114-1 Machine Learning

Week 7 Assignment

Ming Hsun Wu

October 21, 2025

Problem 1.

Explain the concept of score matching and describe how it is used in score-based (diffusion) generative models.

Note of Problem 1.

1 What is Score Matching

In probabilistic modeling, we often want a model $p_\theta(x)$ that captures the true data distribution $p_{\text{data}}(x)$. A common approach is Maximum Likelihood Estimation (MLE), but for **Energy-Based Models (EBMs)**:

$$p_\theta(x) = \frac{1}{Z(\theta)} \exp(-E_\theta(x)),$$

the normalization constant $Z(\theta) = \int \exp(-E_\theta(x))dx$ is usually hard to compute, especially in high dimensions. To get around this, **Score Matching** provides a clever alternative—instead of maximizing likelihood directly, we train the model to match the **gradient of the log-probability**, called the *score function*:

$$S(x; \theta) = \nabla_x \log p_\theta(x)$$

This function points toward regions where the data density increases. Intuitively, if the model learns the same "direction of probability flow" as the real data, it captures the distribution shape correctly, even without knowing $Z(\theta)$.

2 Three Variants of Score Matching

2.1 Explicit Score Matching (ESM)

The original form tries to minimize the difference between the model's and data's scores:

$$L_{\mathrm{ESM}}(\theta) = E_{x\sim p_{\mathrm{data}}}\|S(x;\theta) - \nabla_x \log p_{\mathrm{data}}(x)\|^2$$

However, since $\nabla_x \log p_{\mathrm{data}}(x)$ is unknown (we don't have an analytic expression for it), this version can't be used directly in practice.

2.2 Implicit Score Matching (ISM)

Hyvärinen (2005) proposed using **integration by parts** to remove the unknown data term, leading to a tractable loss:

$$L_{\mathrm{ISM}}(\theta) = E_{x\sim p_{\mathrm{data}}}[\|S(x;\theta)\|^2 + 2\nabla_x \cdot S(x;\theta)]$$

This way, we avoid the need for the true score. When I first saw this, I thought it was just a trick, but it's actually elegant—turning a difficult integral into a differentiable quantity.

2.3 Denoising Score Matching (DSM)

Computing the divergence term in ISM can still be unstable in high dimensions. Vincent (2011) proposed a practical variant called **Denoising Score Matching**.

We add Gaussian noise to clean samples x_0:

$$x = x_0 + \sigma\epsilon, \quad \epsilon \sim \mathcal{N}(0, I)$$

Then train the model to predict the direction that removes the noise:

$$L_{\text{DSM}}(\theta) = E_{x_0,\epsilon}\|S_\sigma(x_0 + \sigma\epsilon; \theta) + \frac{\epsilon}{\sigma}\|^2$$

Intuitively, the model learns how to "denoise" corrupted data. Over time, this denoising process approximates the true score field.

3 How It Connects to Diffusion Models

Score-based (diffusion) models build directly on DSM. The idea is to gradually **add noise** to data (forward process), train a model to **reverse** it (learning the score), and then **sample** new data by running the process backward.

3.1 Forward Process

We continuously perturb data:

$$dx = g(t)dw_t$$

At time t, the noisy sample follows $p_t(x)$.

3.2 Training Phase

We train a neural network $s_\theta(x_t, t)$ to approximate the score:

$$s_\theta(x_t, t) \approx \nabla_x \log p_t(x_t)$$

This is done across different noise levels.

3.3 Sampling Phase

Once trained, we generate data by reversing the diffusion process using either a **stochastic differential equation (SDE)**:

$$dx = [f(x,t) - g(t)^2 s_\theta(x,t)]dt + g(t)d\bar{w}_t$$

or a **deterministic ODE** version:

$$\frac{dx}{dt} = -\frac{1}{2}g(t)^2 s_\theta(x,t)$$

Starting from pure noise and integrating backward yields realistic samples.

4 Summary and My Takeaways

1. The **score function** tells us the direction of increasing data probability—like a vector field guiding samples toward data regions.

2. **Integration by parts** in ISM removes hard-to-compute terms like $Z(\theta)$, which makes score matching practical.

3. **Denoising** turns a tricky theoretical idea into a trainable process that's now the foundation of modern diffusion models.