General Machine Learning Concepts

1. What is the difference between supervised, unsupervised, and reinforcement learning? Provide examples.
2. How would you explain overfitting and underfitting to a non-technical audience?
3. What is the purpose of a train/test split, and how do you ensure it is representative?
4. Describe the difference between feature scaling and normalization. When are they required?
5. How do you decide when to use a linear vs. non-linear model?
6. What are the main assumptions of linear regression?
7. How does logistic regression work, and why is it considered a classification algorithm?
8. Explain the purpose of activation functions in neural networks. Compare ReLU, Sigmoid, and Tanh.
9. What is an embedding, and where would you use it in a machine learning problem?
10. What is a softmax function, and how is it used in multi-class classification?
11. Why is cross-validation important in machine learning? What types of cross-validation do you know?
12. Explain the difference between model-based and memory-based learning.
13. How does k-nearest neighbors (k-NN) work, and when would you use it?
14. What is the purpose of the learning rate in optimization algorithms?
15. What is gradient descent, and how does stochastic gradient descent differ?
16. Explain the difference between offline and online learning. When would you use each approach?
17. How does feature engineering impact model performance? Can you give an example?
18. What are categorical embeddings, and why are they useful?
19. Describe the purpose of early stopping in neural network training.
20. What are some common pitfalls in data preprocessing, and how do you avoid them?

Mathematics and Theoretical Questions

1. What is the difference between generative and discriminative models?
2. Explain the central limit theorem and its significance in machine learning.
3. How does principal component analysis (PCA) reduce dimensionality? Provide the underlying math.
4. What is the difference between convex and non-convex optimization problems?
5. Explain Lagrange multipliers and their role in constrained optimization.
6. What is the role of eigenvalues and eigenvectors in PCA?
7. How would you explain Bayes' theorem to someone without a math back-

ground?
8. What are Markov chains, and where are they used in machine learning?
9. What is KL divergence, and why is it important in machine learning models like VAEs?
10. Explain the concept of entropy in information theory.
11. What are the advantages of using stochastic gradient descent over batch gradient descent?
12. What is the difference between a loss function and a cost function?
13. Can you explain the math behind support vector machines (SVMs)?
14. What is the curse of dimensionality, and how can it be mitigated?
15. How does the Jacobian matrix help in backpropagation?
16. Explain how L1 and L2 regularization penalize model complexity differently.
17. What is the difference between a prior and a posterior in Bayesian inference?
18. What is Monte Carlo sampling, and where is it used in machine learning?
19. How does a Gaussian mixture model (GMM) work?
20. What are the mathematical foundations of reinforcement learning (Markov decision processes)?

Algorithms and Models

1. How does a decision tree decide splits at each node?
2. What is information gain, and how is it calculated in decision trees?
3. How do random forests reduce overfitting compared to individual decision trees?
4. Explain how boosting algorithms like AdaBoost or XGBoost work.
5. What is the difference between bagging and boosting?
6. How do convolutional layers in CNNs capture spatial features?
7. What is attention in the context of neural networks, and why is it useful?
8. How does a long short-term memory (LSTM) unit work, and what problems does it solve?
9. What are the key differences between GANs and VAEs?
10. How does transfer learning work, and when would you use it?
11. What are the trade-offs of using pre-trained models in production systems?
12. How does a collaborative filtering recommendation system work?
13. What is matrix factorization, and how does it improve recommendations?
14. Explain the role of dropout in neural networks.
15. What are residual connections, and why are they important in deep learning architectures like ResNet?
16. How do you choose the number of clusters in k-means clustering?
17. What is the elbow method in clustering, and how does it work?
18. What are autoencoders, and where are they typically used?
19. How does a transformer architecture process sequential data?
20. What is the difference between semantic and instance segmentation in computer vision?

Evaluation and Metrics

1. What metrics would you use to evaluate a regression model?
2. Explain precision, recall, and F1-score. How do they relate?
3. What is the ROC curve, and how is AUC used to evaluate models?
4. What is the difference between a confusion matrix and a classification report?
5. How do you assess the effectiveness of a clustering algorithm?
6. What is the difference between offline and online evaluation of models?
7. How do you choose an evaluation metric for imbalanced datasets?
8. What is the purpose of an A/B test, and how would you implement it for a recommendation system?
9. How do you evaluate time-series models?
10. What is uplift modeling, and how is it evaluated?
11. How would you identify if your model is suffering from data leakage?
12. What is a calibration curve, and why is it useful for probabilistic models?
13. How do you monitor the performance of a deployed machine learning model?
14. What is multi-class log loss, and when would you use it?
15. Explain the difference between mean absolute error (MAE) and mean squared error (MSE).
16. What is a baseline model, and why is it important?
17. How do you interpret Shapley values for feature importance?
18. What are counterfactual explanations in interpretability?
19. How do you evaluate ranking models?
20. Explain the purpose of hyperparameter tuning and how you measure its success.

System Design and Real-World Applications

1. How would you design a scalable recommendation system for a video platform?
2. Explain how you would handle the cold start problem in recommendation systems.
3. How do you ensure data privacy in machine learning models?
4. What techniques do you use to make real-time predictions at scale?
5. How would you monitor for model drift in production?
6. How do you decide when to retrain a machine learning model?
7. What are feature stores, and how do they fit into an ML pipeline?
8. How do you design an ML system to operate on edge devices?
9. Explain the trade-offs between batch and online learning in production systems.

10. How would you implement a multi-armed bandit system for dynamic content recommendations?
11. What strategies would you use to improve the latency of a deployed model?
12. What is federated learning, and how is it applied in real-world scenarios?
13. How would you deploy and version machine learning models in a CI/CD pipeline?
14. How do you ensure reproducibility in machine learning experiments?
15. How do you design an architecture for a fraud detection system?
16. What is the role of synthetic data, and how would you generate it?
17. Explain the difference between feature pipelines and data pipelines.
18. How would you implement explainability in a black-box model like a neural network?
19. How do you prioritize features for an MVP of an ML product?
20. What challenges have you encountered in deploying ML systems, and how did you overcome them?