JADBio Description of Performed Analysis

Setup

JADBio version **1.4.118** ran on dataset **bmi** with **741** samples and **4** features to create a predictive model for outcome named **BmiClass**. The outcome was discrete leading to a **classification** modeling.

The preferences of the analysis were set to **true** for feature selection and **false** for full feature models tried.
The **AUC** metric was used to optimize for the best model.
The maximum number of features to select was set to **25**.
The effort to spend on tuning the algorithms were set to **Quick**.
The number of CPU cores to use for the analysis was set to **1**.
The execution time was **00:01:59**.

Configuration Space

JADBio's AI decide to try the following algorithms and tuning hyper-parameter values:

Algorithm Type	Algorithm	Hyper-parameter	Set of Values
Preprocessing	Mean Imputation		
	Mode Imputation		
	Constant Removal		
	Variable Normalization		
Feature Selection	Test-Budgeted Statistically Equivalent Signature (SES)	maxK	2.0
		alpha	0.05
	LASSO	penalty	1.0
Modeling	Classification Random Forest with Deviance splitting criterion	nTrees	100
		minLeafSize	3.0
	Ridge Logistic Regression	lambda	1.0
	Classification Decision Tree with Deviance splitting criterion	minLeafSize	3
		alpha	0.05
	Support Vector Machines (SVM) of type C-SVC with Gaussian Kernel	cost	1.0
		gamma	1.0
	Support Vector Machines (SVM) of type C-SVC with Linear Kernel	cost	1.0
	Support Vector Machines (SVM) of type C-SVC with Polynomial Kernel	cost	1.0
		degree	3

Algorithm Type	Algorithm	Hyper-parameter	Set of Values
		gamma	1.0

Leading to **17** combinations and corresponding configurations (machine learning pipelines) to try. For the full configurations tested see the Appendix.

Configuration Estimation Protocol

JADBio's AI system decided to estimate the out-of-sample performance of the models produced by each configuration using **Repeated 10-fold CV without dropping (max. repeats = 20).** Overall, 17 configurations × 20 repeats × 10 folds = 170 models were set out to train.

JADBio Results Summary

Overview

A result summary is presented for analysis optimized for Performance. The model is produced by applying the algorithms in sequence (configuration) on the training data:

Preprocessing	Feature Selection	Predictive algorithm
Mean Imputation, Mode Imputation, Constant Removal, Standardization	Test-Budgeted Statistically Equivalent Signature (SES) algorithm with hyper-parameters: maxK = 2, alpha = 0.05 and budget = 3 * nvars	Classification Random Forest training 100 trees with Deviance splitting criterion, minimum leaf size = 3, splits = 1, alpha = 1, and variables to split = 0.816 sqrt (nvars)

The Area Under the ROC Curve is shown in the figure below:

Metric	Mean estimate	CI
Area Under the ROC Curve	0.999	[0.990, 1.000]
Mean Average Precision (a.k.a. Average Area Under the Precision-Recall curve)	0.999	[0.987, 1.000]
Accuracy	0.995	[0.982, 1.000]
Balanced Accuracy	0.990	[0.929, 1.000]
Average F1 score	0.991	[0.959, 1.000]
Average Matthews correlation	0.989	[0.943, 1.000]
Precision for class Normal Weight	1.000	[1.000, 1.000]
Precision for class Obese Class 1	0.932	[0.761, 1.000]
Precision for class Obese Class 2	0.980	[0.892, 1.000]
Precision for class Obese Class 3	1.000	[1.000, 1.000]

Metric	Mean estimate	CI
Precision for class Overweight	0.999	[0.983, 1.000]
Precision for class Underweight	1.000	[1.000, 1.000]
MCC for class Normal Weight	1.000	[1.000, 1.000]
MCC for class Obese Class 1	0.967	[0.731, 1.000]
MCC for class Obese Class 2	0.977	[0.886, 1.000]
MCC for class Obese Class 3	0.985	[0.895, 1.000]
MCC for class Overweight	0.996	[0.980, 1.000]
MCC for class Underweight	1.000	[1.000, 1.000]
True Positive Rate for class Normal Weight	1.000	[1.000, 1.000]
True Positive Rate for class Obese Class 1	0.987	[0.733, 1.000]
True Positive Rate for class Obese Class 2	0.982	[0.908, 1.000]
True Positive Rate for class Obese Class 3	0.978	[0.895, 1.000]
True Positive Rate for class Overweight	0.994	[0.974, 1.000]
True Positive Rate for class Underweight	1.000	[1.000, 1.000]
Sensitivity for class Normal Weight	1.000	[1.000, 1.000]
Sensitivity for class Obese Class 1	0.987	[0.733, 1.000]
Sensitivity for class Obese Class 2	0.982	[0.908, 1.000]
Sensitivity for class Obese Class 3	0.978	[0.895, 1.000]
Sensitivity for class Overweight	0.994	[0.974, 1.000]
Sensitivity for class Underweight	1.000	[1.000, 1.000]
Specificity for class Normal Weight	1.000	[1.000, 1.000]
Specificity for class Obese Class 1	0.997	[0.992, 1.000]
Specificity for class Obese Class 2	0.998	[0.991, 1.000]
Specificity for class Obese Class 3	1.000	[1.000, 1.000]
Specificity for class Overweight	1.000	[0.996, 1.000]
Specificity for class Underweight	1.000	[1.000, 1.000]
Average Precision for class Normal Weight	1.000	[1.000, 1.000]

Metric	Mean estimate	CI
Average Precision for class Obese Class 1	0.995	[0.914, 1.000]
Average Precision for class Obese Class 2	0.991	[0.922, 1.000]
Average Precision for class Obese Class 3	0.996	[0.956, 1.000]
Average Precision for class Overweight	0.999	[0.987, 1.000]
Average Precision for class Underweight	1.000	[1.000, 1.000]

Feature Selection

There were **1** features selected out of the **4** available.

The selected features consist of the following subset called a signature. **There was a single signature identified.** The first signature identified by the system is the set: **Bmi** in order of importance. The following features cannot be substituted with others and still obtain an equal predictive performance: **Bmi**.

The performance achieved by adding each feature in sequence to the model relative to the performance of the final model with all selected features is shown below. The features are added in order of importance:

Some features may not seem to add predictive performance to the model; however, the feature selection algorithms include them as an effort to make the final model more robust to noise. The performances achieved by a model that contains all features except one, relative to the performance achieved when the feature is removed is shown below:

For some features there is no noticeable drop in performance when they are removed because they carry predictive information that is shared by other features selected.

The separation of the predictions of the classes achieved by the model is shown in the box-plots below. These are the out-of-sample predictions made by model produced by the same configuration as the final model when the sample was used for testing (e.g.., during cross-validation) and was not used to train the model.

Appendix

Configuration	Preprocessing	Name	Hyperparams	Name	Hyperparams	Performance (unadjusted)	Time (miliseconds)	Dropped
1	Mean Imputation, Mode Imputation, Constant Removal, Standardization	Test-Budgeted Statistically Equivalent Signature (SES)	maxK = 2, alpha = 0.05, budget = 3 * nvars	Classification Random Forest with Deviance splitting criterion	ntrees = 100, minimum leaf size = 3	1	00:00:00.167	false
2	Mean Imputation,	Test-Budgeted	maxK = 2, alpha = 0.05,	Ridge Logistic	lambda = 1.0	0.5434869526758073	00:00:00.206	false

Configuration	Preprocessing	Name	Hyperparams	Name	Hyperparams	Performance (unadjusted)	Time (miliseconds)	Dropped
	Mode Imputation, Constant Removal, Standardization	Statistically Equivalent Signature (SES)	budget = 3 * nvars	Regression				
3	Mean Imputation, Mode Imputation, Constant Removal, Standardization	LASSO	penalty = 1.0	Ridge Logistic Regression	lambda = 1.0	0.5484040556939114	00:00:05.5358	false
4	Mean Imputation, Mode Imputation, Constant Removal, Standardization	Test-Budgeted Statistically Equivalent Signature (SES)	maxK = 2, alpha = 0.05, budget = 3 * nvars	Classification Random Forest with Deviance splitting criterion	ntrees = 100, minimum leaf size = 3	1	00:00:00.163	false
5	Mean Imputation, Mode Imputation, Constant Removal, Standardization	LASSO	penalty = 1.0	Classification Random Forest with Deviance splitting criterion	ntrees = 100, minimum leaf size = 3	0.9986292757093841	00:00:05.5293	false
6	Mean Imputation, Mode Imputation, Constant Removal, Standardization	Test-Budgeted Statistically Equivalent Signature (SES)	maxK = 2, alpha = 0.05, budget = 3 * nvars	Classification Decision Tree with Deviance splitting criterion	minimum leaf size = 3, alpha = 0.05	0.9960554430949168	00:00:00.171	false
7	Mean Imputation, Mode Imputation, Constant Removal, Standardization	LASSO	penalty = 1.0	Support Vector Machines (SVM) of type C-SVC	kernel = 'Gaussian Kernel', cost = 1.0, gamma = 1.0	0.9870779032948948	00:00:05.5328	false
8	Mean Imputation, Mode Imputation, Constant Removal, Standardization	LASSO	penalty = 1.0	Classification Random Forest with Deviance splitting criterion	ntrees = 100, minimum leaf size = 3	0.9997188484030589	00:00:05.5308	false

Configuration	Preprocessing	Name	Hyperparams	Name	Hyperparams	Performance (unadjusted)	Time (miliseconds)	Dropped
9	Mean Imputation, Mode Imputation, Constant Removal, Standardization	Test-Budgeted Statistically Equivalent Signature (SES)	maxK = 2, alpha = 0.05, budget = 3 * nvars	Classification Random Forest with Deviance splitting criterion	ntrees = 100, minimum leaf size = 3	1	00:00:00.158	false
10	Mean Imputation, Mode Imputation, Constant Removal, Standardization	Test-Budgeted Statistically Equivalent Signature (SES)	maxK = 2, alpha = 0.05, budget = 3 * nvars	Support Vector Machines (SVM) of type C-SVC	kernel = 'Gaussian Kernel', cost = 1.0, gamma = 1.0	0.9999007700246091	00:00:00.218	false
11	IdentityFactory	FullSelector	-	Trivial model	-	0.5	00:00:00.000	false
12	Mean Imputation, Mode Imputation, Constant Removal, Standardization	LASSO	penalty = 1.0	Classification Decision Tree with Deviance splitting criterion	minimum leaf size = 3, alpha = 0.05	0.9960554430949168	00:00:05.5276	false
13	Mean Imputation, Mode Imputation, Constant Removal, Standardization	Test-Budgeted Statistically Equivalent Signature (SES)	maxK = 2, alpha = 0.05, budget = 3 * nvars	Support Vector Machines (SVM) of type C-SVC	kernel = 'Linear Kernel', cost = 1.0	0.9998811886427985	00:00:00.183	false
14	Mean Imputation, Mode Imputation, Constant Removal, Standardization	LASSO	penalty = 1.0	Support Vector Machines (SVM) of type C-SVC	kernel = 'Linear Kernel', cost = 1.0	0.887478022402161	00:00:05.5282	false
15	Mean Imputation, Mode Imputation, Constant Removal, Standardization	LASSO	penalty = 1.0	Classification Random Forest with Deviance splitting criterion	ntrees = 100, minimum leaf size = 3	0.9997188484030589	00:00:05.5303	false
16	Mean Imputation, Mode Imputation, Constant	Test-Budgeted Statistically Equivalent	maxK = 2, alpha = 0.05, budget = 3 * nvars	Support Vector Machines (SVM) of type C-SVC	kernel = 'Polynomial Kernel', cost = 1.0, gamma	0.9998095519509349	00:00:00.167	false

Configuration	Preprocessing	Name	Hyperparams	Name	Hyperparams	Performance (unadjusted)	Time (miliseconds)	Dropped
	Removal, Standardization	Signature (SES)			= 1.0, degree = 3			
17	Mean Imputation, Mode Imputation, Constant Removal, Standardization	LASSO	penalty = 1.0	Support Vector Machines (SVM) of type C-SVC	kernel = 'Polynomial Kernel', cost = 1.0, gamma = 1.0, degree = 3	0.9686422848425442	00:00:05.5300	false