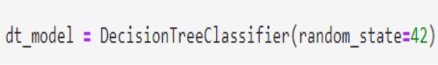
Model Development Phase Template

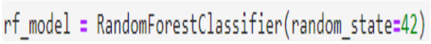
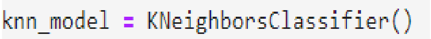
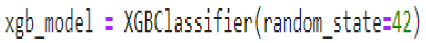
Date	11 July 2024
Team ID	SWTID1720455879
Project Title	Human Resource Management: Predicting Employee Promotions Using Machine Learning
Maximum Marks	6 Marks

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	A decision tree is a non-parametric supervised learning algorithm used for classification and regression tasks. It splits the data into subsets based on the value of input features, creating a tree-like model of decisions.	 `dt_model = DecisionTreeClassifier(random_state=42)` Using ‘random_state’ ensures that the randomness involved in initializing and training the model is controlled, leading to consistent and reproducible results.	Accuracy Score = 93%

Random Forest	Reduces overfitting, handles large datasets well, provides feature importance. An ensemble learning method that constructs multiple decision trees during training and outputs the class that is the mode of the classes or mean prediction of the individual trees.	 `rf_model = RandomForestClassifier(random_state=42)` Using 'random_state' ensures that the randomness involved in initializing and training the model is controlled, leading to consistent and reproducible results.	Accuracy Score = 95%
KNN	A simple, instance-based learning algorithm that classifies a data point based on how its neighbors are classified. The number of neighbors (k) is a key parameter.	 `knn_model = KNeighborsClassifier()` Initialising the model	Accuracy Score = 89%
XGBoost	High performance, handles missing values, robust to overfitting due to regularization. It combines the predictions of multiple weak learners to improve accuracy.	 `xgb_model = XGBClassifier(random_state=42)` Using 'random_state' ensures that the randomness involved in initializing and training the model is controlled, leading to consistent and reproducible results.	Accuracy Score = 94%