Decentralized Learning of GANs from Multi-Client Non-iid Data

Ryo Yonetani (OMRON SINIC X), Tomohiro Takahashi (OMRON), Atsushi Hashimoto (OMRON SINIC X), Yoshitaka Ushiku (OMRON SINIC X)

CVPR LONG BEACH CALIFORNIA June 16-20, 2019

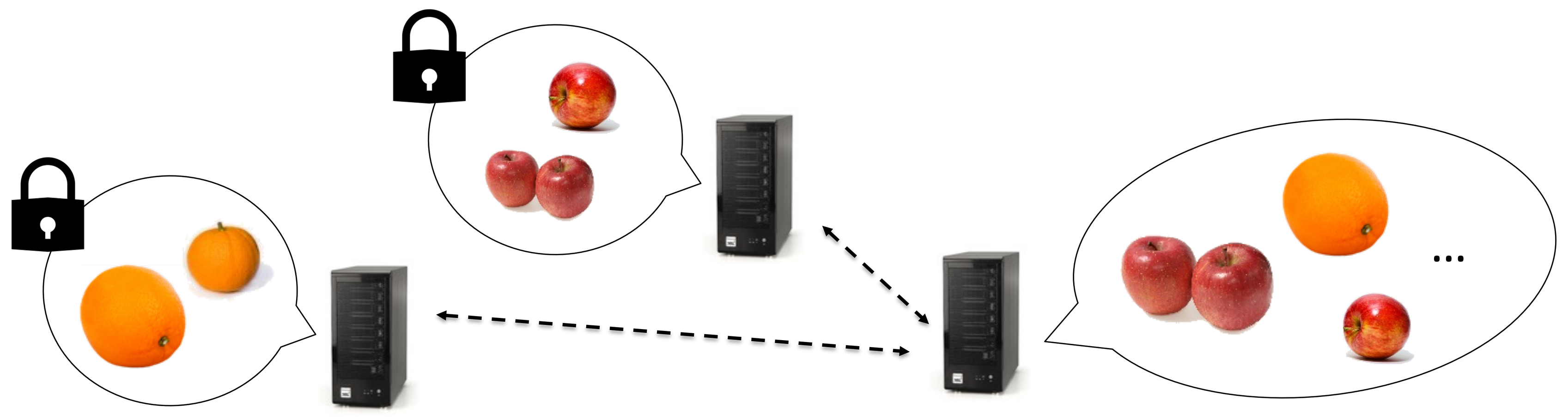
Problem Setting

Input: Multiple image collections $\{X_i \mid i = 1, \dots, N\}$ that are

- Owned separately and privately by different clients
- Drawn from non-identical distributions with different classes, $p_i(x)$

Output: Distribution comprising all the classes, $p_{max}(x) = \frac{1}{Z} \max_i p_i(x)$

s.t. X_i needs to be decentralized and private in each client storage

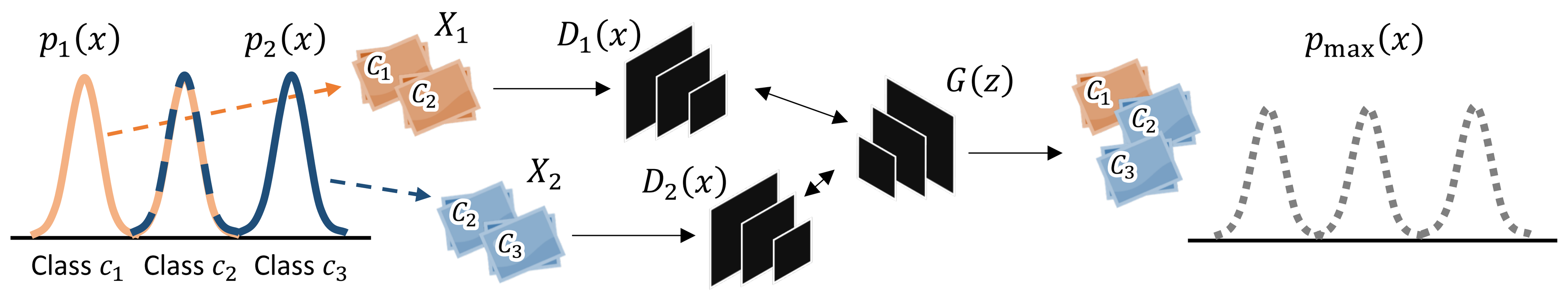


Clients:
Cooperate training in a secure fashion

Server:
Generate images of all the classes

Decentralized GANs

1. Each client trains an individual discriminator $D_i(x)$ with its own data X_i
2. Server updates generator G to fool $D_1, \dots, D_N$ to get $p_g = p_{max}$ -> *but how?*



Our Contributions

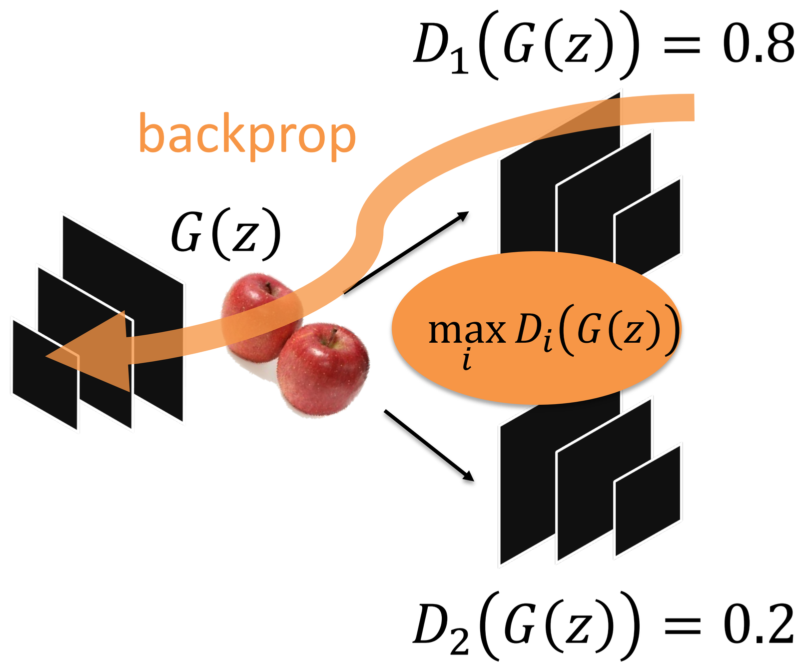
1. **Forgiver-First Update (F2U)**: update G to fool $D_{max}(x) = \max_i D_i(x)$.
   Has a theoretical guarantee to achieve $p_g = p_{max}$ as the global optimum!
2. **Forgiver-First Aggregation (F2A):** update G to fool $D_{agg}(x) = \sum_i w_i D_i(x)$ where w_i adaptively emphasizes forgiving D. Can work well in practice and involve secure-aggregation protocols to make the training provably secure

Forgiver-First Update (F2U)

Lemma 1: *When $D_i(x)$ was trained optimally from $p_i(x)$, $D^*_{max}(x) = \max_i D_i(x)$ can be regarded as the optimal discriminator trained from $p_{max}(x)$.*

Proof sketch:

- Client-wise optimal discriminator $D^*_i(x) = \frac{p_i(x)}{p_i(x)+p_g(x)}$
- $D^*_{max}(x) = \frac{\max_i p_i(x)}{\max_i p_i(x)+p_g(x)} = \frac{p_{max}(x)}{p_{max}(x)+\alpha p_g(x)}$



Theorem 2: *When G is trained against $D^*_{max}(x)$, the GAN achieves its global optimum if and only if $p_g = p_{max}$.*

Proof sketch:

- Optimizing Generator's objective for LSGAN corresponding to minimizing f-divergence

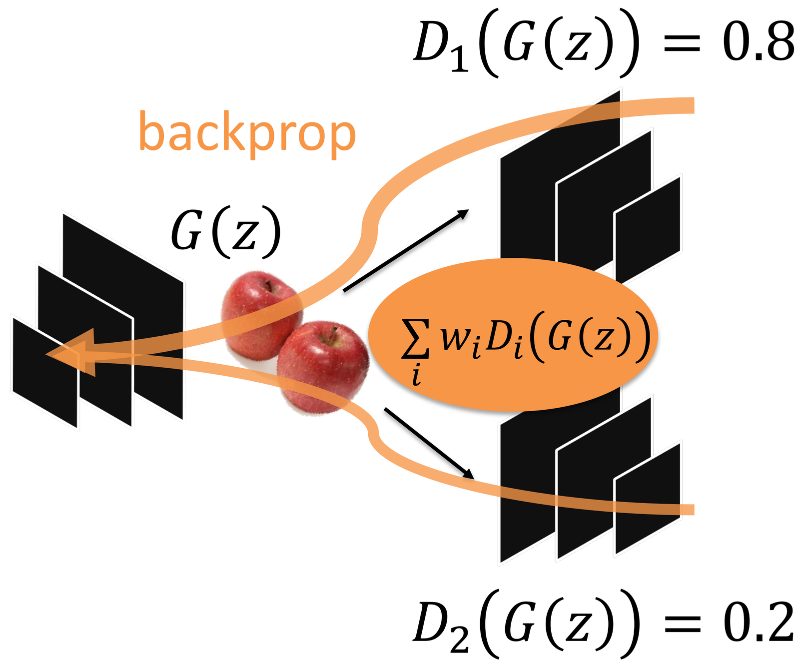
$$L(G) = \frac{1}{2}\int_x \frac{\left(p_{max}(x) + p_g(x)\right)\alpha^2 p_g^2(x)}{\left(p_{max}(x) + \alpha p_g(x)\right)^2} dx = \frac{1}{2} D_f(p_g || p_{max}) + const$$

where $f(x) = \frac{(x+1)\alpha^2 x^2}{(1+\alpha x)^2} - \frac{2\alpha^2}{(1+\alpha)^2}$ is a convex continuous function with $f(1) = 0$

Forgiver-First Aggregation (F2A)

$D_{agg}(x) = \sum_i w_i D_i(x)$ where $w_i = softmax(\lambda D_i(x))$

- λ -> large, $D_{agg}(x)$ will converge to $D_{max}(x)$.
- λ -> small, $D_{agg}(x)$ will average $D_i(x)$ equally.
- λ is updated via backprop to better fit to given data



Security consideration

- $D_i(x)$ can be used to infer if X_i contains certain x.
- $D_{agg}(x)$ and its loss gradient consist of summation over client-wise variables
- All the computations needed to update G can be wrapped by secure-aggregation that allows one to compute $\sum_i a_i$ while keeping a_i secret.

Full paper (arXiv)

Contact information

Ryo Yonetani, Ph.D.
Senior Researcher at OMRON SINIC X
ryo.yonetani@sinicx.com
https://yonetaniryo.github.io

Experimental Results

- **Data**: splitting MNIST, FMNIST, and CIFAR10 into five subsets such that X_1, X_2, X_3, X_4, X_5 contains
  {0, 1}, {2, 3}, {4, 5}, {6, 7}, {8, 9} –th classes
  (**Non-OVL**: non-overlapping condition)
  {0, 1, 2, 3}, {2, 3, 4, 5}, {4, 5, 6, 7}, {6, 7, 8, 9}, {8, 9, 0, 1}-th classes
  (**Mod-OVL**: moderately-overlapping condition)
- **Baselines**: MD-GAN, GMAN
  (both learns G from multiple Ds but with different aggregation strategies)

FID scores

	MNIST		F-MNIST		CIFAR10	
	Non-OVL	Mod-OVL	Non-OVL	Mod-OVL	Non-OVL	Mod-OVL
MD-GAN	38.42	34.33	56.09	47.12	56.64	50.30
GMAN*	67.69	58.65	56.79	49.84	50.50	41.83
F2U (Ours)	22.19	**13.38**	43.67	32.65	66.43	**40.42**
F2A (Ours)	**18.96**	14.53	**37.16**	**29.03**	**38.92**	41.01

Related Work

- **Distributed SGD:** mostly assumes iid data and supervised learning
- **Federated learning:** assumes non-iid data but still with supervised setting
- **GANs with multiple Ds and/or Ds**: mainly for stabilizing training, modeling multi-domain data, etc, not for decentralized learning

Future Directions

- More practical settings: conditional, multiple Gs, etc.
- Other adversarial learning: GAIL, adversarial domain adaptation, etc.