AutoEncoder

ifding

Abstract

Probabilistic PCA, Variational AutoEncoder, ELBO surgery

1. Probabilistic PCA

Principal Component Analysis (known as PCA) is a technique used for dimension- ality reduction, lossy data compression and data visualization. PCA is a technique that can be analyzed from two different points of view: linear algebra and probability. One of Probabilistic PCA is that it can be used as simple generative model.

PCA finds the principal components of data, or in other words, it finds the features, the directions where there is the most variance. Given a dataset $\mathbf{X} = \{\mathbf{x}^{(i)}\}_1^N$ with dimensionality D_x, the main idea beind PCA is to obtain a subspace (called the principal-component subspace) with dimension D_z, being $D_z \ll D_x$. So that each $\mathbf{x}^{(i)}$ is represented with a $\mathbf{z}^{(i)}$ (latent variable) in the best possible way. It's possible to recover $\mathbf{x}^{(i)}$ from $\mathbf{z}^{(i)}$.

$$\boldsymbol{x}^{(i)} \in \Re^{D_x} \rightarrow \boldsymbol{z}^{(i)} \in \Re^{D_z} \tag{1}$$

The probabilistic PCA is a maximum likelihood solution of a probabilistic latent variable model in which all marginal and conditional distributions are Gaussian. The likelihood function to be maximized is expressed as:

$$p(\boldsymbol{x}) = \int p(\boldsymbol{x}|z)p(z)dz \tag{2}$$

As all distributions are Gaussian, the probability distribution of $\boldsymbol{x}$ can be expressed as $p(\boldsymbol{x}) = N(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{C})$, where the covariance matrix is $\boldsymbol{C} = \boldsymbol{W}\boldsymbol{W}^T + \sigma^2$. Therefore, the parameters of the model are the $D_x \times D_z$ matrix $\boldsymbol{W}$, the D_x dimensional vector $\boldsymbol{\mu}$ and the scalar σ^2.

The optimal values of the parameters are obtained using Maximum Likelihood Estimation (MLE), maximizing the log-likelihood $logp(x)$.

$$\log p\left(\boldsymbol{x}|\boldsymbol{W}, \boldsymbol{\mu}, \sigma^2\right) = \sum_{i=1}^{N} \log p\left(\boldsymbol{x}^{(i)}|\boldsymbol{W}, \boldsymbol{\mu}, \sigma^2\right) \tag{3}$$

The optimization can be computed in a closed form or using the EM algorithm. By closed form, the derivative of $\log p\left(\boldsymbol{x}|\boldsymbol{W}, \boldsymbol{\mu}, \sigma^2\right)$ with respect to each parameter is computed and equal to 0 to obtain the optimal values, namely $\boldsymbol{\mu}_{ML}$, $\boldsymbol{W}_{ML}$ and σ^2_{ML}.

The optimal mean $\boldsymbol{\mu}_{ML}$ is the sample mean of the data:

$$\boldsymbol{\mu}_{ML} = \sum_{i=1}^{N} \boldsymbol{x}^{(i)} = \bar{\boldsymbol{x}} \tag{4}$$

The σ^2_{ML} represents the average variance associated with the discarded dimensions.

$$\sigma^2_{ML} = \frac{1}{D_x - D_z} \sum_{i=D_z+1}^{D_x} \lambda_i \tag{5}$$

The generation phase comprises two steps. First, the variable $\boldsymbol{z}$ is sampled from a priori known distribution $p(\boldsymbol{z})$. Then, new data samples are drawn from $p(\boldsymbol{x}|\boldsymbol{z}) = N(\boldsymbol{x}|\mu(\boldsymbol{z}), \sigma^2\boldsymbol{I})$ where $\mu(\boldsymbol{z}) = \boldsymbol{W}\boldsymbol{z} + \boldsymbol{\mu}$. This process is shown in figure 1.

This is a simple case of a generative model where the relation between the latent and the predictive variables is linear:

$$\boldsymbol{x} \sim \boldsymbol{W}z + \boldsymbol{\mu} + \boldsymbol{\epsilon} \tag{6}$$

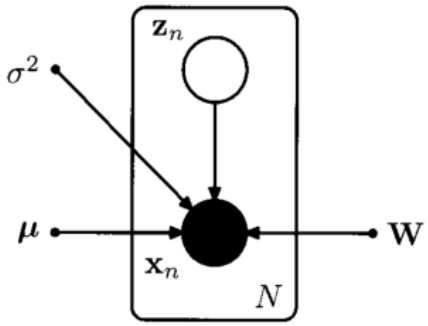

Figure 1: PCA generative model.

2. Variational AutoEncoder (VAE)

The VAE is a type of generative model, which is able to generate realistic data and obtain meaningful latent representations of the input. To achieve this, VAE is based on Bayesian inference. VAE models an approximation of the underlying probability distribution of input data $p(x)$ through a latent variable z. VAE can be understood as a non-linear Probabilistic PCA where the non-linearity is introduced using neural networks.

$$\begin{aligned} \text{Probabilistic } PCA: \quad & p(x|z) = N\left(Wz + \mu, \sigma^2 I\right) \\ VAE: \quad & p_\theta(x|z) = N\left(\mu_\theta(z), \sigma^2 I\right) \end{aligned} \tag{7}$$

What's a variational autoencoder? It's an autoencoder, with a random variable z defined in the latent space, that tries to maximize a lower bound of the data log-likelihood.

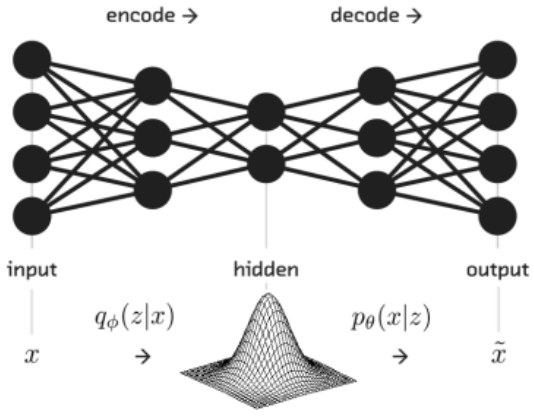

Figure 2: Probabilistic view of VAE.

The VAE is composed of two networks as shown in Figure 2 and Figure 3. The encoder or inference network (parameterized by ϕ) maps the input data into a latent representation that contains important attributes of the data. This latent representation is then mapped, through the decoder of generative network (parameterized by θ), into the reconstructed data.

Figure 3: Graphical models for the VAE showing the generative model (left) and the inference model (right).

The main characteristic of VAE is that the latent representation of the input is motivated to be similar to a known distribution. The usual choice is to define the prior distribution $p(z) = N(z|0, \boldsymbol{I})$. In this way, if the latent variable is distributed similarly to $N(z|0, \boldsymbol{I})$ and it holds the correct information about the input data, it's possible to obtain samples from z and then generate realistic data.

Here are some important points about VAE:

- Similar inputs will have similar representations in the latent space.
- The dimensions of the latent variable z should be smaller than the dimension of the input so that it captures only the most important features.
- VAE builds an approximation of the true distribution of the input: $p_g(\boldsymbol{x}) \approx p(\boldsymbol{x})$.
- A lower bound of $logp(\boldsymbol{x})$ called ELBO is the funtion that will be optimized.
- All density functions involved in the training process are prefixed. They are assumed to be Gaussian, it's possible to compute its derivatives and use optimization techniques.

2.1. Objective

Variational autoencoders approximately maximize the density function $p(\boldsymbol{x})$ of the training set according to the formula:

$$p(\boldsymbol{x}) = \int p(\boldsymbol{x}|z)p(z)dz \tag{8}$$

However, it's not possible to have infinite samples of z to compute the integral, the equation is intractable and

there will be losses. The objective function of the VAE, the Evidence Lower Bound (ELBO):

$$\log p(\boldsymbol{x}) \geq \boldsymbol{ELBO}(\theta, \phi, \boldsymbol{x}) = E_{q_\theta(z|\mathbf{x})}\left[\log p_\theta(\boldsymbol{x}|z)\right] - KL\left(q_\phi(z|\boldsymbol{x})\|p(z)\right) \tag{9}$$

It's important to mention that only a subset of $\mathbf{X} = \{\mathbf{x}^{(i)}\}_1^N$ samples from the true distribution $p^*(\boldsymbol{x})$ is available. Then, the marginal log likelihood will be approximated as the following equation, which is an unbiased estimator:

$$E_{p^*(\boldsymbol{x})}\left[\log p_\theta(\boldsymbol{x})\right] \approx \frac{1}{N}\sum_{i=1}^{N} \log p_\theta(x^{(i)}) \tag{10}$$

To reduce the memory needed to compute it, only a mini batch of size M (M $\ll$ N) will be considered at a time, the estimator will remain unbiased but with greater variance.

$$E_{p^*(x)}\left[\log p_\theta(\boldsymbol{x})\right] \approx \frac{1}{M}\sum_{i=1}^{M} \log p_\theta(\boldsymbol{x}^{(i)}) \tag{11}$$

Finally, the ELBO will be estimated using M samples $\{\mathbf{x}^{(i)}\}_1^M$

$$\frac{1}{M}\sum_{i=1}^{M} \log p_\theta\left(\boldsymbol{x}^{(i)}\right) \geq \frac{1}{M}\sum_{i=1}^{M} \boldsymbol{ELBO}(\theta, \phi; \boldsymbol{x}^{(i)}) \tag{12}$$

1. Dealing with the Integral over z

To maximize $p(\boldsymbol{x})$, the key idea is to approximate the integral with n samples.

$$p(\boldsymbol{x}) = \int p(\boldsymbol{x}|z)p(z)dz = E_{z\sim p(z)}\left[p_\theta(\boldsymbol{x}|z)\right] \tag{13}$$

Then, if it is possible to obtain n samples from z, the equation can be written as:

$$p(\boldsymbol{x}) \approx \frac{1}{n}\sum_{i=1}^{n} p_\theta(\boldsymbol{x}|z^{(i)}) \tag{14}$$

By convenience, the $logp_\theta(\boldsymbol{x}|z)$ will be used instead. It's assumed $p_\theta(\boldsymbol{x}|z)$ is distributed as a Gaussian $p_\theta(\boldsymbol{x}|z) = N(z|\mu_\theta(z), \Sigma)$. Therefore, its log likelihood is proportional to the euclidean distance between μ and $\boldsymbol{x}$ which is simpler to manipulate analytically.

$$\log p_\theta(\boldsymbol{x}|z) = k - \frac{1}{2}\left(\boldsymbol{x} - \mu_\theta(z)\right)^T \Sigma^{-1}\left(\boldsymbol{x} - \mu_\theta(z)\right) \tag{15}$$

where

$$k = -\log\left(|\Sigma|^{1/2}(2\pi)^{D_x/2}\right) \quad \Sigma = \sigma^2 \boldsymbol{I} \tag{16}$$

The value of σ^2 can be chosen, it's desirable to keep it small.

2. Inference Network: $q_\phi(z|\boldsymbol{x})$

According to equation 14, the approximation of $p(\boldsymbol{x})$ is better as the number of samples increases. However, the computation power needed also increases. In practice many values of z do not generate valid data. In conditional probability $p(\boldsymbol{x}|z)$ is close to 0 for the vast majority of z. This is why it is defined a family of conditional distributions $q_\phi(z|\boldsymbol{x})$ parameterized by ϕ that represents the distribution of z for each input datum. Thus, the new approximation is:

$$E_{z\sim q_\phi(z|\boldsymbol{x})}\left[\log\left(p_\theta(\boldsymbol{x}|z)\right]\right. \tag{17}$$

In this way, only values of z that are likely to have produced $\boldsymbol{x}$ are used to compute $p(\boldsymbol{x})$.

3. KL Divergence

The Kullback-Leibler divergence is a measure of how different are two probability functions p and q.

For discrete variables it is defines as follows:

$$KL(P(x)\|Q(x)) = \sum_{x\in X} P(x)\log\frac{P(x)}{Q(x)} \tag{18}$$

For a continuous variable it is defined as follows:

$$KL(p(x)\|q(x)) = \int p(x)\log\frac{p(x)}{q(x)}dx = E_{x\sim p(x)}[\log\frac{p(x)}{q(x)}]. \tag{19}$$

During the VAE optimization, the $KL\left(q_\phi(z|\boldsymbol{x})\|p(z|\boldsymbol{x})\right)$ should be close to 0.

$$KL\left(q_\phi(z|\boldsymbol{x})\|p(z|\boldsymbol{x})\right) = E_{z\sim q_\phi}\left[\log q_\phi(z|\boldsymbol{x}) - \log p(z|\boldsymbol{x})\right] \tag{20}$$

4. Core Equation

Applying Bayes rule in $KL\left(q_\phi(z|\boldsymbol{x})\|p(z|\boldsymbol{x})\right)$ and extracting $p(\boldsymbol{x})$ from the expectation because it does not depend on z:

$$p(z|x) = \frac{p(x|z)p(z)}{p(x)} \tag{21}$$

$$\begin{aligned} KL\left(q_\phi(z|\boldsymbol{x})\|p(z|\boldsymbol{x})\right) = E_{z\sim q_\phi}\left[\log q_\phi(z|\boldsymbol{x}) - \log p(z|\boldsymbol{x})\right] = \\ E_{z\sim q_\phi}\left[\log q_\phi(z|\boldsymbol{x}) - \log p_\theta(\boldsymbol{x}|z) - \log p(z)\right] + \log p(\boldsymbol{x}) = \\ -E_{z\sim q_\phi}\left[\log p_\theta(\boldsymbol{x}|z)\right] + E_{z\sim q_\phi}\left[\log q_\phi(z|\boldsymbol{x}) - \log p(z)\right] + \log p(\boldsymbol{x}) \end{aligned} \tag{22}$$

It's important to notice that the second term is a KL divergence. by rearranging terms, the fundamental equation of the VAE is obtained:

$$\begin{gathered}\log p(\boldsymbol{x}) - KL\left(q_\phi(z|\boldsymbol{x})\|p(z|\boldsymbol{x})\right) = \\ E_{z\sim q_\phi}\left[\log p_\theta(\boldsymbol{x}|z)\right] - KL\left(q_\phi(z|\boldsymbol{x})\|p(z)\right)\end{gathered} \tag{23}$$

Intuitively this equation tell us that the $\log p(\boldsymbol{x})$ minus the encoding error term, is equal to the mean of $\log p_\theta(\boldsymbol{x}|z)$ evaluated at z sampled from $q_\phi(z|\boldsymbol{x})$ minus an error term.

Because the $p(z|\boldsymbol{x})$ is unknown, it's not possible to calculate $KL\left(q_\phi(z|\boldsymbol{x})\|p(z|\boldsymbol{x})\right)$. This is why this term will be removed from the equation, leading to the loss function of the VAE called ELBO:

$$ELBO(\theta,\phi) = E_{z\sim q_\phi}\left[\log p_\theta(x|z)\right] - KL\left(q_\phi(z|x)\|p(z)\right) \tag{24}$$

- The $E_{z\sim q_\phi}\left[\log p_\theta(x|z)\right]$ term represents how good is the reconstrction of the input.
- The $KL\left(q_\phi(z|x)\|p(z)\right)$ term is a regularizer. In order to maximize the ELBO this term must be small. So this term forces $Q_\theta(z|\boldsymbol{x})$ to be similar to $p(z)$.

2.2. ELBO Optimization

In principle, all the distributions that appear in the ELBO are unknown, so they will be assumed to be Gaussian. This assumption is really convenience because it facilitates the computations since the ELBO will have a closed form. In addition, the Gaussian distribution has the minimal prior structure so it maximizes the entropy.

Then $q_\theta(z|\boldsymbol{x})$ is defined as:

$$q_\phi(z|\boldsymbol{x}) = N\left(\mu_\phi(\boldsymbol{x}), \Sigma_\phi(\boldsymbol{x})\right) \tag{25}$$

And $p(z)$ is defined as:

$$p(z) = N(0, \boldsymbol{I}) \tag{26}$$

Both μ and Σ are neural networks with parameters ϕ that are learnt during training. Moreover, Σ is defined as a diagonal matrix so that the $KL(q_\phi(z|\boldsymbol{x})||p(z|\boldsymbol{x}))$ has a closed form:

$$\begin{gathered}KL\left(N\left(\mu_\phi(\boldsymbol{x}), \Sigma_\phi(\boldsymbol{x})\right)\|N(0,\boldsymbol{I})\right) = \\ \frac{1}{2}\left[\mathrm{tr}\left(\Sigma_\phi(\boldsymbol{x})\right) - D_z - \log\det\left(\Sigma_\phi(\boldsymbol{x})\right) + \left(\mu_\phi(\boldsymbol{x})\right)^T\left(\mu_\phi(\boldsymbol{x})\right)\right]\end{gathered} \tag{27}$$

The other term involves solving the next integral:

$$E_{z\sim q_\phi}\left[\log p_\theta(\boldsymbol{x}|z)\right] = \int \log p_\theta(\boldsymbol{x}|z) q_\phi(z|\boldsymbol{x}) dz \tag{28}$$

It is approximated using the Monte Carlo estimator which is short it is an unbiased estimator, its variance is inversely proportional to the number of samples $\sigma^2 \propto \frac{1}{\sqrt{M}}$ and if M is large enough the estimated expectation is similar to the real one.

Then, the Monte Carlo estimator of $E_{z\sim q_\phi}\left[\log p_\theta(\boldsymbol{x}|z)\right]$ by taking M samples of z, $(z^{(1)}, \ldots, z^{(M)})$, from $q_\phi(z|\boldsymbol{x})$ is:

$$E_{z\sim q_\phi}\left[\log p_\theta(\boldsymbol{x}|z)\right] \approx \frac{1}{M}\sum_{i=1}^{M} \log p_\theta(\boldsymbol{x}|z^{(i)}) \tag{29}$$

$$\begin{gathered}\log p_\theta(\boldsymbol{x}|z^{(i)}) = \\ k - \frac{1}{2}(x - \mu_\theta(z^{(i)}))^T \Sigma^{-1}(x - \mu_\theta(z^{(i)}))\end{gathered} \tag{30}$$

where

$$\Sigma = \sigma^2 \boldsymbol{I}_{D_x} \quad k = -\log\left(|\Sigma|^{1/2}(2\pi)^{D_x/2}\right) \quad |\Sigma| = \left(\sigma^2\right)^{D_x} \tag{31}$$

σ^2 acts as a weighting factor between the two terms. This parameter depends on the choice of $p_\theta(\boldsymbol{x}|z)$. Our choice of σ determines how accurately we expect the model to reconstruct $\boldsymbol{x}$. If σ^2 is small, the first term outweighs the second one forcing $\boldsymbol{x} \approx \mu_\theta(z)$.

This formula looks more tractable, but it can not be optimized via backpropagation. Because it's necessary to obtain z samples in an intermediate layer to calculate $\log p_\theta(\boldsymbol{x}|z^{(i)})$:

$$x \to \mu_\phi, \Sigma_\phi \to z \sim N\left(\mu_\phi, \Sigma_\phi\right) \to \mu_\theta \to x \tag{32}$$

The forward pass of this network works fine and, if the output is averaged over many samples of $\boldsymbol{x}$ and z, it produces the correct expected value. However, it's not possible to back-propagate the error through stochastic layers. Sampling is a non-continuous operation and has no gradient.

2.3. Reparameterization Trick

In order to obtain the latent representation z of a given observation $\boldsymbol{x}$, it's necessary to sample it from the approximate posterior $q_\phi(z|\boldsymbol{x})$. The reparameterization trick consists in moving the sampling from an input layer. This is achieved defining a new input ϵ that introduces randomness into the model. Now, rather than sampling $z \sim N(\mu_\phi, \Sigma_\phi)$ directly, the sampling is done in $\epsilon \sim N(0, \boldsymbol{I})$

and then z is calculated in a deterministic way $z = \mu_\phi(\boldsymbol{x}) + \epsilon\sqrt{\Sigma_\phi(\boldsymbol{x})}$:

$$x, \epsilon \to \mu_\phi, \Sigma_\phi \to z = \mu_\phi + \sqrt{\Sigma_\phi} * \epsilon \to \mu_\theta \to x \tag{33}$$

In this way,backpropagation can be applied to optimize the network parameters.

The sampling of new data is quite straightforward: z is sampled from its prior distribution $p(z) = N(0, \boldsymbol{I})$ and injected into the decoder:

$$z \sim N(0, I) \to \mu_\theta(z) \to x \tag{34}$$

3. ELBO surgery

The variational expectation-maximization (EM):

$$p_\theta(\boldsymbol{x}) = \int p_\theta(\boldsymbol{x}|\boldsymbol{z}) p(\boldsymbol{z}) \mathrm{d}\boldsymbol{z} \tag{35}$$

where $p(\boldsymbol{z})$ is a prior on latent variables $\boldsymbol{z} = \{z_n\}_{n=1}^N$, $p_\theta(\boldsymbol{x}|\boldsymbol{z})$ is a likelihood on observations $\boldsymbol{x} = \{x_n\}_{n=1}^N$ parameterized by θ. The prior and likelihood follow from the generative model.

$$\begin{aligned} z_n &\overset{\text{iid}}{\sim} \mathcal{N}(0, I) \\ x_n|z_n \sim \mathcal{N}\left(\mu\left(z_n;\theta\right), \Sigma\left(z_n;\theta\right)\right), &\quad n = 1, 2, \ldots, N \end{aligned} \tag{36}$$

where the mean $\mu\left(z_n;\theta\right)$ and the covariance $\Sigma\left(z_n;\theta\right)$ depend on the latent variable z_n through a neural network with parameters θ.

We write joint densities as products of independent densities.

$$p(\boldsymbol{z}) = \prod_n p\left(z_n\right), \quad p_\theta(\boldsymbol{x}|\boldsymbol{z}) = \prod_n p_\theta\left(x_n|z_n\right) \tag{37}$$

The model is fit by maximizing the log evidence lower-bound (ELBO) $\mathcal{L}$,

$$\begin{aligned} \log p_\theta(\boldsymbol{x}) = \log \int p_\theta(\boldsymbol{z}, \boldsymbol{x}) \mathrm{d}\boldsymbol{z} = \log \int q_\phi(\boldsymbol{z}|\boldsymbol{x}) \frac{p_\theta(\boldsymbol{z}, \boldsymbol{x})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})} \mathrm{d}\boldsymbol{z} \\ \geq \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})} \log \frac{p_\theta(\boldsymbol{z}, \boldsymbol{x})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})} \triangleq \mathcal{L}(\theta, \phi), \end{aligned} \tag{38}$$

where each term in the variational density $q_\phi(\boldsymbol{x}|\boldsymbol{z}) = \prod_n q_\phi\left(z_n|x_n\right)$ is a Gaussian in which the mean $\mu(x_n;\phi)$ and covariance $\Sigma\left(x_n;\phi\right)$ depend on the observation x_n through a neural network with free parameters ϕ.

The variational distribution $q_\phi(z_n|x_n)$ acts as a stochastic "encoder" from an observation x_n to a distribution on the latent variable z_n, and the likelihood $p_\theta(x_n|z_n)$ acts as a stochastic "decoder" from the latent variable z_n to a distribution on the observation x_n.

There are several ways to rewrite the objective $\mathcal{L}(\theta, \phi)$, and each provides its own perspective.

3.1. Evidence minus posterior KL

One form of $\mathcal{L}(\theta, \phi)$ emphasizes that the lower bound becomes tighter as the variational distribution better approximates the posterior:

$$\mathcal{L}(\theta, \phi) = \log p_\theta(\boldsymbol{x}) - \mathrm{KL}\left(q_\phi(\boldsymbol{z}|\boldsymbol{x}) \| p_\theta(\boldsymbol{z}|\boldsymbol{x})\right) \tag{39}$$

Thus we cam improve the ELBO by improving the model log evidence $\log p_\theta(\boldsymbol{x})$, through the prior $p(\boldsymbol{z})$ or the likelihood $p_\theta(\boldsymbol{x}|\boldsymbol{z})$, or by improving the variational posterior approximation $q_\phi(\boldsymbol{z}|\boldsymbol{x})$.

3.2. Average negative energy plus entropy

Another way to rewrite the ELBO is as

$$\begin{aligned} \mathcal{L}(\theta, \phi) = \mathbb{E}_{q_\phi(z|\boldsymbol{x})}\left[\log p_\theta(\boldsymbol{z}, \boldsymbol{x})\right] + \mathbb{H}\left[q_\phi(\boldsymbol{z}|\boldsymbol{x})\right], \\ \mathbb{H}\left[q_\phi(\boldsymbol{z}|\boldsymbol{x})\right] \triangleq -\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})} \log q_\phi(\boldsymbol{z}|\boldsymbol{x}), \end{aligned} \tag{40}$$

where the log joint $\log p_\theta(\boldsymbol{z}, \boldsymbol{x})$ is interpreted as the negative energy in a Boltzmann distribution. Since we choose (θ, ϕ) to maximize the ELBO, this version highlights that a good posterior approximation $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ must assign most of its probability mass to regions of low energy (i.e. high joint probability density) while also maximizing the entropy of $q_\phi(\boldsymbol{z}|\boldsymbol{x})$.

This perspective is useful in contrasting variational EM with a maximum a-posteriori (MAP) approach; while MAP need only find a single value of $\boldsymbol{z}$ that maximizes the joint density (even if it lies in a region with very low posterior mass), the entropy term in the ELBO prevents $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ from collapsing to an atom.

3.3. Average term-by-term reconstruction minus KL to prior

Finally, we can write

$$\begin{aligned} \mathcal{L}(\theta, \phi) = \frac{1}{N} \sum_{n=1}^N \mathbb{E}_{q_\phi(z_n|x_n)}\left[\log p_\theta\left(x_n|z_n\right)\right] - \\ \mathrm{KL}\left(q_\phi\left(z_n|x_n\right) \| p\left(z_n\right)\right) \end{aligned} \tag{41}$$

For each observation index n, this version has a reconstruction term for the nth observation and a KL divergence from each encoding distribution to the prior. This KL-divergence term can be interpreted as a regularizer that "prune out" many of the latent dimensions in $\boldsymbol{z}$.

3.4. Terms of the average encoding distribution

For the last decomposition, what's a reasonable value for the KL-divergence term? Ideally it would be small, but we do not want it to approach 0 ($q_\phi(z_n|x_n) = p(z_n)$), since that would imply that x_n and z_n were almost independent, whereas virtually all of our modeling power comes from strongly coupling x_n to z_n. So if the KL term is large, is that a sign of underfitting, overfitting, or neither?

The *average encoding distribution* is defined as

$$q_\phi^{\text{avg}}(z) \triangleq \frac{1}{N}\sum_{n=1}^{N} q_\phi\left(z|x_n\right) \tag{42}$$

The *marginal* KL divergence $\mathrm{KL}(q_\phi^{\text{avg}}(z)\|p(z))$ is important because, unlike the individual terms $q_\phi(z_n|x_n)$, the average encoding distribution $q_\phi^{\text{avg}}(z)$ can be made arbitrarily close to the prior $p(z)$ without sacrificing model power. Indeed, if the data are drawn from the model, $x_n \sim p_\theta(x)$, and the posterior approximation is accurate, $q_\phi\left(z|x_n\right) \approx p_\theta\left(z|x_n\right)$, then for large N we would expect

$$\begin{aligned} p(z) = \int p_\theta(z|x)p_\theta(x)\mathrm{d}x = \mathbb{E}_{x\sim p_\theta(x)} p_\theta(z|x) \approx \\ \frac{1}{N}\sum_n p_\theta\left(z|x_n\right) \approx \frac{1}{N}\sum_n q_\phi\left(z|x_n\right) = q_\phi^{\text{avg}}(z) \end{aligned} \tag{43}$$

To simplify the notation, we drop parameter subscripts, and it's convenient to treat the index n as a random variable.

$$q(n,z) \triangleq q(n)q(z|n), \quad q(z|n) \triangleq q\left(z|x_n\right), \quad q(n) \triangleq \frac{1}{N} \tag{44}$$

$$p(n,z) \triangleq p(n)p(z|n), \quad p(z|n) \triangleq p(z), \qquad p(n) \triangleq \frac{1}{N} \tag{45}$$

where $p(z)$ denotes a standard Gaussian prior density form $z \sim \mathcal{N}(0, I)$, the average encoder distribution $q^{\text{avg}}(z)$ is now simply the marginal $q(z) = \sum_{n=1}^{N} q(z,n)$.

$$\begin{aligned} \frac{1}{N}\sum_{n=1}^{N} \mathrm{KL}\left(q\left(z_n|x_n\right)\|p\left(z_n\right)\right) &= \sum_n q(n,z)\log\frac{q(n,z)}{p(n,z)} \\ &= q(z)\log\frac{q(z)}{p(z)} + \sum_n q(n|z)q(z)\log\frac{q(n|z)}{p(n|z)} \\ &= \mathrm{KL}(q(z)\|p(z)) + \mathbb{E}_{q(z)}[\mathrm{KL}(q(n|z)\|p(n))] \\ &= \mathrm{KL}(q(z)\|p(z)) + \mathbb{I}_{q(n,z)}[n,z] \\ &= \mathrm{KL}(q(z)\|p(z)) + \left(\log N - \mathbb{E}_{q(z)}[\mathbb{H}[q(n|z)]]\right) \end{aligned} \tag{46}$$

where $\mathbb{I}_{q(n,z)}[n,z] = \mathbb{E}_{q(n,z)}\left[\log\frac{q(n,z)}{q(n)q(z)}\right]$ denotes the mutual information of n and z in $q(n,z)$, $p(n) = \frac{1}{N}$. The mutual information expression:

$$\begin{aligned} \mathbb{I}_{q(n,z)}[n,z] = \mathbb{E}_{q(z)}\left[\mathbb{E}_{q(n|z)}\left[\log\frac{q(n|z)}{q(n)}\right]\right] = \\ \log N - \mathbb{E}_{q(z)}[\mathbb{H}[q(n|z)]] \end{aligned} \tag{47}$$

The ELBO can be written in three terms:

$$\begin{aligned} \mathcal{L}(\theta,\phi) = &\underbrace{\left[\frac{1}{N}\sum_{n=1}^{N}\mathbb{E}_{q(z_n|x_n)}\left[\log p\left(x_n|z_n\right)\right]\right.}_{\text{(1) average reconstruction}} \\ &\underbrace{-\left(\log N - \mathbb{E}_{q(z)}[\mathbb{H}[q(n|z)]]\right)}_{\text{(2) index-code mutual info.}} \\ &\underbrace{-\quad \mathrm{KL}(q(z)\|p(z))}_{\text{(3) marginal KL to prior}} \end{aligned} \tag{48}$$

First, the two terms (1) and (2) are in tension with each other because to get a good average reconstruction score for (1), we typically need each encoding z_n to be specific to its corresponding observation x_n and hence $q(n|z)$ should have low entropy. Term (2) acts as a regularizer, in that it encourages the encodings $q(z|x_n)$ to overlap for distinct observation n, but this effect is likely to be weak relative to the reconstruction term (1). (2) is bounded above and below,

$$0 \leq \log N - \mathbb{E}_{q(z)}\mathbb{H}[q(n|z)] \leq \log N \tag{49}$$

Empirically, the reconstruction is precise and correspondingly, $q(z|n)$ is concentrated relative to $q(z)$, resulting in (2) is close to its maximum value of $\log N$.

Second, while $q(z)$ appears in all terms, $p(z)$ only appears in (3). Thus when considering choosing priors $p(z)$ to optimize the ELBO, only this term is affected. Observe that we could set (3) to zero without sacrificing model power by simply defining the prior to be $q(z)$. This choice would not be amenable to scalable computation because it is difficult to evaluate (2) in isolation: to normalize $q(n|z)$

at each evaluation requires accessing all N observations (and the normalization also precludes us from making unbiased Monte Carlo estimates). Setting (3) to zero may also be undesirable due to the potential for overfitting or the inability to use the prior to sculpt the latent representation. Because (3) can in principle be set to zero, whenever it is large it indicates a very strong and potentially unwanted regularization effect from the prior.

References

[1] Sánchez Martín, P., 2018. Unsupervised deep learning research and implementation of variational autoencoders (Bachelor's thesis).

[2] Bishop, Christopher M. Pattern recognition and machine learning. springer, 2006.

[3] Kingma, D.P. and Welling, M., 2013. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114.

[4] Hoffman, M.D. and Johnson, M.J., 2016, December. Elbo surgery: yet another way to carve up the variational evidence lower bound. In Workshop in Advances in Approximate Bayesian Inference, NIPS (Vol. 1).