Assumptions of Machine Learning Classifiers

1. K-Nearest Neighbors (KNN)

KNN is a non-parametric, instance-based learning method. It does not assume a specific functional form, but it has several implicit assumptions:

Assumptions

1. **Distance Meaningfulness**
   KNN assumes that data points that are close in feature space are similar.
   → Therefore, features must be on similar scales (standardization needed).

2. **Local Smoothness**
   The target value changes smoothly in the neighborhood.
   Points near each other have similar labels.

3. **No Irrelevant Features**
   Too many irrelevant features reduce KNN performance ("curse of dimensionality").

4. **Balanced Classes**
   KNN assumes class distribution in neighborhoods is not heavily imbalanced.

2. Decision Tree Classifier

Decision Trees are flexible and non-linear, but still have some assumptions.

Assumptions

1. **No Need for Feature Scaling**
   Tree splits are based on thresholding, so scaling does not matter.

2. **Features Are Sufficient to Separate Classes**
   The tree assumes the data can be separated using recursive binary splits.

3. **Large Sample Size**
   Small datasets may cause highly unstable splits (high variance).

4. **No Strong Multicollinearity**
   Highly correlated features may confuse split selection, though trees can still handle them better than linear models.

5. **Data Purity Can Be Achieved**
   Tree assumes that splitting will eventually lead to homogeneous nodes.

3. Ensemble Learning

A. Random Forest (Bagging)

Random Forests combine many decision trees trained on bootstrapped samples.

Assumptions

1. **Independence of Trees**
   Each tree should be different due to different samples and feature subsets.

2. **Low Bias Base Learners**
   Decision Trees must be strong enough to capture patterns.

3. **No Feature Scaling Required**
   Trees do not depend on distance.

4. **Sufficient Data**
   More data helps prevent random noise from dominating.

B. Boosting (AdaBoost, Gradient Boosting, XGBoost)

Boosting builds models sequentially, where each new model corrects previous mistakes.

Assumptions

1. **Weak Learners Can Improve Sequentially**
   Boosting assumes each base learner performs slightly better than random.

2. **No Noise-Dominated Features**
   Boosting is sensitive to noise and outliers.

3. **Features Don't Need Scaling**
   Trees are used; no distance metrics involved.

4. **Additive Model Works**
   Boosting assumes errors can be reduced by combining many small models.

4. Support Vector Machines (SVM)

SVM finds the maximum-margin hyperplane to separate classes.

Assumptions

1. **Data Is Separable (linearly or with kernel)**
   If linearly inseparable, kernel trick is used.

2. **Large Margin Exists**
   SVM assumes the best decision boundary is the one with maximum margin.

3. **Feature Scaling Is Required**
   SVM uses distance-based calculations; features must be standardized.

4. **Low Noise**
   SVM is sensitive to noise and overlapping classes.

5. **Balanced Classes**
   Imbalanced datasets can shift the margin unfavorably.

5. Linear Regression

Assumptions

1. **Linearity**
   The relationship between features and target is linear.

2. **Independence of Errors**
   Residuals must not be correlated.

3. **Homoscedasticity (constant error variance)**
   Variance of residuals remains constant across predictions.

4. **Normality of Residuals**
   Needed for statistical significance tests.

5. **No Perfect Multicollinearity**
   Features must not be perfectly correlated.

6. **No Extreme Outliers**
   Outliers affect slope drastically.