Model 1: VAE with factorized Gaussian posteriors

Prashant Shekhar

March 23, 2023

Table of Contents

The model

The **inference/encoding** model: $q_\phi(z|x)$:

$$EncoderNeuralNet_\phi(x) \rightarrow (\mu, \log \sigma)$$

$$\epsilon \sim N(0, I)$$

$$z = \mu + \sigma \odot \epsilon$$

Here $\odot$ is an elementwise product.

- This is equivalent to saying $q_\phi(z|x) \equiv N(\mu, \Sigma)$, where μ and Σ are mean and covariance matrices and both of these are learnt by encoder neural network.
- Particularly Σ is a diagonal covariance matrix with squared elements of σ vector on the diagonal.
- The diagonal nature of Σ in the gaussian model $N(\mu, \Sigma)$ for the posterior $q_\phi(z|x)$ makes it a **factorized gaussian posterior**.

The **generative/decoding** model: $p_\theta(x|z)$

$$DecoderNeuralNet_\theta(z) \rightarrow \hat{x}$$

Computing ELBO

From previous lectures we know:

$$\mathcal{L}_{\theta,\phi}(x) = \mathbf{E}_{q_\phi(z|x)}[log(p_\theta(x,z)) - log(q_\phi(z|x))]$$

But instead of maximizing ELBO, as before, we prefer to minimize negative of ELBO. Hence we have:

$$\begin{aligned}
\mathcal{U}_{\theta,\phi}(x) &= -\mathcal{L}_{\theta,\phi}(x) \\
&= -\mathbf{E}_{q_\phi(z|x)}[log(p_\theta(x,z)) - log(q_\phi(z|x))] \\
&= \mathbf{E}_{q_\phi(z|x)}[log(q_\phi(z|x))] - \mathbf{E}_{q_\phi(z|x)}[log(p_\theta(x,z))] \\
&= \mathbf{E}_{q_\phi(z|x)}[log(q_\phi(z|x))] - \mathbf{E}_{q_\phi(z|x)}[log(p_\theta(x|z)p_\theta(z))] \\
&= \mathbf{E}_{q_\phi(z|x)}[log(q_\phi(z|x))] - \mathbf{E}_{q_\phi(z|x)}[log(p_\theta(x|z)) + log(p_\theta(z))] \\
&= \mathbf{E}_{q_\phi(z|x)}[log(q_\phi(z|x))] - \mathbf{E}_{q_\phi(z|x)}[log(p_\theta(z))] - \mathbf{E}_{q_\phi(z|x)}[log(p_\theta(x|z))] \\
&= \mathbf{E}_{q_\phi(z|x)}\left[log\left[\frac{q_\phi(z|x)}{p_\theta(z)}\right]\right] - \mathbf{E}_{q_\phi(z|x)}[log(p_\theta(x|z))]
\end{aligned}$$

Computing ELBO

contining from previous slide..

$$\mathcal{U}_{\theta,\phi}(x) = \mathbf{E}_{q_\phi(z|x)}\left[log\left[\frac{q_\phi(z|x)}{p_\theta(z)}\right]\right] - \mathbf{E}_{q_\phi(z|x)}[log(p_\theta(x|z))]$$

$$\approx \underbrace{\mathbf{E}_{q_\phi(z|x)}\left[log\left[\frac{q_\phi(z|x)}{p_\theta(z)}\right]\right]}_{\text{Encoder regularizarion}} + \underbrace{-log(p_\theta(x|z))}_{\text{Decoder reconstruction error}} \quad ;\ \text{Monte Carlo estimate}$$

Here:

- The encoder regularization term is the **KL divergence between** the inference/encoder model $q_\phi(z|x)$ and the standard multivariate gaussian $p_\theta(z) \sim N(0, I)$. This forces the encoder to learn simpler/meaningful representations by forcing it to be close to a gaussian.
- The decoder reconstruction error term is the **negative conditional likelihood** term which is minimized if the $\hat{x}$ produced by the decoder is very close to the encoder input x.

Computing ELBO

Term1: Encoder Regularization For $\mathbf{E}_{q_\phi(z|x)}\left[log\left[\frac{q_\phi(z|x)}{p_\theta(z)}\right]\right]$,

- $q_\phi(z|x) \sim N(\mu, \Sigma)$ where Σ is a diagonal matrix with σ_i values on the diagonal.
- $p_\theta(z) \sim N(0, I)$
- Hence: $\mu_1 = \mu$, $\mu_2 = 0$, $\Sigma_1 = \Sigma$ and $\Sigma_2 = I$ and assuming $z \in \mathbf{R}^m$
- Therefore:

$$\mathbf{E}_{q_\phi(z|x)}\left[log\left[\frac{q_\phi(z|x)}{p_\theta(z)}\right]\right] = D_{KL}(q_\phi(z|x)||p_\theta(z)) = \frac{1}{2}\left[-\sum_{i=1}^{m}\log\sigma_i^2 - m + \sum_{i=1}^{m}\sigma_i^2 + \sum_{i=1}^{m}\mu_i^2\right]$$

Term2: Decoder reconstruction error

- We can use Mean Squared Error (MSE). Suppose there are n samples and every sample has d features

$$MSE = (1/nd)\sum_{i=1}^{n}\sum_{j=1}^{d}(x_{ij} - \hat{x}_{ij})^2$$

Cost functions

Kullback-Leibler(KL) distance/divergence

- Kullback–Leibler divergence (also called relative entropy and I-divergence), denoted $D_{KL}(P||Q)$, is a type of statistical distance: a measure of how one probability distribution P is different from a second, reference probability distribution Q
- Assuming both P and Q have normal distributions with means μ_1 and μ_2 and variances Σ_1 and Σ_2 respectively. Then KL divergence from Q to P is:

$$
\begin{aligned}
D_{KL}(P||Q) &= \mathbf{E}_{P(x)}\left[\log\left[\frac{P(x)}{Q(x)}\right]\right] \\
&= \int [\log(P(x)) - \log(Q(x))]P(x)dx \\
&= \frac{1}{2}\left[\log\frac{|\Sigma_2|}{|\Sigma_1|} - d + Tr(\Sigma_2^{-1}\Sigma_1) + (\mu_2 - \mu_1)^T\Sigma_2^{-1}(\mu_2 - \mu_1)\right]
\end{aligned}
$$

Cross-Entropy loss function

- Also referred to as logarithmic loss, log loss or logistic loss.
- Each predicted class probability is compared with actual class label/probability of 0 or 1.
- Cross-entropy is defined as:

$$L_{CE} = -\sum_{i=1}^{m} p_i \log(q_i)$$

  where p_i is the true class label and q_i is the softmax probability of i^{th} class. Also, m is the number of classes.
- For example, if we have 3 classes (1/2/3) and for a sample, the target class is class 2, then the true class label vector can be: [0,1,0] and if at the last layer the predicted probabilities are $[q_1, q_2, q_3]$, then the loss is:

$$L_{CE} = -log(q_2)$$

This also shows why cross entropy loss is sometimes equivalent to negative log-likelihood

Mean Squared/Sum Squared loss function

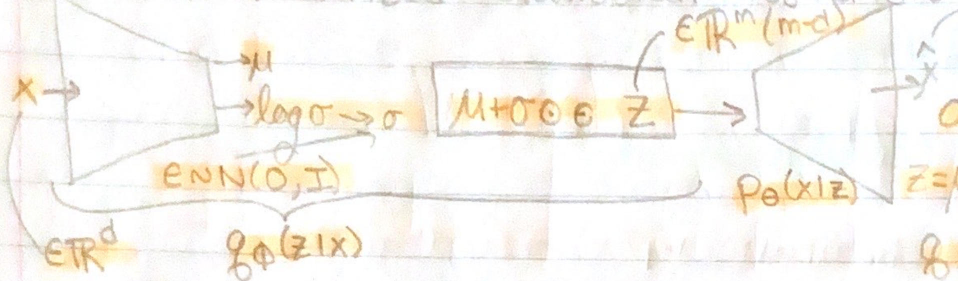
- Mainly used for regression problems.
- With n samples, if the true target value vector is $y \in \mathbf{R}^n$ and the predicted value vector is $\hat{y} \in \mathbf{R}^n$, then Sum Squared Error (SSE) is:

$$SSS = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

- And, Mean Squared Error (MSE) is:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

03/28/23 VAE with Factorized Gaussian Posteriors



$\in \mathbb{R}^d$

$\epsilon \sim N(0, I)$

$\sigma\epsilon \sim N(0, \sigma^2 I)$

$z = \mu + \sigma\epsilon \sim N(\mu, \sigma^2 I)$

$q_\phi(z|x) = N(\mu, \sigma^2 I)$

1.) $P_\theta(z) \sim N(0, I)$

2.) $q_\phi(z|x)$: neural network

3.) $P_\theta(x|z)$: neural network

- review of KL distance/divergence

$$D_{KL}(P \| Q) = E_{P(x)}\left[\log\left[\frac{P(x)}{Q(x)}\right]\right]$$

$$P(x) \sim N(\mu_1, \Sigma_1)$$

$$Q(x) \sim N(\mu_2, \Sigma_2)$$

$$= \frac{1}{2}\left[\log\left(\frac{|\Sigma_2|}{|\Sigma_1|}\right) - d + Tr(\Sigma_2^{-1}\Sigma_1) + (\mu_2 - \mu_1)^T \Sigma_2^{-1}(\mu_2 - \mu_1)\right]$$

- encoder regularization

$$D_{KL}(q_\phi(z|x) \| p_\theta(z))$$

$q_\phi(z|x) = N(\mu, \sigma^2 I)$ (μ_1, Σ_1)

$p_\theta(z) = N(0, I)$ (μ_2, Σ_2)

$$\log\left(\frac{|\Sigma_2|}{|\Sigma_1|}\right) = \log\left(\frac{1}{\prod_{i=1}^{m} \sigma_i^2}\right) = \log\left(\prod_{i=1}^{m} \sigma_i^{-2}\right)$$

$$= \sum_{i=1}^{m} \log(\sigma_i^{-2}) = -\sum_{i=1}^{m} \log(\sigma_i^2)$$

$d = m$

$$Tr(\Sigma_2^{-1}\Sigma_1) = Tr(I \sigma^2 I) = Tr(\sigma^2 I) = \sum_{i=1}^{m} \sigma_i^2$$

$$(\mu_2 - \mu_1)^T \Sigma_2^{-1} (\mu_2 - \mu_1) = -\mu^T_{(1\times m)} I (-\mu)_{(m\times 1)} = \sum_{i=1}^{m} \mu_i^2$$

($\mu_2 \to 0$, $\mu_1 \to \mu$, $\Sigma_2^{-1} \to I$, $\mu_2 \to 0$, $\mu_1 \to \mu$)

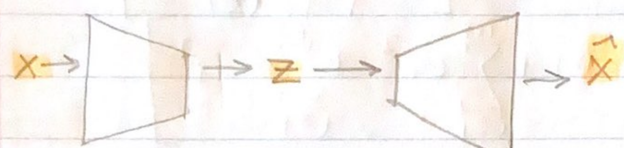
$$D_{KL}(q_\phi(z|x) \| p_\theta(z)) = E_{q_\phi(z|x)}\left[\log\left(\frac{q_\phi(z|x)}{p_\theta(z)}\right)\right]$$

$$= \frac{1}{2}\left[-\sum_{i=1}^{m} \log(\sigma_i^2) - m + \sum_{i=1}^{m} \sigma_i^2 + \sum_{i=1}^{m} \mu_i^2\right]$$

factorized because covariance is diagonal, meaning there are m uncoupled variables.

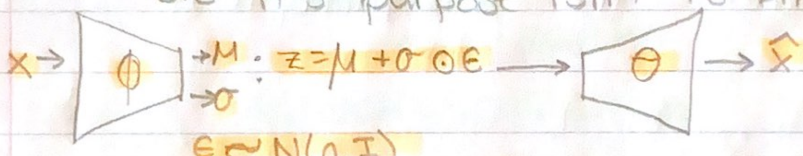
03/30/23 VAE Review

- autoencoder vs variational autoencoder



$x = \begin{bmatrix} 1000 \\ \times \\ 100 \end{bmatrix}$ $z \in \mathbb{R}^2$ $x' = \begin{bmatrix} 1000 \\ \times \\ 2 \end{bmatrix}$

autoencoder is for dimensionality reduction; would not create very good images b/c it's purpose isn't to find the pdf



VAE's are meant to find the pdf. It's purpose is to create/sample new images