**Name: Vaibhav Raheja**

**NetID(s): vraheja3**

## **GAN and LSGAN MNIST**

Show final results from training both your GAN and LSGAN (4x4 grid of images for both):

|  |  |
| --- | --- |
| **GAN ITER 4500** | **LSGAN ITER 4500** |
|  |  |

## **GAN and LSGAN Cats**

Show final results from training both your GAN and LSGAN (4x4 grid of images for both):

|  |  |
| --- | --- |
| **GAN ITER 11750** | **LSGAN ITER 11750** |
|  |  |

Discuss any differences you observed in quality of output or behavior during training of the two GAN models.

Loss Functions and Stability:

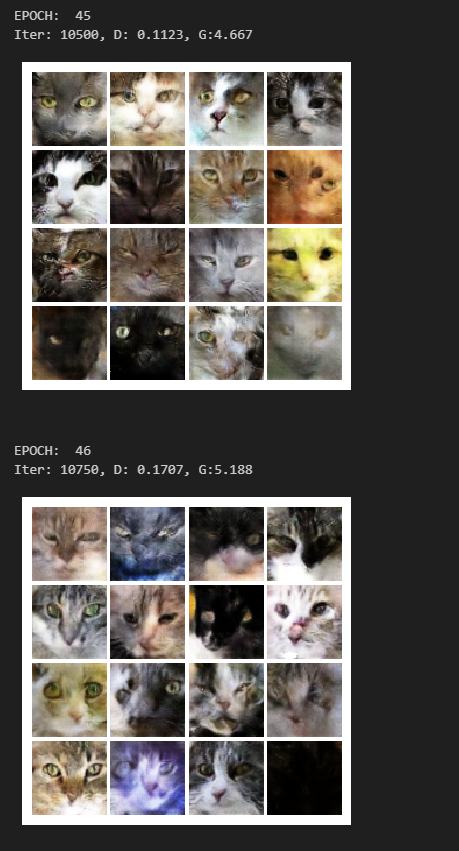
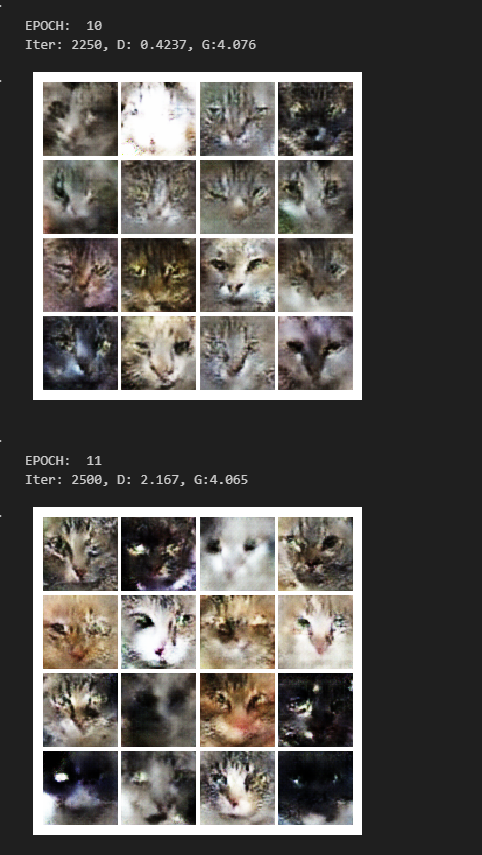
* Original GAN: Uses a binary cross-entropy loss function. This leads to instability in training due to the saturating gradients problem when the discriminator gets too confident outside regions with high data density​​.
* LS-GAN: Employs a least squares loss function, which tends to provide more stable gradients and can be less sensitive to outliers. This leads to a more stable training process as it avoids problems with the discriminator's confidence​​.

Convergence Behavior:

* The LS-GAN tends to show lower variance in loss values across training epochs compared to the original GAN. This leads to a more consistent quality of generated images, as indicated by relatively smaller fluctuations in the discriminator and generator losses

Do you notice any instances of mode collapse in your GAN training? Show some instances of mode collapse from your training output.

Yes

## **Extra Credit - Alternative Dataset**

Using a Pokémon dataset to generate “new” Pokémon.



|  |  |
| --- | --- |
| **GAN** | **LSGAN** |
|  |  |

I did not get an expected output. In this implementation we can see that GAN is giving a somewhat output which look like Pokémon’s.