TTIC 31230 Fundamentals of Deep Learning

Problems for GANs.

Problem 1. Conditional GANs In a conditional GAN we model a conditional distribution $\mathrm{Pop}(y|x)$ defined by a population distribution on pairs $\langle x, y\rangle$. For conditional GANs we consider the probability distribution over triples $\langle x, y, i\rangle$ defined by

$$\begin{aligned} \tilde{P}_\Phi(i = 1) &= 1/2 \\ \tilde{P}_\Phi(y|x, i = 1) &= \mathrm{pop}(y|x) \\ \tilde{P}_\Phi(y|x, i = -1) &= p_\Phi(y|x) \end{aligned}$$

(a) Write the conditional GAN adversarial objective function for this problem in terms of $\tilde{P}(x, y, i)$, $P_\Phi(y|x)$ and $P_\Psi(i|y, x)$.

Solution:

$$\Phi^* = \operatorname*{argmax}_\Phi \min_\Psi E_{x,y,i\sim\tilde{P}(x,y,i)} \; -\ln P_\Psi(i|x, y)$$

Problem 2. GAN instability

Consider the following adversarial objective where x and y are scalars (real numbers).

$$\max_x \min_y xy$$

(a) Write the differential equation for gradient flow of this adversarial objective.

(b) Give a general solution to your differential equation. (Hint: It goes in a circle). You solution should have parameters allowing for any given initial value of x and y.

Problem 3. Contrastive GANs.

A GAN can be built with a “contrastive” discriminator. Rather than estimate the probability that y is from the population, the discriminator must select which of $y_1, \ldots, y_N$ is from the population.

More formally, for $N \geq 2$ let $\tilde{P}_\Phi^{(N)}$ be the distribution on tuples $\langle i, y_1, \ldots, y_N\rangle$ defined by drawing one “positive” from Pop and $N - 1$ IID negatives from P_Φ; then inserting the positive at a random position among the negatives; and returning $(i, y_1, \ldots, y_N)$ where i is the index of the positive.

$$\Phi^* = \operatorname*{argmax}_{\Phi} \min_{\Psi} \; E_{(i,y_1,\ldots,y_{N+1}) \sim \tilde{P}_\Phi^{(N)}} \; -\ln p_\Psi(i|y_1,\ldots,y_{N+1}) \quad (1)$$

Restate the above definition of $\tilde{P}_\Phi^{(N)}$ and the GAN adversarial objective for the case of conditional constrastive GANs.

Solution:

$$\Phi^* = \operatorname*{argmax}_{\Phi} \min_{\Psi} \; E_{(i,y_1,\ldots,y_{N+1},x) \sim \tilde{P}_\Phi^{(N)}} \ln -P_\Psi(i|y_1,\ldots,y_{N+1},x)$$

Problem 4. Reshaping Noise in GANs. A GAN generator is typically given a random noise vector $z \sim \mathcal{N}(0,I)$. Give equations defining a method for computing z' from z such that the distribution on z' is a mixture of two Gaussians each with a different mean and diagonal covariance matrix. Hint: use a step-function threshold on the first component of z to compute a binary value and use the other components of z to define the Gaussian variables.

Solution:

$$\begin{aligned} y &= \mathbf{1}[z[0] \geq 0] \\ z' &= y(\mu_1 + \sigma_1 \odot z[1:d]) + (1-y)(\mu_2 + \sigma_2 \odot z[1:d]) \end{aligned}$$

Problem 5. This problem is on GAN language modeling. A GAN takes noise as input and transforms it to an output. We consider the case where the output is a string of symbols $w_1, \ldots, w_T$ where for simplicity we always generate a string of exactly length T and where the words are integers with $w_t \in \{0, \ldots, I-1\}$ where I is the size of the vocabulary. The GAN parameters are just the parameters of a bigram model, i.e., the parameters are probability tables

$$\begin{aligned} P[i] &= P(w_1 = i) \\ Q[i,j] &= P(w_{t+1} = j \mid w_t = i) \end{aligned}$$

We take the noise input to the GAN to be a sequence of random real numbers $\epsilon_1, \ldots, \epsilon_T$ where each ϵ_t is drawn uniformly from the interval $[0,1]$.

(a) Write a function $\hat{w}(P[I], \epsilon_1)$ which deterministically returns the first word given the noise value ϵ_1 such that the probability over the draw of ϵ_1 that $\hat{w}(P[I], \epsilon_1) = i$ is $P[i]$.

Solution: We can take $\hat{w}(P[I], \epsilon_1)$ to be the unique i such that $\epsilon_1 \in \left[\left(\sum_{j<i} P[j]\right), \left(\sum_{j\leq i} P[j]\right)\right]$

(b) Write a function $\hat{w}(Q[I,I], w_t, \epsilon_t)$ which deterministically returns the word w_{t+1} given w_t such that the probability over the draw of ϵ_t that $\hat{w}(Q[I,I], w_t, \epsilon_t) = j$ is $Q[w_t, j]$.

Solution: We can take $\hat{w}(Q[I,I], w_t, \epsilon_t)$ to be the unique w_j such that $\epsilon_t \in \left[\left(\sum_{j<i} Q[w_t, j]\right), \left(\sum_{k\leq j} Q[w_t, j]\right)\right]$

(c) There is a problem with this GAN. For string generated by the GAN we need to back-propagate the discriminator loss into the GAN generator parameters. Explain why this is problematic. Is this always problematic when the generator output is discrete?

Solution: Yes, there is a problem whever s is discrete. A discrete output will not change under differential updates to the GAN parameters. Hence the gradient of the discriminator loss with respect to the generator parameters is zero. This will happen for any GAN generatng a discrete output. While there are approaches one can try for discrete GANs, GANs are most effective for modeling continuous objects like sounds and images. It does not help to have the GAN sample from a transformer model. To get a gradient on the generator parameters we need a gradient of the discriminator loss with respect to a continuous signal s being generated by the generator.