Assignment 5

Mahdi Tabatabaei 400101515
Github Repository

Deep Learning

Dr. Fatemizadeh

January 13, 2025

Deep Learning

Assignment 5

Mahdi Tabatabaei 400101515
Github Repository

Question 1 (20 Points)

In this question, we aim to discuss the difference between AE and VAE.

1. Suppose we want to generate data similar to a given dataset. We use a standard AE, select a random point in the latent space (e.g., with a uniform distribution), and input it into the decoder module. In your opinion, is the output of the decoder more likely to resemble the dataset or resemble random noise? Why? (5 points)

   Soloution

   If we use a standard Autoencoder (AE) and select a random point in the latent space (e.g., with a uniform distribution), the output of the decoder is more likely to resemble random noise rather than the dataset. This is because the latent space in a standard AE is not explicitly structured to ensure meaningful interpolations or coherent points outside the regions where the training data resides. As a result, random points in the latent space may not correspond to valid representations of the dataset.

2. Draw at least three problems with the method in part (1) for generating data similar to the dataset. Explain how VAE solves these problems. (5 points)

   Soloution

   (a) **Problem 1: Latent Space Coverage.** In a standard AE, the latent space is not explicitly constrained or organized. Randomly selected points in the latent space may fall in regions that do not correspond to valid representations of the dataset, leading to outputs that resemble noise.

   (b) **Problem 2: Lack of Probabilistic Structure.** AEs do not impose a probabilistic structure on the latent space. This means they lack the ability to estimate the likelihood of points in the latent space corresponding to the training data distribution.

   (c) **Problem 3: Non-smooth Interpolation.** When interpolating between points in the latent space, the generated outputs may not smoothly transition between valid samples because the latent space is not continuous or well-structured.

How VAE Solves These Problems:

(a) VAEs impose a probabilistic structure on the latent space by modeling it as a multivariate Gaussian distribution. This ensures that sampled points are more likely to correspond to valid representations of the dataset.

(b) VAEs encourage smooth and continuous latent spaces by minimizing a reconstruction loss and a KL-divergence term, making interpolation between points in the latent space produce realistic data.

(c) By training the encoder to map inputs to a Gaussian distribution, VAEs ensure that random points sampled from the latent space are more likely to generate data similar to the training dataset.

3. Suppose during the AE training process, we add Gaussian noise with a mean of zero and variance $0.05 \times R$ to the output of the AE, where R represents the mean squared distance of the latent space points from the origin. Does this adjustment improve the decoder's performance in making the output more similar to the dataset? If we randomly select a point in the latent space, which of the outputs is more likely to resemble the dataset? (5 points)

Soloution

Adding Gaussian noise during the AE training process can slightly improve the robustness of the decoder by making it more resilient to small variations in the latent space. However, it does not fundamentally address the lack of structure in the latent space of a standard AE. If we randomly select a point in the latent space after this adjustment, the output is still unlikely to resemble the dataset, as the random points are not guaranteed to correspond to valid representations of the training data.
The approach still lacks the probabilistic structure that is introduced in VAE, which explicitly organizes the latent space to ensure meaningful outputs for sampled points. Thus, while noise may help the decoder generalize better, it does not solve the core issue of ensuring that randomly sampled points correspond to realistic data.

4. Does VAE have an advantage over the method proposed in part (c)? What is the key difference between these two methods? (5 points)

Soloution

Yes, VAE has significant advantages over the method proposed in part (c). The key difference lies in how the latent space is organized and utilized:

(a) **Structured Latent Space:** Unlike the method in part (c), where Gaussian noise is manually added to improve robustness, VAE explicitly models the latent space as a probabilistic distribution (e.g., multivariate Gaussian). This ensures that all points sampled from the latent space are more likely to correspond to valid and meaningful data.

(b) **Regularization via KL-Divergence:** VAE uses a regularization term (KL-divergence) during training to ensure the latent space follows a standard normal distribution. This helps maintain a continuous and smooth latent space, making it more effective for generating realistic data.

(c) **Random Sampling Consistency:** In VAE, random points sampled from the latent space are far more likely to generate outputs similar to the dataset compared to the method in part (c), where the latent space lacks an explicit probabilistic structure.

Overall, VAE not only improves the quality of generated data but also provides a principled framework for sampling from the latent space.

Question 2 (30 Points)

We aim to better understand Maximum Likelihood (ML) estimation and its relationship with VAE.

1. Suppose we have a dataset $D = \{x_1, x_2, \ldots, x_n\}$. Study the concept of maximum likelihood estimation and explain why the distribution parameters should maximize the following relationship:
$$\sum_{i=1}^{n} \log(p_\theta(x_i)).$$
Note that $p_\theta(x_i)$ is the probability of observing x_i given the parameters θ. (5 points)

Soloution

Maximum Likelihood Estimation (MLE) is a fundamental approach in statistics for estimating the parameters of a probabilistic model. The goal of MLE is to find the parameter set θ that maximizes the likelihood of observing the given dataset $D = \{x_1, x_2, \ldots, x_n\}$.
The likelihood function is defined as:
$$L(\theta) = \prod_{i=1}^{n} p_\theta(x_i),$$
where $p_\theta(x_i)$ represents the probability of observing each data point x_i given the parameters θ.
Since the likelihood function involves a product of probabilities, it can result in extremely small values for larger datasets, which can lead to numerical instability. To simplify the optimization and improve numerical stability, we typically work with the log-likelihood function, defined as:
$$\log L(\theta) = \sum_{i=1}^{n} \log(p_\theta(x_i)).$$
Reason for Maximization: Maximizing the log-likelihood is equivalent to maximizing the likelihood itself, as the logarithm is a monotonic function. The log-likelihood measures how well the model parameters θ explain the observed data. By maximizing $\log L(\theta)$, we ensure that the model assigns the highest possible probability to the observed dataset under the chosen parameterization.
In the context of variational autoencoders (VAEs), maximizing the likelihood (or log-likelihood) is an integral part of training the model. VAEs aim to approximate the data-generating distribution and, through optimization, learn parameters θ that make the model-generated distribution as close as possible to the true data distribution.

2. Explain the equivalence between minimizing the cross-entropy loss and Maximum Likelihood (ML) estimation. (5 points)

Soloution

Cross-Entropy Loss: For a classification problem with C classes, the cross-entropy loss is defined as:

$$\mathcal{L}_{\text{CE}} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{c=1}^{C} y_{i,c} \cdot \log(p_{i,c}),$$

where:

- n is the number of samples,
- $y_{i,c}$ is the one-hot encoded label for class c for the i-th sample,
- $p_{i,c}$ is the predicted probability for class c for the i-th sample.

Since $y_{i,c}$ is one-hot encoded, only the log-probability of the true class is considered for each sample:

$$\mathcal{L}_{\text{CE}} = -\frac{1}{n}\sum_{i=1}^{n} \log(p_{i,c^*}),$$

where c^* is the true class for the i-th sample.
Maximum Likelihood (ML) Estimation: Maximum Likelihood Estimation (MLE) aims to find the parameters θ of the model that maximize the likelihood of observing the dataset $D = \{x_1, x_2, \ldots, x_n\}$. The likelihood is given by:

$$L(\theta) = \prod_{i=1}^{n} p_\theta(y_i|x_i),$$

where $p_\theta(y_i|x_i)$ is the predicted probability of the true label y_i given the input x_i. Taking the logarithm to simplify calculations, the log-likelihood becomes:

$$\log L(\theta) = \sum_{i=1}^{n} \log(p_\theta(y_i|x_i)).$$

Equivalence: Minimizing the cross-entropy loss is equivalent to maximizing the log-likelihood (or minimizing the negative log-likelihood):

$$\mathcal{L}_{\text{CE}} = -\frac{1}{n}\sum_{i=1}^{n} \log(p_\theta(y_i|x_i)).$$

Both approaches aim to adjust the model's parameters θ to maximize the probability of the true labels y_i given the inputs x_i. Therefore, minimizing the cross-entropy loss directly aligns with performing Maximum Likelihood Estimation.

3. We know that the ultimate goal of VAE is to create a generative model such that the distribution of its output resembles the dataset. In a regular VAE, we use stochastic gradient descent to maximize the total likelihood $\sum_{i=1}^{n} \log(p_\theta(x_i))$. Instead of directly optimizing this total likelihood, in the learning process, we aim to optimize an Evidence Lower Bound (ELBO) function. The ELBO is given by:

$$\log p_\theta(x_i) \geq \mathbb{E}_z[\log p_\theta(x_i|z)] - D_{\mathrm{KL}}(q_\phi(z|x_i)\|p(z)).$$

Where:

- θ represents the parameters of the decoder,
- ϕ represents the parameters of the encoder,
- D_{KL} is the KL-divergence term, which is non-negative and enforces regularization of the latent distribution.

Answer the following questions based on this relationship:

(a) Prove that the KL-divergence term is non-negative and hence acts as a lower bound constraint for the likelihood. (5 points)

Soloution

The KL-divergence is defined as:

$$D_{\mathrm{KL}}(q_\phi(z|x_i)\|p(z)) = \int q_\phi(z|x_i) \log \frac{q_\phi(z|x_i)}{p(z)}\, dz.$$

Proof of Non-Negativity:
Using the property of logarithms, we have:

$$\log \frac{q_\phi(z|x_i)}{p(z)} = \log q_\phi(z|x_i) - \log p(z).$$

Thus, the KL-divergence can be rewritten as:

$$D_{\mathrm{KL}}(q_\phi(z|x_i)\|p(z)) = \int q_\phi(z|x_i) \log q_\phi(z|x_i)\, dz - \int q_\phi(z|x_i) \log p(z)\, dz.$$

The KL-divergence is always non-negative because of Jensen's inequality, which states that for a convex function f, the following holds:

$$\mathbb{E}[f(X)] \geq f(\mathbb{E}[X]).$$

Applying this to the logarithm, we get:

$$\int q_\phi(z|x_i) \log \frac{q_\phi(z|x_i)}{p(z)}\, dz \geq 0,$$

with equality if and only if $q_\phi(z|x_i) = p(z)$ almost everywhere.
Lower Bound for $\log p_\theta(x_i)$:
Starting with the marginal likelihood:

$$\log p_\theta(x_i) = \log \int p_\theta(x_i|z)p(z)\, dz.$$

Introducing $q_\phi(z|x_i)$, the approximate posterior, we can write:

$$\log p_\theta(x_i) = \log \int q_\phi(z|x_i) \frac{p_\theta(x_i|z)p(z)}{q_\phi(z|x_i)} \, dz.$$

Applying Jensen's inequality to the log function:

$$\log p_\theta(x_i) \geq \int q_\phi(z|x_i) \log \frac{p_\theta(x_i|z)p(z)}{q_\phi(z|x_i)} \, dz.$$

Simplifying:

$$\log p_\theta(x_i) \geq \mathbb{E}_z[\log p_\theta(x_i|z)] - D_{\mathrm{KL}}(q_\phi(z|x_i)\|p(z)).$$

This is the Evidence Lower Bound (ELBO), which serves as a lower bound for $\log p_\theta(x_i)$.

(b) Explain why, in many implementations of VAEs, the term $\mathbb{E}_z[\log p_\theta(x_i|z)]$ is referred to as the cross-entropy loss between the data distribution and the decoder's output distribution. (15 points)

Soloution

The term $\mathbb{E}_z[\log p_\theta(x_i|z)]$ in the Evidence Lower Bound (ELBO) measures the expected log-probability of the data point x_i being reconstructed by the decoder, given a latent variable z sampled from the encoder's posterior $q_\phi(z|x_i)$. This term is often referred to as a cross-entropy loss in the context of VAEs due to the following reasons:

1. Cross-Entropy Definition

In general, cross-entropy is used to compare two probability distributions:

$$\text{Cross-Entropy} = -\sum_x p_{\text{data}}(x) \log(p_{\text{model}}(x)),$$

where:

- $p_{\text{data}}(x)$ is the true data distribution,
- $p_{\text{model}}(x)$ is the model's predicted distribution.

2. Role of $\mathbb{E}_z[\log p_\theta(x_i|z)]$ in VAEs

In VAEs:

- $p_\theta(x|z)$ is the decoder's predicted distribution, which is conditioned on the latent variable z,
- The goal of the decoder is to reconstruct the input data x_i as closely as possible.

For a single input x_i, the expected log-probability term is:

$$\mathbb{E}_z[\log p_\theta(x_i|z)] = \int q_\phi(z|x_i) \log p_\theta(x_i|z)\, dz.$$

This measures how well the decoder $p_\theta(x|z)$ can assign high probability to the true data point x_i, averaged over the latent variable distribution $q_\phi(z|x_i)$. **3. Why It Is Called Cross-Entropy**
- In VAE implementations, $\mathbb{E}_z[\log p_\theta(x_i|z)]$ is minimized as part of the ELBO objective. Minimizing the negative of this term encourages the decoder $p_\theta(x|z)$ to closely match the data distribution.
- This is analogous to the cross-entropy loss, where we minimize the divergence between the true data distribution $p_{\text{data}}(x)$ and the predicted distribution $p_\theta(x|z)$.
Discrete Data: If the data is discrete (e.g., classification tasks), $\log p_\theta(x_i|z)$ directly becomes the log-probability of the correct class. In this case, minimizing $-\mathbb{E}_z[\log p_\theta(x_i|z)]$ corresponds to standard cross-entropy loss.
Continuous Data: For continuous data (e.g., images modeled by a Gaussian distribution), $p_\theta(x|z)$ is often chosen as a Gaussian distribution:

$$p_\theta(x|z) = \mathcal{N}(x; \mu_\theta(z), \sigma_\theta^2),$$

where the mean $\mu_\theta(z)$ is the decoder's output, and σ_θ^2 is a fixed or learned variance. In this case, $-\mathbb{E}_z[\log p_\theta(x_i|z)]$ corresponds to the negative log-likelihood of a Gaussian, which is equivalent to the Mean Squared Error (MSE) for reconstruction, a continuous form of cross-entropy.
4. Relationship Between Reconstruction Loss and Cross-Entropy

- The term $\mathbb{E}_z[\log p_\theta(x_i|z)]$ penalizes the decoder when it assigns low probability to the true data x_i.
- This is conceptually the same as cross-entropy, which penalizes the model when the predicted probability distribution deviates from the true data distribution.
- Minimizing $-\mathbb{E}_z[\log p_\theta(x_i|z)]$ ensures that the decoder produces outputs that are as close as possible to the true data distribution.

Question 3 (20 Points)

Why does VAE assume a Gaussian distribution for the latent space? (In addition to the simplicity of calculations, mention other reasons). Investigate whether, in practice, distributions other than Gaussian are also used.

Soloution

VAE assumes a Gaussian distribution for the latent space for several reasons:

1. **Simplicity of Calculations:** The Gaussian distribution simplifies the mathematical formulation, particularly when using the reparameterization trick. This allows for efficient gradient-based optimization during training and makes the implementation straightforward.
2. **Continuity and Smoothness:** The Gaussian distribution is continuous and smooth, which helps in creating a well-organized latent space. This ensures that nearby points in the latent space correspond to similar outputs in the data space, making the interpolation between points more meaningful.
3. **Regularization with KL-Divergence:** The use of a Gaussian prior allows the KL-divergence term in the VAE loss function to effectively regularize the latent space. This ensures that the latent representations are distributed in a way that balances reconstruction accuracy and smoothness.
4. **Sampling Consistency:** With a Gaussian distribution, sampling random points in the latent space is more likely to generate valid outputs. This is crucial for generating realistic data during inference.

Other Distributions: Although Gaussian priors are commonly used in VAEs, other distributions can be employed depending on the dataset and task requirements:

- **Gaussian Mixture Models (GMM):** In some cases, using a mixture of Gaussians can better capture the multi-modal nature of the data, providing a more flexible latent space representation.
- **Uniform Distribution:** While less common, uniform priors can be used in certain cases where a simpler latent space structure is sufficient.
- **Non-Gaussian Priors:** More complex priors, such as VampPrior (variational mixture of posteriors), can be used to adapt the latent space to better fit the data without strictly adhering to Gaussian assumptions.

These alternative distributions are chosen based on the specific requirements of the problem, such as the complexity and diversity of the dataset, and may improve the expressiveness and flexibility of the latent space.

Question 4 (20 Points)

Studty $\beta - VAE$ and answer following questions:

1. Explain the idea of β-VAE briefly and describe its differences with VAE. (15 points)

Soloution

β-VAE is an extension of the Variational Autoencoder (VAE) framework introduced to learn disentangled representations of independent generative factors in data. The main idea is to introduce a hyperparameter β in the VAE objective function that controls the trade-off between reconstruction accuracy and the disentanglement of latent representations.
The β-VAE objective function is given by:

$$L(,;x,z,\beta) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - \beta\, \mathrm{D}_{\mathrm{KL}}(q_\phi(z|x)\|p(z)).$$

Differences between β-VAE and VAE:

- **Latent Space Constraints:** In β-VAE, $\beta > 1$ increases the weight of the KL-divergence term, imposing stronger constraints on the latent space and encouraging the learning of disentangled representations. In contrast, VAE corresponds to $\beta = 1$, which focuses on accurate reconstruction but may lead to entangled representations.
- **Disentanglement:** By prioritizing the independence of latent variables, β-VAE learns a representation where each latent variable encodes a distinct generative factor. VAE, on the other hand, often produces entangled representations where multiple factors are mixed in a single latent variable.
- **Trade-off:** β-VAE achieves better disentanglement at the cost of reconstruction fidelity. As β increases, the reconstructions may lose fine details but gain interpretability through disentangled latent variables.

β-VAE significantly outperforms VAE in discovering interpretable factors of variation in data, enabling its use in tasks requiring disentangled representations, such as transfer learning and novelty detection.

2. Based on the information provided in Section 2 of the paper, explain the importance and functionality of the disentanglement metric. (15 points)

Soloution

The disentanglement metric introduced in Section 2 of the paper quantifies the degree of disentanglement achieved by a model. This is critical for comparing different models and tuning hyperparameters such as β in β-VAE.

Key Functions and Importance:

- **Definition of Disentanglement:** A disentangled representation ensures that each latent variable is sensitive to changes in a single generative factor while being invariant to changes in others. The disentanglement metric evaluates how well a model achieves this property.
- **Quantitative Evaluation:** The metric involves using a linear classifier to predict a specific generative factor (e.g., position, scale, or rotation) from the latent representations. Higher classifier accuracy indicates better disentanglement.
- **Independence and Interpretability:** The metric captures both the independence and interpretability of latent variables, which are essential for applications such as transfer learning and zero-shot inference.
- **Practical Use:** It allows researchers to evaluate different models (e.g., VAE, β-VAE, InfoGAN) quantitatively and choose the best one for a specific task. Additionally, it provides insights into the effect of varying β on the quality of disentangled representations.

By ensuring independence and interpretability in latent representations, the disentanglement metric plays a pivotal role in advancing unsupervised learning techniques like β-VAE and their applications in real-world problems.