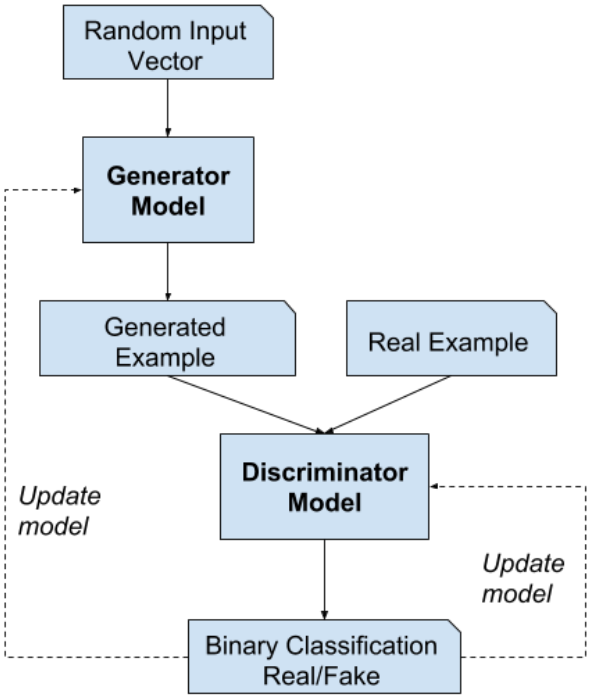
ENGR-UH 3332
Applied Machine Learning

Generative Adversarial Networks

Introduction

Deep neural networks are used mainly for supervised learning: classification or regression. Generative Adversarial Networks or GANs, however, use neural networks for a very different purpose: *Generative modeling is an unsupervised learning task in machine learning that involves automatically discovering and learning the regularities or patterns in input data in such a way that the model can be used to generate or output new examples that plausibly could have been drawn from the original dataset.
*https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/

While there are many approaches used for generative modeling, a Generative Adversarial Network takes the following approach:



There are two neural networks: a *Generator* and a *Discriminator*. The generator generates a "fake" sample given a random vector/matrix, and the discriminator attempts to detect whether a given sample is "real" (picked from the training data) or "fake" (generated by the generator). Training happens in tandem: Train the discriminator for a few epochs, then train the generator for a few epochs, and repeat. This way both the generator and the discriminator get better at doing their jobs.

Dataset

Anime Faces Dataset:
Use the Anime Face Dataset, which consists of over 63,000 cropped anime faces. Note that generative modeling is an unsupervised learning task, so the images do not have any labels.

ENGINEERING DIVISION | NYU ABU DHABI

Requirement

1. Load the anime face data set to your notebook/colab notebook
2. Build a Generator and a Discriminator neural network.
3. Train the Discriminator network using a binary cross entropy loss function
4. Train the Generator network using the Discriminator network as part of loss function (more details in attached notebook).
5. Train the Discriminator and Generator network for multiple epochs to generate 'fake' anime faces (save the output image)

Deliverables

A zip file containing the following:

1. a working project (source code, make files if needed, etc)
2. a report for the detailed description of the project
    a. Instructions on how to run your project
    b. Explain each component of your GAN (add explanation on each module like generator, discriminator, latent tensor, etc.)
    c. Show generated fake output images in the report

Before submitting your project, please make sure to test your program on the given dataset.

Notes

You may discuss the general concepts in this project with other students, but you must implement the program on your own. ***No sharing of code or report is allowed.*** *Violation of this policy can result in a grade penalty.*

Late submission is acceptable with the following penalty policy:
10 points deduction for every day after the deadline