Compressed Sensing Using Generative Models

Kenneth Emeka Odoh

14th Nov 2018 (Kaggle Data Science Meetup | SFU Ventures Lab)

Vancouver, BC

Table of Contents

- Compressed sensing
- Generative model

Inspired by the paper named Compressed sensing using generative model

[Bora et. al, 2017]

{https://arxiv.org/abs/1703.03208 }

Kenneth Emeka Odoh

Compressed Sensing

Compressed sensing is the way of estimating signal from underdetermined system by exploiting the structure of the problem.

$$y = Ax + \eta$$

where x is recovered signal, η is noise, and y is measurement signal.

Kenneth Emeka Odoh

Application of Compressed Sensing

- Dimensionality reduction techniques / Compression techniques
    - PCA, DCT, FFT, JPEG
- Denoising
- Deblurring
- Anomaly detection
- Blind source separation

There are a number of techniques for compressed sensing

Optimization theory e.g Linear programming, gradient descent

Underdetermined Linear System

Ax = b

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ a_{31} & a_{32} & \dots & a_{3m} \\ . & . & \dots & . \\ . & . & \dots & . \\ . & . & \dots & . \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix} \begin{bmatrix} x_{11} \\ x_{21} \\ x_{31} \\ . \\ . \\ . \\ x_{m1} \end{bmatrix} = \begin{bmatrix} b_{11} \\ b_{21} \\ b_{31} \\ . \\ . \\ . \\ b_{n1} \end{bmatrix}$$

A has dimension of n x m, **x** has m x 1, and **b** has n x 1, where n>m
b is signal, **x** is recovered signal
b, **x** are different representation of signals e.g time and frequency domain

- Infinitely many **solutions**

Structure on the Problem

- Sparsity
- Orthogonality
- Symmetry
- Smoothness

Imposing structure on the problem in form of **inductive bias**. This is achieved by adding regularizer which improves efficiency and simplicity of processing.

Handling different representations

- Parseval theorem
- Heisenberg uncertainty principle

Estimating A

The Conditions for extracting A

- Restricted Isometry Property (RIP)

This creates orthonormal matrix to impose sparsity.

- Restricted Eigenvalue Condition (REC)

This is a sufficient condition as it enforces that sparse vectors are far from null space. This can be achieved by initializing the weight from thin-tailed distribution.

Implication of Theorem 1.1, 1.2 lead to the least loss with a high probability if A is sampled for a thin-tailed normal distribution with tight bounds on the training size [Bora et. al, 2017].

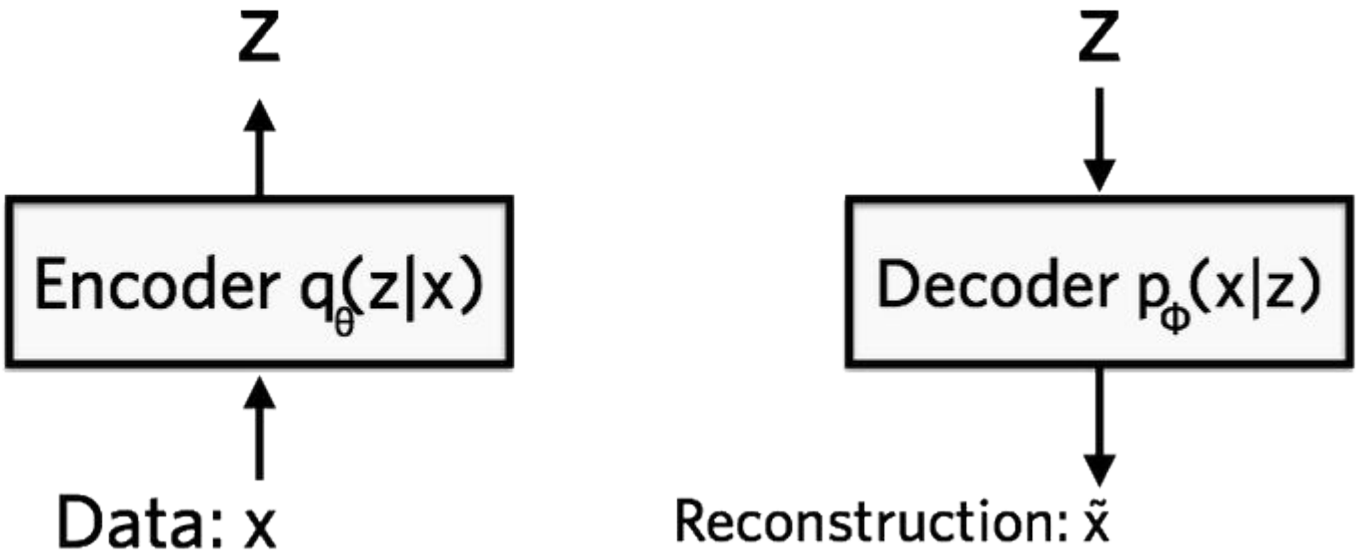
Kenneth Emeka Odoh

Variational Autoencoder (VAE) vs “Vanilla” Autoencoder

Autoencoder learns compressed representation of input by compressing and decompressing back to match to reconstruct the original input. It is an unsupervised learning method with goals to learn functions for compression and decompression.

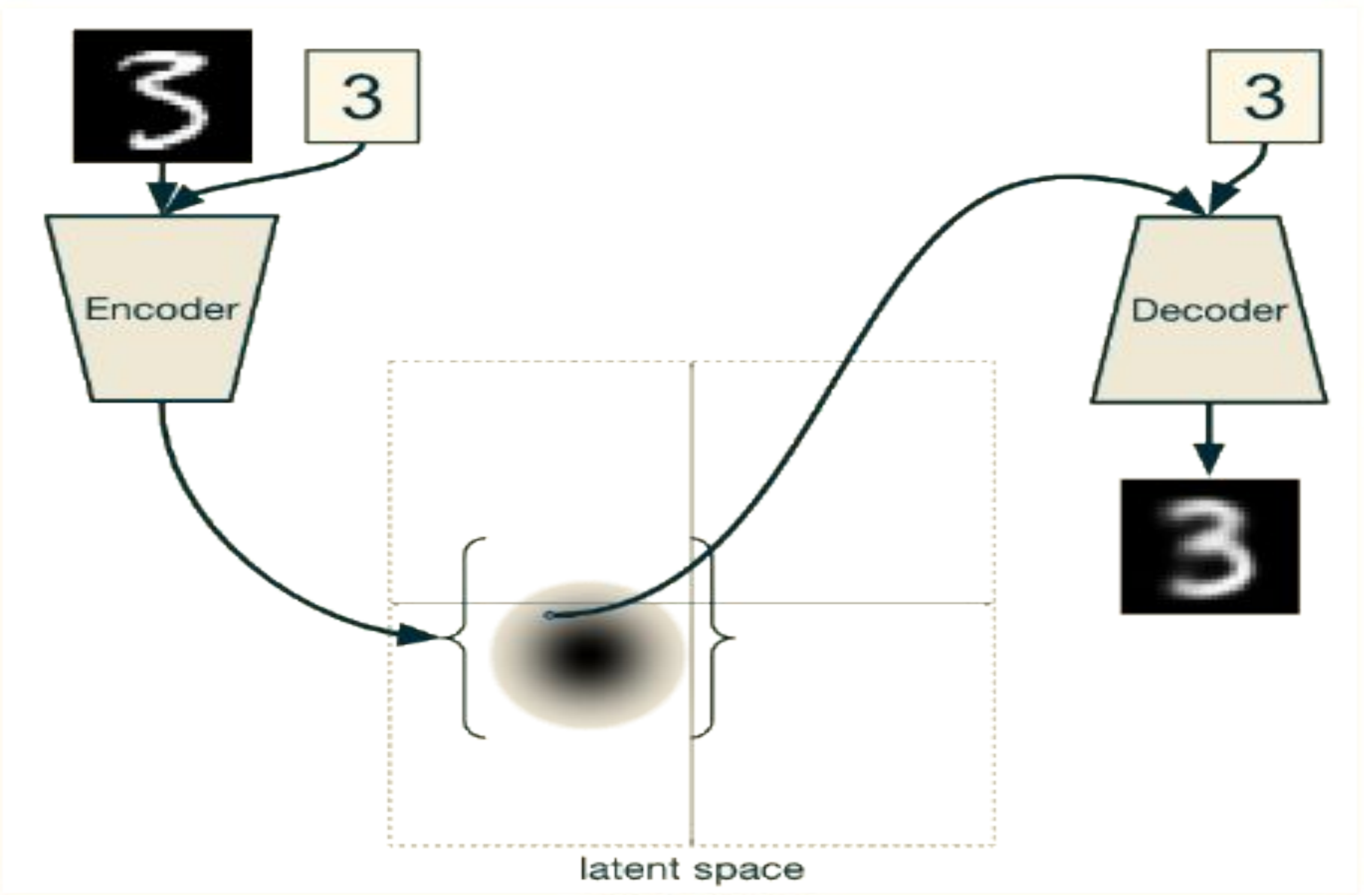
VAE learns probability of the parameter representing the data distribution. This focuses on learning probability distribution instead of learning functions.

VAE



[https://jaan.io/what-is-variational-autoencoder-vae-tutorial/]

Kenneth Emeka Odoh



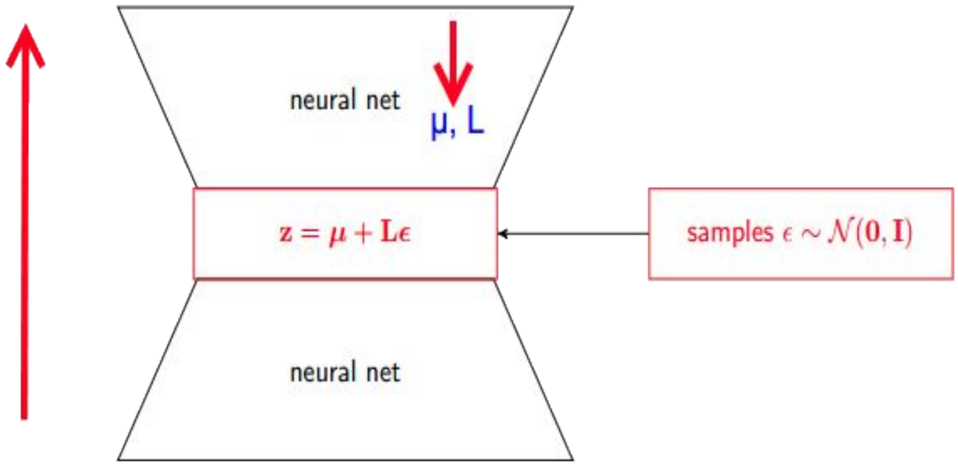
[https://cedar.buffalo.edu/~srihari/CSE676/20.10.3-VAE.pdf]

Reparameterization Trick

This is needed to backpropagate through a random node (latent node)

$$\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \quad \rightsquigarrow \quad \mathbf{z} = \boldsymbol{\mu} + \mathbf{L}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad \boldsymbol{\Sigma} = \mathbf{L}\mathbf{L}^T$$

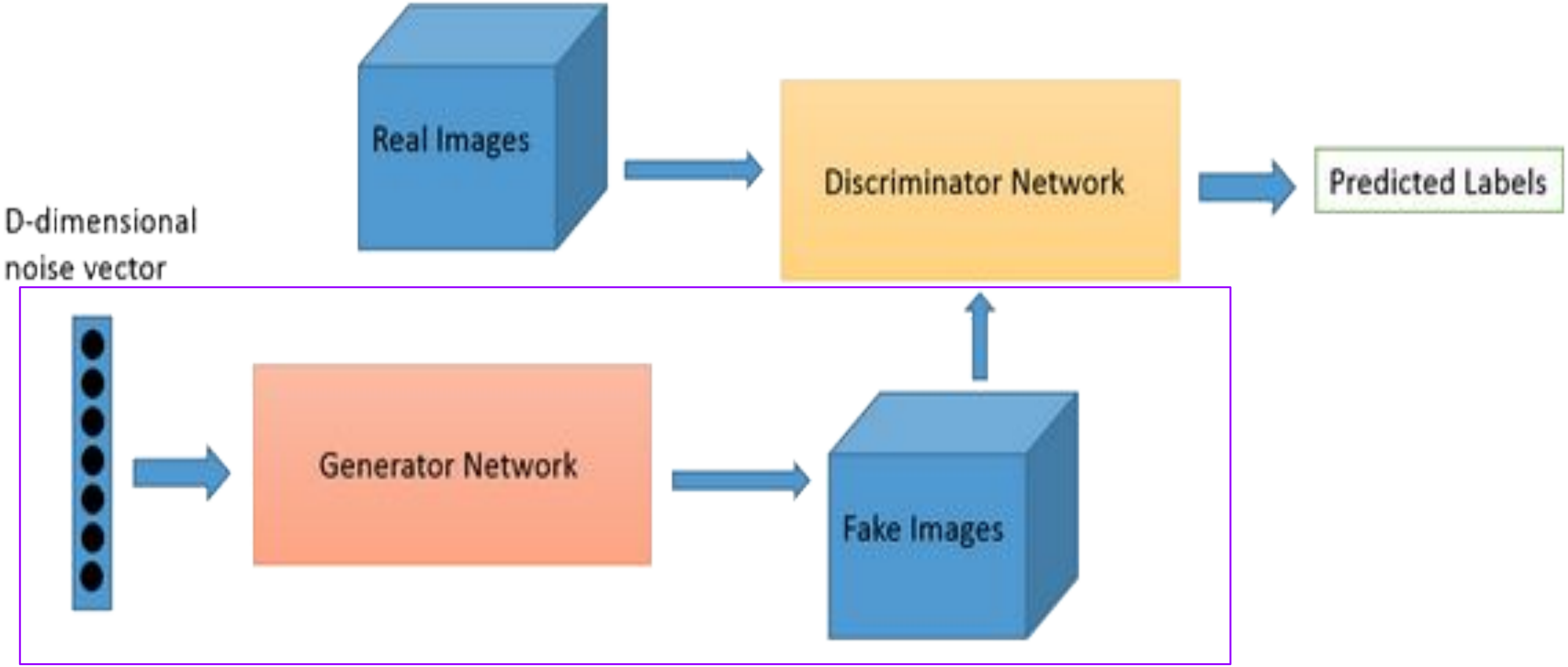
z= f(ϵ,μ,L)
Differentiate z with respect to μ



[https://stats.stackexchange.com/questions/199605/how-does-the-reparameterization-trick-for-vaes-work-and-why-is-it-important]

Kenneth Emeka Odoh

GAN



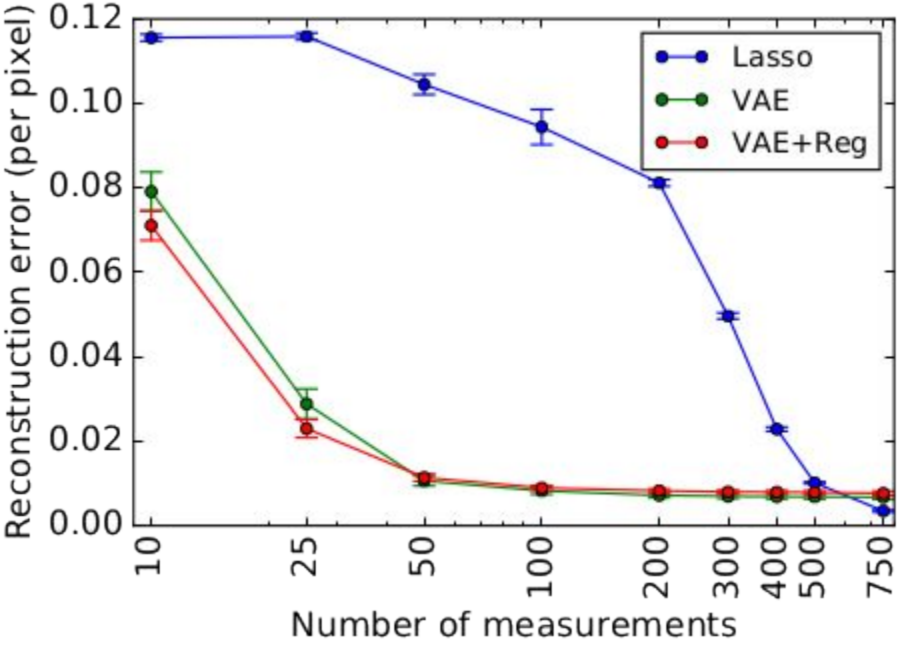
[https://deeplearning4j.org/generative-adversarial-network]

- This consist of a generator and discriminator pitted together in a zero-sum game.
- Generator models the data distribution.
- Discriminator estimate the probability that the sample result from the model distribution or the data distribution.
- The generator keeps creating counterfeit object and the discriminator keeps trying to detect the counterfeits. The process continue until the generator can produce real counterfeit that cannot be discriminated from the originals.
- Generative models and reinforcement learning model are a natural way to incorporate game theoretic principles in machine learning.

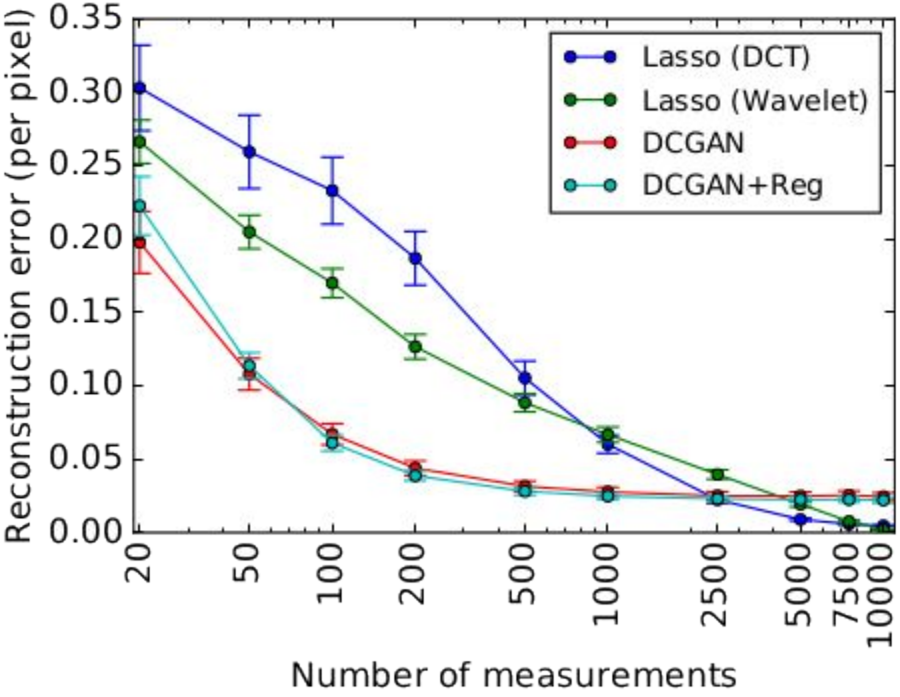
Why Generative model may requires lower number of Training examples

- Regularization is a form of inductive bias that exploit the structure of the problem thereby simplifying the problem space.
- With suitable representation learning with good priors, with smaller **VC-dimension** leads to smaller training size.
    - **VC-dimension** is a measure of capacity of space of functions.

Results

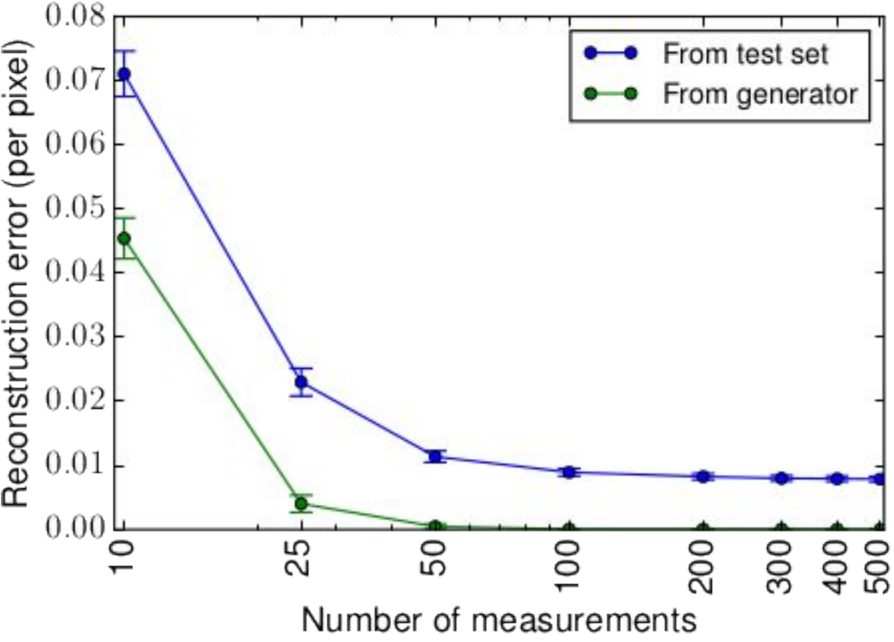


(a) Results on MNIST

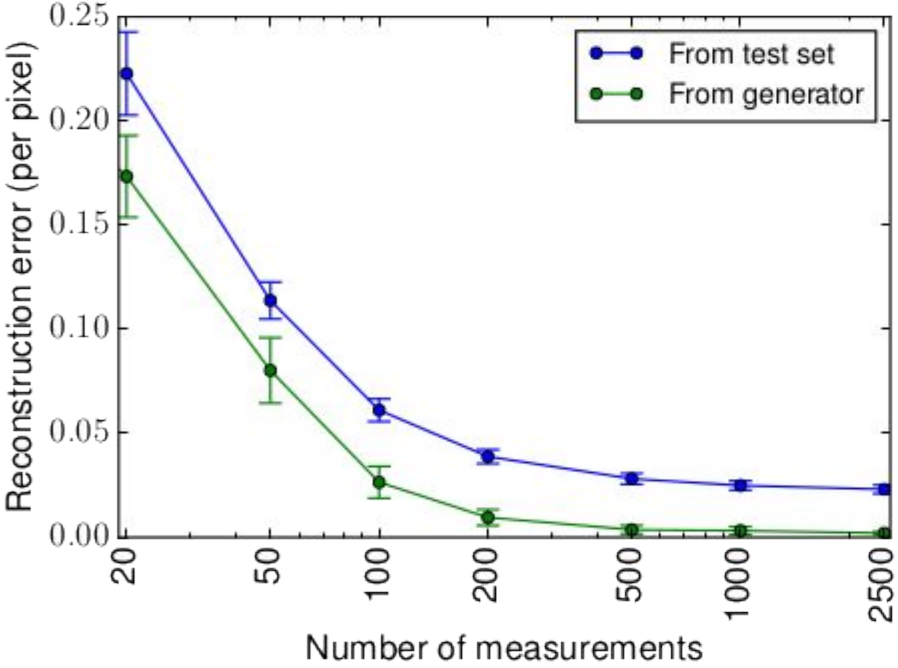


(b) Results on celebA

Kenneth Emeka Odoh



(a) Results on MNIST



(b) Results on celebA

Extensions of Work

- Use **Wasserstein** metric instead of **KL-divergence** in the VAE and GANs formulation. This would reduce error.
- Deciding on a rigorous theoretical justification for regularization leads to underfitting. This is relating to **bias-variance** tradeoff.
- How Early stopping as a regularizer can impact the experiments.
- Quantifying the uncertainty (aleatory and epistemic).
- How to evaluate generative models?
- Putting constraint on the output of the generative model e.g using sum of squares [https://www.wired.com/story/a-classical-math-problem-gets-pulled-into-self-driving-cars]

Conclusions

- Instability of lasso in the presence of multiple collinearity.
- Try elastic net regularizer.
- L2 lead to closed form solution leading to unique solution.
- L1 does not lead to a closed form, supports sparsity and suitable to feature selection.
- Representational learning with regularization may require less training size thereby lowering the training times with less complex models.

References

1. http://slazebni.cs.illinois.edu/spring17/lec12_vae.pdf
2. Carl Doersch, Tutorial on Variational Autoencoders, 2016.
3. Neelakantan et. al, Adding gradient noise improves learning for very deep network, 2016.
4. Bora et. al, Compressed sensing using generative models, 2017
5. Ian Goodfellow, NIPS 2016 Tutorial: Generative Adversarial Networks

Past Talks

Some of my past talks

- Tutorial on Cryptography, slide: https://www.slideshare.net/kenluck2001/crypto-bootcamp-108671356 , 2018
- Landmark Retrieval & Recognition, slide: https://www.slideshare.net/kenluck2001/landmark-retrieval-recognition-105605174 , video: https://youtu.be/YD6ihpBMyso , 2018
- Tracking the tracker: Time Series Analysis in Python from First Principles, slide: https://www.slideshare.net/kenluck2001/tracking-the-tracker-time-series-analysis-in-python-from-first-principles-101506045 , 2018
- WSDM Recommender System, slide: https://www.slideshare.net/kenluck2001/kaggle-kenneth , video: https://youtu.be/exwJmQzDBag , 2018