



## **Model Optimization and Tuning Phase Template**

Date	15 March 2024
Team ID	xxxxxx
Project Title	Human Resource Management: Predicting Employee Promotions Using Machine Learning
Maximum Marks	10 Marks

## **Model Optimization and Tuning Phase**

The Model Optimization and Tuning Phase involves refining machine learning models for peak performance. It includes optimized model code, fine-tuning hyperparameters, comparing performance metrics, and justifying the final model selection for enhanced predictive accuracy and efficiency.

## **Hyperparameter Tuning Documentation (6 Marks):**

Model	Tuned Hyperparameters	Optimal Values
Decision Tree	<pre># Function to train and evaluate a Decision Tree model with hyperparameter tuning def decisionTree(X_train, X_test, y_train, y_test):     # Define the parameter grid     param grid = {         'max_depth': [None, 10, 20, 30, 40, 50],         'min_samples_split': [2, 10, 20],         'min_samples_leaf': [1, 5, 10],         'criterion': ['gini', 'entropy']     } }</pre>	Best Parameters found by GridSearch(I): ('criterion': 'entropy', 'max_depth': 40, 'min_samples_leaf': 1, 'min_samples_split': 2)  Accuracy: 0.94
Random Forest	<pre># Function to train and evaluate a Random Forest model def randomForest(X_train, X_test, y_train, y_test):  # Define the parameter grid param_grid = {     'n_estimators': [100, 200, 300],     'max_depth': [None, 10, 20, 30],     'min_samples_split': [2, 5, 10],     'min_samples_leaf': [1, 2, 4],     'bootstrap': [True, False] }</pre>	Sect Parameters found by GridSecrotCO: ["tootstrap": False, 'max depth": Nove, 'min_samples_leaf': 1, 'min_samples_split': 5, 'n_estimeters': 180]  Accuracy: 0.96





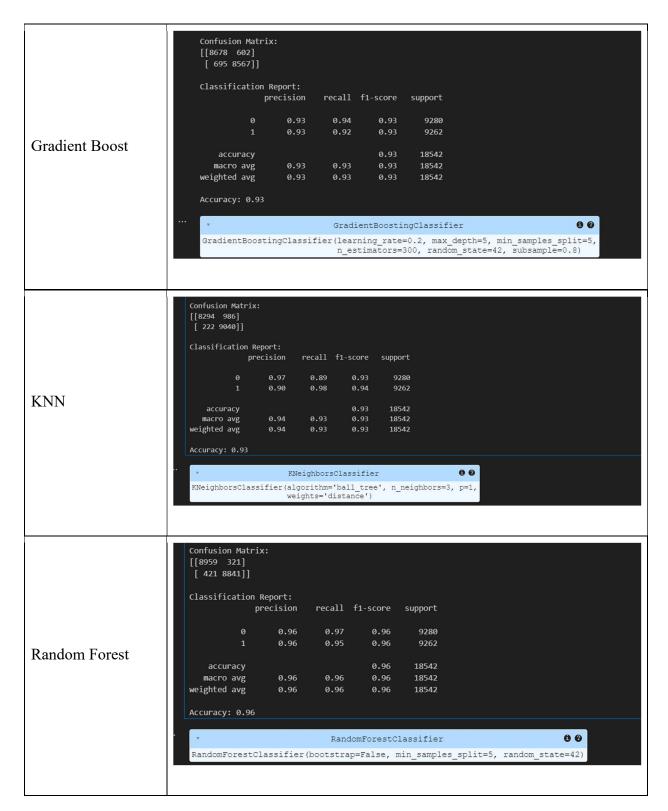


## **Performance Metrics Comparison Report (2 Marks):**

Model	Optimized Metric		
Decision Tree	Confusion Matrix: [[8668 612]         [453 8809]]  Classification Report:		











Final Model	Reasoning
	I chose the Random Forest model as the final model for predicting employee
	promotions due to its superior accuracy (96%) compared to other models
	like Decision Tree, KNN, and Gradient Boosting. Random Forest is robust,
	handles overfitting well, and provides insights into feature importance. It
	captures complex, non-linear relationships within the data and is scalable for
	large datasets. Additionally, hyperparameter tuning further optimized its
Random Forest	performance, making it a reliable and efficient choice for this task