Computers in Biology and Medicine 159 (2023) 106867

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

Methodology to identify a gene expression signature by merging microarray datasets☆

Olga Fajarda [a,*], João Rafael Almeida [a,b], Sara Duarte-Pereira [a,c], Raquel M. Silva [d], José Luís Oliveira [a,*]

[a] DETI/IEETA, LASI, University of Aveiro, Aveiro, Portugal
[b] Department of Computation, University of A Coruña, A Coruña, Spain
[c] Department of Medical Sciences and iBiMED-Institute of Biomedicine, University of Aveiro, Aveiro, Portugal
[d] Universidade Católica Portuguesa, Faculty of Dental Medicine (FMD), Center for Interdisciplinary Research in Health (CIIS), Viseu, Portugal

ARTICLE INFO

Keywords:
Microarray data
Gene expression signature
Random forest
LSVM
Neural network
Heart failure
Autism spectrum disorder

ABSTRACT

A vast number of microarray datasets have been produced as a way to identify differentially expressed genes and gene expression signatures. A better understanding of these biological processes can help in the diagnosis and prognosis of diseases, as well as in the therapeutic response to drugs. However, most of the available datasets are composed of a reduced number of samples, leading to low statistical, predictive and generalization power. One way to overcome this problem is by merging several microarray datasets into a single dataset, which is typically a challenging task. Statistical methods or supervised machine learning algorithms are usually used to determine gene expression signatures. Nevertheless, statistical methods require an arbitrary threshold to be defined, and supervised machine learning methods can be ineffective when applied to high-dimensional datasets like microarrays. We propose a methodology to identify gene expression signatures by merging microarray datasets. This methodology uses statistical methods to obtain several sets of differentially expressed genes and uses supervised machine learning algorithms to select the gene expression signature. This methodology was validated using two distinct research applications: one using heart failure and the other using autism spectrum disorder microarray datasets. For the first, we obtained a gene expression signature composed of 117 genes, with a classification accuracy of approximately 98%. For the second use case, we obtained a gene expression signature composed of 79 genes, with a classification accuracy of approximately 82%. This methodology was implemented in R language and is available, under the MIT licence, at https://github.com/bioinformatics-ua/MicroGES.

1. Introduction

Microarray technology transformed the field of molecular biology by enabling the measurement of thousands of gene expression levels simultaneously [1,2]. Along with the development of this technology, the requirement of scientific publishers and funding agencies that all experimental data should be publicly available [3], led to the creation of public repositories, such as the Gene Expression Omnibus (GEO) [4] and the ArrayExpress [5], where nowadays a vast amount of microarray data is available for re-use.

Microarray data are typically used to identify differentially expressed genes (DEGs) and gene expression signatures (GESs). DEGs are genes whose expression is significantly different in samples from distinct conditions. A GES is a set of differentially expressed genes that can differentiate distinct conditions and can be used for diagnosis, prognosis and therapeutic response [6]. Furthermore, GESs can be used for drug discovery by identifying new potential targets [7].

Due to the cost of acquiring microarray chips, most datasets have a reduced number of samples and therefore low statistical, predictive and generalization power [8]. Michiels et al. [9] reanalyzed seven studies that aimed to identify a prognostic GES of cancer. They concluded that the list of predictors was strongly dependent on the used dataset, and that the different studies share only a small set of prediction genes.

☆ This work has received support from the FCT — Foundation for Science and Technology, Portugal (national funds) within project DSAIPA/AI/0088/2020. SDP and JRA are funded by the FCT — Foundation for Science and Technology, Portugal under the grants SFRH/BD/108890/2015 and SFRH/BD/147837/2019, respectively.

* Corresponding authors.
E-mail addresses: olga.oliveira@ua.pt (O. Fajarda), joao.rafael.almeida@ua.pt (J.R. Almeida), sdp@ua.pt (S. Duarte-Pereira), rmsilva@ucp.pt (R.M. Silva), jlo@ua.pt (J.L. Oliveira).

https://doi.org/10.1016/j.compbiomed.2023.106867
Received 19 October 2022; Received in revised form 1 March 2023; Accepted 30 March 2023
Available online 11 April 2023

They also observed that by increasing the number of patients in a dataset the misclassification decreased, as intuitively expected.

One can increase the number of samples by merging microarray datasets from independent studies [10]. However, this is a challenging task, due to the several platforms that can be used to measure the gene expressions, and that generate incompatible datasets. Moreover, the use of different experimental protocols, microarray platforms, and processing methods introduces non-biological variations to the data [11], known as batch effect, and they can lead to inaccurate findings. To address this problem, we can use several batch effect removal methods, but despite their advantages they also present some weaknesses [12].

The methodologies generally used to identify GESs are statistical methods and supervised machine learning (ML) algorithms. The drawback of statistical methods is the fact that a cut-off must be defined from which the genes are considered to be differentially expressed and there is no consensus on the values to use as a threshold. The drawback of using supervised ML algorithms is the fact that these are usually ineffective when applied to high-dimensional datasets [13].

In this paper, we propose a methodology to identify GES from multiple microarray datasets, by merging them before processing the data. It uses statistical methods and diverse thresholds to obtain several reduced sets of genes. These are then analyzed using a supervised machine learning algorithm that determines the GES based on the highest classification accuracy. We demonstrate the effectiveness and generalizability of this methodology by applying it to two different research applications: (i) heart failure, and (ii) autism spectrum disorder. In the heart failure case, we merged four publicly available microarray datasets and identified a GES consisting of 117 genes with a classification accuracy of approximately 98%. Similarly, for autism spectrum disorder, we obtained a GES of 79 genes with a classification accuracy of approximately 82%. The proposed methodology provided relevant results, by being capable of merging microarray datasets. Based on these promising results, we believe that this strategy can be used to merge microarray datasets from distinct institutions.

2. Background

The integration of different microarray studies can be done in two different ways: by merging the datasets of the different studies before conducting the study, or by combining the individual results of the different studies (meta-analysis) [2]. From these two strategies, it has been argued that merging microarray studies is more powerful for identifying robust biomarkers than meta-analysis [14]. Taminau et al. [15] compared meta-analysis and data merging for the identification of cancer-related biomarkers and obtained substantially more genes differentially expressed using data merging than using meta-analysis. However, most researchers use meta-analysis to integrate different microarray studies. Tseng et al. [16] conducted a systematic review of 191 papers which combine multiple microarray studies, and concluded that only 27% of them merged the different datasets.

Over the years, several tools were developed to merge microarray studies [17–21]. However, most of these tools only merge microarray studies and do not obtain GESs. To our knowledge, the only toolbox still available to merge microarray studies and obtain DEGs is the R package developed by Johannes Vey et al. [20]. However, this toolbox only enables the merging of microarray datasets obtained using platform GPL570, Affymetrix Human Genome U133 Plus 2.0 Array.

GESs are usually obtained using statistical methods, which are based on fold change (FC) and statistical hypotheses test, mostly the *t*-test and variation of this test [22]. The R package `limma` [23], which implements empirical Bayes methods and linear models, is also commonly used to obtain GESs. Chrominski and Tkacz [24] tested six different methods to detect DEGs, while Jeanmougin et al. [25] compared eight methods. In both studies, the method implemented in `limma` was found to be one of the best to detect DEGs. These statistical methods require the selection of a statistical hypothesis test, the definition of a decision rule and the control of false discovery rate [26]. The decision rule and the control of false discovery rate generally rely on arbitrary thresholds [27] which can be very distinct from study to study.

Besides statistical methods, supervised machine learning (ML) algorithms are also used to obtain GESs [28–30]. Supervised ML algorithms construct prediction models which can be used to classify new samples. However, these models can be inefficient when applied to microarray datasets because these are composed of thousands of genes (features) and only a minimal number of samples [13]. Therefore, to minimize this unbalance, several feature selection methods can be applied to the microarray dataset, namely filter, wrapper and embedded methods [31]. Wrapper and embedded methods rely on learning methods and therefore have an expensive computational cost [32]. Filter methods only rely on the characteristics of the data and are computationally efficient and independent of a learning algorithm [33]. However, filter methods present generally worse performance results than wrapper and embedded methods because they do not interact with the learning algorithm [34].

RNA-Sequencing (RNA-seq) is another technology used to measure gene expression that has emerged in the last decade, with the advent of next-generation sequencing. However, microarray chips continue to be widely used because they are cheaper than RNA-seq technology [35]. In the last five years, more than twenty-one thousand microarray datasets have been published in the GEO repository.

In recent years, several computational tools have been developed to merge metabolomic data and identify metabolic signatures [36–41]. These tools use methods for batch effect correction and biomarker identification that can be adapted also for microarray data. One can also find several web-based tools have been developed to assess the performance of various methods used in metaproteomics studies, including normalization and biomarker discovery [42–45].

3. Methods

The proposed methodology enables the merging of multiple microarray datasets and the identification of a gene expression signature. The process, illustrated in Fig. 1, consists of several steps.

First, we perform data preprocessing, which involves cleaning and formatting the data to ensure that it is suitable for analysis. This may involve filtering out low-quality data points, normalizing the data to account for technical variations, and selecting a subset of genes to focus on.

In the next stage, it performs feature selection, which involves selecting a subset of genes from the preprocessed data that are most relevant to the classification task. There are several methods that can be used for feature selection, including filter methods, wrapper methods, and embedded methods. The choice of method depends on the nature of the data and the specific requirements of the classification task.

Finally, it uses a supervised machine learning algorithm to classify individuals based on their gene expression levels. There are many different algorithms that can be used for this purpose, including decision trees, support vector machines, and neural networks. The choice of algorithm depends on the characteristics of the data. In this section, we describe in more detail all the steps of the methodology.

3.1. Selection of the datasets

Several filter criteria can be considered in order to select the datasets suitable for a study. Some of them may be:

- the platform used to obtain the expressions of a dataset since several platforms have few genes in common;
- the number of samples in a dataset, since datasets with few samples have low statistical power and only datasets with impact on the end-results should be merged;
- datasets with unprocessed data, in order to apply the same preprocessing to all and thus obtain comparable data.

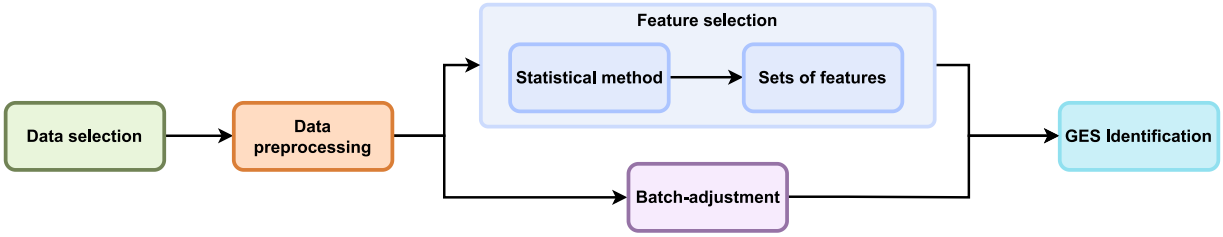

Fig. 1. The methodology pipeline to merge datasets and identify a GES.

3.2. Pre-processing the data

Data pre-processing consists of performing background correction, normalization and probe summarization, with summarization only being needed for Affymetrix arrays. Background correction is the process of removing non-specific background noise and normalization is the process of correcting systematic array bias. In Affymetrix arrays, several probes represent the same gene and therefore the different signals obtained must be summarized in one unique value [46]. Over the years, several methods and packages have been developed to pre-process microarray data. A summary of different background correction methods can be found in [47], and a good comparison of normalization methods was performed by Bolstad et al. [48].

For microarray data obtained using Affymetrix chips the commonly used pre-processing method is robust multichip average (RMA) [49], which combines background correction, normalization and summarization. This pre-processing method is implemented in the R/Bioconductor software package `oligo` [50]. The R/Bioconductor software package `limma` [23], which includes several background correction and normalization methods, is commonly used for microarrays from manufacturers other than Affymetrix. Raw data obtained using the same platform are merged and pre-processed together and whenever possible the same pre-processing method should be used to perform all the pre-processing.

In a microarray dataset, every probe corresponds to a biological sequence that can be uniquely identified by the GenBank sequence accession identifier. Different sequences can represent the same gene and therefore the probes are identified using the GenBank sequence accession identifier. Probes with no GenBank sequence accession identifier are removed from the datasets. In some microarrays, different probes corresponding to the same biological sequence have different expression measurements and therefore these probes are also removed from the datasets. Moreover, in some platforms, a probe is associated with a list of GenBank sequence accession identifiers. In this case, only lists containing unique GenBank sequence accession identifiers are maintained. Furthermore, every list is separated into individual GenBank sequence accession identifiers and the expression measurement of a list is assigned to every GenBank sequence accession identifier on that list.

The next step after removing all the conflicting probes is to identify the common GenBank sequence accession identifier across the different microarray platforms, which will be used to merge the datasets. The merged dataset is randomly divided into a training set and a test set and this procedure is repeated several times. Doing so, several training sets and several test sets are obtained. The training sets are used to obtain the feature sets and to train the supervised machine learning algorithm and the test sets are used to evaluate the performance of the machine learning algorithm.

3.3. Feature selection

Features are selected using statistical methods such as determining the fold change or determining the p-value of a statistical test. A survey of statistical methods to identify DEGs can be found in [51]. In order to select the features using statistical methods, a threshold must be defined. However, there is no consensus as to the choice of that threshold. Therefore, in our pipeline, several cut-offs are chosen, leading to the definition of several feature sets for each training set. To obtain a feature set for every cut-off, the feature sets obtained using the various training sets and corresponding to the same cut-off are intersected. Concerning the cut-off values, the ones generally used for the fold change are 1.5, and between 2 and 3 [52,53], while the values for the adjusted p-value are 0.01 and 0.05 [54].

3.4. Batch-adjustment of the data

In most cases, the datasets to be merged are obtained using different platforms, which introduces non-biological variation, i.e. batch effect, to the gene expression measurements. Larsen et al. [55] demonstrated the importance of using batch-adjustment methods when merging different microarray datasets and prior to the analysis.

To deal with the batch effect, two approaches are used. When selecting features using statistical methods, the batch variables, i.e. the platform type, are included as a covariate [56] and before determining the GES using machine learning algorithms, the data are adjusted for batch effects. A survey on batch effect adjustment methods can be found in [12]. Chen et al. [57] compared six batch adjustment methods and concluded that the Empirical Bayes method ComBat [58] exceeded the performance of the other methods. In turn, Müller et al. [59] compared several approaches to reduce the batch effect and concluded that the best approach is quantile normalization followed by ComBat.

Several packages implement batch effect adjustment methods. For example, the R/Bioconductor software package `sva` [60] implements ComBat and Frozen Surrogate Variable Analysis methods and the R package `bapred` [61] implements the following methods: FAbatch, ComBat, Frozen Surrogate Variable Analysis, mean-centering, standardization, arithmetic mean ratio-based and geometric mean ratio-based.

3.5. Identification of the gene expression signature

To evaluate the predictive accuracy of the various feature sets selected and obtain a GES, *i.e.*, the set with the best predictive accuracy, a supervised machine learning algorithm is used. A survey of machine learning algorithms can be found in [62].

The effectiveness of supervised machine learning algorithms is highly dependent on the quality and quantity of the input data, as well as the inherent characteristics of the problem being addressed [40]. Given the potential variability in these factors, it is recommended to use a diverse set of algorithms and compare their performance in order to identify the most suitable approach for a given problem. In our study, we sought to identify GES in microarray datasets using three well-established and widely used classification methods: neural networks (NN), random forests (RF), and linear support vector machines (SVMs).

These algorithms were implemented using the `caret` package, which provides a standardized and efficient framework for training and evaluating machine learning models. Neural networks, random forests, and SVMs are all highly effective approaches for classification

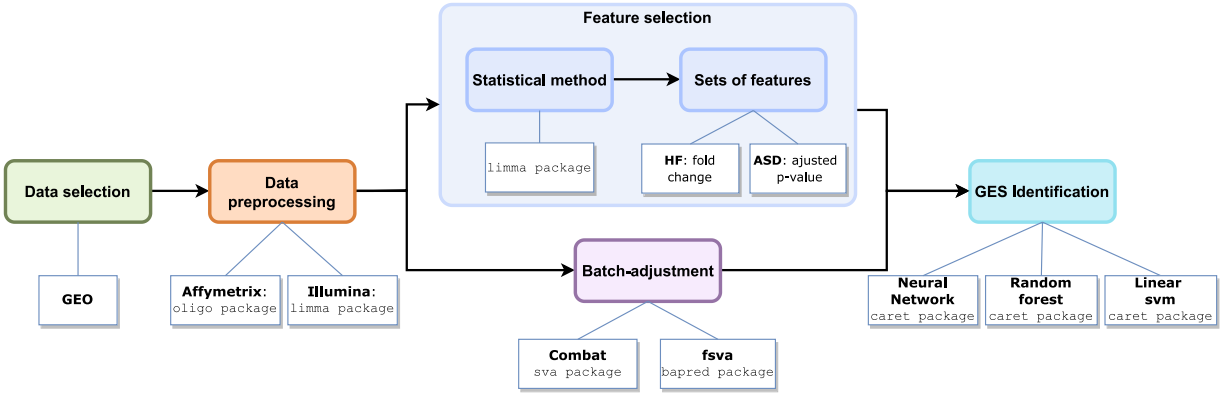

Fig. 2. The pipeline with the tools used in the two use cases.

tasks and have demonstrated strong performance on microarray data in previous research [63,64]. By using multiple algorithms, we aimed to provide a more comprehensive evaluation of GES identification and to increase the robustness of our results. Our approach of leveraging multiple algorithms allowed us to account for the potential influence of data quality and problem-specific characteristics on the identification of GES in microarray datasets.

The R package `caret` [65] implements a large number of machine learning algorithms that can be used to create predictive models. Moreover, this package provides a range of functions to perform various tasks inherent to building a predictive model, namely model training and data splitting, among others.

Every machine learning algorithm has a set of parameters that are fine-tuned to maximize accuracy. To select the best parameters for every set of features, the training sets and repeated *k*-fold cross-validation are used. The test sets are used to evaluate the performance of the model obtained. Besides accuracy, different metrics can be used to evaluate this performance, such as precision, recall, specificity, and the F1 score.

4. Results

To evaluate the methodology we used two distinct use cases: one to identify a GES in heart failure and another to identify a GES in autism spectrum disorder. In both use cases, we used the R/Bioconductor package `limma` to calculate the fold change and the adjusted *p*-value of every feature. For batch-adjust we used the ComBat and fsva methods, in both use cases. To evaluate the predictive accuracy of the feature sets we used three distinct machine learning algorithms: neural network, random forest and linear SVM (LSVM). Fig. 2 presents the pipeline with the tools used in the two use cases.

In the first use case, all data were collected using Affymetrix microarray chips, while in the second use case the data were collected using a combination of Affymetrix and Illumina microarray chips. These two types of microarray chips are commonly used in gene expression analysis and have distinct characteristics that may influence the quality and reliability of the resulting data. For both use cases, we selected publicly available datasets from GEO. To pre-process data obtained using Affymetrix platforms we used the R/Bioconductor software package `oligo`, which implements the pre-processing method RMA (Robust Multi-Array Average). For data obtained using Illumina platforms we used the `limma` package and chose for background correction the method `normexp` [66] and for normalization the quantile normalization method [48]. The methods `normexp` and quantile normalization were chosen because these are the methods used by RMA for background correction and normalization, respectively.

After pre-processing the data and merging the datasets, the merged dataset was randomly divided into a training set (75% of the samples) and a test set (the remaining 25%). Repeating the division procedure 50 times, we obtained 50 different training sets and 50 different test sets for each use case. For feature selection, we used the `limma` package to determine in each training set, for each feature (i.e. each GenBank sequence accession identifier), the fold change and the adjusted *p*-value, adjusted using Benjamini and Hochberg's method [67] to control the false discovery rate. Concerning the list of cut-offs to obtain the different feature sets, the choice was specific to each use case as presented below. For each cut-off, we obtained 50 sets of features corresponding to the 50 training sets. To obtain one feature set per cut-off we intersected the 50 sets of features.

Before the GES can be identified, the merged dataset must be batch-adjusted. To be able to compare, we decided to use two different methods for batch-adjustment. The first method used was ComBat implemented in the R package `sva` and the second one was the frozen surrogate variable analysis (fsva) method implemented in the R package `bapred`.

All three algorithms used in the methodology have parameters that were fine-tuned to maximize accuracy and using repeated 10-fold cross-validation with five repeats. Neural network has two parameters: `decay` and `size`, random forest has also two parameters: `mtry` and `ntree` and linear SVM has only one parameter: `cost`. The values used for the parameters are presented in Table 1. As can be seen in Table 1 the number of different values used for the parameters of neural network and linear SVM is constant while the number of different values used for the parameters of random forest varies with the size of the feature set.

Although the time consumption of hyperparameter tuning, training, and testing machine learning algorithms is dependent on the system used, it can be an important parameter to consider when choosing an algorithm. Therefore, in this section, we also report the time consumption of the three machine learning algorithms employed for each use case. In our evaluation, we used a server with an Intel(R) Xeon(R) CPU E5-2670 v3 operating at 2.30 GHz and with 8 GB of RAM. The system was running the Ubuntu 18.04 operating system, with Python 3.8.3 and caret 6.0.86 installed.

The R package LargeMetabo [40] was used to compare the results obtained using our methodology against other methodologies. This tool was specifically designed for processing and analyzing metabolomic data. It provides the capacity for metabolomic data integration and it offers three distinct batch-adjustment methods, including Combat which was employed in our use cases. This package also includes 13 different methods for biomarker identification and evaluates the performance of the biomarkers by calculating the area under the receiver operator characteristic curve (AUC) of an SVM model using 2-fold cross-validation. The package was adapted to use accuracy as an evaluation metric instead of the AUC, keeping the original SVN model and 2-fold cross-validation. However, some of the methods were not possible to be tested because some of the required dependencies were unavailable. Additionally, other methods have returned empty sets of biomarkers, due to the predefined threshold for selecting them.

Table 1
Parameters' values used for tuning.

Algorithm	Parameter	Values	N. of values
Neural network	`decay`	5×10^{-5}, 1×10^{-4}, 5×10^{-4}, 1×10^{-3}, 5×10^{-3}, 1×10^{-2}, 5×10^{-2}, 1×10^{-1}, 5×10^{-1}, 0.	150
	`size`	1, 2, … , 15.	
Random forest	`mtry`	2, 3, … , n, where n is the total number of features in the set.	$3\times(n-1)$
	`ntree`	100, 500, 1000.	
Linear SVM	`cost`	1×10^{-4}, 3×10^{-4}, … , 9×10^{-4}, 1×10^{-3}, 2×10^{-3}, … , 9×10^{-3}, 1×10^{-2}, 1.5×10^{-2}, … , 9.5×10^{-2}, 1×10^{-1}, 1.5×10^{-1}, … , 9.5×10^{-1}, 1, 2, … , 90, 95, 100, 125, 150, … , 300.	150

Table 2
Summary of the datasets used in the use case HF.

Dataset	Platform	Manufacturer	No. of samples (HF/control)
GSE1145 [69]	GPL570	Affymetrix	**90** (79/11)
GSE21610 [70]	GPL570	Affymetrix	**38** (30/8)
GSE22253 [71]	GPL6244	Affymetrix	**87** (0/87)
GSE57338 [72]	GPL11532	Affymetrix	**313** (177/136)
Total			**528** (286/242)

Table 3
The number of features and the number of genes for every fold change used as a threshold in use case HF.

Fold change	No. of features	No. of genes
1.50	126	117
1.75	54	50
2.00	25	23
2.25	15	13
2.50	10	10
2.75	7	7
3.00	6	6

4.1. Heart failure

Heart failure (HF) affects millions of people worldwide and its prevalence is increasing as the population ages. Furthermore, HF is related to a high risk of mortality and morbidity [68]. For this use case, we used four publicly available microarray datasets. Table 2 presents the four datasets, as well as the respective platform, manufacturer and number of samples.

For some datasets, we did not use all the data available. In relation to the dataset GSE1145, we only used the data obtained using platform GPL570 and not the seventeen obtained using platform GPL8300 since the intersection of genes between this platform and the other platforms used in this use case is very reduced. Concerning the dataset GSE21610, we did not use the 30 samples collected after implementing a ventricular assist device, since such a device can change the gene expression patterns. Regarding the dataset GSE22253, we excluded the samples which have rs1333049 genotype CC, since it was found that C is the risk allele associated with coronary heart disease which can lead to heart failure [71].

Concerning feature selection, in this use case, we chose to use the fold change to obtain the different sets of features. The fold change cut-offs used are: 1.50, 1.75, 2.00, 2.25, 2.50, 2.75 and 3.00.

For this use case, we used datasets obtained using three different Affymetrix platforms. These three platforms have 9892 GenBank sequence accession identifiers in common. So by merging the four datasets we obtained a dataset with 9892 features and 528 samples. By dividing the merged dataset we obtained 50 training sets and the correspondent 50 test sets. Each training set is composed of 9892 features and 397 samples and each test set is composed of 9892 features and 131 samples. To obtain the different feature sets we used the fold change and seven different cut-offs. Table 3 presents the number of features and the number of genes of the seven feature sets obtained.

Before using machine learning algorithms, we needed to correct the batch effects. Fig. 3 shows the multidimensional scaling (MDS) plots with the distribution of the data before and after data batch-adjustment, using ComBat and fsva. Before data batch-adjustment, three clusters can be observed, corresponding to the three platforms used. After the batch-adjustment, this platform bias was removed.

The different mean accuracies, as well as the standard deviation obtained when applying the model to the test sets, using the three machine learning algorithms and the two batch-adjustment methods, are presented in Table 4. All the results of this use case can be found in the Supplementary File `results_HD.xlsx`.

Analyzing the results we can see that the best mean accuracy was obtained using the linear SVM algorithm and the feature set obtained using a fold change of 1.50. This feature set is composed of 117 genes and has a mean accuracy of approximately 98%. Furthermore, this feature set obtained the best mean recall and mean F1 score. The 47 up-regulated genes and the 70 down-regulated genes are presented in the Supplementary File `results_HD.xlsx`. For all three supervised machine learning algorithms, the results are slightly better when using the fsva batch-adjustment method. The mean precision and the mean recall are mostly higher than the mean accuracy and the mean specificity is mostly lower than the mean accuracy.

In Table 4 we can also observe that the feature set obtained using the fold change 3.00 has a mean accuracy over 96% when using the batch-adjustment method fsva and any of the three machine learning algorithms. This feature set is composed of 8 genes: ASPN, EIF1AY, FCN3, IL1RL1, NPPA, PLA2G2A, SERPINA3, and SFRP4. To determine which genes had previously been associated with HF we used DisGeNet (Version 7.0) [73]; 48 of the 117 genes have previously been associated with HF. Regarding the 6 genes of the feature set obtained using a fold change of 3.00, two of them have previously been associated with HF: PLA2G2A, and SERPINA3.

The processing times for the different algorithms, considering this use case, are represented in Table 5. This table includes the mean duration in minutes for hyperparameter tuning of a single training set, mean duration in seconds for training a single training set, and mean duration in seconds for testing a single testing set. These results were collected for each of the three machine-learning algorithms, combined with the two batch-adjustment methods considered in this study. As demonstrated in Table 5, the most time-consuming process is the hyperparameter tuning. The training process can be concluded in a few seconds, and the testing stage is almost immediate, taking less than a second. The parameter tuning of the linear SVM model requires the least amount of time. Linear SVM and neural network models use a constant number of values for hyperparameter tuning, which

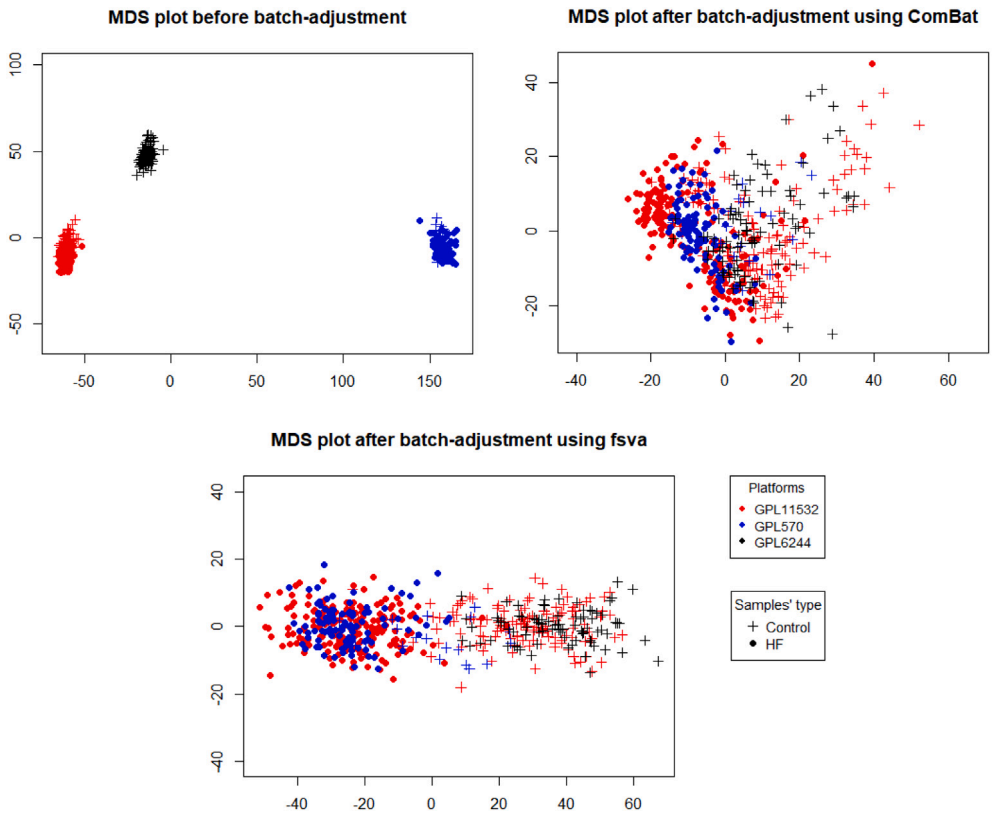

Fig. 3. MDS plot of the HF dataset before and after data batch-adjustment using ComBat and fsva.

Table 4
HF use case: For every fold change, the mean accuracy and standard deviation of the classifier.

Batch-adjustment	Fold change	Neural network	Random forest	Linear SVM
ComBat	1.50	**0.9777 ± 0.0125**	0.9646 ± 0.0150	**0.9782 ± 0.0109**
	1.75	0.9472 ± 0.1042	0.9664 ± 0.0162	0.9693 ± 0.0129
	2.00	0.9701 ± 0.0116	**0.9673 ± 0.0155**	0.9711 ± 0.0142
	2.25	0.9554 ± 0.0157	0.9592 ± 0.0161	0.9576 ± 0.0159
	2.50	0.9638 ± 0.0139	0.9592 ± 0.0168	0.9595 ± 0.0147
	2.75	0.9545 ± 0.0193	0.9528 ± 0.0174	0.9548 ± 0.0189
	3.00	0.9510 ± 0.0188	0.9455 ± 0.0177	0.9518 ± 0.0198
fsva	1.50	0.9713 ± 0.0629	**0.9744 ± 0.0115**	**0.9826 ± 0.0093**
	1.75	0.9713 ± 0.0128	0.9721 ± 0.0113	0.9753 ± 0.0111
	2.00	**0.9777 ± 0.0109**	0.9703 ± 0.0127	0.9771 ± 0.0114
	2.25	0.9750 ± 0.0113	0.9682 ± 0.0123	0.9788 ± 0.0117
	2.50	0.9736 ± 0.0111	0.9679 ± 0.0134	0.9742 ± 0.0121
	2.75	0.9685 ± 0.0159	0.9721 ± 0.0135	0.9696 ± 0.0127
	3.00	0.9635 ± 0.0171	0.9632 ± 0.0148	0.9656 ± 0.0140

Note: Accuracy is given as the mean of the accuracy obtained ± the standard deviation

reduces the computational time of the processes. However, the number of values used for hyperparameter tuning in the random forest model varies from 15 for 3.00 fold change to 375 for 1.50 fold change. This variation may explain the considerable reduction in time required for hyperparameter tuning using the feature set obtained using a 3.00 fold change compared to the features set obtained with a 1.50 fold change. Table 5 also reveals that the time used for tuning, training and testing is comparable for both batch-adjustment methods.

Table 6 presents the results obtained using the LargeMetabo package with the dataset after by processed by the batch-adjustment methods Combat and fsva. The accuracy values present in this table, may not provide fully accurate estimates of the performance for the defined biomarker set, because a 2-fold cross-validation method was employed to evaluate the classification model. This strategy is usually considered inadequate for this purpose. This table contains the results of different methods included in the LargeMetabo, that returned large sets of potential biomarkers. From these, it is included the Student's t-test and Wilcoxon rank test, which produced biomarker sets consisting of almost all the features in the dataset. The fold change method returned an empty set due to the defined threshold for biomarker selection. The proposed methodology employs several thresholds to be more flexible when dealing with potential issues, such as empty or excessively large sets of biomarkers.

4.2. Autism spectrum disorder use case

Autism spectrum disorder (ASD) is a neurodevelopmental disorder that in 2014 affected about 1% of the world's population [74]. Furthermore, over the years the prevalence of ASD has been increasing [75]. For this use case, we also used four publicly available microarray datasets. Table 7 presents the four datasets, as well as the respective platform, manufacturer and number of samples.

As for the previous use case, some of the data were excluded from the datasets. Several datasets had some replicate samples and these

Table 5
HF use case: For every fold change, the mean time for hyperparameter tuning, training and testing using the different classifiers.

Batch-adj.	FC	Tuning (min)			Training (min)			Testing (s)		
		NN	RF	LSVM	NN	RF	LSVM	NN	RF	LSVM
ComBat	1.50	10.80	60.89	**0.77**	4.90	3.54	**1.09**	0.016	**0.010**	**0.010**
	1.75	22.38	14.76	**0.94**	1.02	2.60	**0.84**	0.007	**0.005**	**0.005**
	2.00	8.17	4.22	**1.65**	2.91	1.72	**0.70**	0.004	0.005	**0.003**
	2.25	5.20	2.02	**1.00**	0.89	2.82	**0.66**	0.003	0.004	**0.002**
	2.50	3.76	1.09	**0.54**	0.81	2.07	**0.66**	**0.002**	0.004	**0.002**
	2.75	3.05	0.63	**0.39**	1.00	2.18	**0.69**	**0.002**	0.003	**0.002**
	3.00	2.75	0.51	**0.32**	0.71	0.83	**0.80**	0.002	0.002	0.002
fsva	1.50	10.53	52.26	**0.73**	4.14	1.86	**0.99**	0.016	**0.009**	0.010
	1.75	22.20	12.66	**0.49**	0.97	1.32	**0.81**	0.007	**0.005**	**0.005**
	2.00	8.33	3.73	**1.12**	1.96	1.00	**0.68**	0.004	**0.003**	**0.003**
	2.25	5.13	1.80	**0.77**	0.88	0.94	**0.67**	0.003	0.003	**0.002**
	2.50	3.74	1.00	**0.45**	1.83	0.86	**0.73**	0.003	**0.002**	**0.002**
	2.75	3.05	0.61	**0.37**	1.07	1.18	**0.64**	**0.002**	0.004	**0.002**
	3.00	2.77	0.49	**0.38**	0.78	0.81	**0.73**	0.002	0.002	0.002

Table 6
HF use case: Results obtained using LargeMetabo package.

Batch-adj.	Method	N. of features	Accuracy
ComBat	Correlation-based feature selection	107	0.9640
	Fold change	0	–
	Linear models and empirical Bayes method	37	0.9621
	Orthogonal partial least squares discrimination analysis	3138	0.9735
	Random forest-recursive feature elimination	5	0.9735
	Student's t-test	6702	0.9678
	Wilcoxon rank sum test	6988	0.9659
fsva	Correlation-based feature selection	5473	0.9735
	Fold change	0	–
	Linear models and empirical Bayes method	515	0.9640
	Orthogonal partial least squares discrimination analysis	5213	0.9735
	Random forest-recursive feature elimination	2	0.9716
	Student's t-test	9376	0.9754
	Wilcoxon rank sum test	9409	0.9754

Table 7
Summary of the datasets used in the use case ASD.

Dataset	Platform	Manufacturer	No. of samples (ASD/control)
GSE6575 [76]	GPL570	Affymetrix	**47** (35/12)
GSE18123 [77]	GPL570	Affymetrix	**99** (66/33)
	GPL6244	Affymetrix	**186** (104/82)
GSE42133 [78]	GPL10558	Illumina	**142** (87/55)
GSE111175 [79]	GPL10558	Illumina	**98** (34/64)
Total			**572** (326/246)

Table 8
The number of feature and the number of genes for every adjusted *p*-value used as a threshold.

Adjusted p-value	No. of features	No. of genes
0.050	108	108
0.040	79	79
0.030	52	52
0.020	25	25
0.010	8	8
0.009	7	7
0.008	6	6

were excluded from these datasets since they were obtained from the same patient. Concerning the dataset GSE6575, we maintained the control samples and the samples with ASD and excluded the samples referring to other perturbations. In relation to the dataset GSE111175, we used the samples with ASD and pervasive developmental disorder, as well as the control samples and excluded the samples referring to other perturbations.

For feature selection in this use case, we chose to use the adjusted *p*-value to obtain the different sets of features. The adjusted *p*-value thresholds used are: 0.008, 0.009, 0.010, 0.020, 0.030, 0.040, 0.050.

The ASD use case was validated using the datasets obtained from three different platforms, namely two manufactured by Affymetrix (GPL570 and GPL6244) and one manufactured by Illumina (GPL105 508). These three platforms have 6222 GenBank sequence accession identifiers in common. Therefore, the merged dataset has 6222 features and 572 samples. By dividing the merged dataset we obtained 50 training sets and the corresponding 50 test sets. Each training set is composed of 6222 features and 430 samples and each test set is composed of 6222 features and 142 samples. We used the adjusted *p*-value and seven thresholds to obtain the different feature sets. Table 8 contains the number of features and the number of genes of the seven feature sets.

Fig. 4 displays the MDS plot of the data before and after batch-adjustment using ComBat and fsva methods. However, in this case, the ComBat batch-adjustment method did not eliminate all the batches. One reason for this insufficient batch-adjustment may be the fact that not all platforms are manufactured by the same company. As can be seen, the remaining clusters are between the Affymetrix platforms and the Illumina platform.

Table 9 presents the mean accuracy and the standard deviation of the model achieving the best mean accuracy, using the three machine learning algorithms and the two batch-adjustment methods. All the results of this use case can be found in the Supplementary File `results_ADS.xlsx`.

The results show that the best mean accuracy was obtained using an adjusted *p*-value of 0.040, the ComBat batch-adjustment method and neural network algorithm. The feature set obtained using an adjusted *p*-value of 0.040 is composed of 79 genes and has an accuracy of

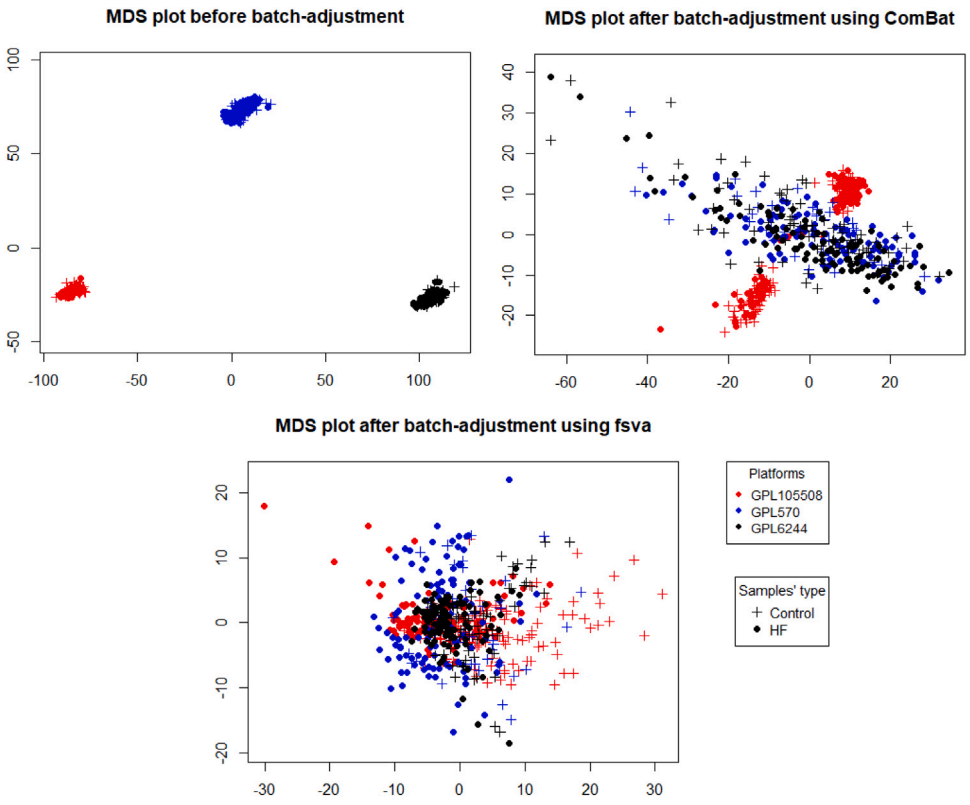

Fig. 4. MDS plot of the ASD dataset before and after data batch-adjustment using ComBat and fsva.

Table 9
ASD use case: For every adjusted *p*-value, the mean accuracy and standard deviation of the classifier.

Batch-adjustment	Adjusted p-value	Neural network	Random forest	Linear SVM
ComBat	0.050	0.8151 ± 0.0353	0.6862 ± 0.0361	0.7207 ± 0.0307
	0.040	**0.8176 ±0.0340**	**0.6908 ±0.0339**	**0.7361 ±0.0273**
	0.030	0.7054 ± 0.0424	0.6721 ± 0.0332	0.6845 ± 0.0285
	0.020	0.6706 ± 0.0348	0.6661 ± 0.0354	0.6887 ± 0.0276
	0.010	0.6125 ± 0.0264	0.6383 ± 0.0317	0.6396 ± 0.0338
	0.009	0.6203 ± 0.0398	0.6411 ± 0.0330	0.6545 ± 0.0311
	0.008	0.6314 ± 0.0417	0.6325 ± 0.0311	0.6524 ± 0.0313
fsva	0.050	**0.7439 ± 0.0319**	**0.7739 ± 0.0307**	**0.7551 ± 0.0324**
	0.040	0.7326 ± 0.0353	0.7483 ± 0.0307	0.7492 ± 0.0308
	0.030	0.7321 ± 0.0326	0.7452 ± 0.0355	0.7435 ± 0.0296
	0.020	0.7097 ± 0.0334	0.7186 ± 0.0318	0.7180 ± 0.0271
	0.010	0.6928 ± 0.0305	0.6818 ± 0.0280	0.7015 ± 0.0254
	0.009	0.6899 ± 0.0367	0.6773 ± 0.0279	0.7059 ± 0.0277
	0.008	0.6952 ± 0.0372	0.6672 ± 0.0276	0.7065 ± 0.0298

Note: Accuracy is given as the mean of the accuracy obtained ± the standard deviation

approximately 82%. Furthermore, this feature set obtained the best mean specificity, mean precision, mean recall and mean F1 score. The 18 up-regulated genes and the 61 down-regulated genes are presented in Supplementary File `results_ADS.xlsx`.

Neural network performed better when using the ComBat batch-adjustment algorithm, however, random forest and linear SVM performed better when using the fsva batch-adjustment algorithm. The mean precision and the mean recall are mostly higher than the mean accuracy and the mean specificity is mostly lower than the mean accuracy.

Once more, we used DisGeNet to identify which genes had previously been associated with ASD. 3 of the 79 genes have previously been associated with ASD: CD79A, EIF3A, and NDUFS1. CD79A is up-regulated while EIF3A and NDUFS1 are down-regulated. These results are not as good as the previous use case, which may be explained by the datasets that were obtained using chips from different manufacturers, leading to a reduced the number of features in the merged dataset.

Table 10 considers similar parameters as defined in Table 5. However, the metrics obtained in this table relate to the ASD use case. Similar to the previous use case, the most time-consuming process is the hyperparameter tuning. However, in contrast to the previous use case, the parameter tuning of the linear SVM model did not exhibit the shortest duration. For the present use case, the number of values used for hyperparameter tuning in the random forest model varied between 15 and 321 for adjusted p-values of 0.008 and 0.050, respectively. The presented results may justify the reason for the shortest feature sets the random forest model being the one that requires less time for tuning the parameters.

Table 11 presents the results of biomarker identification for the ASD use case. For this use case, several methods have returned empty sets of biomarkers, including correlation-based feature selection, fold change and linear models and the empirical Bayes method. Additionally, when using the batch-adjustment method Combat, orthogonal

Table 10
ASD use case: For every fold change, the mean time for hyperparameter tuning, training and testing using the different classifiers.

Batch-adj.	Adj. p-value	Tuning (min)			Training (s)			Testing (s)		
		NN	RF	LSVM	NN	RF	LSVM	NN	RF	LSVM
ComBat	0.050	**20.96**	104.90	87.34	27.37	4.37	**2.38**	0.014	**0.008**	0.013
	0.040	**27.10**	61.28	75.48	30.28	3.51	**1.98**	0.011	**0.008**	0.010
	0.030	**25.72**	30.47	49.36	15.93	6.56	**1.87**	0.008	**0.007**	0.008
	0.020	10.14	**8.94**	15.32	5.14	3.32	**1.57**	**0.004**	0.006	**0.004**
	0.010	3.77	**1.31**	2.24	1.88	2.99	**0.80**	**0.002**	0.005	**0.002**
	0.009	3.62	**1.04**	1.75	2.26	2.02	**0.79**	**0.002**	0.005	**0.002**
	0.008	3.38	**0.81**	1.43	0.97	1.11	**0.77**	0.002	0.002	0.002
fsva	0.050	**20.59**	110.30	91.08	17.08	19.29	**1.82**	0.014	0.015	**0.012**
	0.040	**27.01**	63.15	76.97	7.15	7.49	**1.64**	**0.010**	0.014	**0.010**
	0.030	**25.89**	31.16	45.31	6.04	11.07	**1.41**	**0.008**	0.010	**0.008**
	0.020	10.30	**9.56**	13.47	5.14	5.60	**0.98**	**0.004**	0.009	**0.004**
	0.010	4.05	**1.44**	1.90	1.18	4.74	**1.13**	**0.002**	0.008	**0.002**
	0.009	3.81	**1.15**	1.66	0.97	2.88	**0.90**	**0.002**	0.008	**0.002**
	0.008	3.47	**0.89**	1.29	0.88	6.04	**0.74**	**0.002**	0.007	**0.002**

Table 11
ASD use case: Results obtained using LargeMetabo package.

Batch-adj.	Method	N. of features	Accuracy
ComBat	Correlation-based feature selection	0	–
	Fold change	0	–
	Linear models and empirical Bayes method	0	–
	Orthogonal partial least squares discrimination analysis	0	–
	Random forest-recursive feature elimination	37	0.7378
	Student's t-test	780	0.6976
	Wilcoxon rank sum test	1953	0.6626
fsva	Correlation-based feature selection	0	–
	Fold change	0	–
	Linear models and empirical Bayes method	0	–
	Orthogonal partial least squares discrimination analysis	1826	0.7710
	Random forest-recursive feature elimination	24	0.7640
	Student's t-test	2473	0.7570
	Wilcoxon rank sum test	3016	0.7549

partial least squares discrimination analysis also resulted in an empty set of biomarkers. Most of the remaining biomarker identification methods returned relatively large sets of potential biomarkers. However, as previously mentioned, the proposed methodology employs several thresholds that help mitigate issues, such as returning empty or excessively large sets of potential biomarkers.

5. Discussion

The proposed methodology demonstrated a good efficiency to identify gene expression signatures. The pipeline consists of several steps, including data preprocessing, feature selection, and supervised machine learning. For each step of the pipeline, we provide a list of potential methods that can be used. These lists are intended to serve as a starting point and guide for researchers, rather than prescribing a single approach that must be followed.

It is important to note that the performance of feature selection and supervised machine learning methods is dependent on various factors, including the quality of the data, the nature of the problem being addressed, the choice of algorithm, and computational resources. Therefore, we recommend that researchers carefully consider these factors when selecting methods for each step of the pipeline.

To demonstrate the efficacy of this methodology, we employed two research applications using publicly available microarray datasets. For each case study, we selected one or more methods from the provided lists for each step of the pipeline. By using multiple methods, we were able to compare their performance in the specific context of each case study.

As the primary metric for evaluating the performance of the GES, we used accuracy. In the heart failure case study, we obtained a GES consisting of 117 genes with a classification accuracy of approximately 98%. In the autism spectrum disorder case study, we obtained a GES of 79 genes with a classification accuracy of approximately 82%. In addition to accuracy, we also measured precision, recall, F1 score, and specificity, with the results presented in the supplementary material.

To ensure the generalizability and robustness of the proposed strategy for GES, we employed a cross-validation approach in which we randomly divided the merged dataset into training and test sets 50 times, applying the feature selection process to the 50 training sets and intersecting the results. This allowed us to assess the stability of the GES across different training and test sets. In this section, we also evaluate the biological relevance of the genes in the GES in relation to the problem at hand. These analyses helped to provide insight into the mechanisms underlying the classification results and may inform future research on these conditions.

5.1. Biological insight of the results

Considering the complexity of the study context, this section does not explore in depth every gene and disease mechanism but aims to provide a biological insight from the results, supporting the application of our methodology.

To obtain an overview of the functional meaning of the results, we analyzed the gene ontology annotations of each gene signature. In the heart failure case study, we observed an enrichment of inflammatory and immune response processes in the down-regulated genes. Among those genes, we found IL1RL1, PLA2G2A and SERPINA3, which have been previously associated with heart disease, according to the DisGeNet database. Additionally, the calcium binding proteins encoded by genes S100A8 and S100A9 are specifically expressed by myeloid cells and are involved in several inflammation related processes, via the Toll-like receptor 4 (TLR4) activation. The regulation of the inflammatory response has a dual role in the development and progression of heart failure, as it can often act as a trigger, but its suppression is also

required to induce the regenerative process. Therefore, the modulation of the inflammation associated pathways has been identified as a therapeutic strategy in heart failure [80]. Considering the up-regulated genes, the most relevant process was the epithelial proliferation, which is essential to ensure the formation of a scar and for the development of heart tissue and cardiac remodeling in a post-infarction response. Among those genes and their respective proteins, we found several proteoglycans and secreted frizzled-related proteins (such as SFRP4 and Frizzled-7). The latter, SFRPs, belong to the non-canonical Wnt signaling pathway and participate in the up-regulation of epithelial cell proliferation [81].

In the second case study, ASD, the number of up-regulated genes (61) was much higher than the down-regulated ones (18), which may be related to an increase of brain activity. This is concordant with recent studies, which report a higher number of synapses in the autistic brain, associated with an mTOR-dependent increase of dendritic spines density [82]. About one-third of the up-regulated genes (15 to 20 out of 61) are involved in processes of anatomical structure development, cell differentiation and signal transduction, which are critical in a neurodevelopment disorder such ASD. Although distinct, half the down-regulated genes (9 out of 18) were also involved in processes related to the development of anatomical structures. The most prominent down-regulated gene was the GABA type A receptor-associated protein (GABARAP) gene, which participates in many processes related to cellular signaling, including cell–cell signaling, protein transport and targeting, response to stress, among others. Although this specific gene is not reported as such in DisGeNet, the GABA receptor modulators have been recently suggested as a potential therapeutic target in ASD, due to the critical regulation exerted by GABAergic inhibitory transmission on neuronal activities during brain development [83].

Among the up-regulated genes, EIF3A and NDUFS1 were previously associated with ASD, according to the DisGeNet. Both have essential physiological cell functions and, when dysregulated, may lead to different pathological conditions. Namely, eIF3 is one of the subunits of the eukaryotic initiation factor 3, participating in the translation process of gene expression, and the protein encoded by the NDUFS1 gene participates in the energy production through the mitochondrial respiration, due to its NADH dehydrogenase and oxidoreductase activity. The Toll-like receptor 2 (TLR2) was one of the genes that we found to be involved in more processes, according to gene ontology annotations. The TLR pathway is mainly involved in innate immune responses. In the pathophysiology of ASD, there is a strong component of immune dysfunction, and the levels of cytokines in B cells were recently studied in children with autism [84]. Among their main findings, the authors observed that activation of the TLR4 was responsible for the increase of the inflammatory potential of B cells in ASD.

Interestingly, analysis of the two case studies highlighted the importance of inflammation in both pathogenic processes, particularly through the presence of the TLR pathway. As a curiosity, TLR4 can be activated by the extracellular forms of two NAD related enzymes (eNAPRT and eNAMPT) and induce inflammation [85]. In addition, a recent study has suggested the involvement of NAPRT in neural development [86]. Altogether, these findings are mentioned here to show that the results obtained through the methodology presented in this study are supported by the literature.

5.2. Multicentre processing

The proposed methodology was created focused on aggregating multiple datasets to increase the accuracy of the data processing algorithms. Several data owners are reluctant to share their dataset publicly, and therefore, some initiatives have invested in strategies for multicentre studies [87–89]. These strategies enable collaboration between researchers from distinct institutions, with the possibility of sharing their datasets in a secure way.

Almeida et al. [89] proposed a semi-automatic methodology to analyze distributed repositories of genomic data. This strategy aims to conduct a multicentre study without the need for data owners to release the datasets. To accomplish this, three different actors have specific responsibilities in the study, namely: (i) researcher, the entity interested in conducting the study; (ii) study manager, the entity with higher privileges who knows the data owners; and (iii) the data owners.

With our methodology, we can apply similar principles, which would help researchers to conduct a multicentre microarray study, by merging the datasets of interest without ever accessing the data. We include in this work all the necessary tools for the study manager to process the datasets in a Private Remote Research Environment (PRRE). Then, this entity can share the results with the researcher without exposing the data.

6. Conclusion

In this paper, we present a general pipeline for identifying a gene expression signature. Using statistical methods and supervised machine learning algorithms, this approach overcomes limitations such as the need to set an arbitrary threshold for gene selection when using statistical methods and the ineffectiveness of supervised machine learning algorithms when applied to microarray data.

We applied the methodology to two use cases: one using heart failure microarray datasets and the other using autism spectrum disorder microarray datasets. For the HF use case, we identified a GES of 117 genes with high accuracy (98%). For the ASD use case, we identified a GES of 79 genes with approximately 82% accuracy. For this second use case, we used datasets obtained using microarrays from different manufacturers. This study shows that the methodology presented is appropriate to identify common GESs across multiple microarray experiments, helping to increase the statistical power of small datasets. Furthermore, this methodology can be used for different diseases.

The methodology presented has several limitations. One potential limitation is that merging microarray datasets obtained from different platforms may result in the loss of genes that are not present on all platforms. Additionally, the methodology provides a list of methods that could be used at various stages of the pipeline, requiring the researcher to choose the most appropriate method for their specific study.

In future work, we plan to apply this methodology to a diverse range of microarray data and evaluate the performance of various methods at different stages of the pipeline. We also plan to utilize ensemble learning techniques to combine multiple methods in both the feature selection and gene expression signature identification steps.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has received support from the FCT — Foundation for Science and Technology (national funds) within project DSAIPA/AI/0088/2020. SDP and JRA are funded by the FCT — Foundation for Science and Technology under the grants SFRH/BD/108890/2015 and SFRH/BD/147837/2019, respectively.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.compbiomed.2023.106867.

References

[1] N.J. Armstrong, M.A. Van de Wiel, Microarray data analysis: From hypotheses to conclusions using gene expression data, Anal. Cell. Pathol. 26 (5–6) (2004) 279–290, http://dx.doi.org/10.1155/2004/943940.

[2] C. Kumar Sarmah, S. Samarasinghe, Microarray data integration: Frameworks and a list of underlying issues, Curr. Bioinform. 5 (4) (2010) 280–289, http://dx.doi.org/10.2174/157489310794072517.

[3] E.M. Price, W.P. Robinson, Adjusting for batch effects in DNA methylation microarray data, a lesson learned, Front. Genet. 9 (2018) 83, http://dx.doi.org/10.3389/fgene.2018.00083.

[4] E. Clough, T. Barrett, The gene expression omnibus database, in: E. Mathé, S. Davis (Eds.), Statistical Genomics, Springer, New York, 2016, pp. 93–110, http://dx.doi.org/10.1007/978-1-4939-3578-9_5.

[5] A. Athar, A. Füllgrabe, N. George, H. Iqbal, L. Huerta, A. Ali, C. Snow, N.A. Fonseca, R. Petryszak, I. Papatheodorou, et al., ArrayExpress update–from bulk to single-cell expression data, Nucleic Acids Res. 47 (D1) (2018) D711–D715, http://dx.doi.org/10.1093/nar/gky964.

[6] F. Chibon, Cancer gene expression signatures–The rise and fall? Eur. J. Cancer 49 (8) (2013) 2000–2009, http://dx.doi.org/10.1016/j.ejca.2013.02.021.

[7] S. Sithara, T.M. Crowley, K. Walder, K. Aston-Mourney, Gene expression signature: A powerful approach for drug discovery in diabetes, J. Endocrinol. 232 (2017) R131–39, http://dx.doi.org/10.1530/joe-16-0515.

[8] J. Canul-Reich, L.O. Hall, D. Goldgof, S.A. Eschrich, Feature selection for microarray data by AUC analysis, in: 2008 IEEE International Conference on Systems, Man and Cybernetics, IEEE, 2008, pp. 768–773, http://dx.doi.org/10.1109/icsmc.2008.4811371.

[9] S. Michiels, S. Koscielny, C. Hill, Prediction of cancer outcome with microarrays: A multiple random validation strategy, Lancet 365 (9458) (2005) 488–492, http://dx.doi.org/10.1016/s0140-6736(05)17866-0.

[10] J. Wang, K.A. Do, S. Wen, S. Tsavachidis, T.J. McDonnell, C.J. Logothetis, K.R. Coombes, Merging microarray data, robust feature selection, and predicting prognosis in prostate cancer, Cancer Inform. 2 (2006) 117693510600200009, http://dx.doi.org/10.1177/117693510600200009.

[11] L. Xu, A.C. Tan, R.L. Winslow, D. Geman, Merging microarray data from separate breast cancer studies provides a robust prognostic test, BMC Bioinformatics 9 (1) (2008) 125, http://dx.doi.org/10.1186/1471-2105-9-125.

[12] C. Lazar, S. Meganck, J. Taminau, D. Steenhoff, A. Coletta, C. Molter, D.Y. Weiss-Solís, R. Duque, H. Bersini, A. Nowé, Batch effect removal methods for microarray gene expression data integration: A survey, Brief. Bioinform. 14 (4) (2013) 469–490, http://dx.doi.org/10.1093/bib/bbs037.

[13] C. Lazar, J. Taminau, S. Meganck, D. Steenhoff, A. Coletta, C. Molter, V. de Schaetzen, R. Duque, H. Bersini, A. Nowe, A survey on filter techniques for feature selection in gene expression microarray analysis, IEEE/ACM Trans. Comput. Biol. Bioinform. 9 (4) (2012) 1106–1119, http://dx.doi.org/10.1109/tcbb.2012.33.

[14] C.J. Walsh, P. Hu, J. Batt, C.C. Dos Santos, Microarray meta-analysis and cross-platform normalization: Integrative genomics for robust biomarker discovery, Microarrays 4 (3) (2015) 389–406, http://dx.doi.org/10.3390/microarrays4030389.

[15] J. Taminau, C. Lazar, S. Meganck, A. Nowé, Comparison of merging and meta-analysis as alternative approaches for integrative gene expression analysis, Int. Sch. Res. Notices 2014 (2014) http://dx.doi.org/10.1155/2014/345106.

[16] G.C. Tseng, D. Ghosh, E. Feingold, Comprehensive literature review and statistical considerations for microarray meta-analysis, Nucleic Acids Res. 40 (9) (2012) 3785–3799, http://dx.doi.org/10.1093/nar/gkr1265.

[17] J. Feichtinger, R.J. McFarlane, L.D. Larcombe, CancerMA: A web-based tool for automatic meta-analysis of public cancer microarray data, Database 2012 (2012) http://dx.doi.org/10.1093/database/bas055.

[18] A. Heider, R. Alt, Virtualarray: A R/bioconductor package to merge raw data from different microarray platforms, BMC Bioinformatics 14 (1) (2013) 1–10, http://dx.doi.org/10.1186/1471-2105-14-75.

[19] J. Taminau, S. Meganck, C. Lazar, D. Steenhoff, A. Coletta, C. Molter, R. Duque, V.d. Schaetzen, D.Y. Weiss Solís, H. Bersini, et al., Unlocking the potential of publicly available microarray data using inSilicoDb and inSilicoMerging R/Bioconductor packages, BMC Bioinformatics 13 (1) (2012) 1–9, http://dx.doi.org/10.1186/1471-2105-13-335.

[20] J. Vey, L.A. Kapsner, M. Fuchs, P. Unberath, G. Veronesi, M. Kunz, A toolbox for functional analysis and the systematic identification of diagnostic and prognostic gene expression signatures combining meta-analysis and machine learning, Cancers 11 (10) (2019) 1606, http://dx.doi.org/10.3390/cancers11101606.

[21] X.-Q. Xia, M. McClelland, S. Porwollik, W. Song, X. Cong, Y. Wang, WebArrayDB: Cross-platform microarray data analysis and public data repository, Bioinformatics 25 (18) (2009) 2425–2429, http://dx.doi.org/10.1093/bioinformatics/btp430.

[22] L.L. Elo, S. Filén, R. Lahesmaa, T. Aittokallio, Reproducibility-optimized test statistic for ranking genes in microarray studies, IEEE/ACM Trans. Comput. Biol. Bioinform. 5 (3) (2008) 423–431, http://dx.doi.org/10.1109/tcbb.2007.1078.

[23] M.E. Ritchie, B. Phipson, D. Wu, Y. Hu, C.W. Law, W. Shi, G.K. Smyth, Limma powers differential expression analyses for RNA-sequencing and microarray studies, Nucleic Acids Res. 43 (7) (2015) e47, http://dx.doi.org/10.1093/nar/gkv007.

[24] K. Chrominski, M. Tkacz, Comparison of high-level microarray analysis methods in the context of result consistency, PLoS One 10 (6) (2015) e0128845, http://dx.doi.org/10.1371/journal.pone.0128845.

[25] M. Jeanmougin, A. De Reynies, L. Marisa, C. Paccard, G. Nuel, M. Guedj, Should we abandon the t-test in the analysis of gene expression microarray data: A comparison of variance modeling strategies, PLoS One 5 (9) (2010) e12336, http://dx.doi.org/10.1371/journal.pone.0012336.

[26] J. Fan, Y. Ren, Statistical analysis of DNA microarray data in cancer research, Clin. Cancer Res. 12 (15) (2006) 4469–4473, http://dx.doi.org/10.1158/1078-0432.CCR-06-1033.

[27] M. Tumminello, G. Bertolazzi, G. Sottile, N. Sciaraffa, W. Arancio, C. Coronnello, A multivariate statistical test for differential expression analysis, Sci. Rep. 12 (1) (2022) 1–10, http://dx.doi.org/10.1038/s41598-022-12246-w.

[28] S. Karthik, M. Sudha, A survey on machine learning approaches in gene expression classification in modelling computational diagnostic system for complex diseases, Int. J. Eng. Adv. Technol. 8 (2) (2018) 182–191.

[29] Q. Yang, B. Li, J. Tang, X. Cui, Y. Wang, X. Li, J. Hu, Y. Chen, W. Xue, Y. Lou, et al., Consistent gene signature of schizophrenia identified by a novel feature selection strategy from comprehensive sets of transcriptomic data, Brief. Bioinform. 21 (3) (2020) 1058–1068, http://dx.doi.org/10.1093/bib/bbz049.

[30] F. Li, J. Yin, M. Lu, Q. Yang, Z. Zeng, B. Zhang, Z. Li, Y. Qiu, H. Dai, Y. Chen, et al., ConSIG: Consistent discovery of molecular signature from OMIC data, Brief. Bioinform. 23 (4) (2022) bbac253, http://dx.doi.org/10.1093/bib/bbac253.

[31] B. Sahu, S. Dehuri, A. Jagadev, A study on the relevance of feature selection methods in microarray data, Open Bioinform. J. 11 (1) (2018) http://dx.doi.org/10.2174/1875036201811010117.

[32] B. Remeseiro, V. Bolon-Canedo, A review of feature selection methods in medical applications, Comput. Biol. Med. 112 (2019) 103375, http://dx.doi.org/10.1016/j.compbiomed.2019.103375.

[33] M. Qaraad, S. Amjad, H. Fathi, I.I. Manhrawy, Feature selection techniques for cancer classification applied to microarray data: A survey, in: 2019 International Conference on Intelligent Systems and Advanced Computing Sciences, ISACS, IEEE, 2019, pp. 1–8, http://dx.doi.org/10.1109/ISACS48493.2019.9068865.

[34] Z.M. Hira, D.F. Gillies, A review of feature selection and feature extraction methods applied on microarray data, Adv. Bioinform. 2015 (2015) http://dx.doi.org/10.1155/2015/198363.

[35] D. Castillo, J.M. Galvez, L.J. Herrera, F. Rojas, O. Valenzuela, O. Caba, J. Prados, I. Rojas, Leukemia multiclass assessment and classification from Microarray and RNA-seq technologies integration at gene expression level, PLoS One 14 (2) (2019) http://dx.doi.org/10.1371/journal.pone.0212127.

[36] J. Fu, Y. Zhang, J. Liu, X. Lian, J. Tang, F. Zhu, Pharmacometabonomics: Data processing and statistical analysis, Brief. Bioinform. 22 (5) (2021) bbab138, http://dx.doi.org/10.1093/bib/bbab138.

[37] Y.H. Li, X.X. Li, J.J. Hong, Y.X. Wang, J.B. Fu, H. Yang, C.Y. Yu, F.C. Li, J. Hu, W.W. Xue, et al., Clinical trials, progression-speed differentiating features and swiftness rule of the innovative targets of first-in-class drugs, Brief. Bioinform. 21 (2) (2020) 649–662, http://dx.doi.org/10.1093/bib/bby130.

[38] F. Li, Y. Zhou, Y. Zhang, J. Yin, Y. Qiu, J. Gao, F. Zhu, POSREG: Proteomic signature discovered by simultaneously optimizing its reproducibility and generalizability, Brief. Bioinform. 23 (2) (2022) http://dx.doi.org/10.1093/bib/bbac040.

[39] J. Fu, Y. Zhang, Y. Wang, H. Zhang, J. Liu, J. Tang, Q. Yang, H. Sun, W. Qiu, Y. Ma, et al., Optimization of metabolomic data processing using NOREVA, Nat. Protoc. 17 (1) (2022) 129–151, http://dx.doi.org/10.1038/s41596-021-00636-9.

[40] Q. Yang, B. Li, P. Wang, J. Xie, Y. Feng, Z. Liu, F. Zhu, LargeMetabo: An out-of-the-box tool for processing and analyzing large-scale metabolomic data, Brief. Bioinform. 23 (6) (2022) bbac455, http://dx.doi.org/10.1093/bib/bbac455.

[41] Q. Yang, B. Li, S. Chen, J. Tang, Y. Li, Y. Li, S. Zhang, C. Shi, Y. Zhang, M. Mou, et al., MMEASE: Online meta-analysis of metabolomic data by enhanced metabolite annotation, marker selection and enrichment analysis, J. Proteomics 232 (2021) 104023, http://dx.doi.org/10.1016/j.jprot.2020.104023.

[42] B. Li, J. Tang, Q. Yang, S. Li, X. Cui, Y. Li, Y. Chen, W. Xue, X. Li, F. Zhu, NOREVA: Normalization and evaluation of MS-based metabolomics data, Nucleic Acids Res. 45 (W1) (2017) W162–W170, http://dx.doi.org/10.1093/nar/gkx449.

[43] J. Tang, J. Fu, Y. Wang, B. Li, Y. Li, Q. Yang, X. Cui, J. Hong, X. Li, Y. Chen, et al., ANPELA: analysis and performance assessment of the label-free quantification workflow for metaproteomic studies, Brief. Bioinform. 21 (2) (2020) 621–636, http://dx.doi.org/10.1093/bib/bby127.

[44] J. Tang, M. Mou, Y. Wang, Y. Luo, F. Zhu, MetaFS: Performance assessment of biomarker discovery in metaproteomics, Brief. Bioinform. 22 (3) (2021) bbaa105, http://dx.doi.org/10.1093/bib/bbaa105.

[45] Q. Yang, Y. Wang, Y. Zhang, F. Li, W. Xia, Y. Zhou, Y. Qiu, H. Li, F. Zhu, NOREVA: Enhanced normalization and evaluation of time-course and multi-class metabolomic data, Nucleic Acids Res. 48 (W1) (2020) W436–W448, http://dx.doi.org/10.1093/nar/gkaa258.

[46] A. Sánchez, M. de Villa, A Tutorial Review of Microarray Data Analysis, Universitat de Barcelona, 2008.
[47] M.E. Ritchie, J. Silver, A. Oshlack, M. Holmes, D. Diyagama, A. Holloway, G.K. Smyth, A comparison of background correction methods for two-colour microarrays, Bioinformatics 23 (20) (2007) 2700–2707, http://dx.doi.org/10.1093/bioinformatics/btm412.
[48] B.M. Bolstad, R.A. Irizarry, M. Åstrand, T.P. Speed, A comparison of normalization methods for high density oligonucleotide array data based on variance and bias, Bioinformatics 19 (2) (2003) 185–193, http://dx.doi.org/10.1093/bioinformatics/19.2.185.
[49] R.A. Irizarry, B. Hobbs, F. Collin, Y.D. Beazer-Barclay, K.J. Antonellis, U. Scherf, T.P. Speed, Exploration, normalization, and summaries of high density oligonucleotide array probe level data, Biostatistics 4 (2) (2003) 249–264, http://dx.doi.org/10.1093/biostatistics/4.2.249.
[50] B.S. Carvalho, R.A. Irizarry, A framework for oligonucleotide microarray preprocessing, Bioinformatics 26 (19) (2010) 2363–2367, http://dx.doi.org/10.1093/bioinformatics/btq431.
[51] S. Bandyopadhyay, S. Mallik, A. Mukhopadhyay, A survey and comparative study of statistical tests for identifying differential expression from microarray data, IEEE/ACM Trans. Comput. Biol. Bioinform. 11 (1) (2013) 95–115, http://dx.doi.org/10.1109/tcbb.2013.147.
[52] B. Zhao, A. Erwin, B. Xue, How many differentially expressed genes: A perspective from the comparison of genotypic and phenotypic distances, Genomics 110 (1) (2018) 67–73, http://dx.doi.org/10.1016/j.ygeno.2017.08.007.
[53] S. Draghici, Statistical intelligence: effective analysis of high-density microarray data, Drug Discov. Today 7 (11) (2002) S55–S63, http://dx.doi.org/10.1016/s1359-6446(02)02292-4.
[54] J.D. Storey, R. Tibshirani, Statistical significance for genomewide studies, Proc. Natl. Acad. Sci. 100 (16) (2003) 9440–9445, http://dx.doi.org/10.1073/pnas.1530509100.
[55] M.J. Larsen, M. Thomassen, Q. Tan, K.P. Sørensen, T.A. Kruse, Microarray-based RNA profiling of breast cancer: Batch effect removal improves cross-platform consistency, BioMed. Res. Int. 2014 (2014) http://dx.doi.org/10.1155/2014/651751.
[56] V. Nygaard, E.A. Rødland, E. Hovig, Methods that remove batch effects while retaining group differences may lead to exaggerated confidence in downstream analyses, Biostatistics 17 (1) (2016) 29–39, http://dx.doi.org/10.1093/biostatistics/kxv027.
[57] C. Chen, K. Grennan, J. Badner, D. Zhang, E. Gershon, L. Jin, C. Liu, Removing batch effects in analysis of expression microarray data: An evaluation of six batch adjustment methods, PLoS One 6 (2) (2011) e17238, http://dx.doi.org/10.1371/journal.pone.0017238.
[58] W.E. Johnson, C. Li, A. Rabinovic, Adjusting batch effects in microarray expression data using empirical Bayes methods, Biostatistics 8 (1) (2007) 118–127, http://dx.doi.org/10.1093/biostatistics/kxj037.
[59] C. Müller, A. Schillert, C. Röthemeier, D.-A. Trégouët, C. Proust, H. Binder, N. Pfeiffer, M. Beutel, K.J. Lackner, R.B. Schnabel, et al., Removing batch effects from longitudinal gene expression-quantile normalization plus combat as best approach for microarray transcriptome data, PLoS One 11 (6) (2016) e0156594, http://dx.doi.org/10.1371/journal.pone.0156594.
[60] J.T. Leek, W.E. Johnson, H.S. Parker, A.E. Jaffe, J.D. Storey, The sva package for removing batch effects and other unwanted variation in high-throughput experiments, Bioinformatics 28 (6) (2012) 882–883, http://dx.doi.org/10.1093/bioinformatics/bts034.
[61] R. Hornung, A.-L. Boulesteix, D. Causeur, Combining location-and-scale batch effect adjustment with data cleaning by latent factor adjustment, BMC Bioinformatics 17 (1) (2016) 27, http://dx.doi.org/10.1186/s12859-015-0870-z.
[62] K. Lan, D.-t. Wang, S. Fong, L.-s. Liu, K.K. Wong, N. Dey, A survey of data mining and deep learning in bioinformatics, J. Med. Syst. 42 (8) (2018) 139, http://dx.doi.org/10.1007/s10916-018-1003-9.
[63] M. Daoud, M. Mayo, A survey of neural network-based cancer prediction models from microarray data, Artif. Intell. Med. 97 (2019) 204–214, http://dx.doi.org/10.1016/j.artmed.2019.01.006.
[64] A. Statnikov, L. Wang, C.F. Aliferis, A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification, BMC Bioinformatics 9 (1) (2008) 1–10, http://dx.doi.org/10.1186/1471-2105-9-319.
[65] M. Kuhn, et al., Building predictive models in R using the caret package, J. Stat. Softw. 28 (5) (2008) 1–26, http://dx.doi.org/10.18637/jss.v028.i05.
[66] W. Shi, A. Oshlack, G.K. Smyth, Optimizing the noise versus bias trade-off for illumina whole genome expression BeadChips, Nucleic Acids Res. 38 (22) (2010) e204, http://dx.doi.org/10.1093/nar/gkq871.
[67] Y. Benjamini, Y. Hochberg, Controlling the false discovery rate: A practical and powerful approach to multiple testing, J. R. Stat. Soc. Ser. B Stat. Methodol. 57 (1) (1995) 289–300, http://dx.doi.org/10.1111/j.2517-6161.1995.tb02031.x.
[68] G. Savarese, L.H. Lund, Global public health burden of heart failure, Cardiac Fail. Rev. 3 (1) (2017) 7, http://dx.doi.org/10.15420/cfr.2016:25:2.
[69] A. Paul, M. Schinke, J. Brown, L. Riggi, S. Izumo, J. Bartunek, P. Allen, M. Tsubakihara, Changes in Cardiac Transcription Profiles Brought About by Heart Failure, Bauer Center for Genomic Research. NCBI, Gene Expression Omnibus, 2004.
[70] P. Schwientek, P. Ellinghaus, S. Steppan, D. D'Urso, M. Seewald, A. Kassner, R. Cebulla, S. Schulte-Eistrup, M. Morshuis, D. Röfe, et al., Global gene expression analysis in nonfailing and failing myocardium pre-and postpulsatile and nonpulsatile ventricular assist device support, Physiol. Genomics 42 (3) (2010) 397–405, http://dx.doi.org/10.1152/physiolgenomics.00030.2010.
[71] A.P. Pilbrow, L. Folkersen, J.F. Pearson, C.M. Brown, L. McNoe, N.M. Wang, W.E. Sweet, W.W. Tang, M.A. Black, R.W. Troughton, et al., The chromosome 9p21.3 coronary heart disease risk allele is associated with altered gene expression in normal heart and vascular tissues, PLoS One 7 (6) (2012) e39574, http://dx.doi.org/10.1371/journal.pone.0039574.
[72] Y. Liu, M. Morley, J. Brandimarto, S. Hannenhalli, Y. Hu, E.A. Ashley, W.W. Tang, C.S. Moravec, K.B. Margulies, T.P. Cappola, et al., RNA-Seq identifies novel myocardial gene expression signatures of heart failure, Genomics 105 (2) (2015) 83–89, http://dx.doi.org/10.1016/j.ygeno.2014.12.002.
[73] J. Piñero, J.M. Ramírez-Anguita, J. Saüch-Pitarch, F. Ronzano, E. Centeno, F. Sanz, L.I. Furlong, The DisGeNET knowledge platform for disease genomics: 2019 update, Nucleic Acids Res. 48 (D1) (2020) D845–D855, http://dx.doi.org/10.1093/nar/gkz1021.
[74] M.-C. Lai, M.V. Lombardo, S. Baron-Cohen, Autism, Lancet 383 (9920) (2014) 896–910, http://dx.doi.org/10.1016/s0140-6736(13)61539-1.
[75] F. Chiarotti, A. Venerosi, Epidemiology of autism spectrum disorders: A review of worldwide prevalence estimates since 2014, Brain Sci. 10 (5) (2020) 274, http://dx.doi.org/10.3390/brainsci10050274.
[76] J.P. Gregg, L. Lit, C.A. Baron, I. Hertz-Picciotto, W. Walker, R.A. Davis, L.A. Croen, S. Ozonoff, R. Hansen, I.N. Pessah, et al., Gene expression changes in children with autism, Genomics 91 (1) (2008) 22–29, http://dx.doi.org/10.1016/j.ygeno.2007.09.003.
[77] S.W. Kong, C.D. Collins, Y. Shimizu-Motohashi, I.A. Holm, M.G. Campbell, I.-H. Lee, S.J. Brewster, E. Hanson, H.K. Harris, K.R. Lowe, et al., Characteristics and predictive value of blood transcriptome signature in males with autism spectrum disorders, PLoS One 7 (12) (2012) e49475, http://dx.doi.org/10.1371/journal.pone.0049475.
[78] T. Pramparo, K. Pierce, M.V. Lombardo, C.C. Barnes, S. Marinero, C. Ahrens-Barbeau, S.S. Murray, L. Lopez, R. Xu, E. Courchesne, Prediction of autism by translation and immune/inflammation coexpressed genes in toddlers from pediatric community practices, JAMA Psychiatry 72 (4) (2015) 386–394, http://dx.doi.org/10.1001/jamapsychiatry.2014.3008.
[79] V.H. Gazestani, T. Pramparo, S. Nalabolu, B.P. Kellman, S. Murray, L. Lopez, K. Pierce, E. Courchesne, N.E. Lewis, A perturbed gene network containing PI3k–AKT, RAS–ERK and WNT–β-catenin pathways in leukocytes is linked to ASD genetics and symptom severity, Nature Neurosci. 22 (10) (2019) 1624–1634, http://dx.doi.org/10.1038/s41593-019-0489-x.
[80] C. Riehle, J. Bauersachs, Key inflammatory mechanisms underlying heart failure, Herz 44 (2) (2019) 96–106, http://dx.doi.org/10.1007/s00059-019-4785-8.
[81] A. Huang, Y. Huang, Role of SFRPs in cardiovascular disease, Therapeutic Adv. Chronic Dis. 11 (2020) 2040622320901990, http://dx.doi.org/10.1177/2040622320901990.
[82] M. Pagani, N. Barsotti, A. Bertero, S. Trakoshis, L. Ulysse, A. Locarno, I. Miseviciute, A. De Felice, C. Canella, K. Supekar, et al., mTOR-related synaptic pathology causes autism spectrum disorder-associated functional hyperconnectivity, Nature Commun. 12 (1) (2021) 1–15, http://dx.doi.org/10.1038/s41467-021-26131-z.
[83] H. Zhao, X. Mao, C. Zhu, X. Zou, F. Peng, W. Yang, B. Li, G. Li, T. Ge, R. Cui, Gabaergic system dysfunction in autism spectrum disorders, Front. Cell Dev. Biol. 9 (2021) http://dx.doi.org/10.3389/fcell.2021.781327.
[84] A. Nadeem, S.F. Ahmad, N.O. Al-Harbi, L.Y. Al-Ayadhi, W. Sarawi, S.M. Attia, S.A. Bakheet, S.A. Alqarni, N. Ali, H.M. AsSobeai, Imbalance in pro-inflammatory and anti-inflammatory cytokines milieu in b cells of children with autism, Mol. Immunol. 141 (2022) 297–304, http://dx.doi.org/10.1016/j.molimm.2021.12.009.
[85] V. Audrito, V.G. Messana, S. Deaglio, NAMPT and NAPRT: Two metabolic enzymes with key roles in inflammation, Front. Oncol. 10 (2020) 358, http://dx.doi.org/10.3389/fonc.2020.00358.
[86] S. Duarte-Pereira, O. Fajarda, S. Matos, J. Luís Oliveira, R.M. Silva, NAPRT expression regulation mechanisms: Novel functions predicted by a bioinformatics approach, Genes 12 (12) (2021) 2022, http://dx.doi.org/10.3390/genes12122022.
[87] J.R. Almeida, L.B. Silva, I. Bos, P.J. Visser, J.L. Oliveira, A methodology for cohort harmonisation in multicentre clinical research, Inf. Med. Unlocked 27 (2021) 100760, http://dx.doi.org/10.1016/j.imu.2021.100760.
[88] I. Bos, S. Vos, R. Vandenberghe, P. Scheltens, S. Engelborghs, G. Frisoni, J.L. Molinuevo, A. Wallin, A. Lleó, J. Popp, et al., The EMIF-AD Multimodal Biomarker Discovery study: Design, methods and cohort characteristics, Alzheimer's Res. Therapy 10 (1) (2018) 1–9, http://dx.doi.org/10.1186/s13195-018-0396-5.
[89] J.R. Almeida, D. Pratas, J.L. Oliveira, A semi-automatic methodology for analysing distributed and private biobanks, Comput. Biol. Med. 130 (2021) 104180, http://dx.doi.org/10.1016/j.compbiomed.2020.104180.