# Meta GAN: an adversarial approach to few-shot learning

## Summary

1. A key challenge in FSC is forming decision boundaries for each task with just a few samples in each class.

2. The key idea behind meta GAN is that the imperfect generators in GAN models:
   - provide fake data b/t the manifolds of different real data class.
   - provide additional training signals to the classifier as well as making the decision boundaries much sharper.

3. The classifier is the discriminator and is also trained to classify real-vs-fake data in addition to the standard log loss.

## Notations and background

1. A task $\mathcal{T}$ from $P(\mathcal{T})$ is given by the joint dist. $P^{\mathcal{T}}_{X \times Y}(x, y)$ where the task is to predict $y$ given $x$.

2. We have a set of training tasks $\{\mathcal{T}_i\}_{i=1}^{N}$.

3. Each training sample task $\mathcal{T}$ is a tuple $\mathcal{T} = (S_{\mathcal{T}}, Q_{\mathcal{T}})$ where the
   - support set $S_{\mathcal{T}} = S^s_{\mathcal{T}} \cup S^u_{\mathcal{T}}$
   - query set $Q_{\mathcal{T}} = Q^s_{\mathcal{T}} \cup Q^u_{\mathcal{T}}$

4. The supervised support set $S^s_{\mathcal{T}}$ contains $K$ labeled examples from each of the $N$ classes.

5. The optional unlabeled support set $S^u_{\mathcal{T}} = \{x_1, \dots, x_M\}$ contains unlabeled examples from the same set of $N$ classes.

6. $Q^s_{\mathcal{T}}$ and $Q^u_{\mathcal{T}}$ are defined similarly.

## Basic Algo

1. They increase the dimension of the classifier output from $N$ to $N+1$ to model the probability that input data is fake.

2. The classifier (discriminator) is trained:
   - $\max_{D} \mathbb{E}_{\mathcal{T}} [\mathcal{L}_D^{\mathcal{T}}]$
   - $\mathcal{L}_D^{\mathcal{T}} = \mathcal{L}_{supervised} + \mathcal{L}_{unsupervised}$
   - $\mathcal{L}_{supervised} = \mathbb{E}_{x, y \sim Q_{\mathcal{T}}^{s}} [\log p_D(y \mid x, y \leq N)]$
   - $\mathcal{L}_{unsupervised} = \mathbb{E}_{x \sim Q_{\mathcal{T}}^{u}} [\log p_D(y \leq N \mid x)] + \mathbb{E}_{x \sim p_G^{\mathcal{T}}} [\log p_D(N+1 \mid x)]$

3. The generator is trained:
   - $\min_{G} \mathbb{E}_{\mathcal{T}} [\mathcal{L}_G^{\mathcal{T}}]$
   - $\mathcal{L}_G^{\mathcal{T}} = - \mathbb{E}_{x \sim p_G^{\mathcal{T}}} [\log p_D(y \leq N \mid x)]$.

where $p_G^{\mathcal{T}}$ is the task-conditioned generator.

Scanned with CamScanner