Metabolomic Data Analysis with MetaboAnalyst 5.0

Name: Human Cachexia

February 18, 2023

1 Background

The metabolome is well known to be a sensitive measure of health and disease, reflecting alterations to the genome, proteome, and transcriptome, as well as changes in life style and environment. As such, one common goal of metabolomic studies is biomarker discovery, which aims to identify a metabolite or a set of metabolites capable of classifying conditions or disease with high sensitivity (true-positive rate) and specificity (true negative rate). Biomarker discovery is achieved through building predictive models of one or multiple metabolites and evaluating the performance and robustness of the model to classify new patients into diseased or healthy categories. The Biomarker analysis module supports all common ROC-curve based biomarker analyses. It includes several options for single biomarker or biomarker panel analysis, as well as for manual biomarker model creation and evaluation. For a comprehensive introductory tutorial and further details concerning biomarker analysis, please refer to **Translational biomarker discovery in clinical metabolomics: an introductory tutorial** by Xia et al. 2013 (PMID: 23543913).

2 Biomarker Analysis Overview

The module consists of five steps - uploading the data, data processing, biomarker selection, performance evaluation, and model creation. There are several options within MetaboAnalyst to perform each of these steps, supporting all common ROC-curve based biomarker analyses.

3 Data Input

The biomarker analysis module accepts either a compound concentration table, spectral binned data, or a peak intensity table. The class label must contain only two groups. Multi-group biomarker analysis is supported. The format of the data must be specified, identifying whether the samples are in rows or columns. The data may either be .csv or .txt files.

3.0.1 Data Integrity Check

Before data analysis, a data integrity check is performed to make sure that all of the necessary information has been collected. The class labels must be present and must contain only two groups. Compound concentration or peak intensity values must all be non-negative numbers. By default, all missing values, zeros and negative values will be replaced by the half of the minimum positive value found within the data (see next section).

3.0.2 Data Filtering

The purpose of data filtering is to identify and remove variables that are unlikely to be of use when modeling the data. No phenotype information is used in the filtering process, so the result can be used with any downstream analysis. This step can usually improve the results. Data filtering is strongly recommended for datasets with a large number of variables (> 250) and for datasets which contain a lot of noise (i.e.chemometrics data). Filtering can usually improve your results[1].

For data with < 250 of variables, filtering will reduce 5% of variables; For a total number of variables between 250 and 500, 10% of variables will be removed; For a total number of variables bewteen 500 and 1000, 25% of variables will be removed; Finally, 40% of variables will be removed for data with over 1000 variables.

No data filtering was performed.

3.0.3 Missing value imputations

Too many zeroes or missing values will cause difficulties in the downstream analysis. MetaboAnalystR offers several different methods for this purpose. The default method replaces all the missing and zero values with a small values (the half of the minimum positive values in the original data) assuming to be the detection limit. The assumption of this approach is that most missing values are caused by low abundance metabolites (i.e.below the detection limit). In addition, since zero values may cause problem for data normalization (i.e. log), they are also replaced with this small value. User can also specify other methods, such as replace by mean/median, or use K-Nearest Neighbours (KNN), Probabilistic PCA (PPCA), Bayesian PCA (BPCA) method, Singular Value Decomposition (SVD) method to impute the missing values [2]. Please select the one that is the most appropriate for your data.

Zero or missing values were replaced by 1/5 of the min positive value for each variable.

[1] Hackstadt AJ, Hess AM. *Filtering for increased power for microarray data analysis*, BMC Bioinformatics. 2009; 10: 11.

[2] Stacklies W, Redestig H, Scholz M, Walther D, Selbig J. *pcaMethods: a bioconductor package, providing PCA methods for incomplete data.*, Bioinformatics 2007 23(9):1164-1167

3.1 Data Normalization

The data is stored as a table with one sample per row and one variable (bin/peak/metabolite) per column. The normalization procedures implemented below are grouped into four categories. Sample specific normalization allows users to manually adjust concentrations based on biological inputs (i.e. volume, mass); row-wise normalization allows general-purpose adjustment for differences among samples; data transformation and scaling are two different approaches to make features more comparable. You can use one or combine both to achieve better results.

The normalization consists of the following options:

1. Row-wise procedures:
    - Sample specific normalization (i.e. normalize by dry weight, volume)
    - Normalization by the sum
    - Normalization by the sample median
    - Normalization by a reference sample (probabilistic quotient normalization)[3]
    - Normalization by a pooled or average sample from a particular group
    - Normalization by a reference feature (i.e. creatinine, internal control)
    - Quantile normalization
2. Data transformation :
    - Log transformation (base 10)
    - Square root transformation
    - Cube root transformation
3. Data scaling:
    - Mean centering (mean-centered only)
    - Auto scaling (mean-centered and divided by standard deviation of each variable)
    - Pareto scaling (mean-centered and divided by the square root of standard deviation of each variable)
    - Range scaling (mean-centered and divided by the value range of each variable)

[3]Dieterle F, Ross A, Schlotterbeck G, Senn H. *Probabilistic quotient normalization as robust method to account for dilution of complex biological mixtures. Application in 1H NMR metabonomics*, 2006, Anal Chem 78 (13);4281 - 4290

3.2 Biomarker Analysis Normalization

The normalization step for the biomarker analysis module includes an additional option for calculating ratios between metabolite concentrations. Ratios between two metabolite concentrations may provide more information than the two metabolite concentrations separately. MetaboAnalystR will compute ratios between all possible metabolite pairs and then select the top ranked ratios (based on p-values) to include with the data for further biomarker analysis. Please note, there is a potential overfitting issue associated with this procedure. The main purpose of computing ratios of metabolite concentrations is to improve the chances of biomarker discovery, therefore users will need to validate their performance in future, independent studies. Log normalization of the data will be performed during the process. No ratios between metabolite concentration pairs were computed.

4 Classical ROC curve analysis

The aim of classical ROC curve analysis is to evaluate the performance of a single feature, either one metabolite or a combined metabolite ratio pair, as a biomarker. The ROC curve summarizes the sensitivity and specificity of that single feature to accurately classify data, which can then be used to compare the overall accuracy of different biomarkers. Figure 1 ROC curve of an individual biomarker.Figure 2 Boxplot of an individual biomarker.

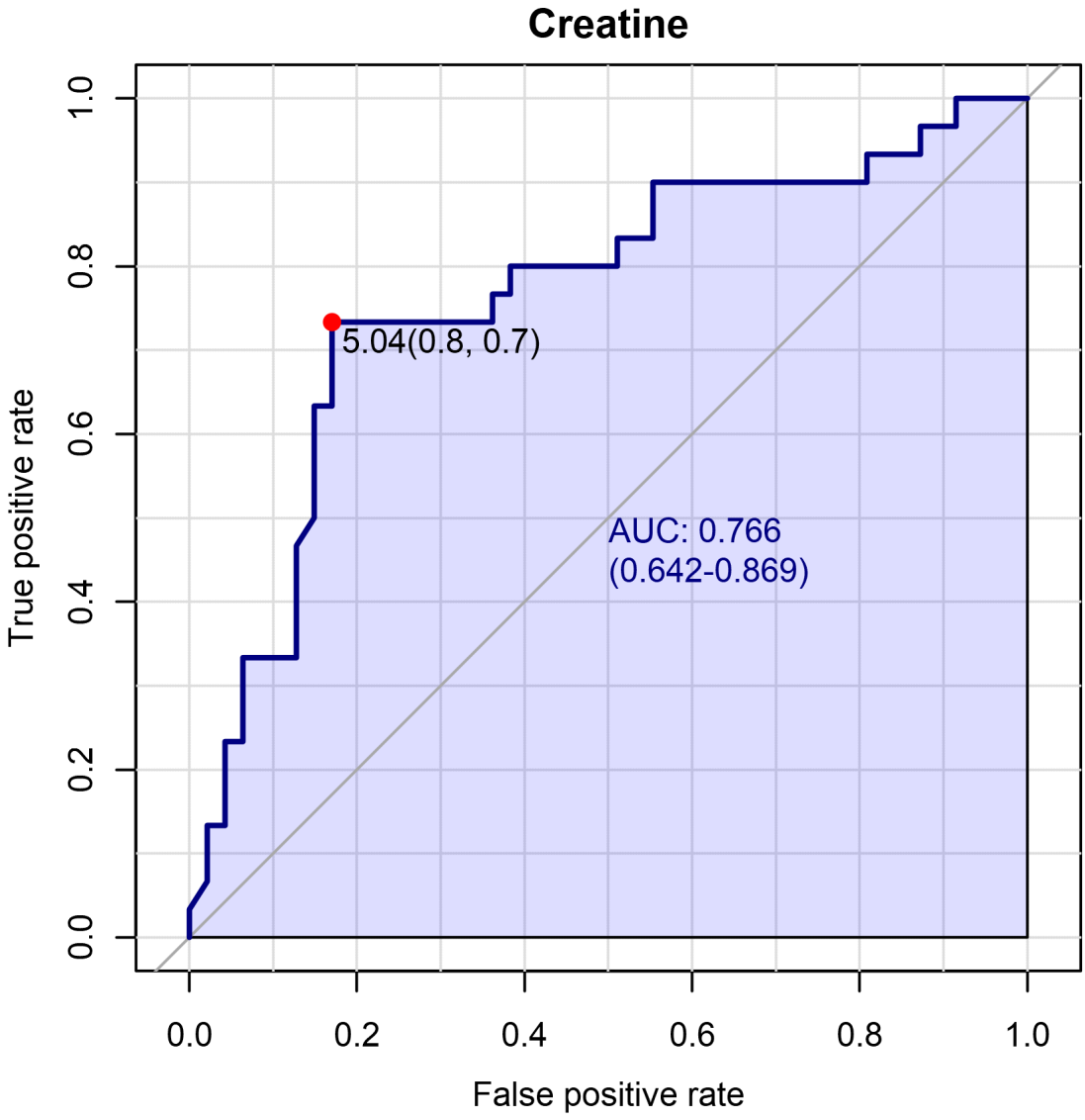

Figure 1: The ROC curve of an individual biomarker. The sensitivity is on the y-axis, and the specificity is on the x-axis. The area-under-the-curve (AUC) is in blue. Selected individual biomarker name : Creatine

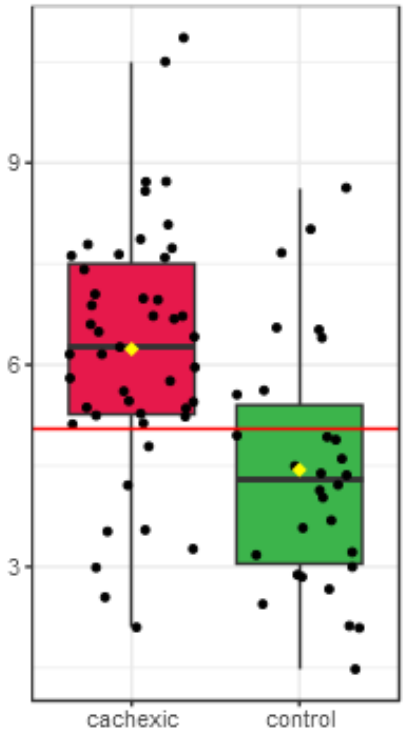

Figure 2: Box-plot of the concentrations of the selected feature between two groups within the dataset. A horizontal line is in red indicating the optimal cutoff. Selected individual biomarker name : Creatine

Table 1: AUC, Log2FC, T-test and K-Means Cluster for univariate biomarker analysis

	Area-under-the-curve	T-test	Log 2 Fold-Change	Cluster
Quinolinate	0.79	0.00	0.33	1.00
Glucose	0.79	0.00	0.31	5.00
Adipate	0.79	0.00	0.60	3.00
N,N-Dimethylglycine	0.78	0.00	0.56	3.00
Glucose/Isoleucine	0.78	0.00	0.34	1.00
Valine	0.78	0.00	0.37	1.00
Leucine	0.77	0.00	0.42	3.00
Succinate/Uracil	0.77	0.00	-99.00	2.00
3-Hydroxyisovalerate	0.77	0.00	0.63	3.00
Betaine	0.77	0.00	0.36	1.00
Uracil/cis-Aconitate	0.77	0.00	0.78	2.00
Creatine/Uracil	0.77	0.00	4.34	3.00
Creatine	0.76	0.00	0.49	1.00
Pantothenate/Succinate	0.76	0.00	-99.00	2.00
myo-Inositol	0.76	0.00	0.35	1.00
Isoleucine/Valine	0.76	0.00	0.57	2.00
Glutamine	0.76	0.00	0.27	5.00
Succinate	0.76	0.00	0.56	1.00
Alanine/Isoleucine	0.76	0.00	0.27	1.00
Methylamine	0.76	0.00	0.48	3.00
N,N-Dimethylglycine/Uracil	0.76	0.00	-3.70	2.00
Glucose/Uracil	0.75	0.00	0.62	3.00
3-Hydroxybutyrate	0.75	0.00	0.52	3.00
Glutamine/Uracil	0.75	0.00	0.52	3.00
Acetate	0.75	0.00	0.41	1.00
cis-Aconitate	0.75	0.00	0.31	5.00
Alanine	0.75	0.00	0.26	5.00
Pyroglutamate	0.75	0.00	0.23	5.00
Sucrose	0.75	0.00	0.34	1.00
Hypoxanthine/cis-Aconitate	0.75	0.00	0.87	2.00
Glutamine/Isoleucine	0.74	0.00	0.28	1.00
Xylose	0.74	0.00	0.24	1.00
Formate	0.74	0.00	0.25	5.00
Dimethylamine	0.74	0.00	0.19	5.00
Tryptophan	0.74	0.00	0.29	1.00
Creatinine	0.74	0.00	0.12	4.00
Isoleucine/Succinate	0.74	0.00	1.03	2.00
Uracil/Valine	0.73	0.00	-99.00	2.00
4-Hydroxyphenylacetate/Glucose	0.73	0.00	1.09	2.00
Alanine/Uracil	0.73	0.00	0.52	3.00
Isoleucine/myo-Inositol	0.73	0.00	0.44	2.00
Lysine	0.73	0.00	0.24	1.00
4-Hydroxyphenylacetate/Succinate	0.73	0.00	-1.12	3.00
Threonine	0.73	0.00	0.26	1.00
Serine	0.73	0.00	0.19	5.00
Asparagine	0.72	0.00	0.25	1.00
3-Indoxylsulfate	0.72	0.00	0.19	5.00
Tyrosine	0.72	0.00	0.30	1.00
trans-Aconitate	0.72	0.00	0.30	1.00
Fumarate	0.71	0.00	0.67	3.00
Lactate	0.71	0.00	0.28	1.00
Histidine	0.71	0.00	0.24	5.00
Adipate/Uracil	0.71	0.00	-1.50	2.00
Pyruvate	0.70	0.00	0.49	3.00
2-Aminobutyrate	0.70	0.00	0.41	3.00
2-Hydroxyisobutyrate	0.70	0.00	0.24	1.00
Hippurate	0.69	0.01	0.15	4.00
Fucose	0.69	0.00	0.23	1.00
Ethanolamine	0.68	0.01	0.17	5.00
Citrate	0.68	0.00	0.15	4.00
Trigonelline	0.68	0.01	0.24	5.00
tau-Methylhistidine	0.68	0.00	0.23	1.00
Trimethylamine N-oxide	0.68	0.01	0.15	5.00
4-Hydroxyphenylacetate	0.67	0.03	0.14	1.00
O-Acetylcarnitine	0.66	0.01	0.46	3.00
pi-Methylhistidine	0.66	0.03	0.19	5.00
Guanidoacetate	0.66	0.03	0.17	1.00
Glycine	0.65	0.01	0.15	5.00
1,6-Anhydro-beta-D-glucose	0.65	0.03	0.20	1.00
Glycolate	0.65	0.03	0.18	5.00
Carnitine	0.65	0.04	0.24	1.00
Taurine	0.64	0.02	0.18	5.00
1-Methylnicotinamide	0.64	0.06	0.19	1.00
2-Oxoglutarate	0.64	0.04	0.23	1.00
Isoleucine	0.63	0.05	0.25	3.00
Pantothenate	0.62	0.22	0.15	1.00
Tartrate	0.61	0.19	0.19	3.00
Hypoxanthine	0.61	0.09	0.17	1.00
3-Aminoisobutyrate	0.59	0.18	0.18	1.00
Methylguanidine	0.57	0.24	0.18	3.00
Uracil	0.57	0.30	0.10	1.00
Acetone	0.54	0.43	0.09	3.00

5 Multivariate ROC curve exploration

The aim of the multivariate exploratory ROC curve analysis is to evaluate the performance of biomarker models created through automated feature selection. MetaboAnalyst currently supports three multivariate algorithms: partial least squares discriminant analysis (PLS-DA), random forests (RF), and support vector machines (SVM). To begin, ROC curves are generated by Monte-Carlo cross validation (MCCV) using balanced subsampling. In each MCCV, 2/3 of the samples are used to evaluate feature importance, and the remaining 1/3 are used to validate the models created in the first step. The top ranking features (max top 100) in terms of importance are used to build the classification models. The process is repeated several times to calculate the performance and confidence intervals of each model. Users must specify the classification method and the feature ranking method for ROC curve analysis. For large datasets, with more than 1000 features, the univariate feature ranking method is recommended to avoid long computation times. For the PLS-DA method, users have the option to specify the number of latent variables (LV) to use (default top 2 LVs). In the plots below, users have selected to create plots for all biomarker models, or a single biomarker model. The plot description will indicate the model selected. If it is 0, it means the plot is for all biomarker models. A -1 means it used the best model, and an input 1-6 to plot a ROC curve for one of the top six models.

Figure 3 . shows the ROC curves of all or a single biomarker model based on the average cross validation performance. Figure 4 . shows the predicted class probabilities of all samples using a selected biomarker model. Figure 5 . shows the predictive accuracy of biomarker models with an increasing number of features. Figure 6 . shows the significant features of single biomarker model ranked by importance.

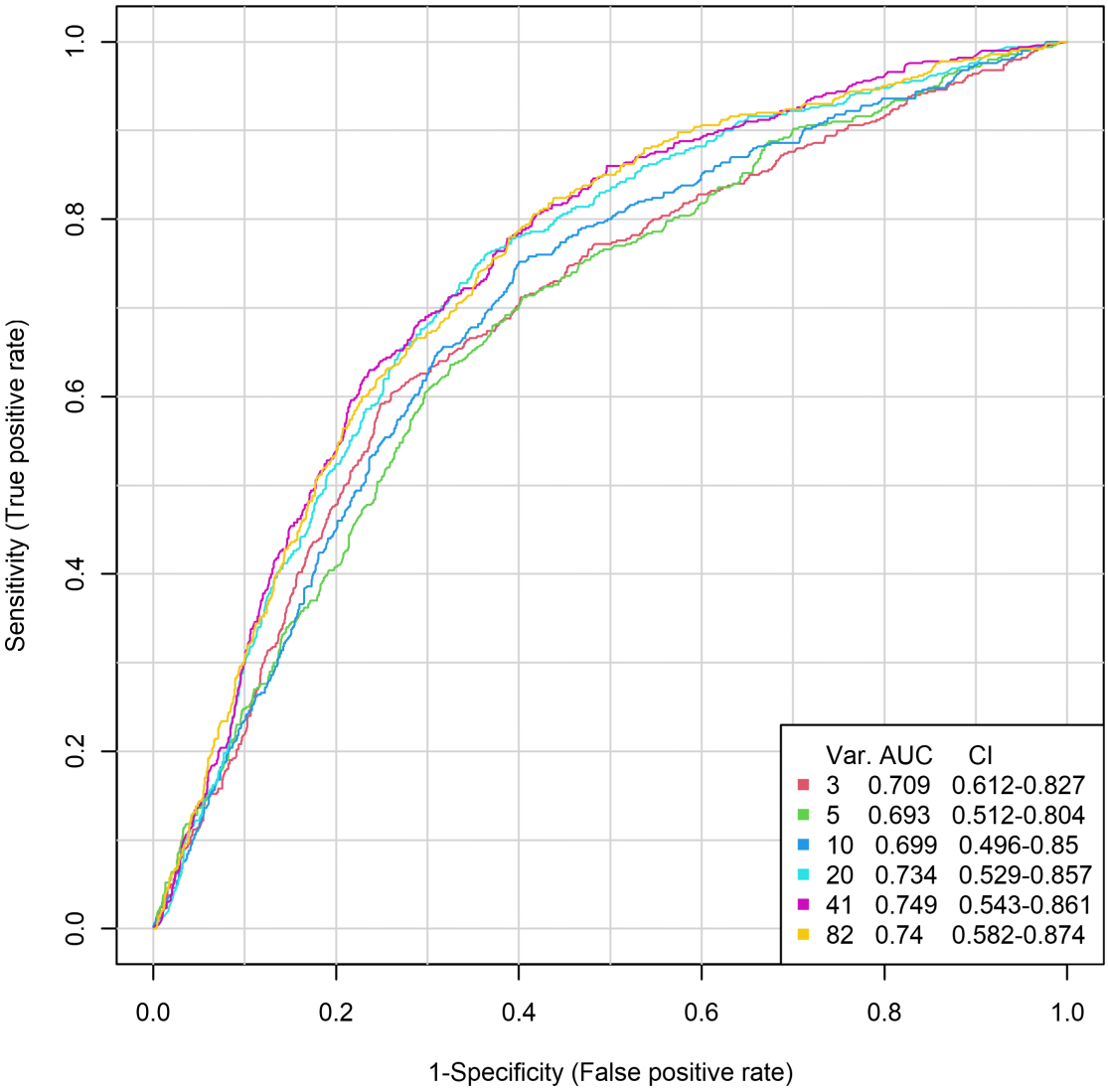

Figure 3: Plot of ROC curves for all or a single biomarker model based on its average performance across all MCCV runs. For a single biomarker, the 95 percent confidence interval can be computed and will appear as a band around the ROC curve.

Selected model : 0

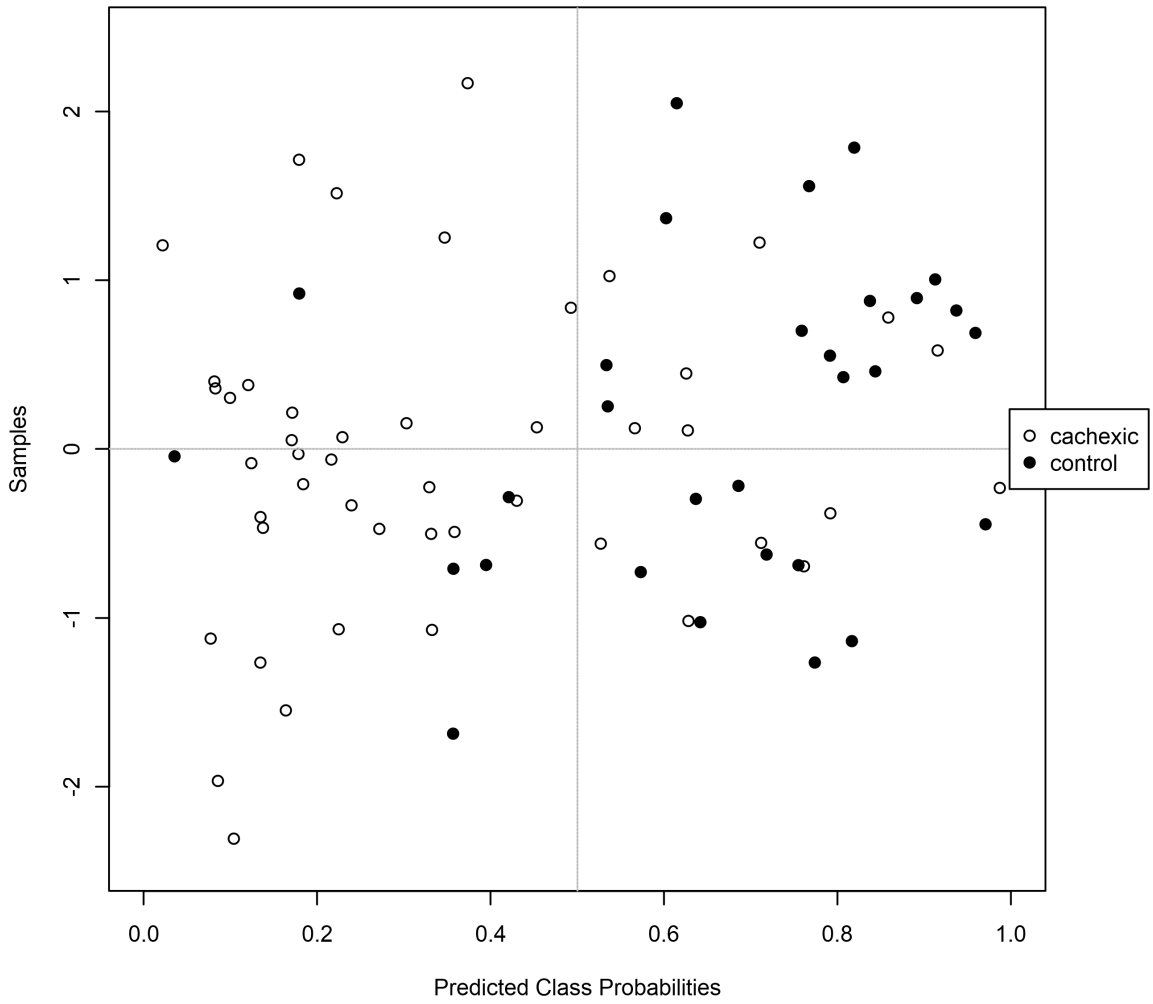

Figure 4: Plot of predicted class probabilities for all samples using a single biomarker model. Due to balanced subsampling, the classification boundary is at the center (x=0.5, dotted line).

Selected model : 5

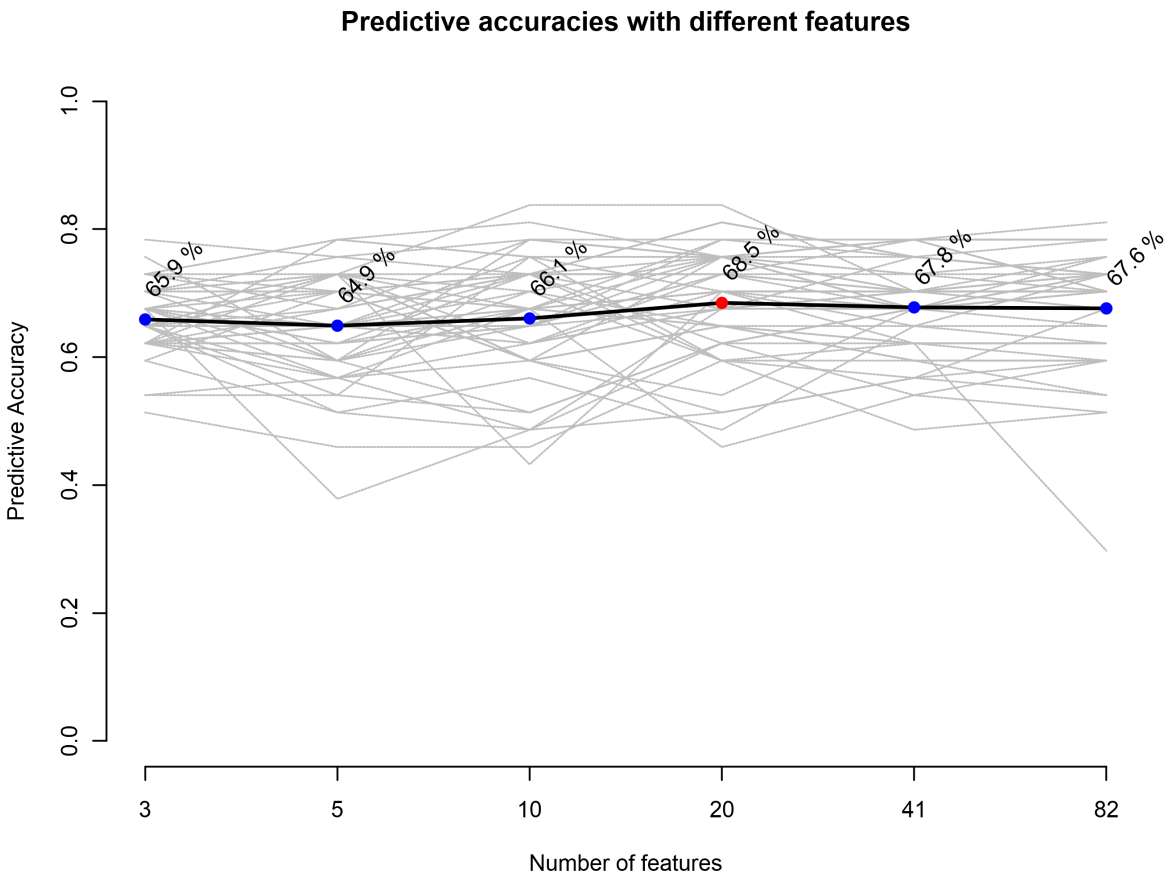

Figure 5: Plot of the predictive accuracy of biomarker models with an increasing number of features. The most accurate biomarker model will be highlighted with a red dot.

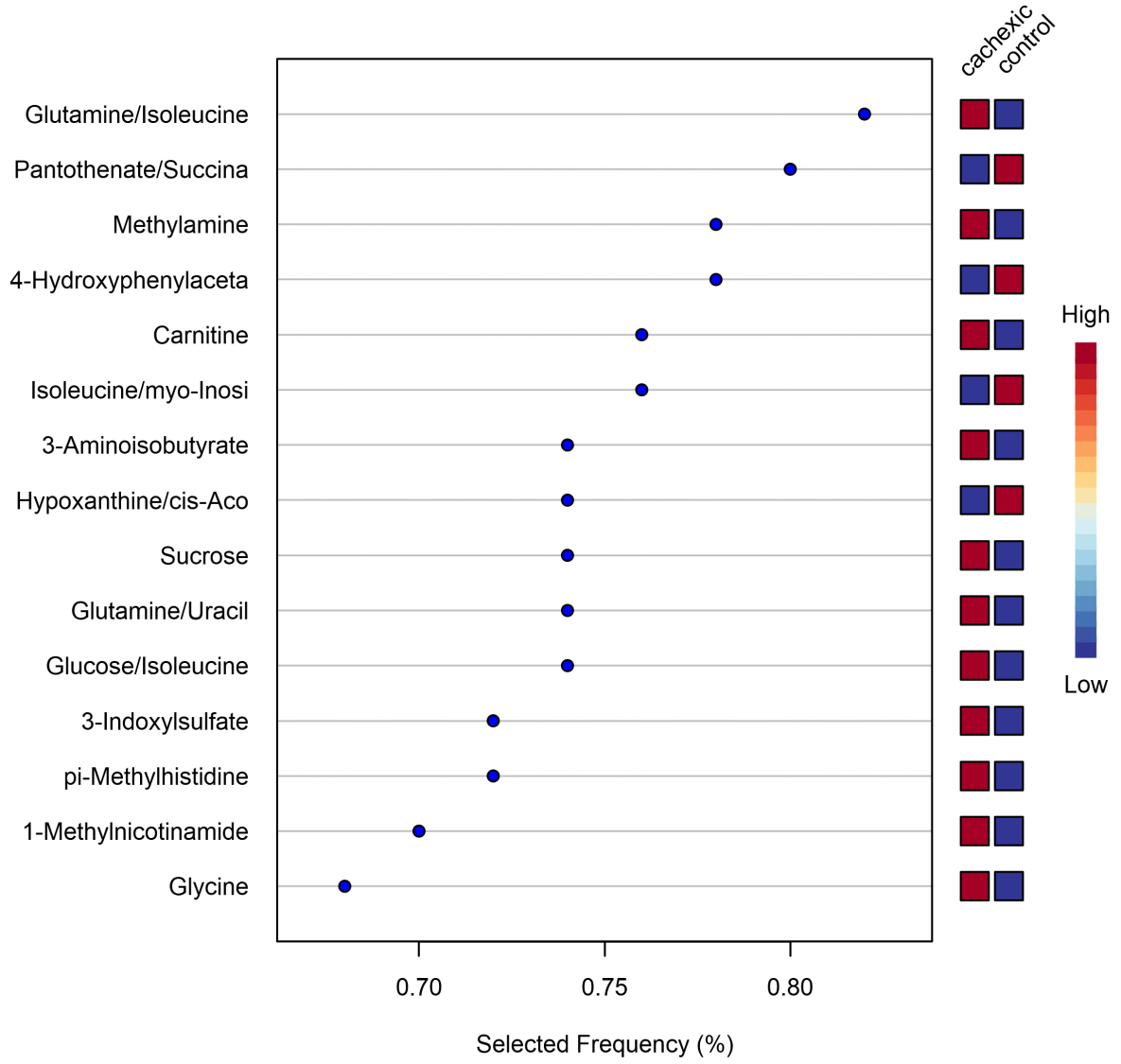

Figure 6: Plot of the most important features of a selected model ranked from most to least important. Selected model :-1

6 ROC Curve Based Model Creation and Evaluation

The aim of ROC curve based model creation and evaluation is to allow users to manually select any combination of features to create biomarker models using any of the three algorithms mentioned previously (PLS-DA, SVM, or RF). The user also has the option to withhold a subset of samples for extra validation purposes. Additionally, it allows a user to predict the class labels of new samples (unlabeled samples within the imported dataset). Features should be selected based on the user's own judgement or prior knowledge (not from the current data). Note, selection of features based on overall ranks (AUC, t-statistic, or fold-change) from current data increases the risk of overfitting. These features may be the best biomarkers for a user's own data, but not for new samples. Additionally, in order to get a decent ROC curve for validation, it is recommended that the hold-out data contains a balanced number of samples from both groups and that it contain at least 8 hold-out samples (i.e. 4 from each group).

Figure 7 . shows the ROC curve of the created biomarker model based upon its average cross validation performance. Figure 8 . shows the predicted class probabilities of all samples using the user-created classifier. Figure 9 . shows the predictive accuracy of the user-created biomarker model. Figure 10 . shows the results of the permutation tests for the user-created biomarker model.

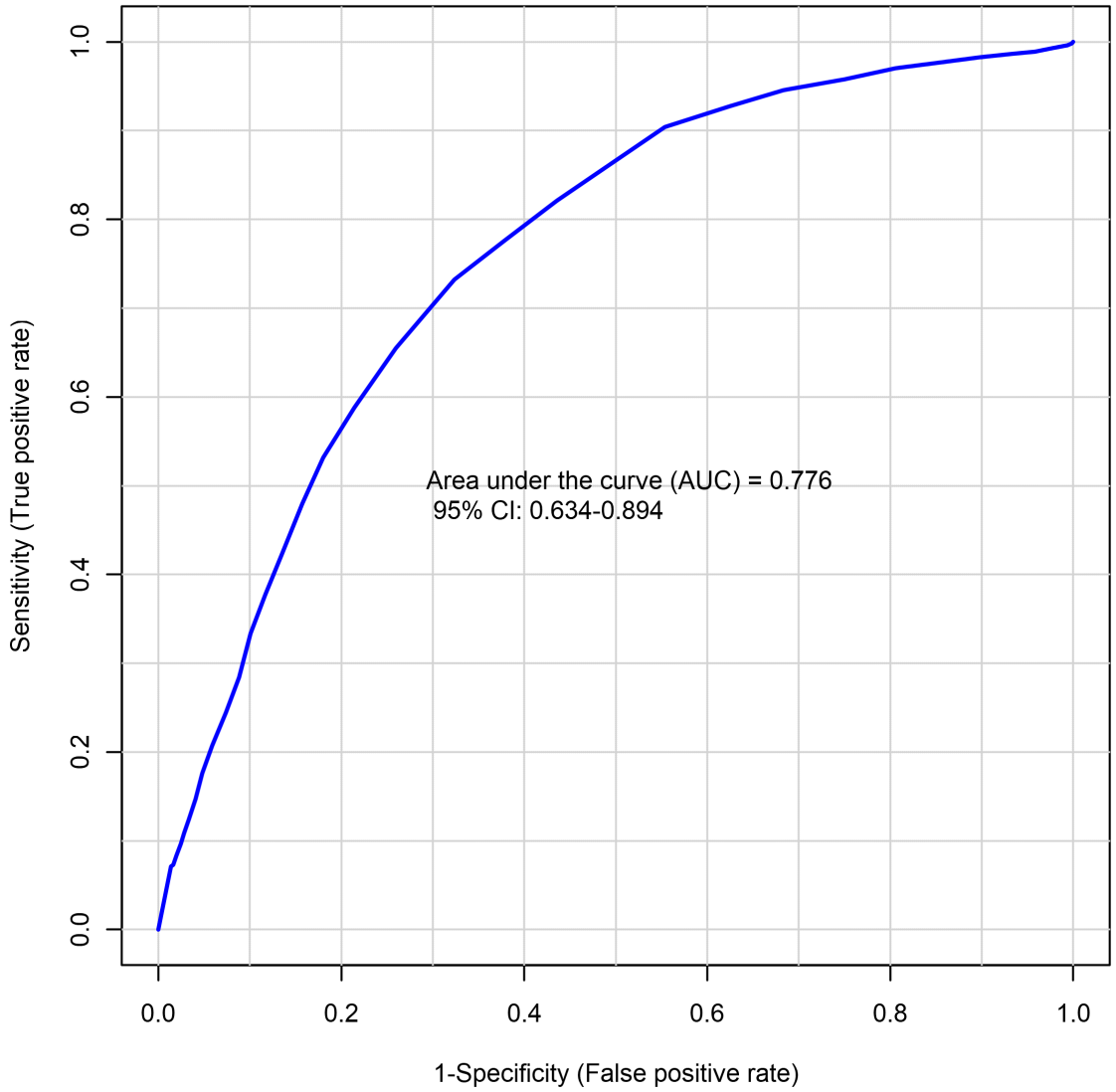

Figure 7: Plot of the ROC curve for the created biomarker model based upon its average performance across all MCCV runs. The 95 percent confidence interval can be computed.

Selected model : 0 Selected method : threshold

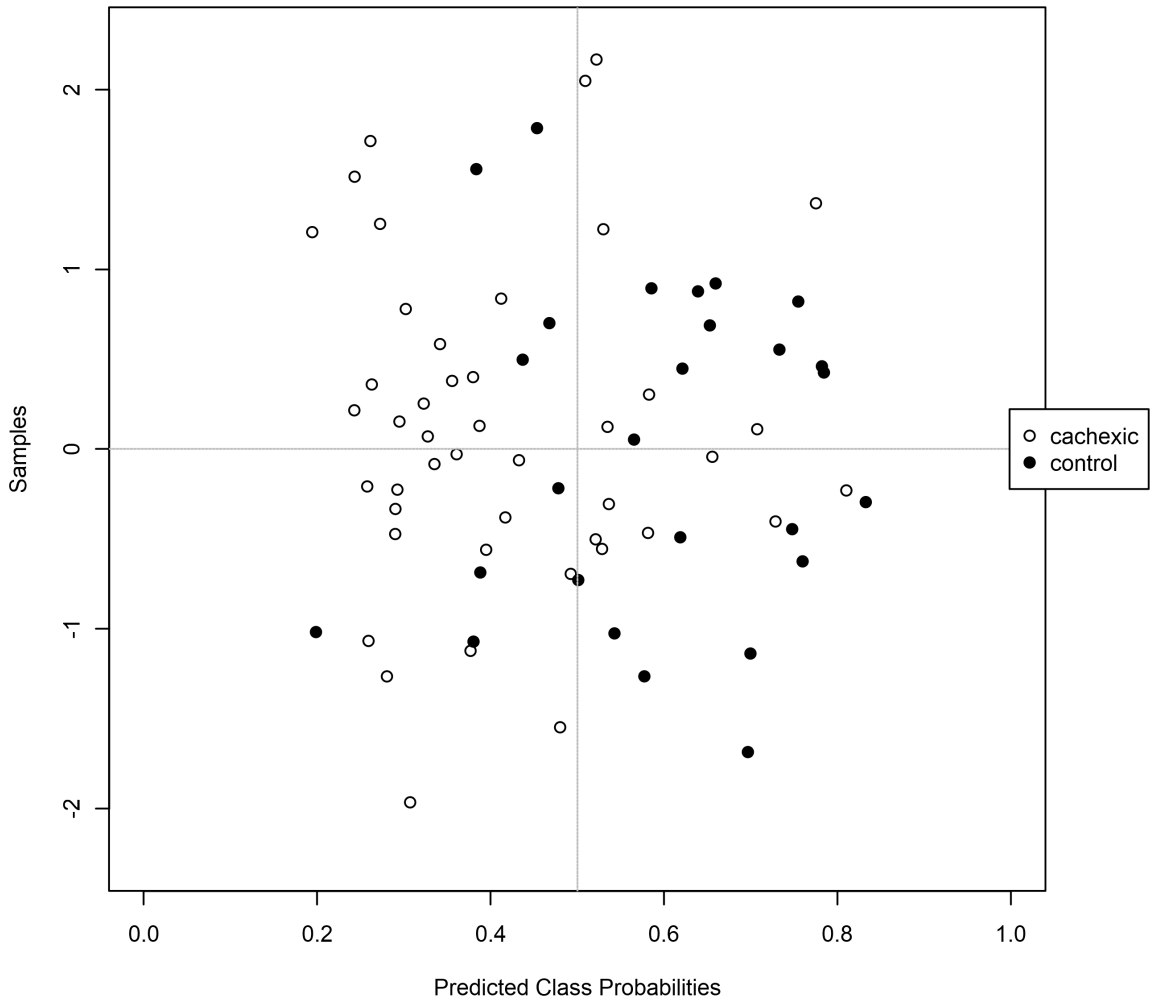

Figure 8: Plot of the predicted class probabilities for all samples using the created biomarker model. Due to balanced subsampling, the classification boundary is at the center (x=0.5, dotted line).
Selected model : -1

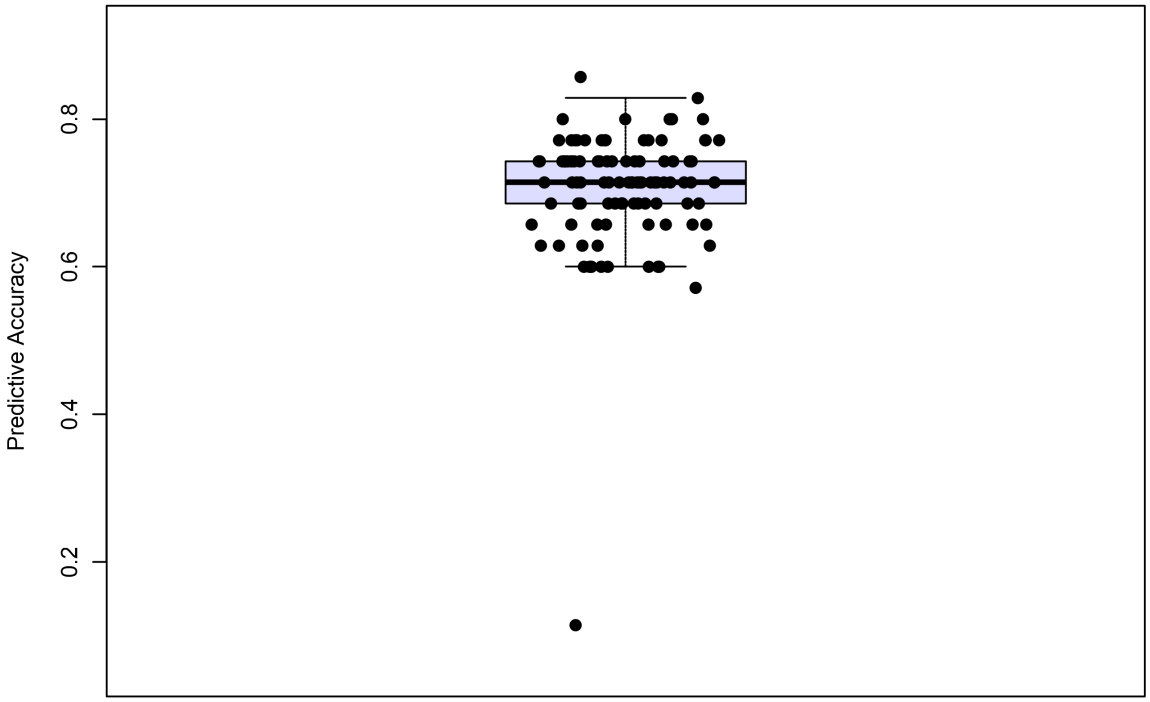

Figure 9: Box plot of the predictive accuracy of the created biomarker model.

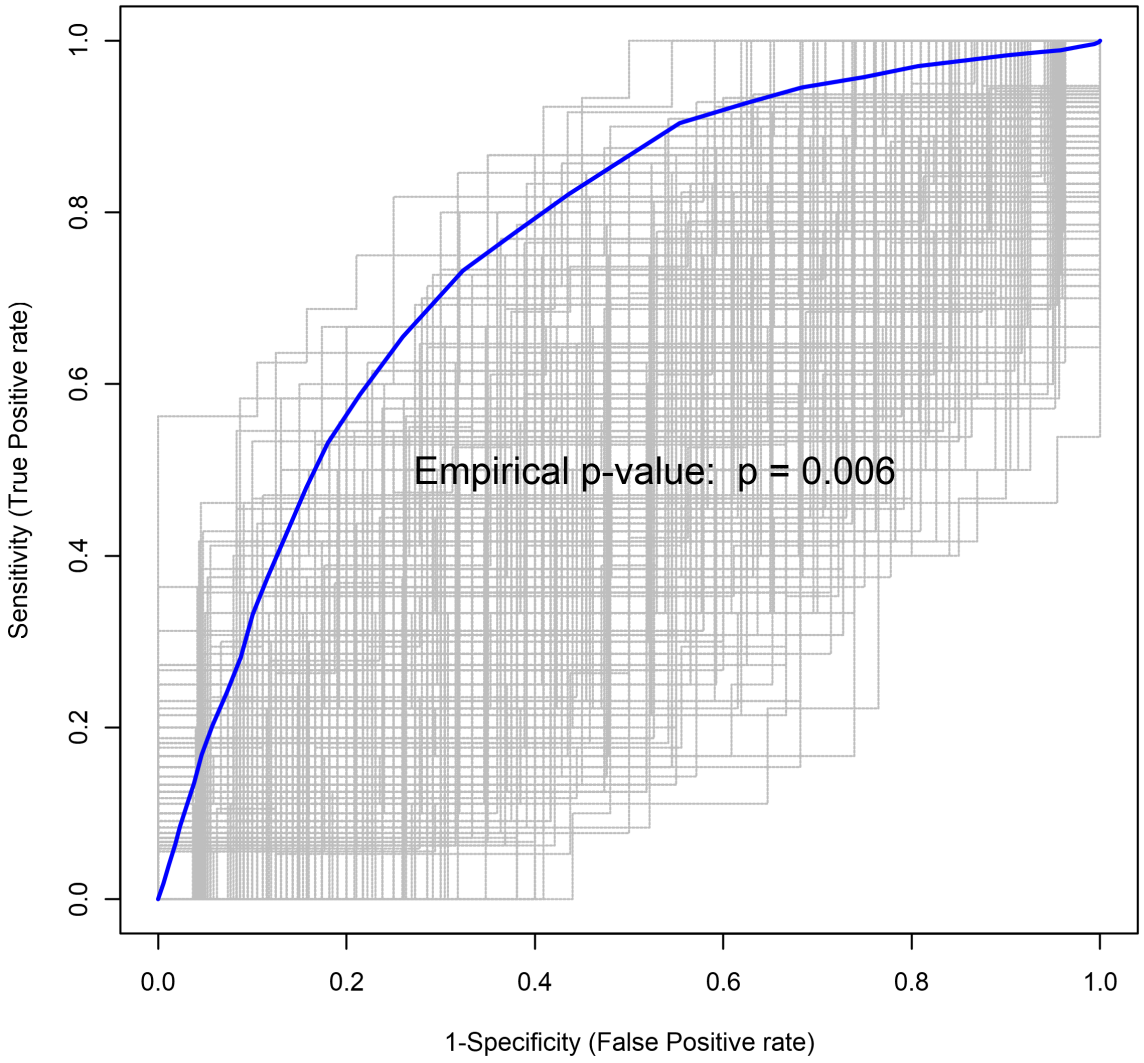

Figure 10: Plot of the permutations tests using the area under the ROC curve or the predictive accuracy of the model as a measure of performance. The plot shows the AUC of all permutations, highlighting the actual observed AUC in blue, along with showing the empirical p-value.
Selected permutation method : auroc

Table 2: Predicted class labels with probabilities for new samples

Probability	Class Label

7 Appendix: R Command History

```
[1] "No commands found"
```

The report was generated on Sat Feb 18 01:27:04 2023 with R version 4.1.1 (2021-08-10), OS system: Windows, version: build 22000 .