University of Crete
Faculty of Health Sciences, School of Medicine
MSc in Bioinformatics
Methods in Bioinformatics, BC203

Semester project: Construction of Machine learning pipeline for classification and its application to transcriptomics data.

Angelos Georgios Mathioudakis*, Christos Kitsoulis*

* Master programme in Bioinformatics, School of Medicine, Faculty of Health Sciences, University of Crete, Heraklion, Greece.

Abstract

The main goal of this project is to construct a machine learning pipeline for data classification, which can distinctively predict the age class and if someone has followed shortly a resistance exercise programme, based on the expression levels of specific genes. This genetic profile is selected from common candidate genes that are connected to physical activity, as mentioned in scientific journals [1-4, 8-9]. For this implementation we chose the following dataset *GDS5218* from NCBI GEO which contains curated gene expression data coming from biopsy samples of muscle vastus lateralis. Two cohorts of adults were studied, some of them who followed a 12-week resistance exercise programme while the rest had the role of control. For this purpose we intend to try multiple classification methods, such as KNN, Support Vector Machines, Decision Trees, Neural Networks etc, for the aforementioned dataset and choose the most precise or some combination. After the main part, if there are any accurate results, our will is to generalize our implementation by using data derived from other studies following the same protocols. If such predictions are accurate, could this constitute an indication of correlation between genetic profile and muscle adaptations to resistance exercise?

Introduction

Continually new technologies in biological research as well as their applications make it easier and cheaper to generate much more variety of data every day [5]. The exponential growth of biological data, that has almost overcame the zettabyte unit, raises two major problems: the first one has to do with the storage and management of data, while the second with the extraction of significant knowledge from these heterogeneous data. The second is a challenging problem of computational biology, which demands the development of methods and tools in order to transform sparse data into biologically

usefull knowledge [6]. Machine learning (ML) is a constantly expanding part of computer science and focuses on performing data-driven predictions by auto-impoving through learning experience. Biological data are combining more and more with ML approaches beacause of several applications of the second. “Classification” has been a niche topic for research in fields of ML in recent years because it has found a huge applicability in -omics data [7]. Such a challenge is to make predictions on predefined categories about the situation of a patient based on the genetic profile she/he is carrying, retrieved from measurements of next-generation sequencing (NGS), such as RNA-seq technique.

An assay of similar nature was arised in the limits of “Methods in bioinformatics” course about transcriptomics data coming from *GDS5218* dataset with title: “Resistance exercise effect on skeletal muscles of young and old adults”. Authors in their investigation examined the effects of resistance exercise and age on the human skeletal muscle transcriptome [4]. Studies in the field of human exercise genomics have identified single nucleotide polymorphisms (SNPs) linked to elite athletes as well as a large number of molecular signalling mechanisms have already been linked to the adaptations in the skeletal muscle by regular exercise [3]. Others, which involves following the inheritance pattern of specific phenotypes in nuclear families, suggest that major genes contribute to a substantial fraction of phenotypic variance in at least some performance-related traits. For example, single genes are reported to account for more than 40% of the variance in oxygen uptake at the ventilatory threshold [9 and referneces therein]. Many of the studies provide evidence for the association of candidate genes with specific phenotypes of physical activity [1-3, 8-9 and references therein], as for the resistance exercise which is a subject of interest in this study. However, an -omics scale glance on molecular exercise physiology is needed, in order to define the relation between specific genes (or sets of genes) to different types of exercise. One of the most important queries that has to be answered concerns the prediction ability of resistance exercise-based situation and age of a person, the circumstances under which a machine learning classifier identifies discrete patterns in genetic expression profiles and the possible correlation of profile and muscle transcriptome response to exercise.

Results

After an attentive curation and filtering to the initial dataset (more to be mentioned in *Materials and Methods* section) and applying our ML pipeline to it, we came to the results, which are presented below. It has to be noticed, foremost, that some results may seem odd and that comes as a consequence of dataset’s the small size. Further explasion on this, it is going to be discussed later on.

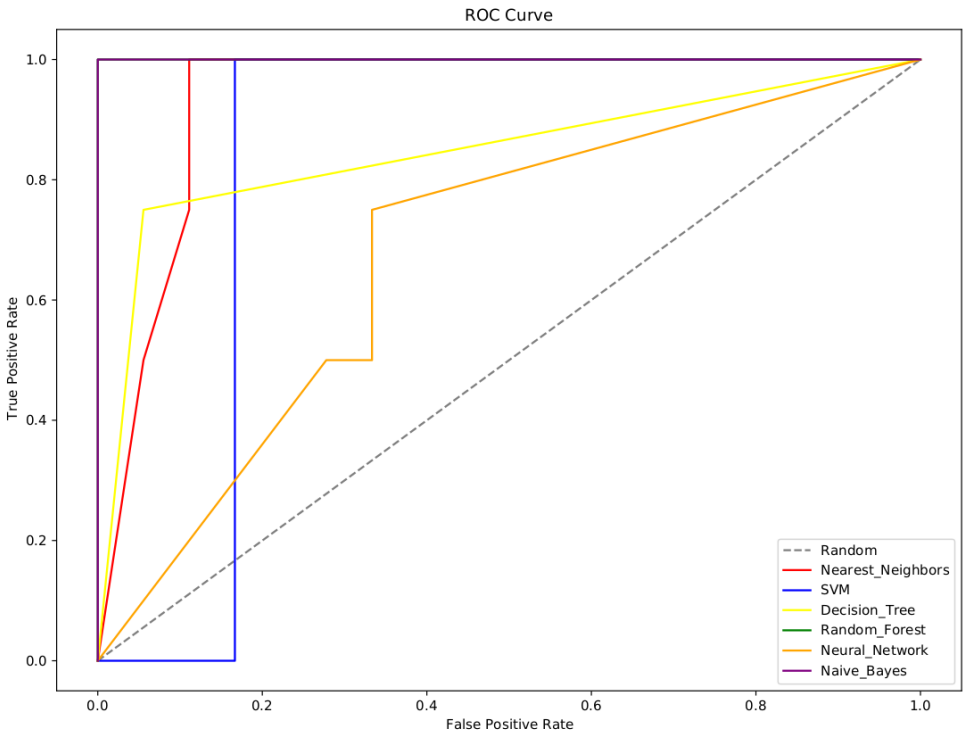

Figure 1. ROC curve plot for default-parameter algorithms' FPR and TPR. Every color represents the line of correlation between false positive and true positive rate, while the one the randomized threshold.

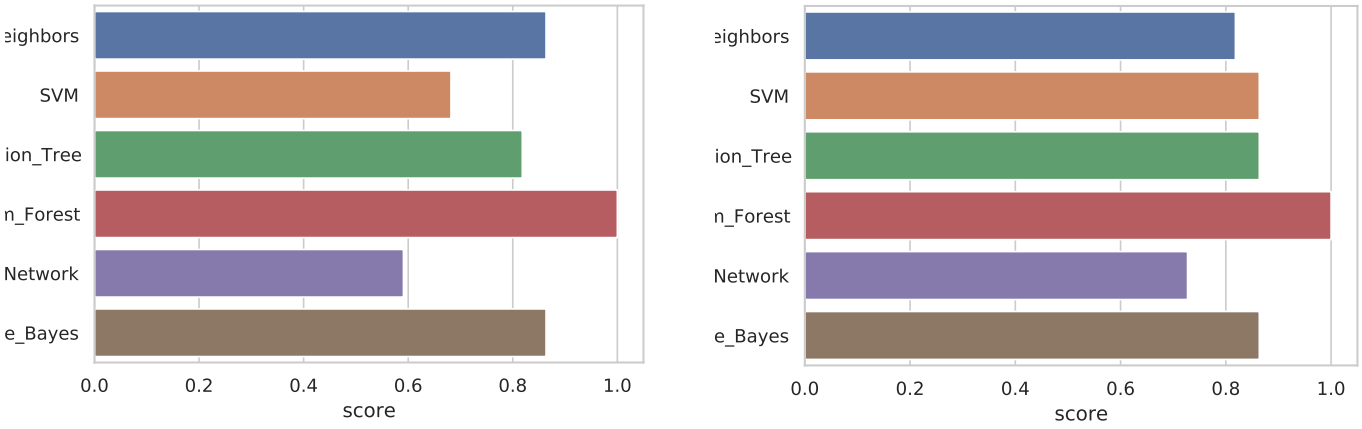

Figure 2. Accuracy bar plots for each classification algorithm used in pipeline (4 predefined labels), before and after hyperparameter tunning. One the left, they are represented the accuracy scores from algorithms trained in default parameters while on the right, after the training procegure with the best parameters, each. Every algorithm is colored differently.

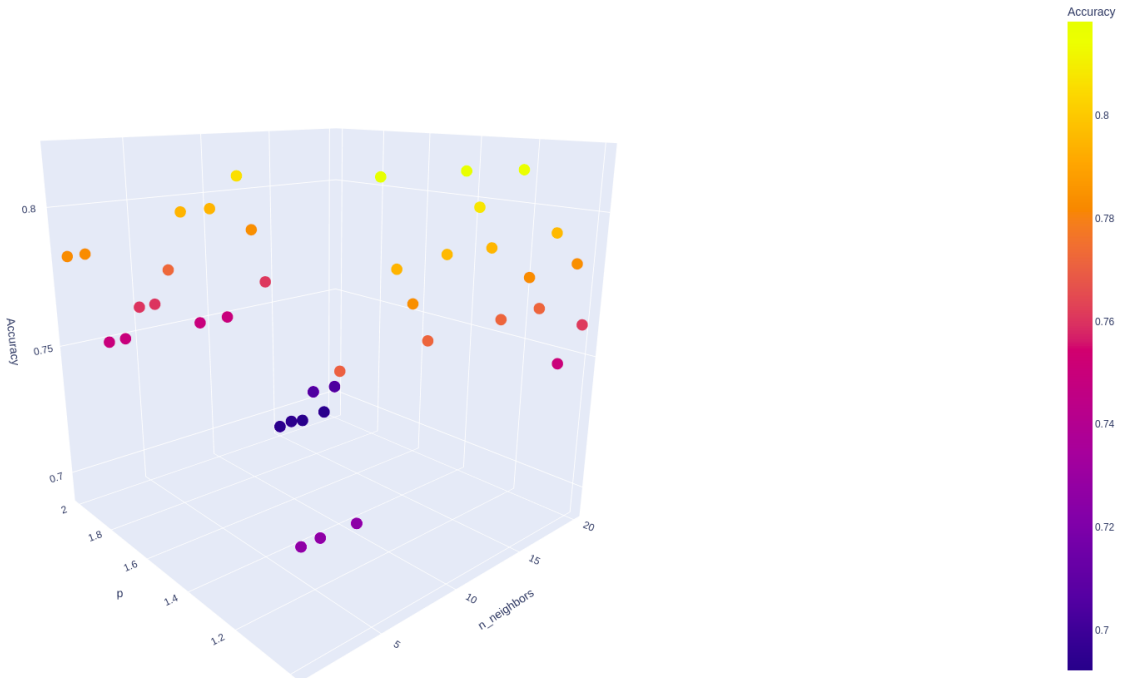

Figure 3. 3D Accuracy plot for KNN algorithm as paramters change during hyperparameter tunning. Every point is colored based on the range of accuracy. The point that matches the highest accuracy suggests the best parameters for the model. In this case: number of neighbors = 5, distance = euclidean, weights = by distance.

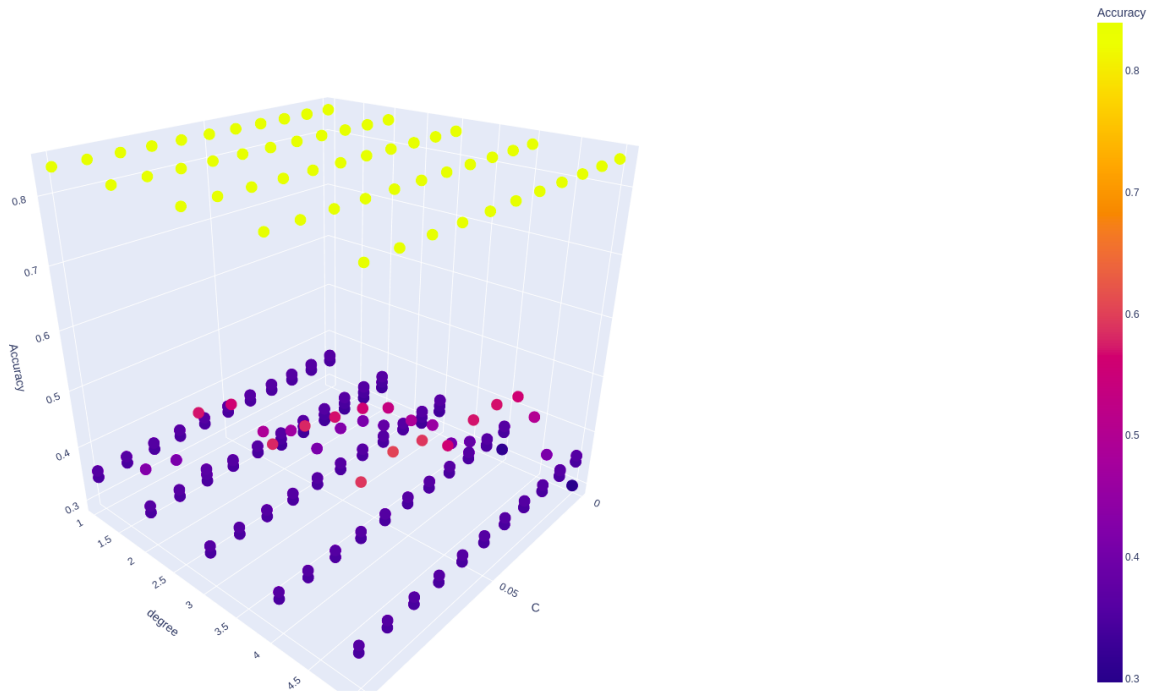

Figure 4. 3D Accuracy plot for SVM by the change of parameters in search of the best combination. The highest accuracy point matches the following parameters: C = 0.001, degree = 1 and kernel = linear.

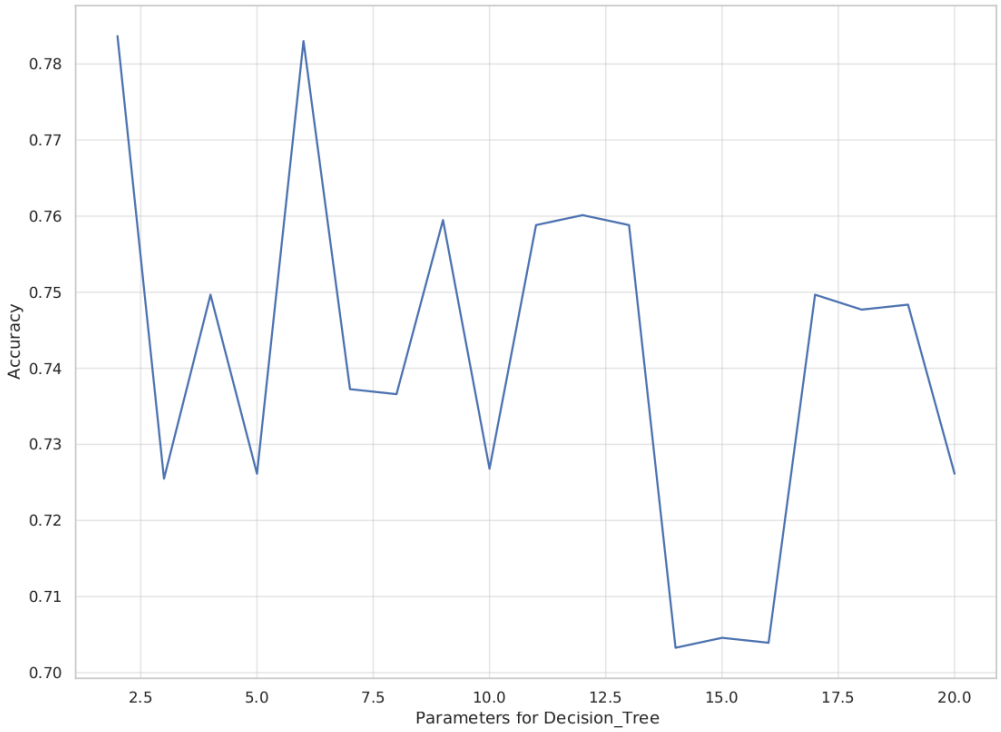

Figure 5. Plot of accuracy by the maximum depth of Decision Tree. The highest value of accuracy during the hyperparameter tunning step corresponds to maximum depth = 2

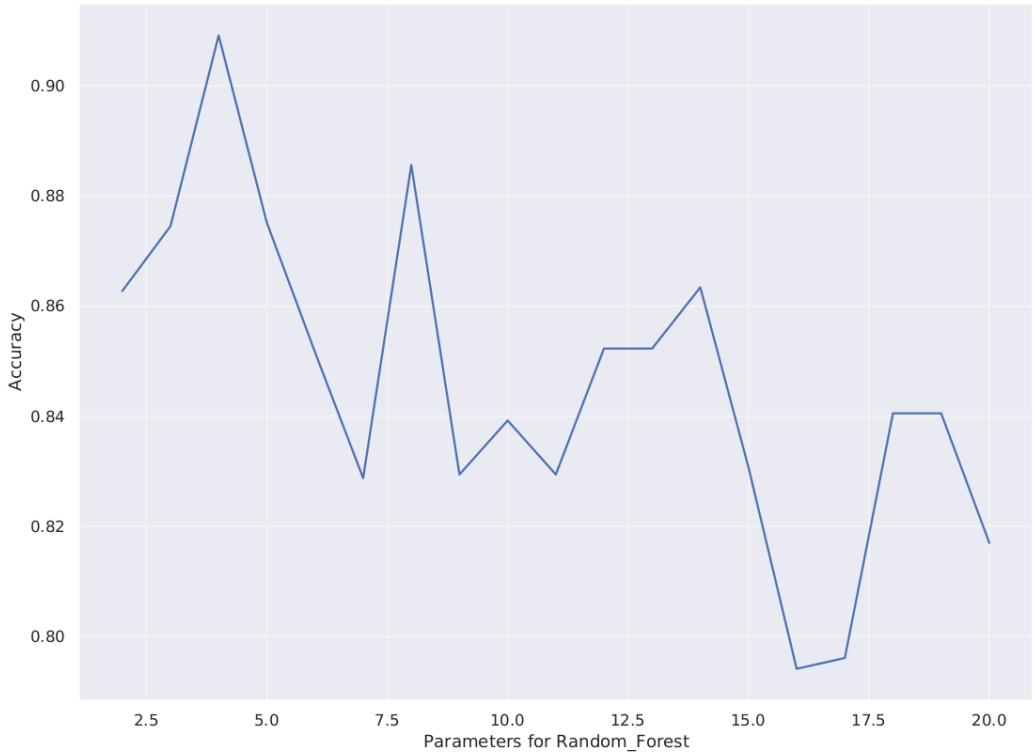

Figure 6. Plot of accuracy by the maximum depth of Random Forest algorithm. In the maximum depth of 4 there is the peak of accuracy value corresponding the best parameter.

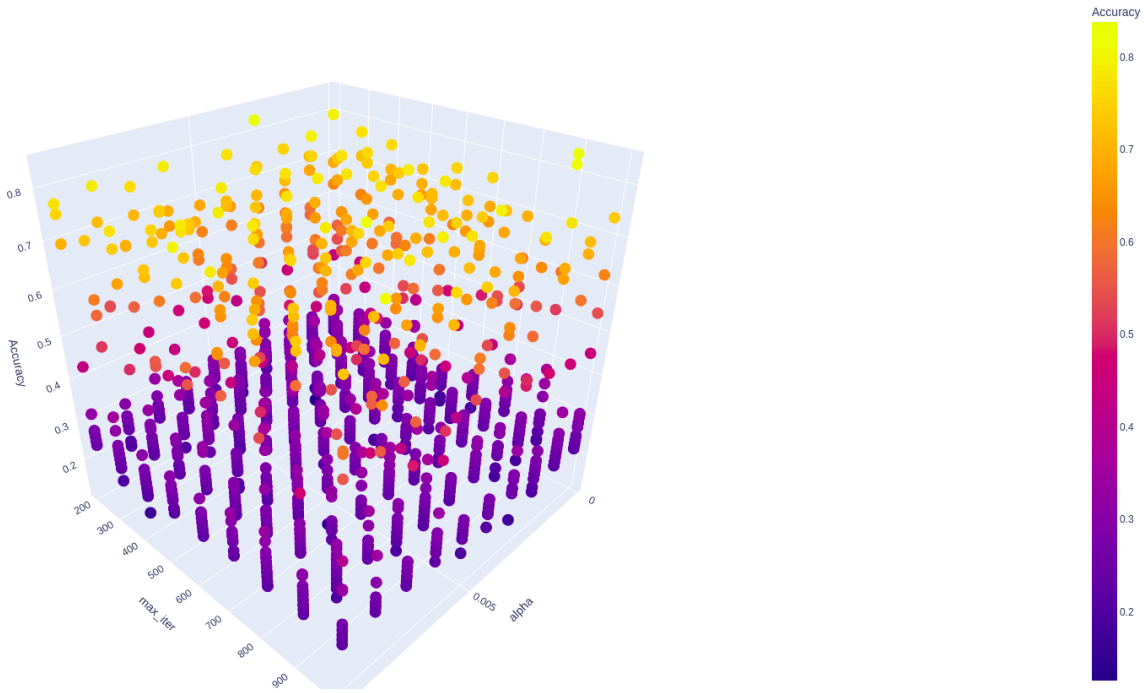

Figure 7. 3D Accuracy plot for Neural network by the combination of parameters α and maximum iterations. The points seems like they are distributed in three different layers based on parameters combination. The highest accuracy matches the following parameters: α = 0.0001, number-size of hidden layers = 1-100, maximum iterations = 900.

Materials and Methods

The curated dataset which was selected from NCBI GEO is *GDS5218* and contains transcriptomic data from biopsies of vastus lateralis, the largest and most powerful part of quadriceps femoris muscle (commonly known as quadriceps). Two cohorts of young (24-25 years) and old (78-84 years) adults were studied during the performance of a specific 12-week resistance exercise programme. A group followed the programme, and sample biopsies were obtained pre- and 4 hours post-workout in conjunction with the first and last training session, while the B group had the role of basal untrained state [4]. Transcriptome profiling was performed using HG U133 Plus 2.0 Arrays. This study concluded in 661 genes that are affected by resistance exercise synergistic to gains in muscle size and strength [4]. The selected transcriptomic profile that we finally used results from the common genes between these 661 proposed genes [4], a number of 558 genes referring as candidates for resistance exercise [3], plus some extra that are already known to have some relationship with athletic performance in general [1-

2,8-9]. At last, this profile consists of 69 genes. Because the initial dataset had multiple measurements of some genes, in order to select only one representative we used a filtering method based on their mean and kept the ones with the highest mean per feature. While, the final dataset was ready (110 samples, 69 genes), was applied on the classification pipeline for training and prediction of the predefined labels (young_PreWorkout, young_PostWorkout, old_PreWorkout, old_PostWorkout). The constructed pipeline was coded in programming language python (version 3.9) and with the use of sklearn functions [10] for the classification algorithms. The final pipeline consists of the following steps and is represented in the figure below (Fig. 8).

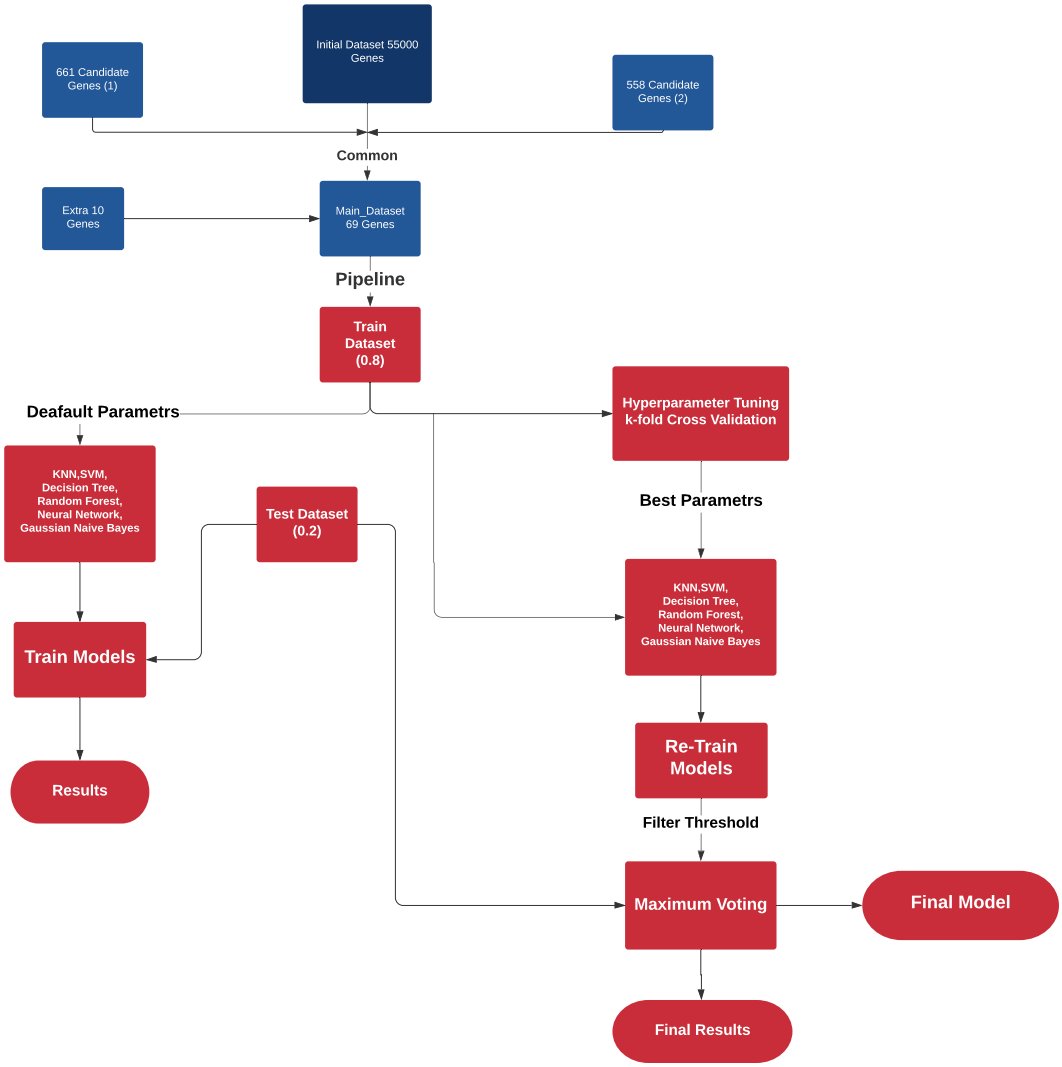

Figure 8. Schematic presentation of pipeline steps from selected dataset to the final trained prediction model [10].

In the step of cross validation, alternatively we changed to the k-fold cross validation (with k=5) method. Thus, the whole procedure, from the first to last step, was performed simultaneously for six known algorithms (Fig. 8) and the result that comes from the pipeline is a model of these algorithms that can be parameterized to satisfiy the needs of the user and its prediction score. For this final step that was added, known as maximum voting method, the user has the ability to select a threshold for prediction accuracy to select the most precise algorithms, after the hyperparameter tunning step, and get a combined model. In our case, this threshold was set to 0.75. More details about the pipeline are contained in the supplementary files (.py).

Some further information about the basic algorithms that are contained in pipeline is presented below:

1. K-Nearest Neighbor (KNN) is a simple algorithm that can be used to solve both classification and regression problems. It is easy to implement and understand, but has a major drawback of becoming significantly slow as the size of data grows. While it is accurate, lacks precision compared to better models.

2. Support Vector Machine (SVM) is a state-of-art algorithm that is used for classification and regression. In addition to linear classification, SVMs can efficiently perform a non-linear classification using the kernel trick, implicitly mapping their inputs into higher-dimensional space. It basically, "draws" margins between the classes. It is effective in cases where the number of dimensions is greater than of samples, but not really suitable for large datasets or noisy ones.

3. Decision Trees is a graph representation of choices and their results in form of a tree. The nodes are an event or choice while the edges represent the decision rule or condition. Each node represents attributes in a group that is to be classified and each branch represents a value that the node can take. Decision trees are easy to interpret, can handle both categorical and continuous data and works well with large datasets. On the other hand, are prone to overfitting, cannot guarantee optimal trees and give lower predictions compared to other methods.

4. Random Forest is yet another powerful algorithm that allows quick identification of significant information from vast datasets, either linear or not. The finest advantage of random forest is that relies on collecting various decision trees to arrive at any solution. On contrary to decision trees, has lower risk of overfitting and has a better accuracy of classification. However, random forests are slow on training.

5. Neural Network is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates. It works by splitting the

problem into a layered network of simpler elements. NNs can be used for both classification and regression problems for large datasets. The predicted output of a neural network is compared with the actual output and based on error, parameters are changed, and then fed into the neural network again. Once is trained, the predictions become pretty fast. Nevertheless, NNs could be black boxes, computationally expensive as well as they mostly lack on generalization because train model relies closely in training data.

6. Naive Bayes is a classification technique based on Bayes theorem with an assumption of independence among predictions. Naïve Bayes mainly targets the text classification industry. It is mainly used for clustering and classification purpose depends on the conditional probability of happening. It is a very fast algorithm, of high accuracy on predictions and could perform well with less training data compared to others. As negatives, the assumption of independence may seem great in theory, but it may be tricky to find independent features in real life though.

Conclusions

The necessity of having methods which could turn hetereogenous data to knowledge is more than urgent and have triggered many scientists to search new and more precise tools to this direction. Among others, machine learning classification could give a stronger push as the data are becoming more and more available.

In our case, the ML pipeline seems to work well for the applied dataset. Thus, it may seem odd that the accuracy rate of an algorithm, for example KNN, is lower after the hyperparater tunning than before with default parameters (Fig. 2), but this is logical. Because of the small size of dataset and the even smaller of training and test parts, just a wrong prediction could lead to a quite big change in accuracy rates. In general, after the search of best parameters and re-training each model, the prediction accuracy was imporoved and was upper than 0.75. The ROC curve plot (True positive rate vs False positive rate, Fig. 1) for default parameters algorithms in the first step shows the performance of every classification model in various thresholds, and in all cases there is no strong indication of randomized results. The final model which is formed by all algorithms in our case (while threshold was to 0.75), gave a prediction accuracy rate of almost 0.90. After examining the wrong predictions in all steps (see supplementary files and confusion matrices), all the algorithms and the combination of them prone to make mistakes in age and not the situation (Pre or PostWorkout). Based on this gene expression profile of 69 selected genes, the final model has the ability to distinct those two situation pretty well while it

makes few mistakes about the age. Because of the high accuracy rate of pipeline, this could mean an indication of how skeletal muscle transcriptome profile responds to the resistance exercise and the impact of age to it.

References

1. Bray MS, Hagberg JM, Perusse L, et al. The Human Gene Map for Performance and Health-Related Fitness Phenotypes. Med Sci Sports Exerc. 2009; 41(1): 34-72.

2. Lavin KM, Bell MB, McAdam JS, et al. Muscle transcriptional networks linked to resistance exercise training hypertrophic response heterogeneity. Physiol genomics. 2021; 53: 206–221.

3. Pacheco C, Felipe S, Soares M, et al. A compendium of physical exercise-related human genes: an 'omic scale analysis. Biol Sport. 2018; 35(1): 3-11.

4. Raue U, Trappe TA, Estrem ST, et al. Transcriptome signature of resistance exercise adaptations: mixed muscle and fiber type specific profiles in young and old adults. J Appl Physiol (1985). 2012; 112(10): 1625-36.

5. Cook C, Bergman M, Finn R, et al. The European Bioinformatics Institute in 2016: Data growth and integration. Nucleic Acids Research. 2015*;* 44(1): 20-26.

6. Larrañaga P, Calvo B, Santana R, Bielza C, et al. Machine learning in bioinformatics. Briefings in Bioinformatics. *2006;* 7(1): 86-112.

7. Jabeen A, Ahmad N and Raza. Machine Learning-based state-of-the-art methods for the classification of RNA-Seq data. BioRxiv. 2017.

8. MacArthur DG & North KN. Genes and human elite athletic performance. In Human Genetics. 2005; 116(5): 331-339.

9. Booth F, Chakravarthy M and Spangenburg E. Exercise and gene expression: physiological regulation of the human genome through physical activity. The Journal of Physiology. 2002; 543(2): 399-411.

10. Pedregosa et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 12. 2011: 2825-2830.