Software Engineering: OOP illustrated through Density Estimating Neural Networks

Albert Montillo, Austin Marckx

UTSouthwestern

Departments: Lyda Hill Department of Bioinformatics, Radiology, and Advanced Imaging Research Center

May 8-12, 2023

SWE course

Lyda Hill Department of Bioinformatics

Outline

1. **Monday**
   1. **Review OOP, Image I/O, Keras**
   2. **New topic: Object Oriented Variational Autoencoders (VAEs)**

2. **Tuesday**
   1. **Review observations on VAE**
   2. **New topic: Symbolic debugger: cond breakpoints and call stack traversal**
   3. **New topic: Object Oriented Generative Adversarial Networks (GANs)**

3. **Wednesday**
   1. **Review GAN observations**
   2. **New topic: Object Oriented *conditional* VAEs (cVAE) and *Auxillary Classifier* GANs (AKA cGAN or acGAN)**
   3. **New topic: Motivate a possible combination of cVAE and cGAN**

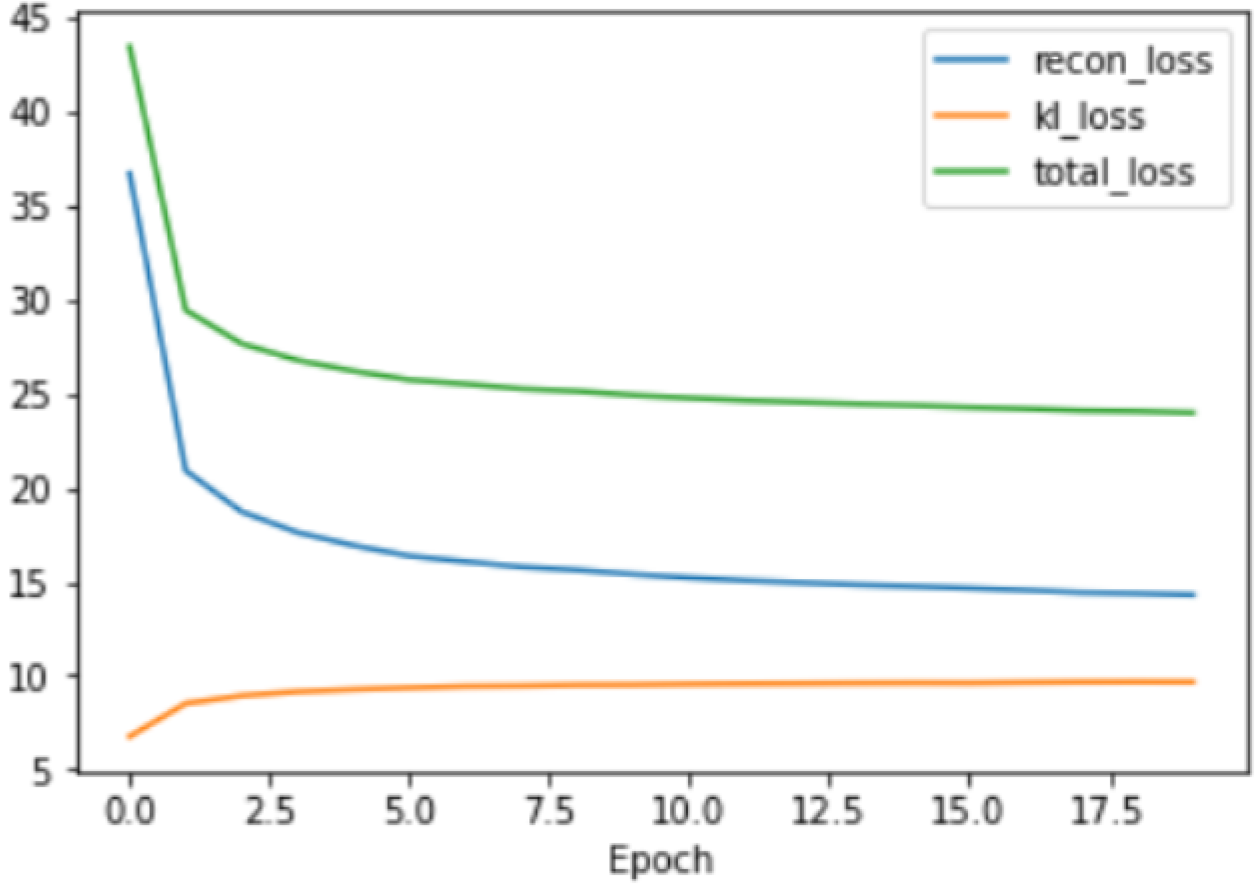
4. **Thursday**
   1. **Review cVAE, cGAN observations**
   2. **Review cVAE-cGAN code**
   3. **New topic: Hyperparameter optimization**
   4. **New topic: training curve and latent space traversal and visualization**

Observations from cVAEs

3

Conditional VAE (cVAE) Training curves

1. Shows evolution of reconstruction and regularizing prior (D_{KL}) loss as well as the total loss (their sum)
2. We observe that the majority of the learning took place in the first 10 epochs. We could train longer but diminishing returns



Conditional VAE Reconstructed images at epoch 1

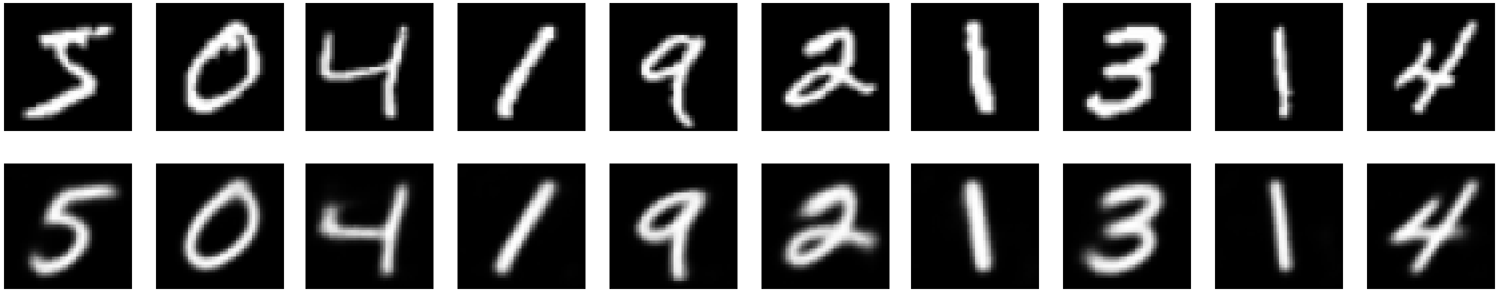
Real:

Recon:

Conditional VAE Reconstructed images at epoch 10

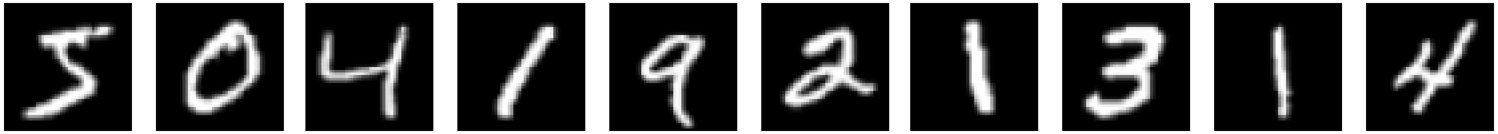
Real:

Recon:

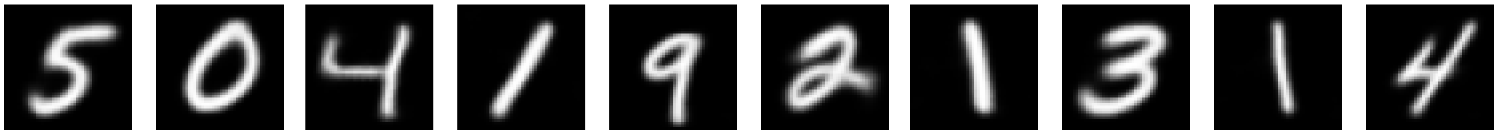


Conditional VAE Reconstructed images at epoch 20

Real:

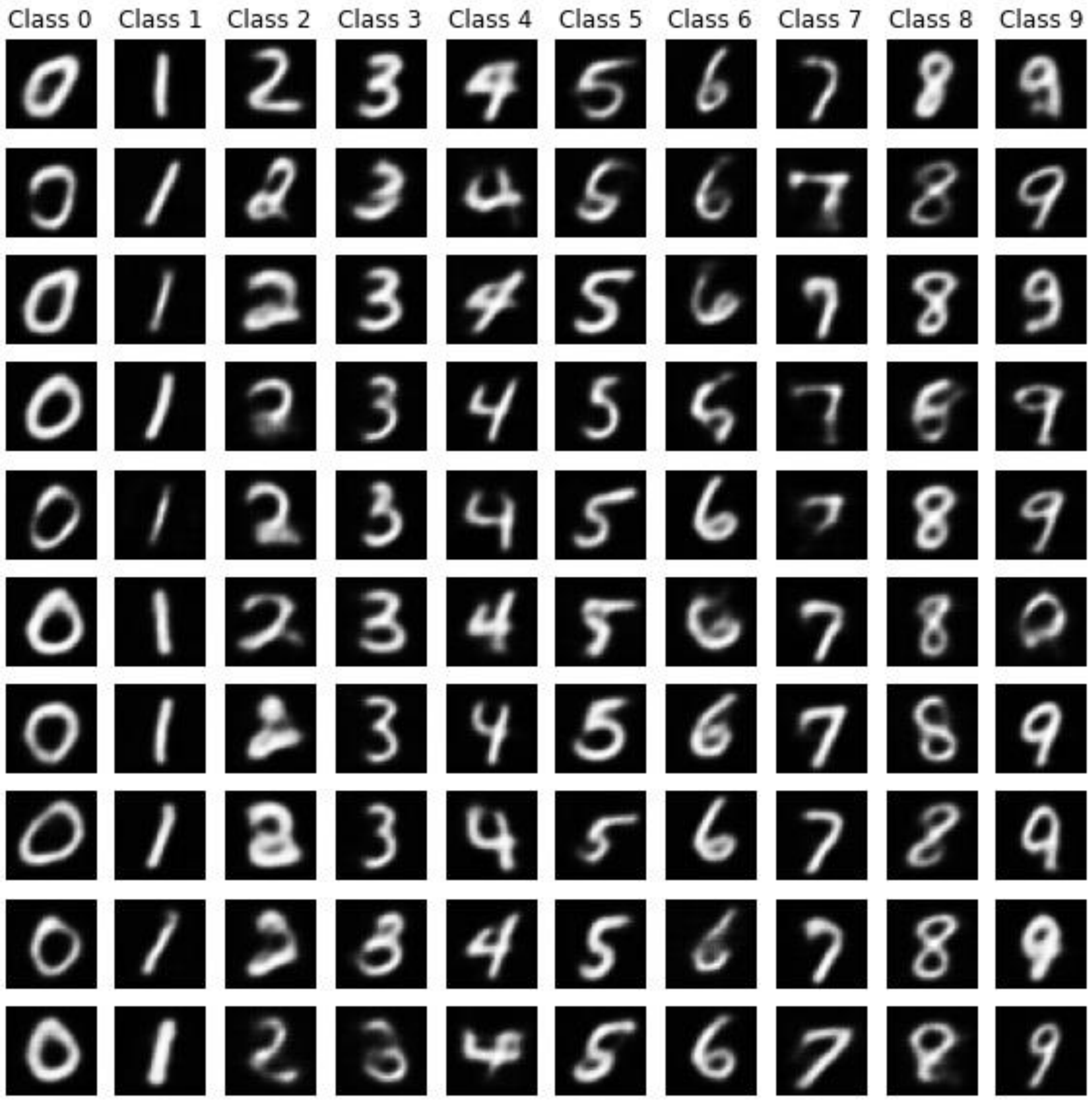


Recon:



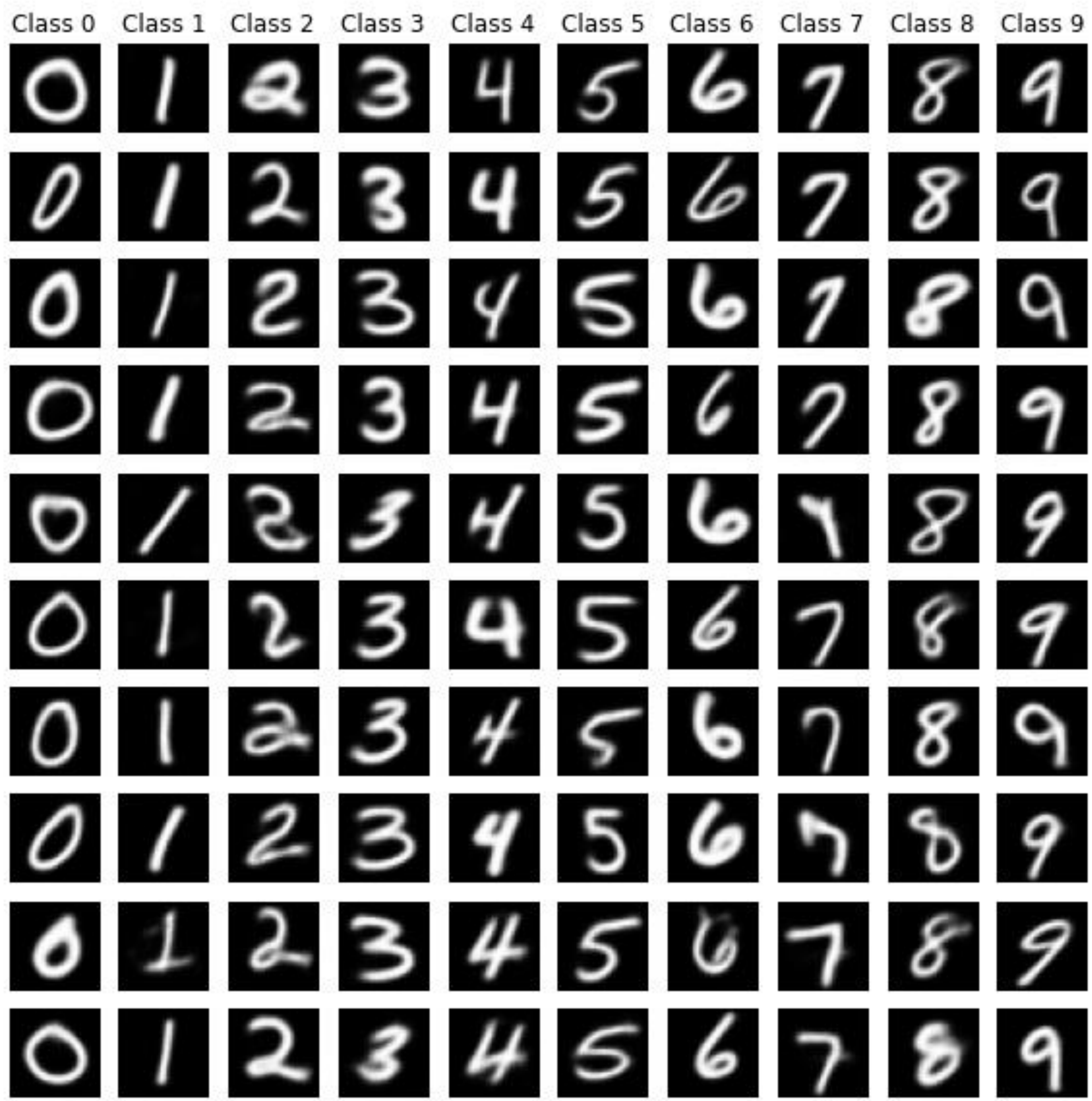
Conditional VAE Purely synthesized images at epoch 1

Note: Steerable class label (column), 100 different random z’s



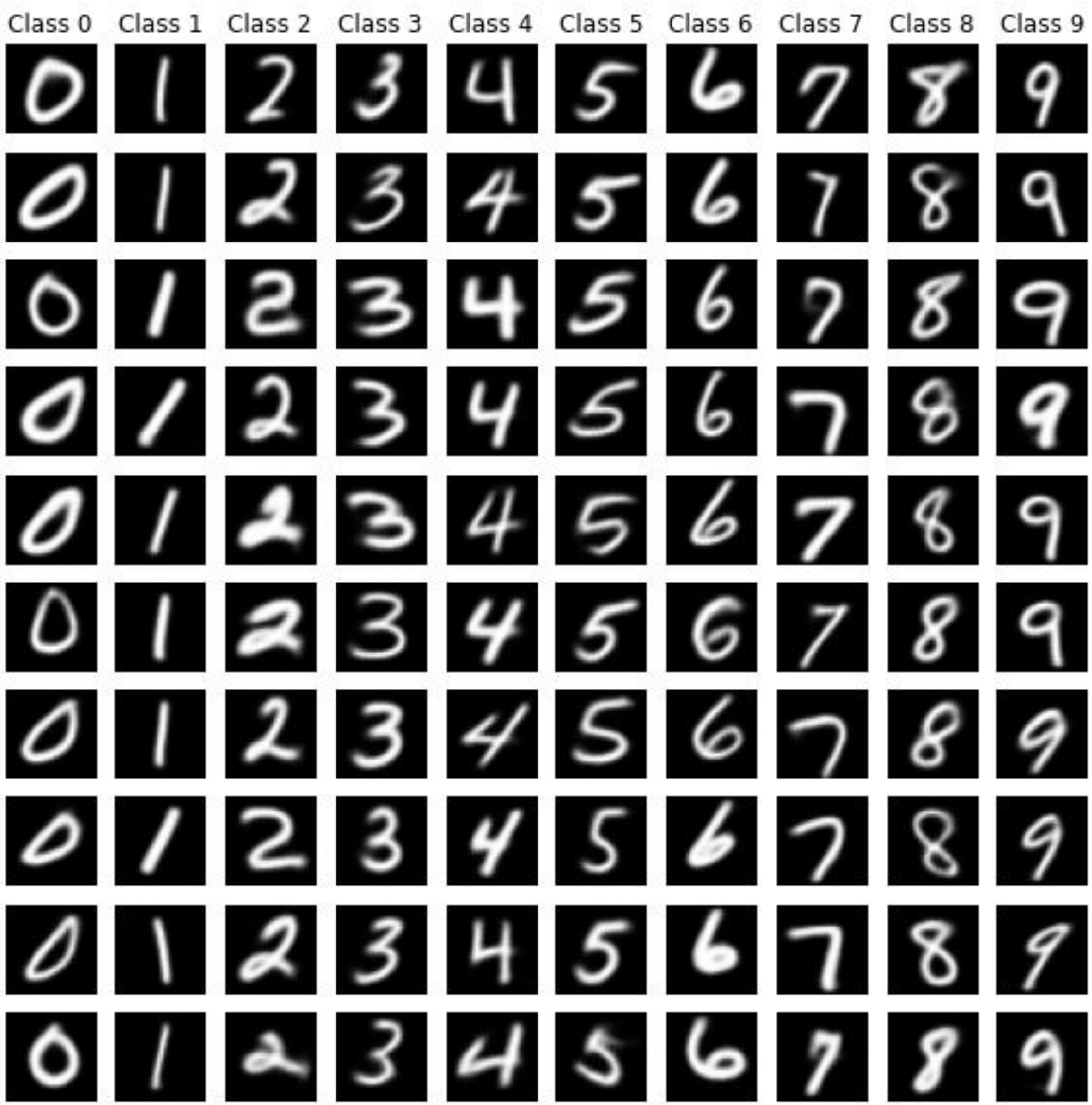
Conditional VAE Purely synthesized images at epoch 10

Note: Steerable class label (column), 100 different random z’s



Conditional VAE Purely synthesized images at epoch 20

Note: Steerable class label (column), 100 different random z's

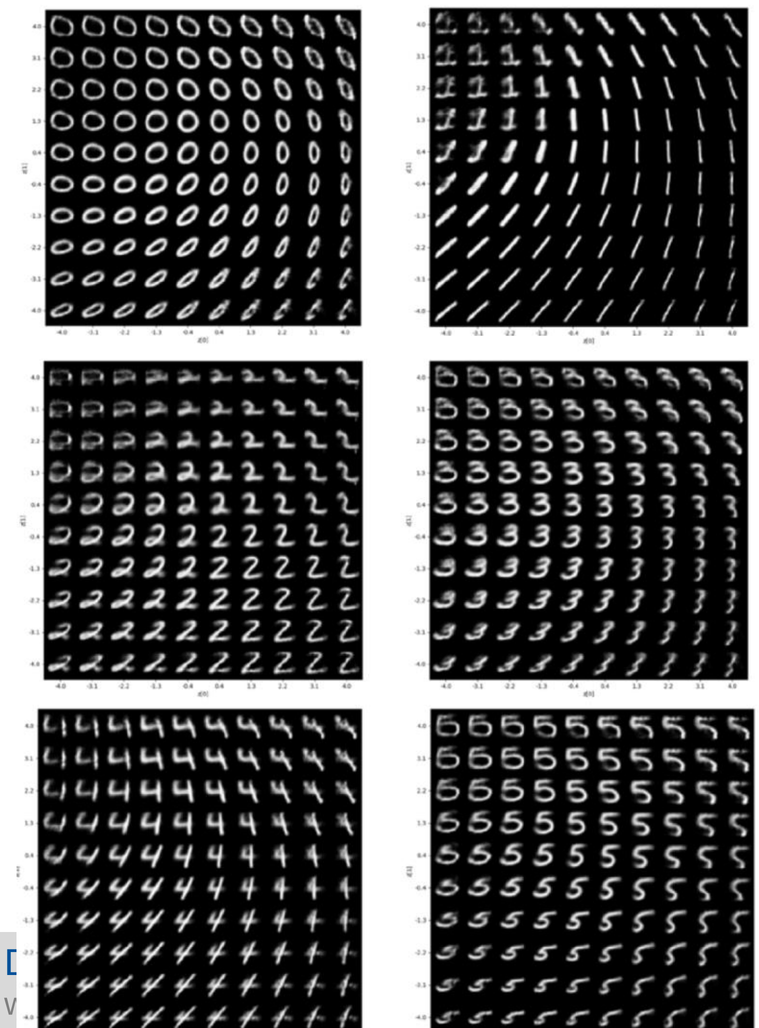


Upshot of cVAE results

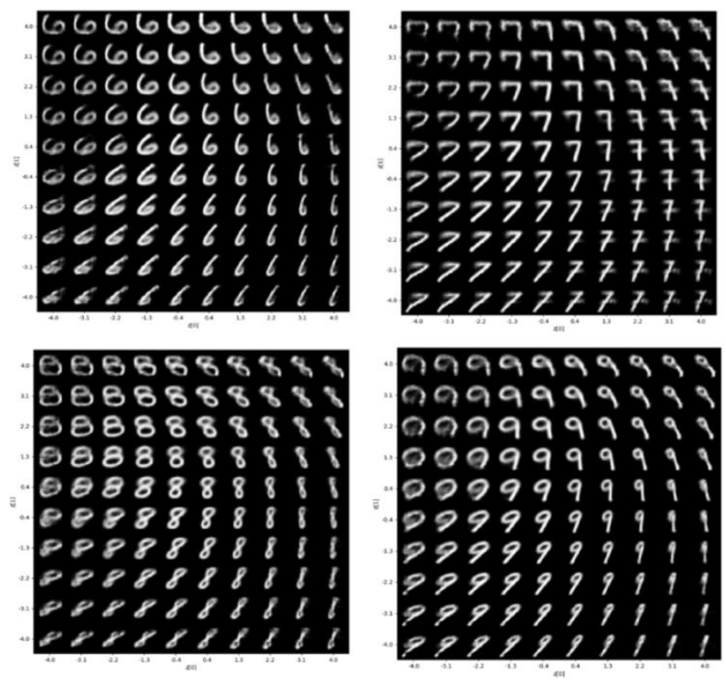
- **Good digits are produced**
- **Strokes are nice and straight, not wavy (better than GAN)**
- **We can now specify on demand which digits to produce (like a conditional GAN)**

Upshot of cVAE results

- We can also traverse the Z space of individual digits, which is smooth and contiguous through the prior we enforced. (better than plain VAE, and GAN simply cannot)
- Digits 0-5



Digits 6-9



12

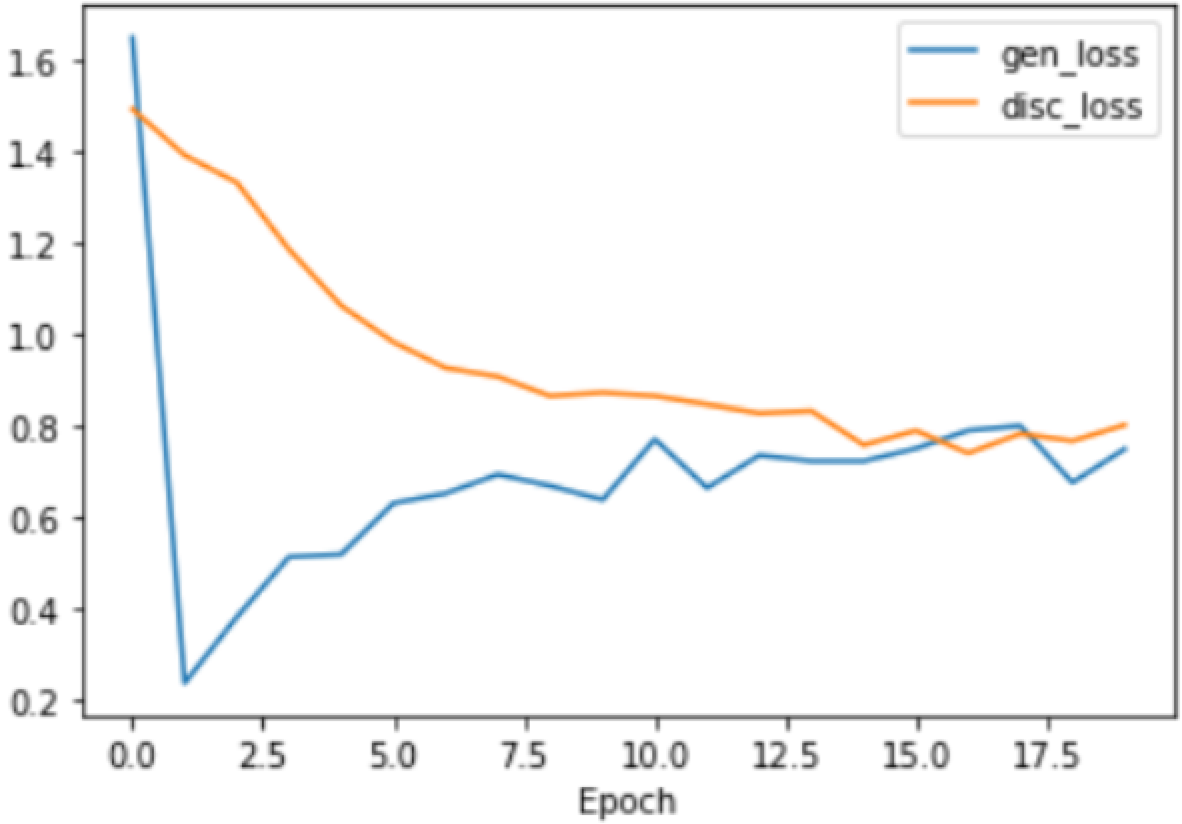
Upshot of cVAE results

- **Borders of digits are still blurry (worse than a GAN)**

- **Wish we could get the best of both worlds.**
- **We can…**
  - **By constructing a cVAE~cACGAN model**

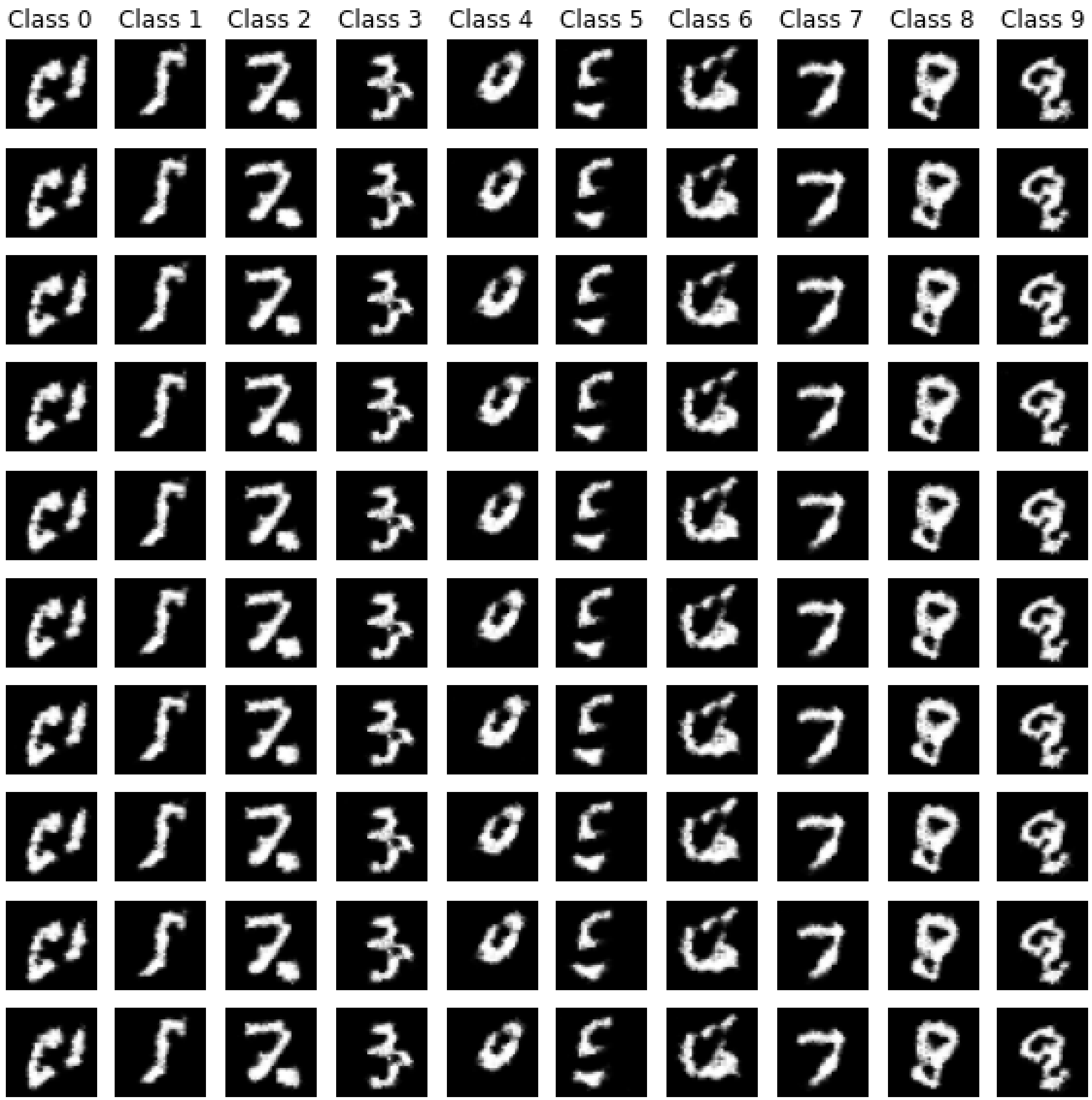
ACGAN Training curves

1. Shows evolution of generator and multitask discriminator losses
2. We observe that the two compete and reach an equilibrium (middle ground). Nice convergence!

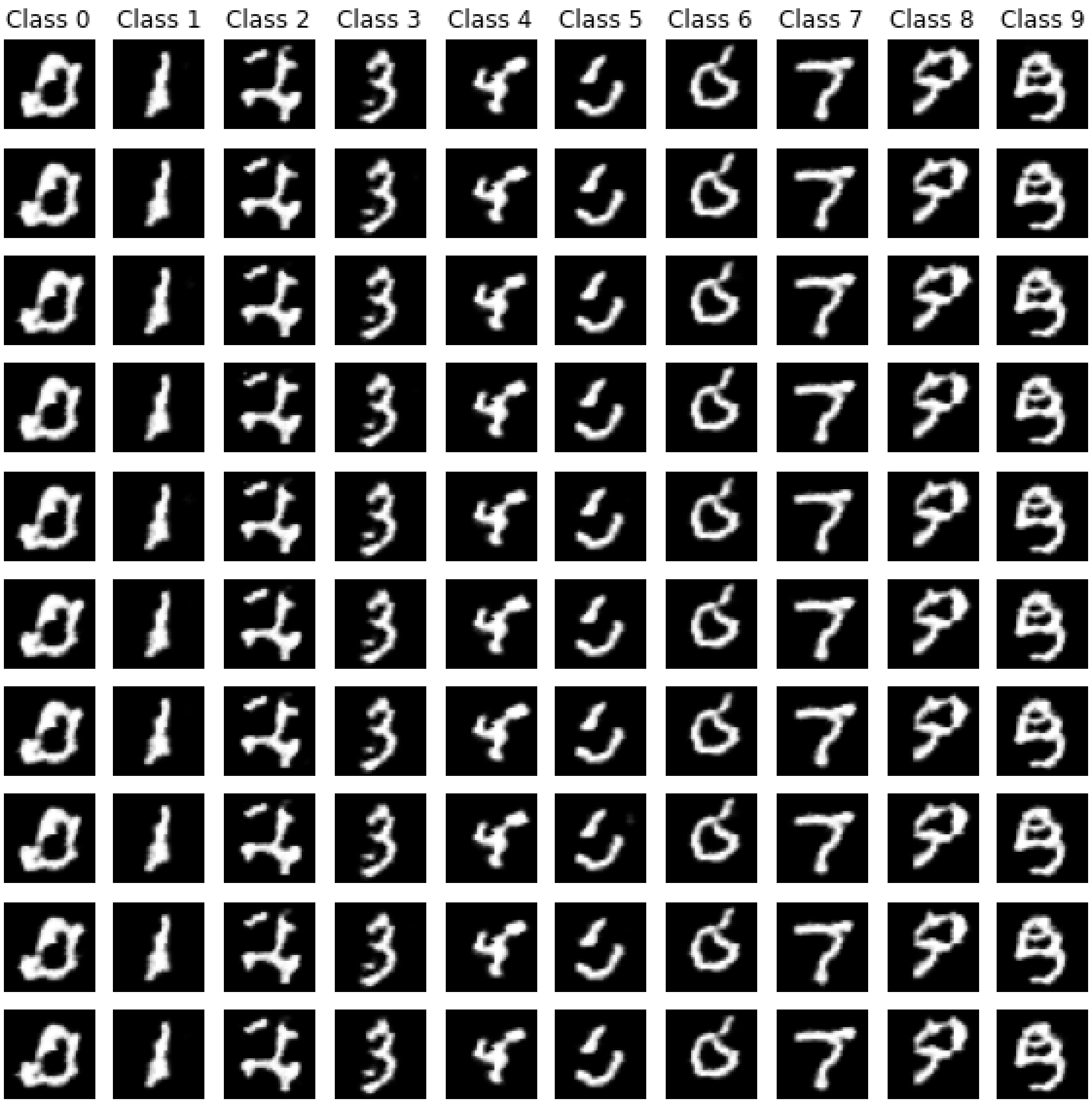


ACGAN Synthesized images at epoch 1

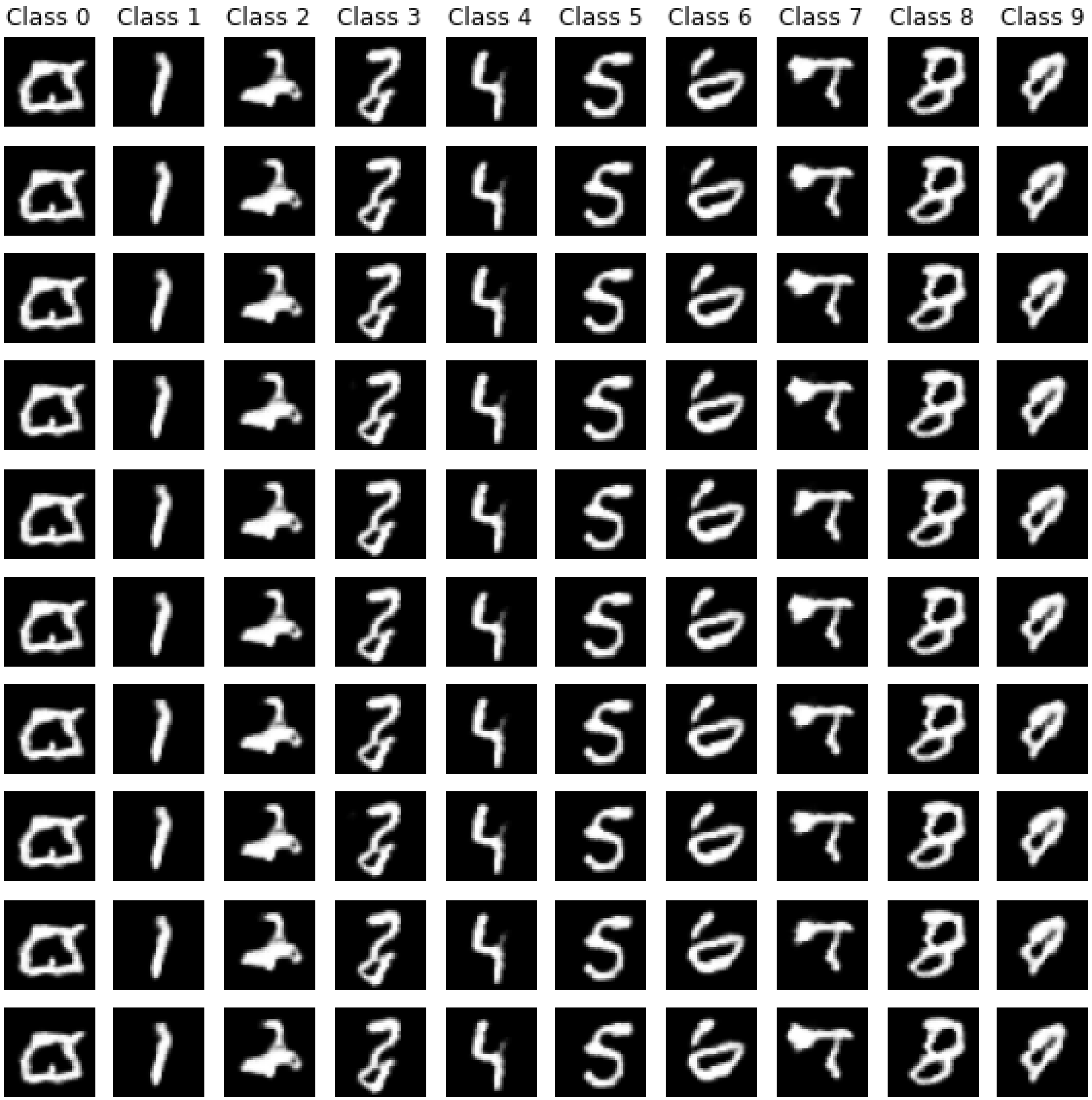
ACGAN Synthesized images at epoch 5



ACGAN Synthesized images at epoch 10



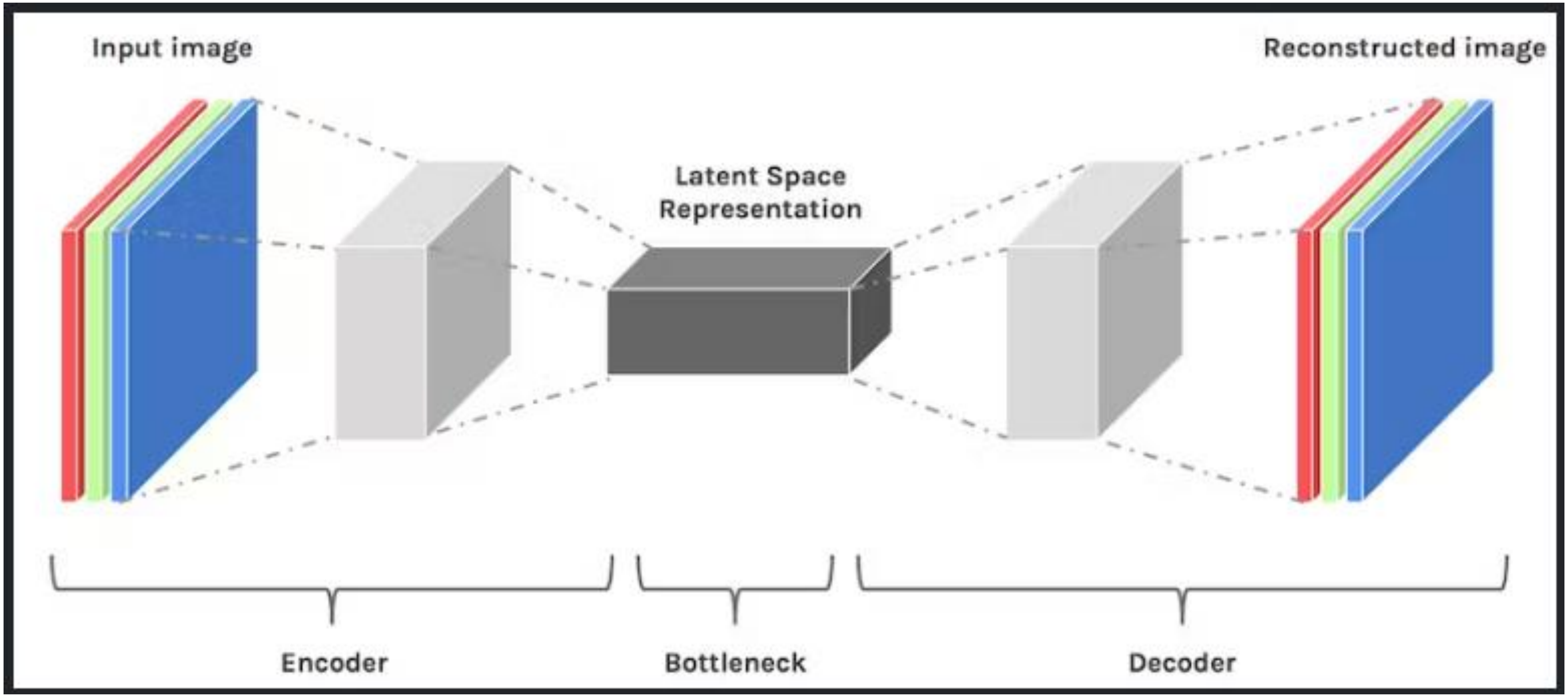
ACGAN Synthesized images at epoch 20



Upshot of ACGAN

- Easier to get it to converge
- Cleary better results
- Borders of digits are sharp not fuzzy
- The quality is OK, perhaps not great… strokes are wavy

Latent space – what is it?

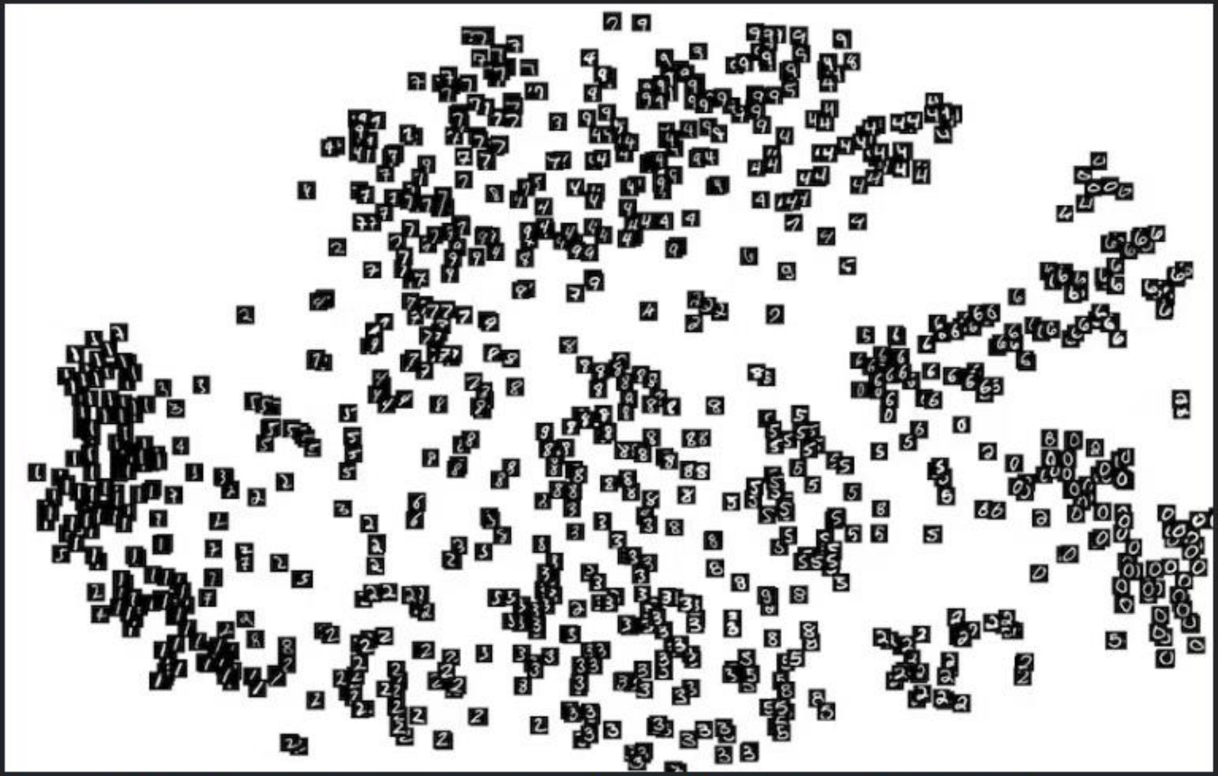


https://hackernoon.com/latent-space-visualization-deep-learning-bits-2-bd09a46920df

Latent space – why do we care?

Projecting the pixel space

Let's start by plotting the t-SNE embedding of our dataset (from image space) and see what it looks like.



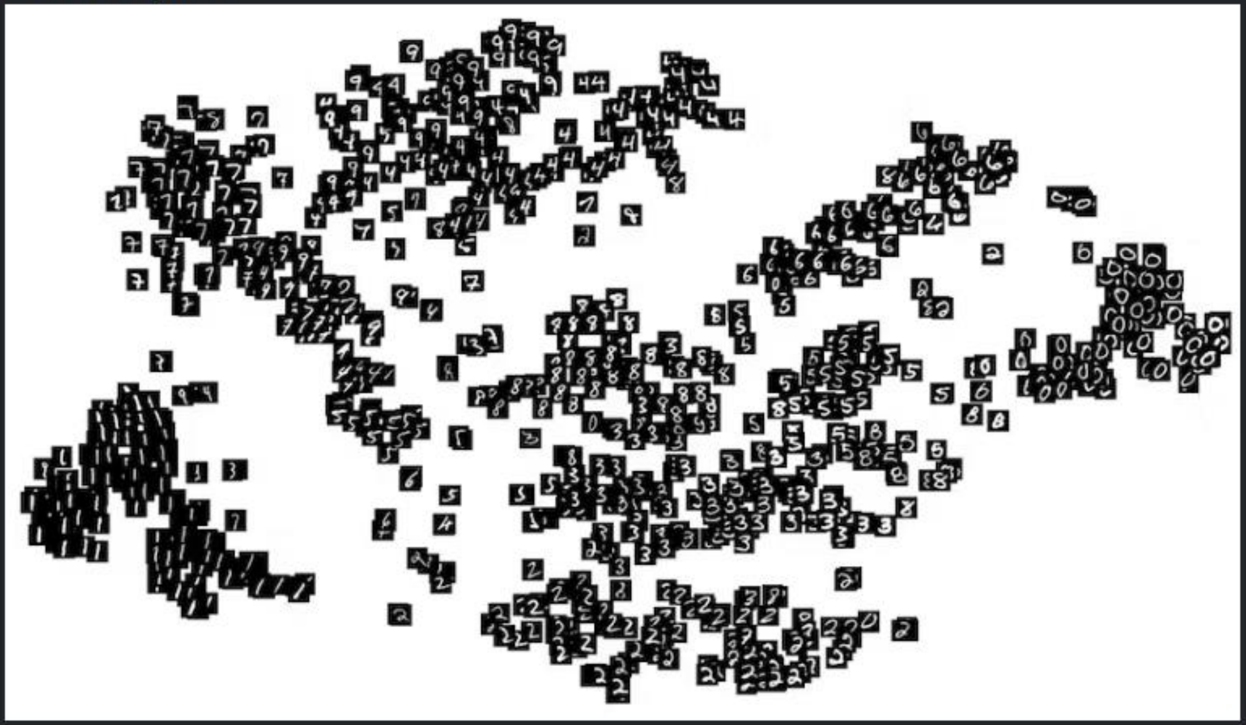
t-SNE projection of **image space** representations from the validation set

https://hackernoon.com/latent-space-visualization-deep-learning-bits-2-bd09a46920df

Latent space – why do we care?

Projecting the latent space

We know that the *latent space* contains **a simpler representation** of our images than the pixel space**,** so we can hope that t-SNE will give us an interesting **2-D projection of the latent space**.



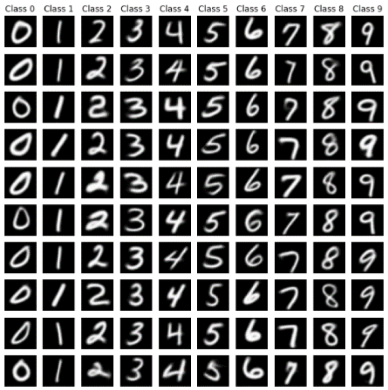
t-SNE projection of **latent space** representations from the validation set

https://hackernoon.com/latent-space-visualization-deep-learning-bits-2-bd09a46920df

Latent space – How can we explore it?

There are many ways, but a few examples:

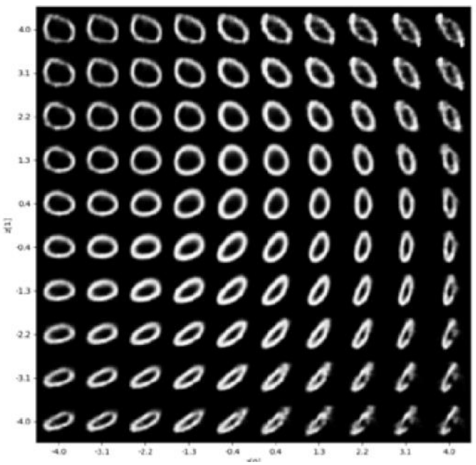
- **Random Samples + recon**



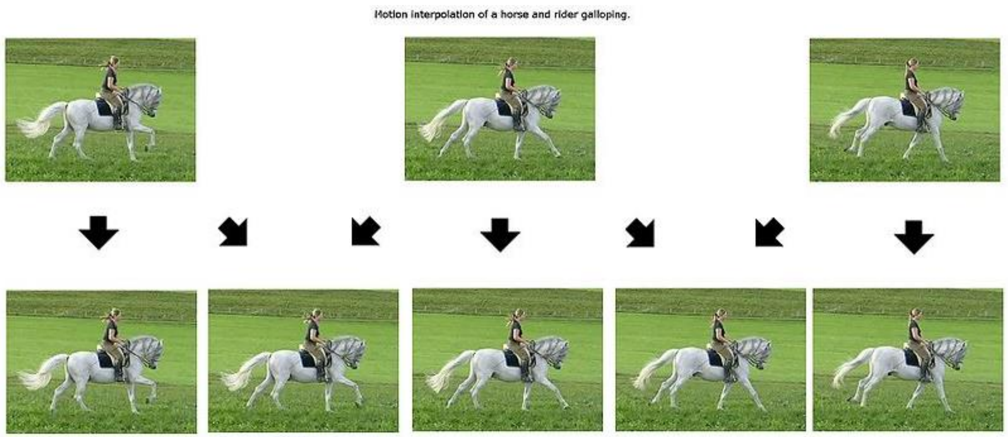
- **Input embedding:**

https://hackernoon.com/latent-space-visualization-deep-learning-bits-2-bd09a46920df

- **Grid search**



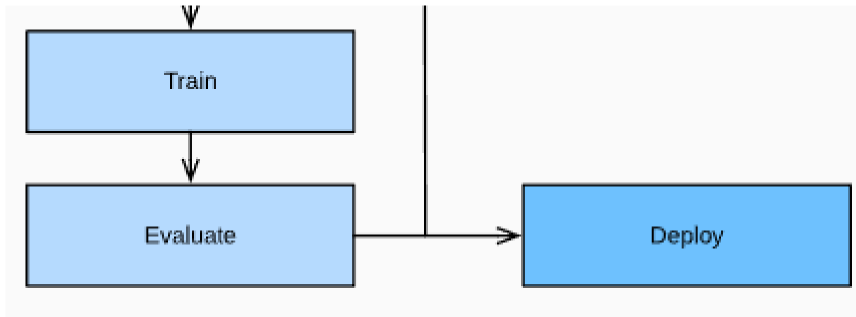
- **Interpolation:**

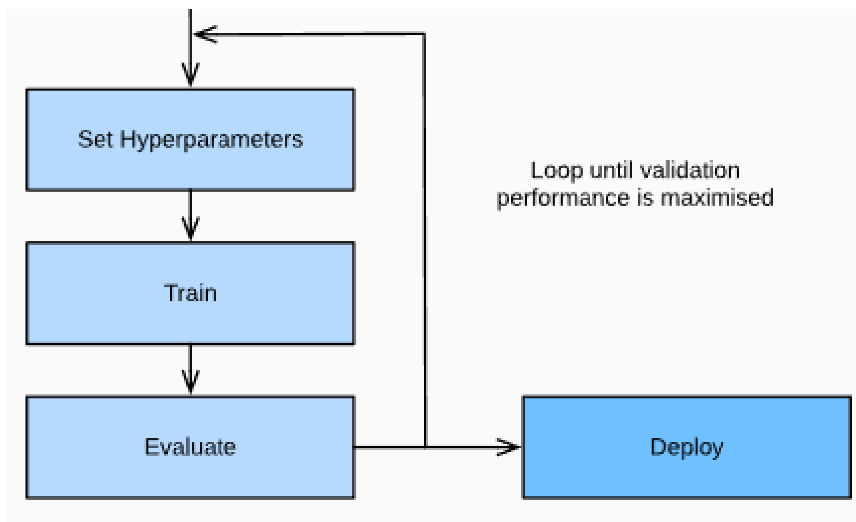


https://commons.wikimedia.org/wiki/File:Motion_interpolation_example.jpg

Hyperparameter Optimization (HPO)

- **So we've trained and evaluated our model… are we done?**
  - **Probably not**

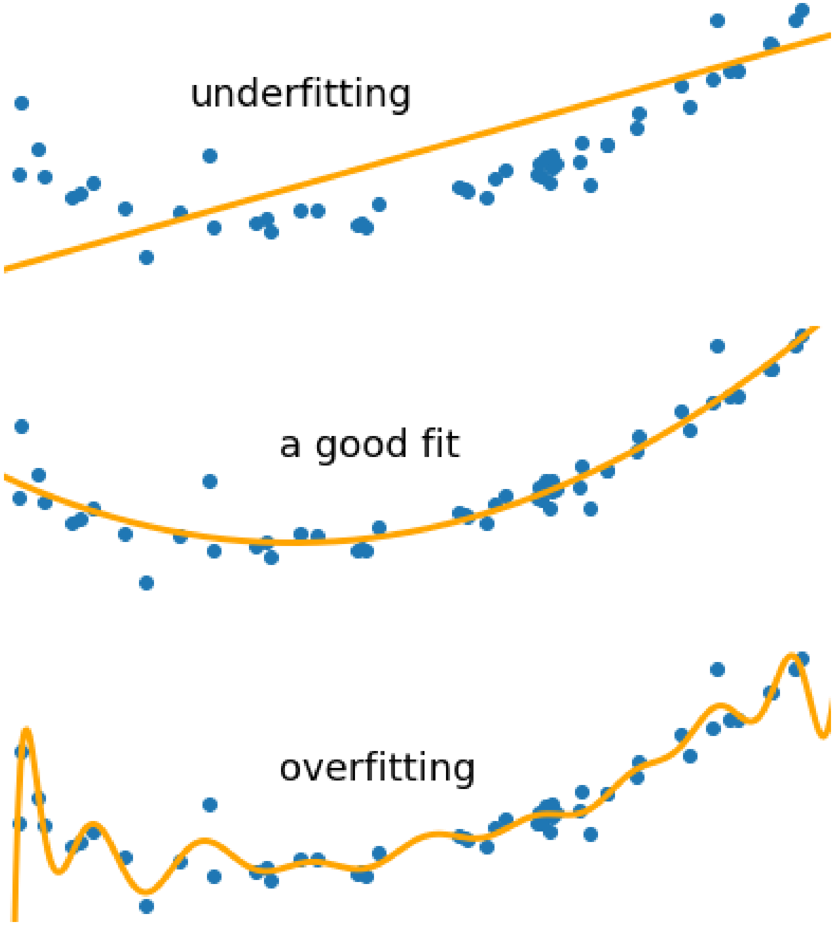




https://d2l.ai/chapter_hyperparameter-optimization/hyperopt-intro.html

Why HPO?

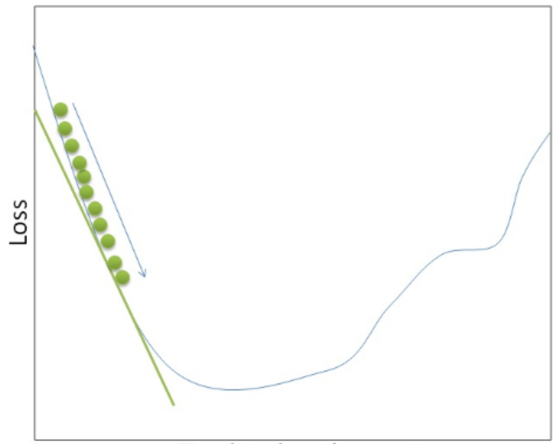
Hyperparameter optimization is a means of tuning our model to make it more generalizable.



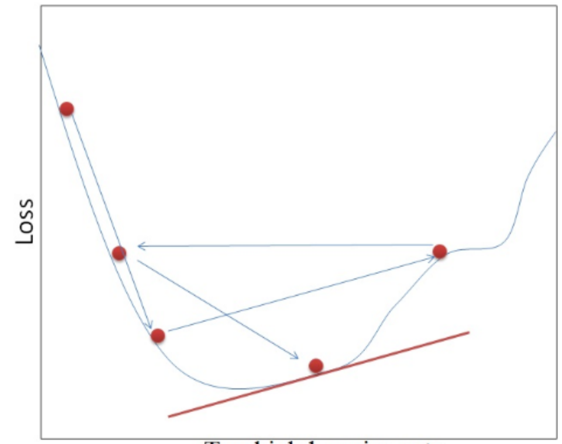
https://bookdown.org/gmli64/do_a_data_science_project_in_10_days/models-underfitting-and-overfitting.html

What can we HPO?

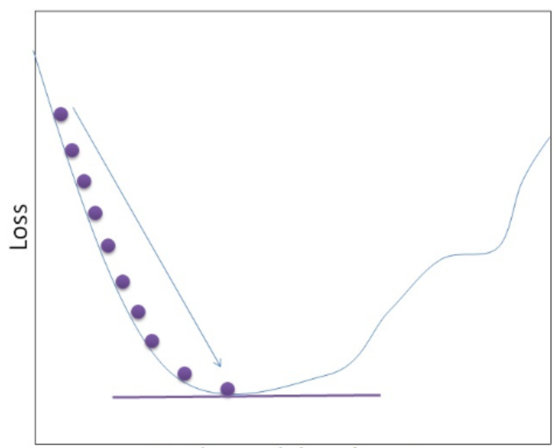
Let's take for example learning rate:



Too low learning rate



Too high learning rate



Good enough learning rate

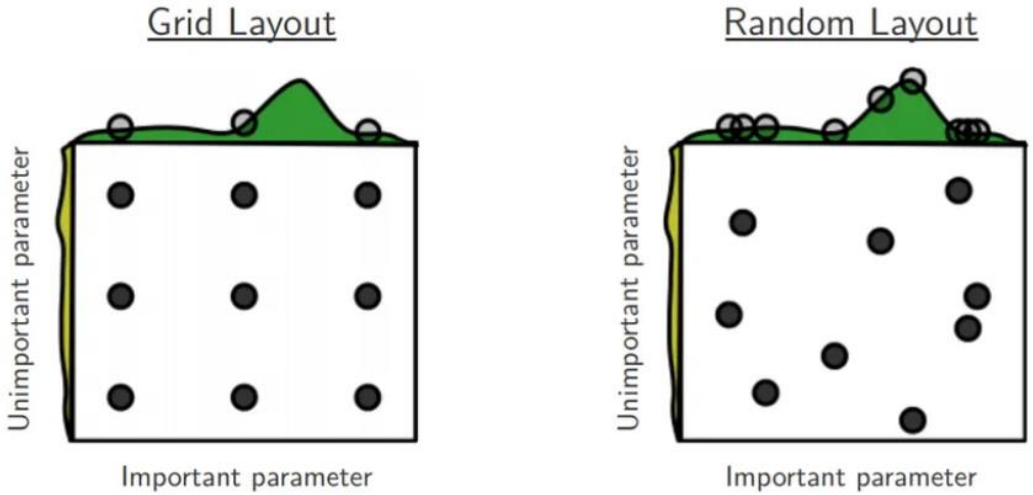
https://www.analyticsvidhya.com/blog/2021/05/tuning-the-hyperparameters-and-layers-of-neural-network-deep-learning/#:~:text=The%20hyperparameters%20to%20tune%20are,layers%20can%20affect%20the%20accuracy

26

How can we HPO?

Hyperparameter optimization, there are many methods but some of the simplest:

- **Manual**
- **Grid**
- **Random**



The 'world famous' grid search vs. random search illustration by James Bergstra James, Yoshua Bengio on "Random Search for HyperParameter Optimization" (http://www.jmlr.org/papers/volume13/bergstra12a/bergstra12a.pdf)

https://towardsdatascience.com/hyperparameter-tuning-explained-d0ebb2ba1d35

- **For your exercises, you've been asked to perform a grid search of at least 2 hyper parameters within given ranges:**

HPO/Latent Space Exercise:

HPO Tuning:

For a cGAN, a cVAE, or a cVAE-cGAN (choose one):

- Look at the provided code associated with your chosen model.
  - Does it differ from the models you've implemented earlier this week?
    - *HINT:* Look at __init__() and call() methods
  - If so, what are the implications of the differences?
- Using the provided 'hpo_mnist.ipynb', implement a grid search for 2 hyperparameters of your choosing (suggestions for ranges are provided in the notebook).
- Using the provided 'image_viewer.ipynb', what impact did your HPO tuning have on your digits (fakes and/or recons)?

Latent space exploration:

- Implement a visualization appropriate for exploring the latent space. Using this visualization, Explore the meaning of the latent space for at least three different MNIST digits.

Acknowledgements


Albert Montillo, PhD, PI

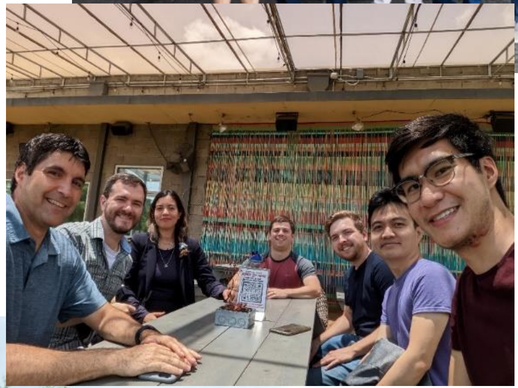

Son Nguyen, PhD Postdoc


Alex Treacher PhD student


Aixa Andrade Hernandez, MS, PhD student

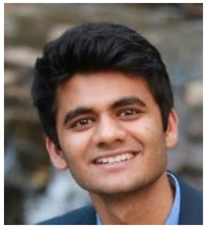

Austin Marckx PhD student


Krishna Chitta Res. Sci.



------------ Recent Alumni -------------


Atef Ali Undergrad


Vyom Raval, BS MD/PhD


Kevin Nguyen MD/PhD student


Cooper Mellema MD/PhD student

Lab Funding

- **NIH/ NIGMS R01** ***Correcting Biases in Deep Learning***
- **King Foundation (PI) : Quantitative AD diagnostics.**
- **Lyda Hill Foundation (PI): Quantitative prognostics of Parkinson's disease**
- **NIH/ NIA R01 Blood Biomarkers for Alzheimer's and Parkinson's**
- **TARCC : Texas Alzheimer's Research and Care Consortium.**
- **NIH / NINDS F31 fellowship : Causal connectivity biomarkers for neurological disorders**

Thank you!

Email: Albert.Montillo@UTSouthwestern.edu

Github: https://github.com/DeepLearningForPrecisionHealthLab

MegNET …. Artifact suppression
BLENDS …. fMRI augmentation
Antidepressant-Reward-fMRI …. response prediction
Parkinson-Severity-rsfMRI … disease trajectory prediction

End of presentation