General AI and ML Basics

1. What is the difference between Artificial Intelligence, Machine Learning, and Deep Learning?
2. Explain the difference between supervised, unsupervised, and reinforcement learning.
3. What is overfitting and underfitting in machine learning? How can you prevent them?
4. Explain the bias-variance tradeoff in machine learning.
5. What are some common metrics used to evaluate the performance of a classification model? Explain their use cases.
6. What is the curse of dimensionality, and how can it be mitigated?
7. How do you handle missing data in a dataset?
8. Explain feature scaling and why it is important in machine learning.

Deep Learning and Neural Networks

1. What is a neural network, and how does it work?
2. Explain the difference between convolutional neural networks (CNNs) and recurrent neural networks (RNNs).
3. What is backpropagation, and how does it work in training a neural network?
4. What is the vanishing gradient problem, and how is it solved?
5. Explain transfer learning and its advantages.
6. What is the role of activation functions in a neural network? Compare ReLU, Sigmoid, and Tanh.
7. How does dropout work in a neural network, and why is it used?

Generative AI (Gen AI)

1. What are Generative Adversarial Networks (GANs), and how do they work?
2. How does a Transformer model work? Why are transformers important in Gen AI?
3. Explain the difference between BERT and GPT models.
4. What is attention mechanism, and how is it used in models like GPT and BERT?
5. How does fine-tuning a foundation model differ from training it from scratch?
6. Explain the concept of reinforcement learning with human feedback (RLHF).

7. What are some ethical considerations in deploying Generative AI models?
8. Describe some common use cases of Generative AI in industries today.
9. How can Generative AI models be evaluated for quality and accuracy?
10. What is the role of multimodal models in Generative AI? Provide examples.

Natural Language Processing (NLP)

1. What are word embeddings, and how do they differ from one-hot encoding?
2. Explain how Word2Vec or GloVe works.
3. What is a sequence-to-sequence model, and where is it used?
4. How does sentiment analysis work? Provide an example pipeline.
5. What is the difference between stemming and lemmatization?
6. Explain the use of positional encoding in Transformer models.
7. How do you deal with out-of-vocabulary (OOV) words in NLP tasks?

Real-World Applications

1. How would you build a recommendation system for an e-commerce platform?
2. How would you approach building a real-time fraud detection system?
3. Design a pipeline for training a computer vision model to classify objects in images.
4. How can you use Generative AI for summarizing documents?
5. Discuss a machine learning project you worked on. What were the challenges, and how did you solve them?

Optimization and Model Deployment

1. What is gradient descent, and what are its variants (e.g., SGD, Adam)?
2. What are some techniques to optimize the training process for deep learning models?
3. How would you deploy a machine learning model in production?
4. What are some challenges in scaling machine learning systems?
5. Explain A/B testing and its importance in ML model deployment.

Advanced Topics

1. What are some differences between explainable AI (XAI) and black-box models?
2. What is model drift, and how do you monitor it in production?
3. Explain the concept of multi-agent systems in AI and their real-world applications.
4. What is Federated Learning, and how is it different from centralized learning?
5. How do you implement machine unlearning in models for compliance with GDPR or CCPA?

Behavioral Questions

1. Describe a time when your machine learning model didn't perform as expected. How did you address it?
2. How do you keep yourself updated with the latest developments in AI/ML?
3. Can you explain a complex AI/ML concept to a non-technical audience? (Prepare to explain one on the spot.)
4. How do you handle ethical dilemmas when deploying AI solutions?

Technical Code Challenges

1. Write Python code to implement linear regression from scratch.
2. Implement a simple feedforward neural network in PyTorch or TensorFlow.
3. Write code to preprocess a text dataset for training an NLP model.
4. Given a dataset, create a machine learning pipeline to predict a target variable.
5. Implement a simple Transformer encoder from scratch.

Tips for Preparation

- Focus on understanding the underlying math behind ML and DL algorithms.
- Keep up-to-date with advancements in Generative AI models like GPT, DALL-E, and multimodal systems.
- Review common Python libraries (e.g., NumPy, pandas, TensorFlow, PyTorch, Hugging Face).
- Practice explaining technical concepts clearly for behavioral and scenario-based questions.