**Interview Question**

1. What are some common hyperparameters of decision tree models, and how do they affect the model's performance?

**Ans:**

* Common hyperparameters of decision tree models include:

1) Maximum Depth (max\_depth)

2) Minimum Samples Split

3) Minimum Samples Leaf

4) Maximum Features

5) Criterion

6) Minimum Impurity Decrease

7) Maximum Leaf Nodes

* These hyperparameters affect the model's performance in the following ways:
* Adjusting the maximum depth can control the model's complexity. Deeper trees can capture more intricate patterns but are more prone to overfitting.
* Minimizing the number of samples required for splitting or leaf nodes can help prevent overfitting by ensuring that nodes have enough samples to generalize.
* Limiting the maximum number of features considered for splitting can prevent overfitting and reduce training time.
* The choice of criterion affects how the decision tree evaluates and splits the data, which can impact its predictive performance.
* Setting a threshold for minimum impurity decrease can control the tree's growth and prevent it from making splits that do not significantly improve predictive performance.
* Constraining the maximum number of leaf nodes helps control the size of the tree and prevent overfitting.

1. What is the difference between the Label encoding and One-hot encoding?

**Label Encoding:**

• In label encoding, each unique category in a categorical variable is assigned a unique integer label. The labels are usually assigned in ascending order starting from 0 or 1.

• Label encoding is suitable for ordinal categorical variables, where the categories have a natural order or hierarchy. For example, "low," "medium," and "high" could be encoded as 0, 1, and 2, respectively.

• However, for nominal categorical variables (where categories have no inherent order), label encoding may introduce unintended ordinal relationships, which could confuse the model. For example, if we encode categories "red," "green," and "blue" as 0, 1, and 2, the model may interpret this as green being "greater" than red, which is not the case.

**One-Hot Encoding**

• In one-hot encoding, each unique category is represented as a binary vector with a length equal to the number of unique categories. Only one element of the vector is "hot" (1), representing the presence of that category, while all other elements are "cold" (0).

• One-hot encoding is suitable for nominal categorical variables, as it avoids introducing ordinal relationships between categories.

• One-hot encoding increases the dimensionality of the feature space, especially when dealing with variables with many unique categories. This can lead to the "curse of dimensionality," where the model's performance may degrade if the number of features becomes too large relative to the number of observations.

• One-hot encoding can be achieved using various libraries or functions in machine learning frameworks, such as OneHotEncoder in scikit-learn or get\_dummies() function in pandas.