# Implementation

Assignment 2 consisted of implementing 4 types of GANs in PyTorch: Simple GAN, DCGAN, WGAN with Gradient Clipping (WGAN-GC), and WGAN with Gradient Penalty (WGAN-GP). The code from <https://github.com/Zeleni9/pytorch-wgan> was used as a foundation.

The Simple GAN was separated into 3 classes: Model.GAN.Generator, Model.GAN.Discriminator, and Model.GAN.Network. The Model.GAN.Network is used as a base class for the remaining GANs and provides all the support methods, such as determining Cuda support, creating torch variables, and evaluating the network.

The DCGAN was also separated into 3 classes: Model.GAN.DC.Generator, Model.GAN.DC.Discriminator, and Model.GAN.DC.Network. Model.GAN.DC.Network extends Model.GAN.Network and overrides the Train method.

WGAN with Gradient Clipping was separated into 2 classes: Model.GAN.W.GC and Model.GAN.W.DiscriminatorGC. Model.GAN.W.GC uses the Model.GAN.DC.Generator as the generator, extends Model.GAN.Network, and overrides the Train method.

WGAN with Gradient Penalty was separated into 2 classes: Model.GAN.W.GP and Model.GAN.W.DiscriminatorGP. Model.GAN.W.GP uses the Model.GAN.DC.Generator as the generator, extends Model.GAN.Network, and overrides the Train method.

# Comparison

All 4 GANs were trained on the Fashion-MNIST, MNIST, and CIFAR-10 datasets. The results of the training are shown in the following section.

The Simple GAN performs the worst. It was able to train moderately well on the Fashion-MNIST and MNIST datasets but produces images with artifacts. The generator’s training became stuck on the CIFAR-10 data and was unable to make sufficient progress, resulting in the generation of “blurry” images.

The DCGAN trained well on all 3 datasets. The generator was able to produce quality images for the Fashion-MNIST and MNIST datasets. It was able to produce clearer images than the Simple GAN for the CIFAR-10 dataset, but also became stuck during training.

The WGAN-GC and WGN-GP require more time to train but generate the best quality images. Between the two, WGAN-GP appears to require less iterations to generate quality images.

# Results

|  | Fashion-MNIST | MNIST | CIFAR-10 |
| --- | --- | --- | --- |
| GAN | 28000 Training Iterations | 28000 Training Iterations | Training Iterations |
| DCGAN | 17000 Training Iterations | 28000 Training Iterations | 23000 Training Iterations |
| WGAN-GC | 26000 Training Iterations | 28000 Training Iterations | 39000 Training Iterations |
| WGAN-GP | 9000 Training Iterations | 28000 Training Iterations | 39000 Training Iterations |