Generative Adversarial Minority Oversampling

Presenter: Renato Nascimento

Sankha Subhra Mullick
Shounak Datta
Swagatam Das

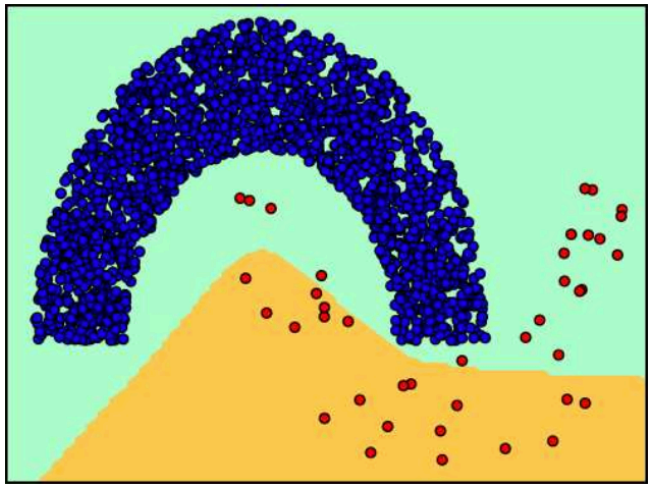
ICCV 2019

Outline

- Introduction
- Generative Adversarial Networks
- Related Works
- Proposed Method
- Experiments and Results

Introduction

- Class imbalance
  - Real-life situations – distribution of examples is skewed – some classes appear more frequently

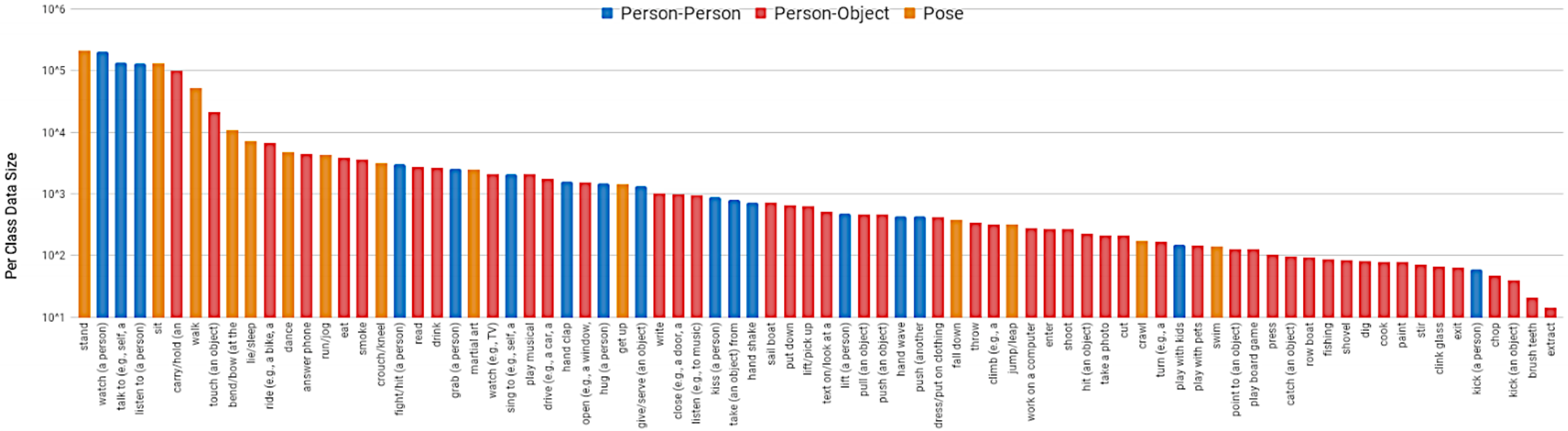


- **Majority class**
- **Minority class**

  - Most learning algorithms produce inductive bias favoring the majority class

Introduction

- Class imbalance
  - AVA: A Video Dataset of Spatio-temporally Localized Atomic Visual Actions



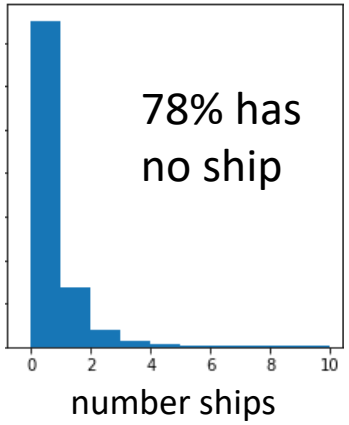
Sizes action class in the dataset, with colors indicating action types

Introduction

- Class imbalance - Real world applications:

Airbus Ship Detection





Satellite Image Classification and Segmentation with Transfer Learning

R. G. Nascimento, F. Viana, AIAA Scitech 2020

Wind turbine inspections



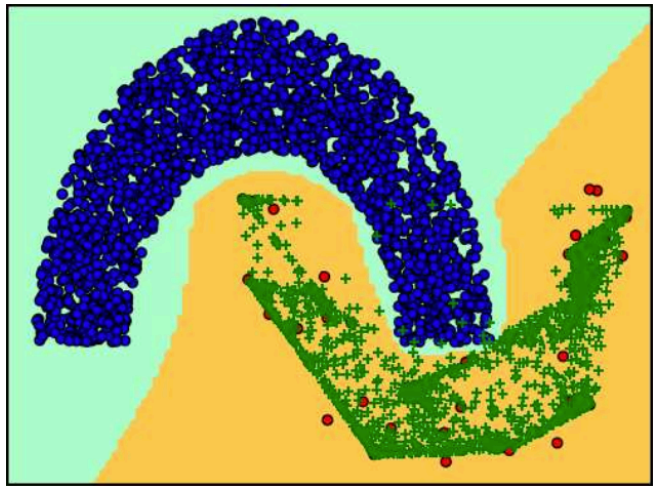


Images out of wind turbine gearbox borescope inspections

[https://onyxinsight.com/2017/03/15/damage-wind-turbine-gearbox-i-looking]

Introduction

- Handling class imbalance
  - Cost tuning (higher costs to minority)
  - Under sampling
  - Oversampling techniques
    - can not be directly applied to end-to-end deep learning systems

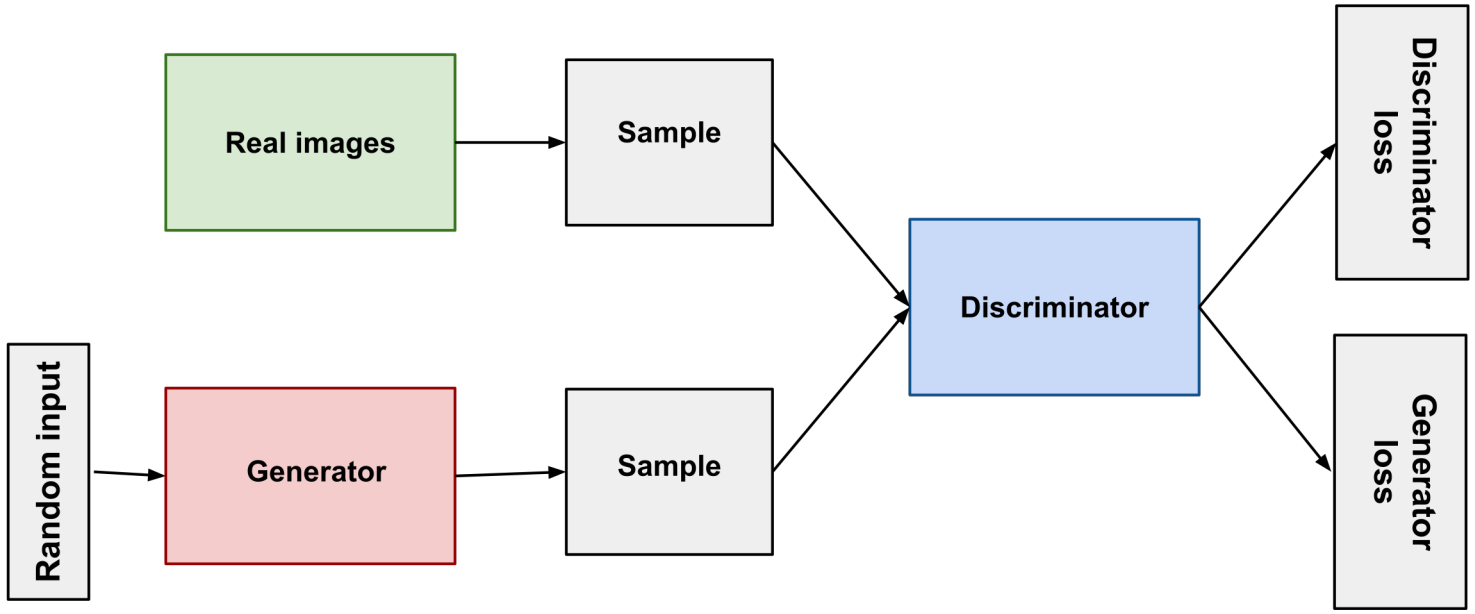


- **Majority class**
- **Minority class**
- **Artificial minority points**

  - Generative Adversarial Oversampling (end-to-end technique)

Generative Adversarial Networks

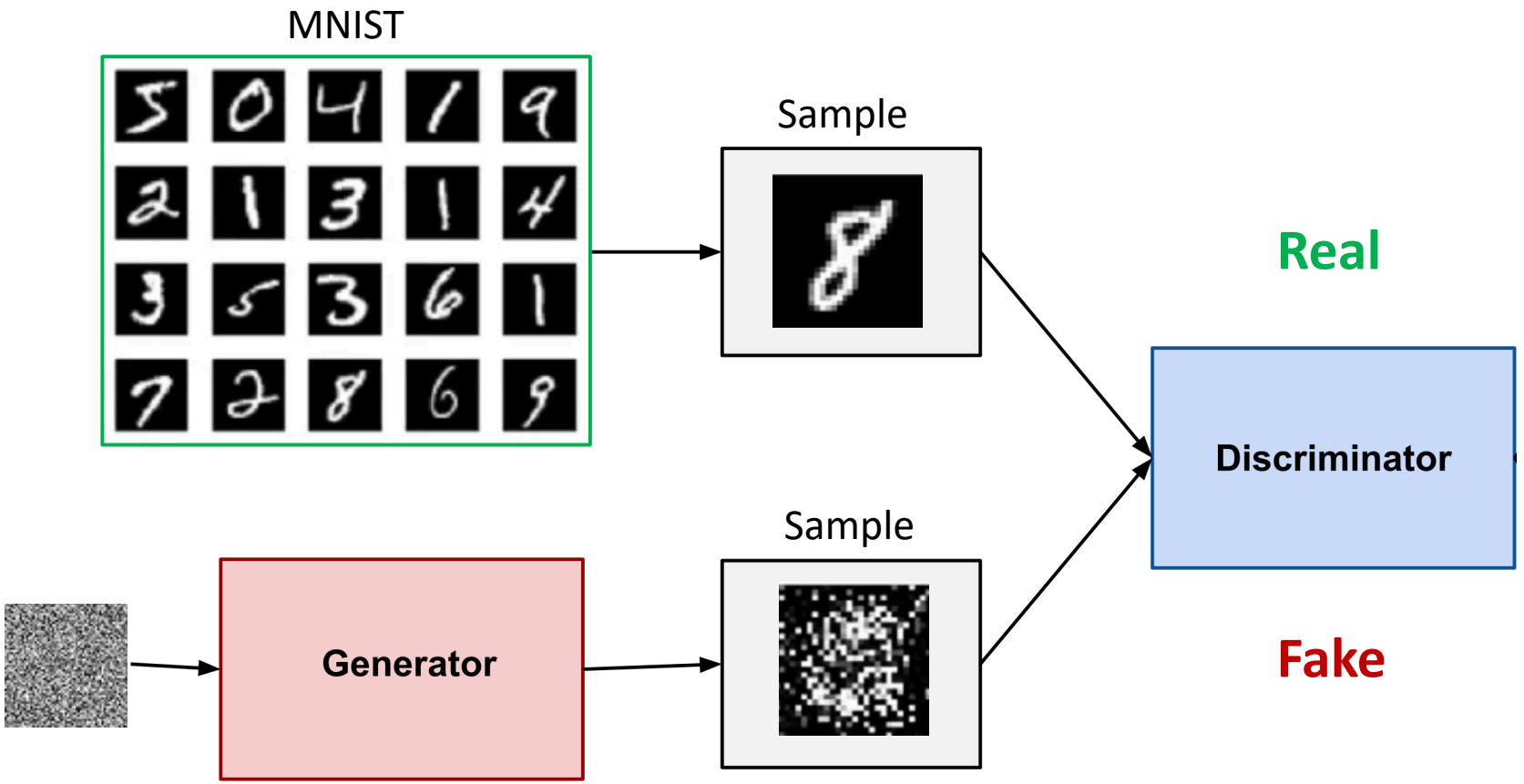
Generative Adversarial Nets, Goodfellow et al. in 2014



source: developers.google.com

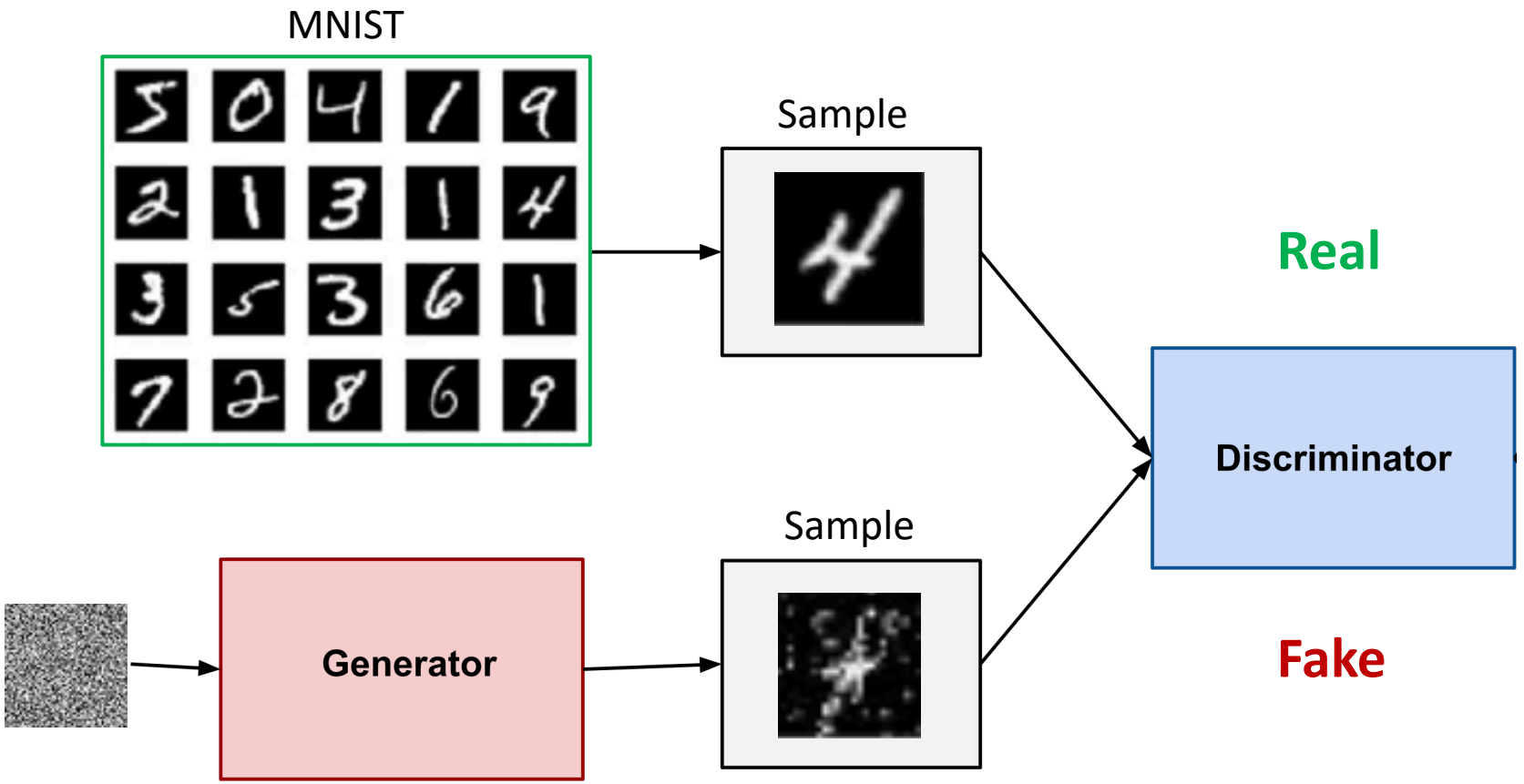
Two player adversarial game

Generative Adversarial Networks



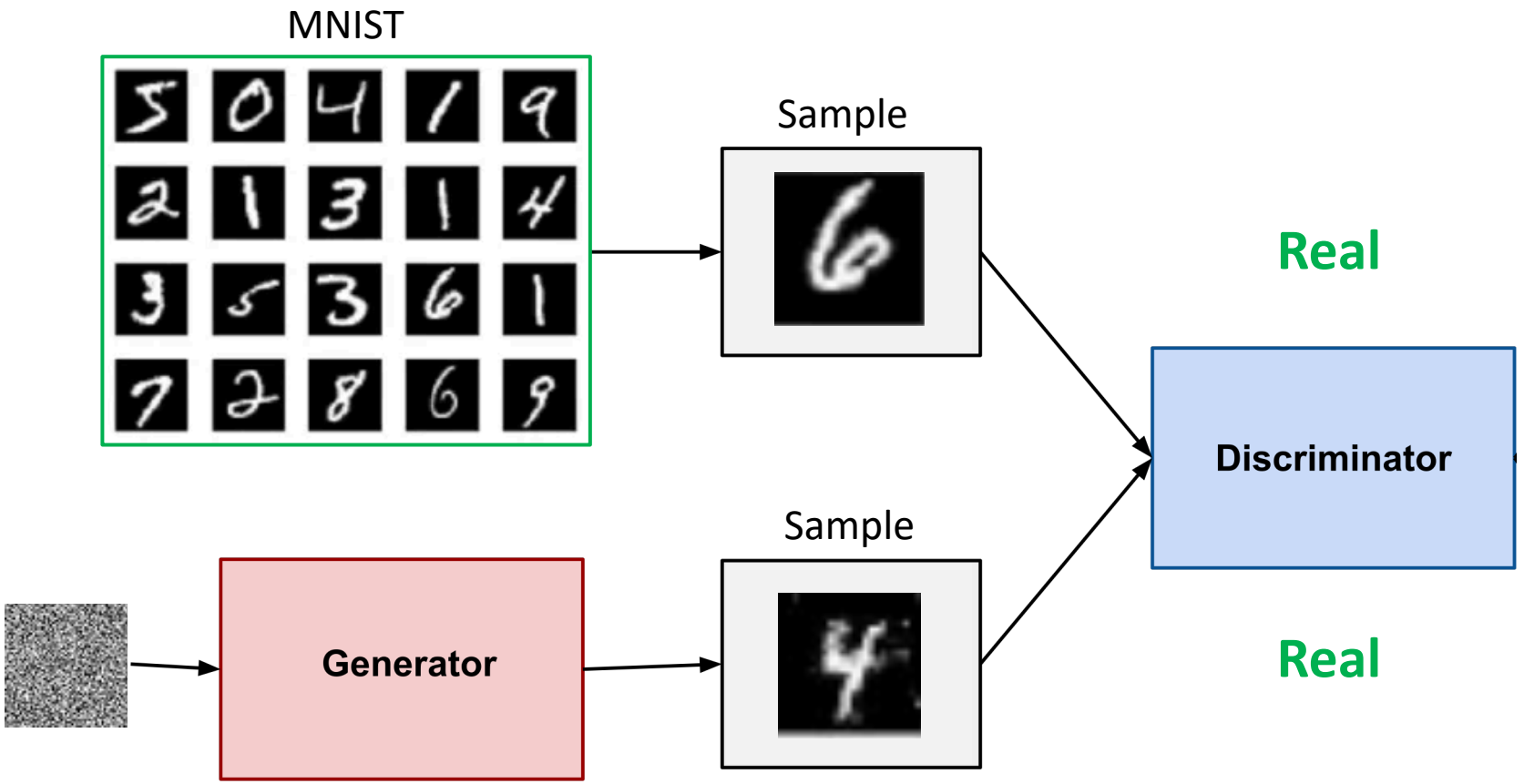
adapted from: developers.google.com

Generative Adversarial Networks



adapted from: developers.google.com

Generative Adversarial Networks

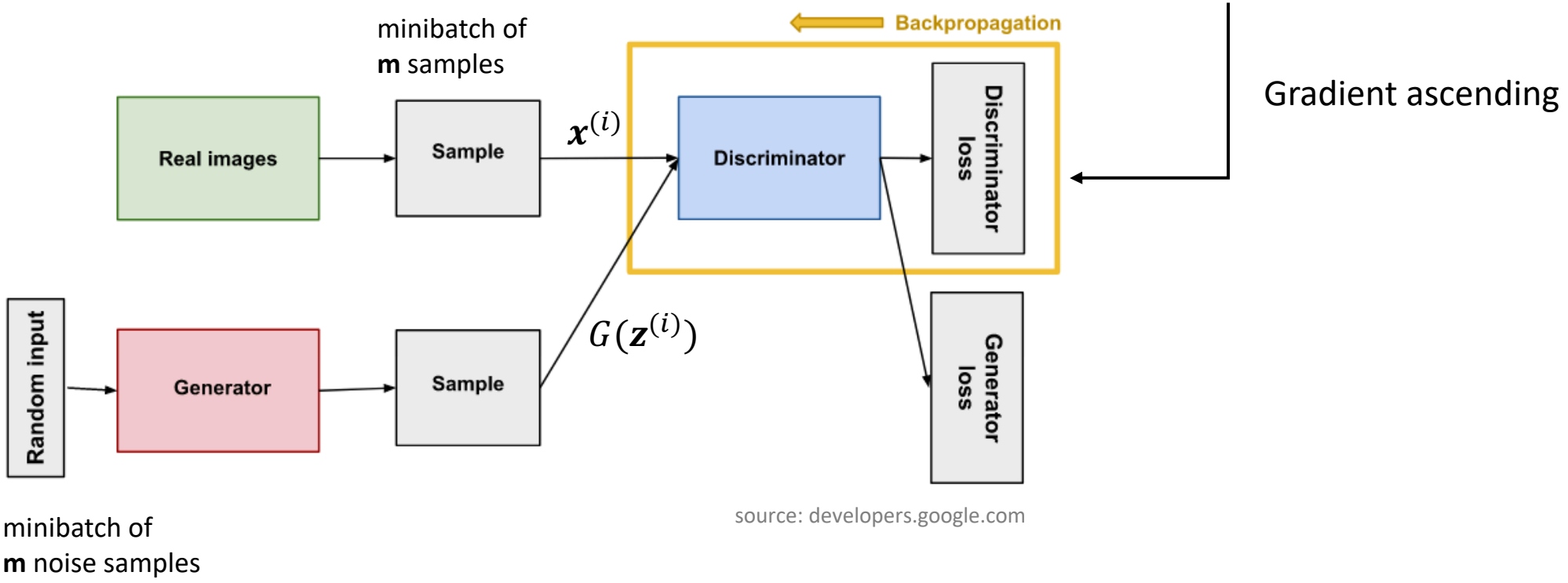


adapted from: developers.google.com

Generative Adversarial Networks

Minimax Loss

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right) \right]$$

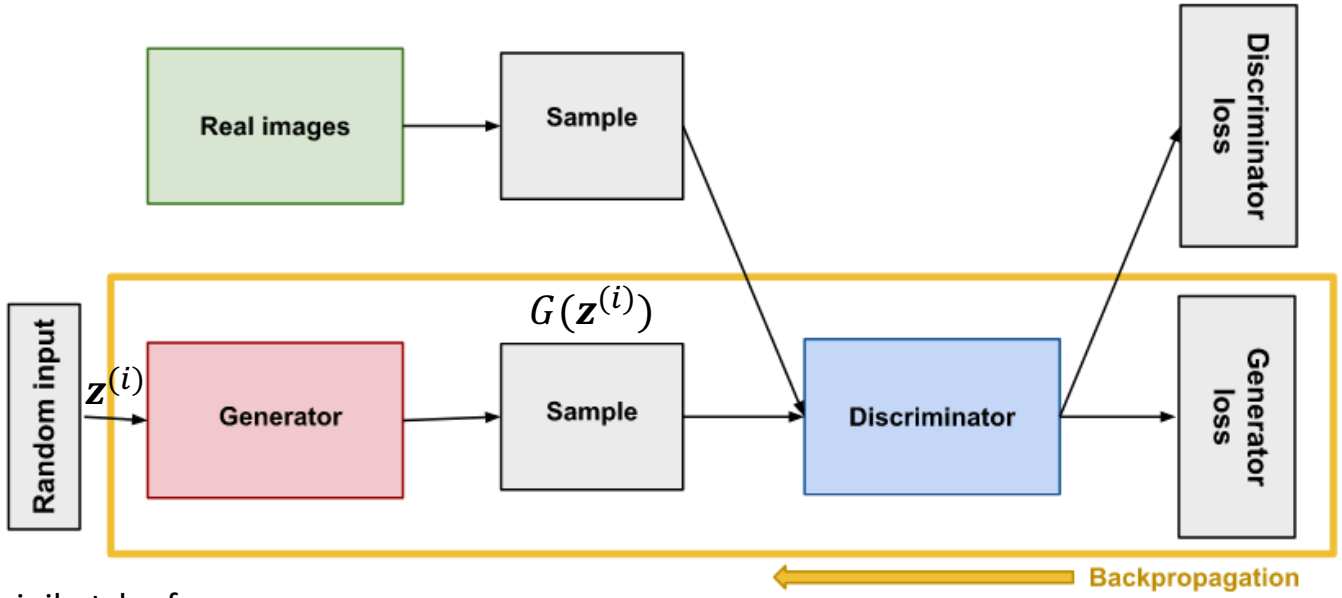


source: developers.google.com

Generative Adversarial Networks

Minimax Loss

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right)$$



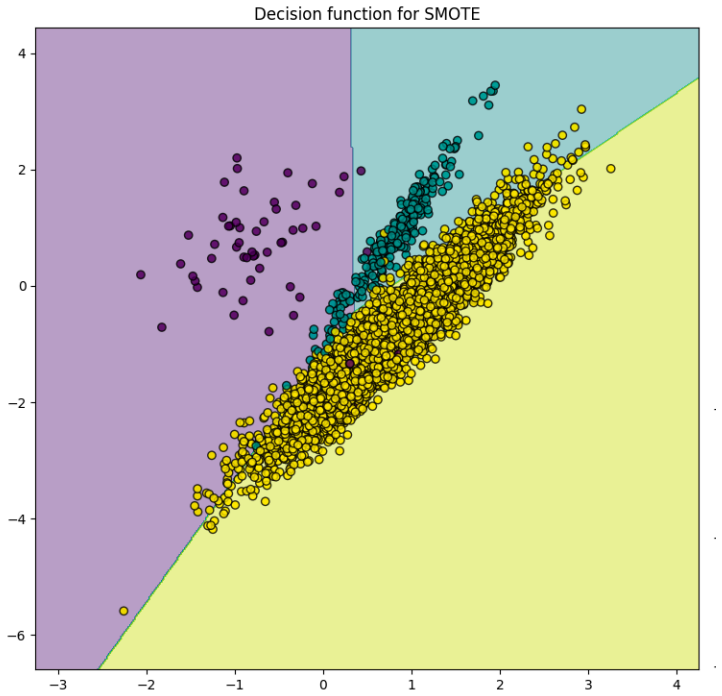
Gradient descending

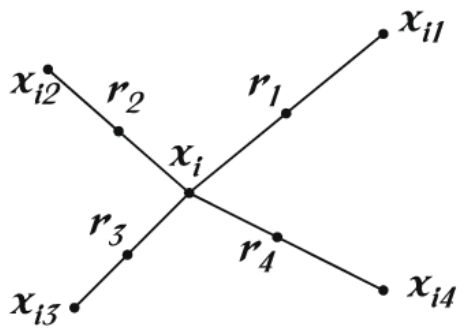
minibatch of
m noise samples

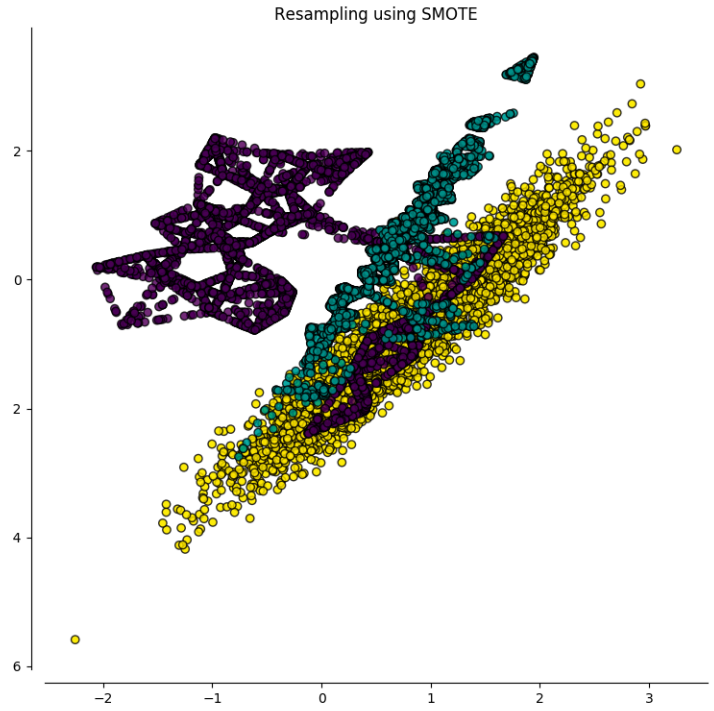
source: developers.google.com

Related Works

- SMOTE: Synthetic Minority Over-sampling Technique (2002)







$\mathbf{x}^R$ is randomly chosen among the k-nearest neighbors of $\mathbf{x}$

source: imbalanced-learn.readthedocs.io

Related Works

- Effective data generation for imbalanced learning using conditional generative adversarial networks (cGAN) (2017)

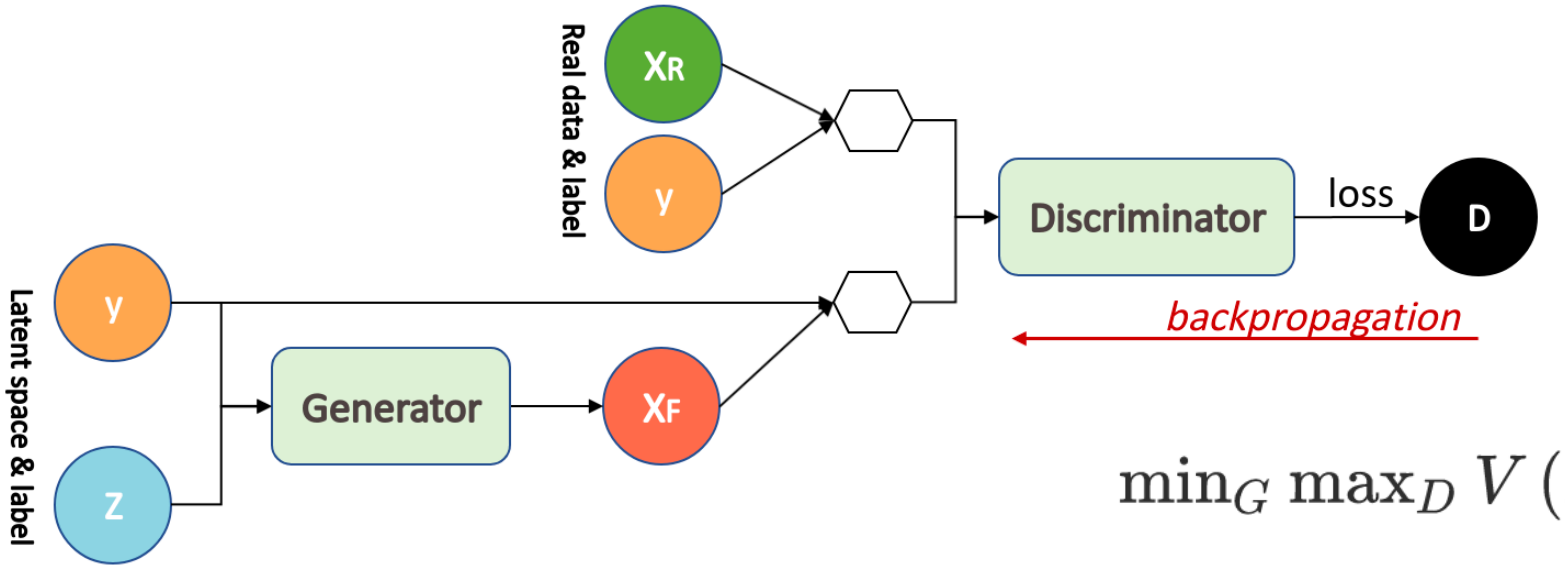


Fig. source: Mohammad Ali Bagheri

Extension of GAN (including additional space Y)

$$\min_G \max_D V(D,G) = E_D + E_G$$

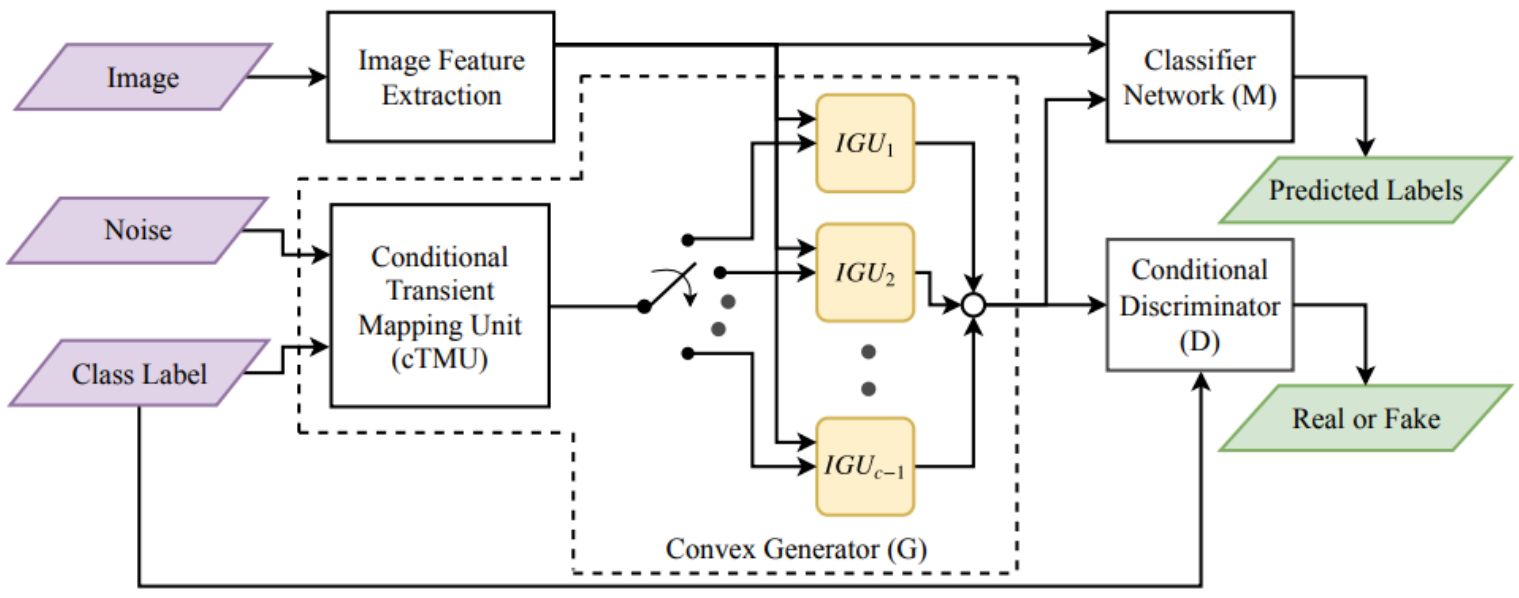
$$E_D = E_{x,y\sim p_{data}(x,y)}\left[logD(x,y)\right]$$

$$E_G = E_{z\sim p_z(z),y\sim p(y)}\left[\log\left(1 - D\left(G\left(z,y\right),y\right)\right)\right]$$

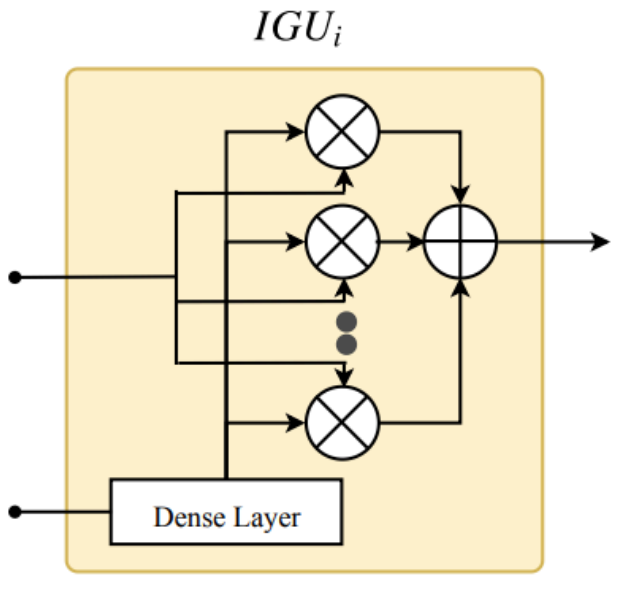
Proposed Method

- Three-player adversarial game
  - convex generator **G**
  - classifier network **M**
  - a discriminator **D**

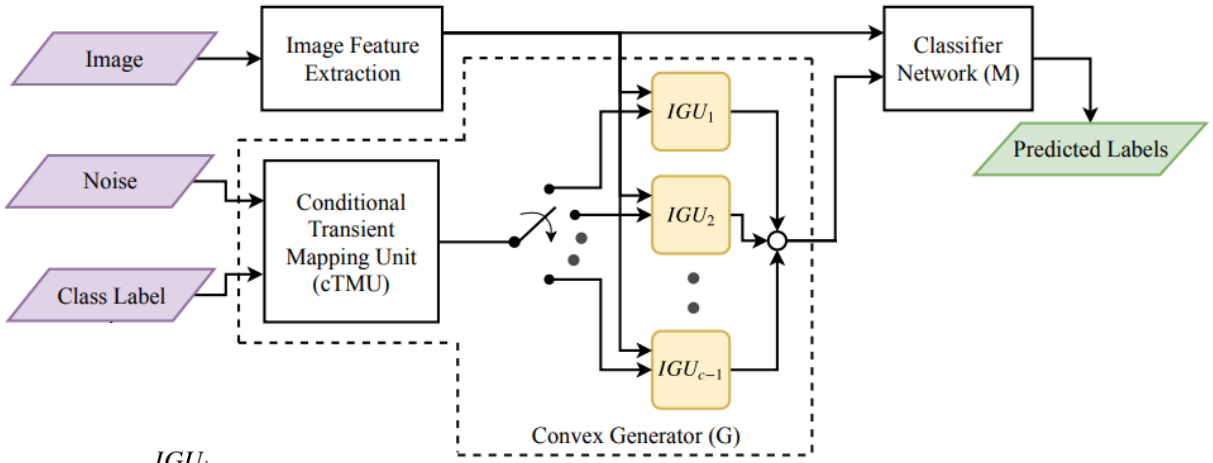
Generator G attempts to fool both M and D

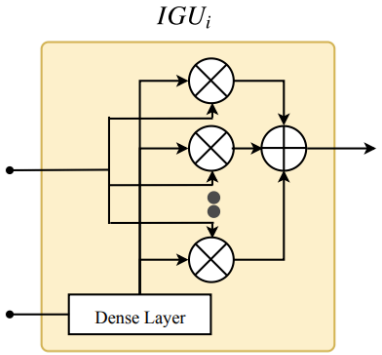


Instance Generation Unit



Convex Generator





Generate the new points only as convex combinations of the existing points from the minority class in question

$$G(\mathbf{z}|i) = g_i(t(\mathbf{z}|i))^T X_i$$

Cross entropy loss

$$\min_G \max_M J(G, M) = \sum\nolimits_{i \in \mathcal{C}} J_i, \qquad J_i = (J_{i1} + J_{i2} + J_{i3} + J_{i4})$$

$$J_{i1} = P_i \mathbb{E}_{\mathbf{x} \sim p_i^d}[\log M_i(\mathbf{x})],$$

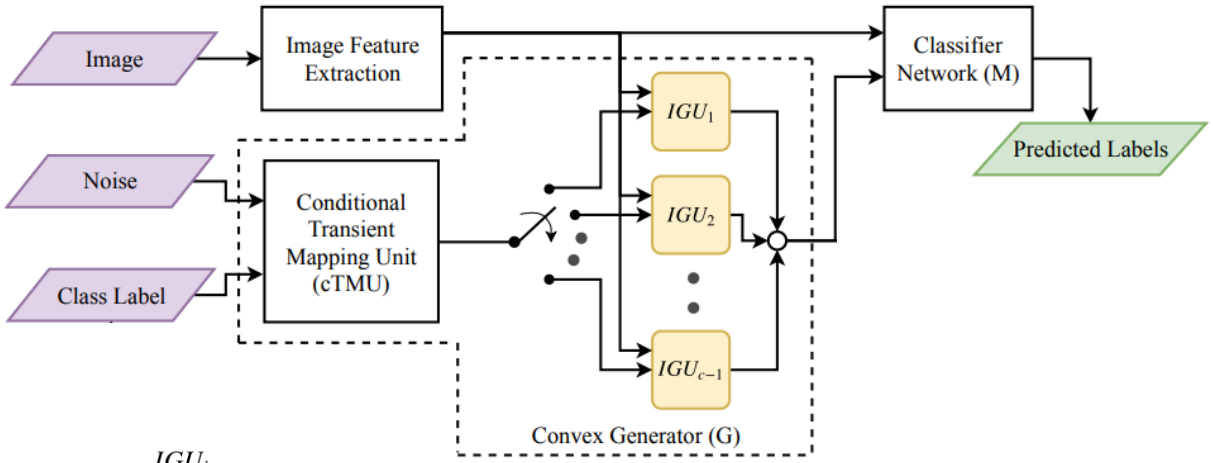
$$J_{i2} = \sum\nolimits_{j \in \mathcal{C} \setminus \{i\}} P_j \mathbb{E}_{\mathbf{x} \sim p_j^d}[\log(1 - M_i(\mathbf{x}))],$$

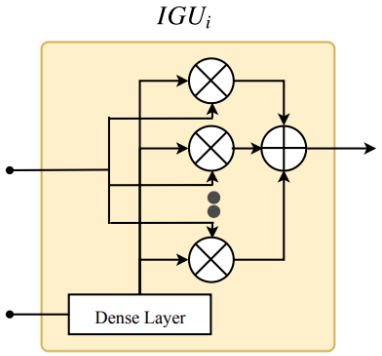
$$J_{i3} = (P_c - P_i)\mathbb{E}_{G(\mathbf{z}|i) \sim p_i^g}[\log M_i(G(\mathbf{z}|i))], \text{and},$$

$$J_{i4} = \sum_{j \in \mathcal{C} \setminus \{i\}} (P_c - P_j)\mathbb{E}_{G(\mathbf{z}|j) \sim p_j^g}[\log(1 - M_i(G(\mathbf{z}|j)))],$$

p_i^d, p_i^g - real and generated class conditional probability distributions of the i-th class

Convex Generator



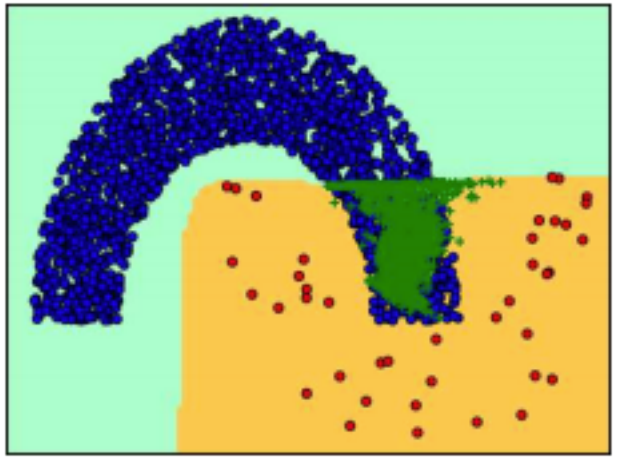


$$G(\mathbf{z}|i) = g_i(t(\mathbf{z}|i))^T X_i$$

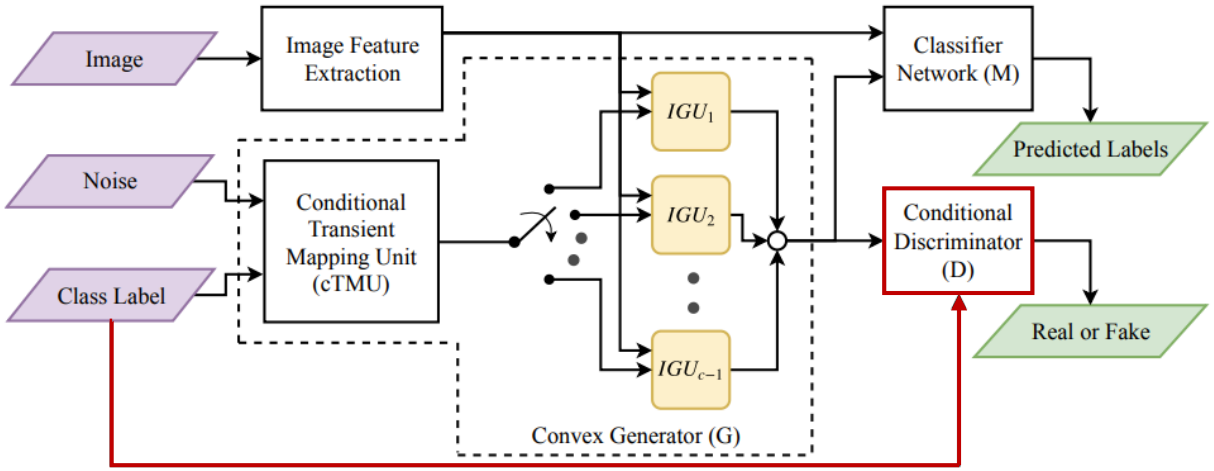
$$\min_G \max_M J(G, M) = \sum_{i \in \mathcal{C}} J_i,$$

Generates points within the convex hull of the samples from the minority

But points may still be placed at locations within the convex hull which do not correspond to the distribution of the intended class



Additional Discriminator



Discriminator ensures that the generated points do not fall outside the actual distribution of the intended minority class

Three-player minimax game

$$\min_G \max_M \max_D Q(G, M, D) = \sum_{i \in C} Q_i,$$

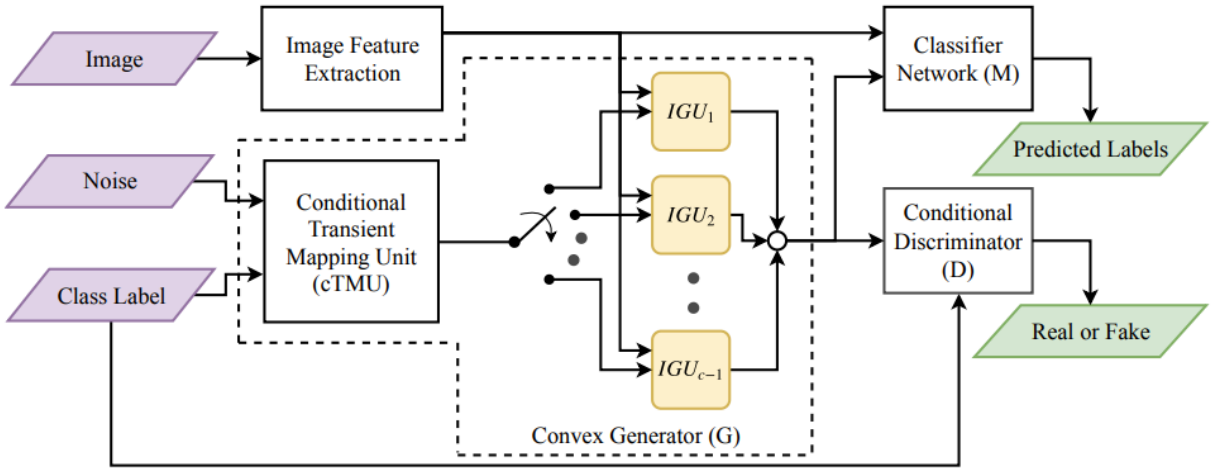
$$Q_i = (J_{i1} + J_{i2} + J_{i3} + J_{i4} + Q_{i1} + Q_{i2})$$

Cross entropy loss

$$Q_{i1} = P_i \mathbb{E}_{\mathbf{x} \sim p_i^d}[\log D(\mathbf{x}|i)], \text{and,}$$

$$Q_{i2} = (P_c - P_i)\mathbb{E}_{G(\mathbf{z}|i) \sim p_i^g}[\log(1 - D(G(\mathbf{z}|i)|i))]$$

Least-Square Loss

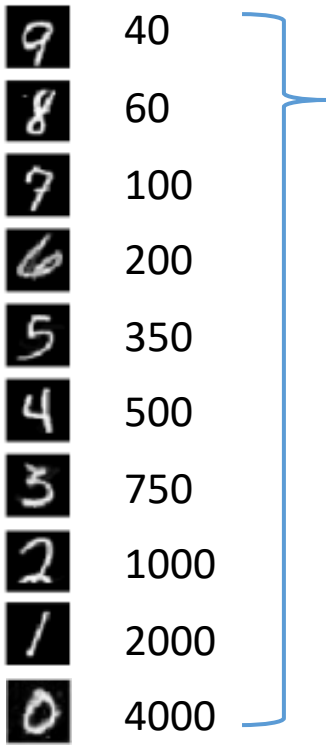


$$\min_M L_M = \sum_{i\in\mathcal{C}}(L_{i1}+L_{i2}+L_{i3}+L_{i4}),$$
$$\min_D L_D = \sum_{i\in\mathcal{C}}(L_{i5}+L_{i6}),$$
$$\min_G L_G = \sum_{i\in\mathcal{C}\setminus\{c\}}(L_{i7}+L_{i8}+L_{i9}),$$

$$L_{i1} = P_i\mathbb{E}_{\mathbf{x}\sim p_i^d}[(1-M_i(\mathbf{x}))^2],$$
$$L_{i2} = \sum_{j\in\mathcal{C}\setminus\{i\}} P_j\mathbb{E}_{\mathbf{x}\sim p_j^d}[(M_i(\mathbf{x}))^2],$$
$$L_{i3} = (P_c-P_i)\mathbb{E}_{G(\mathbf{z}|i)\sim p_i^g}[(1-M_i(G(\mathbf{z}|i)))^2],$$
$$L_{i4} = \sum_{j\in\mathcal{C}\setminus\{i\}}(P_c-P_j)\mathbb{E}_{G(\mathbf{z}|j)\sim p_j^g}[(M_i(G(\mathbf{z}|j)))^2],$$
$$L_{i5} = P_i\mathbb{E}_{\mathbf{x}\sim p_i^d}[(1-D(\mathbf{x}|i))^2],$$
$$L_{i6} = (P_c-P_i)\mathbb{E}_{G(\mathbf{z}|i)\sim p_i^g}[(D(G(\mathbf{z}|i)|i))^2],$$
$$L_{i7} = \mathbb{E}_{G(\mathbf{z}|i)\sim p_i^g}[(M_i(G(\mathbf{z}|i)))^2],$$
$$L_{i8} = \sum_{j\in\mathcal{C}\setminus\{i,c\}}\mathbb{E}_{G(\mathbf{z}|j)\sim p_j^g}[(1-M_i(G(\mathbf{z}|j)))^2],$$
$$L_{i9} = \mathbb{E}_{G(\mathbf{z}|i)\sim p_i^g}[(1-D(G(\mathbf{z}|i)|i))^2].$$

Datasets

- MNIST and Fashion-MNIST
  - 28×28 grayscale, 10 classes



9	40
8	60
7	100
6	200
5	350
4	500
3	750
2	1000
1	2000
0	4000

Subset

Imbalance Ratio: 100
(largest/smallest)

Evaluation Metrics

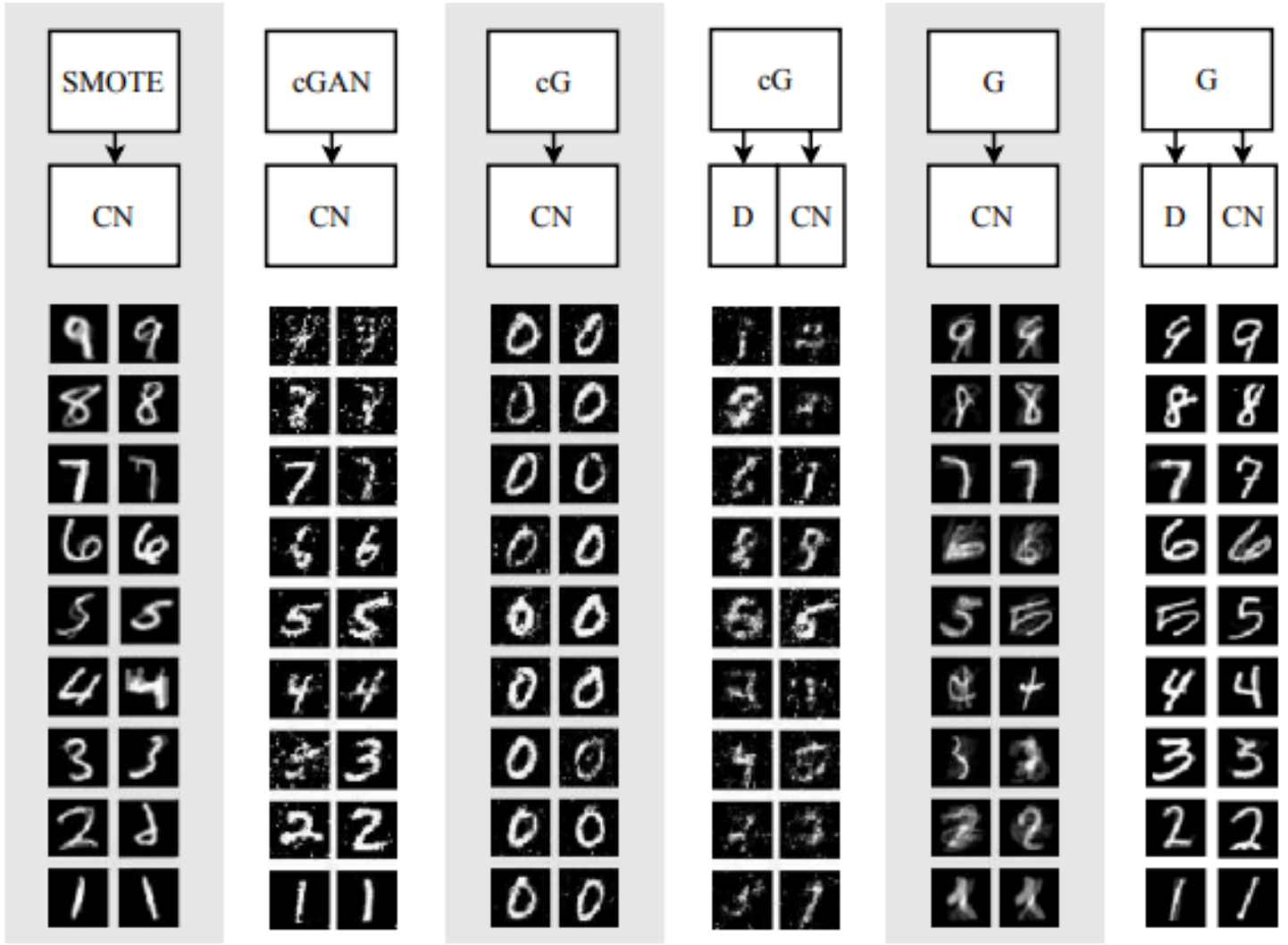
Average Class Specific Accuracy (ACSA)
Geometric Mean (GM)

Fashion-MNIST sample



Experiments and Results

MNIST and Fashion-MNIST



Experiments and Results

MNIST and Fashion-MNIST

Algorithm	MNIST				Fashion-MNIST			
	CE		LS		CE		LS	
	ACSA	GM	ACSA	GM	ACSA	GM	ACSA	GM
Baseline CN	0.88±0.01	0.87±0.02	0.88±0.01	0.86±0.01	0.82±0.01	0.80±0.01	0.81±0.01	0.79±0.01
SMOTE+CN	0.88±0.02	0.87±0.03	0.89±0.01	0.89±0.01	0.82±0.01	0.80±0.02	0.80±0.01	0.77±0.01
Oversample+CN	-	-	-	-	0.81±0.01	0.79±0.01	0.81±0.01	0.79±0.01
Augment+CN	-	-	-	-	0.82±0.01	0.78±0.01	0.82±0.01	0.78±0.01
DOS	-	-	-	-	0.82±0.01	0.79±0.01	0.81±0.01	0.79±0.02
cGAN / cDCGAN+CN	0.88±0.01	0.87±0.01	0.89±0.01	0.88±0.01	0.81±0.02	0.78±0.01	0.82±0.01	0.80±0.01
cG+CN	0.86±0.03	0.85±0.02	0.86±0.03	0.85±0.03	0.79±0.02	0.77±0.02	0.80±0.01	0.77±0.02
cG+D+CN	0.85±0.02	0.83±0.01	0.85±0.02	0.82±0.02	0.79±0.02	0.78±0.01	0.79±0.01	0.78±0.02
GAMO\D (Ours)	0.87±0.01	0.86±0.01	0.88±0.01	0.87±0.01	0.81±0.01	0.80±0.01	0.82±0.01	0.80±0.01
GAMO (Ours)	0.89±0.01	0.88±0.01	**0.91±0.01**	**0.90±0.01**	0.82±0.01	0.80±0.01	**0.83±0.01**	**0.81±0.01**

Average Class Specific Accuracy (ACSA)
Geometric Mean (GM)

Datasets

- Cifar10 and SVHN
  - 32×32 RGB, 10 classes

Label Index	
9	80
8	150
7	250
6	400
5	500
4	600
3	800
2	2000
1	4500
0	

Subset

Imbalance Ratio: 56.25
(largest/smallest)

Evaluation Metrics

Average Class Specific Accuracy (ACSA)
Geometric Mean (GM)

Cifar10 Sample



SVHN Sample



Experiments and Results

CIFAR10 and SVHN – Subset Imbalance Ratio of 56.25 (largest to smallest class)

Algorithm	CIFAR10		SVHN	
	ACSA	GM	ACSA	GM
Baseline CN	0.45±0.01	0.37±0.01	0.74±0.01	0.73±0.01
SMOTE+CN	0.46±0.02	0.4±0.02	0.75±0.01	0.73±0.02
Oversample+CN	0.44±0.02	0.37±0.03	0.74±0.02	0.73±0.02
Augment+CN	0.47±0.01	0.39±0.02	0.69±0.01	0.63±0.01
cDCGAN+CN	0.42±0.02	0.32±0.03	0.69±0.01	0.66±0.02
DOS	0.46±0.02	0.37±0.01	0.71±0.02	0.68±0.01
GAMO\D (Ours)	0.47±0.01	0.40±0.01	0.75±0.01	0.75±0.02
GAMO (Ours)	**0.49±0.01**	**0.43±0.02**	**0.76±0.01**	**0.75±0.02**

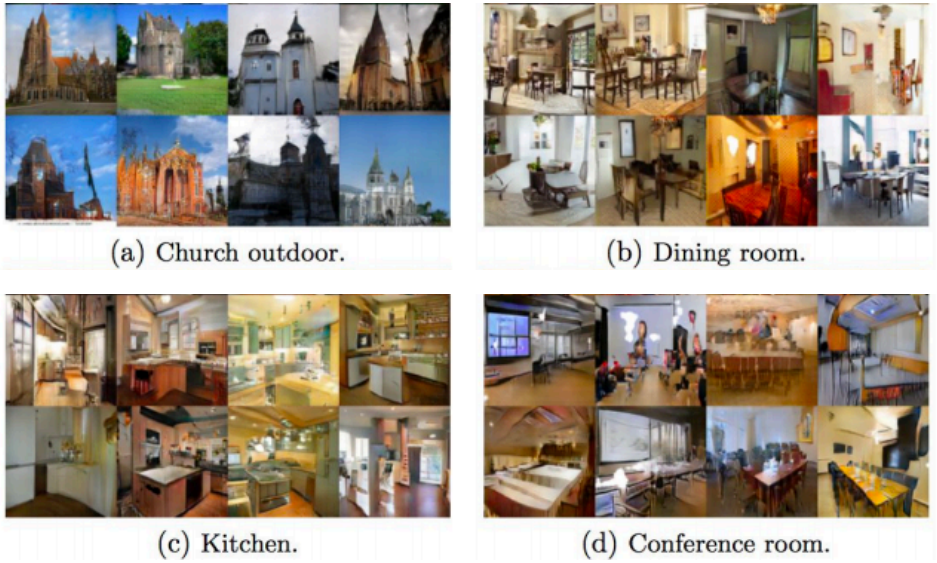
Average Class Specific Accuracy (ACSA)
Geometric Mean (GM)

Datasets - higher resolution

- CelebA - 64×64 RBG, 5 classes
  - Small subset
    - 15000, 1500, 750, 300, and 150
    - Imbalance Ratio: 100
  - Large subset
    - 28000, 4000, 3000, 1500, and 750
    - Imbalance Ratio: 37.33



- LSUN - 224×224 RGB, 5 classes
  - Small subset
    - 15000, 1500, 750, 300, and 150
    - Imbalance Ratio: 100
  - Large subset
    - 50000, 5000, 3000, 1500, and 750
    - Imbalance Ratio: 66.67



Experiments and Results

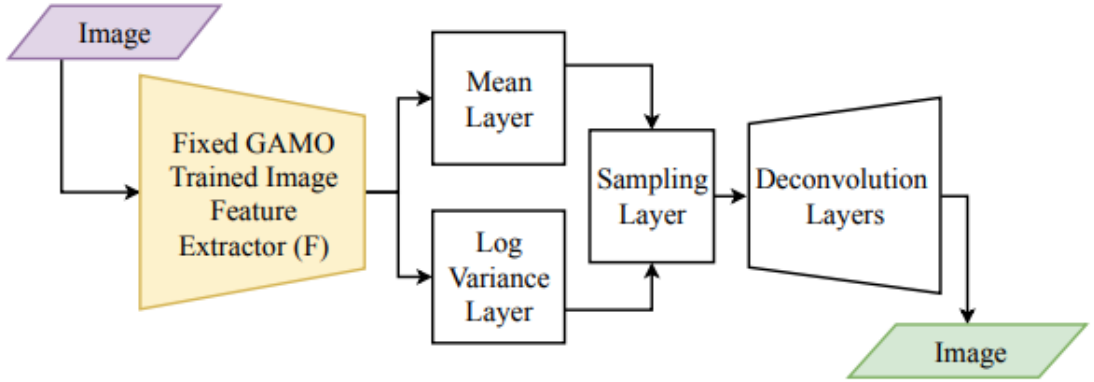
CelebA (64x64): Small set (IR 100), Large set (IR 37.33) and **LSUN** (224x224): Small set (IR 100), Large set (IR 66.67)

Algorithm	CelebA-Small				CelebA-Large			
	During Training		During Testing		During Training		During Testing	
	ACSA	GM	ACSA	GM	ACSA	GM	ACSA	GM
Baseline CN	0.91±0.01	0.91±0.01	0.59±0.01	0.45±0.04	0.93±0.01	0.92±0.01	0.71±0.01	0.60±0.03
SMOTE+CN	0.99±0.01	0.99±0.01	0.62±0.02	0.48±0.03	0.99±0.01	0.99±0.01	0.71±0.02	0.66±0.03
Oversample+CN	0.99±0.01	0.99±0.01	0.59±0.03	0.39±0.04	0.98±0.01	0.98±0.02	0.68±0.02	0.62±0.02
Augment+CN	0.74±0.06	0.70±0.09	0.62±0.05	0.47±0.08	0.82±0.01	0.79±0.01	0.72±0.01	0.66±0.02
cDCGAN+CN	0.86±0.01	0.84±0.01	0.59±0.01	0.36±0.02	0.87±0.01	0.86±0.01	0.67±0.01	0.58±0.02
DOS	0.82±0.03	0.80±0.02	0.61±0.01	0.48±0.02	0.84±0.01	0.83±0.02	0.72±0.01	0.64±0.02
GAMO (Ours)	0.92±0.01	0.91±0.01	**0.66±0.01**	**0.54±0.02**	0.91±0.01	0.91±0.01	**0.75±0.01**	**0.70±0.02**
	LSUN-Small				LSUN-Large			
	During Training		During Testing		During Training		During Testing	
	ACSA	GM	ACSA	GM	ACSA	GM	ACSA	GM
Baseline CN	0.90±0.01	0.89±0.01	0.50±0.01	0.28±0.05	0.87±0.01	0.87±0.01	0.61±0.02	0.54±0.03
SMOTE+CN	0.99±0.02	0.99±0.02	0.50±0.01	0.40±0.02	0.98±0.01	0.98±0.01	0.66±0.03	0.62±0.03
Oversample+CN	0.99±0.01	0.99±0.01	0.52±0.01	0.43±0.02	0.98±0.01	0.98±0.01	0.62±0.03	0.58±0.03
Augment+CN	0.67±0.06	0.64±0.09	0.54±0.03	0.45±0.07	0.70±0.03	0.65±0.03	0.64±0.02	0.58±0.03
cDCGAN+CN	0.80±0.02	0.79±0.02	0.53±0.02	0.43±0.03	0.81±0.02	0.80±0.02	0.60±0.02	0.53±0.03
DOS	0.78±0.03	0.76±0.02	0.54±0.02	0.44±0.02	0.79±0.02	0.77±0.02	0.63±0.02	0.61±0.03
GAMO (Ours)	0.93±0.01	0.93±0.01	**0.57±0.01**	**0.50±0.02**	0.80±0.01	0.80±0.01	**0.70±0.02**	**0.68±0.03**

GAMO2pix

CIFAR10, Fashion-MNIST, SVHN, and CelebA-Small

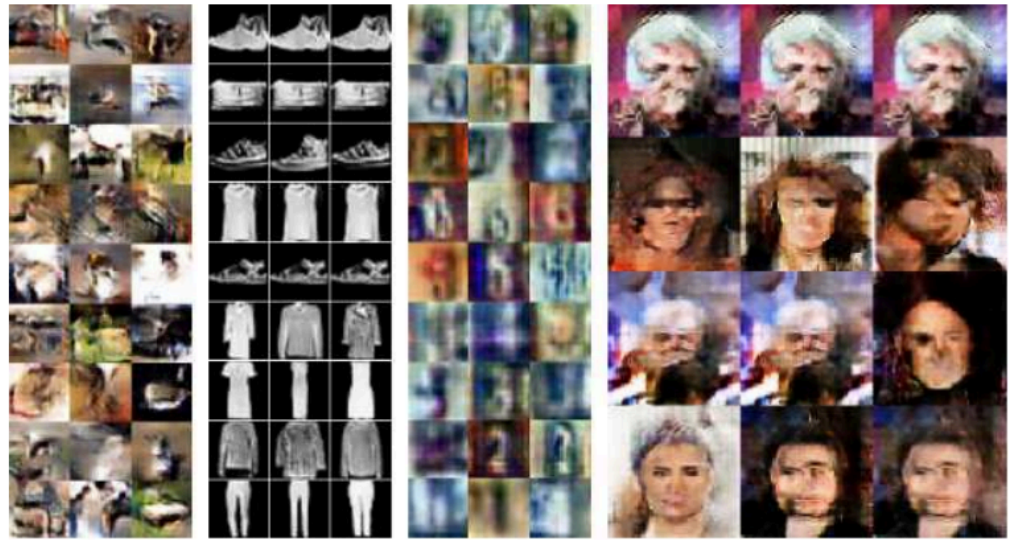
Some application may require that actual samples be generated by oversampling to form an artificially balanced dataset



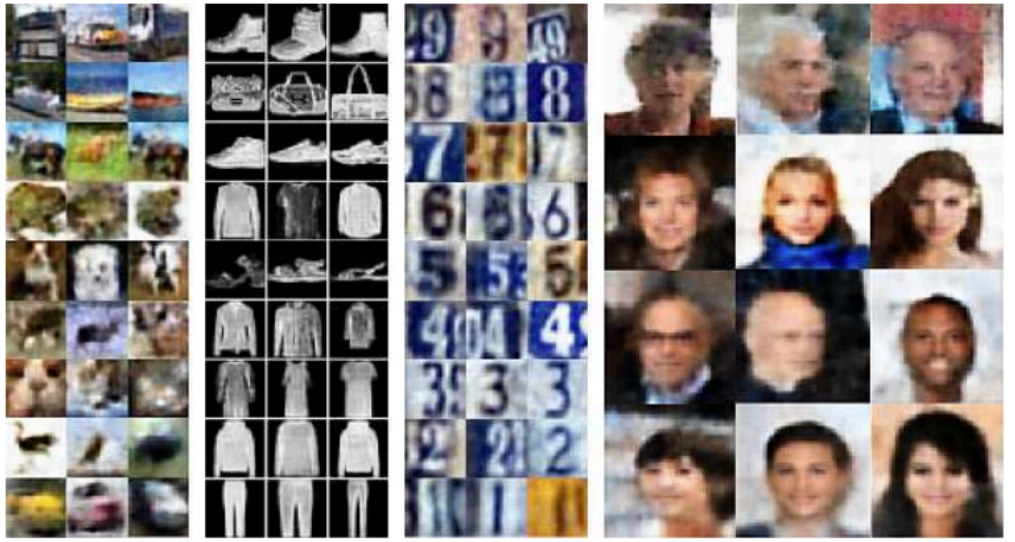
Fréchet Inception Distance (FID)

Dataset	GAMO2pix (Ours)	cDCGAN
Fashion-MNIST	**0.75±0.03**	5.57±0.03
SVHN	**0.17±0.02**	0.59±0.04
CIFAR10	**1.59±0.03**	2.96±0.03
CelebA-Small	**11.13±0.04**	15.12±0.05

cDCGAN



GAMO2pix



Questions?

Generative Adversarial Minority Oversampling

Sankha Subhra Mullick
Shounak Datta
Swagatam Das

ICCV 2019



github.com/SankhaSubhra/GAMO