1. Recognize the differences between supervised, semi-supervised, and unsupervised learning.

Supervised learning: Uses labeled data to train models for tasks like classification and regression.

Semi-supervised learning: Combines a small amount of labeled data with a large amount of unlabeled data during training.

Unsupervised learning: Analyzes and clusters unlabeled data to find hidden patterns or intrinsic structures.

2. Describe in detail any five examples of classification problems.

Examples:

1. Spam Detection: Classifying emails as spam or not spam.

2. Sentiment Analysis: Determining if a text is positive, negative, or neutral.

3. Image Recognition: Identifying objects in an image, like cats or dogs.

4. Disease Diagnosis: Predicting if a patient has a specific disease based on symptoms.

5. Credit Scoring: Assessing if a loan applicant is a good or bad credit risk.

3. Describe each phase of the classification process in detail.

1. Data Collection: Gathering relevant data.

2. Data Preprocessing: Cleaning and preparing data for analysis.

3. Feature Selection: Identifying important features.

4. Model Training: Using labeled data to train the model.

5. Model Evaluation: Testing the model’s performance using metrics like accuracy.

6. Deployment: Implementing the model in a real-world environment.

7. Maintenance: Updating the model with new data as needed.

4. Go through the SVM model in depth using various scenarios.

Support Vector Machine (SVM):

- Binary Classification: Separates data into two classes using a hyperplane.

- Linear SVM: Works well when data is linearly separable.

- Non-linear SVM: Uses kernel functions like RBF to handle non-linear data.

- Scenarios: Text classification, image recognition, and bioinformatics.

5. What are some of the benefits and drawbacks of SVM?

Benefits:

- Effective in high-dimensional spaces.

- Uses a subset of training points (support vectors) for the decision function.

- Robust against overfitting in high-dimensional spaces.

Drawbacks:

- Computationally intensive for large datasets.

- Less effective with noisy data.

- Choice of kernel function can significantly impact performance.

6. Go over the kNN model in depth.

k-Nearest Neighbors (kNN):

- Instance-based learning: Stores all instances during training.

- Classification: Assigns class based on the majority class among the k-nearest neighbors.

- Regression: Predicts the average value of k-nearest neighbors.

- Distance Metrics: Commonly uses Euclidean distance.

7. Discuss the kNN algorithm's error rate and validation error.

Error Rate:

- Training Error: Error rate on the training data.

- Validation Error: Error rate on a separate validation set.

- Bias-Variance Tradeoff: Low k value increases variance, high k value increases bias.

8. For kNN, talk about how to measure the difference between the test and training results.

Difference Measurement:

- Performance Metrics: Accuracy, precision, recall, F1-score.

- Cross-validation: Helps estimate the model's performance on unseen data.

- Overfitting and Underfitting: Large differences indicate overfitting, small differences indicate good generalization.

9. Create the kNN algorithm.

1. Initialize: Choose the number of neighbors \( k \).

2. Compute Distance: Calculate distance between test instance and all training instances.

3. Sort: Sort distances in ascending order.

4. Select Neighbors: Pick the first \( k \) neighbors.

5. Predict: Assign the class with the highest frequency among the \( k \) neighbors.

10. What is a decision tree, exactly? What are the various kinds of nodes? Explain all in depth.

Decision Tree: A flowchart-like structure where each internal node represents a feature, each branch represents a decision rule, and each leaf node represents an outcome.

Nodes:

- Root Node: Represents the entire dataset.

- Internal Nodes: Represent a test on an attribute.

- Leaf Nodes: Represent class labels or outcomes.

11. Describe the different ways to scan a decision tree.

Tree Traversal Methods:

- Pre-order Traversal: Visit root, left subtree, right subtree.

- In-order Traversal: Visit left subtree, root, right subtree.

- Post-order Traversal: Visit left subtree, right subtree, root.

- Level-order Traversal: Visit nodes level by level.

12. Describe in depth the decision tree algorithm.

Decision Tree Algorithm:

1. Select Best Attribute: Using measures like Gini index or entropy.

2. Split Dataset: Based on the selected attribute.

3. Create Subtrees: Recursively repeat the process for each subset.

4. Stop Criteria: When all instances in a subset belong to the same class or no more features to split.

13. In a decision tree, what is inductive bias? What would you do to stop overfitting?

Inductive Bias: The set of assumptions a learning algorithm makes to predict outputs given inputs it hasn’t encountered.

Preventing Overfitting:

- Pruning: Removing branches that have little importance.

- Setting Depth Limits: Limiting the depth of the tree.

- Cross-validation: Using to determine the best tree size.

14. Explain advantages and disadvantages of using a decision tree?

Advantages:

- Easy to interpret.

- Handles both numerical and categorical data.

- Requires little data preprocessing.

Disadvantages:

- Prone to overfitting.

- Can be unstable with small changes in data.

- Biased towards dominant classes.

15. Describe in depth the problems that are suitable for decision tree learning.

Suitable Problems:

- Classification: Assigning class labels to instances.

- Regression: Predicting continuous values.

- Multi-output Problems: Handling tasks where multiple outputs are required.

- Feature Selection: Identifying significant features in the data.

16. Describe in depth the random forest model. What distinguishes a random forest?

Random Forest:

- An ensemble of decision trees.

- Combines multiple trees to improve accuracy.

- Uses bagging (bootstrap aggregating) to sample data and build trees.

- Each tree votes on the output.

Distinguishing Features:

- Reduces overfitting by averaging multiple trees.

- Handles large datasets and many features.

- Provides feature importance scores.

17. In a random forest, talk about OOB error and variable value.

OOB Error:

- Out-of-Bag Error: An estimate of the model error using the samples not included in the bootstrap sample.

- Variable Importance: Measures the importance of each feature in making predictions, determined by the decrease in accuracy when the feature is excluded.

Answers to Questions:

1. Recognize the differences between supervised, semi-supervised, and unsupervised learning.

Supervised learning: Uses labeled data to train models for tasks like classification and regression.

Semi-supervised learning: Combines a small amount of labeled data with a large amount of unlabeled data during training.

Unsupervised learning: Analyzes and clusters unlabeled data to find hidden patterns or intrinsic structures.

2. Describe in detail any five examples of classification problems.

Examples:

1. Spam Detection: Classifying emails as spam or not spam.

2. Sentiment Analysis: Determining if a text is positive, negative, or neutral.

3. Image Recognition: Identifying objects in an image, like cats or dogs.

4. Disease Diagnosis: Predicting if a patient has a specific disease based on symptoms.

5. Credit Scoring: Assessing if a loan applicant is a good or bad credit risk.

3. Describe each phase of the classification process in detail.

1. Data Collection: Gathering relevant data.

2. Data Preprocessing: Cleaning and preparing data for analysis.

3. Feature Selection: Identifying important features.

4. Model Training: Using labeled data to train the model.

5. Model Evaluation: Testing the model’s performance using metrics like accuracy.

6. Deployment: Implementing the model in a real-world environment.

7. Maintenance: Updating the model with new data as needed.

4. Go through the SVM model in depth using various scenarios.

Support Vector Machine (SVM):

- Binary Classification: Separates data into two classes using a hyperplane.

- Linear SVM: Works well when data is linearly separable.

- Non-linear SVM: Uses kernel functions like RBF to handle non-linear data.

- Scenarios: Text classification, image recognition, and bioinformatics.

5. What are some of the benefits and drawbacks of SVM?

Benefits:

- Effective in high-dimensional spaces.

- Uses a subset of training points (support vectors) for the decision function.

- Robust against overfitting in high-dimensional spaces.

Drawbacks:

- Computationally intensive for large datasets.

- Less effective with noisy data.

- Choice of

kernel function can significantly impact performance.

6. Go over the kNN model in depth.

k-Nearest Neighbors (kNN):

- Instance-based learning: Stores all instances during training.

- Classification: Assigns class based on the majority class among the k-nearest neighbors.

- Regression: Predicts the average value of k-nearest neighbors.

- Distance Metrics: Commonly uses Euclidean distance.

7. Discuss the kNN algorithm's error rate and validation error.

Error Rate:

- Training Error: Error rate on the training data.

- Validation Error: Error rate on a separate validation set.

- Bias-Variance Tradeoff: Low k value increases variance, high k value increases bias.

8. For kNN, talk about how to measure the difference between the test and training results.

Difference Measurement:

- Performance Metrics: Accuracy, precision, recall, F1-score.

- Cross-validation: Helps estimate the model's performance on unseen data.

- Overfitting and Underfitting: Large differences indicate overfitting, small differences indicate good generalization.

9. Create the kNN algorithm.

1. Initialize: Choose the number of neighbors \( k \).

2. Compute Distance: Calculate distance between test instance and all training instances.

3. Sort: Sort distances in ascending order.

4. Select Neighbors: Pick the first \( k \) neighbors.

5. Predict: Assign the class with the highest frequency among the \( k \) neighbors.

10. What is a decision tree, exactly? What are the various kinds of nodes? Explain all in depth.

Decision Tree: A flowchart-like structure where each internal node represents a feature, each branch represents a decision rule, and each leaf node represents an outcome.

Nodes:

- Root Node: Represents the entire dataset.

- Internal Nodes: Represent a test on an attribute.

- Leaf Nodes: Represent class labels or outcomes.

11. Describe the different ways to scan a decision tree.

Tree Traversal Methods:

- Pre-order Traversal: Visit root, left subtree, right subtree.

- In-order Traversal: Visit left subtree, root, right subtree.

- Post-order Traversal: Visit left subtree, right subtree, root.

- Level-order Traversal: Visit nodes level by level.

12. Describe in depth the decision tree algorithm.

Decision Tree Algorithm:

1. Select Best Attribute: Using measures like Gini index or entropy.

2. Split Dataset: Based on the selected attribute.

3. Create Subtrees: Recursively repeat the process for each subset.

4. Stop Criteria: When all instances in a subset belong to the same class or no more features to split.

13. In a decision tree, what is inductive bias? What would you do to stop overfitting?

Inductive Bias: The set of assumptions a learning algorithm makes to predict outputs given inputs it hasn’t encountered.

Preventing Overfitting:

- Pruning: Removing branches that have little importance.

- Setting Depth Limits: Limiting the depth of the tree.

- Cross-validation: Using to determine the best tree size.

14. Explain advantages and disadvantages of using a decision tree?

Advantages:

- Easy to interpret.

- Handles both numerical and categorical data.

- Requires little data preprocessing.

Disadvantages:

- Prone to overfitting.

- Can be unstable with small changes in data.

- Biased towards dominant classes.

15. Describe in depth the problems that are suitable for decision tree learning.

Suitable Problems:

- Classification: Assigning class labels to instances.

- Regression: Predicting continuous values.

- Multi-output Problems: Handling tasks where multiple outputs are required.

- Feature Selection: Identifying significant features in the data.

16. Describe in depth the random forest model. What distinguishes a random forest?

Random Forest:

- An ensemble of decision trees.

- Combines multiple trees to improve accuracy.

- Uses bagging (bootstrap aggregating) to sample data and build trees.

- Each tree votes on the output.

Distinguishing Features:

- Reduces overfitting by averaging multiple trees.

- Handles large datasets and many features.

- Provides feature importance scores.

17. In a random forest, talk about OOB error and variable value.

OOB Error:

- Out-of-Bag Error: An estimate of the model error using the samples not included in the bootstrap sample.

- Variable Importance: Measures the importance of each feature in making predictions, determined by the decrease in accuracy when the feature is excluded.