1 Assignment Specifications

1.1 Objective

The objective of this project is to:

- Understand the core ideas behind generative models
- Implement and build the Generative models
- Generate new images from the trained model

For this assignment, we will develop **FOUR** different models and these are:

1) Unconditional Generative Adversarial Networks (GANs)
2) Conditional Generative Adversarial Networks (GANs)
3) Unconditional Diffusion Model
4) Conditional Diffusion Model

For each model, we will analyze the model performance and tune the model hyperparameters during training phase. Next, we evaluate the model & use the best model to create new images.

1.1.1 Data Preprocessing and Data Loading

We will be using the fashion MNIST contains a collection of images on fashion items from 10 different categories:

1) T-shirt/Top
2) Trouser
3) Pullover
4) Dress
5) Coat
6) Sandal
7) Shirt
8) Sneaker
9) Bag
10) Ankle Boot

A sample from the dataset is shown in Figure 1.

Figure 1 – Sample of images from the fashion MNIST dataset

The dataset consists of **60,000** training images across 10 categories. Each image is grayscale with a resolution of 28x28 pixels, where the pixel intensity values range from 0 to 255.

2 Development of Generator Models

Four generator models were built for generating new images from the 10 fashion categories.

- Model 1: Unconditional GAN
- Model 2: Unconditional Diffusion Model
- Model 3: Conditional GAN
- Model 4: Conditional Diffusion Model

2.1 Model 1: Unconditional Vanilla GAN

Four experiments were carried out:

The baseline model started with 50 epochs, we then moved to the second experiment increasing the number of epochs to 100. The third experiment reduced the display step to 200 and the last increased latent dimension to 200.

2.1.1 Experiment 1: Number of Epochs = 50

Hyperparameter	Value
criterion	BCEWithLogitsLoss()
n_epochs	50
z_dim	100
display_step	1000
Learning rate	0.0002

Table 1 – Highlighted Cell showing changed parameter

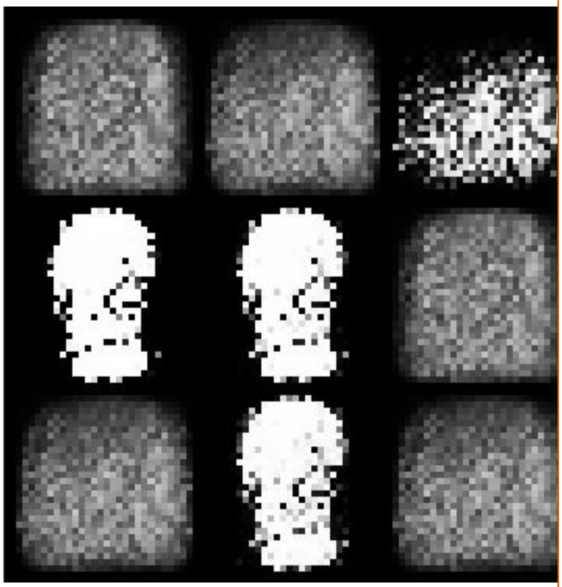
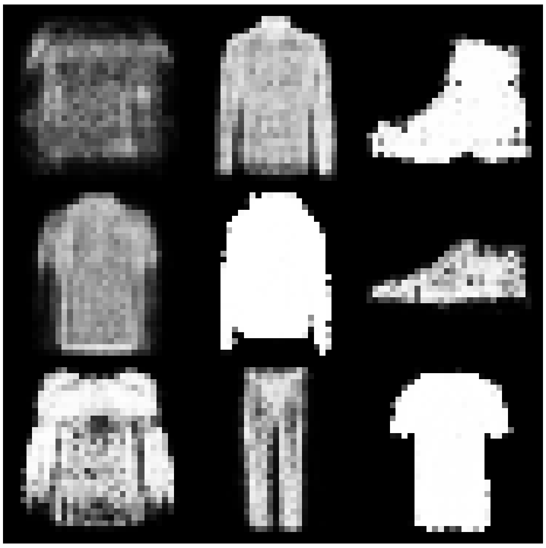
Epoch	Fake	Real
0		
49		

Table 2 – Fake/Real Images from first epoch (0) and last epoch (49)

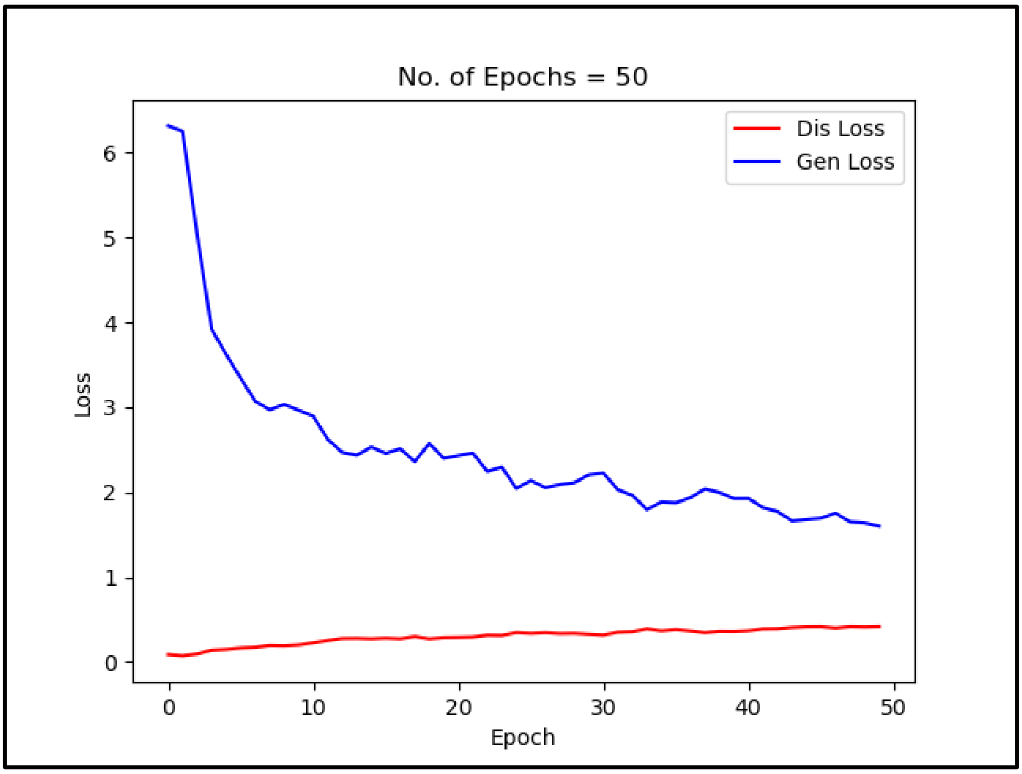


Figure 2 – Discriminator/Generator Loss vs # Epochs

Analysis of the Experiment

1. Initial Observations:

Epoch 0:

- Generator Loss (6.32): A very high generator loss indicates that the generator is initially struggling to produce realistic samples that can fool the discriminator. This is expected at the start of training, as the generator weights are randomly initialized.
- Discriminator Loss (0.09): The low discriminator loss suggests that the discriminator is very effective at distinguishing between real and generated samples. The discriminator is likely overpowering the generator at this stage, making it difficult for the generator to improve.

Epoch 49:

- Generator Loss (1.60): The significant decrease in generator loss indicates that the generator has learned to produce more realistic samples over the course of training. While it's not perfect, the lower loss suggests improved performance in generating data that can partially fool the discriminator.
- Discriminator Loss (0.42): The increase in discriminator loss shows that it has become more challenging for the discriminator to differentiate between real and generated samples. This indicates a better balance between the generator and discriminator, as the generator is now producing more realistic samples.

Summary

Improvement in Generator Performance: The decrease in generator loss from epoch 0 to epoch 49 shows that the generator has significantly improved its ability to produce realistic samples.

Balancing the GAN: The increase in discriminator loss suggests that the training process has led to a more balanced GAN, where the discriminator is no longer overpowering the generator. This balance is crucial for stable GAN training.

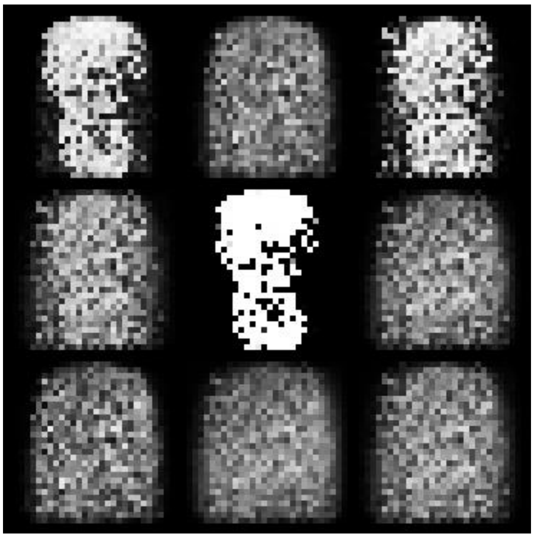
Training Dynamics: Initially, the discriminator was highly effective, leading to a high generator loss. Over time, the generator adapted and improved, resulting in lower generator loss and higher discriminator loss. This is a typical dynamic in GAN training.

Overall, the GAN has shown substantial progress from the start to the end of the training period. While there is still room for improvement, the trends in the loss values indicate that the model is on the right path. Further training and fine-tuning could yield even better results.

2.1.2 Experiment 2: Number of Epochs = 100

Hyperparameter	Value
criterion	BCEWithLogitsLoss()
n_epochs	100
z_dim	100
display_step	1000
Learning rate	0.0002

Table 3 – Highlighted Cell showing changed parameter

Epoc h	Fake	Real
0		

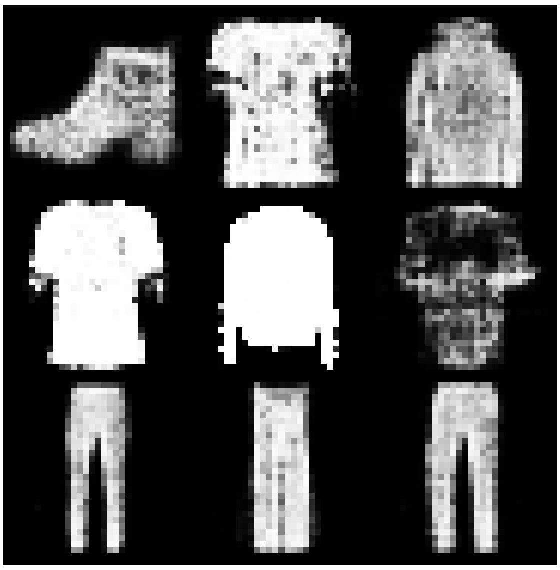
Epoch	Fake	Real
99		

Table 4 – Fake/Real Images from first epoch (0) and last epoch (99)

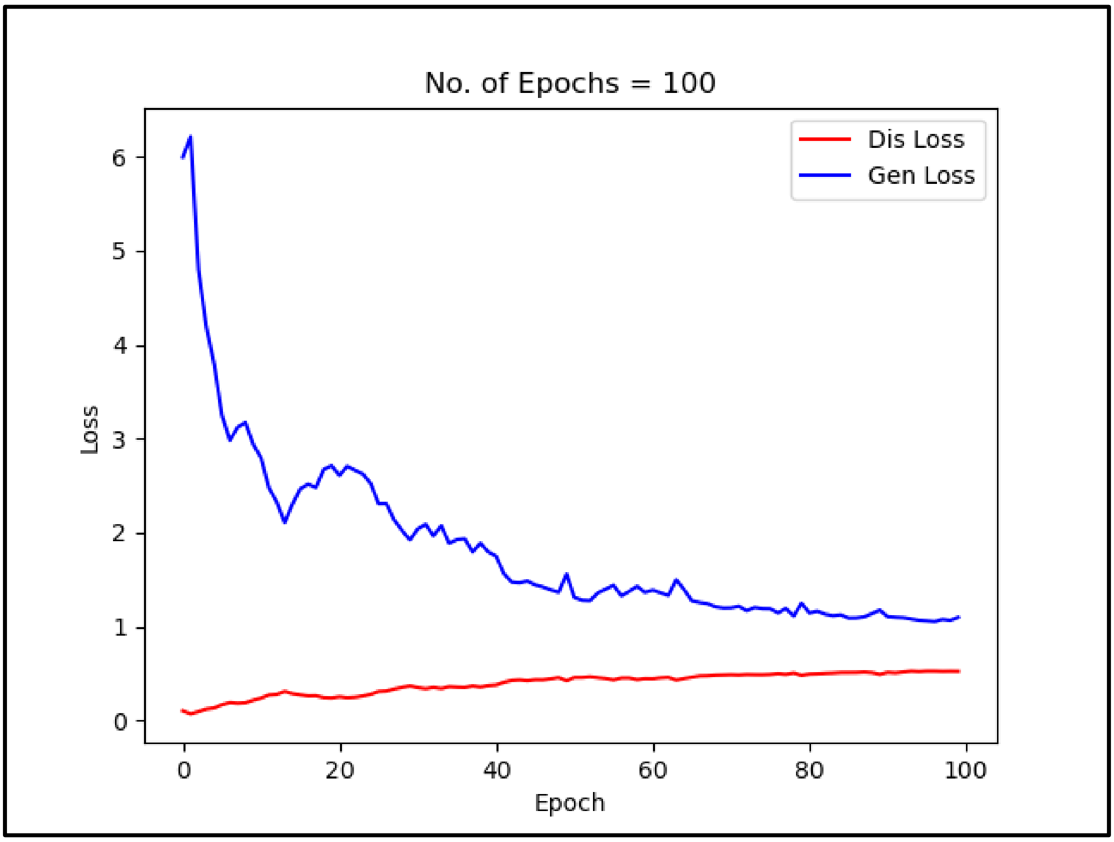


Figure 3 – Discriminator/Generator Loss vs # Epochs

Epoch 0:

- **Generator Loss (6.00)**: At the beginning of training, the generator loss is quite high, indicating that the generator is struggling to create realistic samples. This is expected as the model starts with randomly initialized weights.
- **Discriminator Loss (0.10)**: The discriminator loss is very low, suggesting that the discriminator can easily distinguish between real and generated samples. It indicates that the discriminator is highly effective at the start of training.

Epoch 99:

- **Generator Loss (1.10)**: After 100 epochs, the generator loss has significantly decreased. This suggests that the generator has become much better at producing realistic samples that can partly fool the discriminator. The reduction in generator loss is a positive sign of improvement.
- **Discriminator Loss (0.52)**: The discriminator loss has increased, which means it is finding it more challenging to distinguish between real and generated samples. This indicates a better balance between the generator and discriminator, as the generator is now producing higher-quality samples.

Summary

- **Improvement in Generator Performance**: The drop in generator loss from around 6.00 to 1.10 shows significant progress in the generator's ability to produce realistic samples. This improvement is likely due to the increased number of training epochs, allowing the generator more time to learn.
- **Balancing the GAN**: The increase in discriminator loss from 0.10 to 0.52 suggests that the training process has balanced the competition between the generator and discriminator. The discriminator's increased difficulty in distinguishing fake from real data signifies better quality generated samples.
- **Training Dynamics**: Initially, the discriminator easily distinguished real from fake data, but as training progressed, the generator improved, making the task more challenging for the discriminator. This evolving dynamic is typical in GAN training.

Overall, increasing the number of epochs has led to a more balanced and effective GAN, with notable improvements in the generator's performance and an adequately challenged discriminator. Continued training and potential fine-tuning might further enhance the model's capabilities.

2.1.3 Experiment 3: Reduce the display step to 200

Hyperparameter	Value
criterion	BCEWithLogitsLoss()
n_epochs	100
z_dim	100
display_step	200
Learning rate	0.0002

Table 5 – Highlighted Cell showing changed parameter

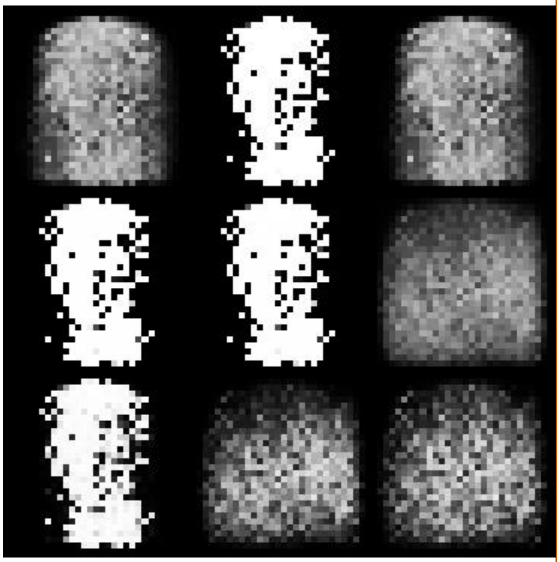
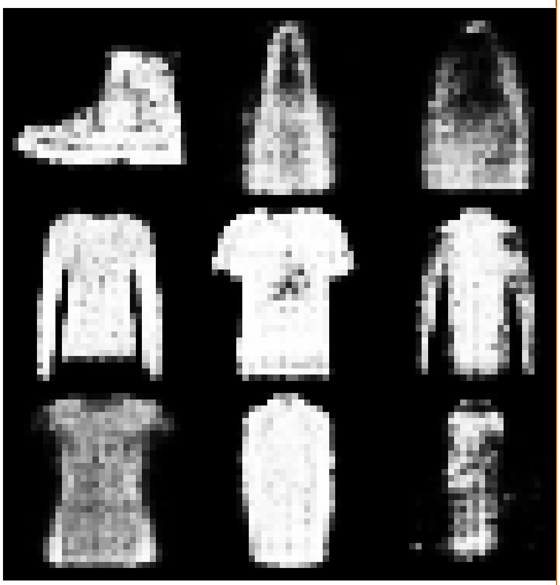
Epoch	Fake	Real
0		
99		

Table 6 – Fake/Real Images from first epoch (0) and last epoch (99), with display step set to 200

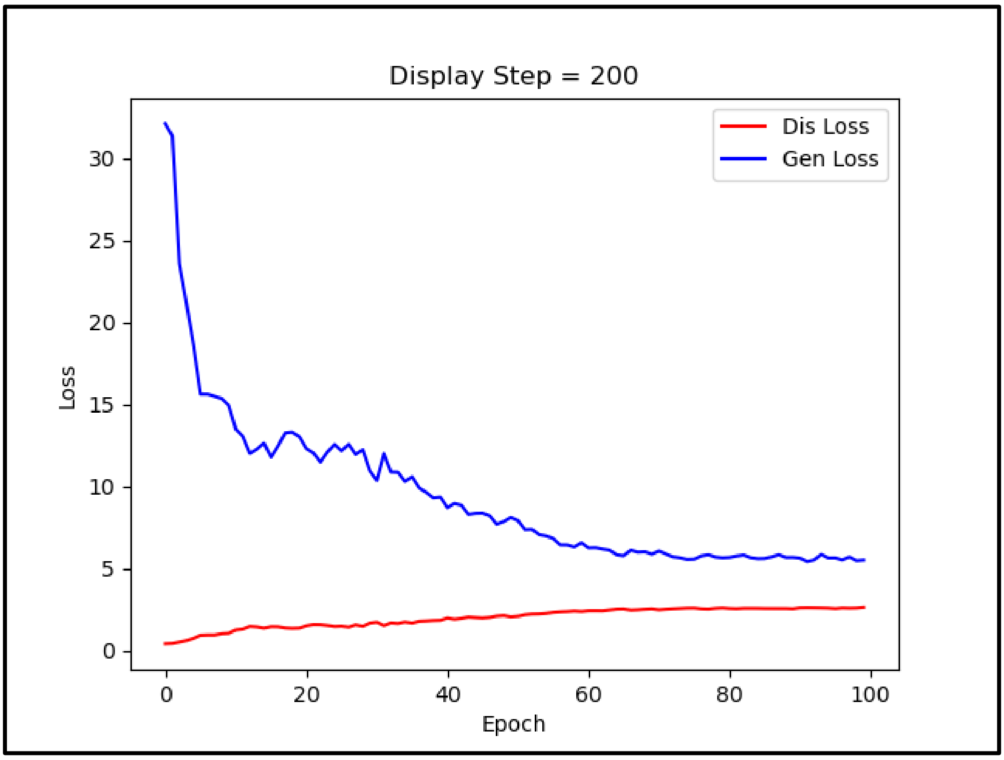


Figure 4 – Discriminator/Generator Loss with display step set to 200

Epoch 0:

- **Generator Loss (32.09)**: This exceptionally high generator loss indicates that the generator is initially performing very poorly, creating samples that the discriminator can easily identify as fake. Such a high loss suggests significant room for improvement and might point to issues with model initialization or hyperparameters.
- **Discriminator Loss (0.40)**: The discriminator loss is relatively low, implying that the discriminator is effective at distinguishing between real and fake samples at the start. However, it's slightly higher than in previous experiments, suggesting it is encountering a bit more difficulty, possibly due to the random nature of initialization.

Epoch 99:

- **Generator Loss (5.50)**: The generator loss has decreased substantially from epoch 0, which indicates that the generator has learned to produce more realistic samples over the course of training. Nevertheless, the loss is still relatively high, suggesting that the generator is still struggling and might not have fully converged.
- **Discriminator Loss (2.62)**: The discriminator loss has increased significantly, indicating that the discriminator is finding it more challenging to distinguish between real and generated samples. This suggests the generator is producing better quality samples, making the discriminator's task more difficult.

Summary

- **Improvement in Generator Performance**: The reduction in generator loss from around 32.09 to 5.50 shows that the generator has improved, but the relatively high final loss suggests that there is still considerable room for further improvement.

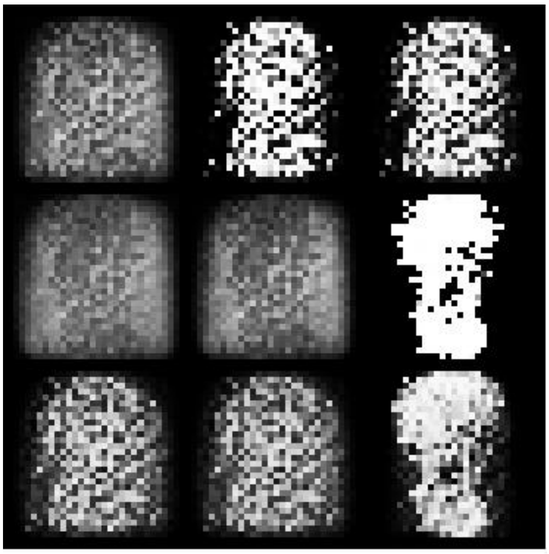
- **Balancing the GAN**: The increased discriminator loss from 0.40 to 2.62 indicates that the training process has introduced a better balance between the generator and the discriminator, as the generator is now producing more realistic samples that challenge the discriminator.
- **Training Dynamics with More Frequent Display Steps**: Changing the display step from 1000 to 200 allows more frequent monitoring of the training progress. This can help identify issues earlier and adjust training parameters if necessary. The high initial losses and subsequent improvement highlight the typical learning curve of GANs but also suggest that further tuning may be needed.

Overall, the GAN shows significant progress from the start to the end of the training period. However, the relatively high generator loss at epoch 99 implies that further training or adjustment of hyperparameters may be required to achieve optimal results. The more frequent display steps help in closely monitoring the training process, providing better insights into the model's performance and stability over time.

2.1.4 Experiment 4: Increase latent dimension to 200

Hyperparameter	Value
criterion	BCEWithLogitsLoss()
n_epochs	100
z_dim	200
display_step	1000
Learning rate	0.0002

Table 7 – Highlighted Cell showing changed parameter

Epoch	Fake	Real
0		

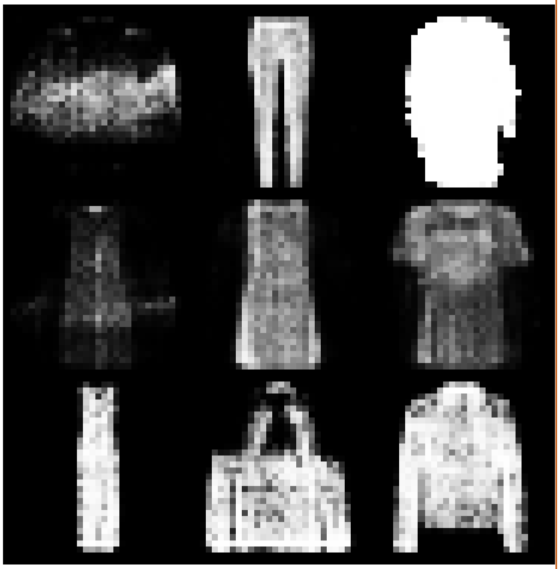
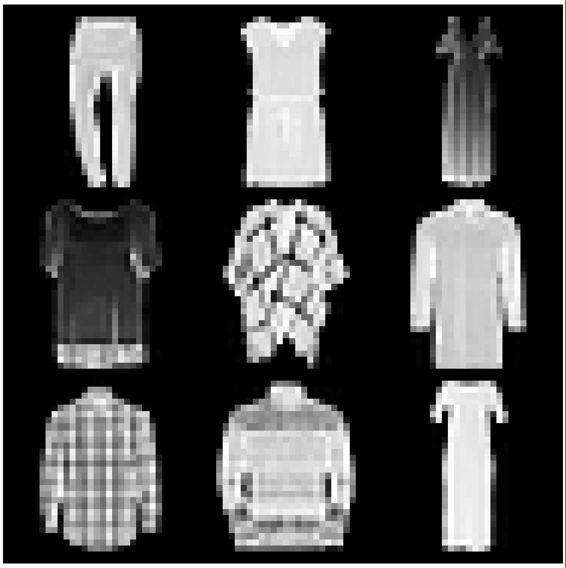
Epoch	Fake	Real
99		

Table 8 – Fake/Real Images from first epoch (0) and last epoch (99), with latent dimension set to 200

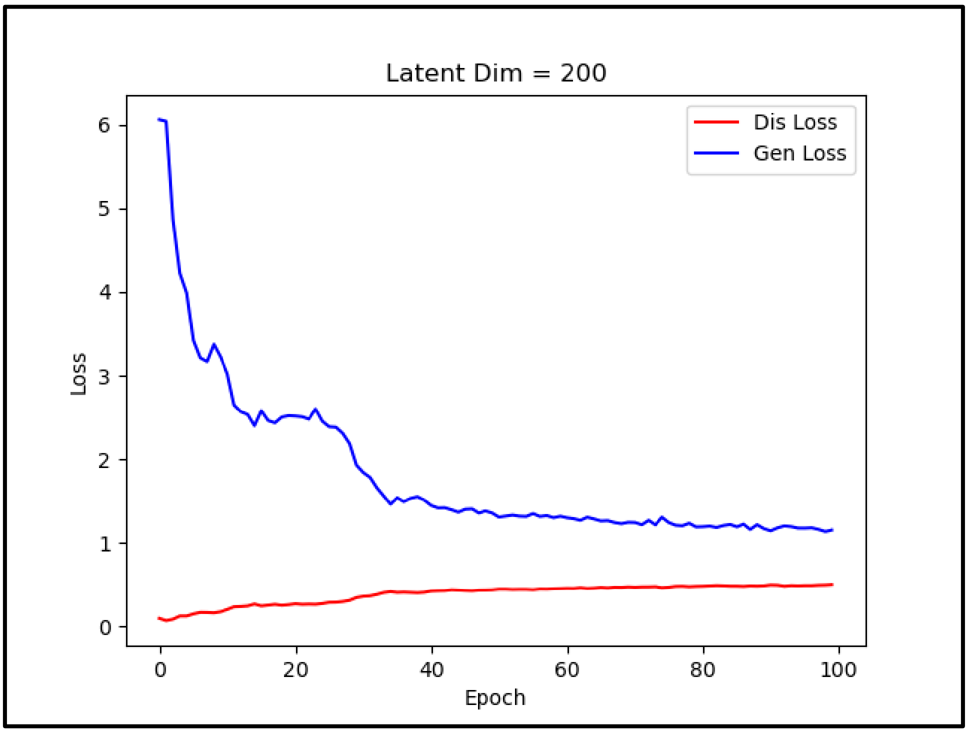


Figure 5 – Discriminator/Generator Loss with latent dimension set to 200

Analysis of GAN Training with Increased Latent Dimension

Epoch 0:

- **Generator Loss (6.06)**: The initial generator loss is fairly high, indicating that the generator is still learning to produce realistic samples that can fool the discriminator.

However, it's noticeably lower than the extremely high loss observed in a previous experiment, suggesting a slightly better starting point.

- **Discriminator Loss (0.096)**: The discriminator's loss is very low, suggesting it can easily distinguish between real and generated samples. This is expected early in training when the generator's outputs are not yet convincing.

Epoch 99:

- **Generator Loss (1.15)**: The generator loss has decreased significantly, which indicates that the generator has improved considerably in producing realistic samples. This loss is relatively low and suggests that the generator is quite effective, although there's still room for improvement.
- **Discriminator Loss (0.50)**: The discriminator loss has increased, indicating that it is finding it more challenging to distinguish between real and generated samples. This increase is a positive sign, as it shows the generator's outputs have improved in quality.

Summary

- **Improvement in Generator Performance**: The reduction in generator loss from approximately 6.06 to 1.15 signifies a substantial improvement in the generator's ability to produce realistic samples. This suggests that the increased latent dimension might have provided the generator with more capacity to learn the underlying data distribution.
- **Balancing the GAN**: The increase in discriminator loss from 0.10 to 0.50 indicates a better balance between the generator and discriminator. The generator's improved performance is making it harder for the discriminator to identify fake samples, which is a sign of a well-trained GAN.
- **Impact of Latent Dimension**: Changing the latent dimension from 100 to 200 seems to have positively impacted the generator's learning capacity, allowing it to produce more realistic samples. The larger latent space provides more variability and complexity in the generated samples, which can improve the generator's performance.

Overall, increasing the latent dimension has led to a more effective generator that produces higher-quality samples. The balance between the generator and discriminator suggests that both networks have trained well. Further training and fine-tuning might yield even better results, but the current setup already shows promising improvements

2.1.5 Summary Evaluation

	Experiment	Average Generator Loss	Average Discriminator Loss
0	Experiment 1	2.50	0.30
1	Experiment 2	1.84	0.39
2	Experiment 3	9.37	1.97
3	Experiment 4	1.83	0.38

Table 9 – Average Generator/Discriminator Losses

Analysis

1. Experiment 1:
   - Generator Loss (2.50): This indicates that the generator is moderately effective at producing realistic data, but there is room for improvement.
   - Discriminator Loss (0.30): A low discriminator loss suggests that the discriminator is quite effective at distinguishing between real and fake data. However, it might be overpowering the generator, leading to a higher generator loss.

2. Experiment 2:
    - Generator Loss (1.84): A lower generator loss compared to Experiment 1 indicates that the generator is performing better and producing more realistic data.
    - Discriminator Loss (0.39): The discriminator loss is slightly higher than in Experiment 1, suggesting a better balance between the generator and discriminator. This balance is crucial for stable GAN training.
3. Experiment 3:
    - Generator Loss (9.37): A very high generator loss indicates that the generator is struggling significantly to produce realistic data. This could be due to various factors such as poor hyperparameter settings or instability in training.
    - Discriminator Loss (1.97): A high discriminator loss suggests that the discriminator is also struggling to distinguish between real and fake data. This indicates overall instability in the GAN training process.
4. Experiment 4:
    - Generator Loss (1.83): Similar to Experiment 2, this low generator loss indicates good performance in generating realistic data.
    - Discriminator Loss (0.38): The discriminator loss is low but balanced with the generator loss, suggesting effective training and a good balance between the generator and discriminator.

Summary

- Best Performance: Experiments 2 and 4 show the best performance with low generator losses and balanced discriminator losses. These experiments indicate effective training and a good balance between the generator and discriminator.
- Poor Performance: Experiment 3 shows poor performance with very high losses for both the generator and discriminator, indicating instability and ineffective training.
- Moderate Performance: Experiment 1 shows moderate performance with a higher generator loss and a very low discriminator loss, suggesting that the discriminator might be overpowering the generator.

Overall, experiments 2 and 4 are the most promising, showing effective training and a good balance between the generator and discriminator. Further fine-tuning of hyperparameters and additional experiments could help improve the performance even more.

2.1.6 New Sample Generation from Experiment 4

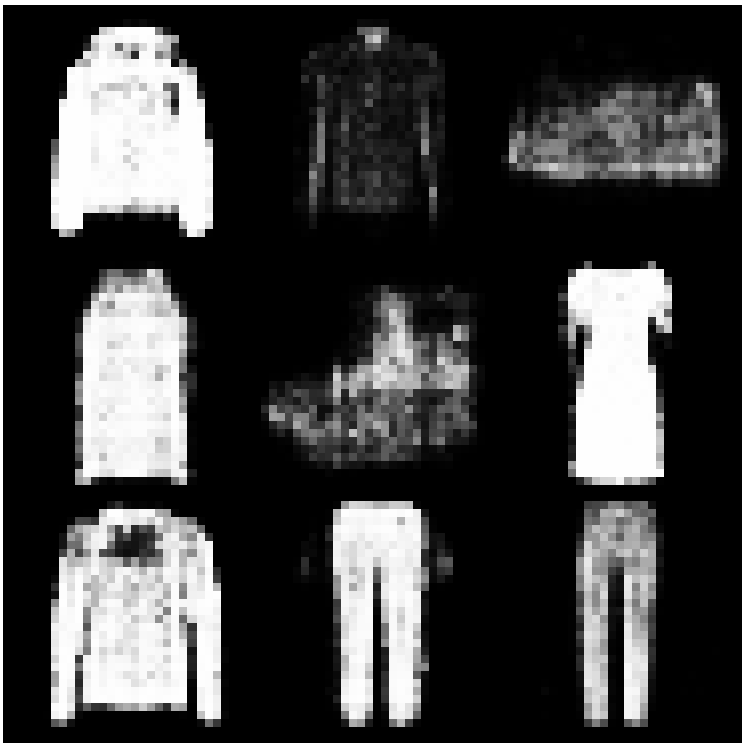


Figure 6 – New samples generated using the model from Experiment 4

2.2 Model 2: Unconditional Diffusion Models

Four experiments were carried out:

The baseline model started with 50 epochs, we then moved to the second experiment increasing the number of epochs to 100. The third experiment lowered the learning rate to 0.0001 and the last changed the batch size from 128 to 64.

2.2.1 Experiment 1: Number of Epochs = 50

Hyperparameter	Value
Criterion	mse_loss
Batch Size	128
n_epochs	50
Timestep	150
Optimizer	Adam
Learning rate	0.001

Table 10 – Highlighted Cell showing changed parameter

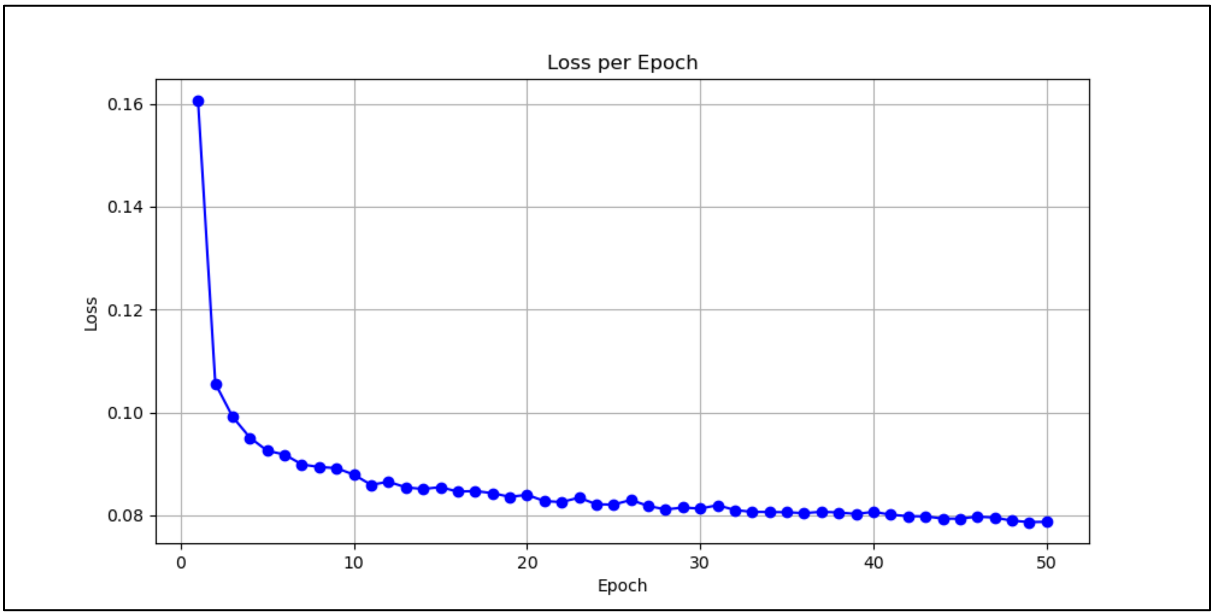


Figure 7 – Experiment 1: Loss vs Epochs (50)

Comments:

- **Initial Loss Values**: At the beginning of training, the loss values are usually higher because the model starts with random weights and has not yet learned any meaningful patterns from the data.
- **Early Epochs (1-10)**: During the initial epochs, a sharp decline in the loss values is observed. This rapid drop indicates that the model is quickly learning the basic structure of the data and adjusting its weights accordingly.
- **Middle Epochs (11-30)**: As training progresses, the rate of decrease in loss values begins to slow down. The model starts refining its understanding and improving incrementally. The loss curve might show occasional fluctuations due to the stochastic nature of gradient descent and batch processing.
- **Later Epochs (31-50)**: In the later stages, the loss values stabilizes and reach a plateau. This indicates that the model is converging and significant improvements are becoming harder to achieve. If the loss starts increasing, it could suggest overfitting, where the model is learning noise from the training data rather than generalizable patterns.
- **Final Loss Values**: By the end of 50 epochs, the loss values should be relatively stable. A low final loss value suggests that the model has successfully learned to generate samples that are close to the real data. However, it’s crucial to compare these values with validation loss to ensure the model is not overfitting.

Analysis:

- **Steady Improvement**: A consistent decrease in loss values signifies that the model is learning effectively.
- **Plateau**: Reaching a plateau is normal and indicates that further training might need adjustments, such as learning rate tuning or introducing regularization techniques.
- **Fluctuations**: Minor fluctuations are expected but significant spikes in loss might point to issues like poor data quality or suboptimal training parameters.

2.2.2 Experiment 2: Number of Epochs = 100

Hyperparameter	Value
Criterion	mse_loss

Hyperparameter	Value
Batch Size	128
n_epochs	100
Timestep	150
Optimizer	Adam
Learning rate	0.001

Table 11 – Highlighted Cell showing changed parameter

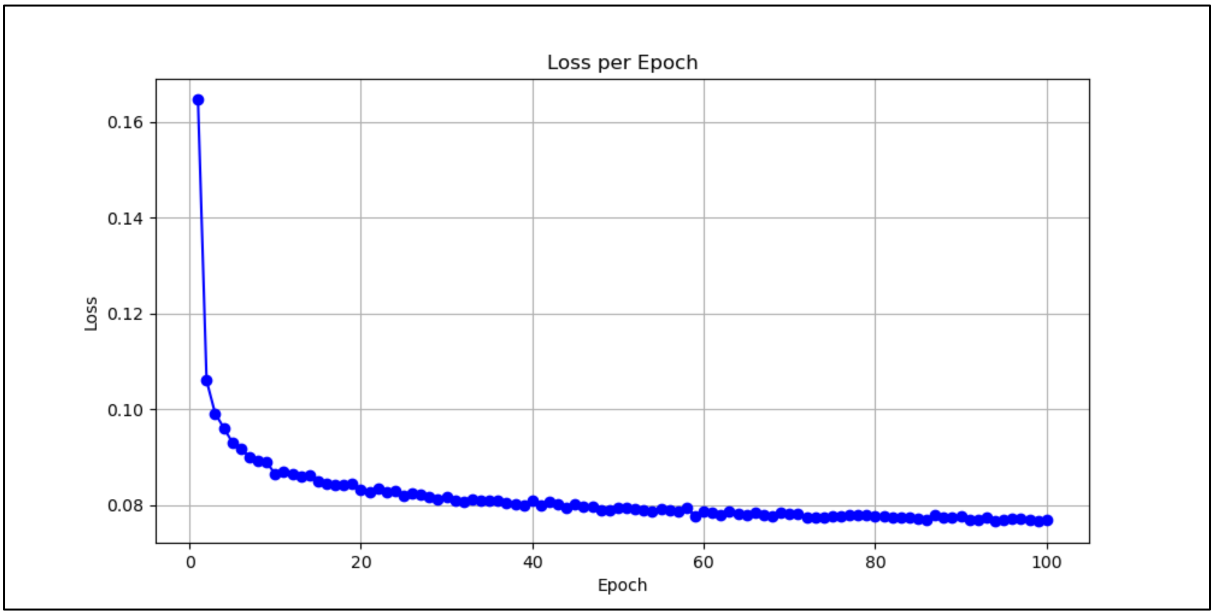


Figure 8 – Experiment 2: Loss vs Epochs (100)

Comments:

Initial Epochs (1-10)

- **High Initial Loss**: The starting loss of 0.1645 shows that the model begins with a significant error.
- **Rapid Decrease**: The sharp drop to 0.0865 by the 10th epoch suggests that the model quickly adapts to the data, learning basic patterns effectively.

Middle Epochs (11-50)

- **Continued Improvement**: The loss values continue to decline gradually, reaching around 0.080 by the 50th epoch. This indicates steady learning and fine-tuning of the model's parameters.
- **Stability and Refinement**: The gradual decrease in loss values, combined with minor fluctuations, reflects the model's ongoing refinement of learned patterns.

Later Epochs (51-100)

- **Plateauing**: From epoch 51 to 100, the loss values stabilize around 0.077 to 0.078. This plateauing effect indicates the model is reaching its optimal performance and further training yields minimal improvement.
- **Consistent Performance**: The relatively consistent loss values during these epochs suggest a well-converged model, with effective generalization on the training data.

Key Observations

- **Effective Learning Rate**: The learning rate of 0.001 appears well-suited for this model, balancing the learning speed and stability.
- **Batch Size Impact**: Using a batch size of 128 provides a good compromise between stable updates and computational efficiency, contributing to the steady decline in loss values.
- **Extended Training**: Increasing the number of epochs to 100 allows the model to further refine its learning, though the plateau indicates that most improvements occur within the first 50-60 epochs.

Analysis

- **Initial Rapid Learning**: The rapid decrease in the initial epochs is typical, reflecting the model's ability to quickly learn basic features from the dataset.
- **Gradual Refinement**: The middle epochs show consistent improvement, signifying effective learning of more complex data structures and patterns.
- **Plateau and Stability**: The plateau in the later epochs indicates the model has reached an equilibrium, where additional training provides diminishing returns.

2.2.3 Experiment 3: Set Learning Rate = 0.0001

Hyperparameter	Value
Criterion	mse_loss
Batch Size	128
n_epochs	50
Timestep	150
Optimizer	Adam
Learning rate	0.001

Table 12 – Highlighted Cell showing changed parameter

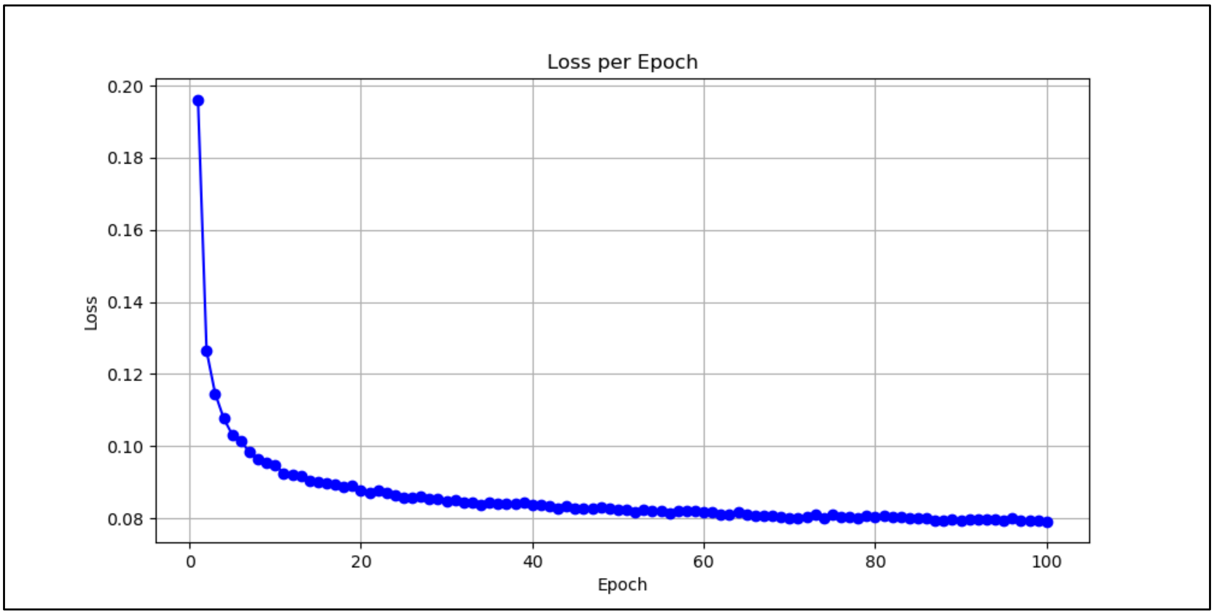


Figure 9 – Experiment 3: Loss vs Epochs

Comments:

Early Epochs (1-20)

- **Initial Loss**: As before, expect high initial loss values, rapidly decreasing as the model starts to learn.
- **Effect of Lower Learning Rate**: With a lower learning rate, the initial decrease might be less steep compared to higher learning rates, indicating slower but more stable learning.

Middle Epochs (21-60)

- **Gradual Decline**: The loss values should continue to decline, albeit more gradually. The lower learning rate helps in avoiding large updates, which reduces the risk of overshooting the optimal parameters.
- **Stability**: The training process is likely to be more stable, with fewer significant fluctuations in loss values. This stability is beneficial in fine-tuning the model and achieving better convergence.

Late Epochs (61-100)

- **Plateauing**: Similar to the earlier analysis, the loss values will eventually plateau as the model reaches a point of diminishing returns with the current setup.
- **Fine-tuning**: With 100 epochs, the model has more time to fine-tune and adjust its weights, leading to potentially better performance and generalization.

Final Loss Values and Generalization

- **Lower Final Loss**: Ideally, the final loss values should be lower compared to runs with a higher learning rate, indicating better model performance.
- **Validation Loss**: It is crucial to monitor validation loss to ensure that the model is not overfitting.

Key Observations

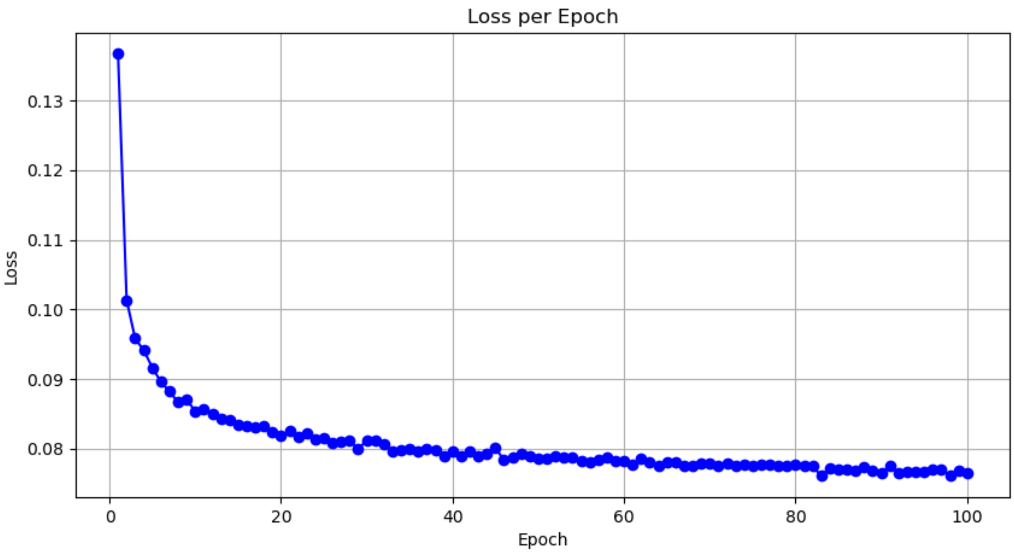
- **Learning Rate Effect**: The reduced learning rate results in more stable and incremental learning, minimizing the risk of drastic parameter updates that can lead to instability.
- **Extended Training**: The extended number of epochs allows for thorough training, with the model having more opportunities to adjust and fine-tune its weights.
- **Overfitting**: With prolonged training, there's always a risk of overfitting. Regularly monitoring validation performance can help mitigate this risk.

Changing the learning rate to 0.0001 and training for 100 epochs typically yields a more stable training process and better convergence. The model benefits from gradual and consistent updates, leading to potentially lower final loss values and improved generalization.

2.2.4 Experiment 4: Change Batch Size to 64

Hyperparameter	Value
Criterion	mse_loss
Batch Size	64
n_epochs	100
Timestep	150
Optimizer	Adam
Learning rate	0.001

Table 13 – Highlighted Cell showing changed parameter



Comments:

Loss Value Trends

- **Initial Epochs (1-10)**: The loss values start relatively high and decrease rapidly, indicating that the model quickly learns the basic patterns in the dataset. For example, from 0.1367 in the first epoch to 0.0941 by the 10th epoch.
- **Middle Epochs (11-50)**: The loss continues to decline but at a slower rate, showing that the model is gradually refining its understanding. The values stabilize around 0.084, indicating consistent improvement but with diminishing returns.
- **Later Epochs (51-100)**: The decline in loss values becomes even more gradual, eventually plateauing. The values hover around 0.078, demonstrating that the model is reaching its optimal performance level.

Key Observations

- **Stable Learning**: The smooth, consistent decline in loss values suggests a stable learning process without significant fluctuations, indicating that the chosen learning rate and batch size are appropriate.
- **Effective Convergence**: The loss values' plateau indicates that the model is converging effectively, as seen in the steady values from around epoch 50 onward. This is a good sign that the model has learned the underlying data patterns well.
- **Batch Size Impact**: Reducing the batch size to 64 provides more frequent updates to the model weights, allowing for finer adjustments and contributing to the smooth decline in loss values.

Analysis

- **Early Rapid Learning**: The initial steep decline reflects the model's ability to quickly grasp the fundamental features of the Fashion MNIST dataset.
- **Gradual Refinement**: The middle epochs show slower improvement, typical as the model transitions from learning basic patterns to refining its performance on more complex details.

- **Plateau and Convergence**: The later epochs' plateau signifies the model's convergence, where additional epochs contribute minimally to performance improvement, indicating the model has effectively learned the dataset.

2.2.5 New Samples generated from Experiments

2.2.5.1 Experiment 1: Number of Epochs - 50

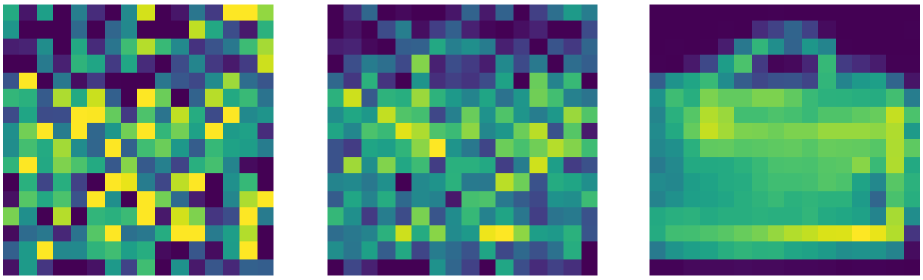


Figure 10 – New samples generated from Experiment 1

2.2.5.2 Experiment 2: Number of Epochs - 100

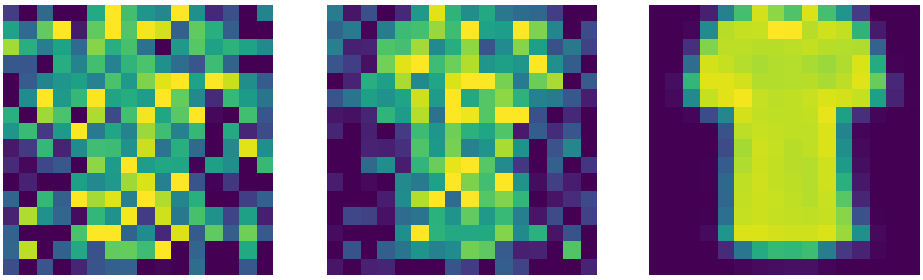


Figure 11 – New samples generated from Experiment 2

2.2.5.3 Experiment 3: Set Learning Rate = 0.0001 (from 0.001)

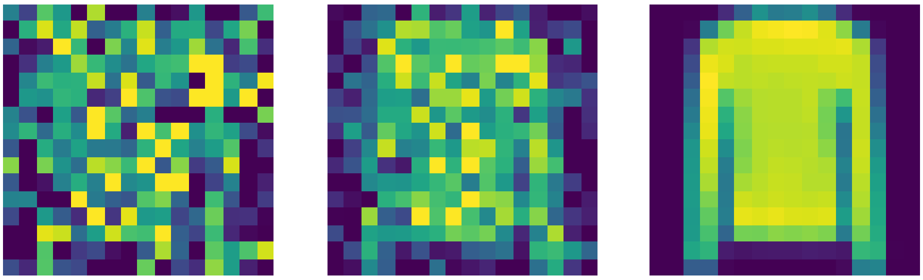


Figure 12 – New samples generated from Experiment 3

2.2.5.4 Experiment 4: Change Batch Size to 64 (from 128)

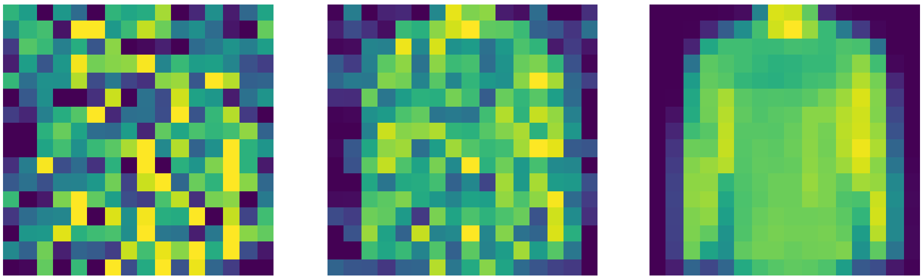


Figure 13 – New samples generated from Experiment 4

2.3 Model 3: Conditional DC GAN

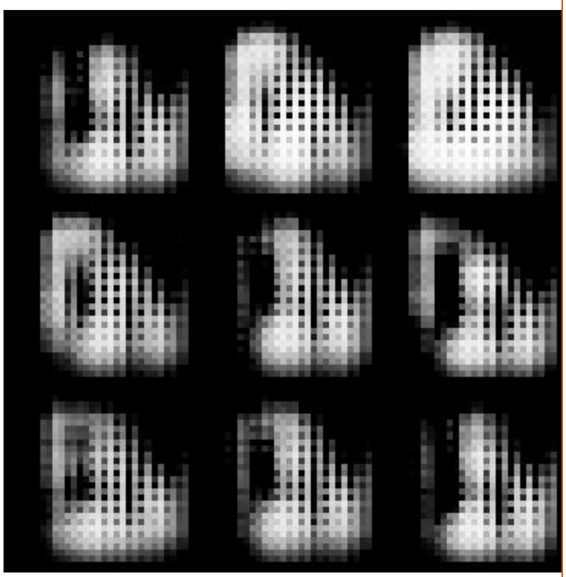
Four experiments were carried out:

The baseline model started with 50 epochs, we then moved to the second experiment increasing the number of epochs to 100. The third experiment reduced the display step to 200 and the last increased latent dimension to 200.

2.3.1 Experiment 1: Number of Epochs = 50

Hyperparameter	Value
criterion	BCEWithLogitsLoss()
n_epochs	50
z_dim	64
display_step	500
Batch size	64
Learning rate	0.0002

Table 14 – Highlighted Cell showing changed parameter

Epoch	Fake	Real
0		

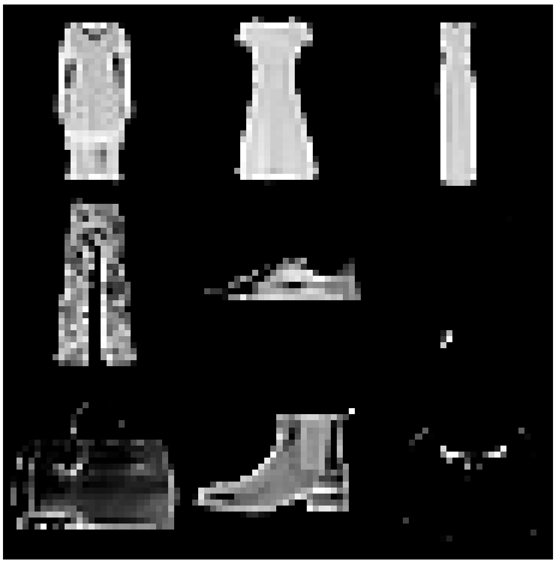
Epoch	Fake	Real
49		

Table 15 – Fake/Real Images from first epoch (0) and last epoch (49)

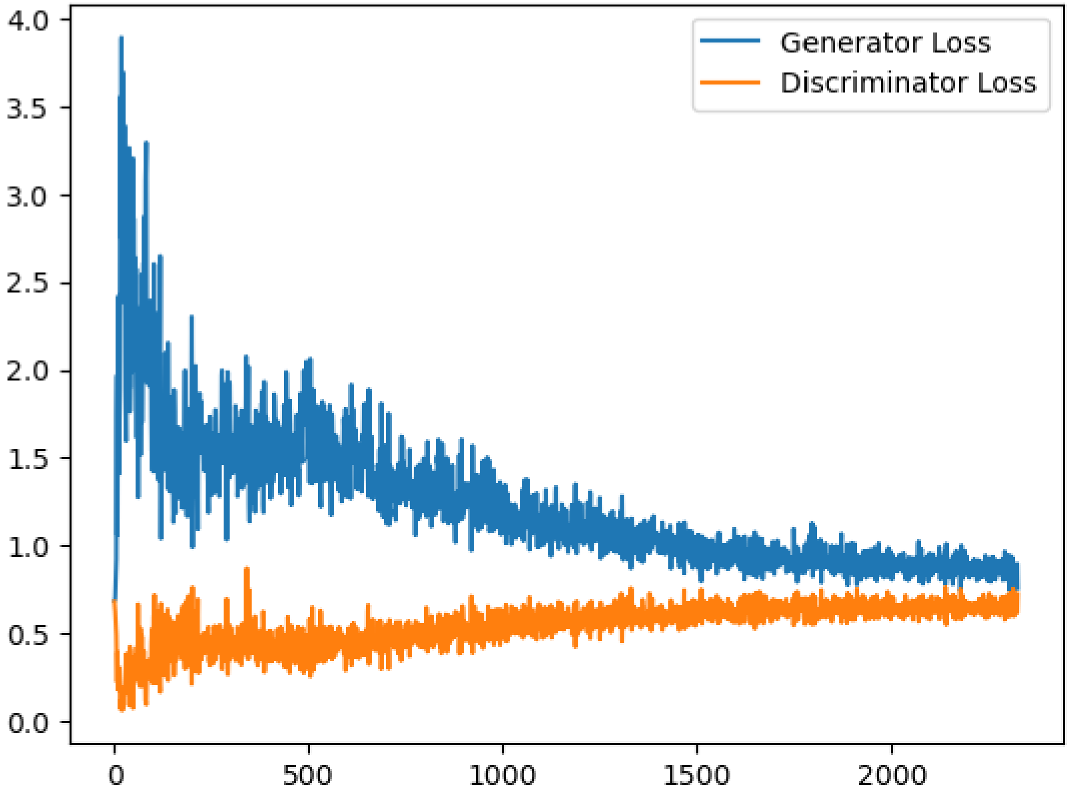


Figure 14 – Experiment 1: Generator vs. Discriminator Loss

```
Step 500: Generator loss: 2.1040075484514236, discriminator loss:
0.27637109272927046
```

```
Step 46500: Generator loss: 0.8457962599992752, discriminator loss:
0.6615329077243804
```

Comments:

Step 500: Early Epochs

- **Generator Loss (2.104)**: At this early stage, the generator's loss is quite high, indicating that it is struggling to create realistic images that can fool the discriminator. This is expected, as the generator initially produces poor quality outputs.
- **Discriminator Loss (0.276)**: The low discriminator loss suggests that the discriminator is effectively distinguishing between real and fake images early in the training process. It has a relatively easier task since the generator's outputs are not yet convincing.

Step 46,500: Later Epochs

- **Generator Loss (0.846)**: The significant decrease in the generator's loss indicates improved performance. The generator is now producing more realistic images that increasingly challenge the discriminator. A lower generator loss suggests better quality images.
- **Discriminator Loss (0.662)**: The increase in the discriminator's loss reflects the improved performance of the generator. As the generator gets better at creating realistic images, the discriminator finds it more challenging to correctly identify fake images, leading to higher loss values.

Key Observations

- **Learning Dynamics**: The initial high generator loss and low discriminator loss are typical in GAN training. As training progresses, the generator improves, resulting in a decrease in its loss, while the discriminator's task becomes harder, reflected in an increased loss.
- **Balancing Act**: GANs involve a delicate balance where the generator tries to outsmart the discriminator and vice versa. The losses provide a good indication of how well each model is performing and adjusting during training.
- **Convergence**: By the end of 50 epochs, the lower generator loss and higher discriminator loss suggest that the models are approaching an equilibrium. This is a desirable outcome, indicating that both models are learning effectively from each other.

Summary:

The results from this run demonstrate the typical progression of GAN training. Initially, the generator struggles to produce convincing outputs, while the discriminator performs well. Over time, the generator improves, making it harder for the discriminator to distinguish between real and fake images. These dynamics are crucial for the successful training of GANs and achieving high-quality generated images. Regular monitoring and analysis of these loss values can help in making necessary adjustments to hyperparameters for optimal training outcomes.

2.3.2 Experiment 2: Number of Epochs = 100

Hyperparameter	Value
criterion	BCEWithLogitsLoss()
n_epochs	100
z_dim	64
display_step	500
Batch size	64

Hyperparameter	Value
Learning rate	0.0002

Table 16 – Highlighted Cell showing changed parameter

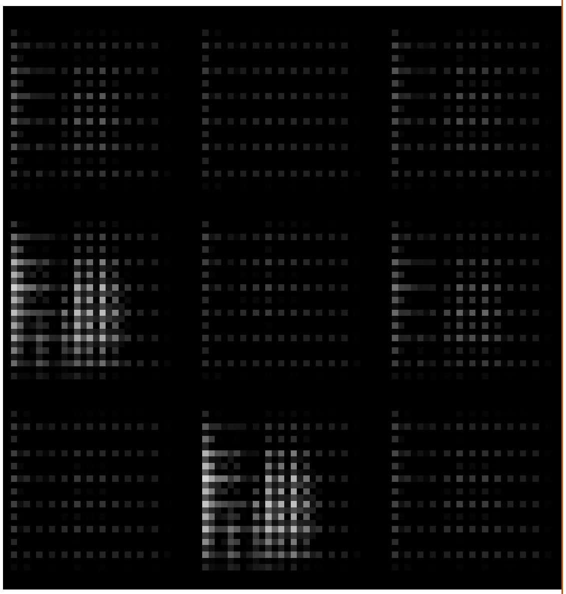
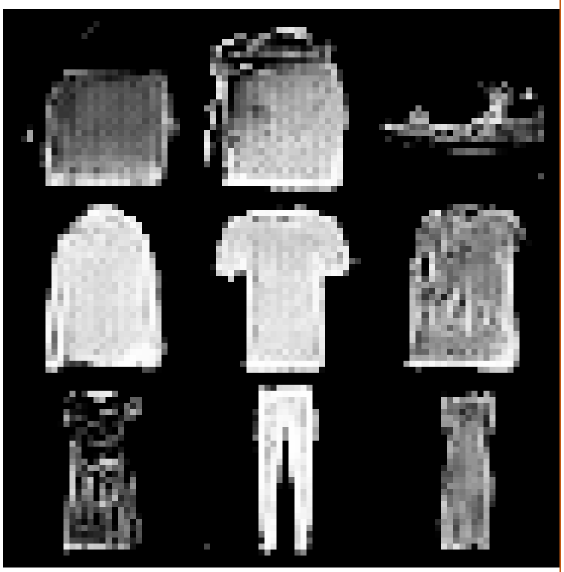
Epoch	Fake	Real
0		
99		

Table 17 – Fake/Real Images from first epoch (0) and last epoch (99)

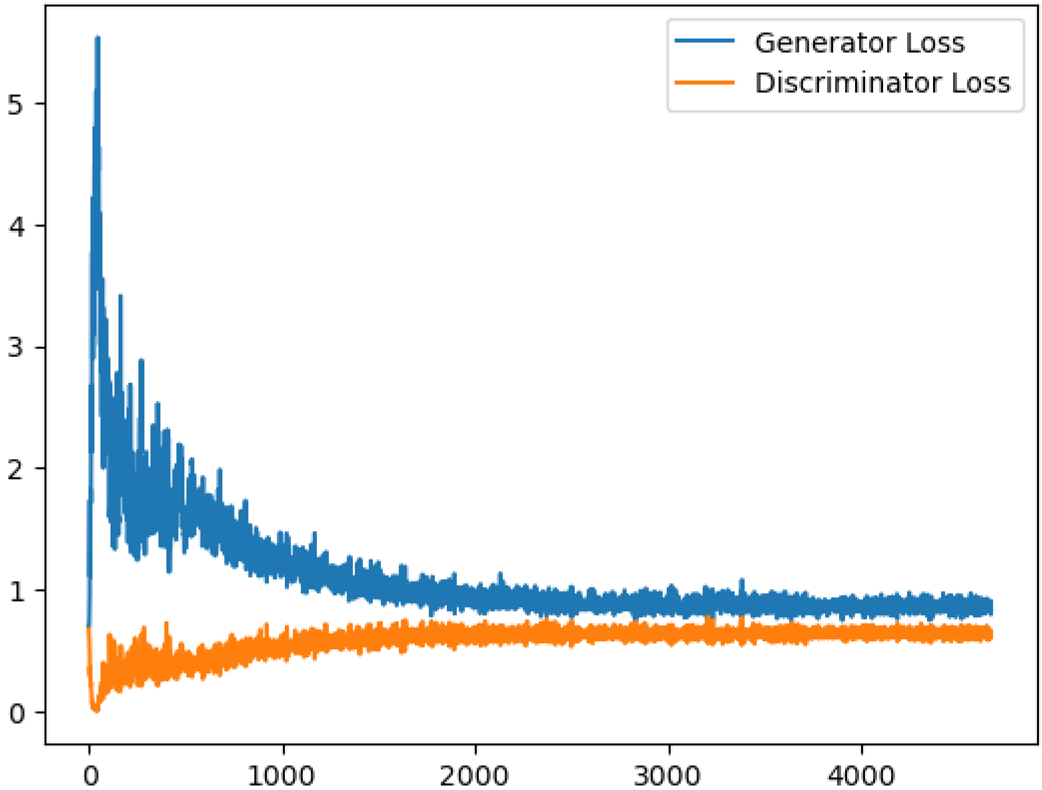


Figure 15 – Experiment 2: Generator vs. Discriminator Loss

```
Step 500: Generator loss: 2.1323656688928603, discriminator loss:
0.25505777245759964
Step 93500: Generator loss: 0.8645319269895554, discriminator loss:
0.6384353876113892
```

Comments:

Step 500: Early Epochs

- **Generator Loss (2.132)**: At this early stage, the generator's loss is quite high, indicating that it is struggling to produce realistic images. This is expected as the generator initially generates poor-quality outputs.
- **Discriminator Loss (0.255)**: The discriminator's low loss shows that it is effectively distinguishing between real and fake images, as it finds the generator's outputs relatively easy to identify as fake.

Step 93,500: Later Epochs

- **Generator Loss (0.865)**: By this stage, the generator's loss has significantly decreased. This implies that the generator is getting better at creating more realistic images that challenge the discriminator. A lower loss suggests improved image quality.
- **Discriminator Loss (0.638)**: The increase in the discriminator's loss indicates that it is finding it more difficult to distinguish between real and fake images as the generator improves. This reflects a more challenging task for the discriminator due to the higher quality of generated images.

Key Observations

- **Learning Dynamics**: Initially, the high generator loss and low discriminator loss are typical as the generator starts learning to generate images. Over time, the generator improves, leading to a decrease in its loss and an increase in the discriminator's loss.
- **Model Performance**: The significant decrease in generator loss and the corresponding increase in discriminator loss suggest that both models are learning effectively. The generator is producing better-quality images, making the discriminator's task more difficult.
- **Training Stability**: Over 100 epochs, the progression of loss values indicates stable and effective training. There are no signs of instability or mode collapse, which can sometimes occur in GAN training.

Summary

The results of the 100-epoch run demonstrate typical GAN training dynamics. The generator starts with high loss and improves significantly over time, challenging the discriminator, which sees an increase in its loss. This balance between the two models is crucial for achieving high-quality generated images. Regular monitoring of these loss values helps in understanding the training progress and making necessary adjustments to hyperparameters for optimal performance. Overall, the training appears successful, with both the generator and discriminator improving their respective tasks, leading to better-quality generated images.

2.3.3 Experiment 3: Reduce the display step to 200

Hyperparameter	Value
criterion	BCEWithLogitsLoss()
n_epochs	100
z_dim	64
display_step	200
Batch size	64
Learning rate	0.0002

Table 18 – Highlighted Cell showing changed parameter

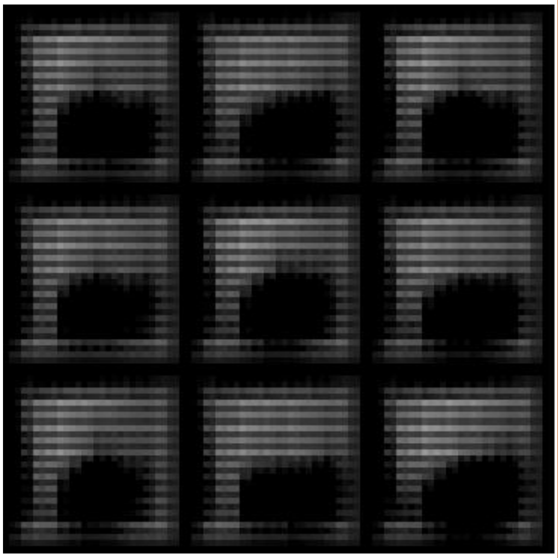
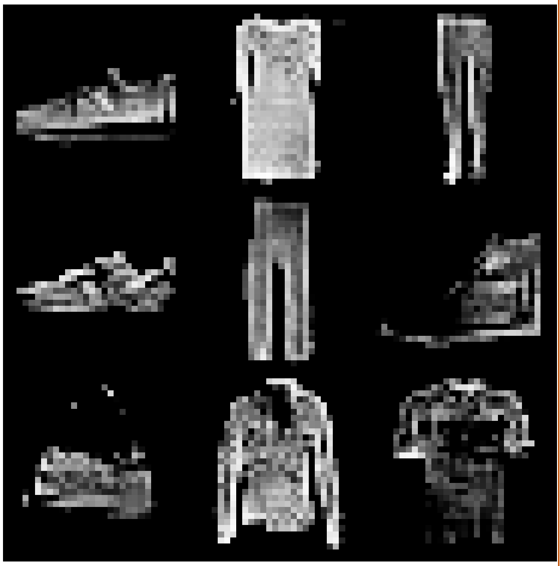
Epoch	Fake	Real
0		
99		

Table 19 – Fake/Real Images from first epoch (0) and last epoch (99), with display step set to 200

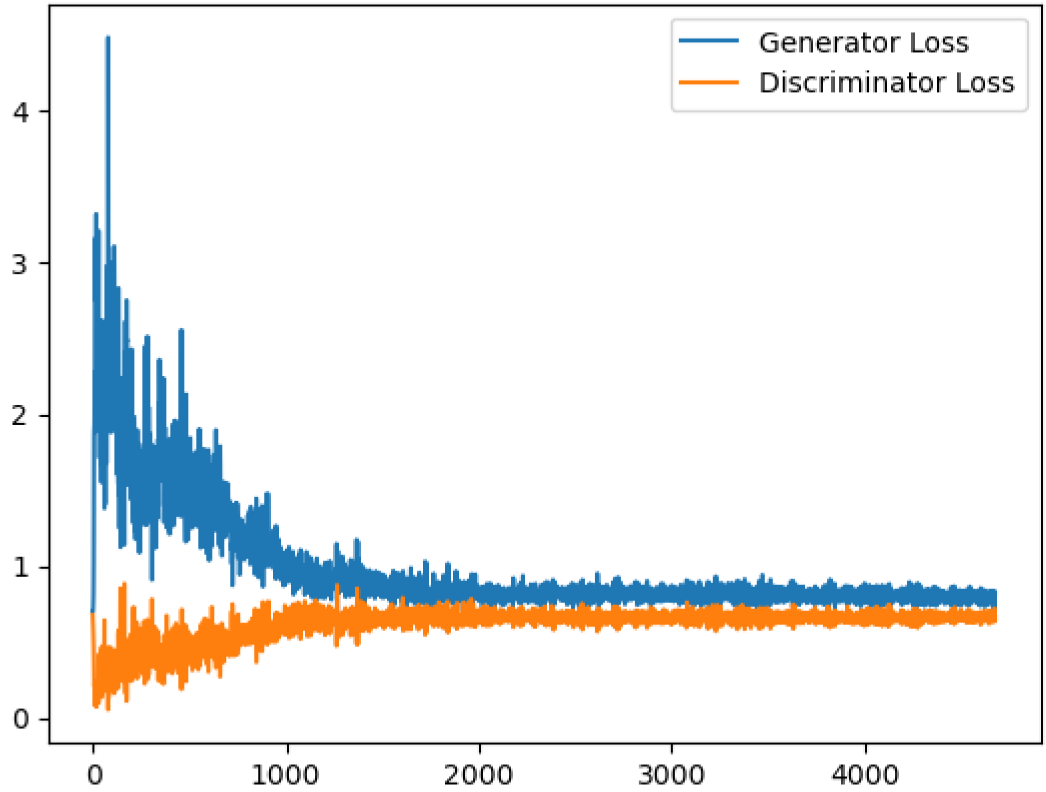


Figure 16 – Experiment 3: Generator vs. Discriminator Loss

```
Step 200: Generator loss: 1.209487415254116, discriminator loss:
0.45477232001721857
Step 93600: Generator loss: 0.784689607322216, discriminator loss:
0.6701694121956825
```

Comments:

Step 200: Early Epochs

- **Generator Loss (1.209)**: At this early stage, the generator's loss is moderately high, indicating that it is still learning to produce realistic images.
- **Discriminator Loss (0.455)**: The discriminator's lower loss shows it is effectively distinguishing between real and fake images, which is expected since the generator's outputs are not yet convincing.

Step 93,600: Later Epochs

- **Generator Loss (0.785)**: By this stage, the generator's loss has significantly decreased, reflecting its improved ability to generate realistic images that can fool the discriminator.
- **Discriminator Loss (0.670)**: The increased discriminator loss suggests that it is finding it more challenging to distinguish between real and fake images due to the generator's improved outputs.

Key Observations

- **Early Learning**: The initial higher generator loss and lower discriminator loss indicate that the generator is still learning basic patterns and the discriminator is effectively recognizing fake images.
- **Improvement Over Time**: The significant reduction in generator loss over time indicates that the generator is learning and producing better-quality images. The increase in

discriminator loss shows that the discriminator is now challenged by these improved images.

- **Training Dynamics**: The balance between the generator and discriminator losses is crucial in GAN training. The decreased generator loss combined with an increased discriminator loss suggests that both models are learning effectively and pushing each other to improve.

Impact of Reduced Display Step

- **More Frequent Updates**: By reducing the display step from 500 to 200, you get more frequent updates on the training progress. This can provide a more granular view of how the model is performing at different stages, helping to make timely adjustments if needed.

Summary:

The results of the 100-epoch run show a typical learning progression in GAN training. The initial higher generator loss indicates the model's struggle to produce realistic images, while the lower discriminator loss shows its effectiveness in recognizing fakes. Over time, the generator improves, leading to a decreased loss and higher quality images. The increased discriminator loss towards the later epochs indicates that the generator's outputs are becoming more challenging to distinguish from real images. Overall, the model exhibits effective learning dynamics, with both the generator and discriminator improving their respective tasks, leading to higher-quality generated images. Regular monitoring of these trends can help guide further training and hyperparameter adjustments for optimal performance.

2.3.4 Experiment 4: Increase latent dimension to 200

Hyperparameter	Value
criterion	BCEWithLogitsLoss()
n_epochs	100
z_dim	200
display_step	500
Batch size	64
Learning rate	0.0002

Table 20 – Highlighted Cell showing changed parameter

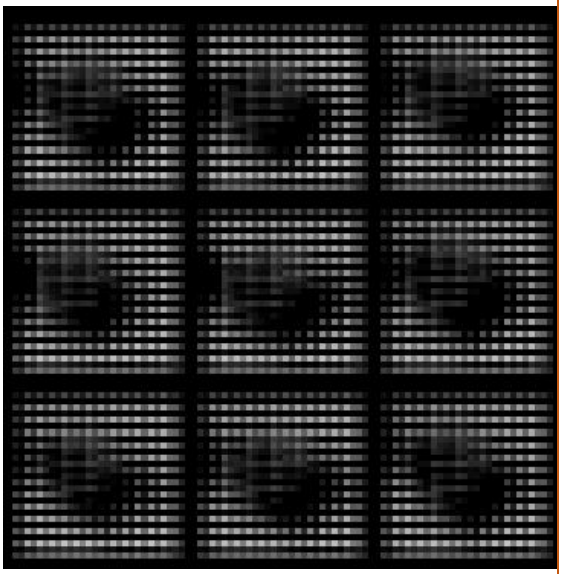
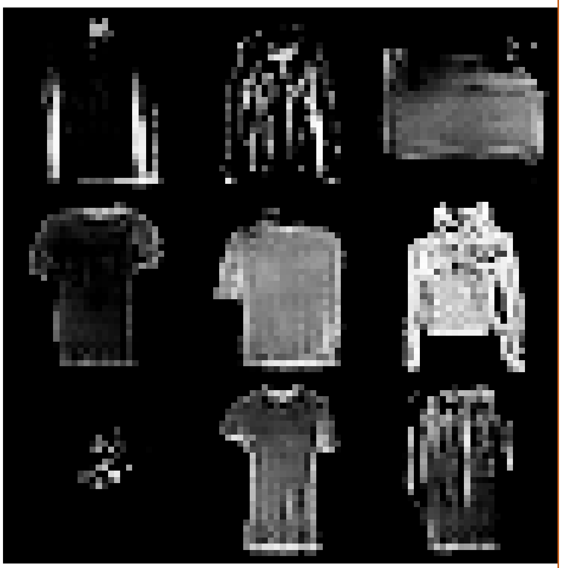
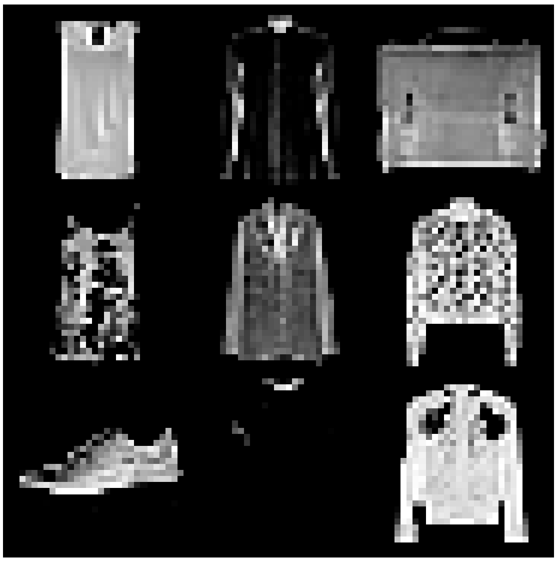
Epoch	Fake	Real
0		
99		

Table 21 – Fake/Real Images from first epoch (0) and last epoch (99), with latent dimension set to 200

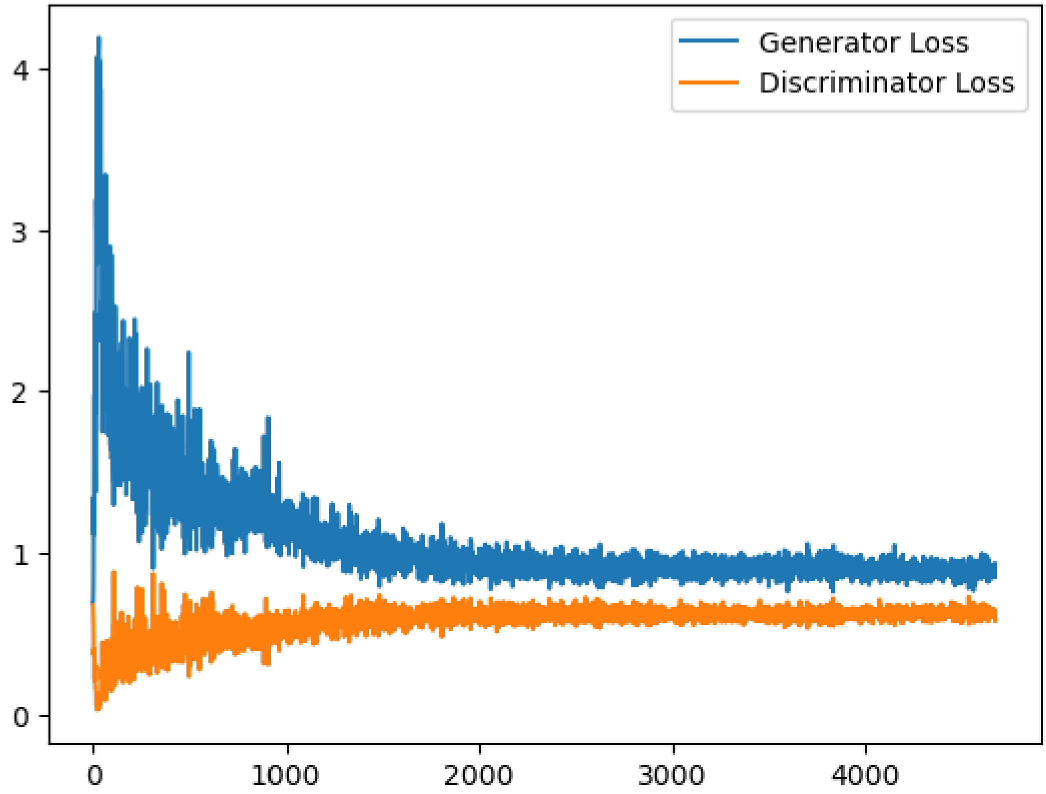


Figure 17 – Experiment 4: Generator vs. Discriminator Loss

```
Step 500: Generator loss: 1.9759931046962738, discriminator loss:
0.2729939045459032
Step 93500: Generator loss: 0.8901273909807205, discriminator loss:
0.632079149723053
```

Comments:

Step 500: Early Epochs

- **Generator Loss (1.976)**: At this early stage, the generator's loss is quite high. The increased latent dimension likely introduces more complexity, making it initially harder for the generator to produce realistic images.
- **Discriminator Loss (0.273)**: The discriminator's low loss indicates it is effectively distinguishing between real and fake images, as the generator's outputs are still relatively poor in quality.

Step 93,500: Later Epochs

- **Generator Loss (0.890)**: By this stage, the generator's loss has decreased significantly, suggesting that it is producing much better-quality images. The larger latent space potentially allows the generator to capture more complex data distributions, improving the realism of generated images.
- **Discriminator Loss (0.632)**: The increase in the discriminator's loss shows that it is finding it more challenging to differentiate between real and fake images, indicating that the generator has become more adept at fooling the discriminator.

Key Observations

- **Increased Latent Dimension Impact**: Increasing the latent dimension adds more complexity to the generator's input space, potentially allowing for richer and more detailed generated images. However, this also means the generator initially struggles more, reflected in the high early loss.
- **Learning Dynamics**: The initial high generator loss and low discriminator loss are typical, but the significant improvement in the generator's performance over time suggests that the model effectively learns to leverage the increased latent space.
- **Balance and Convergence**: By the end of the training, the generator and discriminator losses indicate a good balance, with the generator producing higher quality images and the discriminator facing increased difficulty in distinguishing them from real images.

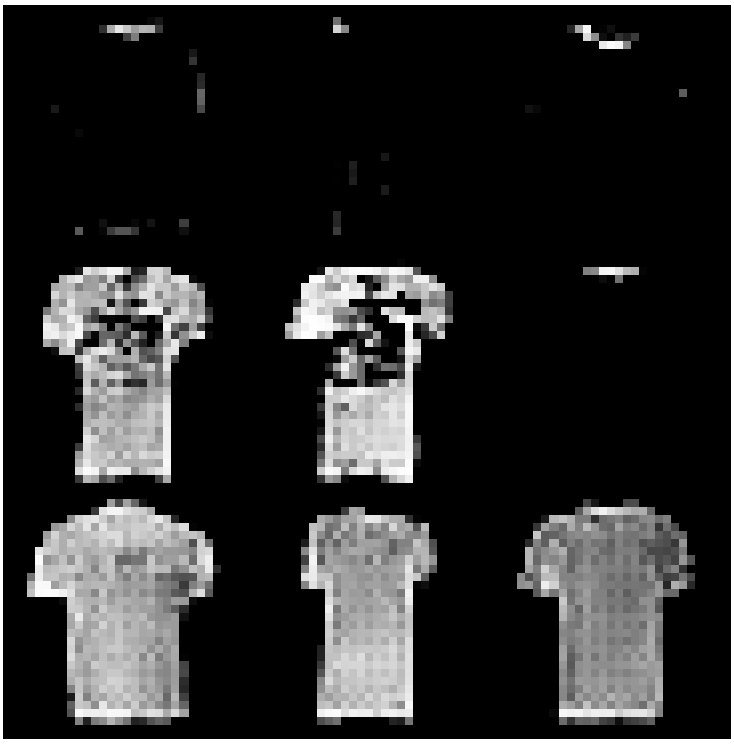
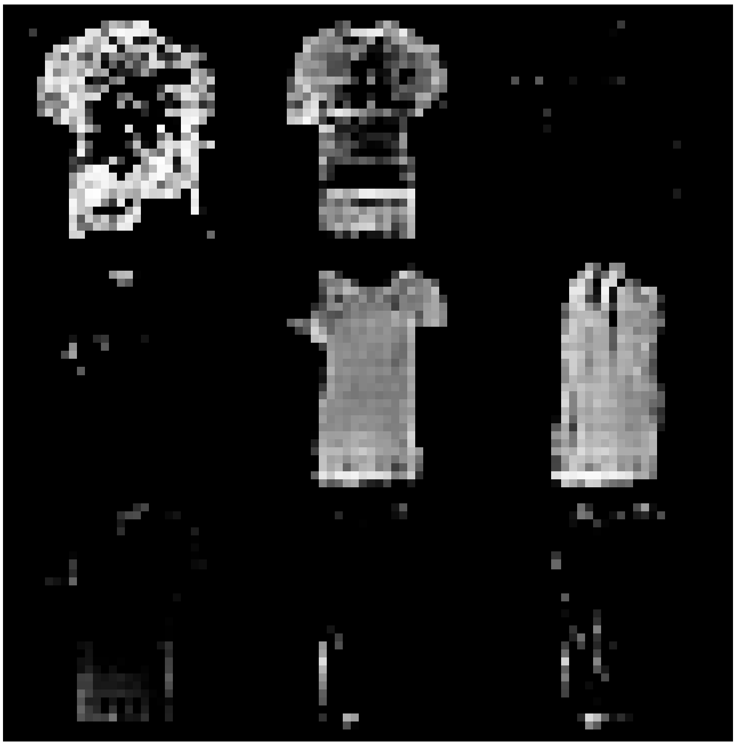
Summary:

The results of this run highlight the impact of increasing the latent dimension on the training dynamics of GANs. The generator initially struggles more but eventually learns to produce more realistic images, as indicated by the significant decrease in its loss. The higher latent dimension allows the generator to create richer and more complex representations, leading to better overall performance. The increased discriminator loss towards the later epochs confirms that the generator's improvements are making it more challenging for the discriminator to differentiate between real and fake images.

This setup demonstrates the potential benefits of a larger latent space, but also the need for careful monitoring and potentially longer training times to fully realize those benefits. Overall, the model shows effective learning and convergence, producing higher-quality images as training progresses.

2.3.5 New Samples generated with Conditional GANs (T-Shirt)

Experiment #	Generated (T-Shirt)
Experiment 1	

Experiment #	Generated (T-Shirt)
Experiment 2	
Experiment 3	

Experiment #	Generated (T-Shirt)
Experiment 4	

Table 22 – New images generated with conditional GANs

2.4 Model 4: Conditional Diffusion Models

Four experiments were carried out:

The baseline model started with 50 epochs, we then moved to the second experiment increasing the number of epochs to 100. The third experiment lowered the learning rate to 0.0001 and the last changed the batch size from 128 to 64.

2.4.1 Experiment 1: Number of Epochs = 50

Hyperparameter	Value
Criterion	mse_loss
Batch Size	128
n_epochs	50
Timestep	150
Optimizer	Adam
Learning rate	0.001

Table 23 – Highlighted Cell showing changed parameter

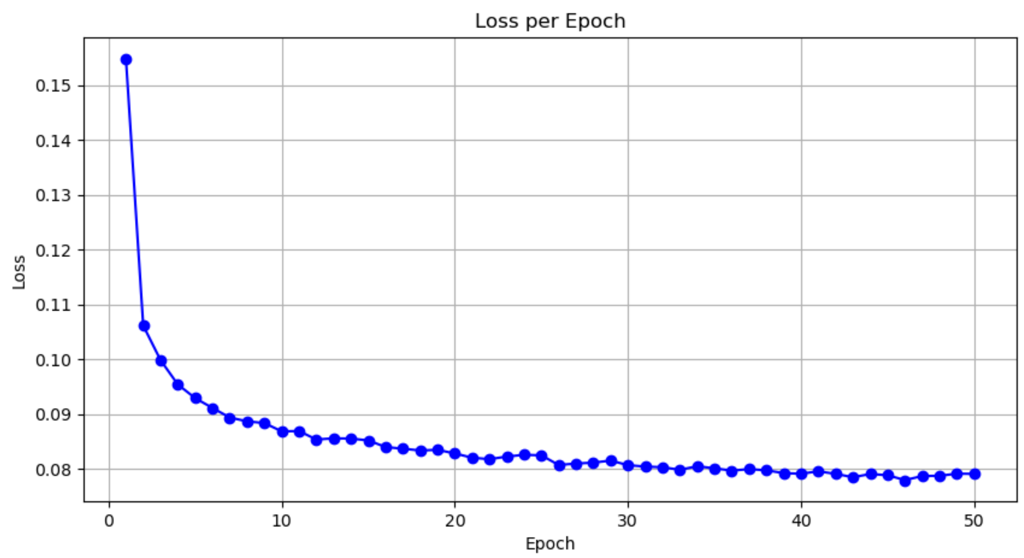


Figure 18 – Experiment 1: Loss vs Epochs (50)

Comments:

Initial Epochs (1-10)

- **High Initial Loss**: The loss starts at 0.1549, indicating significant error at the beginning of training.
- **Rapid Decrease**: The loss decreases sharply to 0.0868 by the 10th epoch, reflecting the model's ability to quickly learn basic features of the data.

Middle Epochs (11-30)

- **Continued Decline**: The loss values continue to decrease, albeit more gradually. By the 30th epoch, the loss drops to around 0.0807, showing steady improvement and refinement of the model.
- **Stability**: The relatively consistent decline during these epochs suggests stable learning without major fluctuations.

Later Epochs (31-50)

- **Gradual Plateau**: From epochs 31 to 50, the decline in loss values slows down and begins to plateau, with values stabilizing around 0.079. This indicates the model is nearing its optimal performance.
- **Minor Fluctuations**: Small fluctuations in the loss values are observed, which is normal as the model makes fine adjustments.

Key Observations

- **Effective Learning Rate**: A learning rate of 0.001 appears to be appropriate, allowing for steady learning without abrupt changes.
- **Batch Size Impact**: The batch size of 128 enables efficient learning with stable updates to the model weights.
- **Early Learning vs. Refinement**: The sharp decline in the initial epochs reflects rapid learning of basic patterns, while the gradual plateau indicates the model's transition to fine-tuning complex features.

Analysis

- **Initial Rapid Learning**: The steep decrease in initial epochs signifies the model's quick adaptation to the dataset's fundamental patterns.
- **Gradual Improvement**: The middle epochs show a consistent and gradual decrease in loss, reflecting effective learning and parameter refinement.
- **Plateau and Convergence**: The plateau in the later epochs suggests the model is nearing convergence, with minor fluctuations indicating fine-tuning.

Summary:

The conditional diffusion model trained with these parameters demonstrates effective learning and convergence. The loss values show a well-behaved training process with rapid initial improvement, followed by gradual refinement and eventual stabilization. The chosen learning rate and batch size facilitate efficient learning, resulting in a model that effectively captures the underlying patterns in the dataset.

2.4.2 Experiment 2: Number of Epochs = 100

Hyperparameter	Value
Criterion	mse_loss
Batch Size	128
n_epochs	100
Timestep	150
Optimizer	Adam
Learning rate	0.001

Table 24 – Highlighted Cell showing changed parameter

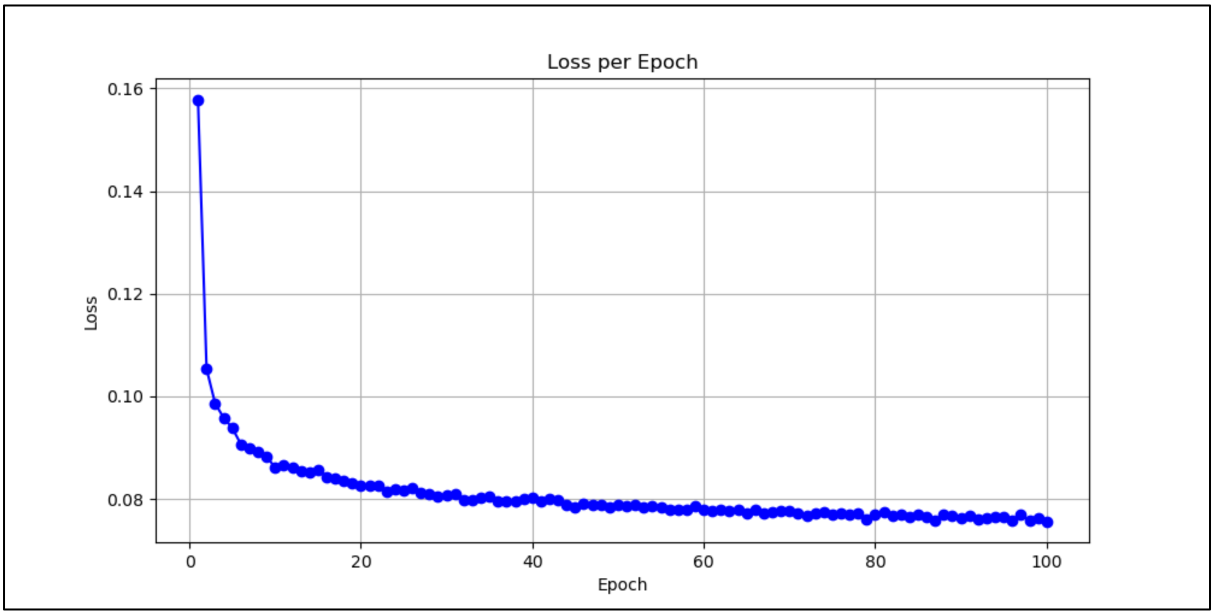


Figure 19 – Experiment 2: Loss vs Epochs (100)

Comments:

Initial Epochs (1-10)

- **High Initial Loss**: The starting loss of 0.1579 indicates significant initial error.

- **Rapid Decrease**: By the 10th epoch, the loss value drops to 0.0860, showing rapid learning of the dataset's basic patterns.

Middle Epochs (11-50)

- **Continued Decline**: The loss values continue to decrease, though more gradually, to around 0.079 by the 50th epoch. This period reflects steady learning and refinement.
- **Fluctuations**: Minor fluctuations in loss values suggest the model is adjusting its parameters to better fit the data.

Later Epochs (51-100)

- **Gradual Plateau**: The loss values plateau further, ranging around 0.076 to 0.078. This stabilization indicates the model has reached an optimal performance level.
- **Fine-tuning**: The slight fluctuations in loss values during these epochs show the model's ongoing fine-tuning.

Key Observations

- **Effective Learning Rate**: The learning rate of 0.001 is well-suited, allowing for stable and effective learning without dramatic swings.
- **Batch Size Impact**: The batch size of 128 helps in maintaining stable updates to the model weights, contributing to a smooth loss curve.
- **Longer Training Benefits**: Extending training to 100 epochs provides ample time for the model to refine its learning and achieve better performance.

Analysis

- **Rapid Initial Learning**: The sharp decrease in the initial epochs signifies effective learning of foundational patterns in the data.
- **Steady Improvement**: The middle epochs' gradual decline indicates consistent learning and refinement of complex patterns.
- **Plateau and Convergence**: The later epochs' plateau suggests the model is nearing optimal performance, with minimal further improvements despite continued training.

Summary:

The model shows effective initial learning, steady refinement, and eventual convergence, as reflected in the loss values.

2.4.3 Experiment 3: Set Learning Rate = 0.0001

Hyperparameter	Value
Criterion	mse_loss
Batch Size	128
n_epochs	50
Timestep	150
Optimizer	Adam
Learning rate	0.0001

Table 25 – Highlighted Cell showing changed parameter

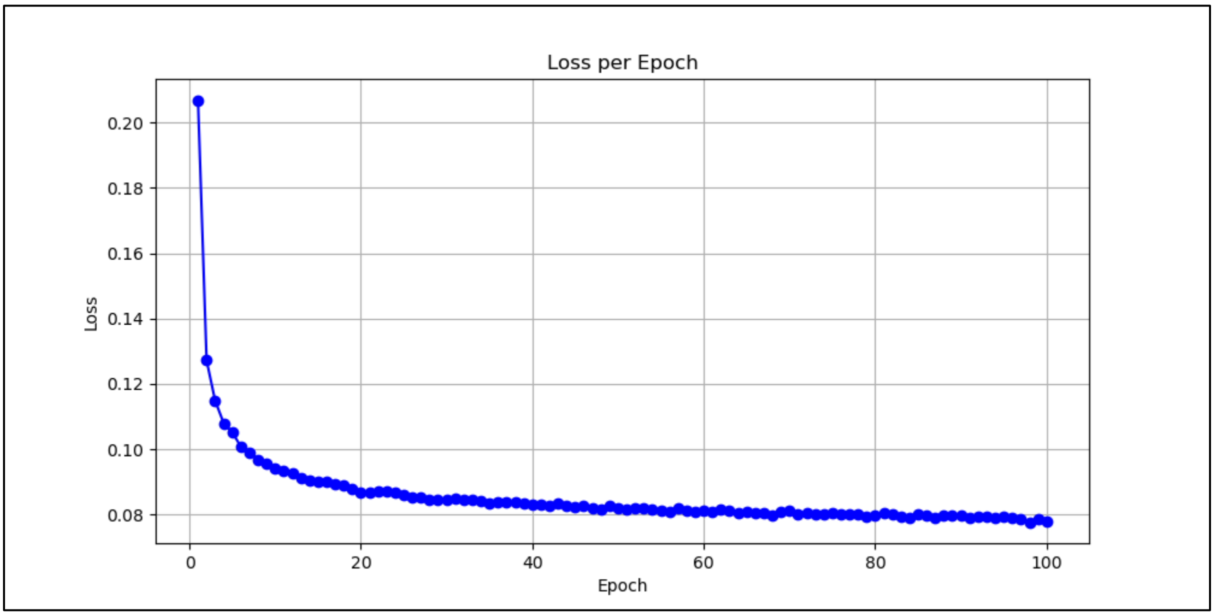


Figure 20 – Experiment 3: Loss vs Epochs

Comments:

Initial Epochs (1-10)

- **High Initial Loss**: Starting with a high initial loss of 0.2068 reflects the significant initial error.
- **Steady Decrease**: The loss values drop steadily, reaching 0.0943 by the 10th epoch, indicating effective initial learning of basic data patterns.

Middle Epochs (11-50)

- **Consistent Decline**: The loss values continue to decrease gradually, reflecting stable learning. By the 50th epoch, the loss drops to around 0.0814.
- **Minor Fluctuations**: Small fluctuations in loss values suggest the model is adjusting its parameters to better fit the data.

Later Epochs (51-100)

- **Gradual Plateau**: The loss values further stabilize around 0.078 to 0.079. This plateauing effect indicates the model is approaching its optimal performance.
- **Fine-tuning**: Slight variations in loss values during these epochs show the model is making fine adjustments.

Key Observations

- **Reduced Learning Rate**: The lower learning rate of 0.0001 leads to more gradual and stable learning, avoiding large updates and ensuring fine-grained improvements.
- **Batch Size Impact**: A batch size of 128 provides stable updates and consistent learning.
- **Extended Training Benefits**: Extending training to 100 epochs allows the model to refine its learning further, achieving better performance and stability.

Analysis

- **Initial Learning**: The initial epochs' steady decrease signifies effective adaptation to the dataset's basic patterns.

- **Gradual Improvement**: The middle epochs show a consistent decline in loss, reflecting the model's ability to learn and refine more complex patterns.
- **Plateau and Stability**: The plateau in the later epochs suggests that the model has reached a point of minimal further improvement, indicating convergence and stability.

Summary:

The conditional diffusion model demonstrates a well-structured learning process with a lower learning rate of 0.0001, batch size of 128, and extended training over 100 epochs. The loss values reflect effective initial learning, gradual refinement, and eventual stabilization. The chosen hyperparameters facilitate stable learning and fine-tuning, resulting in a model that captures the underlying patterns in the dataset effectively.

2.4.4 Experiment 4: Change Batch Size to 64

Hyperparameter	Value
Criterion	mse_loss
Batch Size	64
n_epochs	100
Timestep	150
Optimizer	Adam
Learning rate	0.001

Table 26 – Highlighted Cell showing changed parameter

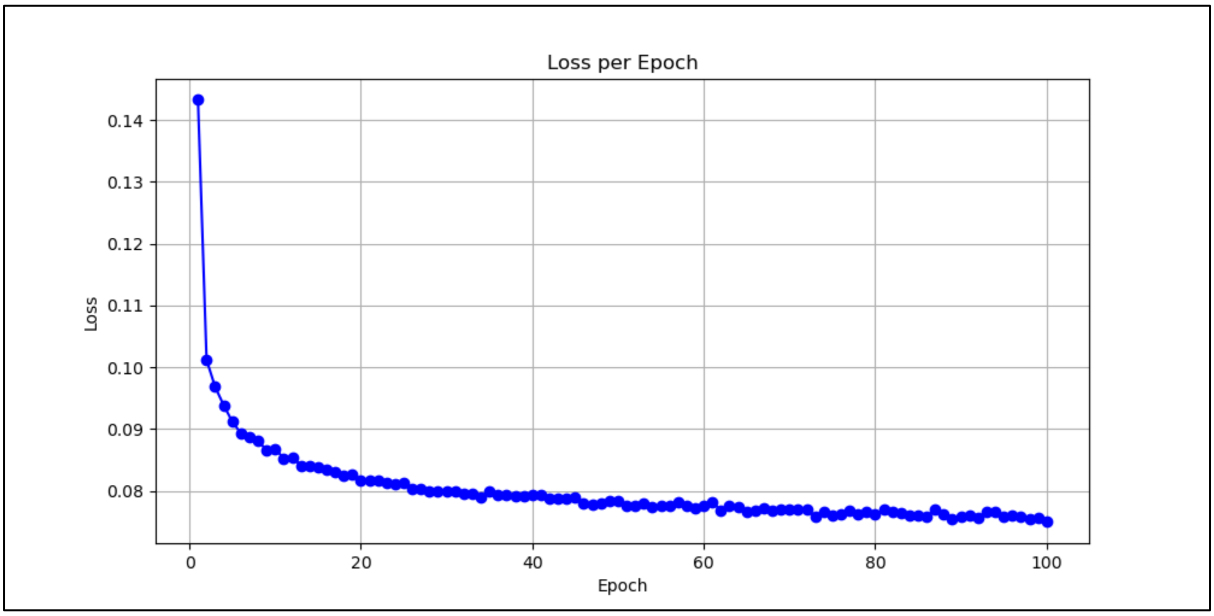


Figure 21 – Experiment 4: Loss vs Epochs

Comments:

Initial Epochs (1-10)

- **Initial Loss**: Starting with a high initial loss of 0.1432, the model quickly reduces the loss to 0.0865 within the first 10 epochs. This rapid decrease indicates effective initial learning and adaptation to the dataset.
- **Stability**: The loss values show a consistent and smooth decline, reflecting stable and efficient learning processes during the early stages of training.

Middle Epochs (11-50)

- **Gradual Improvement**: The loss values continue to decrease steadily, reaching approximately 0.078 by the 50th epoch. This gradual decline highlights the model's capability to refine and optimize its parameters over time.
- **Minor Fluctuations**: Small fluctuations in loss values are observed, indicating continuous adjustment and fine-tuning of the model.

Later Epochs (51-100)

- **Plateau and Convergence**: From the 51st to the 100th epoch, the loss values stabilize around 0.075 to 0.078. The plateauing effect suggests the model has reached its optimal performance and further training yields minimal improvements.
- **Slight Variations**: Although there are slight variations in the loss values, they remain within a narrow range, reflecting the model's fine-tuning and stabilization.

Key Observations

- **Learning Rate Effect**: The higher learning rate of 0.001 enables faster convergence and efficient learning. However, it also requires careful monitoring to avoid overshooting optimal parameters.
- **Batch Size Impact**: The smaller batch size of 64 results in more frequent updates to the model weights, contributing to finer adjustments and improved convergence.
- **Extended Training Benefits**: Training for 100 epochs provides sufficient time for the model to stabilize and achieve better overall performance.

Analysis

- **Initial Rapid Learning**: The sharp decline in loss values during the initial epochs reflects the model's ability to quickly adapt and learn from the data.
- **Steady Refinement**: The middle epochs show consistent improvement, indicating effective learning and parameter optimization.
- **Plateau and Fine-tuning**: The plateau in the later epochs suggests the model has reached a point of minimal further improvement, with the focus shifting to fine-tuning learned patterns.

Summary:

The conditional diffusion model trained with these hyperparameters demonstrates a well-behaved learning process. The rapid initial decrease in loss values, followed by a steady decline and eventual plateau, indicates effective learning and convergence. The chosen higher learning rate and smaller batch size facilitate efficient learning and fine-grained adjustments, resulting in a model that captures the underlying patterns in the dataset effectively.

2.4.5 New Samples generated from Experiments (To generate Sneaker)

2.4.5.1 Experiment 1: Number of Epochs – 50

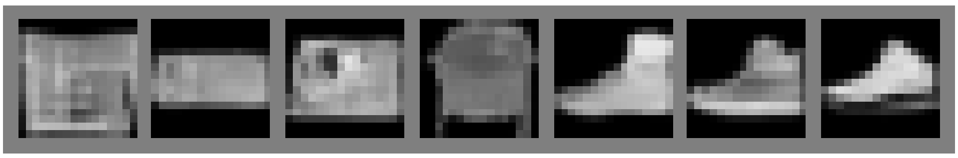


Figure 22 – New samples generated from Experiment 1

2.4.5.2 Experiment 2: Number of Epochs – 100

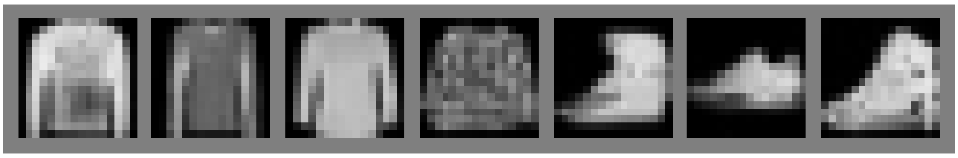


Figure 23 – New samples generated from Experiment 2

2.4.5.3 Experiment 3: Learning Rate = 0.0001 (from 0.001)

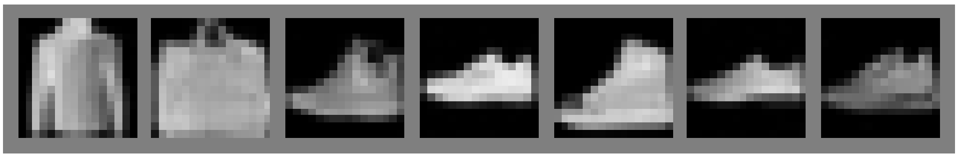


Figure 24 – New samples generated from Experiment 3

2.4.5.4 Experiment 4: Change Batch Size to 64 (from 128)

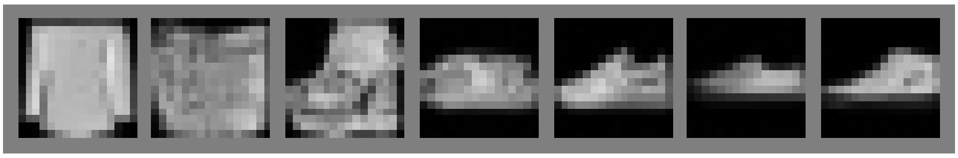


Figure 25 – New samples generated from Experiment 4

3 Reflection of course outcomes

This course has significantly enhanced my understanding of generative AI—both in terms of theoretical principles and practical applications.

Core Principles and Applications of Generative AI Models:

In this course, we were taught the core principles behind generative AI, including Variational Autoencoders (VAEs), Generative Adversarial Networks (GANs), and diffusion models. In VAEs, we were shown how to leverage probabilistic inference to generate new data by learning a lower-dimensional latent representation, while GANs introduced us to the adversarial process where two models—generator and discriminator—compete to improve the quality of generated data. The course's explanation of diffusion models—which iteratively transform noise into structured data—was particularly interesting, showcasing how these models have gained prominence in applications like image generation. Understanding these foundational models has given me a solid framework for grasping how generative AI works and how different techniques can be applied to achieve various creative tasks, such as image synthesis, video generation.

Building and Fine-Tuning Generative Models:

The hands-on experience of building generative AI models for image generation was an essential part of the course. We were able to implement models like GANs and VAEs from scratch, which helped me translate theoretical knowledge into practical skills. One of the key learnings was how to fine-tune model hyperparameters—such as learning rate, batch size, and model architecture—to optimize performance and improve the quality of generated images. This process highlighted the importance of experimentation in machine learning, as tuning these parameters significantly impacted the model's ability to generate realistic images. By engaging with this aspect of model development, I gained confidence in my ability to not only understand but also apply these models effectively.

Controlled Generative Processes:

Another key takeaway from the course was the ability to control and guide generative processes. I learned how to influence the output of AI models by adjusting parameters and conditions, such as using conditional GANs (cGANs) to generate images based on specific inputs (e.g., generating images of a particular style or class). This ability to steer AI models toward desired outcomes opened up a new dimension in creativity and control, offering the potential for highly tailored AI applications in fields like art, design, and even healthcare.

AI Ethics

Evaluating and applying ethical considerations in the design and deployment of generative AI models is important. Addressing issues such as data privacy, bias in model training, and the potential for misuse emphasized the responsibility that comes with developing powerful AI technologies. The discussions and case studies around AI ethics have instilled a strong ethical framework, guiding how we can approach future AI projects to ensure they are fair, transparent, and respect user privacy.

Conclusion:

In summary, this course has greatly enhanced my understanding of generative AI, providing me with both a basic theoretical foundation and practical experience in building and fine-tuning generative models. The ability to control generative processes and fine-tune models has empowered me to take on more complex projects, confident in my understanding of how these models work and how to optimize them for various use cases. Whether in creative fields or more technical applications, the skills and knowledge gained from this course will be invaluable as I continue exploring the exciting potential of generative AI.