Deep Generative Models

Fall 2025

Sharif University of Technology

Alireza Mirrokni - 401106617

Problem Set 4

Problem 1: Divergence minimization

1. To minimize $L_D(\phi;\theta)$, we express the loss in integral form:

$$L_D(\phi;\theta) = \int_{\mathcal{X}} \left[-p_{\text{data}}(x)\log D_\phi(x) - p_\theta(x)\log(1-D_\phi(x))\right] dx$$

To find the optimal $D_\phi(x)$, we minimize the integrand for each x. Let $a = p_{\text{data}}(x)$, $b = p_\theta(x)$, and $t = D_\phi(x)$. Define:

$$f(t) = -a\log t - b\log(1-t)$$

Taking the derivative with respect to t and setting it to zero:

$$f'(t) = -\frac{a}{t} + \frac{b}{1-t} = 0 \implies \frac{b}{1-t} = \frac{a}{t}$$

$$bt = a(1-t) \implies bt = a - at \implies t(a+b) = a$$

$$t = \frac{a}{a+b}$$

Substituting the distributions back:

$$D^*(x) = \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_\theta(x)}$$

Since $f''(t) = \frac{a}{t^2} + \frac{b}{(1-t)^2} > 0$ for $t \in (0,1)$, this solution is a unique global minimum.

2. Given $D_\phi(x) = \sigma(h_\phi(x)) = \frac{1}{1+e^{-h_\phi(x)}}$, we set $D_\phi(x) = D^*(x)$:

$$\frac{1}{1+e^{-h_\phi(x)}} = \frac{p_{\text{data}}(x)}{p_\theta(x) + p_{\text{data}}(x)}$$

Taking the reciprocal:

$$1 + e^{-h_\phi(x)} = \frac{p_\theta(x) + p_{\text{data}}(x)}{p_{\text{data}}(x)} = \frac{p_\theta(x)}{p_{\text{data}}(x)} + 1$$

$$e^{-h_\phi(x)} = \frac{p_\theta(x)}{p_{\text{data}}(x)}$$

Taking the natural logarithm:

$$-h_\phi(x) = \log\frac{p_\theta(x)}{p_{\text{data}}(x)} \implies h_\phi(x) = \log\left(\frac{p_\theta(x)}{p_{\text{data}}(x)}\right)^{-1} = \log\frac{p_{\text{data}}(x)}{p_\theta(x)}$$

3. Substitute $D_\phi = D^*$ into the generator loss $L_G(\theta;\phi) = \mathbb{E}_{x\sim p_\theta(x)}[\log(1 - D_\phi(x)) - \log D_\phi(x)]$:

$$1 - D^*(x) = 1 - \frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_\theta(x)} = \frac{p_\theta(x)}{p_{\text{data}}(x) + p_\theta(x)}$$

The expression inside the expectation becomes:

$$\log\left(\frac{p_\theta(x)}{p_{\text{data}}(x) + p_\theta(x)}\right) - \log\left(\frac{p_{\text{data}}(x)}{p_{\text{data}}(x) + p_\theta(x)}\right) = \log\left(\frac{\frac{p_\theta(x)}{p_{\text{data}}(x)+p_\theta(x)}}{\frac{p_{\text{data}}(x)}{p_{\text{data}}(x)+p_\theta(x)}}\right) = \log\frac{p_\theta(x)}{p_{\text{data}}(x)}$$

Therefore:

$$L_G(\theta;\phi) = \mathbb{E}_{x\sim p_\theta(x)}\left[\log\frac{p_\theta(x)}{p_{\text{data}}(x)}\right] = \text{KL}(p_\theta(x) \,\|\, p_{\text{data}}(x))$$

4.
- The negative log likelihood is:

$$-\mathbb{E}_{x\sim p_{\text{data}}}[\log p_\theta(x)] = -\int p_{\text{data}}(x)\log p_\theta(x)dx$$

Adding and subtracting $\int p_{\text{data}}(x)\log p_{\text{data}}(x)dx$:

$$= \int p_{\text{data}}(x)\log\frac{p_{\text{data}}(x)}{p_\theta(x)}dx - \int p_{\text{data}}(x)\log p_{\text{data}}(x)dx$$

$$= \text{KL}(p_{\text{data}}(x) \,\|\, p_\theta(x)) + H(p_{\text{data}})$$

Since $H(p_{\text{data}})$ does not depend on θ, it is a constant.

- No, they are not learning the same objective. The VAE minimizes $\text{KL}(p_{\text{data}} \,\|\, p_\theta)$, which is mean-seeking (mode-covering), leading the model to cover all modes of the data. The GAN generator minimizes $\text{KL}(p_\theta \,\|\, p_{\text{data}})$, which is mode-seeking, often leading the model to focus on specific sharp modes (mode collapse).

Problem 2: Distinguishing Real Data from Generated Data

1. (a) The risk function $R_l(D)$ is the expectation of the loss $l(yD(x))$ over the joint distribution of (x, y). Given $P(x, y = 1) = 0.5P_d(x)$ and $P(x, y = -1) = 0.5P_g(x)$, we rewrite the expectation as an integral:

$$R_l(D) = \int_{\mathcal{X}} \left[l(D(x))P(x, y = 1) + l(-D(x))P(x, y = -1)\right] dx$$

Substituting the given probabilities:

$$\inf_{D\in\mathcal{D}}\left(\int_{\mathcal{X}} \left(0.5l(D(x))P_d(x) + 0.5l(-D(x))P_g(x)\right) dx\right)$$

(b) Under the assumption of unlimited capacity for the set $\mathcal{D}$, the function $D(x)$ can be optimized independently for each point x. The optimal discriminator D^* minimizes the integrand pointwise:

$$g(D) = 0.5P_d(x)l(D) + 0.5P_g(x)l(-D)$$

Setting the derivative with respect to D to zero:

$$g'(D) = 0.5P_d(x)l'(D) - 0.5P_g(x)l'(-D) = 0 \implies \frac{l'(D)}{l'(-D)} = \frac{P_g(x)}{P_d(x)}$$

This implies that the optimal value $D^*(x)$ is a function of the likelihood ratio $r(x) = \frac{P_d(x)}{P_g(x)}$. Because D is optimized at every point without constraint, the infimum of the integral is the integral of the pointwise infima.

(c) By substituting the pointwise optimum $D^*(r(x))$ back into the risk expression, we obtain:

$$R_l^*(\mathcal{D}) = 0.5 \int_{\mathcal{X}} [P_d(x)l(D^*(r(x))) + P_g(x)l(-D^*(r(x)))]\, dx$$

Factoring out $P_g(x)$ inside the integral:

$$R_l^*(\mathcal{D}) = \frac{1}{2} \int_{\mathcal{X}} \left[\frac{P_d(x)}{P_g(x)} l(D^*(r(x))) + l(-D^*(r(x)))\right] P_g(x) dx$$

Let $u = \frac{P_d(x)}{P_g(x)}$. We define the function $f(u)$ as:

$$f(u) = -\left(u \cdot l(D^*(u)) + l(-D^*(u))\right)$$

To show convexity, observe that $f(u) = \sup_D\{-u \cdot l(D) - l(-D)\}$; since $f(u)$ is the pointwise supremum of a family of functions affine in u, it is a convex function. To show it is decreasing, consider $u_1 < u_2$. By the definition of the infimum,

$$f(u_2) = -\inf_D\{u_2 l(D) + l(-D)\} \leq -(u_2 l(D^*(u_1)) + l(-D^*(u_1)))$$

Since standard margin-based loss functions satisfy $l(z) > 0$, the term $u_2 l(D^*(u_1))$ is strictly greater than $u_1 l(D^*(u_1))$, which implies

$$f(u_2) < -(u_1 l(D^*(u_1)) + l(-D^*(u_1))) = f(u_1)$$

Substituting $f(u)$ back into the risk expression yields:

$$R_l^*(\mathcal{D}) = -\frac{1}{2} \int_{\mathcal{X}} f\left(\frac{P_d(x)}{P_g(x)}\right) P_g(x) dx = -\frac{1}{2}\mathbb{I}_f(P_d, P_g)$$

where $\mathbb{I}_f(P_d, P_g)$ is the f-divergence between the data and generator distributions.

2. (a) We aim to minimize the pointwise objective function for a fixed x:

$$h(D) = u \cdot l_1(D) + l_1(-D) = u \cdot \mathbb{I}(D \leq 0) + \mathbb{I}(D \geq 0)$$

where $u = \frac{P_d(x)}{P_g(x)}$. We analyze the value of $h(D)$ for different signs of D:

- If $D > 0$: $\mathbb{I}(D \leq 0) = 0$ and $\mathbb{I}(D \geq 0) = 1$. Thus, $h(D) = 0 \cdot u + 1 = 1$.
- If $D < 0$: $\mathbb{I}(D \leq 0) = 1$ and $\mathbb{I}(D \geq 0) = 0$. Thus, $h(D) = 1 \cdot u + 0 = u$.
- If $D = 0$: $\mathbb{I}(D \leq 0) = 1$ and $\mathbb{I}(D \geq 0) = 1$. Thus, $h(D) = u + 1$.

Comparing the possible costs, the minimum value is $\min(1, u)$. Therefore, the optimal discriminator D^* chooses the sign based on whether $u < 1$ or $u > 1$:

$$\inf_D h(D) = \min(1, u)$$

Using the definition $f(u) = -\inf_D(ul(D) + l(-D))$, we have:

$$f(u) = -\min(1, u) = \max(-1, -u)$$

The function $f(u)$ is the pointwise maximum of two linear functions (-1 and $-u$), which implies it is convex. Also since $u \geq 0$, as u increases, $-u$ decreases. Thus $\max(-1, -u)$ is non-increasing (decreasing until $u = 1$, then constant).
Substituting $f(u)$ into the risk expression:

$$R(D^*) = -\frac{1}{2}\int f\left(\frac{P_d(x)}{P_g(x)}\right) P_g(x)dx = \frac{1}{2}\int \min\left(1, \frac{P_d(x)}{P_g(x)}\right) P_g(x)dx$$

Simplifying the integrand:

$$\min\left(1, \frac{P_d(x)}{P_g(x)}\right) P_g(x) = \min(P_g(x), P_d(x))$$

Using the identity $\min(a, b) = \frac{a+b-|a-b|}{2}$:

$$\int \min(P_g(x), P_d(x))dx = \int \frac{P_g(x) + P_d(x) - |P_g(x) - P_d(x)|}{2}dx$$

Since $\int P_g(x)dx = 1$ and $\int P_d(x)dx = 1$:

$$R(D^*) = \frac{1}{2}\left[\frac{1 + 1 - \int |P_g(x) - P_d(x)|dx}{2}\right] = \frac{1}{2}\left(1 - \frac{1}{2}\int |P_d - P_g|dx\right)$$

Recognizing the definition of Total Variation distance $\mathbb{I}_{\text{TV}}(P_d, P_g) = \frac{1}{2}\int |P_d - P_g|dx$:

$$R(D^*) = \frac{1}{2}(1 - \mathbb{I}_{\text{TV}}(P_d, P_g))$$

(b) First, we compute the derivative of the loss function:

$$l'(z) = \frac{-e^{-z}}{1 + e^{-z}} = \frac{-1}{e^z + 1}$$

Using the optimality condition derived in Part 1(b), $\frac{l'(D)}{l'(-D)} = \frac{1}{u}$ (where $u = P_d/P_g$):

$$\frac{\frac{-1}{e^D+1}}{\frac{-1}{e^{-D}+1}} = \frac{e^{-D} + 1}{e^D + 1} = \frac{1 + e^{-D}}{e^D(e^{-D} + 1)} = e^{-D}$$

Setting $e^{-D} = \frac{1}{u} \implies -D = -\log u$, we find the optimal discriminator:

$$D^*(u) = \log u = \log \frac{P_d(x)}{P_g(x)}$$

We evaluate the expression inside the supremum:

$$l(D^*) = \log(1 + e^{-\log u}) = \log\left(1 + \frac{1}{u}\right) = \log\left(\frac{u+1}{u}\right)$$

$$l(-D^*) = \log(1 + e^{\log u}) = \log(1 + u)$$

Computing $f(u) = -(ul(D^*) + l(-D^*))$:

$$\begin{aligned} f(u) &= -\left[u \log\left(\frac{u+1}{u}\right) + \log(1+u)\right] \\ &= -\left[u(\log(u+1) - \log u) + \log(u+1)\right] \\ &= u \log u - (u+1)\log(u+1) \end{aligned}$$

Taking derivatives,

$$f'(u) = (\log u + 1) - (\log(u+1) + 1) = \log \frac{u}{u+1}$$

Then

$$f''(u) = \frac{1}{u} - \frac{1}{u+1} = \frac{1}{u(u+1)} > 0$$

for $u > 0$. Thus f is convex. Also since $u < u + 1$, $\frac{u}{u+1} < 1$, so $f'(u) = \log \frac{u}{u+1} < 0$. Thus f is decreasing.

Substituting the derived expression for $f(u)$ into the risk equation $R(D^*) = -\frac{1}{2}\int f(u)P_g(x)dx$, we have:

$$R(D^*) = \frac{1}{2}\int \left[(u+1)\log(u+1) - u \log u\right] P_g(x)dx$$

Substituting $u = \frac{P_d(x)}{P_g(x)}$, note that $(u+1)P_g(x) = P_d(x) + P_g(x)$ and $uP_g(x) = P_d(x)$. The integral becomes:

$$R(D^*) = \frac{1}{2}\int \left[(P_d(x) + P_g(x)) \log\left(\frac{P_d(x) + P_g(x)}{P_g(x)}\right) - P_d(x) \log\left(\frac{P_d(x)}{P_g(x)}\right)\right] dx$$

Expanding the logarithms using $\log(a/b) = \log a - \log b$:

$$\begin{aligned} R(D^*) &= \frac{1}{2}\int \Big[(P_d + P_g)(\log(P_d + P_g) - \log P_g) - P_d(\log P_d - \log P_g)\Big] dx \\ &= \frac{1}{2}\int \Big[(P_d + P_g)\log(P_d + P_g) - (P_d + P_g)\log P_g - P_d \log P_d + P_d \log P_g\Big] dx \end{aligned}$$

Simplifying the terms involving $\log P_g$ (observing that $-(P_d + P_g) + P_d = -P_g$):

$$R(D^*) = \frac{1}{2}\int \Big[(P_d + P_g)\log(P_d + P_g) - P_d \log P_d - P_g \log P_g\Big] dx$$

We introduce the mixture distribution $M(x) = \frac{P_d(x)+P_g(x)}{2}$, which implies $P_d + P_g = 2M$. Substituting this into the first term:

$$(P_d + P_g)\log(P_d + P_g) = 2M \log(2M) = 2M(\log 2 + \log M)$$

Substituting back into the integral and grouping terms:

$$R(D^*) = \frac{1}{2}\int \Big[2M\log 2 + 2M\log M - P_d\log P_d - P_g\log P_g\Big]dx$$
$$= \log 2\int M(x)dx + \frac{1}{2}\int \Big[(P_d+P_g)\log M - P_d\log P_d - P_g\log P_g\Big]dx$$

Since $\int M(x)dx = 1$, the first term is $\log 2$. We regroup the remaining terms to form the KL divergences:

$$R(D^*) = \log 2 + \frac{1}{2}\int \left[P_d(\log M - \log P_d) + P_g(\log M - \log P_g)\right]dx$$
$$= \log 2 - \frac{1}{2}\left[\int P_d \log\frac{P_d}{M}dx + \int P_g\log\frac{P_g}{M}dx\right]$$
$$= \log 2 - \frac{1}{2}\Big[\mathrm{KL}(P_d||M) + \mathrm{KL}(P_g||M)\Big]$$

By definition, the bracketed term is twice the Jensen-Shannon divergence:

$$R(D^*) = \log 2 - \mathrm{JSD}(P_d||P_g)$$

3. We interpret the label Z and sample X as random variables with joint distribution $P(x,z)$. The mutual information is defined as:

$$I(X;Z) = \sum_{z\in\{0,1\}}\int_{\mathcal{X}} P(x,z)\log\left(\frac{P(x,z)}{P(x)P(z)}\right)dx$$

We define the components provided in the problem statement:

- Priors: $P(Z=1) = P(Z=0) = \frac{1}{2}$.
- Conditionals: $P(x|Z=1) = P_d(x)$ and $P(x|Z=0) = P_g(x)$.
- Joints: $P(x,Z=1) = \frac{1}{2}P_d(x)$ and $P(x,Z=0) = \frac{1}{2}P_g(x)$.
- Marginal: $P(x) = M(x) = \frac{P_d(x)+P_g(x)}{2}$.

Substitute these into the mutual information expression:

$$I(X;Z) = \int \frac{1}{2}P_d(x)\log\left(\frac{\frac{1}{2}P_d(x)}{M(x)\cdot\frac{1}{2}}\right)dx + \int\frac{1}{2}P_g(x)\log\left(\frac{\frac{1}{2}P_g(x)}{M(x)\cdot\frac{1}{2}}\right)dx$$

Simplifying the fractions inside the logarithms (canceling the factor $\frac{1}{2}$ in numerator and denominator):

$$I(X;Z) = \frac{1}{2}\int P_d(x)\log\left(\frac{P_d(x)}{M(x)}\right)dx + \frac{1}{2}\int P_g(x)\log\left(\frac{P_g(x)}{M(x)}\right)dx$$

We recognize these integrals as the Kullback-Leibler divergences between the conditional distributions and the mixture marginal M:

$$I(X;Z) = \frac{1}{2}\mathrm{KL}(P_d||M) + \frac{1}{2}\mathrm{KL}(P_g||M)$$

This is precisely the definition of the Jensen-Shannon divergence:

$$I(X;Z) = \mathrm{JSD}(P_d||P_g)$$

Problem 3: Mode collapse and Wasserstein GAN

1. Mode collapse is a pathological failure mode in Generative Adversarial Networks (GANs) where the generator learns to map distinct input noise vectors z to the same output sample (or a very small set of output samples), rather than learning the diverse probability distribution of the real data. In a severe case (complete collapse), the generator produces a single image regardless of the input. In partial collapse, the generator might cover only a few modes (e.g., generating only ones and sevens when trained on the MNIST dataset, ignoring all other digits).

   It typically arises from the nature of the minimax game:

   $$\min_G \max_D V(D, G)$$

   Ideally, the generator and discriminator should converge to a Nash equilibrium where the generator matches the data distribution. However, during training:

   - If the discriminator is too weak or gets stuck in a local minimum, the generator may find a single sample x^* that the discriminator currently rates as highly realistic.
   - The generator then collapses its probability mass onto x^* to minimize its loss quickly.
   - Once the generator collapses, the discriminator learns to identify this specific sample x^* as fake. The generator then simply moves its mass to a distinct single point x' to fool the discriminator again.
   - This results in a "cat-and-mouse" cycle where the generator hops between modes but never spreads its mass to cover the full distribution.

2. Mode collapse is critical because the primary goal of generative modeling is often to capture the diversity of the underlying data distribution, not just to produce high-quality samples.

   - **Loss of Diversity:** In creative applications (e.g., art generation, character design), a generator that outputs identical or near-identical images is useless.
   - **Data Biasing:** If a GAN is used to generate synthetic training data for other models (e.g., for autonomous driving), missing modes (rare scenarios) can lead to dangerous downstream failures.

   Mode collapse is less problematic in conditional tasks where the mapping is inherently deterministic or where specific optimization is preferred over diversity:

   - **Super-Resolution:** The goal is to recover the single "ground truth" high-resolution image from a low-resolution input. Diversity is not the primary objective, fidelity is.
   - **Image Denoising/Restoration:** Similar to super-resolution, there is often one optimal clean signal to recover.
   - **Style Transfer:** If the goal is simply to apply a specific texture to an image, the user may be satisfied with a single, high-quality stylistic translation rather than a distribution of potential styles.

3. The Wasserstein distance is superior to KL divergence (and the related Jensen-Shannon divergence used in standard GANs) because it provides meaningful gradients even when the supports of the generated distribution and the real distribution are disjoint.

Fundamental Differences:

- **Topology:** The KL divergence creates a very strong topology where distributions are only "close" if their densities overlap significantly. If supports are disjoint, the distance is often infinite or undefined.
- **Geometry:** The Wasserstein distance leverages the geometry of the underlying space (the metric $\|x-y\|$). It reflects the horizontal transport cost to move the probability mass from one distribution to the other.

Now to mathematically illustrate the difference, let P be the real data distribution and Q_θ be the generator distribution parameterized by θ. Consider a simple 1D example where both are uniform distributions with width ϵ, but Q is shifted by θ. For simplicity, let $\epsilon \to 0$, effectively treating them as Dirac deltas:

$$P(x) = \delta(x)$$

$$Q_\theta(x) = \delta(x - \theta)$$

Assume $\theta \neq 0$ (the supports are disjoint).

1. Kullback-Leibler Divergence $KL(Q_\theta||P)$:

$$KL(Q_\theta||P) = \int Q_\theta(x) \log\left(\frac{Q_\theta(x)}{P(x)}\right) dx$$

Since the supports are disjoint, at $x = \theta$ where $Q_\theta(x)$ has mass, $P(x) = 0$.

$$KL(Q_\theta||P) = +\infty$$

Similarly, the Jensen-Shannon divergence (used in the standard GAN objective) would saturate to a constant (specifically $\log 2$) when supports are disjoint. Also for the gradient, we have:

$$\nabla_\theta KL(Q_\theta||P) = \text{Undefined or } 0$$

If the divergence is infinite (or a constant like in JS), there is no usable gradient information ($\nabla_\theta \text{Loss} = 0$) to guide the generator towards the data (i.e., to reduce θ). The generator suffers from vanishing gradients.

2. Wasserstein-1 Distance $W(P, Q_\theta)$: Using the primal form definition as the "earth mover's distance":

$$W(P, Q_\theta) = \inf_{\gamma \in \Pi(P, Q_\theta)} \mathbb{E}_{(x,y)\sim\gamma}[\|x - y\|]$$

Since both distributions are point masses (or non-overlapping uniform blocks), the only way to transport mass from $\delta(x - \theta)$ to $\delta(x)$ is to move the entire mass by a distance of $|\theta|$.

$$W(P, Q_\theta) = |\theta|$$

Also for the gradient, we have:

$$\nabla_\theta W(P, Q_\theta) = \nabla_\theta |\theta| = \text{sign}(\theta)$$

The gradient is either 1 or -1 (specifically, it points towards 0). This is a constant, non-vanishing gradient that exists everywhere, regardless of how far apart the distributions are.

Therefore, as the generator moves closer to the data ($\theta \to 0$), the KL divergence jumps abruptly from $+\infty$ to 0, providing no guidance. The Wasserstein distance decreases smoothly ($|\theta| \to 0$), providing a stable gradient flow that allows the W-GAN to train effectively even when the generated distribution and real distribution currently have no overlap.

Problem 4: f-GAN Objective and Hellinger Distance

1. First, we establish the relationship between the last-layer output $V_\omega(x)$ and the discriminator output $D(x)$. Since the activation function is the sigmoid, we have:

$$D(x) = \sigma(V_\omega(x)) = \frac{1}{1 + e^{-V_\omega(x)}}$$

Inverting this relationship to express exponentials of $V_\omega(x)$ in terms of $D(x)$:

$$1 + e^{-V_\omega(x)} = \frac{1}{D(x)} \implies e^{-V_\omega(x)} = \frac{1}{D(x)} - 1 = \frac{1 - D(x)}{D(x)}$$

Consequently, $e^{V_\omega(x)} = \frac{D(x)}{1-D(x)}$.

Now we substitute this into the functions g_f and f^*. For the first term involving $g_f(V_\omega(x))$:

$$g_f(V_\omega(x)) = 1 - \exp(-V_\omega(x)) = 1 - \frac{1 - D(x)}{D(x)} = \frac{D(x) - (1 - D(x))}{D(x)} = \frac{2D(x) - 1}{D(x)} = 2 - \frac{1}{D(x)}$$

For the second term involving $f^*(g_f(V_\omega(x)))$, let $t = g_f(V_\omega(x))$. Note that $t = 1 - e^{-V}$. Then:

$$f^*(t) = \frac{t}{1 - t} = \frac{1 - e^{-V_\omega(x)}}{1 - (1 - e^{-V_\omega(x)})} = \frac{1 - e^{-V_\omega(x)}}{e^{-V_\omega(x)}} = e^{V_\omega(x)} - 1$$

Substituting $e^{V_\omega(x)} = \frac{D(x)}{1-D(x)}$:

$$f^*(g_f(V_\omega(x))) = \frac{D(x)}{1 - D(x)} - 1 = \frac{D(x) - (1 - D(x))}{1 - D(x)} = \frac{2D(x) - 1}{1 - D(x)} = \frac{1}{1 - D(x)} - 2$$

Substituting these into the original objective function $F(\theta, \omega)$:

$$\begin{aligned} F(\theta, \omega) &= \mathbb{E}_{x \sim P}\left[2 - \frac{1}{D(x)}\right] - \mathbb{E}_{x \sim Q_\theta}\left[\frac{1}{1 - D(x)} - 2\right] \\ &= \int p(x)\left(2 - \frac{1}{D(x)}\right) dx - \int q(x)\left(\frac{1}{1 - D(x)} - 2\right) dx \\ &= 2\int p(x)dx + 2\int q(x)dx - \int \frac{p(x)}{D(x)}dx - \int \frac{q(x)}{1 - D(x)}dx \end{aligned}$$

Since $p(x)$ and $q(x)$ are probability distributions, they integrate to 1. Thus:

$$F(\theta, \omega) = 4 - \int \left(\frac{p(x)}{D(x)} + \frac{q(x)}{1 - D(x)}\right) dx$$

2. To optimize the objective function with respect to $D(x)$ (maximizing the objective for the discriminator), we consider the integrand point-wise for a fixed x. Let $y = D(x)$. We want to maximize:

$$J(y) = p(x)\left(2 - \frac{1}{y}\right) - q(x)\left(\frac{1}{1-y} - 2\right)$$

Taking the derivative with respect to y:

$$\frac{dJ}{dy} = p(x)\left(\frac{1}{y^2}\right) - q(x)\left(\frac{1}{(1-y)^2}\right)$$

Setting the derivative to zero to find the critical point:

$$\frac{p(x)}{y^2} = \frac{q(x)}{(1-y)^2} \implies \frac{(1-y)^2}{y^2} = \frac{q(x)}{p(x)}$$

Taking the square root (since $D(x) \in (0,1)$):

$$\frac{1-y}{y} = \sqrt{\frac{q(x)}{p(x)}} \implies \frac{1}{y} - 1 = \frac{\sqrt{q(x)}}{\sqrt{p(x)}}$$

$$\frac{1}{y} = \frac{\sqrt{q(x)} + \sqrt{p(x)}}{\sqrt{p(x)}}$$

Solving for y:

$$D^*(x) = \frac{\sqrt{p(x)}}{\sqrt{p(x)} + \sqrt{q(x)}}$$

3. We substitute the optimal discriminator $D^*(x)$ back into the objective function derived in Part 1. Note that $\frac{1}{D^*(x)} = \frac{\sqrt{p}+\sqrt{q}}{\sqrt{p}}$ and $\frac{1}{1-D^*(x)} = \frac{\sqrt{p}+\sqrt{q}}{\sqrt{q}}$.

$$\begin{aligned}
F(\theta, \omega^*) &= 4 - \int \left(p(x)\frac{\sqrt{p}+\sqrt{q}}{\sqrt{p}} + q(x)\frac{\sqrt{p}+\sqrt{q}}{\sqrt{q}}\right) dx \\
&= 4 - \int \left(\sqrt{p}(\sqrt{p}+\sqrt{q}) + \sqrt{q}(\sqrt{p}+\sqrt{q})\right) dx \\
&= 4 - \int \left(p + \sqrt{pq} + \sqrt{pq} + q\right) dx \\
&= 4 - \left(\int p(x)dx + 2\int \sqrt{p(x)q(x)}dx + \int q(x)dx\right)
\end{aligned}$$

Using $\int p(x)dx = 1$ and $\int q(x)dx = 1$:

$$F(\theta, \omega^*) = 4 - (1 + 2\int \sqrt{p(x)q(x)}dx + 1) = 2 - 2\int \sqrt{p(x)q(x)}dx$$

We can rewrite this expression by completing the square for the Hellinger distance form:

$$\begin{aligned}
2 - 2\int \sqrt{p(x)q(x)}dx &= \int (p(x) + q(x))dx - 2\int \sqrt{p(x)q(x)}dx \\
&= \int \left((\sqrt{p(x)})^2 - 2\sqrt{p(x)}\sqrt{q(x)} + (\sqrt{q(x)})^2\right) dx \\
&= \int \left(\sqrt{p(x)} - \sqrt{q(x)}\right)^2 dx
\end{aligned}$$

Justification: The optimized objective function is exactly the squared Hellinger Distance (specifically $2 \cdot H^2(P, Q)$ under the standard definition $H^2(P, Q) = \frac{1}{2} \int (\sqrt{p} - \sqrt{q})^2 d\mu$). Minimizing this objective with respect to the generator minimizes the distance between the true distribution P and the generated distribution Q.

4. In the case of exact and optimal training of the generator, the generated distribution matches the true distribution perfectly. Therefore, $q(x) = p(x)$ for all x. Substituting this into the optimal discriminator formula found in Part 2:

$$D^*(x) = \frac{\sqrt{p(x)}}{\sqrt{p(x)} + \sqrt{p(x)}} = \frac{\sqrt{p(x)}}{2\sqrt{p(x)}} = \frac{1}{2}$$

Thus, at optimum, the discriminator is unable to distinguish between real and fake data and outputs 0.5 everywhere.

Problem 5: Conditional GAN with Auxiliary Classifier (AC-GAN)

1. The goal of the generator in a Conditional GAN (specifically the AC-GAN framework described) is to approximate the true data distribution.

   (i) The conditional distribution $p_G(x \mid c)$ is trying to match the true conditional data distribution:

   $$p_{\text{data}}(x \mid c)$$

   This means that for a specific class label c, the generator attempts to produce images indistinguishable from real images of that specific class.

   (ii) Under the assumption that the prior over classes is correct (i.e., $p(c) = p_{\text{data}}(c)$), the model is approximating the joint distribution:

   $$p_{\text{data}}(x, c) = p_{\text{data}}(c)\, p_{\text{data}}(x \mid c)$$

   Since the generator samples $c \sim p(c)$ and creates x, the joint generated distribution is $p_G(x, c) = p(c)p_G(x \mid c)$. By matching the conditional term, the joint term is implicitly matched.

2. The four loss terms function as follows:

   - $L_{\text{adv}}(D)$: This is the standard adversarial loss for the discriminator. It optimizes the discriminator to correctly distinguish between real images ($x \sim p_{\text{data}}$) and fake images ($x \sim p_G$). It pushes $D_{\text{adv}}(x) \to 1$ for real data and $D_{\text{adv}}(G(z, c)) \to 0$ for generated data.
   - $L_{\text{adv}}(G)$: This is the adversarial loss for the generator. It optimizes the generator to fool the discriminator. It attempts to maximize the probability that the discriminator identifies generated images as real (pushing $D_{\text{adv}}(G(z, c)) \to 1$). This term drives the realism of the samples.
   - $L_{\text{class}}(D)$: This is the classification loss for the discriminator. It optimizes the auxiliary classifier head to correctly predict the class label c of **real** images. It ensures the discriminator learns the features relevant to the class structure of the dataset.

- $L_{\text{class}}(G)$: This is the classification loss for the generator. It optimizes the generator to produce images that the discriminator's classifier head classifies as the target label c. This term drives the **label consistency** (conditioning) of the samples.

Impact of λ: The hyperparameter λ controls the trade-off between realism and label consistency.

- If λ is very small, the generator focuses almost entirely on fooling the adversarial head (L_{adv}). The generated images may look realistically like "an image" from the dataset, but the generator may ignore the input label c (e.g., generating a dog when asked for a cat), resulting in low label consistency.
- If λ is very large, the generator focuses entirely on the classification loss. It may generate images that are extremely easy for the classifier to identify (e.g., exaggerated features or "adversarial examples" for the classifier) but lack visual realism or diversity.

3. In an unconditional GAN trained only with L_{adv}, the generator often suffers from mode collapse where it outputs samples from only a single mode (or very few modes) of the data distribution, effectively ignoring the diversity of the dataset.

   How classification losses help: The inclusion of $L_{\text{class}}(D)$ and $L_{\text{class}}(G)$ (as in AC-GAN) explicitly forces the generator to cover all classes. Because the generator is penalized if the discriminator cannot classify the generated image as the specific input class c, the generator is forced to learn separate modes corresponding to each class $c \in \{1, \ldots, C\}$. This prevents inter-class mode collapse (e.g., the model cannot simply generate "shoes" for every input; it must generate "shirts" when $c = \text{shirt}$).

   Remaining mode collapse: While AC-GAN alleviates collapse across classes, it is still susceptible to intra-class mode collapse. For a fixed class c (e.g., "shoe"), the generator might still produce only one specific type of shoe (e.g., a red sneaker) regardless of the noise vector z. The classification loss only requires the image to be recognizable as a "shoe", not that it covers the distribution of all possible shoes.

4. (a) We are given the generator classification loss:

$$L_{\text{class}}(G) = -\mathbb{E}_{z\sim p(z), c\sim p(c)} \left[\log D_{\text{class}}(c \mid G_\theta(z, c))\right]$$

   The expectation over z and c where $x = G_\theta(z, c)$ is equivalent to the expectation over the joint generated distribution $(x, c) \sim p_G(x, c)$. Substituting the assumption that the classifier is Bayes-optimal, $D_{\text{class}}(c \mid x) = p_G(c \mid x)$, we get:

$$L_{\text{class}}(G) = -\mathbb{E}_{(x,c)\sim p_G(x,c)} \left[\log p_G(c \mid x)\right]$$

   By the definition of conditional entropy, $H(Y \mid X) = -\mathbb{E}_{x,y}[\log p(y \mid x)]$. Thus, applying this to our variables c and x:

$$L_{\text{class}}(G) = H_G(c \mid x)$$

   (Note: Any additive constant independent of θ is zero here, making the relation exact under the optimality assumption).

(b) We know the definition of Mutual Information between the label c and the generated image x under the generator distribution is:

$$I_G(c; x) = H_G(c) - H_G(c \mid x)$$

Rearranging this for the conditional entropy term:

$$H_G(c \mid x) = H_G(c) - I_G(c; x)$$

Substituting this into the result from part (a):

$$L_{\text{class}}(G) = H_G(c) - I_G(c; x)$$

The term $H_G(c)$ is the entropy of the class labels fed into the generator. Since the classes c are sampled from a fixed prior $p(c)$ (usually uniform or empirical frequencies) which is independent of the generator's parameters θ, $H_G(c)$ is a constant with respect to θ.

Therefore, minimizing the loss $L_{\text{class}}(G)$ is equivalent to minimizing $-I_G(c; x)$, which is exactly equivalent to maximizing the mutual information $I_G(c; x)$ between the generated image and the label.