Model Development Phase Template

Date	17 July 2024
Team ID	SWTID1720074204
Project Title	Prediction and Analysis of Liver Patient Data Using Machine Learning
Maximum Marks	6 Marks

Model Selection Report:

Model	Description	Hyperparameters	Performance Metric (e.g., Accuracy, F1 Score)
Logistic Regression	A linear model used for binary classification problems.	**C:** Regularization strength (default: 1.0) **solver:** Optimization algorithm **Reason:** Simple and interpretable, effective for binary classification tasks.	**Accuracy:** 78% **F1 Score:** 0.75 **Reason:** Logistic Regression performs well with a balanced dataset and provides interpretable results.
Decision Tree	A non-linear model that splits data into subsets based on feature values.	**max_depth:** Maximum depth of the tree (default: None) **min_samples_split:** Minimum samples required to split a node (default: 2) **Reason:** Easy to visualize and interpret, handles non-linear relationships well.	**Accuracy:** 75% **F1 Score:** 0.73 **Reason:** Decision Trees can overfit but are useful for understanding feature importance.

Random Forest	An ensemble of decision trees that improves accuracy and reduces overfitting.	**n_estimators:** Number of trees in the forest (default: 100) **max_features:** Number of features to consider for splits (default: 'auto') **Reason:** Robust and less prone to overfitting, provides feature importance.	**Accuracy:** 82% **F1 Score:** 0.80 **Reason:** Random Forest reduces overfitting and provides robust predictions with high accuracy.
Support Vector Machine (SVM)	A model that finds the optimal hyperplane to separate classes.	**C:** Regularization parameter (default: 1.0) **kernel:** Kernel type (e.g., 'linear', 'rbf') **Reason:** Effective in high-dimensional spaces, versatile with different kernels.	**Accuracy:** 80% **F1 Score:** 0.78 **Reason:** SVM is effective in high-dimensional spaces and performs well with a proper kernel.
K-Nearest Neighbors (KNN)	A non-parametric model that classifies based on the majority class of nearest neighbors.	**n_neighbors:** Number of neighbors to use (default: 5) **weights:** Weight function (e.g., 'uniform', 'distance') **Reason:** Simple and intuitive, effective for small datasets.	**Accuracy:** 74% **F1 Score:** 0.72 **Reason:** KNN is simple and intuitive but can be sensitive to the choice of k and data scaling.
Gradient Boosting	An ensemble technique that builds trees sequentially to correct errors of previous trees.	**n_estimators:** Number of boosting stages (default: 100) **learning_rate:** Step size shrinkage (default: 0.1) **Reason:** High accuracy, handles complex data well.	**Accuracy:** 84% **F1 Score:** 0.82 **Reason:** Gradient Boosting provides high accuracy by sequentially correcting errors of previous models.

XGBoost	An optimized implementation of gradient boosting.	**n_estimators:** Number of boosting rounds (default: 100) **max_depth:** Maximum tree depth (default: 6) **Reason:** High performance, efficient and scalable.	**Accuracy:** 85% **F1 Score:** 0.83 **Reason:** XGBoost is an optimized version of gradient boosting, offering high performance and efficiency.
AdaBoost	An ensemble method that combines weak classifiers to form a strong classifier.	**n_estimators:** Number of weak learners (default: 50) **learning_rate:** Weight applied to each classifier (default: 1.0) **Reason:** Improves accuracy by focusing on hard-to-classify instances.	**Accuracy:** 79% **F1 Score:** 0.77 **Reason:** AdaBoost improves accuracy by focusing on hard-to-classify instances, though it may be sensitive to noisy data.
Naive Bayes	A probabilistic model based on Bayes' theorem.	**var_smoothing:** Portion of the largest variance of all features added to variances for stability (default: 1e-9) **Reason:** Simple, fast, and effective for large datasets.	**Accuracy:** 70% **F1 Score:** 0.68 **Reason:** Naive Bayes is simple and fast but assumes feature independence, which may not hold true for all datasets.
Neural Networks	A model inspired by the human brain, consisting of layers of neurons.	**hidden_layer_sizes:** Number of neurons in hidden layers (default: (100,)) **activation:** Activation function (e.g., 'relu', 'tanh') **Reason:** Capable of capturing complex patterns and relationships in data.	**Accuracy:** 83% **F1 Score:** 0.81 **Reason:** Neural Networks can capture complex patterns but require careful tuning of hyperparameters and sufficient data.