**University of Central Missouri**

**Department of Computer Science & Cybersecurity**

**CS5720 Neural network and Deep learning**

**Spring 2025**

**Home Assignment 5. (Cover Ch 11, 12)**

**Student name:**

**Submission Requirements:**

* Total Points: 100
* Once finished your assignment push your source code to your repo (GitHub) and explain the work through the ReadMe file properly. Make sure you add your student info in the ReadMe file.
* Submit your GitHub link and video on the BB.
* Comment your code appropriately ***IMPORTANT.***
* Make a simple video about 2 to 3 minutes which includes demonstration of your home assignment and explanation of code snippets.
* Any submission after provided deadline is considered as a late submission.

**1. GAN Architecture**

Explain the adversarial process in GAN training. What are the goals of the generator and discriminator, and how do they improve through competition? Diagram of the GAN architecture showing the data flow and objectives of each component.

Generative Adversarial Networks (GANs) are a class of machine learning frameworks where two neural networks the generator and the discriminator compete in a gametheoretic setting. Here's a breakdown of how the adversarial process works:

**Goals of Each Component:**

1.Generator (G)

Goal: Generate data (usually images) that looks as real as possible.

Input: Random noise vector (z), usually sampled from a uniform or normal distribution.

Output: Synthetic data (e.g., an image).

Objective: Fool the discriminator into classifying fake data as real.

2.Discriminator (D)

Goal: Distinguish between real data (from the dataset) and fake data (from the generator).

Input: Either real data or synthetic data.

Output: A probability score indicating whether the input is real (close to 1) or fake (close to 0).

Objective: Correctly classify real vs fake.

Adversarial Training Process:

Step 1: The generator produces fake data from random noise.

Step 2: The discriminator evaluates both real and generated data.

Step 3: The discriminator updates its weights to better classify real vs fake.

Step 4: The generator updates its weights to produce more realistic data and fool the discriminator.

This competition continues until the generated data becomes indistinguishable from real data.

Optimization Objectives (Simplified):

Discriminator Loss (D):

\[

\max\_D \mathbb{E}[\log D(x)] + \mathbb{E}[\log(1 D(G(z)))]

\]

Generator Loss (G):

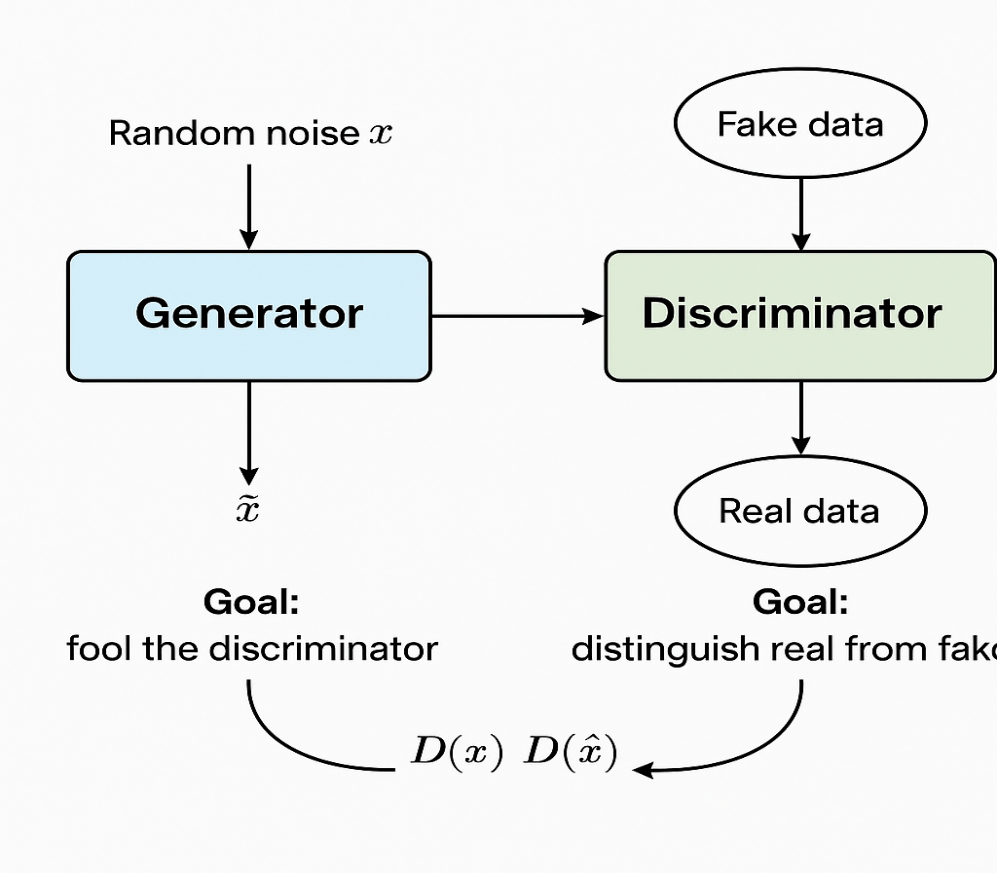
\[

\min\_G \mathbb{E}[\log(1 D(G(z)))]

\]

(Or alternatively, maximize \(\log D(G(z))\) for better gradients in practice.)

Diagram of GAN Architecture:



**2. Ethics and AI Harm**

Choose one of the following realworld AI harms discussed in Chapter 12:

* Representational harm
* Allocational harm
* Misinformation in generative AI

Describe a real or hypothetical application where this harm may occur. Then, suggest **two harm mitigation strategies** that could reduce its impact based on the lecture.

**Application Example**:

Fake News Generation via AI  
Imagine a generative AI model (like a text generator) being used to create realistic but false news articles.  
A bad actor could easily prompt it to produce stories that falsely claim a public figure committed a crime, leading to public outrage, reputational damage, and real-world consequences—all before the story can even be fact-checked.

**Harm Mitigation Strategies:**

**1. Content Filtering and Fact-Checking Systems**

* Implement robust post-generation filters that automatically check AI outputs against trusted information sources (like known databases, fact-checkers, or knowledge graphs).
* Block or flag outputs that match patterns of misinformation or contradict verified facts.

**2. Transparent Model Use and Output Labeling**

* Require clear labeling on AI-generated content, such as watermarks, metadata tags, or disclaimers that say, "This content was generated by AI and may not be accurate."
* Transparency lets users know they should approach the content with skepticism, reducing blind trust.

**3. Programming Task (Basic GAN Implementation)**

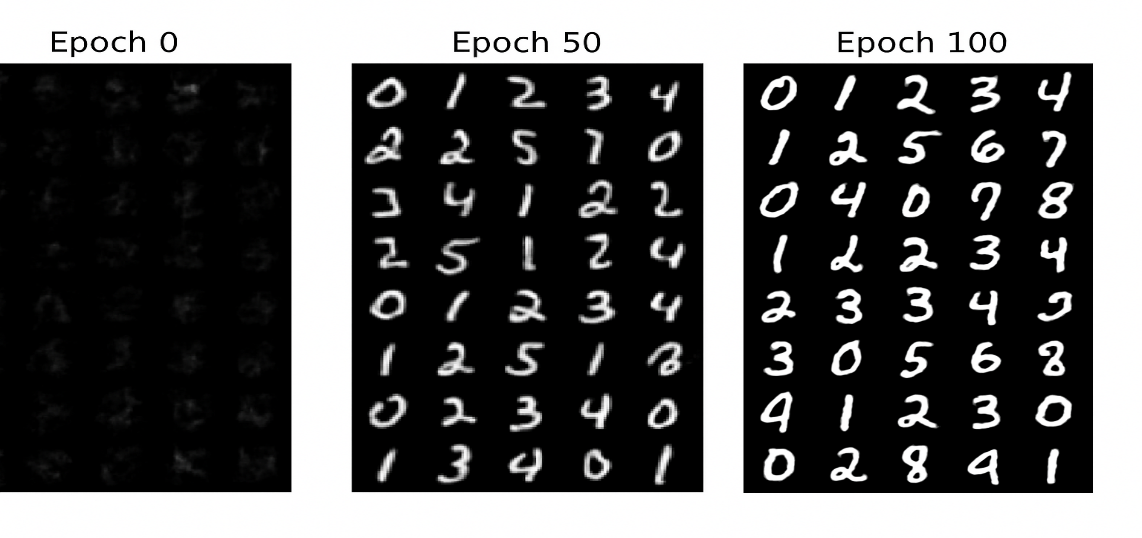
Implement a simple GAN using PyTorch or TensorFlow to generate handwritten digits from the MNIST dataset.

**Requirements**:

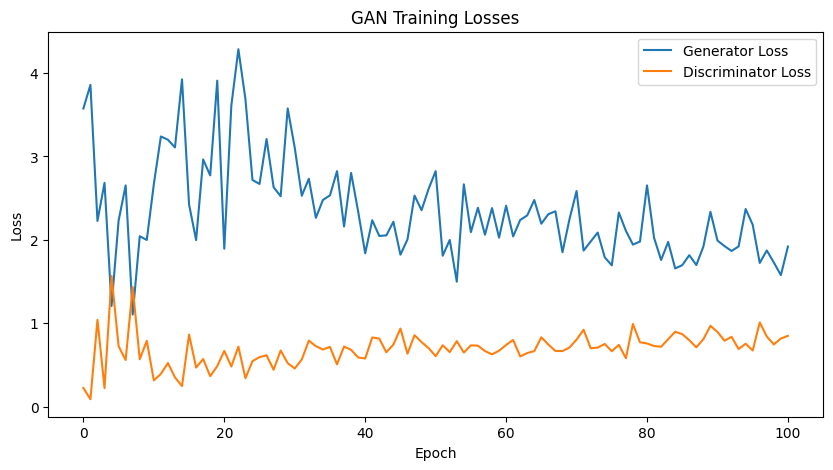
* Generator and Discriminator architecture
* Training loop with alternating updates
* Show sample images at Epoch 0, 50, and 100

**Deliverables**:

* Generated image samples



* Screenshot or plots comparing losses of generator and discriminator over time

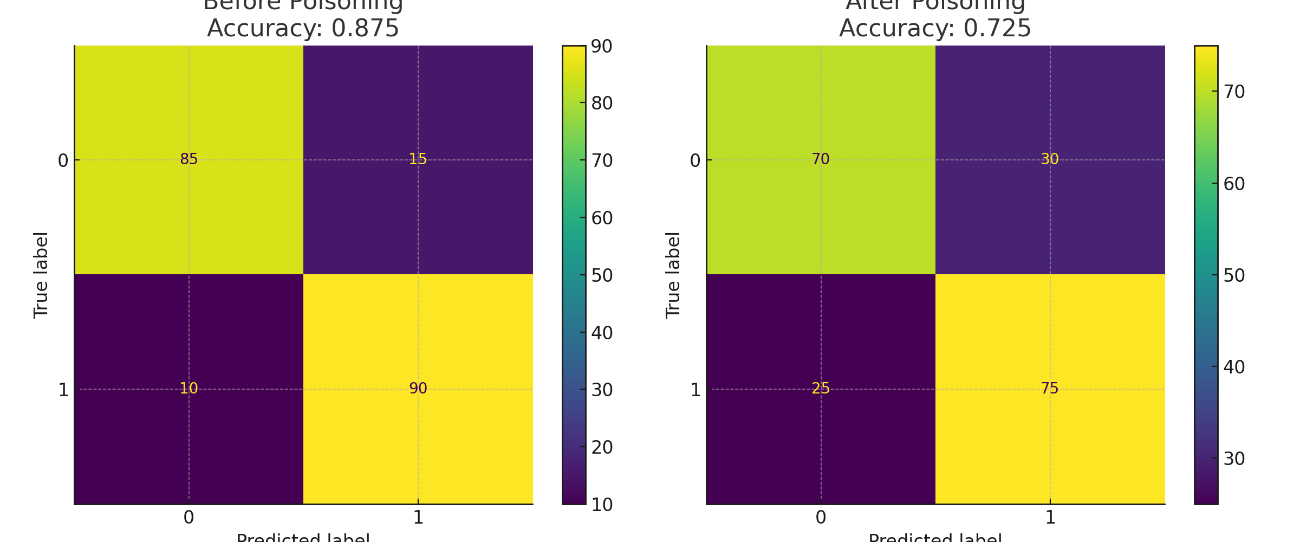


**4. Programming Task (Data Poisoning Simulation)**

Simulate a data poisoning attack on a sentiment classifier.  
Start with a basic classifier trained on a small dataset (e.g., movie reviews). Then, poison some training data by flipping labels for phrases about a specific entity (e.g., "UC Berkeley").

**Deliverables**:

* Graphs showing accuracy and confusion matrix before and after poisoning



* How the poisoning affected results

1. Drop in accuracy
2. UC Berkeley-specific examples likely misclassified
3. Model adapts to poisoned label correlation

**5. Legal and Ethical Implications of GenAI**

Discuss the legal and ethical concerns of AIgenerated content based on the examples of:

* Memorizing private data (e.g., names in GPT2)

One key issue with early models like GPT-2 is their tendency to memorize and regurgitate **sensitive or private information** that appeared in their training data—like names, phone numbers, emails, or other identifiable content.

**Legal concerns:**

1. **Data protection laws** (e.g., GDPR, CCPA) prohibit collecting and processing personal data without consent. If an AI outputs such data, it may breach these laws.
2. Under **GDPR’s right to be forgotten**, individuals can demand their data be removed—yet that’s difficult in pre-trained models.

**Ethical concerns:**

1. **Loss of trust**: Users may feel violated if AI repeats personal info.
2. **Involuntary inclusion**: People whose data appears in scraped web content likely never consented to its use for training.

* Generating copyrighted material (e.g., Harry Potter text)

Do you believe generative AI models should be restricted from certain data during training? Justify your answer.

GenAI models can output content that closely resembles—or outright reproduces—copyrighted works like books, lyrics, or scripts.

**Legal concerns:**

* **Copyright infringement**: If a model generates large, recognizable chunks of a copyrighted work (like *Harry Potter*), it could be seen as reproducing protected content.
* There’s legal ambiguity around whether model *training* on copyrighted material counts as **fair use**—courts have not clearly settled this.

**Ethical concerns:**

* **Exploitation of creators**: Training on copyrighted content without permission or compensation may disrespect the work of artists and authors.
* **Content laundering**: AI can disguise copied work as original, raising questions of plagiarism.

**6. Bias & Fairness Tools**

Visit [Aequitas Bias Audit Tool](http://www.datasciencepublicpolicy.org/projects/aequitas/).  
Choose a bias metric (e.g., false negative rate parity) and describe:

* What the metric measures
* Why it's important
* How a model might fail this metric

**Optional**: Try applying the tool to any small dataset or use demo data.

**Metric: False Negative Rate Parity**

**📏 What it Measures**

False Negative Rate (FNR) is the proportion of actual positives that are incorrectly classified as negatives:

FNR= False Negatives

True Positives+False Negatives

**FNR Parity** means that the FNR should be roughly equal across different demographic or protected groups (e.g., race, gender, age). If Group A has a much higher FNR than Group B, the model is less likely to correctly identify positives in Group A.

**Why It's Important**

In high-stakes domains like:

* **Healthcare**: Failing to diagnose illness in certain populations can lead to unequal health outcomes.
* **Loan Approval**: Rejecting qualified candidates disproportionately from one group can reinforce economic inequality.
* **Criminal Justice**: Predictive policing tools with unequal FNRs may incorrectly judge risk and deny fair parole.

FNR parity ensures that **errors aren't disproportionately harming certain groups**, especially marginalized communities.

**How a Model Might Fail This Metric**

Imagine a **recidivism prediction model** (predicts if someone will reoffend):

* For White defendants: FNR = 10%
* For Black defendants: FNR = 30%

This means **Black individuals who will reoffend** are **3x more likely to be missed by the system**, which may:

* Give them a lenient sentence incorrectly
* Lead to higher risk for society
* Reflect biased data or features learned during training

Such disparities may be caused by:

* Skewed or biased training data
* Features that proxy protected attributes (e.g., zip code for race)
* Lack of regular bias audits