Re-derivation of Out of Distribution Paper

Arjun Gupta

September 5, 2019

This document is a sheet to do some derivations /explanations for the Out of Distribution Paper (https://arxiv.org/pdf/1907.04572.pdf). The majority of the explanation for NRM is based on the original paper at (https://arxiv.org/abs/1811.02657).

0.1 Latent Variable Likelihood

The Out of Distribution Paper uses the likelihood of latent variables in the Neural Rendering Model to assess whether a given sample is in distribution or out of distribution. The neural rendering model re-creates the image by inverting the process the CNN uses to make the prediction. **An important realization is that the NRM model is not just the decoder. It is the combination of the CNN at the beginning and the decoder at the end.** The NRM model uses the same weights and filters ($W(l)^T$) as the CNN transposed to regenerate the image. The latent variables are the relu activations in each layer $s(l)$ and the maximum region of the max pooling $t(l)$. The optimal latent variables $z^*(l) = \{s^*(l), t^*(l)\}$ are derived from the forward pass of the CNN used to get the class probabilities. The process for generating an image in NRM is:

1. Feed the image x into the CNN, keep track of activations $s^*(l)$ and the locations of max pooling $t^*(l)$. The result is a vector of class probabilities y representing $p(y|x)$.

2. Based on $p(y|x)$, we choose a class template μ_y as our coarsest image $h(L)$.

3. Latent variables $z^*(l) = \{s^*(l), t^*(l)\}$ from the CNN step are used at each layer to iteratively refine the base image using the formula:

$$h(l-1) = \sum_{p \in h(l)} T(t^*(l,p))B(l,p)W^T(l)(s^*(l,p)h(l,p)) \tag{1}$$

   Which uses $s^*(l)$ to select whether a given template is to be rendered or not. $W(l)^T$ is the transposed weight at the corresponding layer of the CNN. $B(l,p)$ pads the template (which is the size of the convolution filter) with zeros to make it the size of the image. $T(t^*l,p)$ is the translation matrix, which translates the rendered template to the area of max pooling if applicable.

4. The final rendered image then has noise added to it:

$$x|\{z,y\} = N(h(0), \sigma^2 I) \tag{2}$$

   Why is there noise added?

Therefore, the process for generating the output at each layer requires that we do deconvolutions with each of the filters on the pixels that were activated. The distribution of the latent variables $z = \{t(l), s(l)\}$ can be seen as based on a structured prior. The paper claims this prior is:

$$\eta(y,z) = \sum_{l=1}^{L} \langle b(l,t), s(l) \odot h(l) \rangle = \sum_{l=1}^{L} b^T(l)(s(l) \odot h(l))$$
$$p(z|y) = \pi_{z|y} = \frac{\exp(\eta(y,z))}{\sum_{z_i} \exp(\eta(y,z_i))} \tag{3}$$

Where $b(l)$ are the parameters of the prior distribution, $s(l)$ are the activations, and $t(l)$ are the translations for layer l. All three of these are of the dimensions of $h(l)$ on which they are acting. They construct this prior specifically because it is the conjugate prior for the model likelihood. Because it is a conjugate prior, the parameters $b(l)$ in the prior become the biases in the CNN (**I am still a little shaky on the math behind this**). In words, $s(l) \odot h(l)$ zeros any pixels from the intermediate rendered image $h(l)$ that did not have activations in the CNN. $b(l)$ weights the distribution (?). The author's reasoning for this form of the prior is that the individual terms $b^T(l)(s(l) \odot h(l))$ mimic the piecewise linear nature of CNN estimation, and that this form allows it to be a conjugate prior to the the posterior distribution $p(y, z|x)$. The other important feature of this prior is that puts a structured dependency on the variable $p(z(l)|z(l+1), .., z(L))$. **This dependency is implicit in $h(l)$, since the intermediate rendered template $h(l)$ is based on the the series of latent variables $z(l+1), ..., z(L)$ that came before. When the previous layers had activations for a particular pixel, the value at $h(l)$ will be higher, and therefore a z that also has that pixel activated will be more likely based on this formulation, thus enforcing the constraint they describe that they expect subsequent activations to be in similar locations in the image.**

It is also important to note that the forward pass of the CNN "samples" the optimal latent variables z^* from the prior distribution. This is what Theorem 3.2 in the NRM paper shows.

How do you compute $p(z|y)$ since the bottom of the probability seems relatively expensive to compute? Answer: Compute the bottom preemtively, (it is always the same for a given class y). I am not sure how the paper does it though.

0.2 Data Likelihood

The second theorem that they use in the out of distribution paper is used to estimate the probability of the data x_i:

$$\log p(x_i) \leq E_{y_i,z_i}\left[\log p\left(x_i, (y_i, z_i)\right)\right] \approx -\frac{1}{2\sigma^2}||x_i - h(y_i^*, z_i^*, 0)||^2 + \log \pi_{z_i^*|y_i^*} \tag{4}$$

This is effectively just a combination of the Reconstruction loss (first term) and the likelihood of the latent variables (second term). The first in equality is trivially true. The second is less trivial but can be explained as follows:

$$
\begin{aligned}
p(x,y,z) &= p(x|y,z)\cdot p(z|y)\cdot p(y) \\
&= \frac{1}{\sigma\sqrt{2\pi}}\exp(-\frac{(x-h(y,z,0))^2}{2\sigma^2})\cdot \pi_{z|y}\cdot p(y) \\
\log(p(x,y,z)) &= -\log(\sigma\sqrt{2\pi}) - \frac{1}{2\sigma^2}(x-h(y,z,0))^2 + log\pi_{z|y} + log(p(y)) \\
E[\log p(x,y,z)] &= -\log(\sigma\sqrt{2\pi}) - \frac{1}{2\sigma^2}(x-h(y,z,0))^2 + \log\pi_{z|y} + 0 \\
E[\log p(x,y,z)] &\approx -\frac{1}{2\sigma^2}(x-h(y,z,0))^2 + \log\pi_{z|y}
\end{aligned}
\tag{5}
$$