1 - Introduction to Generative Models

Given some data \( \{ \mathbf{x}_i \}_{i=1}^n \), we want to estimate \( p(\mathbf{x}) \).

- Generate additional data.
- Better understand the observed data.

“What I cannot create, I do not understand.”

--- Richard Feynman

2 - Latent Variable Models

It may be easier to generate first a latent variable \( z \) and then the data \( x \).

\[
\text{Encoder} \quad q(\mathbf{z}|\mathbf{x}; \phi) \quad \text{Decoder} \quad p(\mathbf{x}|\mathbf{z}; \theta)
\]

The latent variable \( z \):

- Captures high-level information about \( x \).
- Compressed representation of \( x \).

\[
p(\mathbf{x}) = \int p(\mathbf{x}|\mathbf{z}) p(\mathbf{z}) d\mathbf{z}
\]

3 - Variational Autoencoders

Let \( p(\mathbf{x}|\mathbf{z}; \theta) \) be a factorizing Gaussian with parameters given by a MLP and \( \mathbf{z} \sim \mathcal{N}(0, I) \).

\[
p(\mathbf{x}|\mathbf{z}; \theta) = \prod_{d=1}^{D} \mathcal{N}(x_d|\mu_d(\mathbf{z}; \theta), \sigma_d^2(\mathbf{z}; \theta))
\]

We want to find \( \theta \) to maximize \( p(\mathbf{x}_i) \) for \( i = 1, \ldots, N \).

Challenges:

- \( p(\mathbf{x}_i) = \int p(\mathbf{x}_i|\mathbf{z}; \theta) p(\mathbf{z}) d\mathbf{z} \) is intractable.
- \( p(\mathbf{x}_i) = \frac{1}{\mathcal{N}} \sum_{j=1}^{n} p(\mathbf{x}_i|\mathbf{z}_j), \mathbf{z}_j \sim p(\mathbf{z}) \) demands extremely large \( n \).

The Variational Autoencoder (Kingma and Welling, 2014) solves this:

- Samples values of \( z \) that are likely to have produced \( x \).
- Adds a recognition network \( q(\mathbf{z}|\mathbf{x}; \phi) \) that approximates \( p(\mathbf{z}|\mathbf{x}) \).

The VAE is trained to maximize in an approximate way:

\[
\log p(\mathbf{x}_i) \geq \mathbb{E}_{q(\mathbf{z}|\mathbf{x}; \phi)} [p(\mathbf{x}_i|\mathbf{z}; \theta)] - \text{KL}(q(\mathbf{z}|\mathbf{x}_i; \phi)|p(\mathbf{z})) \equiv \mathcal{L}(\mathbf{x}_i; \theta, \phi)
\]

4 - Importance Weighted Autoencoders

Consider a tighter lower bound on \( p(\mathbf{x}_i) \) obtained by importance sampling:

\[
\log p(\mathbf{x}_i) = \log \int p(\mathbf{x}_i|\mathbf{z}; \theta) p(\mathbf{z}) d\mathbf{z} = \log \int p(\mathbf{x}_i|\mathbf{z}; \theta) \frac{q(\mathbf{z}|\mathbf{x}_i; \phi)}{q(\mathbf{z}|\mathbf{x}_i; \phi)} d\mathbf{z}
\]

\[
= \log \mathbb{E}_{q(\mathbf{z}|\mathbf{x}_i; \phi)} \left[ \frac{p(\mathbf{x}_i|\mathbf{z}; \theta) q(\mathbf{z}|\mathbf{x}_i; \phi)}{q(\mathbf{z}|\mathbf{x}_i; \phi)} \right] \approx \log \frac{1}{k} \sum_{m=1}^{k} \frac{p(\mathbf{x}_i|\mathbf{z}^{(m)}; \theta) q(\mathbf{z}^{(m)}|\mathbf{x}_i; \phi)}{q(\mathbf{z}^{(m)}|\mathbf{x}_i; \phi)}
\]

On expectation that estimate is a lower bound on \( p(\mathbf{x}_i) \) (Burda et al., 2016):

\[
\mathcal{L}(\mathbf{x}_i; \theta, \phi) = \mathbb{E} \left[ \log \frac{1}{k} \sum_{m=1}^{k} w_m \right] \leq \log \mathbb{E} \left[ \frac{1}{k} \sum_{m=1}^{k} w_m \right] = \mathcal{L}(\mathbf{x}_i)
\]

If \( k = 1 \) we obtain the VAE, \( k > 1 \) can only improve the bound.

5 - Randomness in the Neural Network Weights

We consider more flexible networks by introducing randomness in \( \theta \) and \( \phi \).

\[
p(\mathbf{x}|\mathbf{z}) = \int p(\mathbf{x}|\mathbf{z}; \theta) q(\theta) d\theta,
q(\mathbf{z}|\mathbf{x}) = \int q(\mathbf{z}|\mathbf{x}; \phi) q(\phi) d\phi.
\]

\( q(\theta) \) and \( q(\phi) \) are Gaussians with parameters \( \Omega = \{ \mu_\theta, \sigma_\theta^2, \mu_\phi, \sigma_\phi^2 \} \).

6 - Experimental Results

- We consider 1-layer MLP with 400 units and 40 latent variables.
- We compare with a model that considers randomness only in \( \phi \).
- We set the number of importance samples \( k = 25 \).

<table>
<thead>
<tr>
<th>Dataset</th>
<th>IWAER</th>
<th>IWAE</th>
<th>IWAE_{re}</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNIST</td>
<td>-95.182 ± 0.022</td>
<td>-94.346 ± 0.025</td>
<td>-94.709 ± 0.025</td>
</tr>
<tr>
<td>Omniglot</td>
<td>-118.771 ± 0.035</td>
<td>-118.540 ± 0.049</td>
<td>-118.647 ± 0.031</td>
</tr>
</tbody>
</table>

7 - Conclusions and Future Work

Conclusions:

1. The VAE and the IWAE are powerful generative models for unsupervised machine learning based on latent variables which can help to better understand the data.
2. The performance of the IWAE can be improved by considering random neural network weights in both the generative and the recognition network.

Future Work:

1. We plan to carry out extra experiments to explore if the gains observed are also obtained in the case of bigger and deeper neural networks.
2. We plan to compare these results with black-box-alpha for training and to potentially explore other models (e.g., ladder variational autoencoders).