LATENT VARIABLE MODELS

ABHISHEK SARKAR

1. Introduction

The essential aspect of a *latent variable model* (LVM) is that each observation $\mathbf{x}_i$ (which is p-dimensional, say), has an associated latent variable $\mathbf{z}_i$ (which is k-dimensional). Usually, one assumes k is much less than p, corresponding to an assumption that the differences in the high-dimensional observations $\mathbf{x}_i$ are (mostly) explained by differences in the lower-dimensional latent variables $\mathbf{z}_i$.

2. A simple example: Probabilistic PCA

One of the simplest examples of an LVM is *probabilistic PCA* (PPCA) [1]

$$\mathbf{x}_i \mid \mathbf{z}_i, \mathbf{W}, \sigma^2 \sim \mathcal{N}(\mathbf{W}\mathbf{z}_i, \sigma^2\mathbf{I}) \tag{1}$$

$$\mathbf{z}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{2}$$

In this model, observations $\mathbf{x}_i$ are generated by first applying the linear transform $\mathbf{W}$ to the low-dimensional latent variable $\mathbf{z}_i$, and then adding multivariate Gaussian noise with mean zero and covariance $\sigma^2\mathbf{I}$. There are two inference goals for PPCA:

(1) Estimate $\mathbf{W}, \sigma^2$ given observed data $\mathbf{x}_1, \ldots, \mathbf{x}_n$. This involves maximizing the marginal likelihood, that is, solving the optimization problem

$$\hat{\mathbf{W}}, \hat{\sigma}^2 = \arg\max_{\mathbf{W},\sigma^2} \sum_{i=1}^{n} \ln p(\mathbf{x}_i \mid \mathbf{W}, \sigma^2) \tag{3}$$

$$= \arg\max_{\mathbf{W},\sigma^2} \sum_{i=1}^{n} \ln \left(\int_{\mathcal{Z}} p(\mathbf{x}_i \mid \mathbf{z}_i, \mathbf{W}, \sigma^2) p(\mathbf{z}_i)\, d\mathbf{z}_i \right), \tag{4}$$

where $\mathcal{Z}$ denotes the set of values $\mathbf{z}_i$ can take.

(2) Estimate $\mathbf{z}_i$ given $\mathbf{x}_i$. This involves computing the posterior mean $\mathrm{E}[\mathbf{z}_i \mid \mathbf{x}_i, \hat{\mathbf{W}}, \hat{\sigma}^2]$, using the result $\hat{\mathbf{W}}, \hat{\sigma}^2$ from the previous step.

How does one estimate $\boldsymbol{\theta}$ given observed data $\mathbf{x}_1, \ldots, \mathbf{x}_n$? In the case of PPCA, the integrals in equation (4) have closed forms (following from properties of the Gaussian distribution); thus,

$$\hat{\mathbf{W}}, \hat{\sigma}^2 = \arg\max_{\mathbf{W},\sigma^2} \sum_{i=1}^{n} -\frac{1}{2} \ln \det(\mathbf{W}\mathbf{W}' + \sigma^2\mathbf{I}) - \frac{1}{2}\mathbf{x}_i'(\mathbf{W}\mathbf{W}' + \sigma^2\mathbf{I})^{-1}\mathbf{x}_i. \tag{5}$$

One can further show that the solution to the optimization problem (5) is unique and has closed forms related to the eigendecomposition of $\mathbf{X}'\mathbf{X}$, where $\mathbf{X}$ denotes the $n \times p$ matrix $(\mathbf{x}_1, \ldots, \mathbf{x}_n)'$. Specifically, $\hat{\mathbf{W}}$ is the matrix of the first k eigenvectors (having largest eigenvalues), and $\hat{\sigma}^2$ is the ratio of the sum of the remaining $p - k$ eigenvalues to the sum of all of the eigenvalues. (Equivalently, the closed forms are related to the singular value decomposition of $\mathbf{X}$.)

How does one estimate $\mathbf{z}_i$ given $\mathbf{x}_i$? One can think of $p(\mathbf{x}_i \mid \mathbf{z}_i, \mathbf{W}, \sigma^2)$ as the likelihood and $p(\mathbf{z}_i)$ as the prior in a Bayesian setting. Both are Gaussian; therefore, the posterior is also Gaussian (due to conjugacy):

$$\mathbf{z}_i \mid \mathbf{x}_i, \mathbf{W}, \sigma^2 \sim \mathcal{N}((\mathbf{W}'\mathbf{W} + \sigma^2\mathbf{I})^{-1}\mathbf{W}'\mathbf{x}_i, \sigma^2(\mathbf{W}'\mathbf{W} + \sigma^2\mathbf{I})^{-1}), \tag{6}$$

yielding the closed-form posterior mean $(\hat{\mathbf{W}}'\hat{\mathbf{W}} + \hat{\sigma}^2\mathbf{I})^{-1}\hat{\mathbf{W}}'\mathbf{x}_i$.

3. The general scheme

An LVM is typically given as a joint distribution

$$p(\mathbf{x}_i, \mathbf{z}_i \mid \boldsymbol{\theta}) = p(\mathbf{x}_i \mid \mathbf{z}_i, \boldsymbol{\theta})p(\mathbf{z}_i \mid \boldsymbol{\theta}), \tag{7}$$

where $\boldsymbol{\theta}$ denotes a collection of additional model parameters. One can think of $p(\mathbf{x}_i \mid \mathbf{z}_i, \boldsymbol{\theta})$ as the likelihood, and $p(\mathbf{z}_i \mid \boldsymbol{\theta})$ as the prior in a Bayesian setting. There are two inference goals for LVMs:

(1) Estimate $\boldsymbol{\theta}$ given observed data $\mathbf{x}_1, \ldots, \mathbf{x}_n$. This involves maximizing the marginal likelihood

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \ln \left(\int_{\mathcal{Z}} p(\mathbf{x}_i \mid \mathbf{z}_i, \boldsymbol{\theta})p(\mathbf{z}_i \mid \boldsymbol{\theta})\, d\mathbf{z}_i \right), \tag{8}$$

where $\mathcal{Z}$ denotes the set of values $\mathbf{z}_i$ can take. This problem may not be tractable, and may require approximate inference methods. (In the case of PPCA, the integrals had closed forms.) In a Bayesian setting, this task corresponds to estimating the prior from the data, which is termed *empirical Bayes*. (In the case of PPCA, the prior did not have any free parameters to be estimated; however, in other models, the prior does have such parameters.)

(2) Estimate $\mathbf{z}_i$ given $\mathbf{x}_i, \hat{\boldsymbol{\theta}}$. This involves computing the posterior mean, by combining the likelihood $p(\mathbf{x}_i \mid \mathbf{z}_i, \hat{\boldsymbol{\theta}})$ with the prior $p(\mathbf{z}_i \mid \hat{\boldsymbol{\theta}})$ estimated in the previous step. As above, this may not be tractable, and may require approximate inference methods. (In the case of PPCA, the prior was conjugate to the likelihood; therefore, the posterior mean had a closed form.)

One may also want to generate new observations $\mathbf{x}$. However, this is not "inference" as commonly understood in statistics; it is merely a case of random sampling from the distribution $p(\mathbf{x} \mid \hat{\boldsymbol{\theta}})$, which is achieved by first sampling $\mathbf{z} \sim p(\mathbf{z} \mid \hat{\boldsymbol{\theta}})$, then sampling $\mathbf{x} \sim p(\mathbf{x} \mid \mathbf{z}, \hat{\boldsymbol{\theta}})$.

4. Variational autoencoder

A more complex LVM is the *variational autoencoder* (VAE) [2]

$$\mathbf{x}_i \mid \mathbf{z}_i, \boldsymbol{\theta} \sim \mathcal{N}(f(\mathbf{z}_i), \sigma^2 \mathbf{I}) \tag{9}$$

$$\mathbf{z}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{10}$$

where f denotes a neural network with k-dimensional input and p-dimensional output, and $\boldsymbol{\theta}$ denotes σ^2 and the parameters of f. As a simple example, suppose f is a fully-connected feed-forward network with one hidden layer and a linear output layer, which can be written

$$f(\mathbf{z}_i) = \mathbf{W}_1 h(\mathbf{W}_0 \mathbf{z}_i + \mathbf{b}_0) + \mathbf{b}_1, \tag{11}$$

where h is a non-linearity (say, ReLU) applied element-wise. In this case, $\boldsymbol{\theta} = (\sigma^2, \mathbf{W}_0, \mathbf{W}_1, \mathbf{b}_0, \mathbf{b}_1)$, and the VAE is a non-linear version of PPCA. Specifically, one generates observations $\mathbf{x}_i$ by applying the non-linear transform f to the low-dimensional latent variable $\mathbf{z}_i$, and then adding multivariate Gaussian noise with mean zero and covariance $\sigma^2 \mathbf{I}$. As was the case for PPCA, the transform f will be learned from the data (in PPCA, the transform was $\mathbf{W}$); however, unlike PPCA, in general it does not have a closed form.

How does one estimate $\boldsymbol{\theta}$ given observed data $\mathbf{x}_1, \ldots, \mathbf{x}_n$? The integrals in equation (4) do not have closed forms for the VAE, so one needs to take a different strategy. The strategy taken for VAEs is *variational inference* [3].

4.1. Variational inference.

Recall the objective function,

$$\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \ln \left(\int_{\mathcal{Z}} p(\mathbf{x}_i \mid \mathbf{z}_i, \boldsymbol{\theta}) p(\mathbf{z}_i \mid \boldsymbol{\theta}) \, d\mathbf{z}_i \right) \tag{12}$$

Introducing a distribution $q(\mathbf{z}_i \mid \boldsymbol{\phi})$ (sometimes termed the *variational surrogate* or *variational approximation*) by multiplying and dividing by its density, and optimizing over its parameters $\boldsymbol{\phi}$ also,

$$= \max_{\boldsymbol{\phi}, \boldsymbol{\theta}} \sum_{i=1}^{n} \ln \left(\int_{\mathcal{Z}} p(\mathbf{x}_i \mid \mathbf{z}_i, \boldsymbol{\theta}) \frac{q(\mathbf{z}_i \mid \boldsymbol{\phi})}{q(\mathbf{z}_i \mid \boldsymbol{\phi})} p(\mathbf{z}_i \mid \boldsymbol{\theta}) \, d\mathbf{z}_i \right) \tag{13}$$

By the definition of expectation with respect to $q(\mathbf{z}_i \mid \boldsymbol{\phi})$,

$$= \max_{\boldsymbol{\phi}, \boldsymbol{\theta}} \sum_{i=1}^{n} \ln \left(\mathrm{E}_q \left[\frac{p(\mathbf{x}_i \mid \mathbf{z}_i, \boldsymbol{\theta}) p(\mathbf{z}_i \mid \boldsymbol{\theta})}{q(\mathbf{z}_i \mid \boldsymbol{\phi})} \right] \right) \tag{14}$$

Finally, one can exchange log and expectation using Jensen's inequality, yielding the *evidence lower bound* (ELBO)

$$\geq \max_{\boldsymbol{\phi}, \boldsymbol{\theta}} \sum_{i} \mathrm{E}_q [\ln p(\mathbf{x}_i \mid \mathbf{z}_i, \boldsymbol{\theta}) + \ln p(\mathbf{z}_i \mid \boldsymbol{\theta}) - \ln q(\mathbf{z}_i \mid \boldsymbol{\phi})]. \tag{15}$$

The ELBO (15) is the objective function for variational inference in LVMs, which can be optimized by a number of different algorithms, for example, gradient descent. (Variational inference in other models involves an objective function with similar form, involving the expectation of the log joint probability density $\mathrm{E}_q[\ln p]$ and the expectation of the log variational density $\mathrm{E}_q[\ln q]$.)

Before detailing how the ELBO (15) is optimized for VAEs specifically, first note that equation (13) holds for any distribution q (since we multiplied and divided by its density); however, one can show that the optimal q (equivalently, optimal $\boldsymbol{\phi}$) is the true posterior $p(\mathbf{z}_i \mid \mathbf{x}_i, \boldsymbol{\theta})$ (equivalently, the parameters of this distribution) [4]. Adding and subtracting $\ln p(\mathbf{x}_i \mid \boldsymbol{\theta})$ from the ELBO (15),

$$= \max_{\boldsymbol{\phi},\boldsymbol{\theta}} \sum_{i=1}^{n} \mathrm{E}_q[\ln p(\mathbf{x}_i \mid \boldsymbol{\theta}) + \ln p(\mathbf{x}_i \mid \mathbf{z}_i, \boldsymbol{\theta}) + \ln p(\mathbf{z}_i \mid \boldsymbol{\theta}) - \ln p(\mathbf{x}_i \mid \boldsymbol{\theta}) - \ln q(\mathbf{z}_i \mid \boldsymbol{\phi})] \tag{16}$$

Using the definition of conditional probability,

$$= \max_{\boldsymbol{\phi},\boldsymbol{\theta}} \sum_{i=1}^{n} \mathrm{E}_q[\ln p(\mathbf{x}_i \mid \boldsymbol{\theta}) - \ln p(\mathbf{z}_i \mid \mathbf{x}_i, \boldsymbol{\theta}) - \ln q(\mathbf{z}_i \mid \boldsymbol{\phi})] \tag{17}$$

Using linearity of expectation,

$$= \max_{\boldsymbol{\phi},\boldsymbol{\theta}} \sum_{i=1}^{n} \mathrm{E}_q[\ln p(\mathbf{x}_i \mid \boldsymbol{\theta})] - \mathrm{E}_q[\ln p(\mathbf{z}_i \mid \mathbf{x}_i, \boldsymbol{\theta}) - \ln q(\mathbf{z}_i \mid \boldsymbol{\phi})] \tag{18}$$

Since $p(\mathbf{x}_i \mid \boldsymbol{\theta})$ does not depend on $\mathbf{z}_i$, $\mathrm{E}_q[\ln p(\mathbf{x}_i \mid \boldsymbol{\theta})] = \ln p(\mathbf{x}_i \mid \boldsymbol{\theta})$, and

$$= \max_{\boldsymbol{\phi},\boldsymbol{\theta}} \sum_{i=1}^{n} \ln p(\mathbf{x}_i \mid \boldsymbol{\theta}) - \mathrm{E}_q[\ln p(\mathbf{z}_i \mid \mathbf{x}_i, \boldsymbol{\theta}) - \ln q(\mathbf{z}_i \mid \boldsymbol{\phi})] \tag{19}$$

The latter expectation is the definition of the KL divergence between $q(\mathbf{z}_i \mid \boldsymbol{\phi})$ and $p(\mathbf{z}_i \mid \mathbf{x}_i, \boldsymbol{\theta})$; it equals zero when the two distributions are the same.

$$= \max_{\boldsymbol{\phi},\boldsymbol{\theta}} \sum_{i=1}^{n} \ln p(\mathbf{x}_i \mid \boldsymbol{\theta}) - \mathcal{KL}\left(q(\mathbf{z}_i \mid \boldsymbol{\phi}) \| \, p(\mathbf{z}_i \mid \mathbf{x}_i, \boldsymbol{\theta})\right) \tag{20}$$

Thus, the optimal q (equivalently, optimal ϕ), holding $\boldsymbol{\theta}$ fixed, is the true posterior given $\boldsymbol{\theta}$ (equivalently, its parameters), in which case the inequality (15) becomes an equality,

$$= \max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \ln p(\mathbf{x}_i \mid \boldsymbol{\theta}). \tag{21}$$

Second, note that from equation (20), maximizing the ELBO with respect to $\boldsymbol{\phi}$ holding $\boldsymbol{\theta}$ fixed is equivalent to minimizing the KL divergence between

$q(\mathbf{z}_i \mid \boldsymbol{\phi})$ and $p(\mathbf{z}_i \mid \mathbf{x}_i, \boldsymbol{\theta})$ holding $\boldsymbol{\theta}$ fixed. Thus, the result will be the "best" approximating distribution q, in the sense that it has minimum KL divergence from the true posterior (given $\boldsymbol{\theta}$), even when one cannot compute the true posterior!

Third, note that using linearity of expectation and the definition of KL divergence, the ELBO (15)

$$= \max_{\boldsymbol{\phi},\boldsymbol{\theta}} \sum_i \mathrm{E}_q[\ln p(\mathbf{x}_i \mid \mathbf{z}_i, \boldsymbol{\theta})] - \mathcal{KL}\left(q(\mathbf{z}_i \mid \boldsymbol{\phi}) \| \, p(\mathbf{z}_i \mid \boldsymbol{\theta})\right) \tag{22}$$

revealing that the objective function can be decomposed into two terms:

(1) the *reconstruction error* $\mathrm{E}_q[\ln p(\mathbf{x}_i \mid \mathbf{z}_i, \boldsymbol{\theta})]$, that pushes the model to explain the data,
(2) the *regularization*, $\mathcal{KL}\left(q(\mathbf{z}_i \mid \boldsymbol{\phi}) \| \, p(\mathbf{z}_i \mid \boldsymbol{\theta})\right)$ that pushes the approximate posterior to be close to the prior.

4.2. **Solving the inference problems for VAEs.** For VAEs, one estimates $\boldsymbol{\theta}$ given observed data $\mathbf{x}_1, \ldots, \mathbf{x}_n$ by variational inference: optimizing the ELBO (15) with respect to $\boldsymbol{\phi}, \boldsymbol{\theta}$, where $\boldsymbol{\phi}$ denotes the parameters of a variational approximation q. This procedure also yields estimates $\hat{\boldsymbol{\phi}}$ of the variational parameters. Typically, one assumes that the approximation q is a Gaussian distribution with unknown mean and unknown diagonal covariance matrix

$$q(\mathbf{z}_i \mid \mathbf{x}_i, \boldsymbol{\phi}) = \mathcal{N}(\mathbf{m}_i, \mathrm{diag}(\mathbf{s}_i^2)), \tag{23}$$

where $\boldsymbol{\phi} = (\mathbf{M}, \mathbf{S})$, $\mathbf{M}$ denotes the $n \times k$ matrix $(\mathbf{m}_1, \ldots, \mathbf{m}_n)'$, and $\mathbf{S}$ denotes the $n \times k$ matrix $(\mathbf{s}_1, \ldots, \mathbf{s}_n)'$. One makes this assumption in order that the approximate posterior mean (and certain other quantities) will be easy to compute; the assumed distribution is almost surely not the true posterior. In this case, the result will be the Gaussian distribution with mean and diagonal covariance matrix closest to the true posterior (meaning, having minimum KL divergence to the true posterior, following from equation (20)).

Note that the variational approximation (23) is a simplification and not the way VAEs were originally described – there is clearly no "auto-encoding" going on. Nevertheless, this approach is equivalent in the essential aspects, as we detail below (and possibly better with regards to the objective function achieved in training). Also note that there is considerable subsequent work on relaxing the assumption (23), in order to get more flexible variational approximations that can be closer to the true posterior.

For VAEs, optimizing the ELBO (15) with respect to $\boldsymbol{\phi}, \boldsymbol{\theta}$ is still difficult since it does not have a closed form. Specifically, $\mathrm{E}_q[\ln p(\mathbf{x}_i \mid \mathbf{z}_i, \boldsymbol{\theta})]$ does not have a closed form due to the non-linear transform f; the remaining terms do have a closed form. (Note that the ELBO does have a closed form for other models.) To side-step this problem, one uses *stochastic optimization*: the ELBO (15) is replaced by a stochastic objective function that does have a closed form, whose expected value is the true objective function value, whose gradient can be computed using backpropagation, and whose

expected gradient is the true gradient [2]. We illustrate fitting VAEs using this approach in an online example[1].

How does one estimate $\mathbf{z}_i$ given $\mathbf{x}_i$? The approximate posterior mean is simply $\mathbf{m}_i$.

4.3. **Original description.** Note that VAEs are parameterized by a neural network f, originally termed the *decoder network* (since it "decodes" the low-dimensional latent variable $\mathbf{z}_i$ to produce the noiseless, uncorrupted version of the observed $\mathbf{x}_i$). As originally proposed, one chooses q to also be parameterized by a neural network, termed the *encoder network* (since it "encodes" $\mathbf{x}_i$ to produce $\mathbf{z}_i$). Specifically,

$$q(\mathbf{z}_i \mid \mathbf{x}_i, \boldsymbol{\phi}) = \mathcal{N}(m(\mathbf{x}_i), \mathrm{diag}(s^2(\mathbf{x}_i))), \tag{24}$$

where m, s^2 are each k-dimensional outputs (denoting the mean and diagonal of the covariance matrix, respectively) of a neural network taking p-dimensional input. This choice is the origin of the name "variational autoencoder".

If one assumes the variational approximation (24), then $\boldsymbol{\phi}$ denotes the neural network parameters of m, s^2. In this case, rather than finding (and storing) an optimal $\mathbf{m}_i^*, \mathbf{s}_i^*$ for each $\mathbf{x}_i$, one instead learns functions m, s^2 that map each $\mathbf{x}_i$ to its optimal $\mathbf{m}_i^*, \mathbf{s}_i^*$, an approach termed *amortized inference* (meaning, amortized over observations $\mathbf{x}_i$) [5]. One does not strictly need to do this; in fact, it results in worse objective function values at convergence if the encoder network is not powerful enough to represent the functions mapping $\mathbf{x}_i \mapsto \mathbf{m}_i^*, \mathbf{s}_i^*$, as we demonstrate in our online example. The main advantages of using amortized inference are:

(1) When n is large, the number of neural network parameters of m, s^2 may be smaller than the $2n \times k$ entries of $\mathbf{M}, \mathbf{S}$,
(2) The implementation of minibatch gradient descent to optimize the ELBO (15) when using an encoder network is simpler in commonly used libraries (e.g., `pytorch`, `tensorflow`)
(3) One can re-use the learned m, s^2 for new observations $\mathbf{x}_{i'}$. If one assumed the simpler variational approximation (23), then to fit a new observation $\mathbf{x}_{i'}$ one must re-optimize (15) with respect to $\mathbf{m}_{i'}, \mathbf{s}_{i'}$ holding $\boldsymbol{\theta}$ fixed.

References

1. Tipping, M. E. & Bishop, C. M. Probabilistic Principal Component Analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **61,** 611–622.
2. Kingma, D. P. & Welling, M. *Auto-Encoding Variational Bayes* in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings* (eds Bengio, Y. & LeCun, Y.) (2014).

[1] https://aksarkar.github.io/singlecell-ideas/vae-unamortized.html

3. Blei, D. M., Kucukelbir, A. & McAuliffe, J. D. Variational Inference: A Review for Statisticians. *Journal of the American Statistical Association* **112,** 859–877 (2017).
4. Neal, R. M. & Hinton, G. E. in *Learning in Graphical Models* (ed Jordan, M. I.) 355–368 (Springer Netherlands, Dordrecht, 1998).
5. Gershman, S. & Goodman, N. *Amortized inference in probabilistic reasoning* in *Proceedings of the annual meeting of the cognitive science society* **36** (2014).