**Part 1**

1. **Modify the latent space size: Change the noise vector size from 100 to 50 or 200. Observe how this change affects the quality and variety of generated images.**

* Smaller latent space (50 used for testing): Lower image diversity and quality.
* Larger latent space (used 200): Better diversity and quality in generated images.

1. **Observation after training the GAN for 10,000 epochs.**

* Early stages: Noisy, low-quality images.
* Mid-training: Gradual improvement in clarity and definition.
* Late stages: Well-defined digits, possibly with some imperfections in certain cases.

1. **Change the optimizer from Adam to RMSprop or SGD for both the generator and discriminator. Observe the impact on training performance and image quality.**

* Adam: Typically results in faster convergence and better image quality.
* RMSprop: Might slow down convergence but can still be effective with proper tuning.
* SGD: Often leads to unstable training, especially for GANs.

1. **Experiment with different batch sizes. What impact different batch sizes have on training?**

* Smaller batch sizes lead to more frequent updates of the model weights. These often lead to faster convergence because the model parameters are updated more frequently. However, the updates can be noisy, which might cause fluctuations in the training process.
* Larger batch sizes lead to more stable and accurate gradient estimates. However, they might converge more slowly because the model parameters are updated less frequently and requires more memory and computational resources for training.

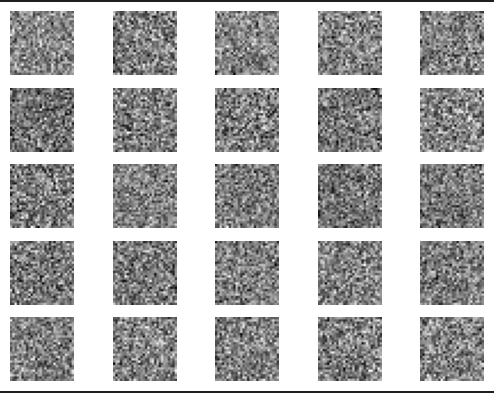


Figure 1: GAN Training

**Part 2**

1. **Modify the CGAN to generate images for specific digits (e.g., generate only '7' or '9').**

****

Figure 2: CGAN Generating Digit 9

1. **Implement label smoothing by replacing real labels of 1 with random values between 0.9 and 1 during training.**

**A computer screen shot of a program code

Description automatically generated**

Figure 3: Label Smoothing

1. **Create noise vectors corresponding to two different digits (e.g., '3' and '8'). Perform interpolation between the two noise vectors and visualize how the generated images morph from one digit to the other.**

**A screenshot of a computer program

Description automatically generated**

Figure 4: Noise Vectors