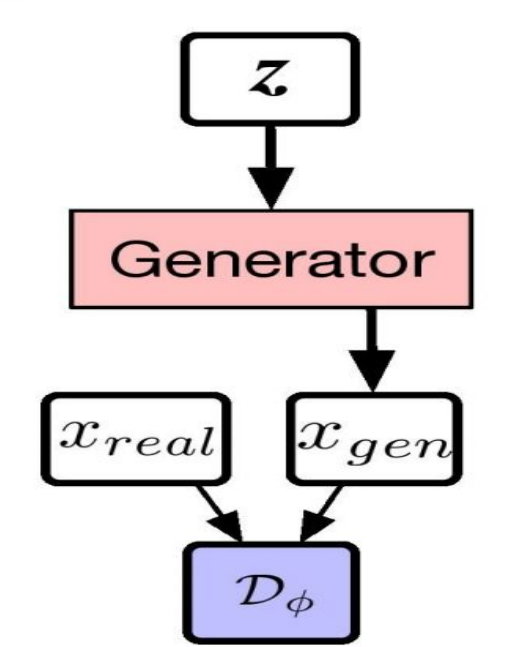
Generative Adversarial Networks for Particle Physics Applications

MENTORS -
SERGEI GLEYZER
MANOS STERGIADIS
LORENZO MONETA
OMAR ZAPATA
GERARDO GUTIERREZ

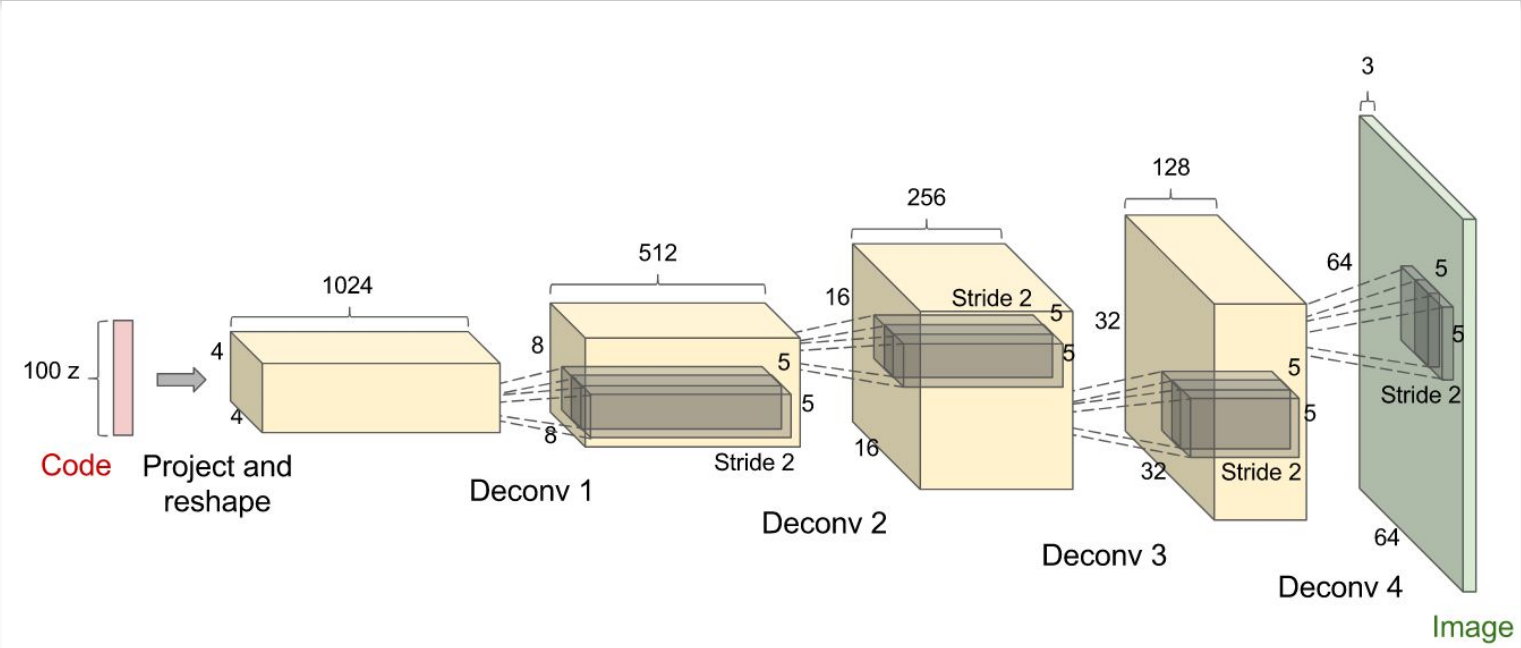
PRESENTED BY -
ASHISH KSHIRSAGAR

Overview of GANs

- **MiniMax game :** Adversarial training between *Generator* and *Discriminator*.
- **Generator :** Generate fake images that fool the discriminator.
- **Discriminator :** Distinguish between real and fake images.

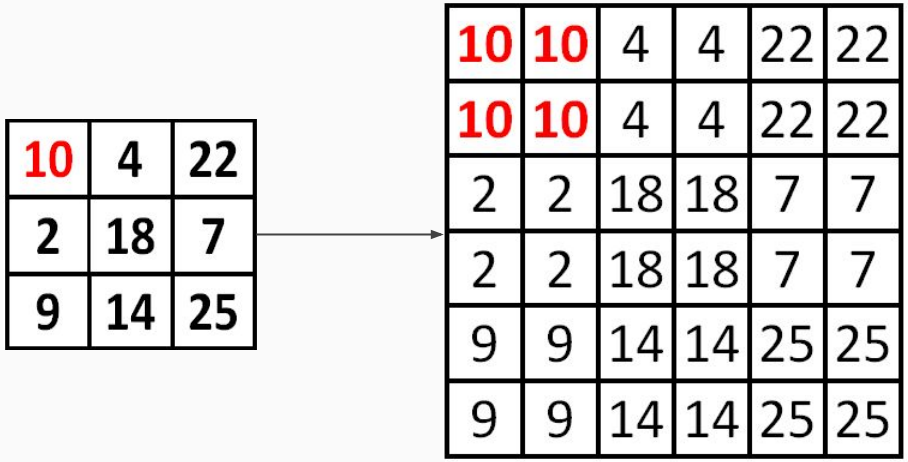


Need for Deconvolution



Upsample Layer

- Layer without weights/ filters.
- Like ‘Pooling’ in Convolution.
- Generates an increased dimension matrix.
- Implemented Nearest Neighbor interpolation supporting batch input.
- Used along with Convolution Layer in generative models.
- Unit Tests passing.

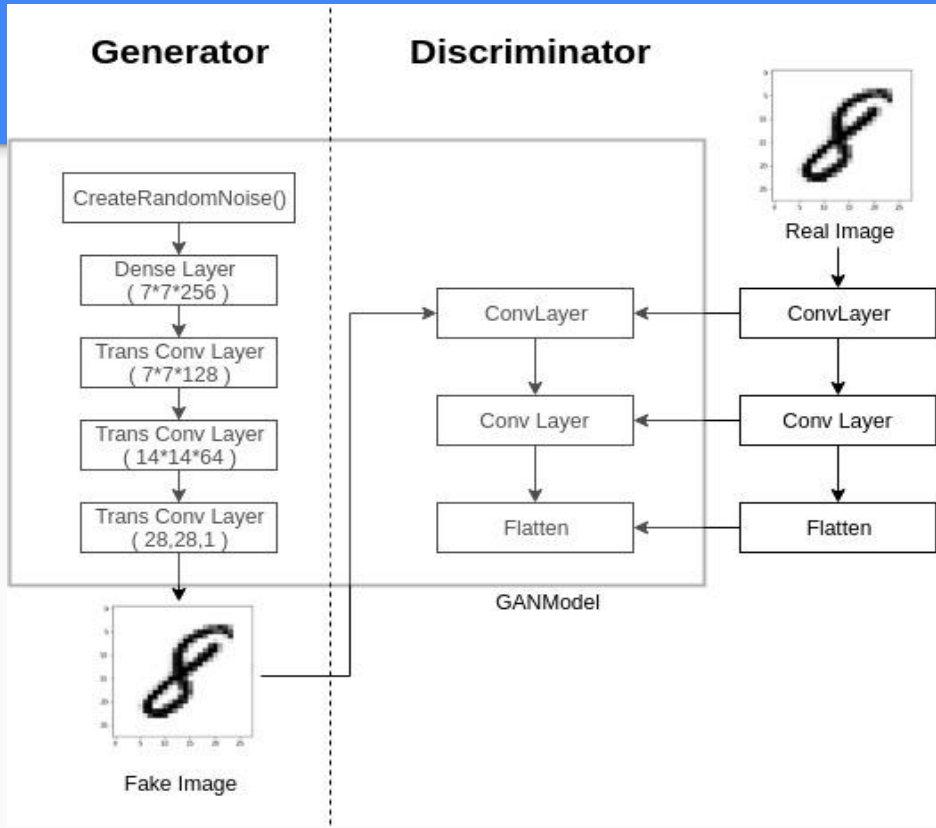


Transpose Convolution Layer

- Performs operations similar to a normal convolution layer in backward direction.
- Implemented forward and backward passes for CPU Architecture.
- Steps
  - Input matrix to input columnar vector.
  - Transpose Convolution Matrix from the given kernel / filters.
  - Compute the output columnar vector.
- Unit Tests passing.

TMVA GANs module

- Designed MethodGAN class with GANs framework.
- Parsing Layouts for
  - ConvLayer & TransConvLayer
  - Upsample & Pooling Layer
  - Input, Batch & Network



Future Work

- Design separate loss functions for Generator and Discriminator.
- Adding support for other variations of GANs for high energy physics applications.
- Benchmarking the results with other standard implementations.

Links

- Final Blog - https://medium.com/@ashishkshirsagar10/cern-gsoc19-generative-adversarial-networks-for-particle-physics-applications-c3da13a3f44b
- Pull Requests
  - Addition of layer support for GANs - https://github.com/root-project/root/pull/4164
  - GANs implementation - https://github.com/Ask149/root/tree/dev/ashish/temp
- Other PR's - https://github.com/root-project/root/pull/4275

Thank you!