TEKNOFEST

AEROSPACE AND TECHNOLOGY FESTIVAL

COMPETITION OF ARTIFICIAL

INTELLIGENCE IN HEALTH

(Category of Developing Artificial Intelligence Based Analysis Methods in the Field of Bioinformatics)

PROJECT DETAIL REPORT

TEAM NAME: META-BEE-AI

TEAM ID: 447819

Contents

1. Application Introduction

In this section, an algorithm called Metabolitics [18] is introduced. Metabolitics is an algorithm developed by our team based in İstanbul Technical University Bioinformatics Laboratories. This algorithm performs systems-level analysis of metabolic pathways based on metabolite concentrations obtained from mass spectrometric data. Since it became evident that the phenotype of many diseases is directly related to metabolite levels in the body [1-3], a metabolic analysis on a patient's sample seems to be a promising diagnostic method. Several studies have tried to quantify metabolite measurements and perform pathway analysis (e.g, [4-5]). And in most cases, scores are assigned to the pathways and are fit into machine learning models which can then be used to predict whether an individual has a particular disease or not. However, most of these studies assume that pathways are independent and therefore, do not account for the interrelationship between pathways. The method employed in Metabolitics considers that pathways are part of a bigger network and therefore performs analysis in a holistic fashion. The first version of Metabolitics makes use of the Recon2 data model which contains 5324 metabolites and 7785 reactions and was compared to the state of the art algorithms Pathifier [6] and Paradigm [7]. Some limitations of the previous Metabolitics[8] is that it relies on a version of Recon that does not cover all metabolites and reactions. Also, despite the large amount of metabolites present in the recon model, only a few metabolites (less than 150) could be mapped to the recon model due to insufficient name mappings in our mapping function. We believe that including more metabolites in the analysis can improve the insights gotten from the analysis and the classification accuracy. Therefore, the Recon3D model which contains more metabolites (5835) and reactions (10600) is used. Our team aims to use the scikit-learn machine learning library to experiment with different machine learning algorithms in order to develop the best solution for applying machine learning on metabolic analytic results obtained by Metabolitics. These results, once fit in classifiers, can be able to determine if the subject has a particular disease or not. However, the applications of machine learning in the medical field are unlike any other. It is expected for the model to be highly sensitive and specific as well as to be able to generalize on different data distributions. Also, the unavailability of sufficient dataset makes it very hard to evaluate these models. With regards to this, The mean sensitivity and Specificity evaluation metrics and a stratified k-fold cross validation strategy are used to evaluate the final model.

2. Project Current Status Assessment

This project, as always, is aimed at coupling machine learning algorithms with metabolomics analysis algorithms for the end purpose of achieving the possibility of fast medical diagnosis from metabolic data. In the initial draft, it was mentioned that one of the major challenges in pushing through with this implementation is the bias that may result in low count of mapped metabolites. It was planned to increase the number of mapped metabolites by cross checking the Recon models, checking any case sensitive errors and finding databases that could provide synonyms to the given metabolite names and then try to map the synonyms. However, no synonyms database were found that match names in the Recon model. Also, despite trying to combine different Recon files (Recon2 and Recon3D), and solving minimal mismatches, only an additional 9 metabolites names were mapped from our dataset; thus making an increase from 115 mapped metabolites to 124 mapped mapped metabolites. Alternative solutions to increase the number of mapped metabolites is by using approximate matching algorithms such as those mentioned in [22]. However, these algorithms may result in more falsely mapped metabolites. For this reason, we decided to focus efforts on improving the machine learning models, with hopes that the data will be improved in the future. This means that effort is now focused on output of Metaboltics, such as data normalization, feature selection algorithms, using different classifiers, solving of class imbalance and using a proper model evaluation strategy.

3. Algorithms and Artificial Intelligence Model

The Metabolitics pipeline employs both statistical, unsupervised and supervised algorithms at three main levels namely; diff score computation (reaction level diff scores and pathway level diff scores), feature selection for classification task and finally supervised machine learning. Figure 1 below shows a summary of our analysis pipeline.

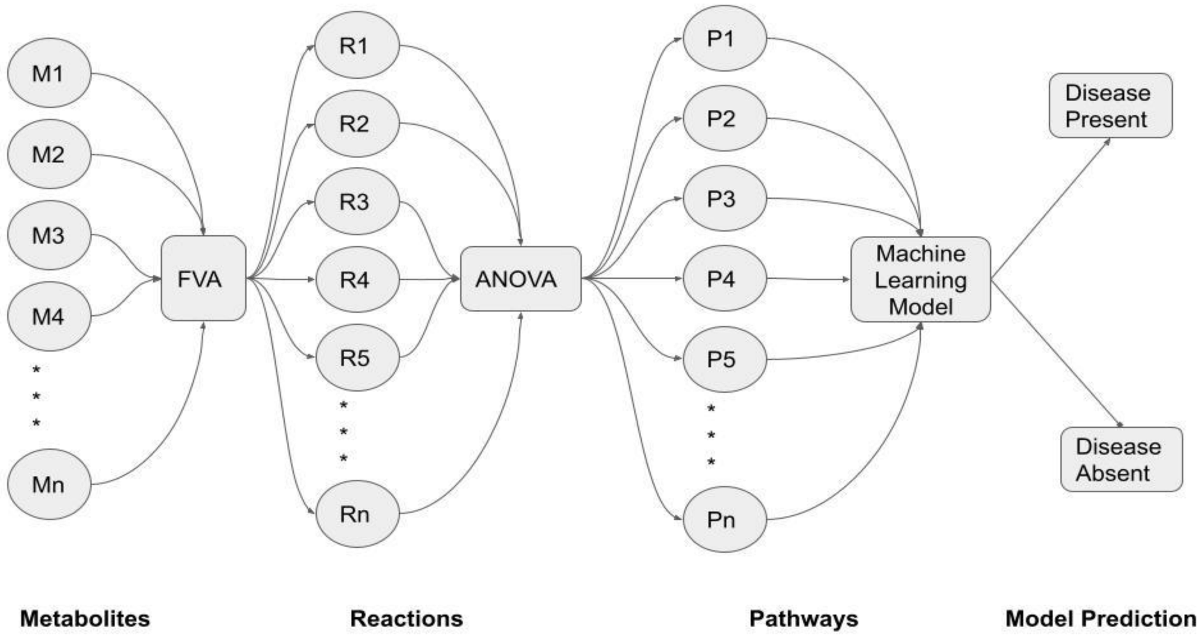

***Figure 1:** Metabolitics Analysis pipeline*

3.1 Diff score computation

Metabolitics performs a flux variability analysis (FVA) [8] on metabolite measurements obtained for a single individual to obtain a personalized analysis. First, the metabolite measurements are scaled to a smaller range of values compared to those obtained by various mass spectrometric techniques using a standard scaler algorithm. Then a concentration fold change is computed from the scaled metabolites as seen in equation 1 below.

$$m_{fc} = log(m^{c}) - log(\mu_{m}^{healthy}) \qquad (1)$$

Where, m^{c} and $\mu_{m}^{healthy}$ are the scaled metabolite concentrations and the average metabolite concentration over healthy samples, respectively. The concentration fold changes m_{fc} are then used to compute the reaction level diff scores through FVA. In our analysis, metabolites are partitioned amongst reactions and reactions amongst pathways according to the Recon [9] data model. Metabolites belonging to a particular reaction are participants in that reaction according to their stoichiometric coefficients and as such, reaction diff scores obtained through FVA carry information that might be correlated to the disease. To be able to make meaning out of these diff scores, reaction level diff scores are converted to pathway level diff scores by averaging the top k significant reactions over a given pathway. The degree of significance of a reaction is determined by using the ANOVA [10] algorithm. Both reaction and pathway scores are used separately as features to fit machine learning models and the results are compared.

3.2 Feature Selection

Due to the high dimensionality of the features (diff scores), there is a need for feature selection in order to boost the performance of the classification models that are used. The dimensionality reduction algorithms [11] tried for feature selection are PCA, LowVarianceThresholding and truncated SVD, SelectFromModel, and ANOVA. Forward Selection (FS) and Backward Elimination (BE) feature selection algorithms were not used because they were too computationally costly.

- **Principal Component Analysis (PCA):** This is a method that is used to represent multidimensional data into a few visualizable dimensions while retaining as much information as possible [19]. To optimize PCA, different values for the number of components are experimented on.
- **LowVarianceThresholding:** This method is used for simply removing features that don't have enough variance across the data samples. Such features do not contribute to any information and may be redundant. Redundant features may cause a model to overfit. Changing the threshold value allowed for the optimization of the algorithm
- **Truncated Singular Value Decomposition (SVD):** SVD is similar to PCA and is used to decompose a matrix in several useful components [20]. SVD is used as a feature selection algorithm to decompose many features (10600 in our case) to a few features that carry almost the same information as the entire set of features. Similar to PCA, different values for the number of components are experimented on to optimze SVD.
- **SelectFromModel:** This is a feature selection method that makes use of another machine learning model in selecting features. The model is trained on the dataset and based on the weights assigned to the different features, a set of features is selected. This algorithm is optimized by using different selector models such as support vector classifiers, random forest, logistic regression, gradient boosting classifier, and so on.
- **Analysis of Variance (ANOVA):** This is a technique used to select statistically significant features amongst 2 or more population groups [21]. In our case, we have the cancer group and the healthy group. This works by analyzing the level of variance between samples from both groups. The top k-features are selected and the k varied to obtain optimal results.

3.3 Machine Learning

The purpose of extending Metabolitics with machine learning models is to be able to predict the disease status of an individual using the diffs scores obtained from their metabolic analysis. As a case study, labeled breast cancer data is mined from the metabolomics workbench database [12]. Therefore, by using metabolite measurements from patients, Metabolitics can predict if a patient has breast cancer or not. The classification algorithms tried are MLPClassifier, SVC, RandomForestClassifier, XGBClassifier, GradientBoostingClassifier, KNeighborsClassifier, LogisticRegression, DecisionTreeClassifier, AdaBoostClassifier and Ensemble Voting. Hyper-parameters are tuned for each of these models and the models are evaluated using the f1-score matrix and a k-fold cross validation strategy. A 10-fold validation was performed on 211 samples. Figure 2 below depicts the machine learning pipeline used in our research.

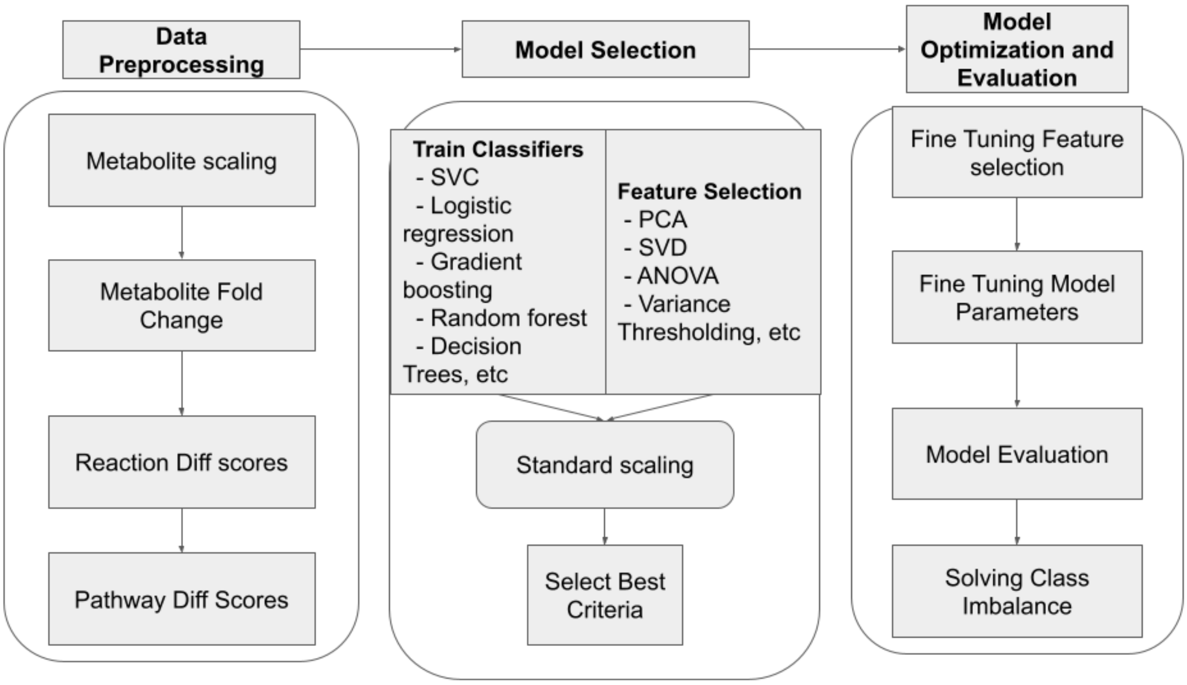

Figure 2. *Steps from Data Processing to Evaluation of Final Model*

3.3.1. Data Preprocessing

The data preprocessing steps are covered by the Metabolitics algorithm. As explained in section 3.1, the metabolites concentrations are scaled user sklearn standard scaler, then fold changes (m_{fc}) are computed by subtracting the mean metabolite concentration from each metabolite in each sample as seen in equation 1. Metabolic analysis is performed on these metabolite fold changes to obtain reaction level changes (reaction diff scores) and pathway changes (pathway diff scores). These diff scores are used for machine learning models. The data is retained in list format format, where each entry in the list represents a data sample and each sample is a python diction of reaction or pathway names mapped to their respective diff scores. The labels on the other hand are in list format where breast cancer group are labeled "c" and control group are labeled "healthy". This data format is compatible with sklearn machine learning pipeline.

3.3.2 Model Selection and Optimization

To select from among the different classifiers and feature selection methods, all classifiers are trained with each feature selection method (mentioned above) applied. Default parameters for both classifiers and feature selection methods are used for this step. From this, the best combination of feature selection algorithms is associated with each classifier as will be shown in section 5. below. The presence or absence of normalization in the pipeline is experimented in each case and the best criteria are taken into consideration. Next, each feature selection algorithm is fine-tuned on the classifier it was associated with and then the corresponding classifier's parameters are fine-tuned using the appropriate feature selection method. Thus, the best results are obtained for each classifier and the best five out nine classifiers are selected for an additional ensemble model building. Ensemble algorithms are known for increasing the accuracy of machine learning models since more than one model is involved in decision making. The best model between the top five models and the ensemble model is selected as the final model.

3.3.3 Model Evaluation

A cross 10-fold stratified cross validation strategy is used all along. However, for evaluation of the final model, the cross validation is customized such that all the evaluation is done on an equal number of samples from each class in each fold. Table 1 below shows how the data split is done.

Table1. *Customized K-fold Cross Validation*

	Fold 1	Fold 2	Fold 3	***	Fold 9	Fold 10	Total Train
Split 1	Test 18 class 1 18 class 2	7 class 1 13 class 2	7 class 1 13 class 2		7 class 1 13 class 2	2 class 1 13 class 2	58 class 1 117 class 2
Split 2	7 class 1 13 class 2	Test 18 class 1 18 class 2	7 class 1 13 class 2		2 class 1 13 class 2	7 class 1 13 class 2	58 class 1 117 class 2
Split 3	7 class 1 13 class 2	7 class 1 13 class 2	Test 18 class 1 18 class 2		7 class 1 13 class 2	7 class 1 13 class 2	58 class 1 117 class 2
* * *							
Split 9	7 class 1 13 class 2	2 class 1 13 class 2	7 class 1 13 class 2		Test 18 class 1 18 class 2	7 class 1 13 class 2	58 class 1 117 class 2
Split 10	2 class 1 13 class 2	7 class 1 13 class 2	7 class 1 13 class 2		7 class 1 13 class 2	Test 18 class 1 18 class 2	58 class 1 117 class 2

For each data split (10), the dataset is split into 10 equal segments (folds). The splitting is done separately for each class to make sure that each fold contains the same number of class 1, and the same number of class 2, except for the test fold which will retain any remaining samples after an 10 division is made, since the number of samples for each class may not be a multiple of 10. For each split, a fold becomes the test set while the remaining 9 folds are used for training. In case the number of cancer and non-cancer samples in a given test set are not exactly equal, one (backup fold) from the other 9 folds is used to balance the test set. The test folds are highlighted in gray on the table, the backup fold in blue and any other entry on a given row becomes part of the training set. In the cancer dataset used, there are 135 samples with cancer and 76 samples with no cancer. Therefore each fold contains 7 cancer samples and 13 non-cancer samples. The test set contains 13 cancer samples and 18 normal samples. So, 5 samples are taken from the backup fold to make 18 cancer samples thus having a test size of 36 samples and a train size of 175 for each validation. The average test score from each split is considered as the actual validation. Also, the standard deviation of test scores is computed to evaluate how the model generalizes on different split data distributions. The matrices of evaluation used are F1-score, and the mean sensitivity and specificity.

$$f1 - score = \frac{2 * precision * recall}{precision + recall}$$

$$precision = \frac{TP}{TP + FP}$$

$$recall(Sensitivity) = \frac{TP}{TP + FN}$$

$$specificity = \frac{TN}{TN + FP}$$

$$Mean_{sensitivity\&specificity} = \frac{2*sensitivity*specificity}{sensitivity + specificity}$$

- **Precision**: the total number of true positives (TP) divided by the total number of true positives plus false positives (FP). It measures how well the model predicts on positive samples.
- **Recall (Sensitivity)**: the total number of true positives divided by the total number of true positives plus false negatives (FN). It is the ability to correctly identify the disease class.
- **Specificity:** the total number of true negatives divided by the total number of true negatives plus false positives. It is the ability to correctly identify the non-disease class.
- **True Positives:** These are samples that belong to the disease group and are predicted as disease.
- **True Negatives:** These are samples do belong to the control (healthy) group and are predicted as healthy
- **False Positives:** These are samples that belong to the control group but are predicted as disease.
- **False Negatives:** These are samples that belong to the disease group but are predicted as healthy.

2.4 Hardware compatibility

The feature selection and machine learning steps can be done on basically any windows or linux based system. On a 64-bit Ubuntu 22.04.6 LTS, the processes finish in a few seconds. However, model optimization steps take a lot of time and require a computer with more power to run faster. Also, the FVA step is an under determined step that requires the computation of a large systems matrix and optimization of a system of equations, the process is relatively very slow and can take hours. The FVA step requires a solver to speed things up and we use IBM CPLEX optimization solver. Despite using CPLEX, the diff score computation process can take up to 2-3 hours on a local PC and uses excessive CPU power. Therefore, using a server with multiple cores is advisable. Our team uses the UHEM Sariyer servers of İstanbul Technical University with SSH to perform a faster analysis.

4. Originality

Mass spectrometers (devices used to separate patients' metabolomics from the taken biofluids or tissues) are becoming cheaper as days pass by. Therefore, if success is achieved by our Metabolitics algorithm at predicting the disease phenotype very accurately, it will result in the development of very fast and cheap diagnostic kits. Whence, the patient's biofluid is measured by mass spectrometers and analyzed by our algorithm which simultaneously will perform a diagnosis. Any algorithm that performs metabolic analysis could also be used, but since they do not account for the interrelationship between pathways, there will be explicability issues in regard to how they work. Metabolitics[18] in its novelty, analyzes metabolites and pathways by considering that pathways are part of an even bigger network. The Metabolitics[18] analysis can reveal biomarkers in metabolic networks. For instance, it was inferred from metabolomic analysis that asparagine synthase activity increases in breast cancer patients as compared to normal patients. Due to these insights that Metabolitics[18] can generate, having a database of the disease profiles based on metabolomic analysis is one of the team's goals. For this, data is mined from the Metabolomics Workbench and for any available disease, metabolic analysis is performed. The database for Metabolitics[18] is another project carried out by a different group of students in the İTÜ Bioinformatics Labs and Databases organization. Collaborating them will make Metabolitics[18] the top successful metabolic analysis projects in Turkey with its uniqueness and provide more accurate results than the state of art algorithms Pathifier [6] and Paradigm [7] in the world. Also, the machine learning steps followed in Figure 2 above, most especially the cross validation strategy are both novel machine learning methodologies developed by our team.

5. Results and Review

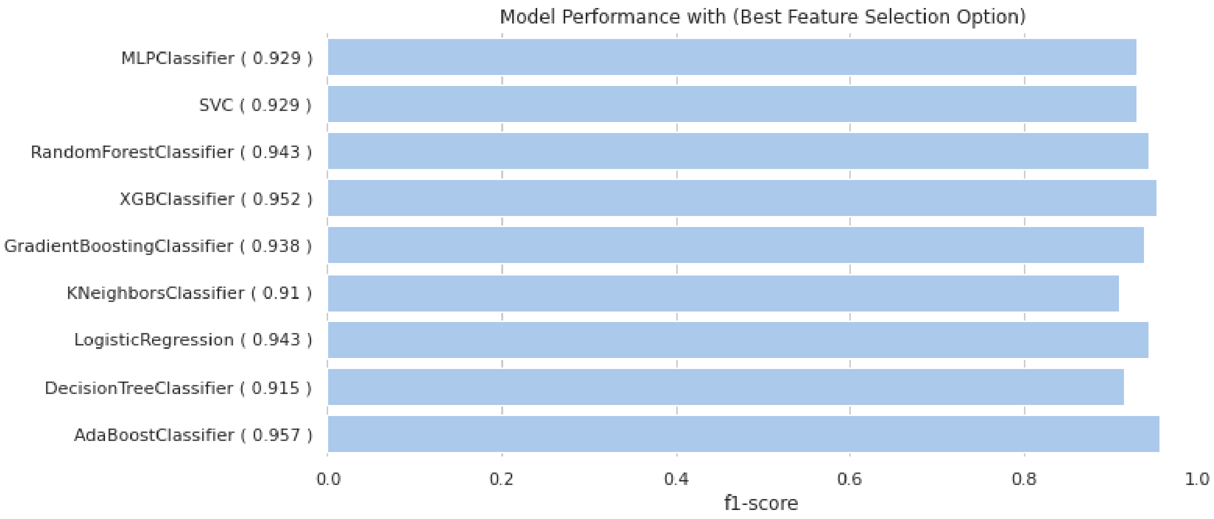

Figure 3. *Comparison of machine learning models based on their maximum scores from each feature selection algorithm*

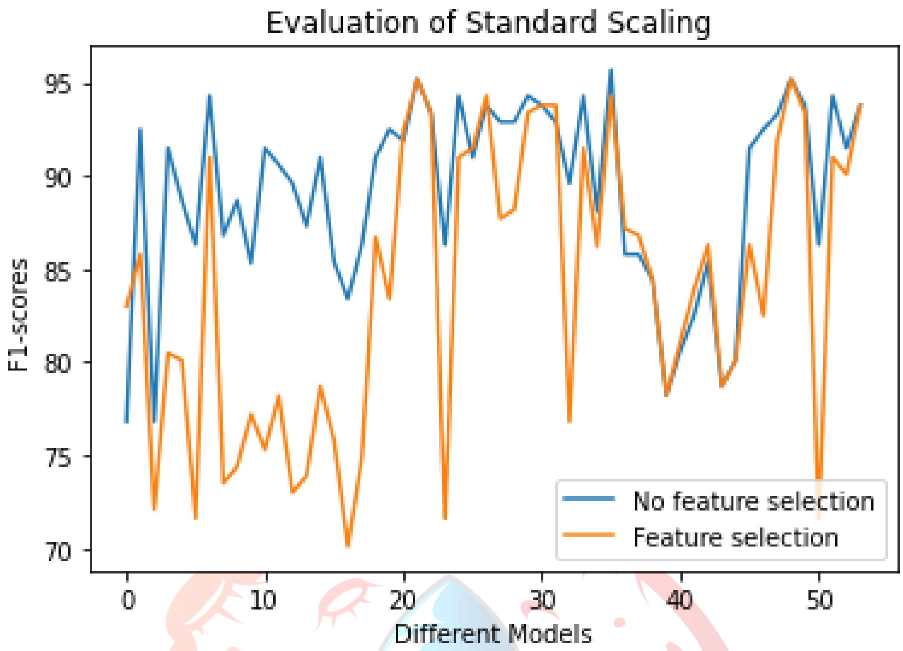

Figure 4. *Comparison of Standard Scaling Performance 9 Classifiers 6 Feature Selection Methods*

Table 2. *Machine Learning Performance with Recon3D (f1-score)*

Models	No Feature Selection (%)	Feature Selection(FS)		Tuned Model Parameters (%)
		FS Method	Scores (%)	
AdaBoost Classifier	93.8	SelectFromModel	96.7 ± 4.3	W.I.P
Decision Tree Classifier	91.5	VarianceThreshold	94.8 ± 5.0	W.I.P
Logistic Regression	94.3	PCA	92.4 ± 6.5	W.I.P
Random Forest Classifier	93.3	SelectFromModel	96.7 ± 4.3	W.I.P
SVC	92.5	SelectFromModel	96.7 ± 4.3	95.7 ± 2.6
Neural Network	91.5	SelectFromModel	94.8 ± 5.0	90.5 ± 4.2
XGB Classifier	95.2	VarianceThreshold	96.7 ± 4.3	W.I.P
K Neighbors Classifier	86.3	TruncatedSVD	93.3 ± 7.1	W.I.P
Gradient Boosting Classifier	93.8	SelectFromModel	96.7 ± 4.3	W.I.P

Table 3. *Machine Learning Performance with Recon2 using Reaction* Diff Scores *(f1-score)*

Models	Original Paper Results (%)		New Results (%)	
	Mean score	Std score	Mean score	Std score
AdaBoost Classifier	88.5	5.6	95.7	-
Decision Tree Classifier	86.5	5.8	91.5	-
Logistic Regression	89.9	4.6	94.3	-
Random Forest	89.0	5.2	94.3	-
SVC	90.4	4.7	92.9	-
Neural Network	86.9	5.3	92.9	-

Table 4. *Evaluation of Ensemble Machine Learning Models (f1-scores)*

Ensemble Models	Results (%)	
	Mean score	Std. score
AdaBoost Classifier, Logistic Regression, XGBClassifier	94.8	4.5
AdaBoost Classifier, Logistic Regression, Random Forest Classifier	95.7	4.5
AdaBoost Classifier, Logistic Regression, Gradient Boosting Classifier	95.3	4.3
AdaBoost Classifier, XGB Classifier, Random Forest Classifier	94.8	4.5
AdaBoost Classifier, XGB Classifier, Gradient Boosting Classifier	95.3	4.2
AdaBoost Classifier, Random Forest Classifier, Gradient Boosting Classifier	95.3	4.2
Logistic Regression, XGB Classifier, Random Forest Classifier	95.8	4.4
Logistic Regression, XGB Classifier, Gradient Boosting Classifier	95.3	4.2
LogisticRegression, Random Forest Classifier, Gradient Boosting Classifier	95.8	4.4
XGB Classifier, Random Forest Classifier, Gradient Boosting Classifier	WIP	WIP

The figures and tables above show our machine learning results so far. Figure 3 above shows the best performance of each of the 9 models after different feature selection algorithms were implemented. This shows that the AdaBoosting classifier has the best model performance. Figure 4 on the other hand, shows the effects of applying normalization (scaling the values to a normal distribution with zero mean and unit variance, in other words z-score standardization) on the dataset. Usually, normalization leads to better model performances, but because Metabolitics performs its own normalization of diffs scores, further normalization leads to worse scores in all models as shown in Figure 4. After optimizing each model separately (only Neural Network and SVC models are optimized with: {'hidden_layer_sizes': (100,), 'learning_rate': 'constant', 'activation': 'tanh', 'solver': 'sgd', 'alpha': 0.05}, {'C': 0.1, 'gamma': 1, 'kernel': 'poly'} hyperparameters respectively shown in Table 2 and Table 3), the five best performing models were AdaBoost Classifier, Logistic Regression, XGB Classifier, Random Forest and Gradient Boosting Classifier. These models were ensemble and a list of combinations of three models was generated and each combination was used as a parameter to finetune the voting classifier. Also, the weight of each classifier, as well as giving priorities to specific classifiers, were fine tuned. The completed fine tuning results (mean f1-scores and standard deviation scores) should be in Table 4. More so, in Table 2, the best feature selection algorithm for each classifier is determined and these feature selection algorithms are fine tuned and this is seen to boost the performance of almost all the models. Table 3 on the other hand shows the performance of our models without optimization compared with the initial results of Metabolitics that were run on the Recon2 data model. Obviously the results have been improved for each model. All models will be optimized and compared again with the initial and state of the art performances.

Since the optimization part needs an excessive amount of computational resources, there are some parts which are not completed yet. These parts are shown as W.I.P (Work in Progress in the tables above). We used UHEM servers for the Python scripts for machine learning as well. They are still running as we write this report.

6. References

[1] F. Ameer, L. Scandiuzzi, S. Hasnain, H. Kalbacher, and N. Zaidi, “De novo lipogenesis in health and disease,” Metabolism, vol. 63, no. 7, pp. 895–902, 2014

[2] J. C. Dodge et al., “Metabolic signatures of amyotrophic lateral sclerosis reveal insights into disease pathogenesis,” Proc. Nat. Acad. Sci. USA, vol. 110, no. 26, pp. 10812–10817, 2013.

[3] A. Raj, E. LoCastro, A. Kuceyeski, D. Tosun, N. Relkin, M. Weiner, and Alzheimer’s Disease Neuroimaging Initiative (ADNI), “Network diffusion model of progression predicts longitudinal patterns of atrophy and metabolism in Alzheimer’s disease,” Cell Rep., vol. 10, no. 3, pp. 359–369, 2015.

[4] J. Wang, D. Duncan, Z. Shi, and B. Zhang, “WEB-based gene set analysis toolkit (WebGestalt): Update 2013,” Nucleic Acids Res., vol. 41, no. W1, pp. W77–W83, 2013.

[5] A. Noronha et al., “The virtual metabolic human database: Integrating human and gut microbiome metabolism with nutrition and disease,” Nucleic Acids Res., vol. 47, pp. D614–D624, 2019.

[6] Y. Drier, M. Sheffer, and E. Domany, “Pathway-based personalized analysis of cancer,” Proc. Nat. Acad. Sci. USA, vol. 110, no. 16, pp. 6388–6393, 2013.

[7] C. J. Vaske et al., “Inference of patient-specific pathway activities from multidimensional cancer genomics data using PARADIGM,” Bioinformatics, vol. 26, no. 12, pp. i237–i245, 2010.

[8] A. C. Muller and A. Bockmayr, “Fast thermodynamically con- € strained flux variability analysis,” Bioinformatics, vol. 29, no. 7, pp. 903–909, 2013.

[9] I. Thiele et al., “A community-driven global reconstruction of human metabolism,” Nature Biotechnol., vol. 31, pp. 419–425 (2013).

[10] X. Cui and G. A. Churchill, “Statistical tests for differential expression in cDNA microarray experiments,” Genome Biol., vol. 4, no. 4, 2003, Art. no. 210.

[11] N. Halko, P. G. Martinsson, and J. A. Tropp, “Finding structure with randomness: Stochastic algorithms for constructing approximate matrix decompositions,” 2009, arXiv:0909.4061.

[12] "The Metabolomics Workbench, https://www.metabolomicsworkbench.org/

[13] Van Rossum, G., & Drake, F. L. (2009). *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace.

[14] Kluyver, T., Ragan-Kelley, B., Fernando Perez, Granger, B., Bussonnier, M., Frederic, J., Willing, C. (2016). Jupyter Notebooks – a publishing format for reproducible computational workflows. In F. Loizides & B. Schmidt (Eds.), *Positioning and Power in Academic Publishing: Players, Agents and Agendas* (pp. 87–90).

[15] Ebrahim, A., Lerman, J.A., Palsson, B.O. *et al.* COBRApy: COnstraints-Based Reconstruction and Analysis for Python. *BMC Syst Biol* 7, 74 (2013). https://doi.org/10.1186/1752-0509-7-74

[16] Pedregosa, F., Varoquaux, Ga"el, Gramfort, A., Michel, V., Thirion, B., Grisel, O., … others. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*(Oct), 2825–2830.

[17] Cplex, I. I. (2009). V12. 1: User’s Manual for CPLEX. *International Business Machines Corporation*, *46*(53), 157.

[18] A. Cakmak and M. H. Celik, "Personalized Metabolic Analysis of Diseases," in *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 18, no. 3, pp. 1014-1025, 1 May-June 2021, doi: 10.1109/TCBB.2020.3008196.

[19] Syms, C. (2018). *Principal Components Analysis ☆. Reference Module in Earth Systems and Environmental Sciences.* doi:10.1016/b978-0-12-409548-9.11152-2

[20] Chengwang, L. (2010). *Singular Value Decomposition in Active Monitoring Data Analysis. Active Geophysical Monitoring, 421–430.* doi:10.1016/s0950-1401(10)04027-9

[21] Smalheiser, N. R. (2017). *ANOVA. Data Literacy, 149–155.* doi:10.1016/b978-0-12-811306-6.00011-7

[22] Qi, X., Ozsoyoglu, Z. M., & Ozsoyoglu, G. (2014). *Matching metabolites and reactions in different metabolic networks. Methods, 69(3), 282–297.* doi:10.1016/j.ymeth.2014.06.007