

Model Development Phase Template

Date	July 2024
Team ID	739675
Project Title	Drug classification using machine learning
Maximum Marks	6 Marks

Model Selection Report

In the forthcoming Model Selection Report, various models will be outlined, detailing their descriptions, hyperparameters, and performance metrics, including Accuracy or F1 Score. This comprehensive report will provide insights into the chosen models and their effectiveness.

Model Selection Report:

Model	Description	Hyperparameters	Performance Metric (e.g., Accuracy, F1 Score)
Decision tree classifier	Decision Trees classify drugs by creating a model that splits the dataset into branches based on feature values, ultimately leading to a decision node that assigns a class label. Each internal node represents a feature, each branch a decision rule, and each leaf node a class label. They are easy to interpret and visualize but can be prone to overfitting.	----	0.97
Random forrest classifier	Random Forest is an ensemble learning method that builds multiple decision trees and merges them to obtain a more accurate and stable prediction. It	----	0.95

	reduces overfitting by averaging the results of many decision trees trained on different parts of the dataset with random feature selection. This method enhances accuracy and robustness in drug classification.		
Gradient boosting classifier	XGBoost (Extreme Gradient Boosting) is an advanced ensemble technique that builds and optimizes multiple decision trees sequentially to improve drug classification performance. It employs gradient boosting, which focuses on correcting errors made by previous trees, and incorporates regularization to prevent overfitting. XGBoost is known for its high efficiency, accuracy, and ability to handle large datasets.	----	0.93
Kneighbors classifier	K-Nearest Neighbors classifies drugs by finding the most similar instances in the feature space and assigning the class most common among them. It is a simple, non-parametric method that relies on distance metrics to determine drug similarity. KNN is intuitive and effective but can be computationally intensive with large datasets.	----	0.37