OCGAN: One-class Novelty Detection Using GANs with Constrained Latent Representations

Pramuditha Perera, Ramesh Nallapati, Bing Xiang
Johns Hopkins University, AWS AI
CVPR2019

Outline

- Introduction
- Method
  - Motivation
  - Proposed Strategy
- Experiments
- Conclusions
- Progress Report

Outline

- Introduction
- Method
  - Motivation
  - Proposed Strategy
- Experiments
- Conclusions
- Progress Report

Introduction

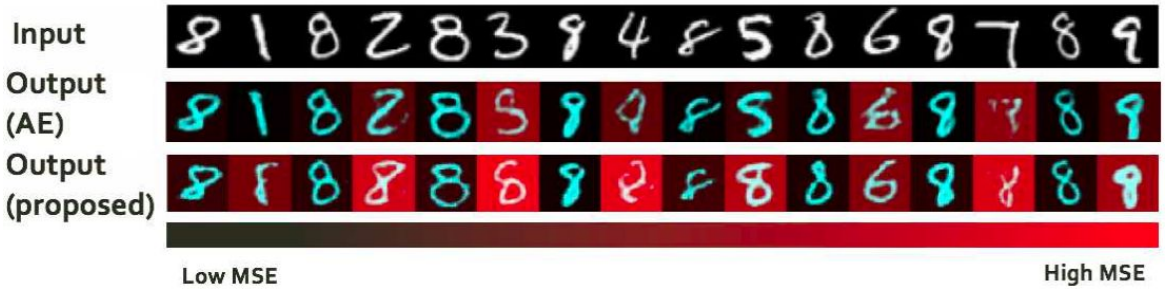
- One-class novelty detection
  - Quantifying the probability that a test example belongs to the distribution defined by training examples
  - Only **a single class** are observed at training time
  - The trained model is expected to accept in-class examples and reject out-of-class examples
- Applications
  - Abnormality detection
  - Intruder detection
  - Bio-medical data processing
  - Imbalance learning

Introduction

- Contemporary works in one-class novelty detection
  - Focus on learning a representative latent space for the given class
    - Novelty detection is performed based on the learned latent space
  - The difference between the query image and its inverse image (**reconstruction**) is used as a novelty detector
- The existing work assumed that when an out-of-class object is presented to the network
  - It will do a poor job of describing the object
  - Thereby reporting a relatively higher reconstruction error

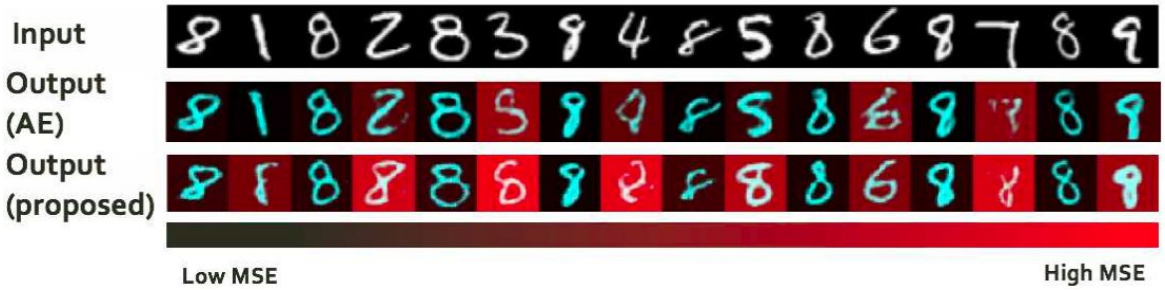
Introduction

- However, this assumption does not hold at all times
  - For an example the auto-encoders trained on digits with complex shapes, such as digit 8, have relatively weaker novelty detection accuracy
  - Because **a latent space learned for a class with complex shapes** inherently learns to represent some of out-of-class objects as well



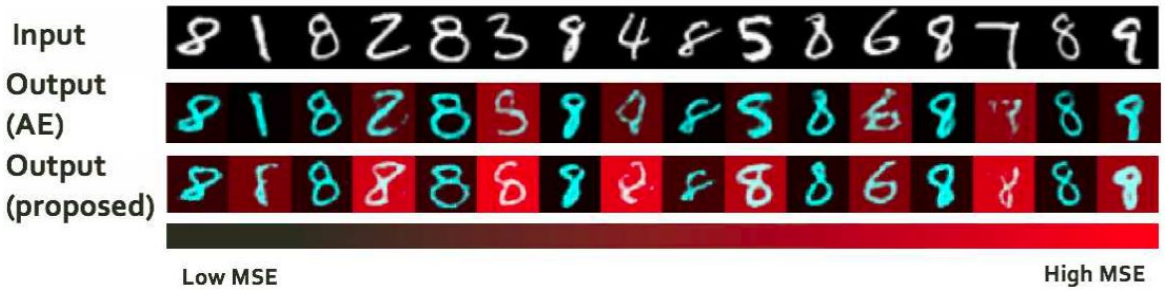
Introduction

- The requirement in novelty detection
  - In-class samples are well represented
  - Out-of-class samples are **poorly represented**



Introduction

- In this work, propose One-Class GAN(OCGAN)
  - A two-fold latent space learning process that considers both these requirements
  - Learn a latent space that represents objects of a given class well
    - Ensure that any example generated from the learned latent space is indeed **from the known class**

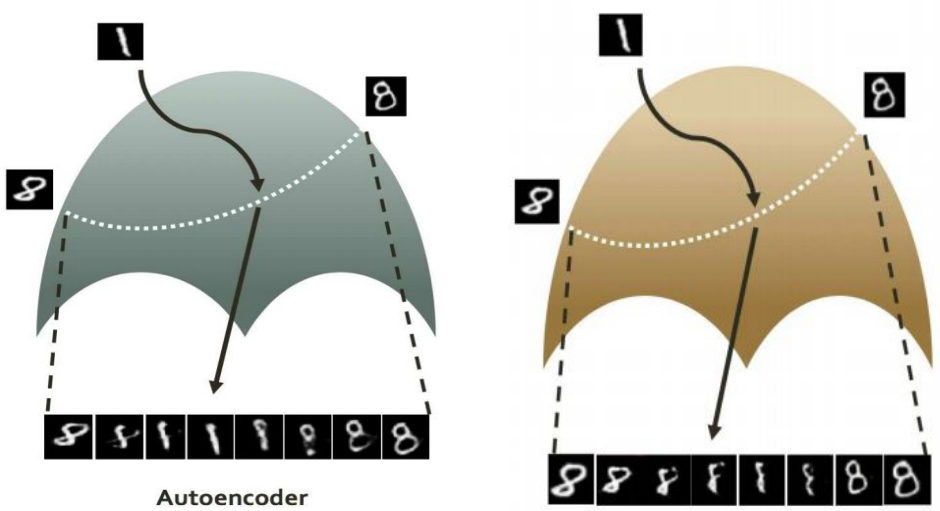


Outline

- Introduction
- Method
  - Motivation
  - Proposed Strategy
- Experiments
- Conclusions
- Progress Report

Method - Motivation

- If the entire latent space is constrained to represent images of the given class, the representation of out-of-class samples will be minimal
  - Thereby producing high reconstruction errors for them
- **Explicitly force the entirety of the latent space to represent only the given class**



Autoencoder

Outline

- Introduction
- Method
  - Motivation
  - Proposed Strategy
- Experiments
- Conclusions
- Progress Report

Method - Proposed strategy

- OCGAN consists of four components
  - Denoising auto-encoder
  - Two discriminators
    - Latent discriminator
    - Visual discriminator
  - Classifier

Method - Proposed strategy

- Denoising auto-encoder
  - Objective
    - Minimizing the distance between the input and the output of the network
  - Noise is added to the input image
    - The network is expected to reconstruct the denoised version of the image
  - Densely sampling from the latent space
    - Having a bounded support for the latent space
      - A tanh activation in the output layer of the encoder
      - Support of the latent space is $(-1, 1)^d$

$$l_{\mathrm{MSE}} = ||x - \mathrm{De}(\mathrm{En}(x + n))||_2^2,$$

$$n \sim \mathcal{N}(0, 0.2)$$

Method - Proposed strategy

- Latent Discriminator
  - The motivation
    - Obtain a latent space where each and every instance from the latent space represents an image from the given class
  - **Force latent representations of in-class examples to be distributed uniformly across the latent space**
  - Trained to differentiate between latent representations of real images of the given class and samples drawn from a $U(-1,1)^d$ distribution

$$l_{\text{latent}} = -(\mathbb{E}_{s\sim\mathbb{U}(-1,1)}[\log D_l(s)] + \mathbb{E}_{x\sim p_x}[\log(1 - D_l(\text{En}(x+n)))])$$

p_x is the distribution of in-class examples

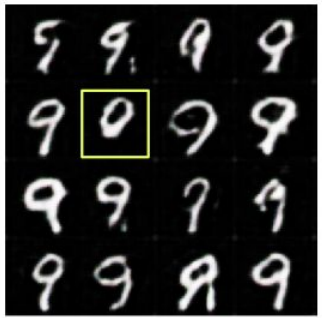
Method - Proposed strategy

- Visual Discriminator
  - In order for the network not to represent any out-of-class objects
    - Force all images generated from latent samples are from the same image space distribution as the given class
  - Trained to differentiate between images of the given class and images **generated from random latent samples** using the decoder
  - Fake images chosen at random in general will look similar to examples from the given class

$$l_{\text{visual}} = -(\mathbb{E}_{s\sim\mathbb{U}(-1,1)}[\log D_v(\text{De}(s))] + \mathbb{E}_{x\sim p_l}[\log(1 - D_v(x))])$$

Method - Proposed strategy

- Informative-negative Mining
  - The components described thus far account for the core of the proposed network
    - There are few cases where the produced output looks different from the given class
  - Despite the proposed training procedure, there are latent space regions that do not produce images of the given class
    - Because sampling from all regions in the latent space is impossible during training



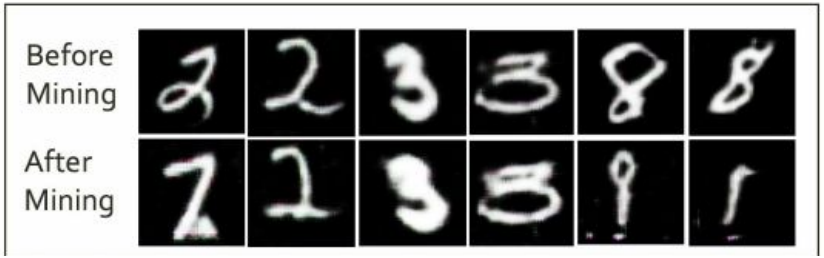
(a)

Method - Proposed strategy

- Informative-negative Mining
  - Propose to actively **seek regions in the latent space that prodice images of poor quality**
    - Learns to produce good quality in-class images even for these latent samples
  - To find informative-negative samples,
    - Start with random latent-space samples
    - Use a classifier to assess the quality of the image generated from the sample
    - Back-propagate and compute gradients in the latent space
    - Take a small step in the direction of the gradient to move to a new point in the latent space where the classifier is confident that the generated image is out-of-class

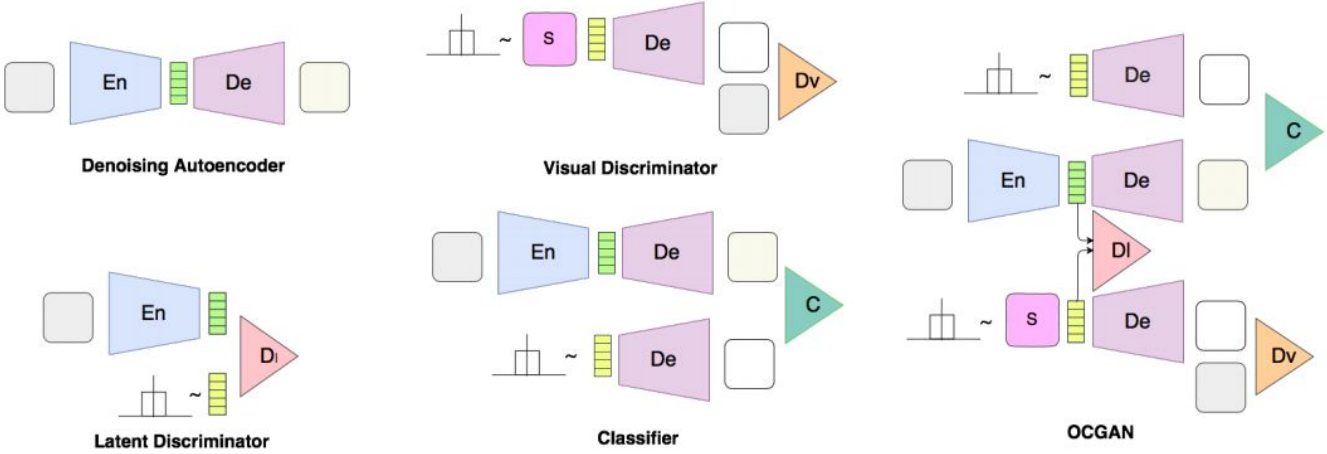
Method - Proposed strategy

- Classifier
  - Determine how well the given image resembles content of the given class
  - Train a weaker classifier instead
    - Reconstructions of in-class samples as positives
    - Generated from random samples in the latent space, as negatives



Method - Proposed strategy

- Full OCGAN Model



Outline

- Introduction
- Method
  - Motivation
  - Proposed Strategy
- Experiments
- Conclusions
- Progress Report

Experiments

- Evaluation Methodology
  - Area Under the Curve(AUC)
  - Receiver Operating Characteristics (ROC) curve
- Protocol 1
  - Training is carried out using 80% of in-class samples. The remaining 20% of in-class data is used for testing. Negative test samples are randomly selected so that they constitute half of the test set.
- Protocal 2
  - Use the training-testing splits of the given dataset to conduct training. Training split of the known class is used for training / validation. Testing data of all classes are used for testing.

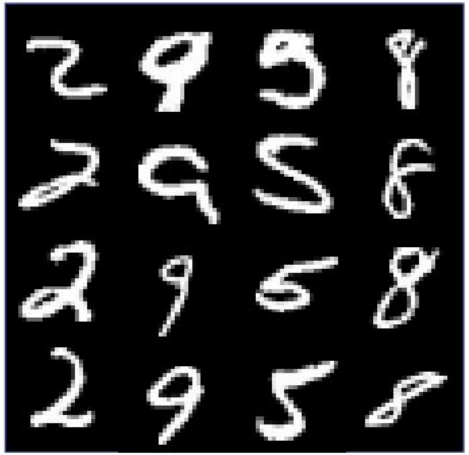
Experiments

- Datasets

CIFAR10

COIL

FMNIST



MNIST

Experiments

- Protocol 1

	MNIST	COIL	fMNIST
ALOCC DR [22]	0.88	0.809	0.753
ALOCC D [22]	0.82	0.686	0.601
DCAE [23]	0.899	0.949	0.908
GPND [18]	0.932	0.968	0.901
OCGAN	**0.977**	**0.995**	**0.924**

[22] Babak Saleh, Ali Farhadi, and Ahmed Elgammal. Objectcentric anomaly detection by attribute-based reasoning. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, pages 787–794, 2013.
[23] Thomas Schlegl, Philipp Seebock, Sebastian M. Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In IPMI, 2017. 2, 8
[18] Stephen J Roberts. Novelty detection using extreme value statistics. IEE Proceedings-Vision, Image and Signal Processing, 146(3):124–129, 1999.

[26] Aaron van den Oord, Nal Kalchbrenner, Lasse Espeholt, koray kavukcuoglu, Oriol Vinyals, and Alex Graves. Conditional image generation with pixelcnn decoders. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, Advances in Neural Information Processing Systems 29, pages 4790–4798. 2016. 8
[2] Christopher M. Bishop. Pattern Recognition and Machine Learning (Information Science and Statistics). 2006
[4] Raia Hadsell, Sumit Chopra, and Yann Lecun. Dimensionality reduction by learning an invariant mapping. In In Proc. Computer Vision and Pattern Recognition Conference (CVPR06. IEEE Press, 2006
[6] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In International Conference on Learning Representations.

Experiments

- MNIST with Protocol 2

Table 2. One-class novelty detection results for MNIST dataset using Protocol 2.

	0	1	2	3	4	5	6	7	8	9	MEAN
OCSVM [26]	0.988	**0.999**	0.902	0.950	0.955	0.968	0.978	0.965	0.853	0.955	0.9513
KDE [2]	0.885	0.996	0.710	0.693	0.844	0.776	0.861	0.884	0.669	0.825	0.8143
DAE [4]	0.894	**0.999**	0.792	0.851	0.888	0.819	0.944	0.922	0.740	0.917	0.8766
VAE [6]	0.997	**0.999**	0.936	0.959	0.973	0.964	0.993	0.976	0.923	0.976	0.9696
Pix CNN [28]	0.531	0.995	0.476	0.517	0.739	0.542	0.592	0.789	0.340	0.662	0.6183
GAN [25]	0.926	0.995	0.805	0.818	0.823	0.803	0.890	0.898	0.817	0.887	0.8662
AND [1]	0.984	0.995	**0.947**	0.952	0.960	0.971	**0.991**	0.970	0.922	0.979	0.9671
AnoGAN [25]	0.966	0.992	0.850	0.887	0.894	0.883	0.947	0.935	0.849	0.924	0.9127
DSVDD [21]	0.980	0.997	0.917	0.919	0.949	0.885	0.983	0.946	**0.939**	0.965	0.9480
OCGAN	**0.998**	**0.999**	0.942	**0.963**	**0.975**	**0.980**	**0.991**	**0.981**	**0.939**	**0.981**	**0.9750**

[28] Yan Xia, Xudong Cao, Fang Wen, Gang Hua, and Jian Sun. Learning discriminative reconstructions for unsupervised outlier removal. In 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015, pages 1511–1519, 2015
[25] David M. J. Tax and Robert P. W. Duin. Support vector data description. Mach. Learn., 54(1):45–66, 2004
[21] Mayu Sakurada and Takehisa Yairi. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In Proceedings of the MLSDA 2014 2Nd Workshop on Machine Learning for Sensory Data Analysis, 2014
[1] D. Abati, A. Porrello, S. Calderara, and R. Cucchiara. AND: Autoregressive Novelty Detectors. In 2019 IEEE Conference on Computer Vision and Pattern Recognition, 2019

[26] Aaron van den Oord, Nal Kalchbrenner, Lasse Espeholt, koray kavukcuoglu, Oriol Vinyals, and Alex Graves. Conditional image generation with pixelcnn decoders. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, Advances in Neural Information Processing Systems 29, pages 4790–4798. 2016. 8
[2] Christopher M. Bishop. Pattern Recognition and Machine Learning (Information Science and Statistics). 2006
[4] Raia Hadsell, Sumit Chopra, and Yann Lecun. Dimensionality reduction by learning an invariant mapping. In In Proc. Computer Vision and Pattern Recognition Conference (CVPR06. IEEE Press, 2006
[6] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In International Conference on Learning Representations.

Experiments

- CIFAR10 with Protocol 2

Table 3. One-class novelty detection results for CIFAR10 dataset using Protocol 2. Plane and Car classes are annotated as Airplane and Automobile in CIFAR10.

	PLANE	CAR	BIRD	CAT	DEER	DOG	FROG	HORSE	SHIP	TRUCK	MEAN
OCSVM [26]	0.630	0.440	0.649	0.487	0.735	0.500	0.725	0.533	0.649	0.508	0.5856
KDE [2]	0.658	0.520	0.657	0.497	0.727	0.496	**0.758**	0.564	0.680	0.540	0.6097
DAE [4]	0.411	0.478	0.616	0.562	0.728	0.513	0.688	0.497	0.487	0.378	0.5358
VAE [6]	0.700	0.386	**0.679**	0.535	**0.748**	0.523	0.687	0.493	0.696	0.386	0.5833
Pix CNN [28]	**0.788**	0.428	0.617	0.574	0.511	0.571	0.422	0.454	0.715	0.426	0.5506
GAN [25]	0.708	0.458	0.664	0.510	0.722	0.505	0.707	0.471	0.713	0.458	0.5916
AND [1]	0.717	0.494	0.662	0.527	0.736	0.504	0.726	0.560	0.680	0.566	0.6172
AnoGAN [25]	0.671	0.547	0.529	0.545	0.651	0.603	0.585	0.625	0.758	0.665	0.6179
DSVDD [21]	0.617	**0.659**	0.508	0.591	0.609	**0.657**	0.677	**0.673**	0.759	**0.731**	0.6481
OCGAN	0.757	0.531	0.640	**0.620**	0.723	0.620	0.723	0.575	**0.820**	0.554	**0.6566**

[28] Yan Xia, Xudong Cao, Fang Wen, Gang Hua, and Jian Sun. Learning discriminative reconstructions for unsupervised outlier removal. In 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015, pages 1511–1519, 2015
[25] David M. J. Tax and Robert P. W. Duin. Support vector data description. Mach. Learn., 54(1):45–66, 2004
[21] Mayu Sakurada and Takehisa Yairi. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In Proceedings of the MLSDA 2014 2Nd Workshop on Machine Learning for Sensory Data Analysis, 2014
[1] D. Abati, A. Porrello, S. Calderara, and R. Cucchiara. AND: Autoregressive Novelty Detectors. In 2019 IEEE Conference on Computer Vision and Pattern Recognition, 2019

Experiments

- Ablation Study

Table 4. Ablation study for OCGAN performed on MNIST.

Without any Discriminators	0.957
With latent Discriminator	0.959
With two Discriminators	0.971
Two Discriminators + Classifier	0.975

Outline

- Introduction
- Method
  - Motivation
  - Proposed Strategy
- Experiments
- Conclusions
- Progress Report

Conclusions

- In this work, we showed a network trained on a single class is capable of representing some out-of-class examples
- A latent-space-sampling-based network learning procedure
  - Restricted the latent space to be bounded and forced latent projections of in-class population to be distributed evenly in the latent space using a latent discriminator
  - Sampled from the latent space and ensured using a visual discriminator that any random latent sample generates an image from the same class
  - Attempt to reduce false positives we introduced an informative-negative mining procedure.

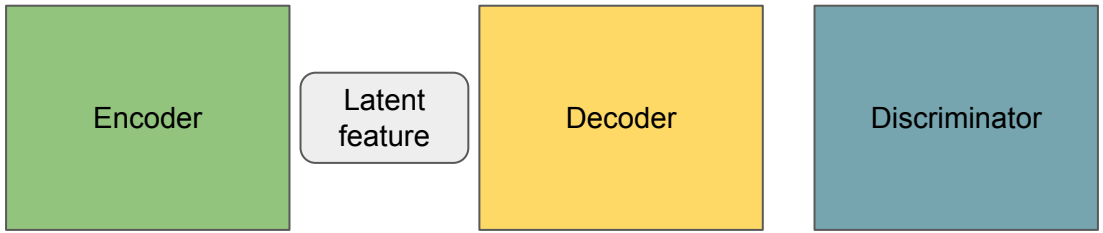
Outline

- Introduction
- Method
  - Motivation
  - Proposed Strategy
- Experiments
- Conclusions
- Progress Report

Progress Report

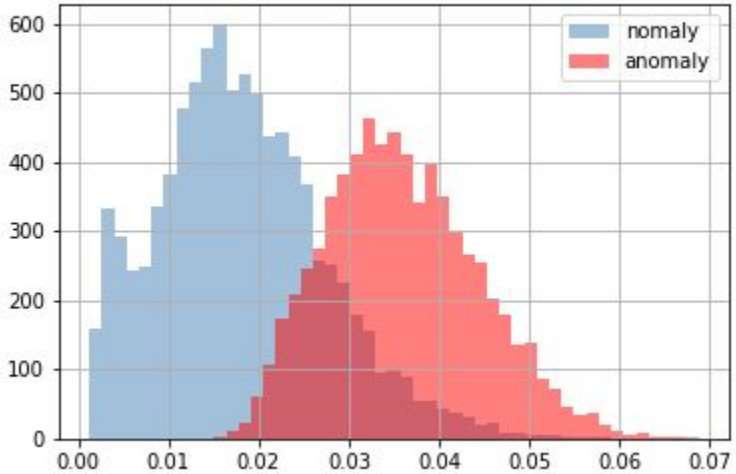
- The drawback of OCGAN
  - If we have many class in our normal data? How to reconstruct the latent space?
- Instead of forcing the decoder to reconstruct the entire latent space to normal data
  - We force the normal data to form a gaussian distribution tightly
  - Then the abnormal data will naturally be outside the distribution

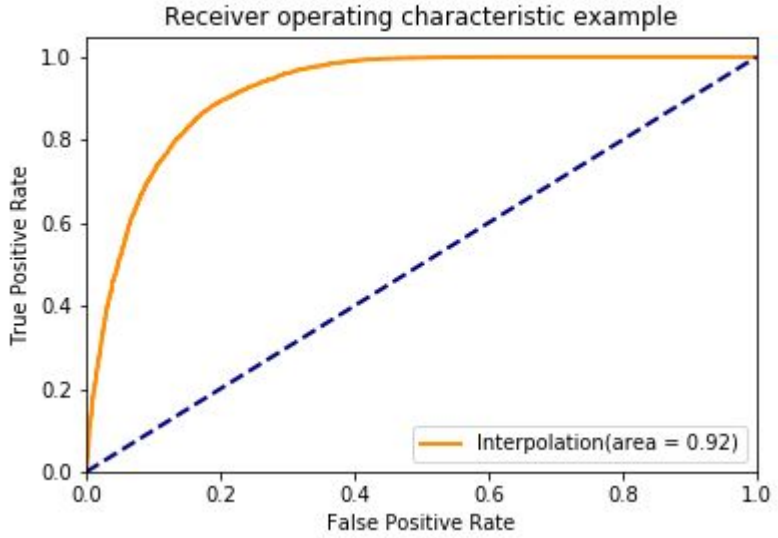
Baseline - MNIST



Baseline - MNIST

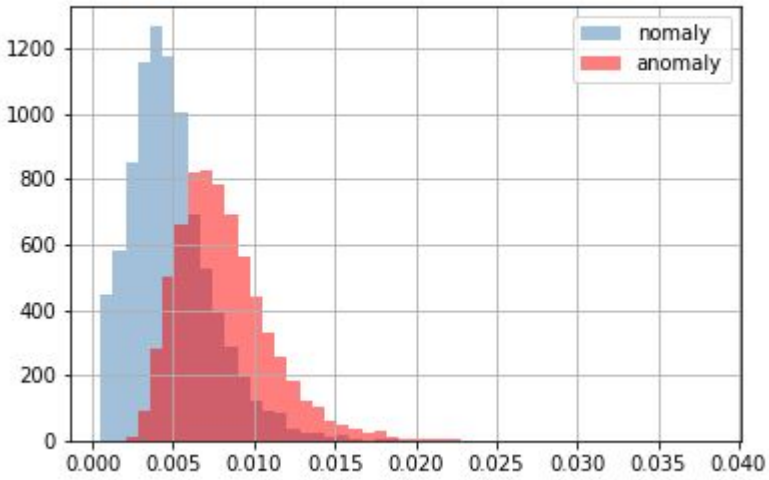
- Best AUROC

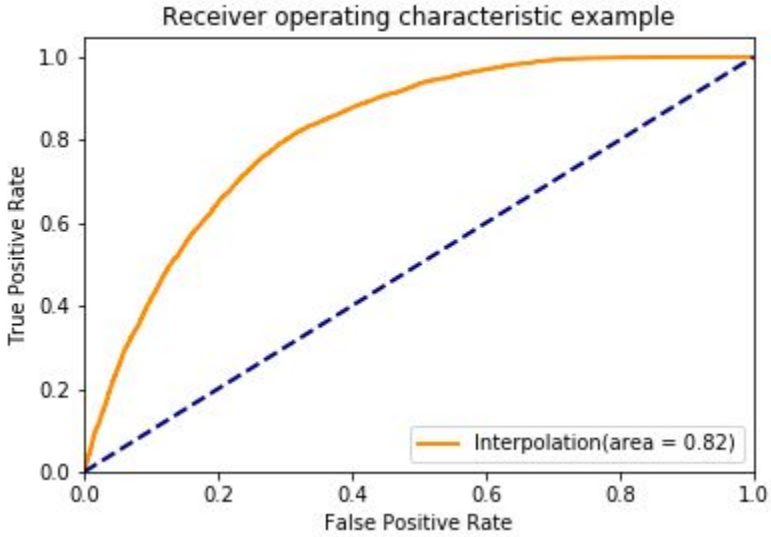




Baseline - MNIST

- Last epoch AUROC

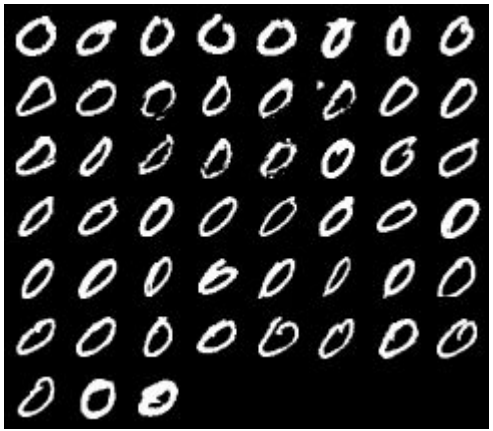




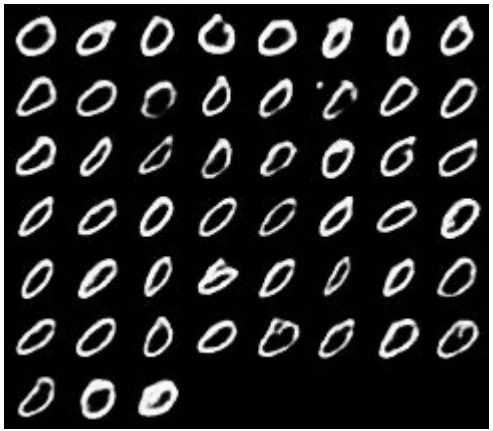
Baseline - MNIST

- Abnormal reconstruction

GT



Reconstruction



Baseline - MNIST

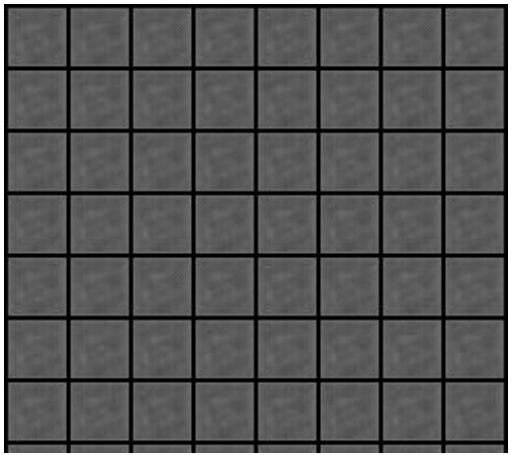
- Normal reconstruction

GT

Reconstruction

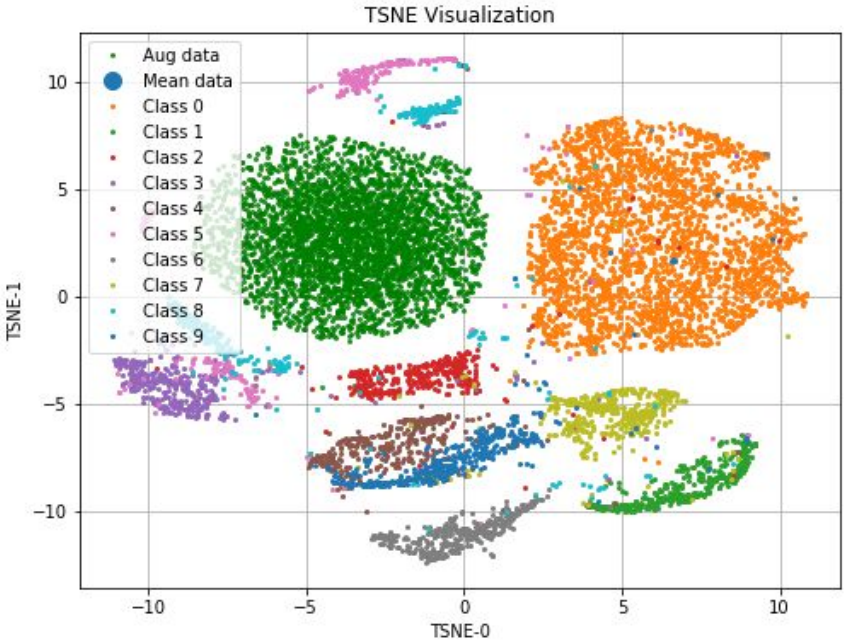
Baseline - MNIST

- Reconstruction from the feature sampled from normal distribution

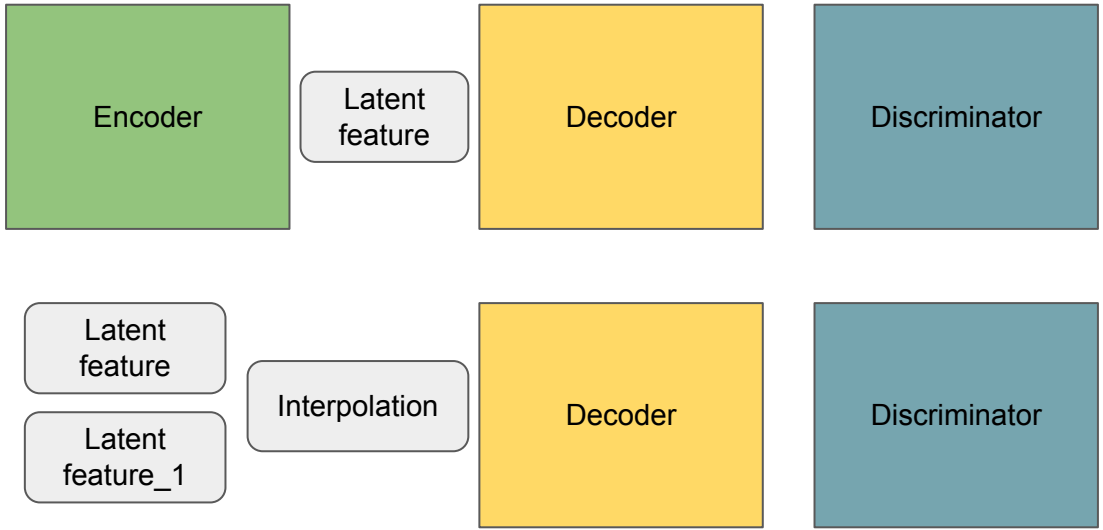


Baseline - MNIST

- Visualization



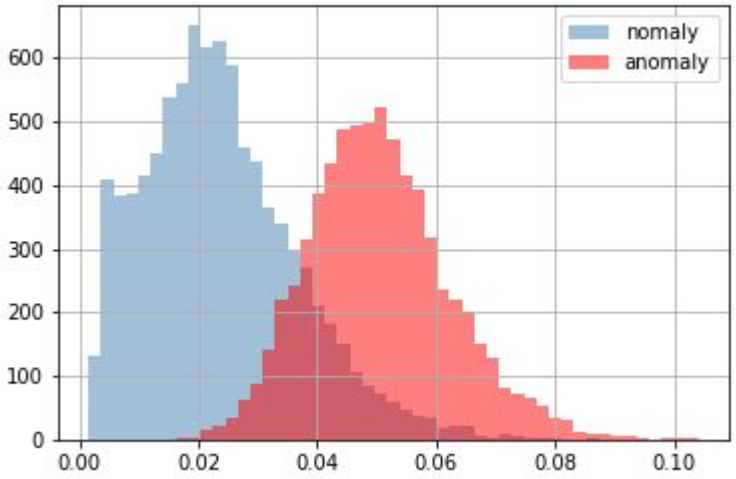
Baseline + Inter - MNIST

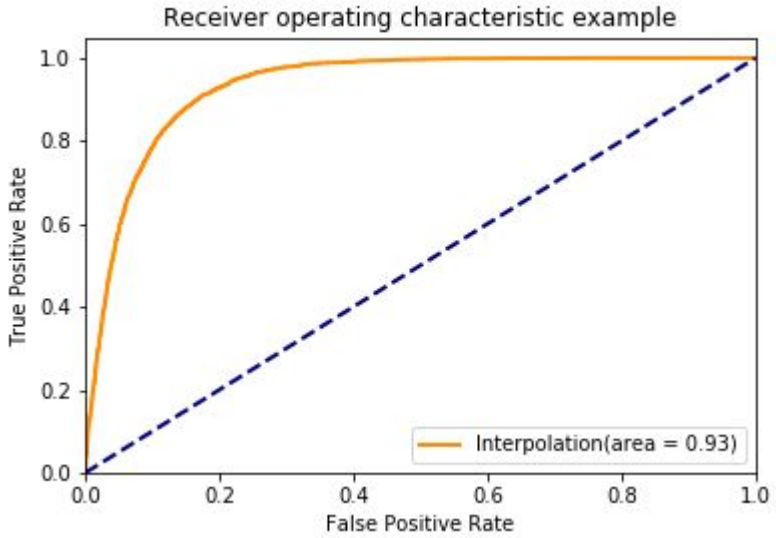


David Berthelot, Colin Raffel, Aurko Roy, Ian Goodfellow.Understanding and Improving Interpolation in Autoencoders via an Adversarial Regularizer. ICLR2018

Baseline + Inter - MNIST

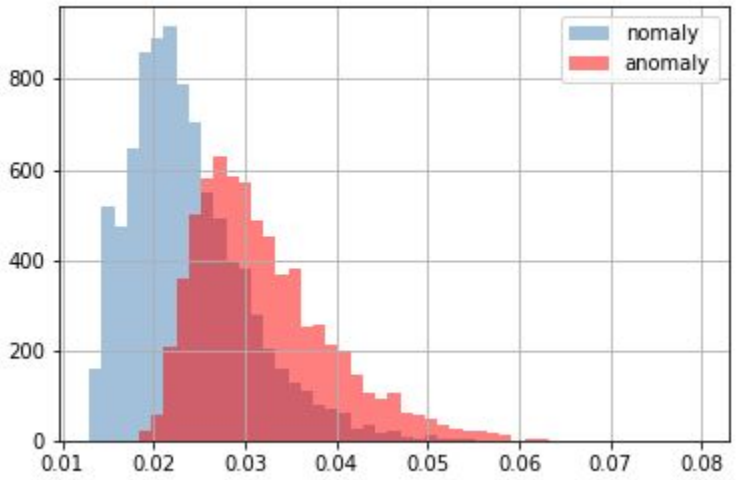
- Best AUROC

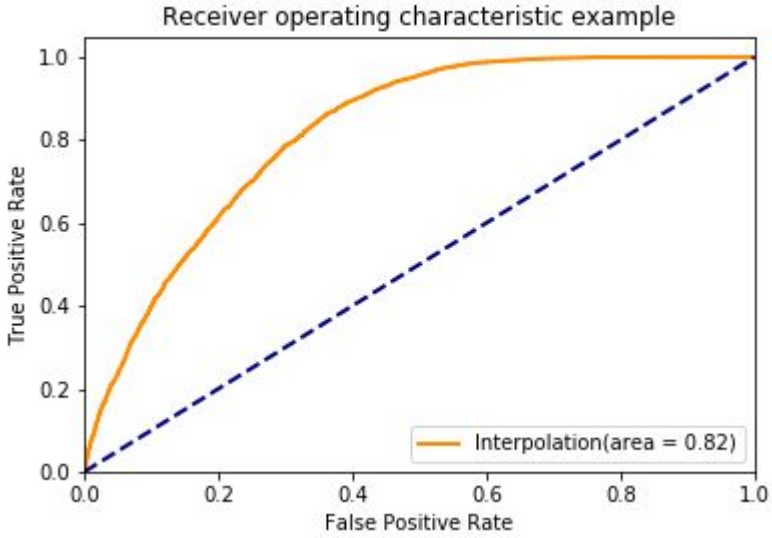




Baseline + Inter - MNIST

- Last epoch AUROC

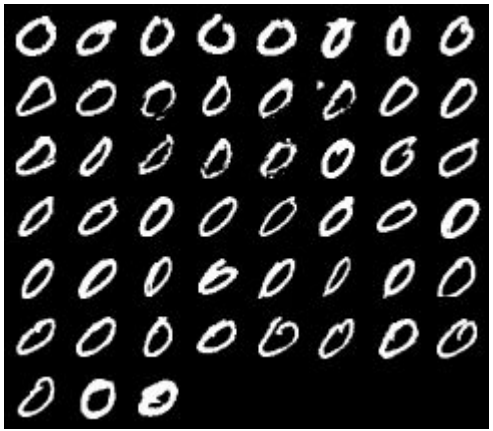




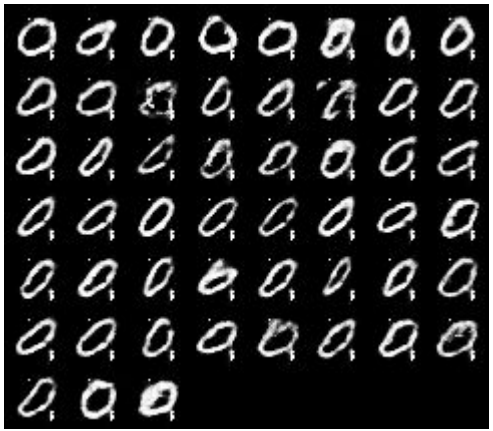
Baseline + Inter - MNIST

- Abnormal reconstruction

GT



Reconstruction



Baseline + Inter - MNIST

- Normal reconstruction

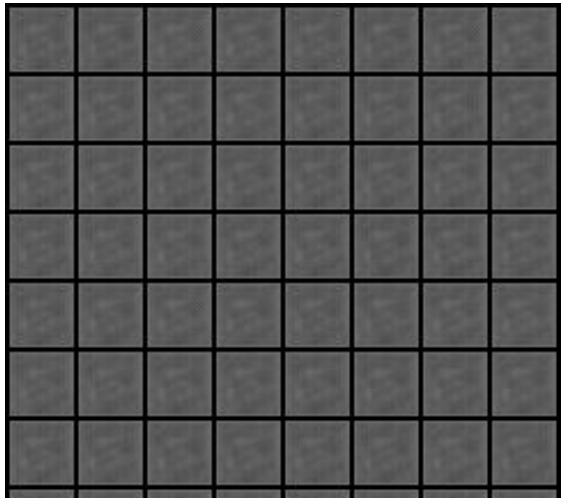
GT

Reconstruction

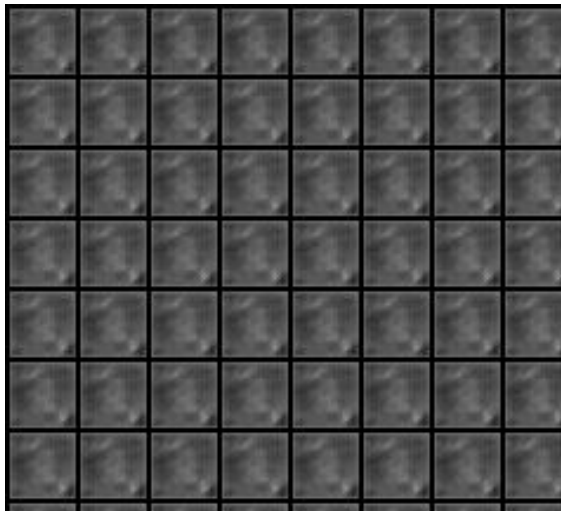
Baseline + Inter - MNIST

- Reconstruction from the feature sampled from normal distribution

Baseline

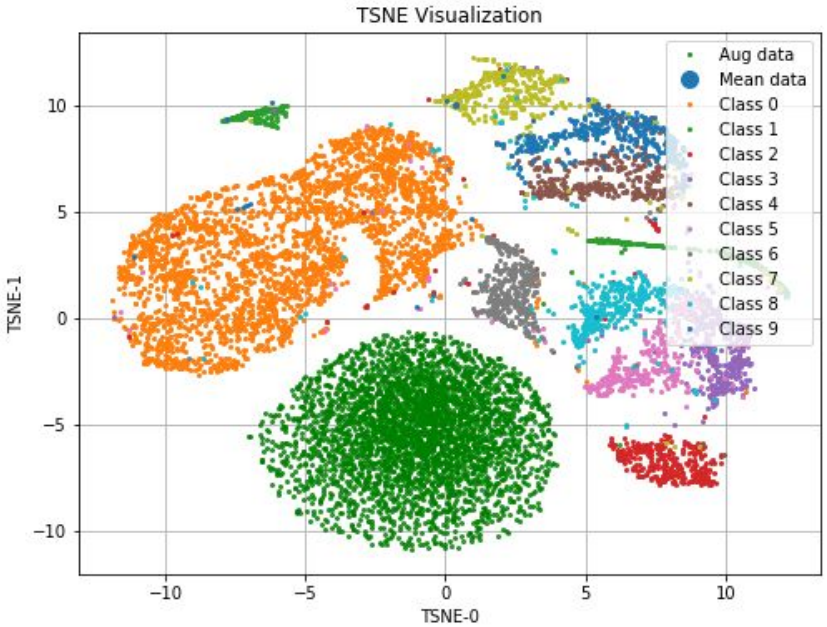


Baseline+Inter

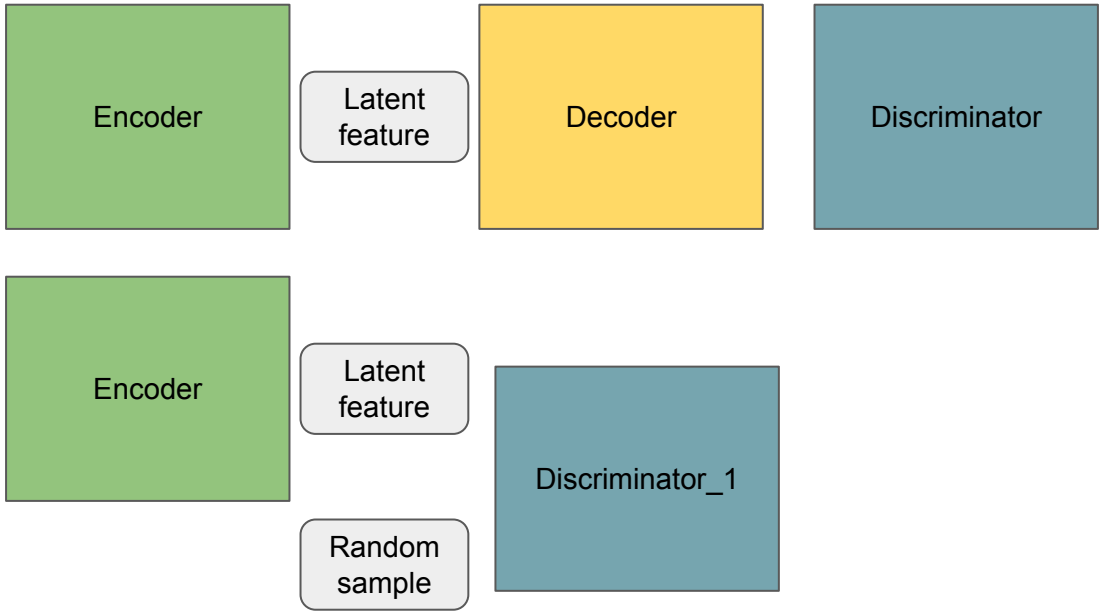


Baseline + Inter - MNIST

- Visualization



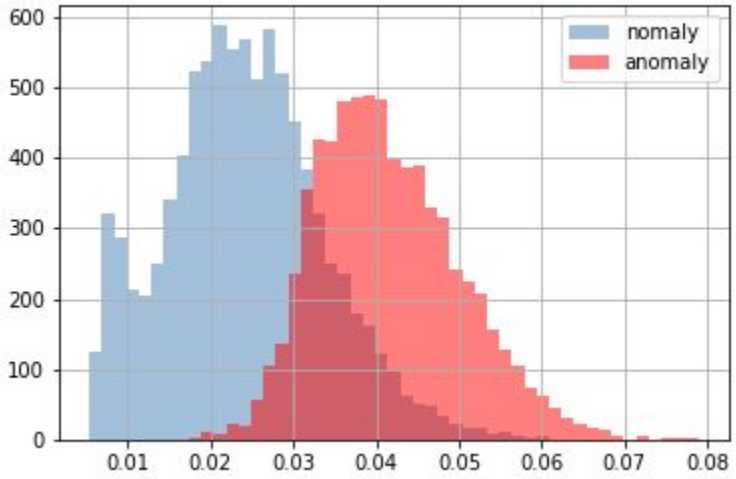
Baseline + LC - MNIST

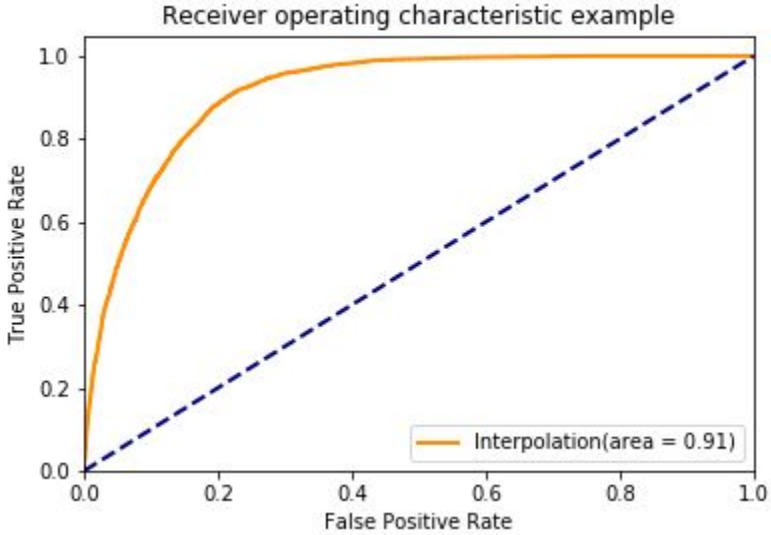


Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, Brendan Frey. Adversarial Autoencoders

Baseline + LC - MNIST

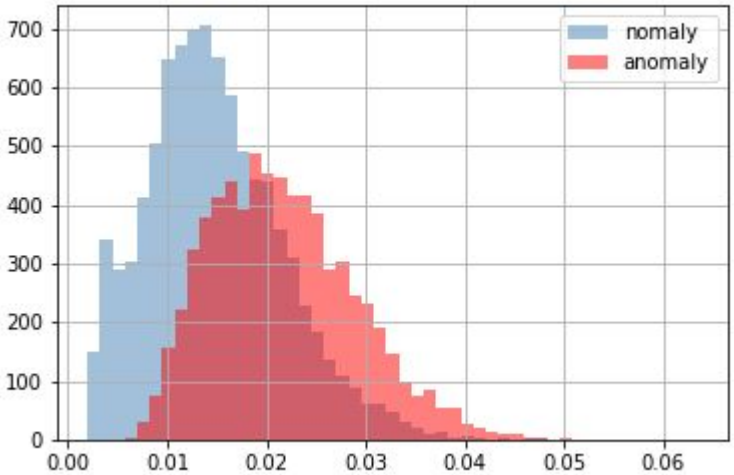
- Best AUROC

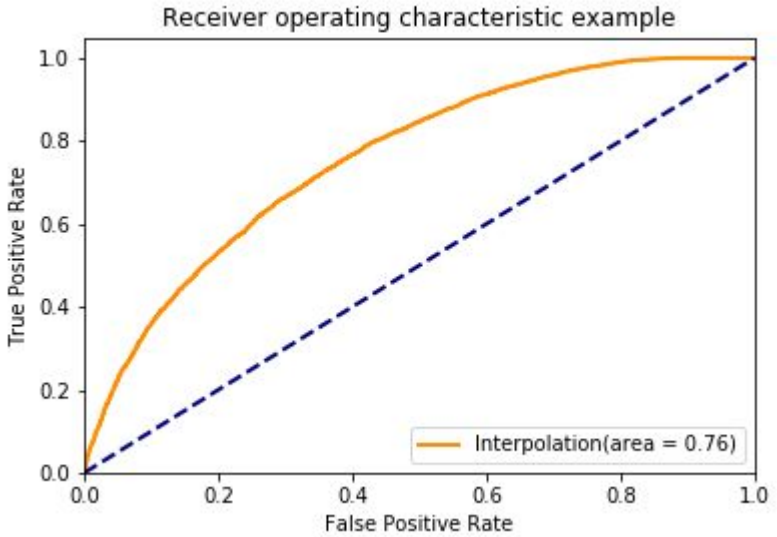




Baseline + LC - MNIST

- Last epoch AUROC

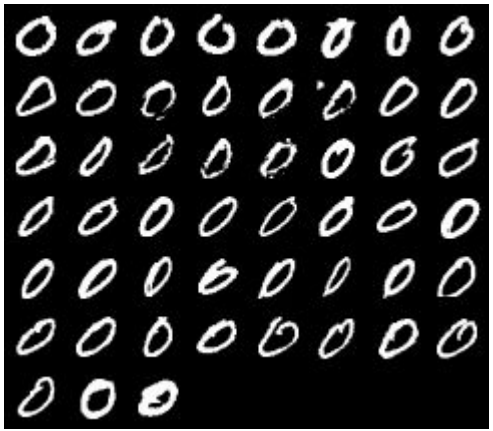




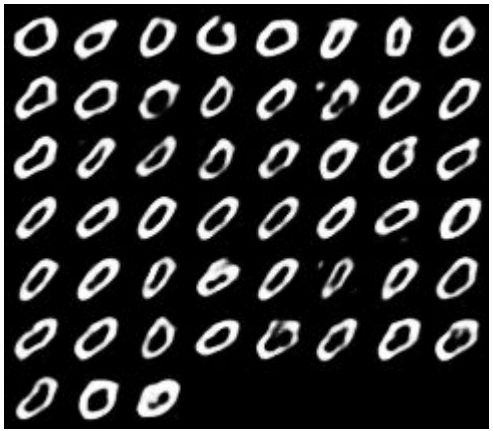
Baseline + LC - MNIST

- Abnormal reconstruction

GT



Reconstruction



Baseline + LC - MNIST

- Normal reconstruction

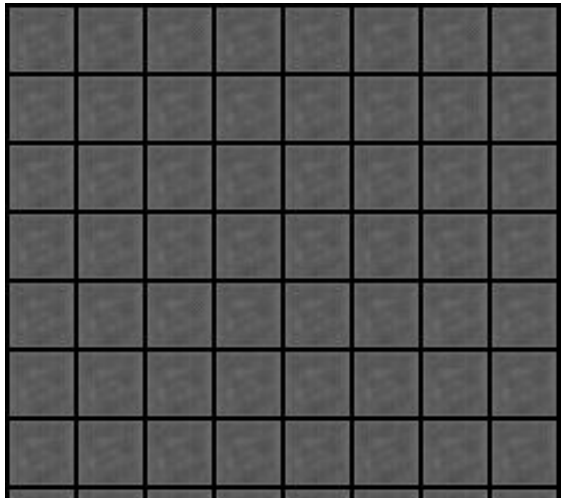
GT

Reconstruction

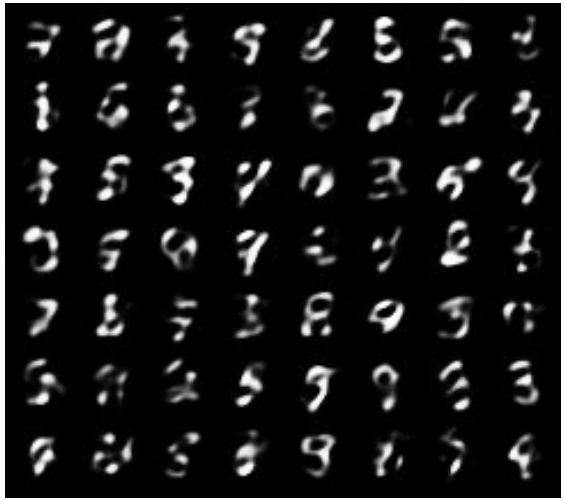
Baseline + LC - MNIST

- Reconstruction from the feature sampled from normal distribution

Baseline

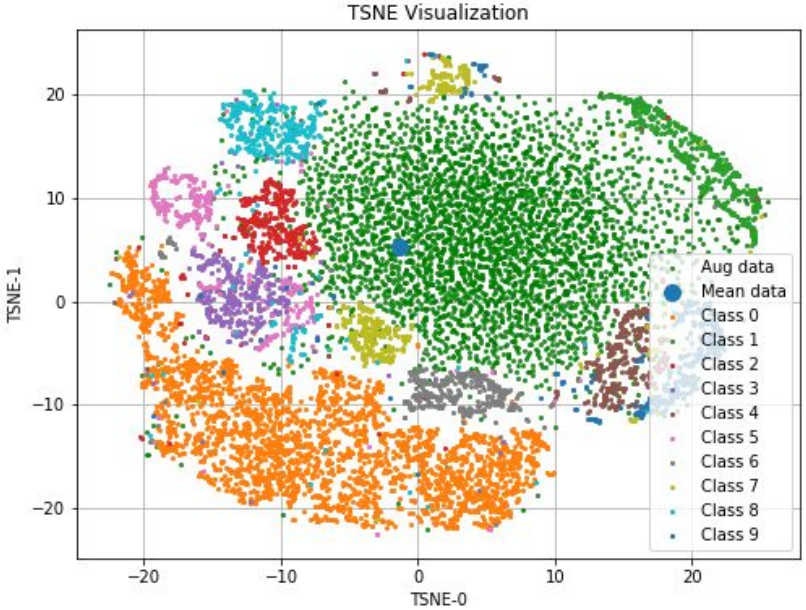


Baseline+LC



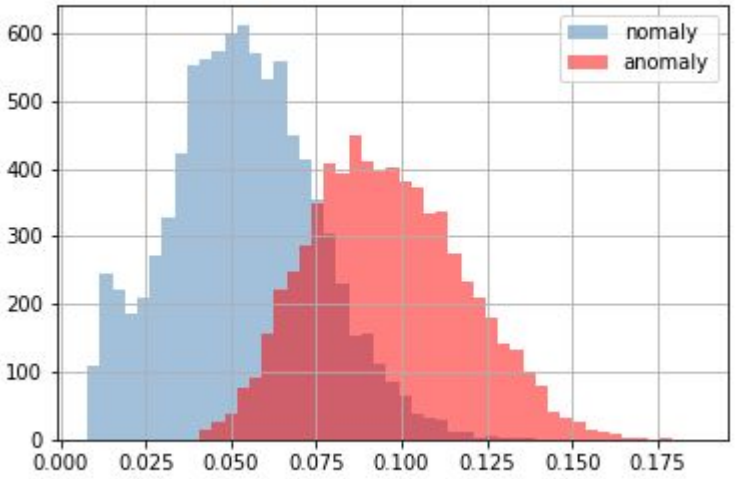
Baseline + LC - MNIST

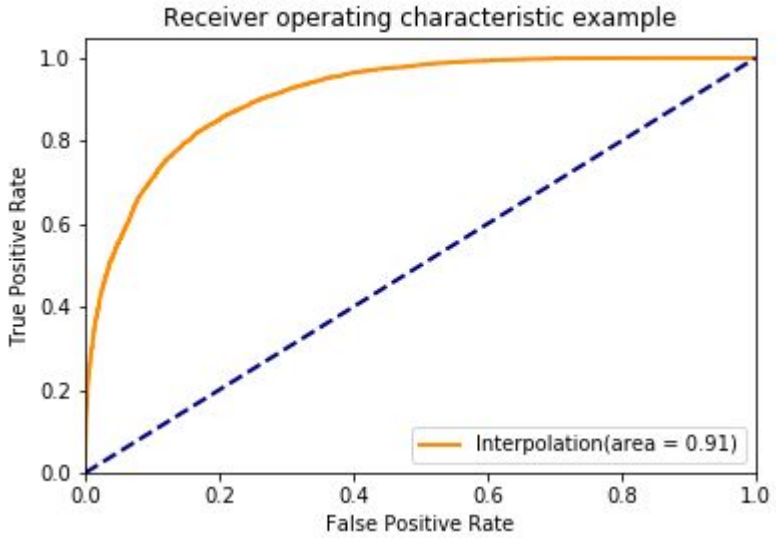
- Visualization



Baseline + Inter + LC - MNIST

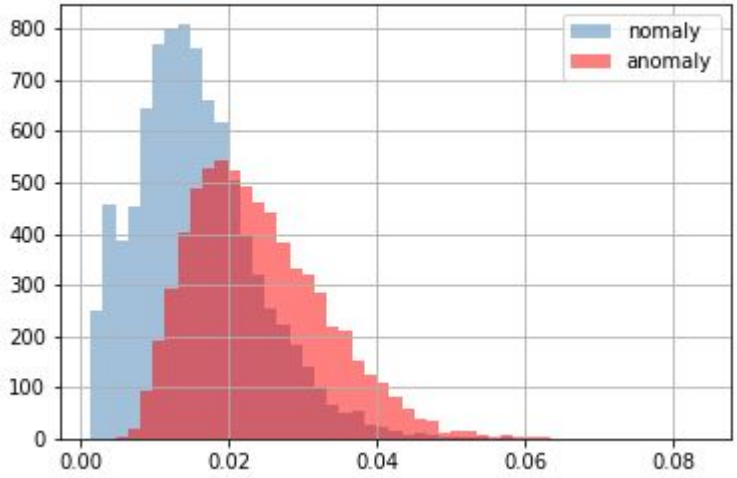
- Best AUROC

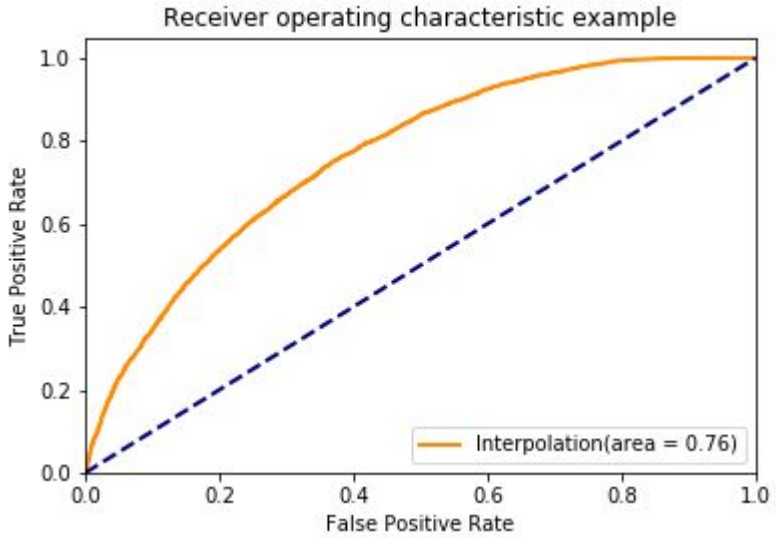




Baseline + Inter + LC - MNIST

- Last epoch AUROC

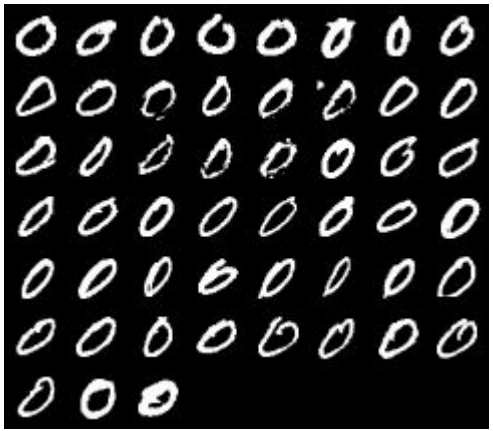




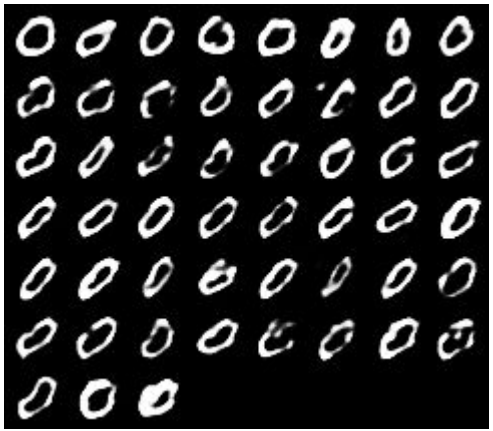
Baseline + Inter + LC - MNIST

- Abnormal reconstruction

GT



Reconstruction



Baseline + Inter + LC - MNIST

- Normal reconstruction

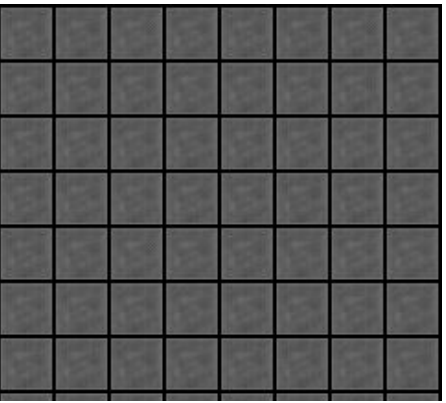
GT

Reconstruction

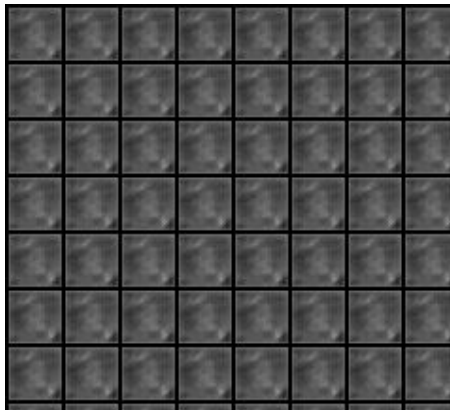
Baseline + Inter + LC - MNIST

- Reconstruction from the feature sampled from normal distribution

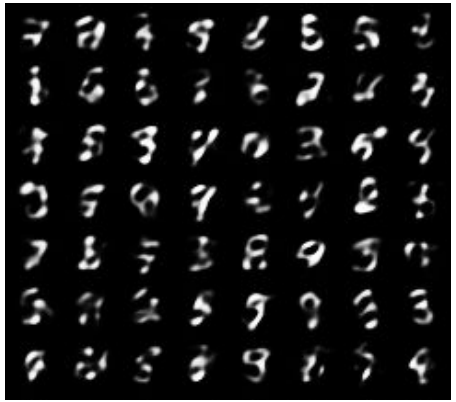
Baseline



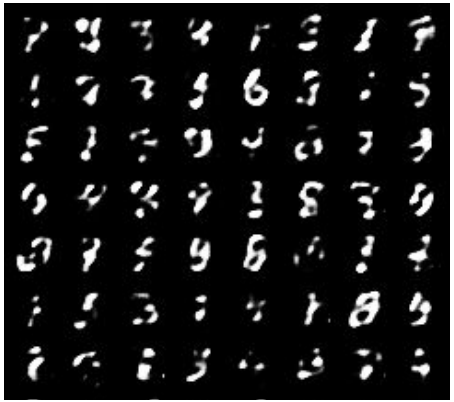
Baseline+Inter



Baseline+LC

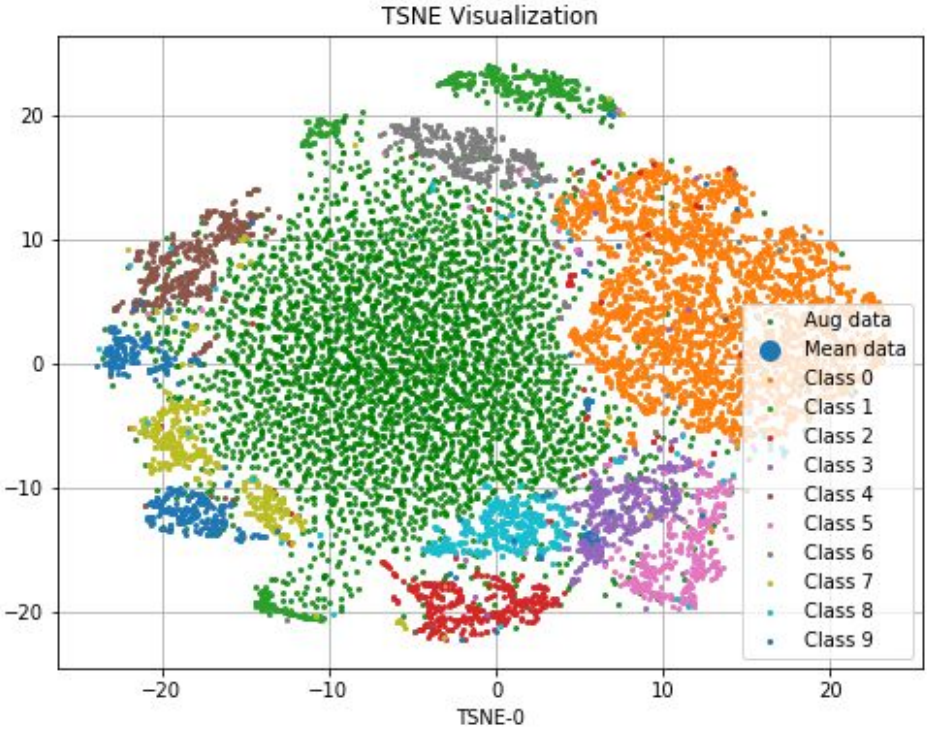


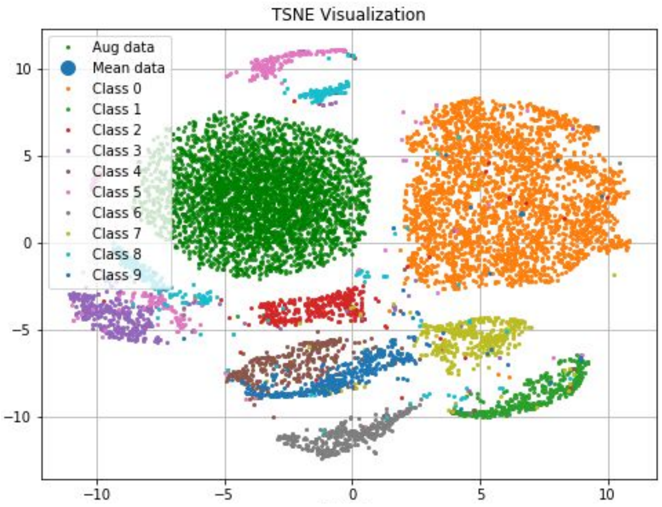
Baseline+LC+Inter

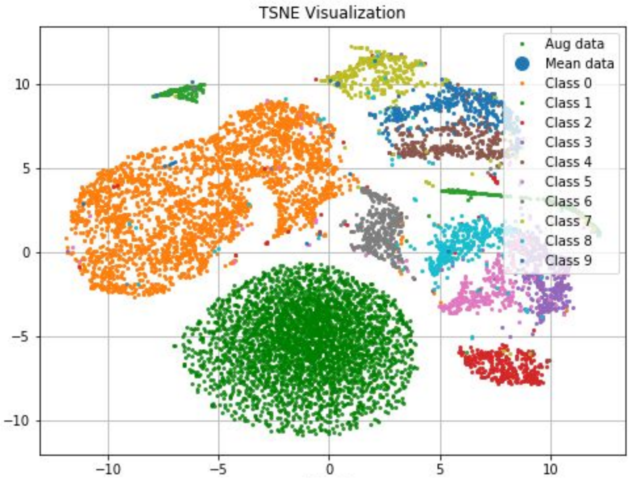


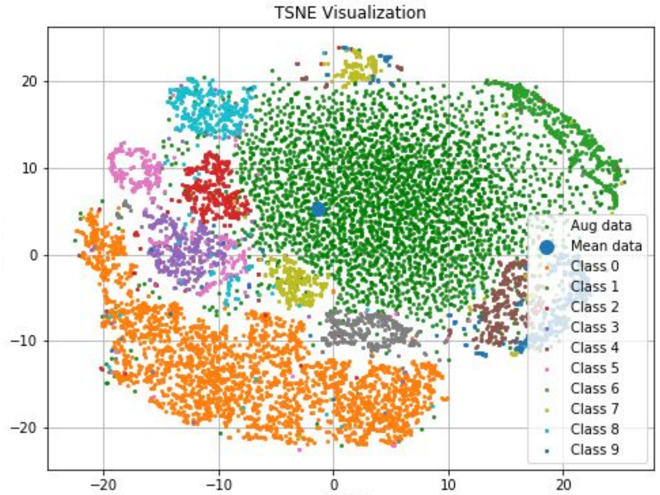
Baseline + Inter + LC - MNIST

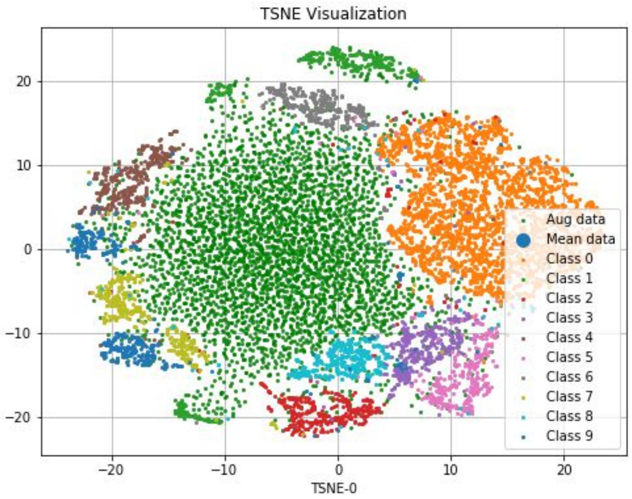
- Visualization




TSNE Visualization
Aug data
Mean data
Class 0
Class 1
Class 2
Class 3
Class 4
Class 5
Class 6
Class 7
Class 8
Class 9
10
5
0
−5
−10
−10
−5
0
5
10


TSNE Visualization
Aug data
Mean data
Class 0
Class 1
Class 2
Class 3
Class 4
Class 5
Class 6
Class 7
Class 8
Class 9
10
5
0
−5
−10
−10
−5
0
5
10


TSNE Visualization
Aug data
Mean data
Class 0
Class 1
Class 2
Class 3
Class 4
Class 5
Class 6
Class 7
Class 8
Class 9
20
10
0
−10
−20
−20
−10
0
10
20


TSNE Visualization
Aug data
Mean data
Class 0
Class 1
Class 2
Class 3
Class 4
Class 5
Class 6
Class 7
Class 8
Class 9
20
10
0
−10
−20
−20
−10
0
10
20
TSNE-0

DONAS

- Comparison experiment
  - Compare with SOTA
  - Transfer to CIFAR10
  - Object Detection
- Ablation study
  - Novel architecture of supernet(Multiple kernel + One shot)
    - Training supernet w/wo specific order
    - Training supernet w/wo SBN
    - Compare the parameter of our supernet and previous supernet
  - Generator searching
    - The generate curve with supernet evaluation
    - Compare with random search (accuracy, time)
    - Compare with evolution algorithm(accuracy, time)
    - Generator w/wo the backbone
    - Generator with different hardware constraint objective

DONAS

- Compare with SOTA

	Paramaters	FLOPs	Top-1 acc	Deploy time(GPU days)
MobileNet V2	3.4M	300M	72.0%	–
EfficientNet B0	5.3M	390M	76.3%	–
MixNet-S	4.1M	256M	75.8%	–
MixNet-M	5.0M	360M	77.0%	–
SCARLET-A	6.7M	365M	76.9%	2
GreedyNAS(CVPR2020)	6.5M	366M	77.1%	<1
DONAS	**6.0M**	**373M**	**77.09%**	**~0**
Scarlet-B	6.5M	329M	76.3%	2
GreedyNAS(CVPR2020)	5.2M	324M	76.8%	<1
DONAS	**5.5M**	**326M**	**76.83%**	**~0**
Scarlet-C	6.0M	280M	75.6%	2
GreedyNAS(CVPR2020)	4.7M	284M	76.2%	<1
DONAS	**4.7M**	**280M**	**76.2%**	**~0**

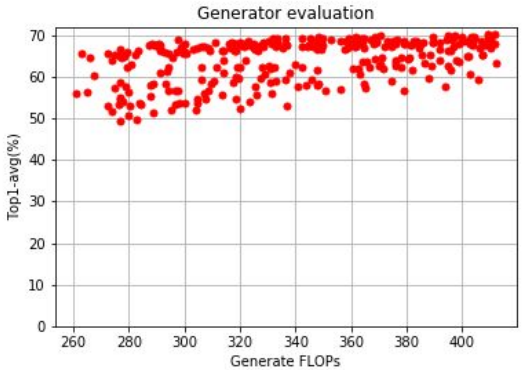
DONAS

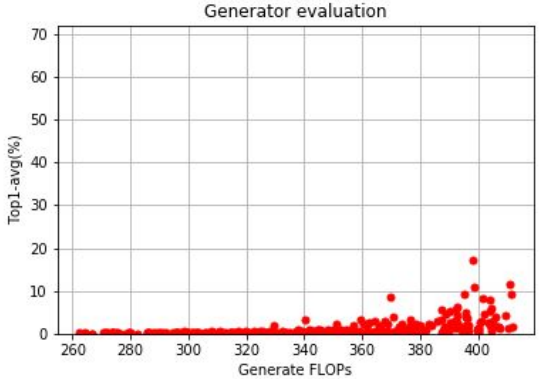
- The effective of our method

	Search Time(GPU hours)	FLOPs	Parameters	Top1-acc
Random search + novel supernet	90.03N			
Evolution algorithm + novel supernet	78.69N			
Single Path One-shot(ECCV2020)	48N(on V100)	326M	3.8M	74.50%
FairNAS-C	78.69M	321M	3.6M	74.69%
DONAS	30.48			

DONAS

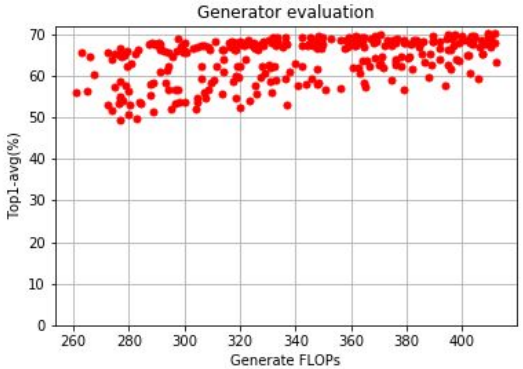
- Training supernet w/wo specific order (w SBN)

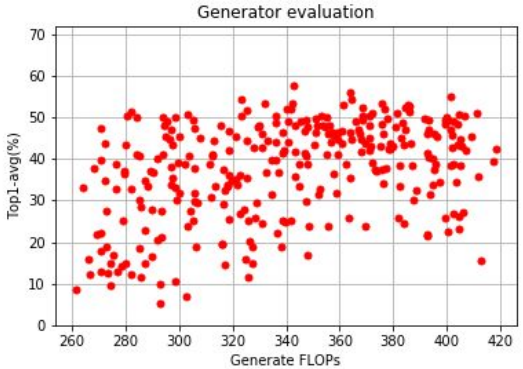




DONAS

- Training supernet w/wo SBN





DONAS

- The generate curve with supernet evaluation