RESEARCH

Paper 1 in applied machine learning

Cuong Gia Pham, Aakash Nepal and Bilal Ashraf

Full list of author information is available at the end of the article
*Equal contributor

Abstract

Goal of the project: Replicate the paper's results for eight different machine-learning (ML) algorithms on one data set and extract important features(i.e. Metabolites).

Main results of the project: The performance of three different ML algorithms were evaluated and compared with previous study. Some important metabolites were extracted like cystine and lysine related to the dataset.

Personal key learning :

1 Pham Gia Cuong: Creating conda environment from GitHub repository, methods to estimate the accuracy of ML models, setter, and getter in Python.
2 Aakash Nepal: About ML algorithms, improvement in Python, jupyter notebook, installing old packages successfully
3 Bilal: Applied two ML algorithms, PCR and PCLR, and analyzed the outcomes.

Estimation working hours:

1 Pham Gia Cuong: 7 hours per week
2 Aakash Nepal: 7 hours per week.
3 Bilal: 7 hours per week, methods: PCR and PCLR

Project evaluation: 1

Number of words : Approx. 1300(without abstract and appendix)

1 Scientific Background

Metabolomics is a kind of important field of study that focuses on improving the understanding of disease mechanisms to help develop individualized medical treatments for the disease. Research in the field of metabolomics focuses mainly on the identification and quantification of metabolites found in biological samples to better understand the metabolic processes in an organism. On the other hand, lots of algorithms have been developed in the field of machine learning, many of which are also popular for the analysis of such complex metabolic data. Such algorithms are helpful for bio-marker identification and the development of predictive models for disease treatment and diagnosis. Machine learning algorithms are subjected to have impact with many variety of factors, including the algorithm chosen and also the complexity of the data being analyzed, so it is necessary to evaluate these algorithms against novel, previously unseen data.

2 Goal of the project

The goal of this project was to replicate the results of a previous study by using a set of eight different machine learning algorithms and carefully identifying the important features of ML models. The eight machine learning methods re-evaluated here are: Partial Least Squares Regression-Discriminant Analysis (PLS-DA), Principal

Component Regression (PCR), Logistic Principal Component Regression (PCLR), Random Forest (RF), Linear Kernel Support Vector Machine (SVM-Lin), Radial-Base Kernel Support Vector Machine (SVM-RBF), and a Linear and Nonlinear Two-Layer Artificial Neural Network (ANN).

3 Data and Preprocessing

For this project, the dataset "MTBLS136" has been used which is a serum LC-MS dataset with in total of 949 metabolites. The dataset was generated using untargeted metabolomics, where a wide range of metabolites is measured in a sample without focusing on any particular metabolites. The dataset describes the amount or concentration of different metabolites between two groups; using medications that have only impact on the estrogen signaling pathway and using medications that impact on estrogen plus progesterone signaling pathway. The estrogen group has 337 samples and has been represented as a case group. The estrogen plus progesterone group has 331 samples and has been represented as a control group.

For preprocessing, the selected data was read into two tables called "DataTable" and "PeakTable" using the function " cb.utils.load_dataXL". Peaks that had more than 20 percent missings were eliminated. The names of metabolites were extracted in a "PeakList" variable. The data containing only two of the classes (0 or 1) were taken out of the "DataTable," and then the class names were extracted to create a binary Y vector. The data was then divided into test and train datasets in a ratio of 2:1. A "Xtrain" variable was created only selecting metabolites names found in "PeakList", which was later used as input features for each ML model. It was then logarithmized, scaled using "cb.utils.scale" function, and the missings were filled using k-nearest neighbor imputation with k = 3.

4 Methods

4.1 Principal Component Regression

Principal Component Regression (PCR) is a regression technique that serves the same goal as standard linear regression models the relationship between a target variable and the predictor variables. However, not like linear regression, in PCR the training data is rotated and projected into a lower dimensional space also called the principal component. And those principal components are used as the predictor variables for regression analysis. PCR models have a single-tuning hyperparameter; the number of principal components. Hence, in this project, the different numbers of principal components are used to measure the performance of this model.

4.2 Linear Kernel Support Vector Machine

The main goal of the Linear Kernel Support Vector Machine (SVM-Lin) is to find a hyperplane (from a set of hyperplanes) with a maximal margin in N-dimensional space that distinctly classifies the dataset, where N is the number of features. SVM-Lin has only one regularization parameter which is a hyperparameter (in Python C for "Cost").

4.3 Random Forest

Unlike other machine learning algorithms used in this project, the Random Forest (RF) uses multiple unique base classifiers called decision trees for training and averaging them to generate a single prediction. For this project, most of the RF hyperparameters(such as number of trees and number of features sampled during training) were not changed as they had little impact on performance. The only hyperparameters adjusted were the maximum tree depth and the minimum number of samples classified at each leaf node during training.

4.4 Cross validation

First, a parameter dictionary was created with values of hyperparameters according to the model used. Then, a 5-fold cross-validation was initialized using the cb.cross_val.KFold function with the selected model, the input data (XTrainKnn), the target variable (YTrain), the parameter dictionary, the number of folds (5) and Monte Carlo simulations (10). After that, cross-validation was executed to compute performance metrics (AUC and R2Q2) and then, the results were visualized.

4.5 Evaluation

To evaluate, the study uses AUC value of the optimized hyperparameters model to infer the precision of the algorithm on the dataset. There was however a possibility of biased results due to smaller sample sizes and hence, in quantifying the unreliability of these models- we used bootstrap method to estimate confidence intervals of both the training and test evaluation metrics. Preparing for this process involved logarithmic and scaling on non-bootstrapped datasets as well as imputation of any missing values using k-nearest neighbors algorithm before being trained on a bootstrap dataset.

5 Results and Discussion

All results shown in this Chapter are acquired by processing and analyzing SVM-Lin, RF and PCR only due to word limit. Other Machine learning models show similar outcomes regarding the evaluation. (see. Appendix). PLS-DA was skipped because it had some error due to unupdated/old library in our local machines.

Figure 1 illustrates the hyperparameter optimization done during the cross-validation step. $|R^2 - Q^2|$ vs $|Q^2|$ and $|R^2|/|Q^2|$ plot were both used to select the optimal hyperparameter by selecting the point of inflection of the $|R^2 - Q^2|$ vs $|Q^2|$ in its outer convex hull (also called elbow method). Nevertheless, many of the optimal hyperparameters used from the previous paper[1] no longer seemed applicable during this re-evaluation because the inflection in the paper was chosen 0.005, but we see here the inflection at 0.004. It could be that five-fold cross-validation has been done differently or due to difference in the library version used.

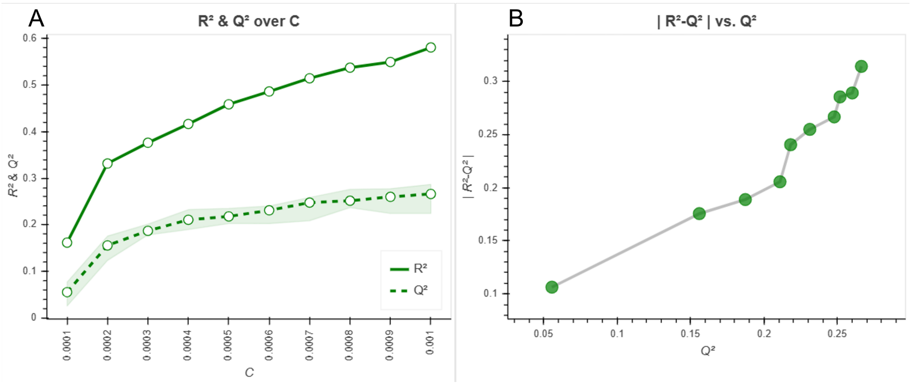

Figure 1 Hyperparameter optimization: a) An example of $|R^2|/|Q^2|$ plot over the number of components C for hyperparameter optimization in PCR model. b) $|R^2 - Q^2|$ vs $|Q^2|$ plot used for optimisation of hyperparameter.

From Figure 2 we can see that, the ROC on the test set is reduced quite a lot in comparison to the ROC on the training set in all algorithms. It shows that every model is overfitted and the more simple the ML method the more severe the overfitted. Besides that, in comparison to the original results from the paper, we also recognize that the ROC of our replicated study has small differences. It could be that the machine error has affected the result.

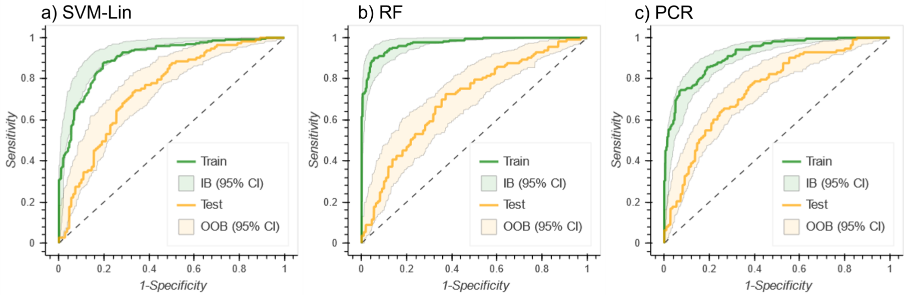

Figure 2 The ROC curves depicting the performance of three different machine learning algorithms (a)SVM-Lin, b) RF, and c) PCR) in predicting test results. The green line with shading around it represents the in-bag 95% confidence interval for the ROCtrain and the yellow line with shading represents the out-of-bag 95% confidence interval for the ROCtest. The AUCtest values for the three algorithms are 0.75 (a and b) and 0.7 (c).

The coefficients of each variable from each model were analyzed to get the top four metabolites that have the most impact on estrogen or estrogen plus progesterone signaling pathway. The higher the coefficient value , the higher the impact of a metabolite. From Table 1, it can be seen that each model has identified many of different top metabolites(M). However, there are still some overlaps between these metabolites, for example, 1-linolenoyl-GPA (18:2)* . Moreover, we found some evidence for cystine, lysine, 1-linolenoylglycerol, and 2-aminoheptanoate that they have a significant association between estrogen only and estrogen plus progesterone users in postmenopausal women from the paper of the dataset [2]. For example,

Table 1 Table showing 4 metabolites(M) found in 3 Machine Learning algorithms

M	PCR	SVM-Lin	RF
1	cystine	1-linoleoyl-GPA(18:2)*	1-stearoyl-2-arachidonoyl-GPC(18:0/20:4)
2	isobutyrylcarnitine(C4)	lysine	1-linoleoyl-GPA (18:2)*
3	succinimide	2-aminoheptanoate	1-linoleoyl-2-arachidonoyl-GPC(18:2/20:4n6)*
4	gluconate	4-acetamidobutanoate	1-linolenoylglycerol (18:3)

cystine (an amino acid) is also associated with colorectal cancer and cardiovascular disease, which may be due to postmenopausal hormone use in women. Another related example is lysine (also an amino acid), which declines during women during pregnancy but has also been linked to colorectal cancer [3]. Thus, we can conclude that these metabolites may have affected the estrogen or estrogen and progesterone signaling pathway. However, we still need biological validation to make this result more certain.

In overall, the study could be replicated and significant features could be discovered which demonstrates the effectiveness of machine learning algorithms in detecting molecular signatures. However, it is difficult and more accurate investigation is required.

6 Contributions

1 Pham Gia Cuong : Overall 48% report and ran all algorithms.
2 Nepal Aakash: Overall 48% report and ran all algorithms.
3 Bilal Ashraf:Overall 4%

References

[1] Kevin M Mendez, Stacey N Reinke, and David I Broadhurst. “A comparative evaluation of the generalised predictive ability of eight machine learning algorithms across ten clinical metabolomics data sets for binary classification”. In: *Metabolomics* 15.12 (2019), p. 150. DOI: 10.1007/s11306-019-1612-4.

[2] Victoria L Stevens et al. “Serum metabolomic profiles associated with postmenopausal hormone use”. In: *Metabolomics* 14.7 (2018), p. 97. DOI: 10.1007/s11306-018-1393-1.

[3] The Human Metabolome Database. *HMDB0000182*. https://hmdb.ca/metabolites/HMDB0000182. [Online; accessed 4-May-2023]. 2021.

Appendix A: About more ML models:

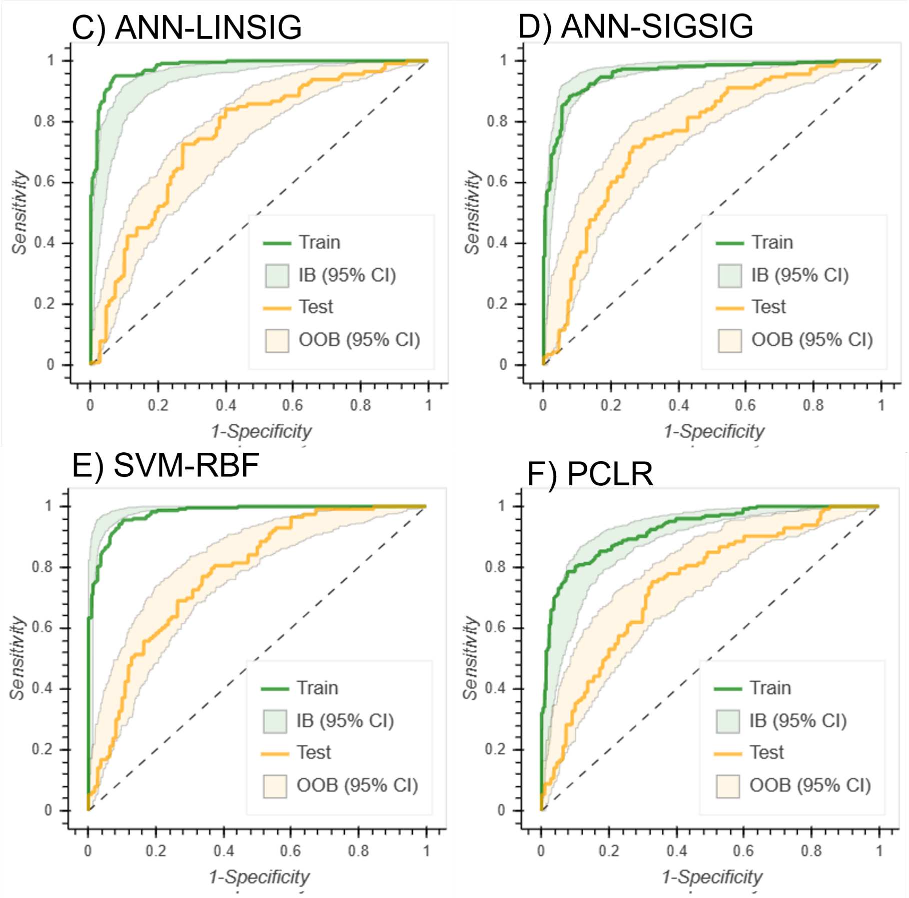

Figure 3 The ROC curves depicting the performance of four different machine learning algorithms (c)ANN-LINSIG, d) ANN-SIGSIG, e)SVM-RBF and f)PCLR) in predicting test results for MTBLS136. The green line with shading around it represents the in-bag 95% confidence interval for the ROCtrain and the yellow line with shading represents the out-of-bag 95% confidence interval for the ROCtest. The AUCtest values for the three algorithms are 0.74(f),0.75(c), 0.76 (d), 0.78(e).