Furthermore, for whitening, we employed a whitening filter of size 9×9 pixel. No redundancy reduction was performed.

Denote by $I(\xi)$ the pixel value of an image $I(.)$ at position ξ. Our model for an image $I(.)$ is

$$\log p_m(I;\theta) = \sum_{\xi,i} f_{\text{th}}\left(\sum_{\xi'} w_i(\xi')I_w(\xi+\xi') + b_i\right) + c,$$

where $I_w(.)$ is the image $I(.)$ filtered with the whitening filter. The parameters θ of the model are the filters $w_i(.)$ (size 7×7 pixel), the thresholds b_i for $i = 1\ldots25$, and c for the normalization of the pdf.

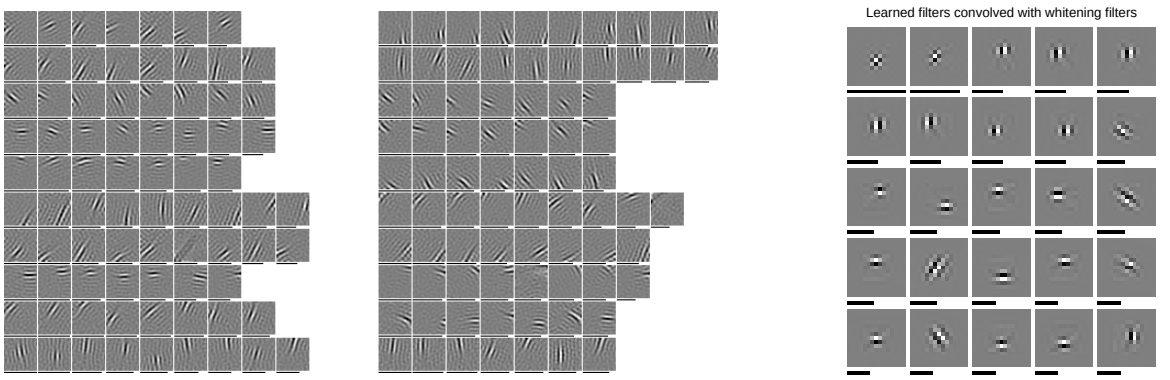
Figure 2 shows the learned filters $w_i(.)$ after convolution with the whitening filter. The filters are rather high-frequency and Gabor-like. This is different compared to (Roth & Black, 2009), where the filters had no clear structure. In (Köster et al., 2009), the filters, which were shown in the whitened space, were also Gabor-like. However, unlike in our model, a norm constraint on the filters was there necessary to get several non-vanishing filters.

4 Conclusion

We proposed here a new estimation principle, noise-contrastive estimation, which consistently estimates sophisticated statistical models that do not need to be normalized (e.g. energy-based models or Markov random fields). In fact, the normalization constant can be estimated as any other parameter of the model. One benefit of having an estimate for the normalization constant at hand is that it could be used to compare the likelihood of several distinct models. Furthermore, the principle shows a new connection between unsupervised and supervised learning.

For a tractable ICA model, we compared noise-contrastive estimation with other methods that can

[6] Although the MRF is a model for an entire image, training can be done with image patches, see (Köster et al., 2009).



(a) Patch-model (patch size: 30 pixels) (b) MRF (clique size: 15 pixels)

Figure 2: Noise-contrastive estimation of models for natural images. (a) Random selection of 2×10 out of 255 pooling patterns. Every vector $\mathbf{v}_i$ corresponds to a pooling pattern. The patches in pooling pattern i_0 show the $\mathbf{w}_i$ and the black bar under each patch indicates the strength $\mathbf{v}_{i_0}(n)$ by which a certain $\mathbf{w}_n$ is pooled by $\mathbf{v}_{i_0}$. (b) Learned filters $w_i(.)$ in the original space, i.e. after convolution with the whitening filter. The black bar under each patch indicates the norm of the filter.

be used to estimate unnormalized models . NCE is found to compare favorably . It offers the best trade-off between computational and statistical efficiency . We then applied nce to the learning of an energy - based two - layer model and a Markov random field model of natural images . The results confirmed the validity of the estimation principle: For the two-layer model, we obtained simple and complex cell properties in the first two layers . For the Markov random field , highly structured Gabor - like filters were obtained . Moreover, the two - layer model could be readily extended to have more layers . An important potential application of our estimation principle lies thus in deep learning.

We used in previous work classification based on logistic regression to learn features from images (Gutmann & Hyvärinen, 2009). However, only one layer of Gabor features was learned in that paper, and, importantly, such learning was heuristic and not connected to estimation theory. Here, we showed an explicit connection to statistical estimation and provided a formal analysis of the learning in terms of estimation theory. This connection leads to further extensions of the principle which will be treated in future work.

References

Gutmann, M., & Hyvärinen, A. (2009). Learning features by contrasting natural images with noise. *Proc. Int. Conf. on Artificial Neural Networks (ICANN2009).*

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

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

Hyvärinen, A., Karhunen, J., & Oja, E. (2001). *Independent component analysis.* Wiley-Interscience.

Karklin, Y., & Lewicki, M. (2005). A hierarchical bayesian model for learning nonlinear statistical regularities in nonstationary natural signals. *Neural Computation*, *17*, 397–423.

Köster, U., & Hyvärinen, A. (2007). A two-layer ICA-like model estimated by score matching. *Proc. Int. Conf. on Artificial Neural Networks (ICANN2007).*

Köster, U., Lindgren, J., & Hyvärinen, A. (2009). Estimating markov random field potentials for natural images. *Int. Conf. on Independent Component Analysis and Blind Source Separation (ICA2009).*

Lyu, S., & Simoncelli, E. (2009). Nonlinear extraction of independent components of natural images using radial gaussianization. *Neural Computation*, *21*, 1485–1519.

MacKay, D. (2002). *Information theory, inference & learning algorithms.* Cambridge University Press.

Osindero, S., & Hinton, G. (2008). Modeling image patches with a directed hierarchy of markov random fields. In *Advances in neural information processing systems 20*, 1121–1128. MIT Press.

Osindero, S., Welling, M., & Hinton, G. E. (2006). Topographic product models applied to natural scene statistics. *Neural Computation*, *18 (2).*

Rasmussen, C. (2006). Conjugate gradient algorithm, version 2006-09-08. available online.

Roth, S., & Black, M. (2009). Fields of experts. *International Journal of Computer Vision*, *82*, 205–229.

Teh, Y., Welling, M., Osindero, S., & Hinton, G. (2004). Energy-based models for sparse overcomplete representations. *Journal of Machine Learning Research*, *4*, 1235–1260.

Wasserman, L. (2004). *All of statistics.* Springer.