Introduction to Generative AI

UNIT - I

G. Dileep Kumar

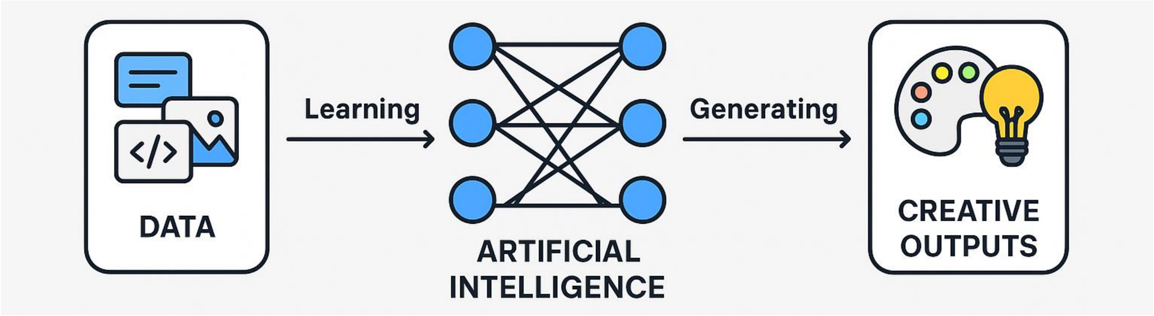
Contents

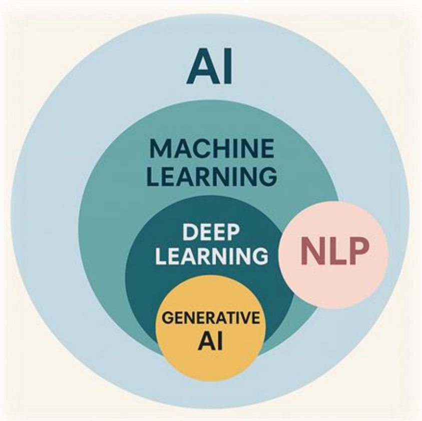
Learning Objectives

- By the end of this unit, you will be able to:
  - Understand the basics and working of Generative AI.
  - Explore key models like GANs, VAEs, and Diffusion models.
  - Learn real-world applications and ethical considerations.
  - Grasp basic probability distributions and likelihood functions.
  - Understand sampling methods used in generative models.

What is Generative AI?

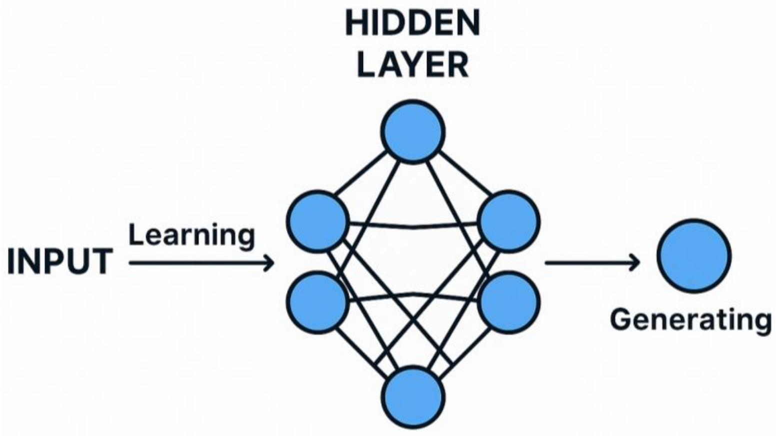
- Definition: AI systems that create new content like text, images, music, and code.
- Comparison with Traditional AI:
    - Traditional AI → Analyzes and classifies data.
    - Generative AI → Creates new data based on learned patterns.





How Generative AI Works

- Deep Learning & Neural Networks:
  - Uses vast datasets to understand and generate new patterns.
  - Probability-based predictions ensure relevant outputs.
- Example Technologies: Transformers (GPT, BERT), Diffusion Models (Stable Diffusion).



Key Generative AI Models

- Text Generation: GPT (Chatbots, content creation).
- Image Generation: DALL·E, Stable Diffusion (AI-generated artwork).
- Music Generation: MusicGen, Jukebox (AI-composed melodies).
- Code Generation: Codex, AlphaCode (AI-assisted programming).

Key Generative AI Models

Model Name	Application	Developer	Use Case
GPT-4 (OpenAI)	Text Generation	OpenAI	Chatbots, content creation
LLaMA 2	Conversational AI	Meta	Natural language processing
Bard (Gemini)	Text & Reasoning AI	Google DeepMind	Search-enhanced AI responses
Stable Diffusionki	Image Generation	Stability AI	AI-generated art & design
DALL·E 2	Image Generation	OpenAI	Text-to-image creation
MusicGen	Music Composition	Meta	AI-generated music & melodies
Jukebox	AI-powered Music	OpenAI	AI-generated songs
Codex	Code Generation	OpenAI	AI-assisted programming
Falcon	Multimodal AI	TII (Technology Innovation Institute)	Language, reasoning, coding
Runway Gen-2	Video Generation	Runway ML	AI-generated short videos

Text Generation Models

1. GPT (Generative Pre-trained Transformer) – OpenAI

- **Architecture**: Transformer-based language model trained on vast text datasets.
- **Variants**:
  - GPT-3.5: Widely used for text-based AI applications.
  - GPT-4: More advanced reasoning and contextual understanding.
- **Capabilities**:
  - Answering questions, summarization, text-based reasoning.
  - Conversational AI for chatbots and assistants.
- **Use Cases**:
  - Chatbots (e.g., ChatGPT).
  - Content writing and coding assistance.

Text Generation Models

2. LLaMA 2 – Meta

- **Architecture**: Optimized transformer model designed for efficiency.
- **Strengths**:
  - Open-source model, freely available for research.
  - Lower computational costs compared to GPT models.
- **Use Cases**:
  - Fine-tuned for dialogue and reasoning.
  - AI-powered assistance tools.

Text Generation Models

3. Bard (Gemini) – Google DeepMind

- **Architecture**: Uses Gemini models with real-time retrieval capabilities.
- **Unique Features**:
  - Strong integration with Google Search.
  - Multimodal abilities (processing text + images + audio).
- **Use Cases**:
  - Scientific research assistance.
  - Search-enhanced chatbot experiences.

Image Generation Models

1. Stable Diffusion – Stability AI

- **Architecture**: Diffusion-based model that converts noise into detailed images.
- **Strengths**:
  - Open-source, customizable, and fine-tunable.
  - Generates high-quality images from text prompts.
- **Use Cases**:
  - AI-generated illustrations and concept art.
  - Realistic photo synthesis.

Image Generation Models

2. DALL·E 2 – OpenAI

- **Architecture**: Transformer-based model trained on vision-language datasets.
- **Capabilities**:
  - Creates images with detailed prompts.
  - Advanced artistic creativity.
- **Use Cases**:
  - Advertising, visual storytelling.
  - Generating design prototypes.

Music Generation Models

1. MusicGen – Meta

- **Architecture**: AI model trained on labeled music data.
- **Capabilities**:
  - Generates melodies based on text descriptions.
  - Supports different musical styles.
- **Use Cases**:
  - Composing background scores for videos.
  - Assisting musicians in creative songwriting.

Music Generation Models

2. Jukebox – OpenAI

- **Architecture**: Transformer-based model designed for raw audio processing.
- **Capabilities**:
  - AI-assisted music generation with vocals.
  - Style transfer for different genres.
- **Use Cases**:
  - Automated remixing.
  - AI-powered music composition.

Code Generation Models

1. Codex – OpenAI

- **Architecture**: Transformer-based code assistant built on GPT.
- **Strengths**:
  - Can write and debug code in multiple languages.
  - Integrates with IDEs for real-time assistance.
- **Use Cases**:
  - AI-powered programming assistants (e.g., GitHub Copilot).
  - Generating functions and scripts based on prompts.

Code Generation Models

2. AlphaCode – DeepMind

- **Architecture**: Reinforcement learning-driven AI for competitive coding.
- **Capabilities**:
  - Generates optimal solutions for complex programming challenges.
  - Learns best coding practices.
- **Use Cases**:
  - Assisting developers in solving algorithmic problems.
  - AI-driven software engineering research.

Multimodal AI Models

1. Falcon – Technology Innovation Institute (TII)

- **Architecture**: Transformer-based model built for high-performance NLP tasks.
- **Capabilities**:
  - Handles text, images, reasoning tasks.
- **Use Cases**:
  - Scientific text processing.
  - AI-powered content creation.

Multimodal AI Models

2. Runway Gen-2 – Runway ML

- **Architecture**: Diffusion-based AI model for video synthesis.
- **Strengths**:
  - Creates high-quality AI-generated short videos.
  - Supports animation-style effects.
- **Use Cases**:
  - AI-powered filmmaking.
  - Video storytelling with text prompts.

Applications of Generative AI

1. Education

- **AI-generated lessons** personalize content for students, adapting to their learning pace.
- **Research assistance** automates literature reviews, helping scholars find relevant papers efficiently.
- **AI tutors** provide instant feedback and guidance, enhancing **interactive learning experiences**.

2. Healthcare

- **AI-assisted drug discovery** accelerates the identification of effective compounds by analyzing vast medical datasets.
- **Diagnostics** utilize AI-powered models to detect diseases like cancer and neurological disorders **with high accuracy**.
- **Predictive analytics** forecast potential **health risks**, enabling early intervention.

Applications of Generative AI

3. Entertainment

- **AI-generated movies & music** allow creators to explore **unique storytelling techniques** and sound compositions.
- **Virtual actors & deepfake technology** enable realistic character animations in films and gaming.
- **AI-powered content personalization** suggests music, movies, and books **based on user preferences**.

4. Programming

- **AI-assisted coding** automates repetitive programming tasks, helping developers **write efficient code faster**.
- **Debugging tools** powered by AI detect and resolve errors **before deployment**, reducing software failures.
- **Code generation & completion** suggest optimized solutions, enabling developers to focus **on complex problem-solving**.

Ethical Considerations

- **Bias & Fairness:** AI models may inherit biases from training data, leading to unfair outcomes.
- **Misinformation & Fake Content:** AI-generated deepfakes and fake news can deceive users and spread false narratives.
- **Copyright & Intellectual Property:** Generative AI challenges ownership rights, raising concerns about originality and legal protections.
- **Privacy & Data Security:** AI may expose personal data, necessitating responsible data handling and compliance.
- **Automation & Job Displacement:** AI-driven automation can replace human jobs, impacting employment in creative industries.
- **Ethical AI Development & Governance:** Lack of regulation may lead to AI misuse, requiring transparent policies and oversight.

1. Bias & Fairness

- Generative AI models learn from large datasets, which may contain societal biases.
- **Issue:** AI-generated content can **reinforce stereotypes**, discriminating against specific groups.
- **Example:** A hiring AI might favour certain demographic groups unintentionally.
- **Mitigation Strategies:**
  - Train models on **diverse datasets** to reduce bias.
  - Implement **fairness-aware algorithms**.
  - Conduct **regular audits** for bias detection.

2. Misinformation & Fake Content

- Generative AI can create highly convincing **deepfakes** and false information.
- **Issue:** AI-generated fake news, altered videos, and misleading texts can deceive the public.
- **Example:** Political deepfakes can be used to spread false narratives.
- **Mitigation Strategies:**
  - Develop **AI watermarking** to detect fake content.
  - Implement **fact-checking mechanisms**.
  - Encourage **responsible AI usage** with clear disclaimers.

3. Copyright & Intellectual Property

- Who owns AI-generated content?
  - This is a growing concern in legal and creative fields.
- **Issue:** AI can create art, music, and text that resemble human work, raising **copyright disputes**.
- **Example:** Artists may find AI-generated paintings **replicating their style**.
- **Mitigation Strategies:**
  - Establish **clear ownership guidelines** for AI-generated content.
  - Introduce **royalty-sharing models** where AI-generated works compensate original creators.
  - Advocate for **legal frameworks** protecting artists and developers.

4. Privacy & Data Security

- AI models require vast amounts of training data, sometimes including **private user information**.
- **Issue:** AI models trained on social media or medical records may unintentionally reveal **sensitive personal data**.
- **Example:** Chatbots trained on user conversations could leak **confidential details**.
- **Mitigation Strategies:**
  - Use **privacy-preserving techniques**, such as differential privacy.
  - Ensure datasets comply with **GDPR and other data regulations**.
  - Encourage **transparent data usage policies**.

5. Automation & Job Displacement

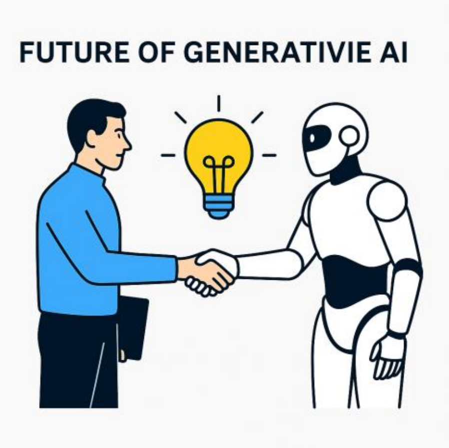
- Generative AI streamlines content creation but could replace **human creative roles**.
- **Issue:** AI automation may reduce job opportunities for writers, artists, and musicians.
- **Example:** AI-generated writing tools replacing **journalists** in news media.
- **Mitigation Strategies:**
    - Promote **human-AI collaboration** instead of full automation.
    - Provide **upskilling programs** for workers in AI-enhanced fields.
    - Create policies ensuring AI **augments** jobs rather than eliminates them.

6. Ethical AI Development & Governance

- The **lack of regulation** in AI development raises accountability concerns.
- **Issue:** Companies prioritize profits over ethical AI deployment.
- **Example:** Unregulated AI models spreading **harmful content** without oversight.
- **Mitigation Strategies:**
  - Encourage **transparent AI model disclosures**.
  - Establish **ethical AI review boards**.
  - Advocate for **government AI regulations** that balance innovation with safety.

Future of Generative AI

- Next-Gen AI Models: More creative, efficient, and personalized.
- Human-AI Collaboration: AI as a creative assistant rather than replacement.
- New Applications: AI-powered personalized learning and automation.



Future of Generative AI

Aspect	Pros (Benefits)	Cons (Challenges)
Creativity & Innovation	Enables AI-generated art, music, writing, and code	Raises concerns about originality and AI replacing human creativity
Automation & Efficiency	Speeds up content generation, reducing workload	Can lead to over-reliance on AI, replacing human effort
Accessibility	Democratizes creative tools, making AI-powered content available to all	Might widen the digital divide if access to AI tools is limited
Personalization	AI can tailor content to individual preferences	Can lead to privacy concerns with excessive data usage
Bias & Fairness	AI can provide diverse perspectives based on training data	AI inherits biases from dataset, leading to unfair outputs
Misinformation	Helps summarize complex topics quickly	Can generate inaccurate or misleading content
Ownership & Copyright	AI can assist in creating unique intellectual property	Raises legal concerns over content ownership and plagiarism
Job Displacement	Helps automate repetitive tasks, freeing up time for higher-level work	May replace creative jobs, affecting employment opportunities
Ethical AI Development	Encourages responsible AI use and policy discussions	Companies may prioritize profit over ethical AI practices

Generative vs. Discriminative Models

- Generative Models:
    - These try to model **how data is generated**, i.e., they learn the joint distribution $P(x, y)$, and can generate synthetic examples. From this, you can compute $P(y|x)$ using Bayes' theorem.
    - Example: "Given the features of a handwritten digit, what is the likelihood it was written as a 3?"
- Discriminative Models:
    - These focus on **modeling the decision boundary**, i.e., they learn $P(y|x)$ directly. They are optimized for classification or regression tasks.
    - Example: "Given this image of a digit, is it a 3 or a 5?"

Generative vs. Discriminative Models

Model Type	**Algorithms**	Use Case
Generative	Naive Bayes, Hidden Markov Models (HMM), GANs, VAEs	Text generation, image synthesis, speech synthesis
Discriminative	Logistic Regression, SVM, Neural Networks, Random Forest	Spam detection, sentiment analysis, object detection

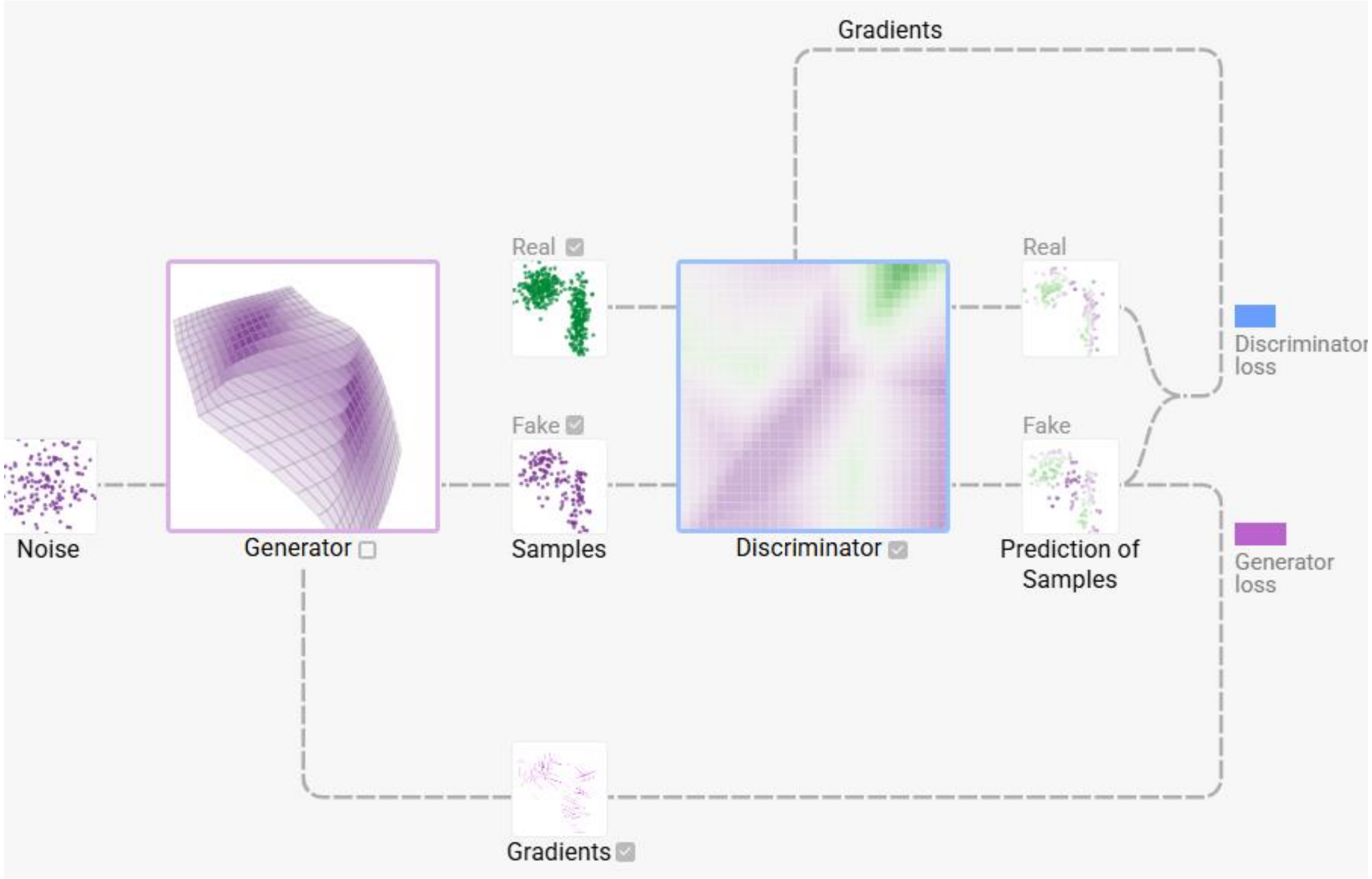
Core Models in Generative AI

Model	Purpose	Example
GAN (Generative Adversarial Network)	Image generation	MRI image super-resolution
VAE (Variational Autoencoder)	Data generation with latent space control	Patient record synthesis
Transformer-based LLMs	Text generation	Generating clinical summaries

GAN – Simplified Workflow

- Generative Adversarial Networks (GANs) are a class of deep learning models designed to generate realistic synthetic data. They consist of two neural networks:
  - Generator: Creates new data samples (e.g., images) from random noise.
  - Discriminator: Evaluates whether the generated samples are real or fake.
- These two networks compete, improving each other over time.
- GANs are widely used in image synthesis, style transfer, and even medical imaging.
- You can explore GAN Lab, an interactive tool that helps visualize how GANs work.

GAN – Simplified Workflow



VAE – Conceptual Overview

- A **VAE** is a type of generative model that:
  - Learns a **compressed (latent)** representation of data
  - Can **generate new data** by sampling from the latent space
- **Key Components of VAE**
  - **Encoder:** Compresses input data into a latent space representation, producing a mean and variance instead of a fixed encoding.
  - **Latent Space:** Adds controlled randomness, ensuring smooth interpolation between data points.
  - **Decoder:** Reconstructs or generates new data from the latent space representation.
- **Applications:**
  - Disease progression simulation
  - Patient clustering visualization

Transformer Model

- **Transformers** are a type of deep learning model introduced in the paper **"Attention is All You Need"** by Vaswani et al. in 2017.
- They revolutionized natural language processing (NLP) and have since become the foundation of state-of-the-art models like **GPT**, **BERT**, and **T5**.
- Key Components:
  1. **Tokenization:** Splits text into smaller units called tokens.
  2. **Embedding Layer:** Converts tokens into vectors (numerical representations).
  3. **Positional Encoding:** Adds information about the **position** of each token.
     - Tokens: ["I", "love", "you"]
     - Position IDs: [0 , 1 , 2]

Transformer Model

- Key Components:

4. **Self-Attention Mechanism:** Allows each word to focus on other words in the sentence, assigning **attention scores**.
   - For the sentence: "The dog chased the ball."
   - While processing "chased", self-attention might pay more attention to "dog" (subject) and "ball" (object).

```
Word:      the   dog   chased   the   ball
Attention: 0.1   0.4    0.1     0.1   0.3  ← for the word "chased"
```

Transformer Model

- Key Components:
    6. **Multi-Head Attention:** Runs multiple attention layers (heads) in parallel to capture different aspects of the sentence.
        - **Example**:
            - One head might focus on subject-verb relationships.
            - Another might track object-pronoun links.
        - This makes the model more flexible and expressive.
    7. **Feed-Forward Neural Network (FFN):** After attention, each token passes through a small MLP (Multi-Layer Perceptron) independently.
    8. **Residual Connections & Layer Normalization:**
        - **Residual connections**: Add the input back to the output to prevent loss of information.
        - **Layer normalization**: Stabilizes learning and improves performance.

Transformer Model

- Key Components:
    8. **Encoder & Decoder Stack**
        - **Encoder**: Processes the input sentence. Used in models like BERT.
        - **Decoder**: Generates output tokens step-by-step. Used in models like GPT and in translation models.

Basics of Probability Distribution

- A **probability distribution** describes how the probabilities are distributed over the possible outcomes of a random experiment or random variable.
- A **random variable** (RV) is a variable whose values depend on the outcomes of a random phenomenon.
- **Types of random variables:**
  - **Discrete random variable:** Takes countable values (e.g., number of heads in coin tosses).
  - **Continuous random variable:** Takes any value in an interval or range (e.g., height, temperature).

Probability Mass Function (PMF) – For Discrete RVs

- Defines the probability that a discrete random variable is **exactly** equal to some value.
- Example: Fair six-sided die $$P(X = x) = \frac{1}{6}, \quad x \in \{1, 2, 3, 4, 5, 6\}$$

Properties:

1. $0 \leq P(X = x) \leq 1$
2. $\sum_x P(X = x) = 1$

Probability Density Function (PDF) – For Continuous RVs

- Describes the **relative likelihood** of a continuous RV taking a value within a range.
- Probability of a single point is **zero** (since continuous RVs have infinite possibilities).
- Instead, we compute probabilities over intervals using integration:

$$P(a \leq X \leq b) = \int_a^b f(x)\, dx$$

where $f(x)$ is the PDF.

Properties:

1. $f(x) \geq 0$
2. $\int_{-\infty}^{\infty} f(x)\, dx = 1$

Cumulative Distribution Function (CDF)

- Gives the probability that a random variable *XX* is **less than or equal to** a value *xx*.
  - For **discrete** RVs:

$$F(x) = P(X \leq x) = \sum_{k \leq x} P(X = k)$$

  - For **continuous** RVs:

$$F(x) = P(X \leq x) = \int_{-\infty}^{x} f(t)\, dt$$

Properties:

1. $0 \leq F(x) \leq 1$
2. $F(x)$ is non-decreasing.
3. $\lim_{x \to -\infty} F(x) = 0$ and $\lim_{x \to \infty} F(x) = 1$

Common Probability Distributions

(A) Discrete Distributions

Distribution	PMF	Example Use Case
Bernoulli	$P(X=1)=p, P(X=0)=1-p$	Coin flip
Binomial	$P(X=k)=\binom{n}{k}p^k(1-p)^{n-k}$	Number of heads in n tosses
Poisson	$P(X=k)=\frac{\lambda^k e^{-\lambda}}{k!}$	Number of calls at a call center per hour

(B) Continuous Distributions

Distribution	PDF	Example Use Case
Uniform	$f(x)=\frac{1}{b-a}$ for $a \leq x \leq b$	Random number generation
Normal (Gaussian)	$f(x)=\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$	Heights, IQ scores
Exponential	$f(x)=\lambda e^{-\lambda x}$	Time between events (e.g., earthquakes)

Likelihood Functions

- While **probability** describes how likely data is **given a model**, **likelihood** describes how likely a **model (parameters)** is **given the data**.
- **Example: Bernoulli Likelihood**

Suppose you flip a coin 10 times and get 7 heads.

Let θ be the probability of heads.

$$L(\theta) = \theta^7 \cdot (1 - \theta)^3$$

To find the most likely value of θ, we can maximize this likelihood (MLE: Maximum Likelihood Estimation).

Sampling in Generative AI

- In Generative AI, **sampling** means:
  - “Choosing the next piece of output (word, image pixel, sound, etc.) based on probabilities learned by the model.”
  - The model doesn’t give you one exact answer — it gives **probabilities** of possible answers.
- **Sampling** decides **what to actually pick** from those possibilities.
- Example: Autocomplete
  - Imagine typing on your phone: "I'm feeling"
  - Now your phone (like a generative model) might suggest:

Word	Probability
happy	0.4
tired	0.3
hungry	0.2
lucky	0.1

Sampling decides: "Should we always pick 'happy' (highest), or sometimes pick 'tired' or 'hungry'?"

Sampling Methods

1. **Greedy Sampling:** Pick the most likely word every time.
   - Input: "I'm feeling"
   - Probabilities: happy (0.4), tired (0.3), hungry (0.2), lucky (0.1)
   - Output: **"happy"**
   - Very predictable, but can become repetitive in long texts.
2. **Random Sampling:** Pick a word randomly according to their probabilities.
   - Might get "tired" or "hungry" sometimes.
   - Good for creativity, but can be messy.
3. **Top-k Sampling (e.g., k = 2):** Only keep the top k words, and randomly pick from them.
   - Top 2: happy (0.4), tired (0.3)
   - Randomly choose between these two
   - **more diverse but still relevant**

Sampling Methods

4. **Top-p (Nucleus) Sampling (e.g., p = 0.8)**
    - **Pick smallest set of words where total probability ≥ p.**
    - happy (0.4) + tired (0.3) + hungry (0.2) → total = 0.9 → keep those
    - Randomly sample from these 3
    - **More dynamic and context-aware**
5. **Temperature**
    - **Control how "random" the sampling is.**
    - **Low temperature (0.5)** = safe and predictable
    - **High temperature (1.5)** = creative and wild
    - Think of temperature as how adventurous the AI is.

Analogy: Ice Cream Flavors

- You're choosing 1 scoop.
  The AI tells you how much it "likes" each flavor:

Flavor	Score
Vanilla	0.5
Chocolate	0.3
Mango	0.2

- **Greedy:** Always choose Vanilla.
- **Random:** Maybe get Mango!
- **Top-k (k=2):** Pick between Vanilla and Chocolate.
- **Temperature:** Turn the "risk dial" up or down.

Where Sampling Happens

Model	What It Samples
GPT / ChatGPT	Next word or token
DALL·E / Stable Diffusion	Pixel patterns (via noise)
VAEs / GANs	Latent vector (z), which is decoded into data

Sampling Type	Creativity	Risk of Irrelevance	Output Style
Greedy	Low	Low	Repetitive but safe
Top-k	Medium	Low–Medium	Balanced
Top-p	Medium–High	Low–Medium	Natural & flexible
Temperature	High (if >1)	High	Creative or chaotic

Hands-on Examples

1. Using Autoencoder (AE) – Image Compression & Reconstruction
2. Implement a Transformer-based Language Model for text generation

Summary & Key Takeaways

- Generative AI creates new content across multiple domains.
- It leverages deep learning models for intelligent output.
- Future AI will be more creative, personalized, and collaborative.

Review Questions

1. Compare and contrast generative and discriminative models. Give an example of each.
2. List three popular generative models and describe the main idea behind each.
3. Explain the role of probability distributions and likelihood functions in generative modeling.
4. What is sampling in the context of generative models? Describe one method and how it affects the output quality.
5. In what way does a Transformer model contribute to generative AI? What makes it suitable for large-scale language generation tasks?