Generative model takes a sample of data drawn from an unknown distribution and learns to represent this distribution

$P \to$ unknown distribution from which the data is drawn

$Q \to$ learned distribution

The algorithm may return an explicit description of Q or it may just allow one to sample from Q. For GANs we will assume that one can sample from Q.

Basic tool of generative models: maximum likelihood

We assume we have a family of models parametrized by a set of parameters θ so we have Q_θ. If the training data are $\{x^{(i)} \mid i=1,..m\}$ Then we have a probability estimate for each choice of parameters $\prod_{i=1}^{m} Q_\theta(x^{(i)})$. We look for the parameters that maximize this probability

$$\theta^* = \arg\max_\theta \prod_{i=1}^{m} Q_\theta(x^{(i)})$$

$$= \arg\max_\theta \sum_{i=1}^{m} \log Q_\theta(x^{(i)})$$

Goodfellow et al. claim that this is equivalent to minimizing the KL divergence (relative entropy) between Q and P. Of course we don't know P but we can define an empirical distribution $\hat{P}$ from the data and

define $\theta^* = \operatorname{argmin}_\theta D_{KL}(\hat{P} \| Q_\theta)$.

One can consider other metrics as well.

There are many generative models e.g. variational autoencoders but I will stick to GANs exclusively.

Basic idea of GANs: Set up a competition between 2 learners. In practice these are neural nets. They can be thought of as players in a game: The players are called the generator and the discriminator. The generator is given training samples and learns to generate new instances from a distribution that is supposed to be a proxy for the original data distribution which is unknown. The discriminator is given samples from the real distribution and has to learn to discriminate between real samples & generated samples.

The discriminator is represented by a function D which has parameters θ & takes input x $D_\theta(x) \to \{\text{true}, \text{fake}\}$ Here x is drawn from a space of observed variables. The generator does not get x directly but uses another space Z called the latent space which is supposed to encode features of the data. The generator is represented by a function $G_\Phi(z)$ that takes $z \in Z$ as input & has Φ as the set of parameters. The players have cost functions that depend on both sets of parameters. The discriminator has a cost function $J(\theta, \Phi)$ and the generator has a cost function $L(\theta, \Phi)$. The discriminator has to minimize J while only controlling θ while the generator has to minimize L while controlling only Φ.

One can formalize all this as a game and search for a Nash equilibrium i.e. a point (θ_E, Φ_E) which is a minimum for J wrt θ and for L wrt Φ. The training process is a simultaneous SGD.

Usual cost function used for the discriminator

$$J(\theta, \Phi) = -\frac{1}{2} \mathop{\mathbb{E}}_{x \sim P} \left[\log D_\theta(x)\right] - \frac{1}{2} \mathop{\mathbb{E}}_{z} \left(1 - D_\theta(G_\Phi(z))\right).$$

The discriminator is trained on 2 minibatches coming from the real data & the fake data.

A variety of generator cost functions are used

(I) zero-sum Here we define $J = -L$

We can search for a solution satisfying

$$\Phi^* = \mathop{\mathrm{argmin}}_{\Phi} \mathop{\mathrm{argmax}}_{\theta} -J(\theta, \Phi) \ (= L(\theta, \Phi)).$$

(II) If the discriminator becomes good then the generator's gradient goes to zero and it cannot learn to improve.

$$\text{Use } L = -\frac{1}{2} \mathop{\mathbb{E}}_{z} \log D(G(z))$$

Goal is to make sure that each player has a strong gradient when losing.

There is much discussion on what divergence makes the GAN work best. Arjoosky et al proposed the Wasserstein GAN where the difference between the distributions is measured by the W_1 metric instead of KL divergence or Jensen-Shannon divergence. Later Bellemare et al. proposed using the Cramer metric after pointing out that from W_1 one did not get unbiased estimates of the gradient. The theoretical understanding of GANs remains elusive and is being actively pursued.

$$JSD(P, Q) = \frac{1}{2} KL\left(P, \frac{P+Q}{2}\right) + \frac{1}{2} KL\left(Q, \frac{P+Q}{2}\right)$$