Meta-Learning Based Recommendation of Ensemble Methods for Gene Expression Classification

José Gilberto Vasconcelos Júnior and Bruno Feres de Souza
Federal University of Maranhão, São Luís, Brazil
bruno.feres@ufma.br, jgv.junior@gmail.com

André C. P. L. F. de Carvalho
University of São Paulo, São Carlos, Brazil
andre@icmc.usp.br

***Abstract*—For the past decade, microarray technology has been used to provide medical scientists a deeper understanding of diverse molecular phenomena. One of its most prominent applications is the identification of class membership of tissue samples based on their genetic profiles. For this task, Machine Learning algorithms have been commonly employed. In this paper, we present a meta-learning approach that recommends a suitable ensemble method for gene expression classification. Due to the nature of data considered, providing accurate recommendation is not trivial. Despite of that, our approach managed to outperform a baseline method, making room for new research directions.**

***Index Terms*—Gene expression data, Meta-learning, Ensemble**

I. Introduction

Cancer diagnosis is traditionally made through analyzing tumors characteristics, like appearance and origin site. New opportunities for a better understanding of treated tissues was brought up by gene expression profiling tools like microarrays [39]. Microarrays are methods based on hybridization that allow a global view of cells and their gene expression levels [33], thus allowing the measurement of the proteins being produced. This technology has helped researchers to achieve better results on understanding healthy and unhealthy states. Since microarray data are not intuitive to analyze, a common approach is to employ Machine Learning (ML) techniques to describe and predict them [30]. This usage is specifically interesting due to ML algorithms' abilities to extract patterns from data, which helps researches in further tissue classification.

Aiming to improve the overall prediction accuracy of Machine Learning algorithms, ensembles of classifiers have been widely considered. Ensemble learning is a technique that blends a set of base classifiers and combines their outputs in order to obtain the classification for new examples. Ensembles are often more accurate than their individual base classifiers [17], yielding this approach to be increasingly adopted on classification problems. These methods have shown advantages that are particularly interesting on biological classification problems, like mitigating the small sample size problem and good handling of high data dimensionality [41]. By incorporating diverse classifiers (classifiers that make different mistakes), ensembles tend to use the training data more efficiently, reducing the overfitting potential. Several researches, like [2], [13], [14], [18], [20], [27], [40], among others, have successfully applied ensemble methods to classify gene expression data.

Given the variety of methods and applications, choosing a Machine Learning algorithm is not an easy task. A simple way to choose a classification model is to just consider the users familiarity with the model instead of analyzing data structure and characteristics. This selection criteria is likely to lead to results that are not optimal, compromising the whole experimental setup. Another way is the trial-and-error approach, which is highly time-consuming, computationally costly and may still not lead to satisfactory results. Thus, a recommendation system that can provide good model suggestions is highly desirable. An useful approach for that is meta-learning [6].

Meta-learning aims to learn about the performance of learning algorithms in order to enhance future applications. Given an unknown set of data, a meta-learning method extracts characteristics from data and predicts the performance of a set of classification algorithms without running them over the set. Thus, this technique can be used to support algorithm selection for new classification problems. In fact, meta-learning has been applied for algorithm recommendation in many problems (see, [26] and references within).

In this work, our goal is to build a meta-learning approach that recommends an adequate ensemble method to classify gene expression data. Similar approaches were studied for problems from distinct domains [11], [12], [32], where the performance of ML algorithms is expected to exhibit a good amount of variation. When dealing with problems from a single domain, both performance of the ensembles and data characteristics tend to be more homogeneous, challenging the use of meta-learning.

This paper is organized as follows. In Section 2, an overview of popular ensemble methods and some applications to gene expression data is provided. In Section 3, the general architecture of the meta-learning method employed is explained. Section 4 presents the experimental results obtained. Finally, Section 5 draws the conclusions of this work.

II. Ensemble learning for gene expression data

In computational biology, a variety of ensemble methods have been used to perform gene expression analyses, mass

spectrometry-based proteomics, gene-gene interaction identification and prediction of regulatory elements from DNA and protein sequences [41]. Despite recent developments in the area (see, for instance, [21]), Bagging, Boosting and Random Forest remains the most popular methods when dealing with gene expression data and will be reviewed next.

A. Bagging

Bagging (Bootstrap Aggregating) [7] is an ensemble method that generates multiple different instances of the same classification algorithm by training them with distinct training sets. These sets are generated by resampling the original dataset through a bootstrap technique. Each algorithm is then trained with a different resampled training set, improving the distinction among the base predictors [31]. When a new example arrives, the outputs of all base models are combined (usually by majority voting) and the final prediction of the ensemble is calculated.

One of Bagging advantages is that it improves its generalization capability by decreasing model's variance [41]. As seen in [40], this characteristic makes it an interesting way to deal with biological datasets, which may present high variance due to small sampling and biological variability. Besides that, Bagging is also used as a tool to improve other algorithms with good results. In [20], Bagging is used to enhance clustering procedures, achieving substantial improvements on DNA microarrays. Also regarding to gene expression classification, [14] brings up a combination of two popular ensemble methods, using Bagging as a module of Boosting to create a novel algorithm.

B. Boosting

Like Bagging algorithm, Boosting [23] is an ensemble method that resamples the training dataset in order to build distinct classifiers, but in a different way. Boosting works by reweighing every sample of the dataset that trains the base predictors. At first, all the examples in the original dataset receive equal weights and are used to train the first classifier. After that, the algorithm reweighs the dataset, increasing the weight of misclassified samples and reducing the weight of the ones that were correctly predicted [31]. Then, the reweighed set is used to train the second base classifier, and the process goes on repeatedly until all the base classifiers are trained. When facing new examples, the final decision is made through weighted voting, where more accurate classifiers are given greater weights than less accurate classifiers.

Although the classical Boosting algorithm introduced by Freund and Schapire, AdaBoost [22], has shown to not suit very well for raw microarray data [28] due to it's vulnerability to noise, there are variants of this method that can perform well in gene expression classification. In [15] we find a variant of the LogitBoost [24] algorithm that shows consistent improvements on classifying gene expression data by reducing ensemble's sensitivity to noise. This variant is further adopted in [14], where the author combines both Bagging and Boosting to improve prediction accuracy on microarray data. Following the idea of data noise reduction, [2] filters the dataset reducing the number of irrelevant features. More recently, a variant of boosting named XGBoosting [10] has been successfully applied to gene expression data [27], [18], [13], yielding interesting results on this area.

C. Random Forest

Random Forest [8] is a technique that ensembles a set of decision trees built upon random features and bootstraped datasets. For each base tree, training phase is done through one bootstrapped set of data. While being built, the set of candidate features on each split point is randomly generated from the whole variable set. This randomization approach, for both examples and features, provides a model with low bias and low variance [16]. Random Forests have good generalization potential and are also consistent when it comes to avoiding overfitting [42].

In gene expression domain, [16] lists some characteristics of Random Forests that suggest a good suitability for classifying this kind of data. Good predictive performance over noisy data, ruling out the necessity of feature pre-selection, makes this approach suitable for datasets with a high number of features, which is particularly interesting for microarray data. Besides that, random forests are efficient to learn from datasets where the number of features is much bigger than the number of examples. Hence, this method has been widely applied to classify gene expression data, with consistently good results.

Investigating gut microbiome of different races, [9] achieved better results for Random Forests when compared to other classical algorithms like kNN and SVM. Introducing BIRF (Balanced Iterative Random Forest) algorithm, [3] takes advantage of Random Forest's good generalization potential and yields better results for gene selection when compared to SVM variants and Naive Bayes classifiers, specially for unbalanced data. In the task of predicting human MicroRNA target genes, [29] presents the Random Forest based framework RFMirTarget, which outperformed kNN, Naive Bayes, SVM, J48 and General Linear Model.

III. Meta-learning

In the context of this work, we define meta-learning as the application of a learning algorithm to model the relation between the characteristics of learning problems and the performance of a set of algorithms [6]. Specifically, we intend to develop a method that selects a suitable classifier for a given gene expression dataset without the need to actually run any of the available algorithms. The general framework of such meta-learning approach is depicted in Figure 1.

The process begins with the acquisition of an appropriate set of problems that are representative of those for which the subsequent recommendation will be made. Then, two steps are applied to each element in the Data Repository: the extraction of data characteristics, according to some predefined measures, and the evaluation of a set of algorithms. Ideally, the characterization of the problems must be predictive of the

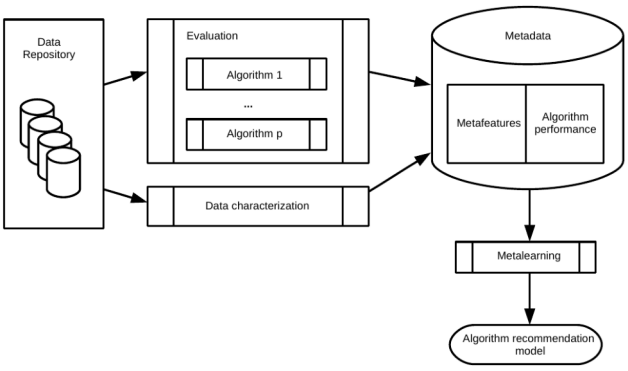

Fig. 1. Meta-learning general framework [6]

behavior of the algorithms. By associating these two information for each problem, we obtain a meta-example, formed by input meta-attributes and target meta-class, respectively. The set of available meta-examples is called the meta-data. In order to induce the mapping between the input meta-attributes and the target meta-class, an ML algorithm, referred to as a meta-learner, is applied. Through it, one can use the meta-knowledge obtained from the learning process to provide the recommendation of algorithms in the context of meta-learning.

In this work, data characterization is done by the Landmarking approach [34]. As far as we know, it has not yet been employed to characterize gene expression data. Basically, it consists of using the performance of simple algorithms (the so called landmarkers) to describe a problem and correlating this information with the performance of more complex learning algorithms. The rationale for this data characterization approach is the assumption that problems that perform similarly on the ladmarkers space will behave similarly on more advanced classification methods. Of course, it is mandatory that landmarkers are computationally efficient, specially when considering microarray data. Here, we considered 6 landmarkers proposed in [4]: Naive Bayes, Linear Discriminant, One Nearest Neighbor, Decision Node, Randomly Chosen Node and Worst Node. Thus, the input meta-features of each microarray problem consists of the performances of such landmarkers, as measured by an efficient 5 fold stratified cross validation process.

The target meta-class represents the performance of the ML algorithms on the dataset. Plenty of performance metrics can be used to evaluate supervised algorithms, such as accuracy, Area Under a Curve (AUC), F-score, mean square error, etc. For most case, accuracy remains as the standard metric to use. But in the gene expression domain, AUC could be a more suitable option [38], mainly due to its ability deal with unbalanced class distribution and different misclassification costs. Thus, in this work, we will employ AUC as performance metric. The target meta-class of each microarray problem is the ensemble method whose performance is the best, as measured by the AUC values embedded into a stratified cross validation scheme with 10 folds. Now, we are facing here a meta-classification problem.

IV. Experiments

The main goal of the experiments conducted here is to assess the performance of our meta-learning approach to recommend ensemble methods for gene expression classification data. For such, the framework presented in Section III was employed. Next, we present the experimental setup used and the results obtained.

A. Experimental setup

The problems used to generate the meta-data came from 49 publicly available microarray datasets. They are all related to cancer diagnostics. Mainly, the task is either discriminating between normal and tumor cases or among different types of tumor. As usual for the gene expression domain, there is a disproportional rate between data dimensionality and number of examples, which makes classification more difficult. A full description of the datasets can be retrieved elsewhere [37].

For base level learning, a representative of each ensemble approach previously commented was chosen. For Bagging, the ensemble components are based on C5.0 decision trees [25], which extends the seminal work of [35]. Following the suggestion of [19], 50 trees were bagged. For Boosting, XGBoosting algorithm [10] was considered. It is an implementation of gradient boosting that focus on computational efficiency and predictive accuracy. Here, we considered a maximum of 50 trees [19]. Finally, Random Forests have also been employed. Over the years, they have delivered state-of-the-art performance in gene expression analysis [42]. For simplicity, all algorithms have used the default parameters provided by the employed R packages (`C50`, `xgboost` and `randomForests`, respectively).

For meta-learning, we have selected 4 algorithms, from distinct classification families [5]: Support Vector Machines (SVM), k Nearest Neighbours (kNN), Decision Trees (DT) and Naive Bayes (NB). Since they have very different biases, we expected to have a fairly comprehensive picture of the behavior of our approach. We note that such algorithms have already been used as meta-learners before, with varying degree of success [1]. They were implemented by the following R packages, respectively: `e1071` (with `kernel='linear'`), `class` (with textttk=1), `C50` (with `winnow=TRUE`) and `caret`.

After a meta-model is created using a given meta-learner applied to the meta-data, it is necessary to produce evidence to the user that meta-learning is able to generate accurate predictions. The approach used here is to use Leave-one-out Cross Validation (LOOCV), which iteratively, for each meta-example, computes the accuracy of the prediction using a meta-model obtained on all the remaining meta-examples [6].

B. Experimental results

In order to better evaluate the results presented here, Figure 2 exhibits the distribution of the best performing ensemble approaches over the 49 datasets. It indicates the frequency

each algorithm serves as target meta-class for a given meta-example. Note that Bagging and Random Forest present similar frequency, while Boosting is less frequent. Such distribution represents a somehow unbalanced multiclass setup.

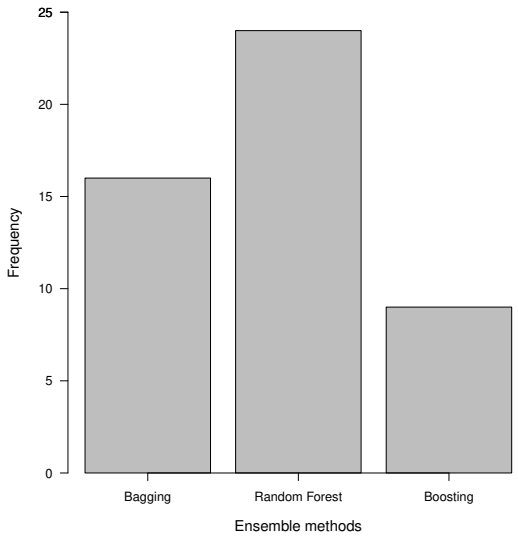

Fig. 2. Distribution of best performing ensemble approaches over the 49 datasets

Table I summarizes the classification performance of the proposed meta-learning approach. It provides mean and standard deviation of accuracies for 4 meta-learners using the LOOCV process. In order to determine whether the performance of a particular recommended ensemble can be regarded as high or not, we also exhibits the performance of a baseline method, called Default. It simply predicts the most frequent class every time. With an accuracy of 0.61, SVM presents the best performance. In fact, it is the only approach that convincingly outperformed the Default method. Naive Bayes and kNN exhibited poor results, while Decision Tree is just average. Redundancy and/or lack of relevant information in Landmarking characterization may have impacted greatly Naive Bayes and kNN. SVM and Decision tree seems more robust, specially the former. This is expected, since they have internal mechanisms for weighting/selection of features during learning phase. Note that the high standard deviation of all meta-learners are due to the LOOCV procedure.

SVM	**NB**	**kNN**	**DT**	**Default**
0.61/0.49	0.44/0.50	0.48/0.50	0.51/0.50	0.48/0.50

TABLE I
PERFORMANCE OF 4 META-LEARNERS AND DEFAULT METHOD. MEAN AND STANDARD DEVIATION OF ACCURACIES ARE PROVIDED.

In order to further investigate the performance of our approach, we will focus on SVM algorithm from now on. Table II presents its confusion matrix. One can see that SVM is able to discriminate relatively well between classes Bagging and Random Forest. In fact, if we recast our meta-classification problem into a binary one, with only those 2 meta-classes, SVM would be able to separate Bagging and Random Forest meta-examples with a mean accuracy of 0.69, while Default method would present a mean accuracy of 0.61. Now, considering the minority class Boosting, Table II clearly shows that SVM never predicts it. Is may indicate that data characterization was not informative enough to allow finer distinction among the ensemble methods or the Boosting algorithm used here presented an anomalous behavior.

		Predicted		
		Bagging	Random Forest	Boosting
Actual	Bagging	10	6	0
	Random Forest	4	20	0
	Boosting	2	7	0

TABLE II
CONFUSION MATRIX FOR THE SVM ALGORITHM.

Such results show how tricky predicting ensemble methods for gene expression data classification can be. In domains like this, where few data samples are available, classification algorithms tend to present comparable results [36], masking the true differences in the predictive accuracy of some algorithms. Figure 3 shows that this issue may be affecting, to a certain degree, the results obtained. It shows the boxplots of the distribution of AUC values for Bagging, Random Forest and Boosting on the 49 datasets used here. As can been seen, Bagging and Random Forest present similar boxplots, suggesting they perform similarly. Boosting seems to perform worse than its contenders for the problems at hand. Nevertheless, it can be expected that, as more patient data become available, classification problems will reveal a more complex structure and the differences in performance of the algorithms will increase, making meta-learning on gene expression domain easier.

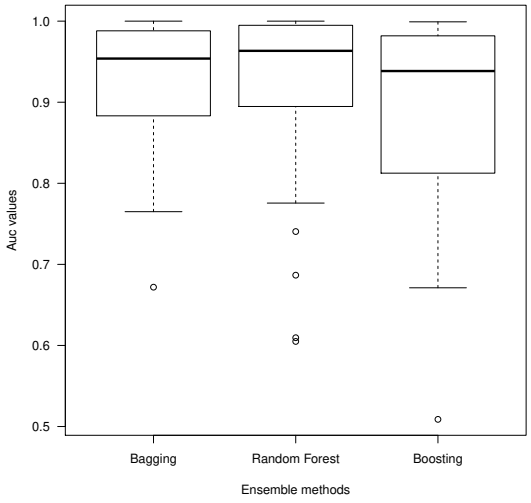

Fig. 3. Distribution of AUC values for the 3 ensemble methods on 49 datasets.

So far, we have presented results regarding the performance of the proposed meta-learning approach. However, the quality

of a given recommendation does not contain any information about the performance of the ensemble method on a new gene expression dataset. In order to shed some light on the matter, Figure 4 exhibits, for each of the 49 datasets, the AUC values that would be expected in 3 situations: the best ensemble method is used, the recommendation provided by the SVM meta-learner is used and the default ensemble method is used. To improve readability, the datasets are sorted using the performance of the best ensemble method. As can be seen, SVM predictions yields base level performances close to the best methods, with minor deviations. On the other hand, when default ensemble method is employed, errors tend to be much more severe in some cases. These results indicate our meta-learning approach yield good base level performance.

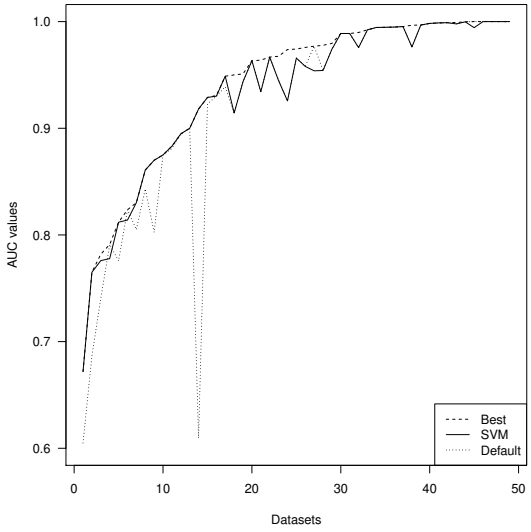

Fig. 4. AUC values for base learning on 49 datasets in three recommendation scenarios.

V. Conclusion

In this paper, we presented a meta-learning approach to recommend ensemble methods for gene expression classification. In such scenario, two main difficulties may raise: first, due to small sample size, performances of ML algorithms over a microarray dataset tend to be more similar and second, since problems come from the same domain, data are more homogeneous, which can impair characterization methods to generate discriminating meta-feaures.

Despite of that, the experiments conducted here support the application of meta-learning for the task. Considering a setup with 49 publicly available microarray datasets and 3 popular ensemble methods, our approach was able to outperform the baseline method when SVM was used as meta-learner. Analyzing the results, one can see that there is room for improvements. Specifically, a refinement on data characterization and performance assessment is welcome and will be addressed in a future work. Besides that, we plan to deal with feature selection and parameter optimization issues, in both base level learning and in meta-learning.

References

[1] *2018 International Joint Conference on Neural Networks, IJCNN 2018, Rio de Janeiro, Brazil, July 8-13, 2018*. IEEE, 2018.

[2] Tan AC and Gilbert D. Ensemble machine learning on gene expression data for cancer classification. *Appl Bioinformatics*, 2(3):75–83, 2003.

[3] Ali Anaissi, Paul J. Kennedy, Madhu Goyal, and Daniel R. Catchpoole. A balanced iterative random forest for gene selection from microarray data. *BMC Bioinformatics*, 14(1):261, Aug 2013.

[4] Hilan Bensusan and Christophe Giraud-Carrier. Casa batlo is in passeig de gracia or landmarking the expertise space. In *Proceedings of the ECML'2000 workshop on Meta-Learning: Building Automatic Advice Strategies for Model Selection and Method Combination*, pages 29–47. ECML'2000, June 2000.

[5] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[6] Pavel Brazdil, Christophe Giraud-Carrier, Carlos Soares, and R. Vilalta. *Metalearning: Applications to Data Mining*. Cognitive Technologies. Springer, January 2009.

[7] Leo Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[8] Leo Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, October 2001.

[9] Lei Chen, Yu-Hang Zhang, Tao Huang, and Yu-Dong Cai. Gene expression profiling gut microbiota in different races of humans. *Scientific Reports*, 6:23075, 03 2016.

[10] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 785–794, New York, NY, USA, 2016. ACM.

[11] Rafael M.O. Cruz, Robert Sabourin, and George D.C. Cavalcanti. Meta-des.oracle: Meta-learning and feature selection for dynamic ensemble selection. *Information Fusion*, 38:84 – 103, 2017.

[12] Robercy Alves da Silva, Anne Magaly de Paula Canuto, João Carlos Xavier Junior, and Teresa Bernarda Ludermir. Using meta-learning in the selection of the combination method of a classifier ensemble. In *2018 International Joint Conference on Neural Networks, IJCNN 2018, Rio de Janeiro, Brazil, July 8-13, 2018* [1], pages 1–8.

[13] Yang Y. Lu ProfileKarine Le Roch William Noble David F. Read, Kate Cook. Predicting gene expression in the human malaria parasite plasmodium falciparum. *bioRxiv*, 1:1–18, February 2019.

[14] Marcel Dettling. BagBoosting for tumor classification with gene expression data. *Bioinformatics*, 20(18):3583–3593, 10 2004.

[15] Marcel Dettling and Peter Bhlmann. Boosting for tumor classification with gene expression data. *Bioinformatics (Oxford, England)*, 19:1061–9, 07 2003.

[16] Ramón Díaz-Uriarte and Sara Alvarez de Andrés. Gene selection and classification of microarray data using random forest. *BMC Bioinformatics*, 7(1):3, Jan 2006.

[17] Thomas G. Dietterich. Ensemble methods in machine learning. In *Proceedings of the First International Workshop on Multiple Classifier Systems*, MCS '00, pages 1–15, London, UK, UK, 2000. Springer-Verlag.

[18] Georgios N. Dimitrakopoulos, Aristidis G. Vrahatis, Vassilis P. Plagianakos, and Kyriakos N. Sgarbas. Pathway analysis using xgboost classification in biomedical data. In *SETN*, 2018.

[19] S. Dudoit, J. Fridlyand, and T. P. Speed. Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association*, 97(457):77–87, 2002.

[20] Sandrine Dudoit and Jane Fridlyand. Bagging to improve the accuracy of a clustering procedure. *Bioinformatics*, 19 9:1090–9, 2003.

[21] A. Espichan and E. Villanueva. A novel ensemble method for high-dimensional genomic data classification. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 2229–2236, Dec 2018.

[22] Y. Freund and R. E. Schapire. A decision-theoretic generalization of online learning and an application to boosting. *Journal of Computer and System Sciences*, 55:119–139, 1999.

[23] Yoav Freund and Robert E. Schapire. Experiments with a new boosting algorithm. In *Proceedings of the Thirteenth International Conference on International Conference on Machine Learning*, ICML'96, pages 148–156, San Francisco, CA, USA, 1996. Morgan Kaufmann Publishers Inc.

[24] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Additive logistic regression: A statistical view of boosting. *The Annals of Statistics*, 28:337–407, 04 2000.

[25] Max Kuhn and Kjell Johnson. Applied predictive modeling, 2013.

[26] Christiane Lemke, Marcin Budka, and Bogdan Gabrys. Metalearning: a survey of trends and technologies. *Artificial Intelligence Review*, (1):117130, 06 2015.

[27] Yuanyuan Li, Kai Kang, Juno Krahn, Nicole Croutwater, Kevin Lee, David M. Umbach, and Leping Li. A comprehensive genomic pan-cancer classification using the cancer genome atlas gene expression data. *BMC Genomics*, 18, 12 2017.

[28] Philip M. Long and Vinsensius Berlian Vega. Boosting and microarray data. *Machine Learning*, 52(1):31–44, Jul 2003.

[29] M. R. Mendoza, G. C. da Fonseca, G. Loss-Morais, R. Alves, R. Margis, and A. L. C. Bazzan. RFMirTarget: Predicting Human MicroRNA Target Genes with a Random Forest Classifier. *PLoS ONE*, 8:e70153, July 2013.

[30] Tom Mitchell. *Machine Learning*. McGraw-Hill Science/Engineering/Math, March 1997.

[31] Sajid Nagi and Dhruba K Bhattacharyya. Classification of microarray cancer data using ensemble approach. *Network Modeling Analysis in Health Informatics and Bioinformatics*, 2, 09 2013.

[32] D. S. C. Nascimento, A. M. P. Canuto, and A. L. V. Coelho. An empirical analysis of meta-learning for the automatic choice of architecture and components in ensemble systems. In *2014 Brazilian Conference on Intelligent Systems*, pages 1–6, Oct 2014.

[33] Jayadeep Pati. Gene expression analysis for early lung cancer prediction using machine learning techniques: An eco-genomics approach. *IEEE Access*, 7:4232–4238, 2019.

[34] Bernhard Pfahringer, Hilan Bensusan, and Christophe Giraud-Carrier. Meta-learning by landmarking various learning algorithms. In *Proceedings of the Seventeenth International Conference on Machine Learning, ICML'2000*, pages 743–750. Morgan Kaufmann, Junho 2000.

[35] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

[36] R.L. Somorjai, B. Dolenko, and R. Baumgartner. Class prediction and discovery using gene microarray and proteomics mass spectroscopy data: curses, caveats, cautions. *Bioinformatics*, 19(12):1484–91, August 2003.

[37] Bruno Feres Souza. *Meta-aprendizagem aplicada classificao de dados de expresso głnica*. Tese, Instituto de Ciłncias Matemticas e de Computao da Universidade de So Paulo, So Carlos - SP, October 2010.

[38] Alexander Statnikov, Lily Wang, and Constantin F. Aliferis. A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification. *BMC Bioinformatics*, 9:319–329, 2008.

[39] David Tax and Robert Duin. Support vector data description. *Machine Learning*, 54:45–66, 01 2004.

[40] Giorgio Valentini, Marco Muselli, Francesca Ruffino, and Infm Istituto Nazionale. Bagged ensembles of support vector machines for gene expression data analysis. In *Proceedings of the International Joint Conference on Neural Networks*, pages 20–24.

[41] Pengyi Yang, Jean Yang, Bing B Zhou, and Albert Zomaya. A review of ensemble methods in bioinformatics. *Current Bioinformatics*, 5, 12 2010.

[42] Q. Yanjun. Random forest for bioinformatics. In *Ensemble Machine Learning*, pages 307–323. Springer, New York, NY, 2012.