ESTIMATION OF UNNORMALIZED STATISTICAL MODELS WITHOUT NUMERICAL INTEGRATION

Michael U. Gutmann[1,3] *and Aapo Hyvärinen*[1,2,3]

michael.gutmann@helsinki.fi aapo.hyvarinen@helsinki.fi
[1] Department of Mathematics and Statistics, University of Helsinki, Finland
[2] Department of Computer Science, University of Helsinki, Finland
[3] Helsinki Institute for Information Technology, Finland

ABSTRACT

Parametric statistical models of continuous or discrete valued data are often not properly normalized, that is, they do not integrate or sum to unity. The normalization is essential for maximum likelihood estimation. While in principle, models can always be normalized by dividing them by their integral or sum (their partition function), this can in practice be extremely difficult. We have been developing methods for the estimation of unnormalized models which do not approximate the partition function using numerical integration. We review these methods, score matching and noise-contrastive estimation, point out extensions and connections both between them and methods by other authors, and discuss their pros and cons.

1. INTRODUCTION

We consider the problem of estimating a parametric statistical model from n_x independent observations $\boldsymbol{x}_i$, $i = 1, \ldots, n_x$, of a m-dimensional random variable $\boldsymbol{x}$ with probability distribution f_x. The variable can be continuous, so that f_x is a probability density function (pdf), or discrete, so that f_x is a probability mass function (pmf).

The statistical model may be unnormalized, that is, the largest measure it assigns to an event is not one. This makes parameter estimation difficult, as will be explained later in detail. The purpose of this paper to review two estimation methods that are applicable to unnormalized models: Score matching and noise-contrastive estimation.

We start with classifying statistical models into normalized and unnormalized models (Section 2), and then explain why unnormalized models are important but difficult to estimate (Sections 3 and 4). This is followed by a brief overview of different approaches to the estimation of unnormalized models (Section 5). Score matching is the topic of Section 6, and Section 7 is on noise-contrastive estimation. Section 8 concludes the paper.

2. NORMALIZED VS UNNORMALIZED MODELS

In this paper, a statistical model is a family of nonnegative functions that are indexed by a vector of parameters $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^d$. A statistical model is normalized if each member of the family integrates (sums) to one. The largest measure it assigns to an event is thus one. For example, the univariate Gaussian

$$f(u;\theta) = \frac{\exp\left(-\theta\frac{u^2}{2}\right)}{\sqrt{\frac{2\pi}{\theta}}}, \quad \theta > 0, \tag{1}$$

defines a normalized model with the precision as parameter. We use $f(\boldsymbol{u};\boldsymbol{\theta})$ to denote normalized models. If the integration (normalization) condition is not satisfied, we call the model unnormalized. To denote unnormalized models, we use $p(\boldsymbol{u};\boldsymbol{\theta})$. For example, the models

$$p(u;\theta) = \exp\left(-\theta\frac{u^2}{2}\right), \quad \theta > 0, \tag{2}$$

and

$$p(u;\boldsymbol{\theta}) = \exp\left(-\theta_1\frac{u^2}{2} + \theta_2\right), \quad \theta_1 > 0,\ \theta_2 \in \mathbb{R}, \tag{3}$$

with $\boldsymbol{\theta} = (\theta_1, \theta_2)$, are unnormalized. In the latter model, θ_1 affects the shape of $p(u;\boldsymbol{\theta})$ while θ_2 affects its scale. This model only integrates to one if θ_1 and θ_2 satisfy $\theta_2 = 1/2\log(\theta_1/(2\pi))$.

In some literature, unnormalized models are called energy based models [1, 2] since a nonnegative function can be specified through the energy function $E(\boldsymbol{u};\boldsymbol{\theta})$,

$$p(\boldsymbol{u};\boldsymbol{\theta}) = \exp\left(-E(\boldsymbol{u};\boldsymbol{\theta})\right). \tag{4}$$

Regions of low energy have a large probability.

An unnormalized model does not automatically specify a pdf (or pmf) since it does not integrate (or sum) to one for all parameters. If $p(\boldsymbol{u};\boldsymbol{\theta})$ is integrable for all $\boldsymbol{\theta}$, an unnormalized model can be converted into a normalized one by dividing $p(\boldsymbol{u};\boldsymbol{\theta})$ by the partition function $Z(\boldsymbol{\theta})$,

$$Z(\boldsymbol{\theta}) = \int p(\boldsymbol{u};\boldsymbol{\theta})\mathrm{d}\boldsymbol{u}. \tag{5}$$

For the model in (2), for example, $Z(\theta) = \sqrt{2\pi/\theta}$. By the definition of $Z(\boldsymbol{\theta})$,

$$f(\boldsymbol{u};\boldsymbol{\theta}) = \frac{p(\boldsymbol{u};\boldsymbol{\theta})}{Z(\boldsymbol{\theta})} \tag{6}$$

satisfies the normalization condition.

Conversely, any normalized model $f(\boldsymbol{u};\boldsymbol{\theta})$ can be split into unnormalized model $p(\boldsymbol{u};\boldsymbol{\theta})$ and partition function $Z(\boldsymbol{\theta})$. With (6), the inverse partition function is given by the multiplicative factor of $f(\boldsymbol{u};\boldsymbol{\theta})$ that does not depend on $\boldsymbol{u}$.

We show in Section 4 that the partition function is essential for maximum likelihood estimation. The partition function $Z(\boldsymbol{\theta})$ is defined via a parameter-dependent integral. Often, this integral cannot be computed in closed form. Estimation methods for unnormalized models differ in how they handle the analytical intractability of the integral.

One class of estimation methods relies on the possibility to approximate the partition function pointwise by numerically integrating $p(\boldsymbol{u};\boldsymbol{\theta})$ for any fixed value of $\boldsymbol{\theta}$. However, such methods are computationally rather expensive and also tricky to use (see Section 5). The estimation methods which we review in this paper, score matching and noise-contrastive estimation, belong to another class of methods which does not rely on numerical integration to approximate the partition function (see Sections 6 and 7).

3. OCCURRENCE OF UNNORMALIZED MODELS

Unnormalized models are useful and practical tools to describe a data distribution. The reason is that, often, it is easier and more meaningful to model the shape of the data distribution without worrying about its normalization. Thus, in probabilistic modeling we often encounter unnormalized models. The following is an incomplete list of examples:

- Graphical models which represent conditional dependencies between the variables (undirected graphical networks, Markov networks) are unnormalized [2].
- In the modeling of images, the pixel value at a particular location is often assumed to only depend on the values of the pixels in its neighborhood. That is, the images are modeled as Markov networks (Markov random fields). Capturing the local interaction between the pixels is often enough to obtain a good global model of the image. Markov random fields are used in various image processing applications such as image restoration, edge detection, texture analysis, or object classification [3, 4].
- The structure of natural language (text) has been modeled using neural probabilistic language models (kind of neural networks) which specify unnormalized models [5]. Among other applications, neural probabilistic language models can be used for machine translation, sentence completion, or speech recognition [1].
- Unnormalized models occur in the area of unsupervised feature learning (representation learning), and deep learning [1], where a goal is to extract statistics from the data which are useful for classification or other tasks.
- Exponential random graphs which are used to model social networks [6] are unnormalized models. The presence or absence of links between nodes in a network are the (binary) random variables, and network statistics define the model. The models are usually unnormalized because summing over all network configurations to compute the partition function is rarely feasible in practice.
- We have used unnormalized models in our research in computational neuroscience [7, 8]. Making the basic hypothesis that the visual system is adapted to the properties of the sensory environment, we modeled natural image (patches) and related the learned features and computations to visual processing.

4. THE PARTITION FUNCTION IN MAXIMUM LIKELIHOOD ESTIMATION

Next, we show that the partition function is essential in maximum likelihood estimation.

Consider for instance the estimation of the precision of the zero mean univariate Gaussian with pdf as in (1). Given a sample with $n_x = 300$ data points x_i drawn from $f_x(u) = f(u;\theta^*)$ with $\theta^* = 1$, we can estimate the precision by maximizing the log-likelihood ℓ,

$$\ell(\theta) = \frac{n_x}{2}\log\frac{\theta}{2\pi} - \theta\sum_{i=1}^{n_x}\frac{x_i^2}{2}. \tag{7}$$

Figure 1 plots $\ell(\theta)$ (black curve), together with the variable-dependent part (blue dashed curve) and the part due to the normalizing partition function $Z(\theta)$ (red solid curve). The partition function "balances" the data-dependent term by punishing small precisions. This means that the partition function is essential for maximum likelihood estimation (MLE): Errors in the partition function translate immediately into errors in the estimate.

The importance of the partition function in MLE becomes also apparent if we consider estimating the unnormalized models in (2) and (3) by maximizing their "log-likelihood". The examples will show that maximizing the "likelihood" of an unnormalized model does not provide a meaningful estimator. We use the quotation marks because, strictly speaking, these models do not have a likelihood function as they do not specify a pdf. With their "log-likelihood" we mean the sum of the log-models over the data, in analogy to normalized models: For the unnormalized model in (2), the "log-likelihood" $\tilde{\ell}$ is the data-dependent part of $\ell(\theta)$,

$$\tilde{\ell}(\theta) = -\theta\sum_{i=1}^{n_x}\frac{x_i^2}{2}. \tag{8}$$

For the unnormalized model in (3), we obtain as "log-likelihood" $\breve{\ell}$,

$$\breve{\ell}(\boldsymbol{\theta}) = n_x\theta_2 - \theta_1\sum_{i=1}^{n_x}\frac{x_i^2}{2}. \tag{9}$$

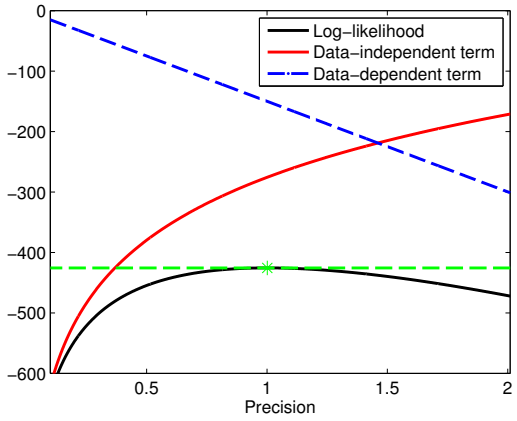

Figure 1: The log-likelihood of a Gaussian random variable with unknown precision (inverse variance). The log-likelihood consists of two balancing parts, the data-dependent and the normalizing part due to the partition function. The data consisted of $n_x = 300$ observations of a zero mean Gaussian with precision $\theta^* = 1$.

As the precision is positive, $\theta \to 0$ is maximizing $\tilde{\ell}$, and $\breve{\ell}(\boldsymbol{\theta})$ is maximized if the shape parameter $\theta_1 \to 0$ and the scaling parameter $\theta_2 \to \infty$. These estimates are obtained irrespective of the data and are not meaningful. From the example of $\breve{\ell}$, we find that separate estimation of the shape and scaling parameter is not possible by maximizing the "likelihood" of the unnormalized model.

In conclusion, for MLE, having an excellent model for the shape of the data distribution does not yield much if we do not know the proper scaling of the model in form of the partition function.

5. APPROACHES TO ESTIMATE UNNORMALIZED MODELS

We give here an overview of possible approaches to estimate unnormalized models. We assume that the partition function cannot be computed by analytical integration. Hereafter, an unnormalized model is thus an analytically unnormalizable model.

The previous section showed that maximizing the likelihood of unnormalized models does not lead to meaningful estimates. Hence, other estimation approaches need to be taken. The approaches can be divided into two categories: Those which approximate the partition function and those which avoid it.

5.1. Approximating the partition function

We present here two estimation methods that stay in the likelihood framework and approximate the intractable partition function, or the gradient of its logarithm, by numerical integration.

Numerical integration methods can be broadly divided into deterministic methods, like Simpson's rule, or (stochastic) Monte Carlo methods. Deterministic numerical integration becomes quickly computationally very expensive as the dimension m increases ("curse of dimensionality"). In practice, they may only be applied for $m \leq 3$. Monte Carlo integration is applicable for larger dimensions, and the two estimation methods reviewed here use this form of numerical integration.

The first method uses importance sampling to approximate the partition function as

$$Z(\boldsymbol{\theta}) \approx \frac{1}{n_y} \sum_{i=1}^{n_y} \frac{p(\boldsymbol{y}_i; \boldsymbol{\theta})}{f_y(\boldsymbol{y}_i)}, \qquad (10)$$

where the $\boldsymbol{y}_i$ are independent samples from a known auxiliary distribution f_y. The justification for the approximation is that for large n_y, it converges to $Z(\boldsymbol{\theta})$. Using this approximation in the log-likelihood gives a method called Monte-Carlo maximum likelihood estimation [9, 10]. A possible drawback is that the variance of the approximation in (10) may be unbounded if f_y decays more rapidly than $p(\boldsymbol{u}; \boldsymbol{\theta})$. Given the strong influence of the partition function in MLE, this mismatch between the two distributions results in an estimate with large variance.

The second method is obtained when the log-likelihood is maximized by steepest ascent. The gradient of the log-likelihood contains a term with the gradient of the log-partition function,

$$\nabla_{\boldsymbol{\theta}} \log Z(\boldsymbol{\theta}) = \int \frac{p(\boldsymbol{u}; \boldsymbol{\theta})}{Z(\boldsymbol{\theta})} \nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{u}; \boldsymbol{\theta}) \mathrm{d}\boldsymbol{u}, \qquad (11)$$

which is the expectation of $\nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{u}; \boldsymbol{\theta})$ under the model. The expectation is intractable if the partition function is intractable. The gradient can be approximated by a sample average where the samples are drawn from a Markov chain with $f(\boldsymbol{u}; \boldsymbol{\theta}) = p(\boldsymbol{u}; \boldsymbol{\theta})/Z(\boldsymbol{\theta})$ as target distribution. It is possible to draw the samples after only a few transitions of the chain: The resulting estimation method is known as contrastive divergence learning [11]. A possible drawback of this method is the sensitivity to the choice of the step-size in the optimization. If the step-size is too small, the learning is slow, if too large, it is unstable.

5.2. Avoiding the partition function

In this review, we focus on two methods which avoid the partition function. They are treated in Sections 6 and 7 in more detail.

In score matching [12], instead of inferring f_x or $\log f_x$ from the data, its slope $\boldsymbol{\Psi}_x(\boldsymbol{u}) = \nabla_{\boldsymbol{u}} \log f_x(\boldsymbol{u})$ is inferred. In the log-domain, the partition function corresponds to an additive offset, -$\log Z(\boldsymbol{\theta})$, and by considering the slope $\boldsymbol{\Psi}_x$, one gets rid of the partition function. As taking derivatives suggests, score matching is only applicable for continuous random variables, that is, if f_x is a pdf.

In noise-contrastive estimation [13], the partition function is avoided by replacing it with a scaling parameter. The partition function normalizes $p(\boldsymbol{u}; \boldsymbol{\theta})$ for all parameters $\boldsymbol{\theta}$, which is, however, not necessary for the purpose of estimation: It is enough that the model $p(\boldsymbol{u}; \hat{\boldsymbol{\theta}})$ after estimation is normalized, which can be achieved by having a scaling parameter as part of $\boldsymbol{\theta}$. An example of such a scaling parameter is θ_2 in (3).

6. SCORE MATCHING

6.1. The method

In score matching [12], parameter $\boldsymbol{\theta}$ is identified by minimizing the expected squared distance between the slope $\boldsymbol{\Psi}_x$ and the slope under the model, $\boldsymbol{\Psi}(\boldsymbol{u};\boldsymbol{\theta})$,

$$\boldsymbol{\Psi}(\boldsymbol{u};\boldsymbol{\theta}) \quad = \quad \nabla_{\boldsymbol{u}} \log p(\boldsymbol{u};\boldsymbol{\theta}), \tag{12}$$

that is, by minimizing

$$J^{\text{SM}}(\boldsymbol{\theta}) = \frac{1}{2}\,\mathrm{E}_x\,||\boldsymbol{\Psi}(\boldsymbol{x};\boldsymbol{\theta}) - \boldsymbol{\Psi}_x(\boldsymbol{x})||^2, \tag{13}$$

where E_x denotes the expectation with respect to f_x. The slope under the model is the Fisher score function with respect to a hypothetical location parameter. Minimizing J^{SM} thus consists in matching the score of the model to the score of the data, which gave the procedure its name.

The objective in (13) depends on the data Fisher score function $\boldsymbol{\Psi}_x$, which is unknown because the pdf f_x is unknown. However, under weak conditions, it is possible to compute J^{SM} up to a term not depending on $\boldsymbol{\theta}$ without actually knowing $\boldsymbol{\Psi}_x$ [12],

$$J^{\text{SM}}(\boldsymbol{\theta}) = \mathrm{E}_x \left[\sum_{i=1}^{m} \partial_k \Psi_k(\boldsymbol{x};\boldsymbol{\theta}) + \frac{1}{2}\Psi_k(\boldsymbol{x};\boldsymbol{\theta})^2 \right] + \text{const.} \tag{14}$$

Here, $\Psi_k(\boldsymbol{u};\boldsymbol{\theta})$ is the k-th element of the score $\boldsymbol{\Psi}(\boldsymbol{u};\boldsymbol{\theta})$ and $\partial_k \Psi_k(\boldsymbol{u};\boldsymbol{\theta})$ is its partial derivative with respect to the k-th argument,

$$\partial_k \Psi_k(\boldsymbol{u};\boldsymbol{\theta}) = \frac{\partial \Psi_k(\boldsymbol{u};\boldsymbol{\theta})}{\partial u_k} = \frac{\partial^2 \log p(\boldsymbol{u};\boldsymbol{\theta})}{\partial u_k^2}. \tag{15}$$

An important regularity condition needed to go from (13) to (14) is visible in the latter equation: $\log p(\boldsymbol{u};\boldsymbol{\theta})$ must be smooth enough so that its second derivative exists. If the optimization is performed by gradient-based methods, the third derivative needs to exist as well.

In practice, $J^{\text{SM}}(\boldsymbol{\theta})$ is computed by replacing the expectation in (14) with the sample average over the observed data. Parameter estimation consists in minimizing $J_T^{\text{SM}}(\boldsymbol{\theta})$,

$$J_T^{\text{SM}}(\boldsymbol{\theta}) = \frac{1}{n_x} \sum_{i=1}^{n_x} \sum_{k=1}^{m} \partial_k \Psi_k(\boldsymbol{x}_i;\boldsymbol{\theta}) + \frac{1}{2}\Psi_k(\boldsymbol{x}_i;\boldsymbol{\theta})^2, \tag{16}$$

which can be done with standard optimization tools.

Score matching has been used to estimate, for example, a Markov random field and a two-layer model of natural images [14, 7], as well as a model of coupled oscillators [15].

6.2. Simple example

We consider here the estimation of the precision for the unnormalized Gaussian in (2), or (3). The score function is in both cases

$$\Psi(u;\theta) = -\theta u, \tag{17}$$

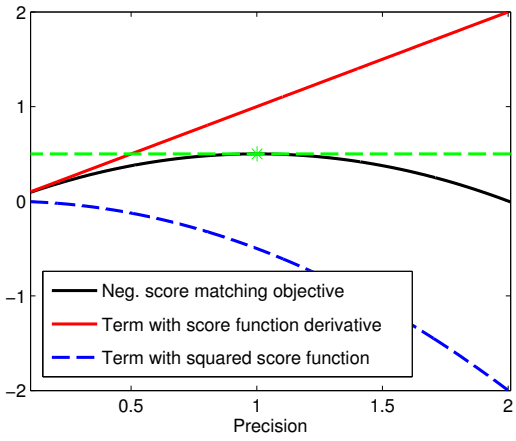

Figure 2: Estimation of the precision of a Gaussian by score matching, using the same data as in Figure 1.

and its derivative $\Psi'(u;\theta) = -\theta$. The score matching objective is

$$J_T^{\text{SM}}(\theta) = -\theta + \theta^2 \frac{1}{n_x} \sum_{i=1}^{n_x} \frac{x_i^2}{2}, \tag{18}$$

which we show in Figure 2. We plot the sign-inverted objective in order to facilitate the comparison with the log-likelihood. Like for the log-likelihood, the objective has two parts, visualized in red and blue, that balance each other. The optimum is at $\hat{\theta} = n_x/(\sum_{i=1}^{n_x} x_i^2)$ which is the same as the maximum likelihood estimator. In fact, for Gaussian distributions, the estimators obtained with score-matching and maximum-likelihood are always the same [12].

6.3. Score matching and denoising

Score matching has initially been proposed as presented above, namely based on computational considerations to avoid the partition function [12]. The score matching objective function J^{SM} is also obtained if optimal denoising is the goal. It occurs in two scenarios: One where $\boldsymbol{x}$ is the corrupted signal and one where $\boldsymbol{x}$ is the clean one. The corruption is additive uncorrelated Gaussian noise in both cases.

As for the first scenario, assume that $\boldsymbol{x}$ is the corrupted version of an unobserved random variable $\boldsymbol{\phi}$, $\boldsymbol{x} = \boldsymbol{\phi} + \sigma\boldsymbol{n}$, with $\boldsymbol{n}$ being a standard normal random variable. The estimate $\hat{\boldsymbol{\phi}}$ which minimizes the mean squared error

$$\mathrm{MSE}_1(\hat{\boldsymbol{\phi}}) = \mathrm{E}_{\boldsymbol{x},\boldsymbol{\phi}} \left(||\hat{\boldsymbol{\phi}}(\boldsymbol{x}) - \boldsymbol{\phi}||^2 \right), \tag{19}$$

is given by the posterior expectation, $\hat{\boldsymbol{\phi}} = \mathrm{E}_{\boldsymbol{\phi}|\boldsymbol{x}}\,\boldsymbol{\phi}$. It has been shown that the posterior expectation can be written in terms of the pdf of $\boldsymbol{x}$ only, without reference to the distribution of the unobserved $\boldsymbol{\phi}$ [16],

$$\hat{\boldsymbol{\phi}}(\boldsymbol{u}) = \boldsymbol{u} + \sigma^2 \nabla_{\boldsymbol{u}} \log f_x(\boldsymbol{u}) = \boldsymbol{u} + \sigma^2 \boldsymbol{\Psi}_x(\boldsymbol{u}). \tag{20}$$

If the score function $\boldsymbol{\Psi}_x$ is known, optimal denoising can be performed. If, however, the distribution of $\boldsymbol{x}$ is not

known but modeled by $p(\boldsymbol{u};\boldsymbol{\theta})$, with score function $\boldsymbol{\Psi}(\boldsymbol{u};\boldsymbol{\theta})$, the estimate depends on $\boldsymbol{\theta}$,

$$\hat{\boldsymbol{\phi}}(\boldsymbol{u};\boldsymbol{\theta}) = \boldsymbol{u} + \sigma^2 \boldsymbol{\Psi}(\boldsymbol{u};\boldsymbol{\theta}). \tag{21}$$

Consequently, also the mean squared error depends on $\boldsymbol{\theta}$, and it is natural to ask which parameter $\boldsymbol{\theta}$ yields the smallest error. The answer is that the optimal choice is given by the score matching estimator $\hat{\boldsymbol{\theta}} = \operatorname{argmin} J^{\text{SM}}(\boldsymbol{\theta})$ [16]. Hence, in order to optimally denoise $\boldsymbol{x}$, its pdf should be estimated by score matching.

The above result relates score matching to regression. Denoising score-matching [17] exploits this connection: The observed $\boldsymbol{x}$ is artificially corrupted to give $\boldsymbol{\chi} = \boldsymbol{x} + \sigma\boldsymbol{n}$ and the mean-squared error

$$\text{MSE}_2(\boldsymbol{\theta}) = \text{E}_{\boldsymbol{x},\boldsymbol{\chi}} \left(||\hat{\boldsymbol{x}}(\boldsymbol{\chi};\boldsymbol{\theta}) - \boldsymbol{x}||^2 \right) \tag{22}$$

is minimized, using $\hat{\boldsymbol{x}}(\boldsymbol{u};\boldsymbol{\theta}) = \boldsymbol{u} + \sigma^2 \tilde{\boldsymbol{\Psi}}(\boldsymbol{u};\boldsymbol{\theta})$ analogue to (21). The above result shows that the minimization of the mean-squared error allows one to estimate an unnormalized model for $\boldsymbol{\chi}$, but not for $\boldsymbol{x}$. The distribution of $\boldsymbol{\chi}$ is a smoothed version of f_x, and σ determines the strength of the smoothing.

As for the second scenario, assume now that only $\boldsymbol{\chi}$ is observed and that $\boldsymbol{x}$ is estimated from $\boldsymbol{\chi}$ as

$$\hat{\boldsymbol{x}}(\boldsymbol{\chi}) = \operatorname{argmax}_{\boldsymbol{u}} \log f_x(\boldsymbol{u}) - \frac{1}{2\sigma^2}||\boldsymbol{u} - \boldsymbol{\chi}||^2, \tag{23}$$

which is the maximum-a-posteriori (MAP) estimate under the additive noise model. The distribution f_x is the prior in the inference. If f_x is not known but modeled by $f(\boldsymbol{u};\boldsymbol{\theta})$, the estimate depends on $\boldsymbol{\theta}$,

$$\hat{\boldsymbol{x}}(\boldsymbol{\chi};\boldsymbol{\theta}) = \operatorname{argmax}_{\boldsymbol{u}} \log f(\boldsymbol{u};\boldsymbol{\theta}) - \frac{1}{2\sigma^2}||\boldsymbol{u} - \boldsymbol{\chi}||^2. \tag{24}$$

The parameter $\boldsymbol{\theta}$ can be chosen so that the mean-squared error is minimized. Assuming that both the noise level σ and the mean squared error are small (and of the same order), it has been shown that the optimal parameter is given by $\hat{\boldsymbol{\theta}} = \operatorname{argmin} J^{\text{SM}}(\boldsymbol{\theta})$ [18]. Hence, for small levels of noise, estimating the prior model by score matching minimizes (in a first-order approximation) the mean squared error for MAP inference.

6.4. Key properties

The following are key properties of score matching. On the positive side:

- Score matching yields a consistent estimator of $\boldsymbol{\theta}$ [12].
- For the continuous exponential family, J^{SM} is a convex quadratic form and thus relatively easy to optimize [19].
- Score matching does not rely on auxiliary samples unlike typical Monte Carlo methods.

On the negative side:

- Score matching only works for continuous random variables. Further, J^{SM} is only defined if the model is smooth.
- For some models, like multilayer networks used in deep learning, the analytical calculation of the derivatives in J^{SM} or its gradient can be difficult.

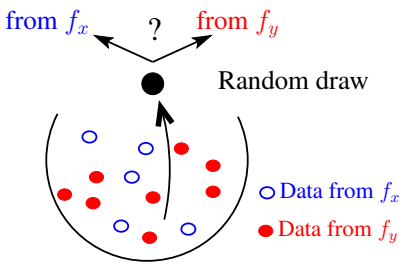

Figure 3: Noise-contrastive estimation formulates the estimation problem as a logistic regression task, the task of learning to distinguish between two data sets.

6.5. Extensions

Score matching has been extended in various ways. It has been modified to work with binary data (the resulting method is called ratio matching), or non-negative data [19]. Further, the idea of matching the model Fisher score to the data Fisher score has been generalized to matching $\mathcal{L}(p(\boldsymbol{u};\boldsymbol{\theta}))/p(\boldsymbol{u};\boldsymbol{\theta})$ to $\mathcal{L}(f_x(\boldsymbol{u}))/f_x(\boldsymbol{u})$ where $\mathcal{L}$ is a linear operator with the property that the mapping from p to $\mathcal{L}(p)/p$ is injective [20]. The unknown partition function is canceled in the ratio $\mathcal{L}(p(\boldsymbol{u};\boldsymbol{\theta}))/p(\boldsymbol{u};\boldsymbol{\theta})$, and the injectivity condition ensures that minimizing the squared distance between the transformed distributions can be used for parameter estimation. Another possibility is to modify the distance measure between the score functions in (13): The rather large class of Bregman divergences can be used instead of the Euclidean norm [21].

7. NOISE-CONTRASTIVE ESTIMATION

7.1. The method

Noise-contrastive estimation [22, 13] formulates the estimation problem as a logistic regression task, that is, the task of learning to discriminate between two data sets. Logistic regression works by estimating the ratio of the two distributions. The important point is that the distributions are not required to be normalized which allows for the estimation of unnormalized models.

In more detail, let $\boldsymbol{y}_i, i = 1 \ldots n_y$, be some auxiliary data that were independently drawn from a distribution f_y. Assume also that the $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ are mixed together and that the task is to decide whether a data point from the mixture is from f_x or f_y, see Figure 3. Logistic regression solves this task by estimating a regression function $h(\boldsymbol{u};\boldsymbol{\theta})$,

$$h(\boldsymbol{u};\boldsymbol{\theta}) = (1 + \nu \exp(-G(\boldsymbol{u};\boldsymbol{\theta}))^{-1}, \tag{25}$$

with $\nu = n_y/n_x$ and $G(\boldsymbol{u};\boldsymbol{\theta})$ being some function parametrized by $\boldsymbol{\theta}$. The regression function is the probability that the data point is from f_x. The factor ν biases the decision according to the relative frequency of the $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$. The regression function can be optimized by maximizing the negative log-loss $J_T^{\text{NCE}}(\boldsymbol{\theta})$,

$$J_T^{\text{NCE}}(\boldsymbol{\theta}) = \frac{1}{n_x}\Big(\sum_{i=1}^{n_x} \log h(\boldsymbol{x}_i;\boldsymbol{\theta}) + \sum_{i=1}^{n_y} \log[1 - h(\boldsymbol{y}_i;\boldsymbol{\theta})]\Big), \tag{26}$$

which is the sample version of

$$J^{\text{NCE}}(\boldsymbol{\theta}) = \mathrm{E}_x \log h(\boldsymbol{x}; \boldsymbol{\theta}) + \nu \, \mathrm{E}_y \log[1 - h(\boldsymbol{y}; \boldsymbol{\theta})], \quad (27)$$

where E_y denotes the expectation with respect to f_y.

Noise-contrastive estimation makes use of the fact that J^{NCE} is maximized by the parameter $\hat{\boldsymbol{\theta}}$ for which [13]

$$G(\boldsymbol{u}; \hat{\boldsymbol{\theta}}) = \log f_x(\boldsymbol{u}) - \log f_y(\boldsymbol{u}). \quad (28)$$

Hence, if f_y is known in closed from and $G(\boldsymbol{u}; \boldsymbol{\theta})$ specified as

$$G(\boldsymbol{u}; \boldsymbol{\theta}) = \log p(\boldsymbol{u}; \boldsymbol{\theta}) - \log f_y(\boldsymbol{\theta}), \quad (29)$$

the unnormalized model can be estimated by maximizing J^{NCE}, or, in practice J_T^{NCE}. The key point is that no assumption about normalization of the model is needed: We can work with the unnormalized model $p(\boldsymbol{u}; \boldsymbol{\theta})$ and if $\boldsymbol{\theta}$ contains a parameter which allows for scaling, maximizing J^{NCE} will automatically scale the model correctly [13]. In some cases, the model is rich enough so that no separate scaling parameter is needed.

In summary, noise-contrastive estimation of $p(\boldsymbol{u}; \boldsymbol{\theta})$ consists of the following three steps:

1. Choose a random variable $\boldsymbol{y}$ whose distribution f_y is known in closed form and where sampling is easy.
2. Sample $n_y = \nu n_x$ independent "noise" data points $\boldsymbol{y}_i \sim f_y$.
3. Perform logistic regression to discriminate between the $\{\boldsymbol{x}_i\}$ and $\{\boldsymbol{y}_i\}$: Maximize $J_T^{\text{NCE}}(\boldsymbol{\theta})$ in (26), using the log-ratio $G(\boldsymbol{u}; \boldsymbol{\theta})$, defined in (29), in the regression function $h(\boldsymbol{u}; \boldsymbol{\theta})$.

The objective J_T^{NCE} is maximized if $\hat{\boldsymbol{\theta}}$ is such that $G(\boldsymbol{u}; \hat{\boldsymbol{\theta}})$ takes, on average, large (positive) values for data from f_x and large negative values for data from f_y. These opposing requirements generate a balancing mechanism similar to what we have observed for likelihood-based estimation or score matching, visualized using the blue dashed and red solid curves in Figures 1 and 2.

The intuition behind noise-contrastive estimation is the idea of learning by comparison [23]: f_x is deduced from the difference between f_x and a known f_y, and the difference is learned from the data. This procedure is related to but more than classification: While in classification, we are interested in the decision boundary defined by $G(\boldsymbol{u}; \boldsymbol{\theta}) = 0$, here, for the purpose of estimating an unnormalized model, we are interested in the complete function $G(\boldsymbol{u}; \boldsymbol{\theta})$.

Examples where noise-contrastive estimation was used in practice include the estimation of two and three-layer models of natural images [13, 24, 8] and the estimation of models of natural language [25, 26].

7.2. Simple example

We estimate here the unnormalized Gaussian in (3) from the same data as before. The parameters are the precision

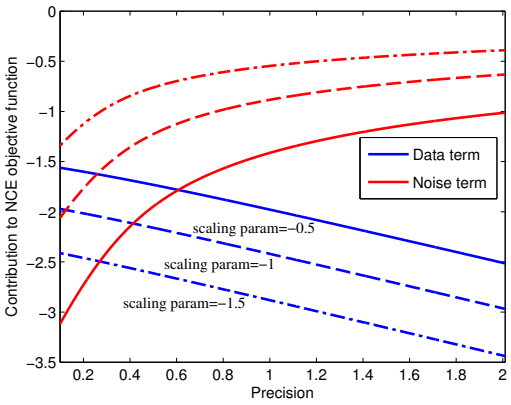

Figure 4: Balancing mechanism in noise-contrastive estimation of the precision of a Gaussian. The data-dependent part of J_T^{NCE} drives the precision to small values while the noise-dependent part drives it to large values.

θ_1 which is the parameter of primary interest and θ_2 which is the scaling parameter. As noise distribution, we take a zero mean Gaussian with precision $\tau_y = 1/2$, and we set the ratio ν to 10. The log-ratio $G(u; \boldsymbol{\theta})$ is

$$G(u; \boldsymbol{\theta}) = (\theta_2 - c_y) + \frac{1}{2}(\tau_y - \theta_1)u^2, \quad (30)$$

where $c_y = 1/2 \log(\tau_y/(2\pi))$. For fixed θ_2, $G(u; \boldsymbol{\theta})$ is maximized for $\theta_1 \to 0$ and minimized for $\theta_1 \to \infty$. The data-dependent part of the noise-contrastive objective function J_T^{NCE} drives θ_1 to small values while the noise-dependent part drives it to large values, see Figure 4. The objective function J_T^{NCE} combines these opposing requirements and thereby allows for estimation of $\boldsymbol{\theta}$.

Figure 5 shows a contour plot of J_T^{NCE} as a function of the precision θ_1 and the scaling parameter θ_2. Each point (θ_1, θ_2) corresponds to a model. The models on the black solid curve are normalized. The green lines show three optimization trajectories when J_T^{NCE} is optimized with a nonlinear conjugate gradient method. Starting from their initial points, the optimization trajectories traverse the space of unnormalized models. This visualizes the difference between estimating a scaling parameter and approximating the partition function: In the methods where the partition function is numerically approximated (estimated), the optimization trajectories would be constrained to (approximately) lie on the black curve; in noise-contrastive estimation, however, there is no such constraint and one can move freely in the space of unnormalized models towards the optimum. Due to the properties of the objective function, after optimization, the learned $\hat{\theta}_2$ is an estimate of the value which the partition function takes at $\hat{\theta}_1$. Hence, instead of approximating a function, only a normalizing scalar is here estimated.

7.3. The auxiliary distribution

The auxiliary distribution f_y influences the accuracy of the estimate. We next briefly discuss its choice, a longer

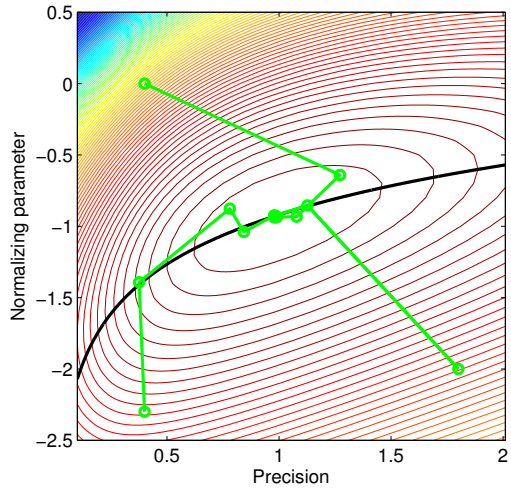

Figure 5: Contour plot of $J_T^{\text{NCE}}(\boldsymbol{\theta})$ for the estimation of an unnormalized Gaussian from the same data as in Figure 1. The parameters located on the black curve specify unnormalized models. Sample optimization trajectories are shown in green.

discussion can be found in our main reference on noise-contrastive estimation [13, Section 2.4].

We derived an expression for the asymptotic mean squared estimation error [13, Theorem 3]. Theoretically, it would thus be possible to choose f_y such that this error is minimized. Practically, however, one faces a couple of issues: First, the minimization is difficult. Second, the optimal auxiliary distribution will likely depend on the data distribution f_x, which is unknown in the first place. Third, we need to have an analytical expression for f_y available and also be able to sample from it easily, which is probably not the case for the optimal one.

In our work on natural images [13, 24, 8], satisfactory performance was obtained with choosing f_y to be a uniform distribution or a Gaussian distribution with the same covariance structure as the data.

For a specific choice of the auxiliary distribution, it is possible to relate noise-contrastive estimation to score matching [21]: Assume that $\boldsymbol{y}$ is obtained by shifting $\boldsymbol{x}$ by a small amount $\boldsymbol{\epsilon}$ so that $f_y(\boldsymbol{u}) = f_x(\boldsymbol{u}+\boldsymbol{\epsilon})$. Assume also that $p(\boldsymbol{u}+\boldsymbol{\epsilon};\boldsymbol{\theta})$ is used instead of $f_x(\boldsymbol{u}+\boldsymbol{\epsilon})$ in $G(\boldsymbol{u};\boldsymbol{\theta})$, and that $\nu = 1$. The objective $J^{\text{NCE}}(\boldsymbol{\theta})$ depends on the particular $\boldsymbol{\epsilon}$ chosen and may be denoted by $J_{\boldsymbol{\epsilon}}^{\text{NCE}}(\boldsymbol{\theta})$. From the more general proof given in previous work [21], it follows that if $\boldsymbol{\epsilon}$ is an uncorrelated random vector of variance σ^2, the averaged objective is

$$\mathrm{E}_{\boldsymbol{\epsilon}}\, J_{\boldsymbol{\epsilon}}^{\text{NCE}}(\boldsymbol{\theta}) = \text{const} - \frac{\sigma^2}{2}\,\mathrm{E}_x\left[\sum_{k=1}^{m}\partial_k \Psi_k(\boldsymbol{x};\boldsymbol{\theta}) + \frac{1}{2}\Psi_k(\boldsymbol{x};\boldsymbol{\theta})^2\right] + \mathrm{E}_{\boldsymbol{\epsilon}\boldsymbol{x}}\,\phi(\boldsymbol{\epsilon},\boldsymbol{x}), \quad (31)$$

where $\mathrm{E}_{\boldsymbol{\epsilon}}$ denotes expectation with respect to $\boldsymbol{\epsilon}$ and $\phi(\boldsymbol{\epsilon},\boldsymbol{x})$ is a function depending on $\boldsymbol{x}$ and third- or higher-order terms of $\boldsymbol{\epsilon}$. Maximizing the term of order σ^2 with respect to $\boldsymbol{\theta}$ is the same as minimizing J^{SM}.

7.4. Key properties

Noise-contrastive estimation has the following key properties. On the positive side:

- It yields a consistent estimator of $\boldsymbol{\theta}$ [22, 13].
- It is applicable to both continuous and discrete random variables, that is, f_x can be a pdf or a pmf [21].
- It is less sensitive to a mismatch between data and auxiliary distribution than importance sampling [27, 13, 25].
- The objective is algebraically not more complicated than the likelihood, and existing classifier architectures may be adapted to the estimation of unnormalized models.

On the negative side:

- It is not clear how to best choose the auxiliary distribution f_y in practice.
- The requirement that f_y needs to be known in closed form and that sampling is possible is an important limitation.

7.5. Extensions

The objective J^{NCE} is the sum of two expectations over functions that depend on the ratio $p(\boldsymbol{u};\boldsymbol{\theta})/f_y(\boldsymbol{u})$, with the first expectation being taken with respect to the data $\boldsymbol{x}$ and the second with respect to the noise $\boldsymbol{y}$, see (27). Figure 4 shows that the two terms balance each other. We investigated whether other kinds of functions are also suitable for consistent estimation of $\boldsymbol{\theta}$ [27]. We found that a rather large set of functions is suitable and derived a necessary condition for consistency; in later work, it was shown that this set is a special case of an even larger estimation framework for unnormalized models [21]. It is an open question which estimator of this framework to choose for a given model.

8. CONCLUSIONS

Unnormalized statistical models occur in various domains. Methods for their estimation can be broadly classified into those which are based on approximations of the partition function (or likelihood) and those which avoid the partition function. We reviewed two of the latter methods: Score matching and noise-contrastive estimation.

Score matching has the advantage that it does not require sampling. Its downside is that the models need to be smooth and that the objective function can get algebraically rather complicated for some models. Noise contrastive estimation does not have these drawbacks; its downside is the choice of the auxiliary distribution and that it needs to be known in closed form.

9. REFERENCES

[1] Y. Bengio, A.C. Courville, and P. Vincent, "Unsupervised feature learning and deep learning: A review and new perspectives," *arXiv*, vol. 1206.5538 [cs.LG], 2012.

[2] D. Koller, N. Friedman, L. Getoor, and B. Taskar, *Introduction to Statistical Relational Learning*, chapter Graphical Models in a Nutshell, pp. 13–55, MIT Press, 2007.

[3] A. Rangarajan and R. Chellappa, *The Handbook of Brain Theory and Neural Networks*, chapter Markov random field models in image processing, pp. 564–567, MIT Press, 1995.

[4] S. Z. Li, *Markov Random Field Modeling in Image Analysis*, Springer, 2009.

[5] J. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *Journal of Machine Learning Research*, vol. 3, pp. 1137–1155, 2003.

[6] G. Robins, P. Pattison, Y. Kalish, and D. Lusher, "An introduction to exponential random graph (p*) models for social networks," *Social Networks*, vol. 29, no. 2, pp. 173–191, 2007.

[7] U. Köster and A. Hyvärinen, "A two-layer model of natural stimuli estimated with score matching," *Neural Computation*, vol. 22, no. 9, pp. 2308–2333, 2010.

[8] M.U. Gutmann and A. Hyvärinen, "A three-layer model of natural image statistics," *Journal of Physiology-Paris*, 2013, in press.

[9] C.J. Geyer, "On the convergence of Monte Carlo maximum likelihood calculations," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 56, no. 1, pp. 261–274, 1994.

[10] A. Gelman, "Method of moments using Monte Carlo simulation," *Journal of Computational and Graphical Statistics*, vol. 4, no. 1, pp. 36–54, 1995.

[11] G.E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, no. 8, pp. 1771–1800, 2002.

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

[13] M.U. Gutmann and A. Hyvärinen, "Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics," *Journal of Machine Learning Research*, vol. 13, pp. 307–361, 2012.

[14] U. Köster, J. Lindgren, and A. Hyvärinen, "Estimating Markov random field potentials for natural images," in *Int. Conf. on Independent Component Analysis and Blind Source Separation*, 2009.

[15] C.F. Cadieu and K. Koepsell, "Phase coupling estimation from multivariate phase statistics," *Neural Computation*, vol. 22, no. 12, pp. 3107–3126, 2010.

[16] M. Raphan and E.P. Simoncelli, "Least squares estimation without priors or supervision," *Neural Computation*, vol. 23, no. 2, pp. 374–420, 2011.

[17] P. Vincent, "A connection between score matching and denoising autoencoders," *Neural Computation*, vol. 23, no. 7, pp. 1661–1674, 2011.

[18] A. Hyvärinen, "Optimal approximation of signal priors," *Neural Computation*, vol. 20, no. 12, pp. 3087–3110, 2008.

[19] A. Hyvärinen, "Some extensions of score matching," *Computational Statistics & Data Analysis*, vol. 51, pp. 2499–2512, 2007.

[20] S. Lyu, "Interpretation and generalization of score matching," in *Proc. Conf. on Uncertainty in Artificial Intelligence*, 2009.

[21] M.U. Gutmann and J. Hirayama, "Bregman divergence as general framework to estimate unnormalized statistical models," in *Proc. Conf. on Uncertainty in Artificial Intelligence*, 2011.

[22] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *Proc. Int. Conf. on Artificial Intelligence and Statistics*, 2010.

[23] M. Gutmann and A. Hyvärinen, "Learning features by contrasting natural images with noise," in *Proc. Int. Conf. on Artificial Neural Networks*, 2009.

[24] M.U. Gutmann and A. Hyvärinen, "Learning a selectivity–invariance–selectivity feature extraction architecture for images," in *21st Int. Conf. on Pattern Recognition*, 2012.

[25] A. Mnih and Y.W. Teh, "A fast and simple algorithm for training neural probabilistic language models," in *Proc. of the 29th Int. Conf. on Machine Learning*, 2012.

[26] M. Xiao and Y. Guo, "Domain adaptation for sequence labeling tasks with a probabilistic language adaptation model," in *Proc. of the 30th Int. Conf. on Machine Learning*, 2013.

[27] M. Pihlaja, M. Gutmann, and A. Hyvärinen, "A family of computationally efficient and simple estimators for unnormalized statistical models," in *Proc. Conf. on Uncertainty in Artificial Intelligence*, 2010.