Deep Learning Assignment 3

Jochem Loedeman
12995282

1 Variational Auto Encoders

1.1 Latent Variable Models

1.2 Decoder: The Generative Part of the VAE

Question 1.1

To able to sample from the given model we could follow the procedure of forward sampling. In this approach, we sample from the distributions of the random variables in the order that is implied by the corresponding graphical model. For the VAE, the graphical model consists of only two nodes Z and X, connected by a directed edge from Z to X. Therefore, we first sample from the distribution for Z which, for our current model, is a unit normal distribution. This gives us a vector $\boldsymbol{z}$, which we use as input to the neural network f_θ to calculate the M Bernoulli parameters $f_\theta(\boldsymbol{z})_m$, $m = \{1, \dots, M\}$. We can now obtain a sample image by sampling a pixel from each distribution $\text{Bern}(x, f_\theta(\boldsymbol{z})_m)$, with $m = \{1, \dots, M\}$ and where x is a single scalar value.

Question 1.2

Approximating $\log p(\boldsymbol{x}_n)$ using Monte-Carlo Integration is inefficient, because one would need a very large number of latent samples $\boldsymbol{z}_n^{(l)}$ to get an acceptable approximation. The reason for this is that for most $\boldsymbol{z}_n$, the likelihood $p(\boldsymbol{x}_n|\boldsymbol{z}_n)$ is a very small number. This is illustrated in Figure 2 in the assignment sheet, which suggests that the posterior $p(\boldsymbol{z}_n|\boldsymbol{x}_n)$ decreases rather steeply with increasing distance from the MAP solution for $\boldsymbol{z}_n$ (Note that $p(\boldsymbol{x}_n|\boldsymbol{z}_n)$ and $p(\boldsymbol{z}_n|\boldsymbol{x}_n)$ are proportional to each other through Bayes' rule). This effect increases with the latent dimensionality, since the average distance between randomly sampled points in latent space will strongly increase, causing the likelihoods $p(\boldsymbol{x}_n|\boldsymbol{z}_n)$ to be extremely small even more often.

Question 1.3

We obtain the smallest possible KL divergence by taking two identical Gaussians:

$$(\mu_q, \mu_p, \sigma_q^2, \sigma_p^2) = (0, 0, 1, 1)$$

For this pair of p, q, we have $\text{D}_{\text{KL}} = 0$. In order to obtain a large KL divergence, we should choose Gaussians that have little overlapping probability mass. Therefore, let

$$(\mu_q, \mu_p, \sigma_q^2, \sigma_p^2) = (0, 10, 1, 1)$$

The quadratic term in the difference between the Guassian means will make sure that the KL divergence is large.

Question 1.4

From Equation 14 in the assignment sheet we can immediately conclude that the RHS is a lower bound on $\log p(\boldsymbol{x}_n)$, because we subtract an always non-negative quantity from it (the KL divergence). Therefore, $\log p(\boldsymbol{x}_n)$ is always larger than or equal to the RHS, which implies that the RHS is a lower bound for $\log p(\boldsymbol{x}_n)$. We optimize the lower bound instead of $\log p(\boldsymbol{x}_n)$ itself, because the latter is very costly to compute (as discussed before). The ELBO however, is much easier to compute and therefore easier to optimize. By choosing a suitable variational distribution, we can ensure that the gap between them is sufficiently small.

Question 1.5

The lower bound can either be pushed up by optimizing it with respect to the model parameters or with respect to the variational distribution q_ϕ. In the first case, the KL divergence term on the LHS decreases to maintain the equality. For the second case, the log-likelihood term on the LHS increases. In other words, pushing up the lower bound can either increase the log-likelihood, or decrease the KL divergence term.

Question 1.6

Reconstruction loss is an appropriate name for the first term, because the distribution $p_\theta(\boldsymbol{x}|Z)$ can be used to create a "reconstruction" for an input $\boldsymbol{x}_n$ by sampling from it. We want reconstructions to be similar to the data points $\boldsymbol{x}_n$, which is why we want $p_\theta(\boldsymbol{x}_n|Z)$ to be as large as possible. This knowledge is then formulated as the loss $\mathcal{L}_n^{\text{recon}}$. The second term is appropriately called a regularization term, since it penalizes dissimilarity of the variational distribution q_ϕ with respect to the prior $p_\theta(Z)$. It ensures that q_ϕ does not become too complex.

Question 1.7

We start with $\mathcal{L}_n^{\text{recon}}$:

$$\begin{aligned}
\log p_\theta(\boldsymbol{x}_n|Z) &= \log \prod_{m=1}^{M} \text{Bern}\left(\boldsymbol{x}_n^{(m)}|f_\theta(Z)_m\right) \\
&= \sum_{m=1}^{M} \log\left(f_\theta(Z)_m^{\boldsymbol{x}_n^{(m)}}(1-f_\theta(Z)_m)^{1-\boldsymbol{x}_n^{(m)}}\right) \\
&= \sum_{m=1}^{M} \boldsymbol{x}_n^{(m)} \log\big(f_\theta(Z)_m\big) + (1-\boldsymbol{x}_n^{(m)}) \log\big(1-f_\theta(Z)_m\big)
\end{aligned}$$

Now, we will approximate the expectation of the above quantity using a single sample $\boldsymbol{z}_n^*$ from $q_\phi(Z|\boldsymbol{x}_n)$. Then,

$$\mathcal{L}_n^{\text{recon}} \approx -\sum_{m=1}^{M} \boldsymbol{x}_n^{(m)} \log\big(f_\theta(\boldsymbol{z}_n^*)_m\big) + (1-\boldsymbol{x}_n^{(m)}) \log\big(1-f_\theta(\boldsymbol{z}_n^*)_m\big)$$

Now, we work out the explicit form of $\mathcal{L}_n^{\text{reg}}$. We will use the fact that for two normal distributions [1],

$$D_{\text{KL}}(\mathcal{N}_0||\mathcal{N}_1) = \frac{1}{2}\left(\text{trace}(\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\Sigma}_0) + (\boldsymbol{\mu}_1-\boldsymbol{\mu}_0)^T\boldsymbol{\Sigma}_1^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_0) - D + \log\left(\frac{\det\boldsymbol{\Sigma}_1}{\det\boldsymbol{\Sigma}_0}\right)\right).$$

For the model at hand, we have $\mathcal{N}_0 = q_\phi(Z|\boldsymbol{x}_n) = \mathcal{N}\left(\boldsymbol{\mu}_\phi(\boldsymbol{x}_n), \text{diag}(\boldsymbol{\Sigma}_\phi(\boldsymbol{x}_n))\right)$ and $\mathcal{N}_1 = \mathcal{N}(0, \boldsymbol{I})$, and therefore

$$\mathcal{L}_n^{\text{reg}} = \frac{1}{2}\left(\sum_{i=1}^{D}\boldsymbol{\Sigma}_\phi(\boldsymbol{x}_n)_i + \boldsymbol{\mu}_\phi(\boldsymbol{x}_n)^T\boldsymbol{\mu}_\phi(\boldsymbol{x}_n) - D - \log\left(\prod_{i=1}^{D}\boldsymbol{\Sigma}_\phi(\boldsymbol{x}_n)_i\right)\right)$$

where we wrote the trace explicitly as the sum over the diagonal elements. The final objective can now be written as

$$\begin{aligned}\mathcal{L} &= \sum_{n=1}^{N} \mathcal{L}_n^{\text{recon}} + \mathcal{L}_n^{\text{reg}} \\ &\approx \sum_{n=1}^{N} \Bigg[- \sum_{m=1}^{M} \boldsymbol{x}_n^{(m)} \log\big(f_\theta(\boldsymbol{z}_n^*)_m\big) + (1 - \boldsymbol{x}_n^{(m)}) \log\big(1 - f_\theta(\boldsymbol{z}_n^*)_m\big) \\ &\quad + \frac{1}{2} \left(\sum_{i=1}^{D} \boldsymbol{\Sigma}_\phi(\boldsymbol{x}_n)_i + \boldsymbol{\mu}_\phi(\boldsymbol{x}_n)^T \boldsymbol{\mu}_\phi(\boldsymbol{x}_n) - D - \log\left(\prod_{i=1}^{D} \boldsymbol{\Sigma}_\phi(\boldsymbol{x}_n)_i \right) \right) \Bigg]\end{aligned}$$

Question 1.8

Computing $\mathcal{L}$ involves sampling from $q_\phi(\boldsymbol{z}_n|\boldsymbol{x}_n) = \mathcal{N}(\mu_\phi(\boldsymbol{x}_n), \text{diag}(\boldsymbol{\Sigma}_\phi(\boldsymbol{x}_n))$. Since this distribution depends on the parameters ϕ, we must backpropagate through this sampling procedure to be able to calculate the gradient $\nabla_\phi \mathcal{L}$. Since sampling is inherently stochastic and therefore non-differentiable, this is not possible. We can solve this problem by decoupling the sampling procedure from the computation step that includes the parameters ϕ. This is done by sampling from an external unit normal distribution ϵ, and transforming this variable with $\boldsymbol{\mu}_\phi(\boldsymbol{x}_n)$ and $\text{diag}(\boldsymbol{\Sigma}_\phi(\boldsymbol{x}_n))$ according to

$$\boldsymbol{z}_n^* = \boldsymbol{\mu}_\phi(\boldsymbol{x}_n) + \boldsymbol{\Sigma}_\phi(\boldsymbol{x}_n))^{1/2} \odot \epsilon$$

to obtain the sample. Now, ϕ does no longer have a path in the computation graph that goes through the sampling procedure.

Question 1.9

References

[1] C. Doersch. Tutorial on variational autoencoders. arxiv 2016. *arXiv preprint arXiv:1606.05908*, 2016.