Deep Generative Models

Fall 2025

Sharif University of Technology

Alireza Mirrokni - 401106617

Problem Set 5

Problem 1: Ergodicity Conditions in MCMC

(a) 1.

- **Irreducibility:** Irreducibility ensures that the Markov chain can visit any state $y \in \mathcal{X}$ from any starting state $x \in \mathcal{X}$ in a finite number of steps (i.e., $P^n(x, y) > 0$ for some n).
  Importance: This is crucial for MCMC because we need the sampler to be able to explore the entire state space of the target distribution π. If the chain were not irreducible, the sampler might get stuck in a subset of the state space depending on the initialization, preventing it from converging to the true global target distribution.
  Failure Example: Consider a state space $\mathcal{X} = \{A, B\}$ where $P(A, A) = 1$ and $P(B, B) = 1$. If we start at A, we never sample B. The chain is reducible and fails to sample from any π that assigns non-zero probability to both A and B.
- **Aperiodicity:** Aperiodicity ensures that the chain does not get trapped in a deterministic cycle of states. Formally, it requires that the greatest common divisor of the return times to any state is 1.
  Importance: Aperiodicity is necessary to guarantee that the chain's distribution $P^n(x, \cdot)$ actually converges to the stationary distribution π as $n \to \infty$. Without this condition, the chain may be trapped in a deterministic cycle of states, causing the distribution to oscillate indefinitely instead of stabilizing.
  Failure Example: Consider a chain on $\mathcal{X} = \{A, B\}$ where $P(A, B) = 1$ and $P(B, A) = 1$. The chain flips $A \to B \to A \dots$ deterministically. The distribution at step n depends entirely on the parity of n and never settles into a stationary distribution $\pi = [0.5, 0.5]$.
- **Positive Recurrence:** Positive recurrence ensures that if the chain leaves a state x, it will return to x in a finite expected time ($\mathbb{E}_x[T_x] < \infty$).
  Importance: This condition guarantees the existence of a unique stationary distribution. While always true for finite irreducible chains, it is critical for infinite state spaces (e.g., $\mathbb{Z}$). If a chain is null recurrent or transient, the probability mass escapes to infinity, and no proper distribution can be stationary.
  Failure Example: Simple Random Walk on $\mathbb{Z}$ (symmetric). It is null recurrent. Although it returns to 0 with probability 1, the expected return time is infinite. There is no probability vector π such that $\pi P = \pi$; the mass spreads out infinitely, approaching a uniform measure of 0 everywhere.

2. According to the standard ergodic theorem, if a Discrete Time Markov Chain (DTMC) is irreducible, aperiodic, and positive recurrent, two key results follow that are essential for MCMC:

- **Existence and Uniqueness:** The combination of irreducibility and positive recurrence guarantees the existence of a unique probability distribution π that satisfies the stationary equation $\pi P = \pi$ subject to $\sum_x \pi(x) = 1$. This ensures that the MCMC algorithm targets a single, well-defined distribution regardless of the initialization, rather than having multiple stationary distributions (reducible) or no stationary distribution at all (null recurrent/transient).
- **Convergence:** Aperiodicity is the critical condition that ensures the distribution of the chain converges to π in the limit. Specifically, it guarantees that for any starting state x, $\lim_{n\to\infty} ||P^n(x,\cdot) - \pi(\cdot)||_{TV} = 0$, where $||\cdot||_{TV}$ is the total variation distance. Without aperiodicity, the limit may not exist due to oscillation. But with it, we are mathematically justified in treating samples after a burn-in period as coming from π.

(b) 1. The detailed balance condition is given by $\pi(x)P(x,y) = \pi(y)P(y,x)$. Summing both sides over x:

$$\sum_{x\in\mathcal{X}} \pi(x)P(x,y) = \sum_{x\in\mathcal{X}} \pi(y)P(y,x) = \pi(y)\sum_{x\in\mathcal{X}} P(y,x)$$

Since P is a stochastic transition kernel, $\sum_x P(y,x) = 1$. Thus:

$$\sum_{x\in\mathcal{X}} \pi(x)P(x,y) = \pi(y)$$

This is precisely the definition of stationarity ($\pi P = \pi$). Therefore, detailed balance guarantees that π is a stationary distribution for the kernel P.

2. The Metropolis-Hastings transition kernel $P_{MH}(x,y)$ for $x \neq y$ is defined as $q(x,y)\alpha(x,y)$, where $\alpha(x,y) = \min\left(1, \frac{\pi(y)q(y,x)}{\pi(x)q(x,y)}\right)$. We verify detailed balance:

$$\pi(x)P_{MH}(x,y) = \pi(x)q(x,y)\min\left(1, \frac{\pi(y)q(y,x)}{\pi(x)q(x,y)}\right) = \min(\pi(x)q(x,y), \pi(y)q(y,x))$$

By symmetry of the min function:

$$\pi(y)P_{MH}(y,x) = \pi(y)q(y,x)\min\left(1, \frac{\pi(x)q(x,y)}{\pi(y)q(y,x)}\right) = \min(\pi(y)q(y,x), \pi(x)q(x,y))$$

Since $\pi(x)P_{MH}(x,y) = \pi(y)P_{MH}(y,x)$, the MH algorithm satisfies detailed balance with respect to π. As shown in the previous step, this guarantees that π is the stationary distribution of the constructed chain.

Problem 2: Score-Matching Variants

1. (a) We treat x_{-i} as fixed and consider the integration with respect to x_i. By the Fundamental Theorem of Calculus, for the scalar function $h(x_i) = f(x)g(x)$:

$$\int_{-\infty}^{\infty} \frac{\partial}{\partial x_i} h(x_i)dx_i = \lim_{a\to\infty, b\to-\infty} [h(x_i)]_{x_i=b}^{x_i=a}$$

Therefore:

$$\int_{-\infty}^{\infty} \frac{\partial}{\partial x_i}(f(x)g(x))dx_i = \lim_{a\to\infty, b\to-\infty} [f(x)g(x)]_{x_i=b}^{x_i=a}$$
$$= \lim_{a\to\infty, b\to-\infty} (f(x_{-i}, a)g(x_{-i}, a) - f(x_{-i}, b)g(x_{-i}, b))$$

Using the product rule for differentiation,

$$\frac{\partial}{\partial x_i}(f(x)g(x)) = f(x)\frac{\partial g(x)}{\partial x_i} + g(x)\frac{\partial f(x)}{\partial x_i}$$

. Substituting this into the integral on the LHS:

$$\int_{-\infty}^{\infty} \left(f(x)\frac{\partial g(x)}{\partial x_i} + g(x)\frac{\partial f(x)}{\partial x_i} \right) dx_i = \int_{-\infty}^{\infty} f(x)\frac{\partial g(x)}{\partial x_i}dx_i + \int_{-\infty}^{\infty} g(x)\frac{\partial f(x)}{\partial x_i}dx_i$$

Equating the two expressions establishes the identity:

$$\lim_{a\to\infty, b\to-\infty} (f(x_{-i}, a)g(x_{-i}, a) - f(x_{-i}, b)g(x_{-i}, b))$$
$$= \int_{-\infty}^{\infty} f(x)\frac{\partial g(x)}{\partial x_i}dx_i + \int_{-\infty}^{\infty} g(x)\frac{\partial f(x)}{\partial x_i}dx_i$$

(b) We expand the squared norm in the definition of $L_{\text{ESM}}(\theta)$:

$$L_{\text{ESM}}(\theta) = \frac{1}{2}\mathbb{E}_{x\sim p_{\text{data}}}[\|s_\theta(x) - s_{\text{data}}(x)\|_2^2]$$
$$= \frac{1}{2}\mathbb{E}_{x\sim p_{\text{data}}}[\|s_\theta(x)\|_2^2 - 2s_\theta(x)^\top s_{\text{data}}(x) + \|s_{\text{data}}(x)\|_2^2]$$
$$= \mathbb{E}_{x\sim p_{\text{data}}}\left[\frac{1}{2}\|s_\theta(x)\|_2^2\right] - \mathbb{E}_{x\sim p_{\text{data}}}[s_\theta(x)^\top s_{\text{data}}(x)] + \underbrace{\frac{1}{2}\mathbb{E}_{x\sim p_{\text{data}}}[\|s_{\text{data}}(x)\|_2^2]}_{C}$$

The term C is independent of θ. We analyze the cross term $A = -\mathbb{E}_{x\sim p_{\text{data}}}[s_\theta(x)^\top s_{\text{data}}(x)]$. By definition, $s_{\text{data}}(x) = \nabla_x \log p_{\text{data}}(x) = \frac{\nabla_x p_{\text{data}}(x)}{p_{\text{data}}(x)}$.

$$A = -\int p_{\text{data}}(x) \sum_{i=1}^{d} s_\theta(x)_i \frac{\partial \log p_{\text{data}}(x)}{\partial x_i}dx = -\sum_{i=1}^{d} \int s_\theta(x)_i \frac{\partial p_{\text{data}}(x)}{\partial x_i}dx$$

We apply the identity from part (a) to each integral component, setting $f(x) = p_{\text{data}}(x)$ and $g(x) = s_\theta(x)_i$. The boundary term (LHS of the identity) vanishes because $p_{\text{data}}(x)s_\theta(x) \to 0$ as $\|x\|_p \to \infty$ by assumption.

$$0 = \int_{-\infty}^{\infty} p_{\text{data}}(x)\frac{\partial s_\theta(x)_i}{\partial x_i}dx_i + \int_{-\infty}^{\infty} s_\theta(x)_i \frac{\partial p_{\text{data}}(x)}{\partial x_i}dx_i$$
$$\implies \int_{-\infty}^{\infty} s_\theta(x)_i \frac{\partial p_{\text{data}}(x)}{\partial x_i}dx_i = -\int_{-\infty}^{\infty} p_{\text{data}}(x)\frac{\partial s_\theta(x)_i}{\partial x_i}dx_i$$

Substituting this back into the expression for A:

$$A = -\sum_{i=1}^{d} \left(-\int p_{\text{data}}(x)\frac{\partial s_\theta(x)_i}{\partial x_i}dx \right) = \sum_{i=1}^{d} \mathbb{E}_{x\sim p_{\text{data}}}\left[\frac{\partial s_\theta(x)_i}{\partial x_i}\right] = \mathbb{E}_{x\sim p_{\text{data}}}[\text{tr}(\nabla_x s_\theta(x))]$$

Finally, substituting A back into the expanded objective:

$$L_{\text{ESM}}(\theta) = \mathbb{E}_{x\sim p_{\text{data}}}\left[\frac{1}{2}\|s_\theta(x)\|_2^2 + \text{tr}(\nabla_x s_\theta(x))\right] + C = L_{\text{ISM}}(\theta) + C$$

2. (a) We aim to prove the identity

$$S(\theta) = \mathbb{E}_{\tilde{x}\sim q_\sigma}[\langle s_\theta(\tilde{x}), \nabla_{\tilde{x}} \log q_\sigma(\tilde{x})\rangle] = \mathbb{E}_{(x,\tilde{x})\sim q_\sigma(x,\tilde{x})}[\langle s_\theta(\tilde{x}), \nabla_{\tilde{x}} \log q_\sigma(\tilde{x} \mid x)\rangle]$$

Let us start with the Right Hand Side (RHS). By definition of the expectation over the joint distribution $q_\sigma(x, \tilde{x}) = p_{\text{data}}(x)q_\sigma(\tilde{x} \mid x)$:

$$\text{RHS} = \iint p_{\text{data}}(x)q_\sigma(\tilde{x} \mid x)\langle s_\theta(\tilde{x}), \nabla_{\tilde{x}} \log q_\sigma(\tilde{x} \mid x)\rangle \, dx \, d\tilde{x}$$

Using the identity $\nabla \log f = \frac{\nabla f}{f}$, we substitute $\nabla_{\tilde{x}} \log q_\sigma(\tilde{x} \mid x) = \frac{\nabla_{\tilde{x}} q_\sigma(\tilde{x}|x)}{q_\sigma(\tilde{x}|x)}$:

$$\text{RHS} = \iint p_{\text{data}}(x)q_\sigma(\tilde{x} \mid x)\left\langle s_\theta(\tilde{x}), \frac{\nabla_{\tilde{x}} q_\sigma(\tilde{x} \mid x)}{q_\sigma(\tilde{x} \mid x)}\right\rangle dx \, d\tilde{x}$$

The term $q_\sigma(\tilde{x} \mid x)$ cancels out:

$$\text{RHS} = \int \left\langle s_\theta(\tilde{x}), \int p_{\text{data}}(x)\nabla_{\tilde{x}} q_\sigma(\tilde{x} \mid x)\, dx\right\rangle d\tilde{x}$$

Assuming regularity conditions allow exchanging the derivative and integral:

$$\int p_{\text{data}}(x)\nabla_{\tilde{x}} q_\sigma(\tilde{x} \mid x)\, dx = \nabla_{\tilde{x}} \int p_{\text{data}}(x)q_\sigma(\tilde{x} \mid x)\, dx = \nabla_{\tilde{x}} q_\sigma(\tilde{x})$$

Substituting this back into the expression for RHS:

$$\text{RHS} = \int \langle s_\theta(\tilde{x}), \nabla_{\tilde{x}} q_\sigma(\tilde{x})\rangle \, d\tilde{x}$$

Now, we multiply and divide by $q_\sigma(\tilde{x})$ (where $q_\sigma(\tilde{x}) > 0$) and use $\frac{\nabla q}{q} = \nabla \log q$:

$$\begin{aligned}\text{RHS} = \int q_\sigma(\tilde{x})\left\langle s_\theta(\tilde{x}), \frac{\nabla_{\tilde{x}} q_\sigma(\tilde{x})}{q_\sigma(\tilde{x})}\right\rangle d\tilde{x} &= \int q_\sigma(\tilde{x})\langle s_\theta(\tilde{x}), \nabla_{\tilde{x}} \log q_\sigma(\tilde{x})\rangle \, d\tilde{x} \\ &= \mathbb{E}_{\tilde{x}\sim q_\sigma}[\langle s_\theta(\tilde{x}), \nabla_{\tilde{x}} \log q_\sigma(\tilde{x})\rangle] = \text{LHS}\end{aligned}$$

Thus, $S(\theta)$ is well-defined by either expression.

(b) We expand the squared norm in the definitions of both objectives.
First, for $L_{\text{ESM},q_\sigma}(\theta)$:

$$L_{\text{ESM},q_\sigma}(\theta) = \mathbb{E}_{\tilde{x}\sim q_\sigma}\left[\frac{1}{2}\|s_\theta(\tilde{x})\|^2 - \langle s_\theta(\tilde{x}), \nabla_{\tilde{x}} \log q_\sigma(\tilde{x})\rangle + \frac{1}{2}\|\nabla_{\tilde{x}} \log q_\sigma(\tilde{x})\|^2\right]$$

We can separate this into terms dependent on θ and constant terms:

$$L_{\text{ESM},q_\sigma}(\theta) = \frac{1}{2}\mathbb{E}_{\tilde{x}\sim q_\sigma}[\|s_\theta(\tilde{x})\|^2] - \underbrace{\mathbb{E}_{\tilde{x}\sim q_\sigma}[\langle s_\theta(\tilde{x}), \nabla_{\tilde{x}} \log q_\sigma(\tilde{x})\rangle]}_{S(\theta)} + C_1$$

where $C_1 = \frac{1}{2}\mathbb{E}_{\tilde{x}}[\|\nabla \log q_\sigma(\tilde{x})\|^2]$ is independent of θ. Next, for $L_{\mathrm{DSM},q_\sigma}(\theta)$:

$$L_{\mathrm{DSM},q_\sigma}(\theta)$$
$$= \mathbb{E}_{(x,\tilde{x})\sim q_\sigma(x,\tilde{x})}\left[\frac{1}{2}\|s_\theta(\tilde{x})\|^2 - \langle s_\theta(\tilde{x}), \nabla_{\tilde{x}} \log q_\sigma(\tilde{x} \mid x)\rangle + \frac{1}{2}\|\nabla_{\tilde{x}} \log q_\sigma(\tilde{x} \mid x)\|^2\right]$$

Using linearity of expectation:

- The first term is $\frac{1}{2}\mathbb{E}_{(x,\tilde{x})}[\|s_\theta(\tilde{x})\|^2]$. Since the integrand depends only on $\tilde{x}$, marginalizing over x yields $\frac{1}{2}\mathbb{E}_{\tilde{x}\sim q_\sigma}[\|s_\theta(\tilde{x})\|^2]$.
- The second term is $-\mathbb{E}_{(x,\tilde{x})}[\langle s_\theta(\tilde{x}), \nabla_{\tilde{x}} \log q_\sigma(\tilde{x} \mid x)\rangle]$. From Part (a), this is exactly equal to $-S(\theta)$.
- The third term $C_2 = \frac{1}{2}\mathbb{E}_{(x,\tilde{x})}[\|\nabla_{\tilde{x}} \log q_\sigma(\tilde{x} \mid x)\|^2]$ is independent of θ.

Substituting these back:

$$L_{\mathrm{DSM},q_\sigma}(\theta) = \frac{1}{2}\mathbb{E}_{\tilde{x}\sim q_\sigma}[\|s_\theta(\tilde{x})\|^2] - S(\theta) + C_2$$

Comparing the expanded forms:

$$L_{\mathrm{ESM},q_\sigma}(\theta) - L_{\mathrm{DSM},q_\sigma}(\theta) = C_1 - C_2 = \text{const}$$

Therefore, $L_{\mathrm{ESM},q_\sigma}(\theta) = L_{\mathrm{DSM},q_\sigma}(\theta) + \text{const}$.

3. (a) We analyze the term $\mathbb{E}_x[(v^\top s_\theta(x))(v^\top s_{data}(x))]$. Let $u = v^\top s_\theta(x)$. Then the integrand is $u\sum_i v_i \frac{\partial \log p_{data}}{\partial x_i}$.

$$\mathbb{E}_x[u(v^\top \nabla_x \log p_{data}(x))] = \int p_{data}(x) u (v^\top \nabla_x \log p_{data}(x)) dx$$
$$= \int u \sum_i v_i \frac{\partial p_{data}}{\partial x_i} dx = \sum_i v_i \int u \frac{\partial p_{data}}{\partial x_i} dx$$

Using integration by parts (assuming boundary terms vanish):

$$\int u \frac{\partial p_{data}}{\partial x_i} dx = -\int p_{data} \frac{\partial u}{\partial x_i} dx$$

Substitute $u = \sum_j v_j s_{\theta,j}(x)$:

$$\frac{\partial u}{\partial x_i} = \sum_j v_j \frac{\partial s_{\theta,j}}{\partial x_i} = v^\top \frac{\partial s_\theta}{\partial x_i}$$

Thus:

$$\sum_i v_i \left(-\mathbb{E}_x\left[v^\top \frac{\partial s_\theta}{\partial x_i}\right]\right) = -\mathbb{E}_x\left[\sum_i v_i v^\top \frac{\partial s_\theta}{\partial x_i}\right] = -\mathbb{E}_x[v^\top (\nabla_x s_\theta) v]$$

Therefore:

$$\mathbb{E}_x[(v^\top s_\theta)(v^\top s_{data})] = -\mathbb{E}_x[v^\top (\nabla_x s_\theta) v]$$

Multiplying by -1 gives the requested identity.

(b) Expand the square in L_{ESSM}:

$$(v^\top s_\theta - v^\top s_{data})^2 = (v^\top s_\theta)^2 - 2(v^\top s_\theta)(v^\top s_{data}) + (v^\top s_{data})^2$$

Taking expectations $\mathbb{E}_v\mathbb{E}_x$:

$$L_{ESSM}(\theta) = \mathbb{E}_{v,x}\left[\frac{1}{2}(v^\top s_\theta)^2 - (v^\top s_\theta)(v^\top s_{data})\right] + C$$

Using the identity from Part 3a:

$$-\mathbb{E}_{v,x}[(v^\top s_\theta)(v^\top s_{data})] = \mathbb{E}_{v,x}[v^\top(\nabla_x s_\theta)v]$$

Substituting this back:

$$L_{ESSM}(\theta) = \mathbb{E}_{v,x}\left[\frac{1}{2}(v^\top s_\theta)^2 + v^\top(\nabla_x s_\theta)v\right] + C = L_{ISSM}(\theta) + C$$

(c) The objective is $L(\theta) = \frac{1}{2}\mathbb{E}_v\mathbb{E}_x[(v^\top(s_\theta(x) - s_{data}(x)))^2]$. Since the integrand is non-negative,

$$L(\theta) = 0 \implies (v^\top(s_\theta(x) - s_{data}(x)))^2 = 0$$

almost surely w.r.t p_v and p_{data}. Let $\Delta s(x) = s_\theta(x) - s_{data}(x)$. The condition is $v^\top \Delta s(x) = 0$. Thus

$$\mathbb{E}_v[(v^\top \Delta s(x))^2] = \mathbb{E}_v[\Delta s(x)^\top v v^\top \Delta s(x)] = \Delta s(x)^\top \mathbb{E}_v[vv^\top]\Delta s(x) = 0$$

Given $\Sigma_v = \mathbb{E}[vv^\top] > 0$ (positive definite), $\Delta s(x)^\top \Sigma_v \Delta s(x) = 0$ implies $\Delta s(x) = 0$ for almost all x. $s_\theta(x) = s_{data}(x)$ implies

$$\begin{aligned}\nabla \log p_\theta(x) = \nabla \log p_{data}(x) &\implies \int \nabla \log p_\theta(x)dx = \int \nabla \log p_{data}(x)dx \\ &\implies \log p_\theta(x) = \log p_{data}(x) + c\end{aligned}$$

Since both are normalized probability densities, $c = 0$, so $p_\theta = p_{data}$. By the identifiability assumption, $p_\theta = p_{data} \implies \theta = \theta^*$.

(d) (i) We have $L_{ESSM}(\theta) = \frac{1}{2}\mathbb{E}_x\mathbb{E}_v[\Delta s(x)^\top v v^\top \Delta s(x)]$. Since $\Delta s(x)$ is constant w.r.t v:

$$L_{ESSM}(\theta) = \frac{1}{2}\mathbb{E}_x[\Delta s(x)^\top \mathbb{E}_v[vv^\top]\Delta s(x)]$$

If $\mathbb{E}[vv^\top] = I_D$:

$$L_{ESSM}(\theta) = \frac{1}{2}\mathbb{E}_x[\Delta s(x)^\top I_D \Delta s(x)] = \frac{1}{2}\mathbb{E}_x[||\Delta s(x)||_2^2] = L_{ESM}(\theta)$$

(ii) **Rademacher:** $v_i \in \{-1, 1\}$ with prob 0.5. $v_i \in \{-1, 1\}$ with prob 0.5.

$$\mathbb{E}[v_i^2] = (-1)^2(0.5) + (1)^2(0.5) = 1$$

For $i \neq j$, independence implies

$$\mathbb{E}[v_i v_j] = \mathbb{E}[v_i]\mathbb{E}[v_j] = 0 \cdot 0 = 0$$

Thus $\mathbb{E}[vv^\top] = I_D$.

Standard Normal: $v \sim \mathcal{N}(0, I_D)$. The probability density function is given by:

$$p(v) = \frac{1}{(2\pi)^{D/2}} \exp\left(-\frac{1}{2} v^\top v\right)$$

The expectation $\mathbb{E}[vv^\top]$ is defined by the integral:

$$\mathbb{E}[vv^\top] = \int_{\mathbb{R}^D} vv^\top p(v)\, dv$$

Considering an arbitrary element (i, j) of the matrix $vv^\top$:

$$\mathbb{E}[v_i v_j] = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} v_i v_j \prod_{k=1}^{D} \left(\frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}v_k^2}\right) dv_1 \dots dv_D$$

Case $i \neq j$: We can separate the integrals because the density factorizes:

$$\mathbb{E}[v_i v_j] = \left(\int_{-\infty}^{\infty} v_i \frac{e^{-v_i^2/2}}{\sqrt{2\pi}} dv_i\right) \left(\int_{-\infty}^{\infty} v_j \frac{e^{-v_j^2/2}}{\sqrt{2\pi}} dv_j\right) \prod_{k \neq i,j} \left(\int_{-\infty}^{\infty} \frac{e^{-v_k^2/2}}{\sqrt{2\pi}} dv_k\right)$$

Since v_i is an odd function and the Gaussian density is even, the integral $\int_{-\infty}^{\infty} x e^{-x^2/2} dx = 0$. Thus, the entire product is 0.
Case $i = j$:

$$\mathbb{E}[v_i^2] = \left(\int_{-\infty}^{\infty} v_i^2 \frac{e^{-v_i^2/2}}{\sqrt{2\pi}} dv_i\right) \prod_{k \neq i} \underbrace{\left(\int_{-\infty}^{\infty} \frac{e^{-v_k^2/2}}{\sqrt{2\pi}} dv_k\right)}_{1}$$

The remaining integral is the variance of a standard scalar Gaussian, which is known to be 1. Thus, $\mathbb{E}[vv^\top]_{ij} = \delta_{ij}$, which implies $\mathbb{E}[vv^\top] = I_D$.

Problem 3: Score-based Models

1. **Langevin Dynamics Problems:** In practice, Langevin dynamics encounters problems because the estimated score function $s_\theta(x)$ is often inaccurate in low-density regions. According to the manifold hypothesis, data resides on a low-dimensional manifold; in the vast ambient space far from this manifold, there is no training data, so the score estimate is undefined or random. Consequently, if the sampling chain starts in or traverses these regions, the gradients will not effectively guide the sample towards the data distribution, leading to poor mixing and invalid samples.

   Solution: The standard solution is **Noise Conditional Score Networks (NCSN)** or **SDE-based diffusion**. By perturbing the data with various levels of Gaussian noise, we populate the low-density regions with noisy data support. We train a score network conditioned on the noise level σ. During sampling (Annealed Langevin Dynamics), we start with high noise (where the score is well-defined everywhere) and gradually reduce σ, effectively guiding the sample from the ambient space onto the data manifold.

2. To generate samples from a class y (conditional distribution $p(x|y)$), we can use Bayes' rule: $p(x|y) = \frac{p(x)p(y|x)}{p(y)}$. The score of the conditional distribution is:

$$\nabla_x \log p(x|y) = \nabla_x \log p(x) + \nabla_x \log p(y|x) - \nabla_x \log p(y)$$

Since $\nabla_x \log p(y) = 0$, we have:

$$s(x|y) = s_\theta(x) + \nabla_x \log p(y|x)$$

where $s_\theta(x)$ is the pretrained unconditional score model and $p(y|x)$ is a classifier. We run Langevin dynamics using this modified score. The update step (with step size ϵ) becomes:

$$x_{t+1} = x_t + \frac{\epsilon}{2}\left(s_\theta(x_t) + \nabla_x \log p_{classifier}(y|x_t)\right) + \sqrt{\epsilon} z_t, \quad z_t \sim \mathcal{N}(0, I)$$

3. We cannot use a standard frozen pretrained classifier because the score-based generation process (especially in SDE or NCSN variants) involves sampling x at various noise levels (from pure noise to clean data). A standard classifier is trained only on clean data and will fail to provide meaningful gradients $\nabla_x \log p(y|x)$ when x is a noisy intermediate image. Therefore, the classifier must be explicitly trained (or fine-tuned) on noisy data with the same noise schedule as the generative model, ensuring it provides accurate guidance signal throughout the entire reverse diffusion process.

4. The objective is $J(\theta) = \mathbb{E}_{p_{data}}[\frac{1}{2}||s_\theta(x)||_2^2 + \text{tr}(\nabla_x s_\theta(x))]$.

   - **First term ($\frac{1}{2}||s_\theta(x)||^2$):** This term penalizes large score vectors. Without the second term, the optimal solution would be $s_\theta(x) = 0$ everywhere. Intuitively, this term tries to make the density flat (uniform), as the score (gradient of log-density) is zero for a uniform distribution.
   - **Second term ($\text{tr}(\nabla_x s_\theta(x))$):** This is the divergence of the score field (or the Laplacian of the log-density). $\text{tr}(\nabla s) = \sum \partial^2 \log p / \partial x_i^2$. At the mode (peak) of a distribution, the curvature is negative (concave). Minimizing this term encourages the trace to be negative, which corresponds to creating peaks (high probability mass) at the data points.

   Together, the objective balances these forces: minimizing the squared norm encourages smoothness, while minimizing the trace forces the function to curve downwards (accumulate mass) at the observed data points x, resulting in a normalized probability density.

Problem 4: Energy-based Models

1. **Contrastive Divergence (CD) Method:** We consider an Energy-Based Model with probability density $p_\theta(x) = \frac{e^{-E_\theta(x)}}{Z(\theta)}$, where $Z(\theta) = \int e^{-E_\theta(x)} dx$ is the intractable partition function. The goal is to maximize the log-likelihood of the data $\mathcal{L}(\theta) = \mathbb{E}_{x \sim p_{data}}[\log p_\theta(x)]$. The gradient of the log-likelihood is derived as follows:

$$\nabla_\theta \mathcal{L}(\theta) = \nabla_\theta \mathbb{E}_{p_{data}}[-E_\theta(x) - \log Z(\theta)] = -\mathbb{E}_{p_{data}}[\nabla_\theta E_\theta(x)] - \nabla_\theta \log Z(\theta)$$

Using the identity

$$\nabla_\theta \log Z(\theta) = \frac{1}{Z(\theta)} \nabla_\theta \int e^{-E_\theta(x)} dx = \int \frac{e^{-E_\theta(x)}}{Z(\theta)} (-\nabla_\theta E_\theta(x)) dx = \mathbb{E}_{p_\theta}[-\nabla_\theta E_\theta(x)]$$

we get

$$\nabla_\theta \mathcal{L}(\theta) = \underbrace{-\mathbb{E}_{x \sim p_{data}}[\nabla_\theta E_\theta(x)]}_{\text{Positive Phase}} + \underbrace{\mathbb{E}_{x \sim p_\theta}[\nabla_\theta E_\theta(x)]}_{\text{Negative Phase}}$$

The "Negative Phase" requires calculating an expectation over the model distribution p_θ, which is computationally intractable because exact sampling from p_θ requires running an MCMC chain to convergence (theoretically infinite steps).

Contrastive Divergence (CD-k) approximates this expectation using a biased estimator. Instead of starting a Markov chain from random noise and running it to equilibrium, CD-k initializes the chain at a data sample $x^{(0)} \sim p_{data}$ and runs the MCMC transition kernel (e.g., Langevin Dynamics or Gibbs Sampling) for a small, fixed number of steps k. Let $\tilde{x}$ be the sample obtained after k steps ($x^{(0)} \to x^{(1)} \cdots \to x^{(k)} = \tilde{x}$). The gradient is approximated as:

$$\nabla_\theta \mathcal{L}(\theta) \approx -\nabla_\theta E_\theta(x^{(0)}) + \nabla_\theta E_\theta(\tilde{x})$$

Intuitively, this update lowers the energy of the data point $x^{(0)}$ (increasing its probability) and raises the energy of the confabulation point $\tilde{x}$ (decreasing its probability), thereby shaping the energy landscape locally around the data manifold.

Computational Cost and Time Complexity: Training EBMs via Contrastive Divergence is inherently more time-consuming than training standard feed-forward networks (such as classifiers, VAEs, or GAN discriminators) due to the nested iterative sampling process within the optimization loop.

In a standard neural network training step, the computational flow is linear:

$$\text{Input} \xrightarrow{\text{Forward}} \text{Loss} \xrightarrow{\text{Backward}} \text{Gradient}$$

In an EBM trained with CD-k, the "Negative Phase" requires an inner loop for sample generation:

$$\text{Input } x^{(0)} \xrightarrow{\text{MCMC Step 1}} x^{(1)} \dots \xrightarrow{\text{MCMC Step } k} \tilde{x} \xrightarrow{\text{Loss computation}} \text{Gradient}$$

Crucially, each single step of the MCMC process (e.g., Langevin Dynamics $x_{t+1} \leftarrow x_t - \frac{\epsilon}{2}\nabla_x E_\theta(x_t) + \sqrt{\epsilon}\xi$) requires computing the gradient of the energy with respect to the input, $\nabla_x E_\theta(x)$. This computation involves a full backpropagation pass through the network (from output energy to input pixels). Therefore, a single parameter update with CD-k requires roughly k times more computational operations (forward and backward passes) than a standard supervised update. Even for small k (e.g., $k = 10$), this effectively increases the training cost by an order of magnitude, making it computationally expensive to scale to high-dimensional data or complex architectures.

2. • **Advantage of Rejection Sampling (Exactness & Independence):** Rejection sampling produces *exact*, independent, and identically distributed (i.i.d.) samples from the target distribution $p(x)$ immediately upon acceptance. This stands in contrast to Metropolis-Hastings (and MCMC methods in general), which theoretically

converge to $p(x)$ only asymptotically as $n \to \infty$ (requiring a burn-in period) and produce samples that are serially correlated (Markovian dependence), reducing the effective sample size.

- **Disadvantage of Rejection Sampling (Curse of Dimensionality):** Rejection sampling requires a proposal distribution $q(x)$ and a constant M such that $p(x) \leq Mq(x)$ for all x. In high-dimensional spaces ($x \in \mathbb{R}^D$), the concentration of measure phenomenon causes the probability mass of $p(x)$ to occupy a vanishingly small fraction of the volume covered by $q(x)$ unless the distributions are identical. Consequently, the required bounding constant M typically grows exponentially with the dimension D (i.e., $M \propto c^D$). Since the acceptance probability is $1/M$, the efficiency drops exponentially to zero, rendering the method computationally intractable for high-dimensional models, whereas Metropolis-Hastings scales polynomially.

3. The target distribution is a multimodal mixture $p(x) = 0.6\mathcal{N}(50, 1) + 0.4\mathcal{N}(0.001, \sigma^2)$ with two distinct modes separated by a distance of $\Delta \approx 50$. The Metropolis-Hastings algorithm uses a local random-walk proposal $Q(x'|x) = \mathcal{N}(x, 0.5)$, which has a standard deviation of $\sigma_{prop} \approx 0.7$.

   The algorithm effectively fails to mix (is extremely slow) for two primary reasons:

   1. **Vanishing Probability of Direct Mode Jumping:** For the chain to jump directly from one mode (e.g., $x \approx 0$) to the other ($x \approx 50$) in a single step, the proposal distribution must generate a sample $x' \approx 50$ given $x \approx 0$. The probability of this event is proportional to the tail of the Gaussian proposal:

$$Q(50|0) \propto \exp\left(-\frac{(50-0)^2}{2(0.5)}\right) = \exp(-2500)$$

   This probability is numerically indistinguishable from zero. Thus, a direct transition between modes is statistically impossible.

   2. **Inability to Traverse the Low-Density Region:** Since direct jumps are impossible, the chain must traverse the intermediate region (e.g., $x \in [10, 40]$) via a sequence of small steps. However, in this region, the target probability density $p(x)$ is negligible. If the chain attempts to move from a high-probability mode into this low-probability "valley", the Metropolis acceptance ratio $\alpha = \min\left(1, \frac{p(x')}{p(x)}\right)$ becomes extremely small (since $p(x') \ll p(x)$). Consequently, almost all proposed moves away from the mode are rejected. The chain remains trapped in the local mode for an exponentially large number of iterations (metastability), leading to high autocorrelation and failure to explore the full distribution.

   Proposed Solutions:

   (a) **Parallel Tempering:** Run multiple MCMC chains in parallel at different temperatures T. Distributions with higher T (i.e., $p(x)^{1/T}$) are flatter, bridging the low-density valley and allowing chains to move freely between modes. Swapping states between high-temperature and low-temperature chains allows the target chain ($T = 1$) to escape local modes.

   (b) **Global Proposal / Mixture Kernel:** Modify the proposal distribution to be a mixture $Q(x'|x) = (1-\beta)Q_{local}(x'|x) + \beta Q_{global}(x')$, where Q_{global} has high variance

or covers the domain (e.g., independent MH). This allows the chain to occasionally propose global jumps that land directly in other modes, bypassing the low-density barrier.

Problem 5: Score Matching

We are given the prior $p(z) = e^{-z}$ for $z \geq 0$ and likelihood $p(x|z) = zxe^{-\frac{zx^2}{2}}$. The joint distribution is:

$$p(z, x) = p(z)p(x|z) = e^{-z} \cdot zxe^{-\frac{zx^2}{2}} = xze^{-z(1+\frac{x^2}{2})}$$

The true posterior is proportional to the joint (treating x as fixed):

$$p(z|x) \propto ze^{-z(1+\frac{x^2}{2})}$$

This is a Gamma distribution with shape parameter $\alpha = 2$ and rate parameter $\beta = 1 + \frac{x^2}{2}$. The true score with respect to z is:

$$s_{true}(z) = \frac{\partial}{\partial z} \log p(z|x) = \frac{\partial}{\partial z} \left(\log z - z \left(1 + \frac{x^2}{2} \right) + C \right) = \frac{1}{z} - \left(1 + \frac{x^2}{2} \right)$$

The approximation is $q(z|x) = \theta^2 ze^{-\theta z}$ (Gamma with shape 2, rate θ). The model score is:

$$s_\theta(z) = \frac{\partial}{\partial z} \log q(z|x) = \frac{\partial}{\partial z} (2 \log \theta + \log z - \theta z) = \frac{1}{z} - \theta$$

Score matching minimizes the expected squared difference between the scores:

$$J(\theta) = \frac{1}{2} \mathbb{E}_{z \sim p(z|x)} [(s_\theta(z) - s_{true}(z))^2]$$

Substitute the expressions:

$$s_\theta(z) - s_{true}(z) = \left(\frac{1}{z} - \theta \right) - \left(\frac{1}{z} - \left(1 + \frac{x^2}{2} \right) \right) = \left(1 + \frac{x^2}{2} \right) - \theta$$

The difference is a constant with respect to z. Let $K = 1 + \frac{x^2}{2}$. The objective becomes:

$$J(\theta) = \frac{1}{2} \mathbb{E}_{z \sim p(z|x)} [(K - \theta)^2] = \frac{1}{2} (K - \theta)^2$$

To minimize this, we set $\theta = K$.

$$\theta = 1 + \frac{x^2}{2}$$

This value results in $q(z|x)$ being exactly equal to $p(z|x)$, so the approximation is perfect.