1. What is a Decision Tree, and how does it make decisions during test time?

A Decision Tree is a machine learning tool used for tasks like classification and regression. It organizes data into a branching structure, where each branch represents a decision based on specific feature values. The tree is formed by progressively selecting the most informative features to split the data into distinct classes or values. During test time, a new data point is processed from the root through the tree, following decision criteria at each node until it reaches a leaf. The leaf node provides the predicted outcome, which can be a category label or a numerical value.

2. How does Bagging improve the performance of a Decision Tree?

Bagging, or Bootstrap Aggregating, enhances the performance of a Decision Tree by reducing its variance and increasing its robustness. It works by training multiple Decision Trees on various random subsets of the original training data, generated through sampling with replacement. Each tree in the ensemble independently makes predictions, and the result is derived by averaging predictions (in regression) or using a majority vote (in classification). By combining the outputs of multiple trees, Bagging reduces the risk of overfitting, as the overall model becomes less sensitive to the biases of individual trees.

3. In what situations might a Decision Tree overfit the training data, and how can this be mitigated?

A Decision Tree is prone to overfitting when it becomes overly complex, capturing noise and intricate patterns specific to the training data that do not generalize well to new data. This usually occurs when the tree is allowed to grow too deep, resulting in many branches that fit the training data perfectly but perform poorly on unseen data. To mitigate overfitting, techniques such as pruning can be used to remove unnecessary branches, setting a maximum depth for the tree, or requiring a minimum number of samples to split a node.

4. How does Random Forest differ from a single Decision Tree?

Random Forest is an ensemble learning technique that builds multiple trees during training and combines their predictions to enhance accuracy. Unlike a single Decision Tree, which operates on the entire dataset, Random Forest trains each tree on a different random subset of the data and considers random subsets of features for each split. This randomness reduces the correlation between individual trees, making the overall model more resilient to overfitting. The final prediction in Random Forest is determined by a majority vote (in classification) or by averaging (in regression) across all the trees, resulting in a model that is more accurate and stable than a single Decision Tree.

5. What is the main idea behind Boosting in ensemble methods?

Boosting is an ensemble approach where models are trained sequentially, with each new model aiming to correct the mistakes of the previous ones. The core idea is to combine several weak learners, typically shallow Decision Trees, into a strong predictive model. As the training progresses, more weight is given to the data points that were misclassified by earlier models, ensuring that subsequent models focus on these harder-to-classify instances. The final model combines the predictions of all weak learners, typically through weighted voting or averaging, which effectively reduces bias and variance, leading to a highly accurate model suitable for complex prediction tasks.