Deriving KL Divergence for Gaussians

Posted on January 30, 2019
2 minutes read

If you read (implement) machine learning (and application) papers, there is a high probability that you have come across Kullback–Leibler divergence a.k.a. KL divergence loss. I frequently stumble upon it when I read about latent variable models (like VAEs). I am almost sure all of us know what the term means (don't worry if you don't as I have provided a brief explanation below and Google wil get you hundreds of resources on it), but may not have actually derived it till the end. In my opinion, deriving this term would make its implementation much clearer.

Below, I derive the KL divergence in case of univariate Gaussian distributions, which can be extended to the multivariate case as well 1.

What is KL Divergence?

KL divergence is a measure of how one probability distribution differs (in our case q) from the reference probability distribution (in our case p). Its valuse is always >= 0. Though, I should remind you that it is not a distance metric as it is not symmetric, KL(q || p) is not equivalent to KL(p || q).

KL(q || p) = Cross Entropy(q, p) - Entropy (q), where q and p are two univariate Gaussian distributions.

More specifically:

$$\begin{aligned} KL(q||p) &= -\int q(z)\log p(z)dz - (-\int q(z)\log q(z)dz) \\ &= -\int q(z)\log p(z)dz + \int q(z)\log q(z)dz \\ &= \int q(z)\log \frac{q(z)}{p(z)} \end{aligned}$$

KL Divergence for Gaussian distributions?

We know that PDF of Gaussian distribution can be written as:

$$q(z;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}e^{-}\frac{(z-\mu)^2}{2\sigma^2}$$

After taking the logarithm of the PDF above we get:

$$\begin{aligned}\log q(z;\mu,\sigma^2) &= \log(\frac{1}{\sqrt{2\pi\sigma^2}}) + (-\frac{(z-\mu)^2}{2\sigma^2}) \\ &= -\frac{1}{2}\log(2\pi\sigma^2) - \frac{(z-\mu)^2}{2\sigma^2}\end{aligned}$$

Let's also assume that we have that our two distributions have parameters as follows: $q(z) \sim N(\mu, \sigma^2)$ and $p(z) \sim N(0,1)$.

To add some more context in terms of latent variable models, we try to fit an approximate posterior to the true posterior by minimizing the *reverse KL divergence* (computationally better than the forward one, read more here 2). Think of z as the latent variable, $q(z)$ as the approximate distribution and $p(z)$ as the prior distribution. Usually, we model q and p as Gaussian distributions. Prior distribution is assumed to have mean of 0 and variance of 1 (standard Normal distribution) and parameters of q are the output of the inference (encoder) network.

Now, let's look at Cross Entropy and Entropy seperately for ease of evaluation.

Entropy

$$\begin{aligned}-\int q(z)\log q(z)dz &= \int q(z)[\frac{1}{2}\log(2\pi\sigma^2) + \frac{1}{2\sigma^2}(z-\mu)^2]dz \\ &= \frac{1}{2}\log(2\pi\sigma^2)\int q(z)dz + \frac{1}{2\sigma^2}\int (z-\mu)^2 q(z)dz \\ &= \frac{1}{2}\log(2\pi\sigma^2) + \frac{1}{2} \\ &= \frac{1}{2}\log(2\pi) + \frac{1}{2}\log(\sigma^2) + \frac{1}{2}\end{aligned}$$

Cross Entropy

$$\begin{aligned}-\int q(z)\log q(z)dz &= \int q(z)[\frac{1}{2}\log(2\pi\sigma^2) + \frac{1}{2\sigma^2}(z-\mu)^2]dz \\ &= \frac{1}{2}\log(2\pi)\int q(z)dz + \frac{1}{2}\int z^2 q(z)dz \\ &= \frac{1}{2}\log(2\pi) + \frac{1}{2}(\mu^2+\sigma^2)\end{aligned}$$

Note that:

1. The integral over a PDF is always 1 $\int q(z)dz = 1$.
2. And, expectation over square of a random variable is equivalent to sum of square of mean and variance $\int z^2 q(z)dz = \mu^2 + \sigma^2$.

Cross Entropy - Entropy

Now let's put both the terms together:

$$KL(q||p) = \frac{1}{2}\log(2\pi) + \frac{1}{2}(\mu^2 + \sigma^2) - \frac{1}{2}\log(2\pi) - \frac{1}{2}\log(\sigma^2) - \frac{1}{2}$$
$$= -\frac{1}{2}(1 + \log(\sigma^2) + -\mu^2 - \sigma^2)$$

By stretch of the imagination, the above equation could be generalized to multivariate cases (D dimensions) by summing over all the dimensions:

$$KL(q||p) = -\frac{1}{2}\sum_{d=1}^{D}(1 + \log(\sigma^2) + -\mu^2 - \sigma^2)$$

The above equation can be easily implemented in frameworks like Pytorch. I hope the post helped you to understand this concept a little better!

References:

1. Auto-Encoding Variational Bayes by Kingma and Welling (https://arxiv.org/abs/1312.6114)
2. KL-divergence as an objective function by Tim Vieira (https://timvieira.github.io/blog/post/2014/10/06/kl-divergence-as-an-objective-function/)
3. Allison Chaney for the post image.

(htt

NEXT POST → (/2019-06-13-ACL-2019-PAPERS/)

(mailto:leena.mesra@gmail.com) (https://github.com/leenashekhar)