ITD105 Case Study #1

Comparing Machine Learning Algorithms

Name: Allan Raymart C. Paraiso
Video Link: https://drive.google.com/file/d/1h-D_u2XaEjVgdSOuNNI10_vDoCwXeg7-/view?usp=sharing

I CLASSIFICATION

Train the **classification dataset** using various machine learning algorithms designed for classification. Evaluate and compare these models by applying different resampling techniques and utilizing appropriate performance metrics.

Classification Dataset

Dataset Name : Cirrhosis Patient Survival Prediction

Features: ID, N_Days, Status, Age, Bilirubin, Cholesterol, Albumin, Copper, Alk_Phos, SGOT, Tryglicerides, Platelets, Prothrombin, Stage

Set A
Resampling Technique : Split into Train and Test Sets
Classification Metric : Confusion Matrix and Classification Report

ML Algorithm (Classification)	Confusion Matrix ***(Provide the matrix and classification report of each algorithm)***
CART (Classification and Regression Trees)	Accuracy: 76.190% Confusion Matrix: [[41 2 2] [3 0 0] [11 2 23]] Classification Report: precision recall f1-score support 0 0.75 0.91 0.82 45 1 0.00 0.00 0.00 3 2 0.92 0.64 0.75 36 accuracy 0.76 84 macro avg 0.56 0.52 0.52 84 weighted avg 0.79 0.76 0.76 84
Gaussian Naive Bayes/Naive Bayes	Label Mapping: Label: C, Numerical Value: 0 Label: CL, Numerical Value: 1 Label: D, Numerical Value: 2 Accuracy: 73.810% Confusion Matrix: [[46 0 1] [2 0 0] [33 0 2]] Classification Report: precision recall f1-score support 0 0.57 0.98 0.72 47 1 0.00 0.00 0.00 2 2 0.67 0.06 0.11 35 accuracy 0.57 84 macro avg 0.41 0.35 0.27 84 weighted avg 0.60 0.57 0.45 84

Gradient Boosting Machines (AdaBoost)	Label Mapping: Label: C, Numerical Value: 0 Label: CL, Numerical Value: 1 Label: D, Numerical Value: 2 Accuracy: 83.333% Confusion Matrix: [[42 1 4] [2 0 0] [6 1 28]] Classification Report: precision recall f1-score support 0 0.84 0.89 0.87 47 1 0.00 0.00 0.00 2 2 0.88 0.80 0.84 35 accuracy 0.83 84 macro avg 0.57 0.56 0.57 84 weighted avg 0.83 0.83 0.83 84
K-Nearest Neighbors (K-NN)	Label Mapping: Label: C, Numerical Value: 0 Label: CL, Numerical Value: 1 Label: D, Numerical Value: 2 Accuracy: 71.429% Confusion Matrix: [[41 0 6] [2 0 0] [16 0 19]] Classification Report: precision recall f1-score support 0 0.69 0.87 0.77 47 1 0.00 0.00 0.00 2 2 0.76 0.54 0.63 35 accuracy 0.71 84 macro avg 0.48 0.47 0.47 84 weighted avg 0.71 0.71 0.70 84
Logistic Regression	Label Mapping: Label: C, Numerical Value: 0 Label: CL, Numerical Value: 1 Label: D, Numerical Value: 2 Accuracy: 78.571% Confusion Matrix: [[41 0 6] [2 0 0] [10 0 25]] Classification Report: precision recall f1-score support 0 0.77 0.87 0.82 47 1 0.00 0.00 0.00 2 2 0.81 0.71 0.76 35 accuracy 0.79 84 macro avg 0.53 0.53 0.53 84 weighted avg 0.77 0.79 0.77 84
Multi-Layer Perceptron (MLP)	Label Mapping: Label: C, Numerical Value: 0 Label: CL, Numerical Value: 1 Label: D, Numerical Value: 2 Accuracy: 73.810% Confusion Matrix: [[41 0 6] [2 0 0] [14 0 21]] Classification Report: precision recall f1-score support 0 0.72 0.87 0.79 47 1 0.00 0.00 0.00 2 2 0.78 0.60 0.68 35 accuracy 0.74 84 macro avg 0.50 0.49 0.49 84 weighted avg 0.73 0.74 0.72 84
Perceptron	Label Mapping: Label: C, Numerical Value: 0 Label: CL, Numerical Value: 1 Label: D, Numerical Value: 2 Accuracy: 44.048% Confusion Matrix: [[3 0 44] [0 0 2] [1 0 34]] Classification Report: precision recall f1-score support 0 0.75 0.06 0.12 47 1 0.00 0.00 0.00 2 2 0.42 0.97 0.59 35 accuracy 0.44 84 macro avg 0.39 0.35 0.24 84 weighted avg 0.60 0.44 0.31 84

Random Forest	Label Mapping: Label: C, Numerical Value: 0 Label: CL, Numerical Value: 1 Label: D, Numerical Value: 2 Accuracy: 78.571% Confusion Matrix: [[3 0 44] [0 0 2] [1 0 34]] Classification Report: precision recall f1-score support 0 0.75 0.06 0.12 47 1 0.00 0.00 0.00 2 2 0.42 0.97 0.59 35 accuracy 0.44 84 macro avg 0.39 0.35 0.24 84 weighted avg 0.60 0.44 0.31 84
Support Vector Machines (SVM)	Label Mapping: Label: C, Numerical Value: 0 Label: CL, Numerical Value: 1 Label: D, Numerical Value: 2 Accuracy: 78.571% Confusion Matrix: [[41 1 5] [1 1 0] [10 1 24]] Classification Report: precision recall f1-score support 0 0.79 0.87 0.83 47 1 0.33 0.50 0.40 2 2 0.83 0.69 0.75 35 accuracy 0.79 84 macro avg 0.65 0.69 0.66 84 weighted avg 0.79 0.79 0.79 84

Set B *(should use different resampling technique and classification metric)*
Resampling Technique: Repeated Random Train-Test splits
Classification Metric: Logarithmic Loss

ML Algorithm (Classification)	
CART (Classification and Regression Trees)	Mean Logarithmic Loss: 2.995
Gaussian Naive Bayes/Naive Bayes	Mean Logarithmic Loss: 0.703
Gradient Boosting Machines (AdaBoost)	Mean Logarithmic Loss: 0.858
K-Nearest Neighbors (K-NN)	Mean Logarithmic Loss: 3.497
Logistic Regression	Mean Logarithmic Loss: 0.563
Multi-Layer Perceptron (MLP)	Mean Logarithmic Loss: 7.086
Perceptron	Mean Logarithmic Loss: 0.700
Random Forest	Mean Logarithmic Loss: 0.63732
Support Vector Machines (SVM)	Mean Logarithmic Loss: 0.56745

Set C *(should use different resampling technique and classification metric)*
Resampling Technique: K-fold Cross Validation
Classification Metric: Classification Report

ML Algorithm (Classification)	Average Precision
CART (Classification and Regression Trees)	0.761
Gaussian Naive Bayes/Naive Bayes	0.710
Gradient Boosting Machines (AdaBoost)	0.759
K-Nearest Neighbors (K-NN)	0.687
Logistic Regression	0.772
Multi-Layer Perceptron (MLP)	0.634
Perceptron	0.673
Random Forest	0.759
Support Vector Machines (SVM)	0.646

Results interpretation

Overall, the results suggest that Logistic Regression, SVM, and Random Forest are the best machine learning algorithms for the task of binary classification for the reason that Logistic Regression has the lowest mean logarithmic loss, followed by Support Vector Machines (SVM) and Random Forest. Logistic Regression also has the highest average precision, followed by CART (Classification and Regression Trees) and Gradient Boosting Machines (AdaBoost)..

Based on the results, perform algorithm/hyperparameter tuning (at least 3) of the chosen ML algorithm.

ML Algorithm: Gaussian Naïve Bayes
Sampling Technique - Train/Test Split (80:20)
Classification Metrics – Accuracy

ML Algorithm: Gradient Boosting Adaboost
Sampling Technique - Train/Test Split (80:20)
Classification Metrics – Accuracy

ML Algorithm: K-Nearest Neighbors
Sampling Technique - Train/Test Split (80:20)
Classification Metrics – Accuracy

	SVM Hyperparameters			
	random_state	**Precision Weighted avg**	**Precision macro avg**	**Accuracy**
Gaussian Naïve Bayes	7	0.41	0.60	77.810%
Adaboost	7	0.57	0.83	83.333%
KNN	7	0.48	0.71	71.428%

Results interpretation:

Based on the result the adaboost has the most accuracy given the precision weighted average, macro average and the overall accuracy.

II REGRESSION

Train the **regression dataset** using various machine learning algorithms designed for regression. Evaluate and compare these models by applying different resampling techniques and utilizing appropriate performance metrics.

Regression Dataset
Dataset Name : insurance costs
Features: age, sex,bmi,chidren,smoker,region

Set A
Resampling Technique : train test split 80:20
Regression Metric : Mean Absolute Error

ML Algorithm (Regression)	
CART (Classification and Regression Trees)	2651.093
Elastic Net	7423.916
Gradient Boosting Machines (AdaBoost)	3922.616
K-Nearest Neighbors (K-NN)	7872.695
Lasso Regression	3972.271
Ridge Regression	3984.913
Linear Regression	3971.629
Multi-Layer Perceptron (MLP)	3970.469
Random Forest	2575.167

Set B *(should use different resampling technique and regression metric)*

Resampling Technique: K Fold

Regression Metric: Mean Squared Error

ML Algorithm (Regression)	
CART (Classification and Regression Trees)	44243399.200
Elastic Net	89789075.134
Gradient Boosting Machines (AdaBoost)	25196255.396
K-Nearest Neighbors (K-NN)	121422424.326
Lasso Regression	37003521.552
Ridge Regression	37005349.740

Linear Regression	37004496.989
Multi-Layer Perceptron (MLP)	123438921.241
Random Forest	25494408.001

Results interpretation (Set A and Set B):

Given the results in set a and set b, it is very evident that the dataset is seen as inaccurate given the large number of mean absolute errors and mean squared errors, but if for some reason the dataset is still used, the best choice for machine learning algorithm is random forest given that it has the lowest error of all the models tested.

Based on the results, perform at algorithm tuning (at least 3) of the chosen ML algorithm.

ML Algorithm: Random Forest
Sampling Technique - Train/Test Split (80:20)
Regression Metrics – MAE

ML Algorithm: Support Vector Machines
Sampling Technique – K fold
Regression Metrics – MAE

ML Algorithm: Random Forest
Sampling Technique – K-FOLD
Regression Metrics – MAE

	SVM Hyperparameters			
	epsilon	**Kernel**	**C**	**MAE**
Model I	0.1	linear	1.0	6912.019
Model II	0.2	linear	1.25	6554.593
Model III	0.2	poly	1.0	7655.444

Results interpretation:

Based on the results it can be concluded that the dataset is faulty given the high numbers of mean absolute errors, across all tested models, and therefore deemed inaccurate