Evaluating SHAP's Robustness in Precision Medicine: Effect of Filtering and Normalization

Masrur Sobhan
Knight Foundation School of Computing and Information Sciences
Florida International University
Miami, Florida, USA
msobh002@fiu.edu

Ananda Mohan Mondal
Knight Foundation School of Computing and Information Sciences
Florida International University
Miami, Florida, USA
amondal@fiu.edu

2023 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) | 979-8-3503-3748-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/BIBM58861.2023.10385704

***Abstract*—Local interpretation of explainable AI, SHAP (SHapley Additive exPlanations), in disease classification problems offers significant feature scores for each sample, potentially identifying precision medicine targets. Tailoring treatments based on individual genetic and molecular targets can enhance therapeutic outcomes while minimizing side effects. However, the suitability of SHAP's local interpretation at the patient level remains uncertain. It generates different sets of patient-specific genes in various runs, even with consistent overall accuracies. This uncertainty challenges the reliability of SHAP's local interpretations for precision medicine applications. Not only that, different filtering criteria and normalization techniques may influence the contribution scores of patient-specific features. To validate our hypothesis, SHAP was applied to machine learning algorithms from different genres to identify patient-specific feature contributions from the breast cancer subtype classification problem. The program underwent multiple runs to assess the robustness of SHAP.**

Our study demonstrates that shallow machine learning algorithms, like Logistic Regression, consistently provided stable and reliable results across multiple runs. In contrast, complex machine learning models like XGBoost and MLP exhibited inconsistencies across different runs. Moreover, we found that data normalization techniques, particularly z-score and min-max normalization, had a minimal effect on the performance of XGBoost models. Our study also shows that the accuracy scores of complex machine learning models remained relatively constant across different runs but produced different sets of patient-specific features. In conclusion, our findings underscore the importance of selecting appropriate filtering and normalization techniques, given the variability in SHAP results across different runs. Our study indicates that combining SHAP with shallow machine learning algorithms yields more stable and dependable results compared to complex machine learning approaches.

Keywords— explainable machine learning, robust AI, patient-specific biomarkers, precision medicine, SHAP

I. Introduction

Data preprocessing is crucial for achieving effective classification performance when applying machine learning algorithms. It involves various tasks, including data discretization, outlier removal, and data normalization. Data normalization is especially important to ensure that features with larger numeric values do not dominate those with smaller values, reducing potential bias and promoting equal feature importance in the learning process [1]. Data normalization's significance in enhancing predictive models has been demonstrated across multiple machine learning algorithms [2]–[4] and in several applications [5]–[8]. Normalization techniques for large-scale expression data, including RNAseq, serve the purpose of mitigating systematic experimental bias and technical variability while preserving the underlying biological distinctions [9]. Additionally, various data filtering methods have been applied to RNA-seq data. These methods involve filtering genes based on criteria such as having a total read count below a specified threshold [10] and excluding genes that contain at least one zero count in each experimental condition [11]. Typically, noise in the large-scale expression data with exceptionally low counts is eliminated by establishing a minimum threshold. However, the selection of this threshold is a subject of controversy [12]. For simplicity, it is important to establish a difference between filtering and normalization. Normalization entails the rescaling of values, for example, FPKM values. Filtering involves the removal of values that meet specific criteria. Implementing independent filtering of RNA-seq data, sometimes referred to as filtering or cleaning, and rescaling or normalizing the values may enhance the machine learning model's performance. The study empirically examines the influence of normalization methods and filtering in precision medicine. As a case study, we conducted our analysis on breast cancer expression data, highlighting the effects of normalization and filtering.

Breast cancer is a diverse disease with various factors contributing to its development, and the most of cases are sporadic [13]. Hereditary cases make up only around 10-15%, often involving BRCA1 or BRCA2 gene mutations [14]. Patients are categorized based on subtype, which impacts treatment decisions. The most common classification identifies five main breast cancer subtypes: Basal, Luminal A, Luminal B, HER2, and Normal-like breast cancer [13]. These subtypes are distinguished by hormone receptor (ER or PR) and HER2 status [15], helping guide therapeutic approaches for each patient [16]. So, identifying significant biomarkers for each patient is crucial to tailor effective therapeutic strategies for breast cancer treatment.

Several studies have delved into breast cancer subtype classification by employing a range of machine learning and deep learning methods on gene expression data [17]–[20]. Additionally, researchers have turned their attention to

integrating multi-omics and multi-modal data to enhance breast cancer subtype classification [21]–[23]. Graph neural network architectures have also been applied to tackle similar tasks [24], [25]. In addition to cancer classification, both supervised and unsupervised feature selection algorithms have been utilized to identify significant gene biomarkers from a variety of omics data. These selected features or biomarkers are population-based. However, it is well-known that there exists intra-tumor heterogeneity (ITH) resulting from genetic variability among patients [26] with various computational tools available for quantifying ITH scores [27], [28]. As a result, recent research has focused on identifying patient-specific biomarkers using advanced machine learning and deep learning approaches, such as attention mechanisms and graph convolutional neural networks [29], [30]. Additionally, explainable machine learning algorithms have been used to find important patient-specific biomarkers based on feature attribution methods [31]. Surprisingly, none of these studies have addressed the robustness of their outcomes or the robustness of identifying patient-specific biomarkers.

This study takes a significant step by addressing the critical question of outcome stability. By systematically examining the stability of patient-specific biomarkers across multiple runs, we provide insights into the consistency and reliability of these crucial findings, shedding light on the path to more dependable and personalized cancer treatment strategies. In this study, we addressed the issue of robustness in precision medicine while applying an explainable machine learning algorithm, SHAP [32]. We implemented a range of machine learning and deep learning algorithms for breast cancer subtype classification using gene expression data and employed SHAP to compute feature contribution scores for each patient and each gene. Additionally, we explored preprocessing and normalization techniques, providing valuable insights into achieving robust outcomes.

We hypothesize that the identification of biomarker genes using different normalization and filtering techniques utilizing different machine learning and deep learning algorithms may vary while calculating the contribution score using SHAP. We also hypothesize that shallow machine learning algorithms may produce robust results due to their simple architecture, though they may have lower predictability scores. In contrast, complex machine learning algorithms may act differently by producing unreproducible results. This study showed a comparative analysis of identifying patient-specific significant genes.

The remaining part of this paper follows this structure: The "Materials and Methods" section encompasses dataset preparation, and research methods. The "Experimental Results" section provides research outcomes and result analysis. A concise discussion of our findings is presented in the "Discussion" section. Lastly, the "Conclusion" section delves into conclusions drawn from the study and outlines potential future avenues.

II. Materials and Methods

A. Data Collection

To define genes linked to breast cancer subtypes, we gathered copy number variation (CNV), DNA methylation (MET), mRNA expression (EXP) and mutation (MUT) profiles, and clinical information from the GDC portal [33]. The downloaded dataset consists of 1096 cases of CNV, 1097 cases of MET, 1076 cases of EXP, and 969 cases of MUT, as summarized in Table I. To investigate the effect of four omics on each sample, we considered samples with all omics, which resulted in 949 common tumor samples.

The omics datasets contain 105 normal adjacent to tumor (NAT) samples, which were considered healthy samples for this analysis. Thus, a total of 1054 samples were used for further analysis for this study.

Reason for using common genes in multi-omics data: This study is a preliminary step of a big project of multi-omics analysis. This is why we considered common genes among the multi-omics in this study. The goal of the present study is to demonstrate how the filtering and normalization of transcriptome data affect the discovery of patient-specific biomarkers leading to precision medicine.

TABLE I. Sample distribution of four data types of breast cancer subtypes. Due to small number of samples, Normal-like cohort was not used for analysis. DH: Duplicate Handling, NL: Normal-Like.

Data Type	Number of Samples		
	Tumor	Common	Common after DH and removal of NL samples
Gene expression	1076	949	900
Copy Number Variation	1096		
Mutation	969		
DNA Methylation	1097		
Normal Adjacent to Tumor (NAT) / Healthy	105		
Total (Common after DH+ Healthy)	1005 (Basal: 165, Her2: 75, LumA: 471, LumB: 189, Healthy: 105)		

B. Data Preparation

Removal of Normal-like Samples: The number of normal-like samples, 34 in total, was considerably low compared to other subtypes. Thus, we excluded normal-like samples from our analysis.

Duplicate Sample Handling: Within the mRNA gene expression data, we encountered instances of duplicate samples, distinguishable through the TCGA barcode (i.e., TCGA-02-0001-01B). In some cases, the first 15 characters of the barcode were identical, differing only in the vial designation (16^{th} character). In such instances, we retained only the samples with the first vial, thus addressing the issue of duplicate samples. The removal of duplicates resulted in a total of 900 common tumor samples.

The dataset for final analysis consists of 1005 samples with 900 tumor and 105 healthy samples, as shown in Table I. Transcriptome or mRNA expression profiles of 1005 samples was used for further analysis without and with filtering and normalization.

Dataset with All Genes (Without Filtering): Initially, our dataset encompassed a total of 19,962 features or mRNA

expressions with ensemble IDs. The ensemble IDs were then mapped to gene symbols. We refer it as the "Dataset with All Genes."

Dataset with Reduced Genes (With Filtering): Our strategy involved filtering the original feature set based on mRNA expression values in FPKM (fragments per kilobase of transcript per million fragments mapped). Specifically, we retained only those features where FPKM values were equal to or greater than 1 in at least 15% of the samples. This filtering process was executed independently on tumor and healthy expression datasets, which resulted in 13,101 and 13,033 genes for the tumor and healthy cohorts. Next, we combined the genes from both groups, resulting in a total of 13,703 genes, which constituted the "Dataset with Reduced Genes."

Normalization Techniques: In our study, we explored two normalization methods: z-score normalization and min-max normalization. These normalization techniques were applied to "Dataset with All Genes" and "Dataset with Reduced Genes."

We also used the datasets without normalization. Altogether, we have six distinct datasets for the experiment. To facilitate clarity, we assigned the following names to these datasets: 'all genes not normalized', 'all genes z-score', 'all genes min-max', 'reduced genes not normalized', 'reduced genes z-score', and 'reduced genes min-max'.

C. Workflow of the study

The overall workflow of methodology is shown in Fig. 1. The objective of this study is to identify the patient-specific significant genes via explainable AI, SHAP (Shapley Additive exPlanations), while investigating the effect of filtering and normalization techniques applied to the predictor variables (mRNA expressions). Based on the filtering process and normalization techniques, we generated six sets of mRNA expression data of 900 breast cancer and 105 healthy samples as the input to the pipeline.

Fig. 1. **Overall Workflow**. Identifying breast cancer patient-specific genes using gene expression data.

D. Classification Algorithms

We employed three distinct classification algorithms across the six datasets to classify five breast cancer subtypes (Basal, Her2, LumA, LumB, and Healthy). These algorithms encompass a range of methodologies, including logistic regression (LR) [34], multi-layer perceptron (MLP) [35], and extreme gradient boosting (XGBoost) [36]. These algorithms can be categorized into three types: tree-based (XGBoost), probabilistic (LR), and neural network-based (MLP).

Tree-based machine learning models utilize a hierarchical tree structure to make predictions based on input data. They are known for their global interpretability and are widely used in applications like decision-making and data analysis. Common tree-based models include Decision Trees, Random Forests, and Gradient Boosting Trees. XGBoost is a decision tree-based machine learning algorithm that uses a process called boosting to help improve performance. It is an optimized gradient-boosting algorithm through parallel processing, tree pruning, handling missing values, and regularization to avoid bias or overfitting. It belongs to the family of ensemble learning methods, which means it combines the predictions of multiple models to create a strong predictive model.

Logistic Regression is a statistical method used for binary classification, but it can be extended for multiclass classification scenarios. This algorithm is well-suited for scenarios where the relationship between the independent and dependent variables is assumed to be linear. Its simplicity, interpretability, and efficiency make Logistic Regression an attractive choice for multiclass classification tasks, especially when dealing with linearly separable data.

Neural networks, including the Multilayer Perceptron (MLP), are fundamental to deep learning. An MLP is an artificial neural network consisting of multiple layers of interconnected neurons, with an input layer, one or more hidden layers, and an output layer. These networks can approximate complex, non-linear functions and are widely used for various machine learning tasks.

To evaluate the classification performance, a 5-fold cross-validation approach was employed. For stratified sampling, we utilized the StratifiedKFold function from the scikit-learn library. Additionally, we conducted hyperparameter tuning to fine-tune the classifiers for optimal results.

Machine Learning in Precision Medicine: In the field of precision medicine, where treatments are developed and tailored to individual patients, it is essential to have confidence in the consistency and reliability of the machine learning algorithms used to make critical decisions. The robustness of these algorithms, which ensures that they produce consistent and dependable results across multiple runs or iterations, is of utmost importance. A rigorous testing process is employed to evaluate and validate the robustness of these algorithms. It involved executing the same machine learning algorithms ten times, using the same dataset but changes in random seeds. By running the algorithms ten times, we observed how consistent and reliable their results are. This process helped to identify any potential issues with variability or instability in the algorithm's predictions.

Following that, we assessed the contribution of each feature in individual samples. We aimed to discern the factors/features influencing the success or accuracy of the machine learning models. Analyzing feature contributions on a local and global scale is crucial for understanding the models' performance and suitability in precision medicine applications. We utilized an explainable machine learning tool, SHAP, capable of identifying the feature contributions responsible for the model's success.

E. Local Feature Interpretation using SHAP

SHapley Additive exPlanations (SHAP) is a powerful technique rooted in game theory that enables the interpretation of machine learning models' outputs. It's a versatile approach that can be applied to any machine learning or deep learning model. What sets SHAP apart is its ability to calculate a score for each feature for each sample, providing insights into how each feature influences the models' predictions.

The process of calculating SHAP scores involves several steps. First, it considers the machine learning or deep learning algorithms in use and assesses how a specific feature affects

the model's predictions. It does this by comparing the model's predictions with and without the feature for various combinations of features, known as coalition sets. The differences in predictions are computed for each coalition set.

The crucial insight provided by SHAP comes from taking the average of these differences across all possible coalition sets. This average serves as the SHAP score for a specific feature, alternatively referred to as local feature interpretation. Additionally, by averaging the local feature scores, one can derive the global score, known as global feature interpretation.

Local Feature Interpretation: Local feature interpretation involves determining how much each feature or predictor contributed to the model's prediction for a particular sample or patient. It provides a detailed understanding of why a specific prediction was made for that individual patient. Local interpretation is particularly valuable for understanding model predictions on an individual basis, making it useful in personalized medicine. SHAP achieves this by assigning Shapley values to each feature for each sample, quantifying their contributions to the model's output for that particular patient. These values help to elucidate the rationale behind a model's prediction at the local level.

III. Experimental Result

A. Classification Accuracy with Three Classifiers

We employed three diverse classifiers on the dataset, representing three genres of machine learning algorithms, as shown in Fig. 2. To assess their performance, we conducted 5-fold cross-validation to evaluate the models. The testing accuracy of each algorithm was measured across the 5 folds, and subsequently, the average accuracy was calculated as the final performance metric. Fig. 2 summarizes the results obtained from the 5-fold cross-validation of the three classifiers across six distinct datasets. Notably, XGBoost consistently demonstrated the highest accuracy levels, surpassing 91% across all six datasets.

Effect of Filtering (Not normalized case): Here, we compare the results derived from "All genes not normalized" and. "Reduced genes not normalized" datasets. The performance of classifiers LR and XGBoost remain almost the same in the case of all genes and reduced genes datasets. For example, LR produces the same accuracy of 82% in both cases. This means that there is no effect of filtering for classifiers LR and XGBoost. On the other hand, MLP performs worse with filtered data (76%) than without filtered data (83%).

Effect of Filtering (z-score normalization case): Comparing accuracies derived from "All genes z-score" and "Reduced genes z-score" datasets, we observed that LR (88%), MLP (86%), and XGBoost (91%) perform about the same with and without filtering, meaning no effect on filtering.

Effect of Filtering (min-max normalization case): Comparing accuracies derived from "All genes min-max" and "Reduced genes min-max" datasets, we observed that LR, MLP, and XGBoost perform about the same with and without filtering, meaning no effect on filtering.

Effect of Normalization (All genes case): Tree-based algorithm, XGBoost (~92%) performs about the same in "all genes case" without and with normalization (z-score and min-max), meaning that normalization does not affect the performance of XGBoost classifiers using data without filtering. But LR and MLP perform better with normalized data. For example, accuracies of MLP are 84%, 86%, and 88% using "not normalized", z-score, and min-max normalized datasets, respectively.

Effect of Normalization (Reduced genes case): The similar trends are observed as in the "All genes case." The tree-based algorithm, XGBoost (~91%), performs about the same in the "Reduced genes case" without and with normalization (z-score and min-max), meaning that normalization does not affect the performance of tree-based classifiers. On the other hand, LR and MLP perform better with normalized data.

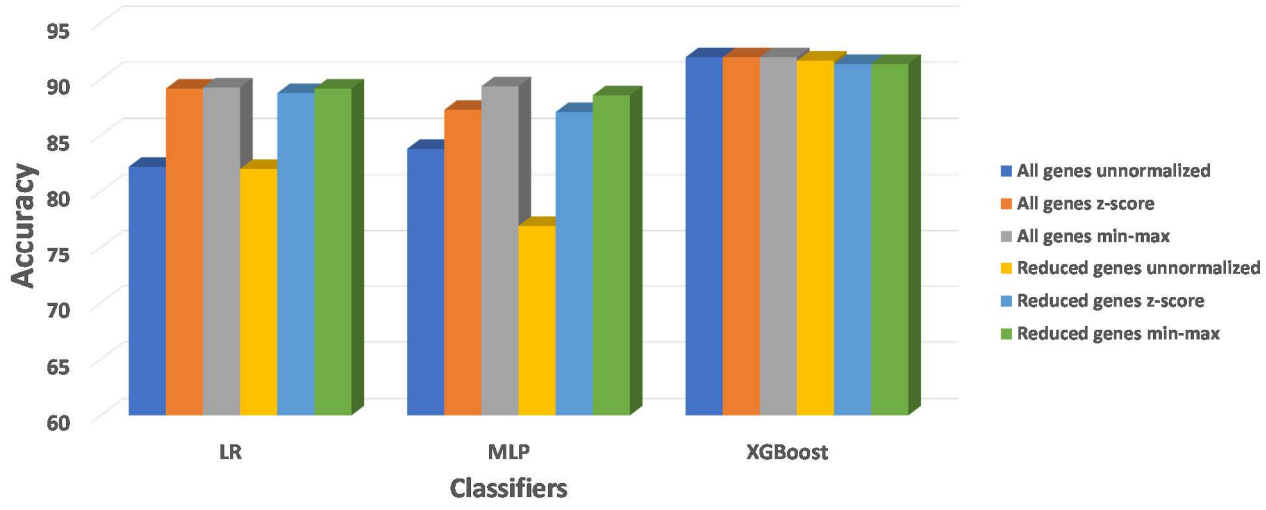

Fig. 2. **Effect of filtering and normalization on classification performance**. Filtering using the threshold on FPKM values ≥ 1 in at least 15% of the samples was used to generate the reduced gene set. Two types of normalization: z-score and min-max. Three classifiers from different genres (tree-based: XGBoost, probabilistic: LR, and neural network-based: MLP) were used to classify five subtypes of breast cancers. LR: Logistic Regression, MLP: Multi-Layer Perceptron, XGBoost: Extreme Gradient Boosting.

The performance of tree-based classifiers is not affected by normalization because these algorithms are independent of distance or similarity between two subjects. On the other hand, the classifiers LR and MLP depend on the distance or similarity between two subjects in different aspects. The normalization process brings all the features or predictor variables under the same scale, which overcomes the bias or dominance of predictor variables with a large range of values. Thus, LR and MLP perform better with normalized data.

Effect of Normalization Techniques (z-score vs. min-max): It is clear from Fig. 2 that min-max normalized data performs as per with z-score data with LR and XGBoost and significantly better with MLP. Thus, we used a dataset with min-max normalization for the subsequent analysis.

B. *Hyperparameter Tuning*

At this point, hyperparameters of classifiers from different genres are tuned up. From Fig. 2, it is evident that the min-max normalization with and without filtering performs at the highest level of accuracy for all three classifiers. Since filtering provides fewer features, we utilized the 'reduced genes min-max' dataset for hyperparameter tuning.

Hyperparameters to Tune: The hyperparameters of logistic regression (LR) from the scikit-learn package were optimized, focusing on random seed and multi-class settings. For the MLP model from TensorFlow's Keras, hyperparameter tuning encompassed random seed, hidden layer configurations, activation functions, loss functions, batch size, and the number of training epochs. Meanwhile, for the XGBoost algorithm, hyperparameters such as learning rate, the number of estimators, maximum tree depth, minimum child weight, gamma, regularization lambda, subsampling rate, column subsampling by tree, scaling of positive weights, objective function, and the number of classes were optimized.

TABLE II. EFFECT OF RANDOM SEEDS ON MODEL PERFORMANCE (ACCURACY). THE EXPERIMENT WAS PERFORMED ON THE 'REDUCED GENES MIN-MAX' DATASET. LR: LOGISTIC REGRESSION, MLP: MULTI-LAYER PERCEPTRON, XGBOOST: EXTREME GRADIENT BOOSTING.

Random Seed	LR	MLP	XGBoost
0	89.15	88.95	92.33
10	89.15	88.45	92.13
20	89.15	84.37	92.23
30	89.15	89.25	92.03
40	89.15	87.96	91.64
50	89.15	89.05	92.13
60	89.15	89.25	92.13
70	89.15	89.05	91.74
80	89.15	87.76	92.43
90	89.15	88.95	92.83

Effect of Random Seeds on Model Performance: Table II shows the effect of random seeds on the machine learning model's performance. We used the model hyperparameter "random state" and changed its values for LR and XGBoost to assess the impact of randomness on the model's performance. In the case of MLP, we initialized the random seed value and systematically changed the value in different runs to evaluate the models' performance.

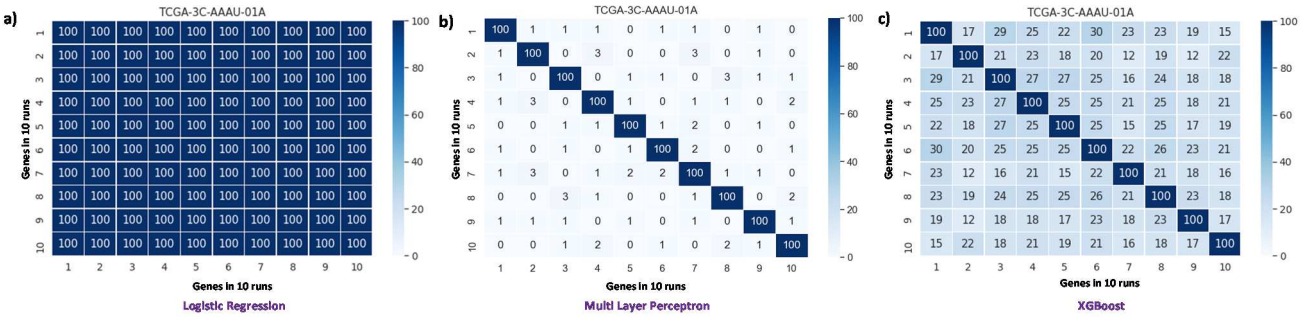

Fig. 3. **Shared genes across 10 different runs for a particular patient (TCGA barcode: TCGA-3C-AAAU-01A).** A reduced gene set with min-max normalization was used for the experiment. Patient-specific top 100 genes from each run were selected using the SHAP score. a) Logistic Regression outcome: the same set of 100 significant genes were identified at each run. b) Multi-Layer Perceptron outcome: almost no common genes across 10 different runs. c) XGBoost outcome: around 20% of the significant genes were common in 10 runs.

Upon reviewing Table II, it becomes evident that there are no differences in accuracies across the different runs in LR. In contrast, there is a slight difference in the accuracies of MLP and XGBoost. The results shown in the table indicate that LR does not have any impact on seed values, hence generating the same accuracies across all runs. In contrast, seed values affected the model's accuracies for MLP and XGBoost.

C. *Local Interpretation using SHAP*

We employed SHAP to identify the most significant genes across ten different runs of the LR, MLP, and XGBoost models, considering each gene and sample individually. This level of local interpretability allowed us to pinpoint patient-specific biomarkers, potentially valuable for personalized medicine or therapy. We adopted a procedure involving multiple iterations to obtain scores for each gene-sample combination. Specifically, we trained the XGBoost model using 80% of the data and evaluated its performance on the remaining 20% of the data. We repeated this process five times, ensuring that a distinct 20% subset was used for testing each time. Consequently, we utilized 100% of the data for testing in five runs (times). However, a small number of false predictions were detected. Subsequently, we removed these falsely predicted samples, retaining only those with accurate predictions for every run. Each of these retained samples was then assigned scores for all genes, calculated using SHAP values. We arranged all the genes in descending order to analyze the data further based on their respective scores.

We employed various model-specific SHAP algorithms to obtain SHAP scores. Specifically, we utilized a linear explainer for logistic regression (LR), a deep explainer for multi-layer perceptron (MLP), and a tree explainer for XGBoost.

Consistency of Genes in Multiple Runs: The consistency of genes appearing in multiple runs is a crucial aspect of our analysis. It signifies the stability and reliability of specific genetic markers associated with an individual across different iterations or experiments. This consistency provides valuable insights into the robustness of these genes as potential biomarkers or indicators for personalized medicine and treatment strategies. By identifying genes that consistently emerge across multiple runs, we can enhance our understanding of the patient-specific genetic factors that play a pivotal role in disease prognosis, diagnosis, and therapeutic interventions.

Our methodology was initiated by selecting the patient-specific top 100 genes across ten independent runs based on their SHAP scores, and this process was applied to LR, MLP, and XGBoost. Fig. 3 summarizes the results of this experiment. It is clear that the patient-specific top 100 genes for LR remained consistent across all runs. In contrast, for MLP and XGBoost, the top 100 genes exhibited variability, which presents a notable challenge in precision medicine. MLP produces sets of top 100 genes with minimal overlap (ranging from 0 to 10), indicating that SHAP identified almost distinct sets of significant genes in different runs. XGBoost exhibited a similar trend, with more overlapped genes (ranging from 12 to 30) in different runs. This observation suggests that LR might be a more reliable choice when seeking consistent results for identifying important genes.

Next, we aimed to assess the variability in outcomes across the different datasets employed in our study. Fig.2 shows that overall accuracy varies, except for XGBoost when different datasets are utilized. Furthermore, the study reveals that the normalization status (normalized or unnormalized) of the dataset does not impact the accuracies, as consistent accuracies were observed for both the normalized and unnormalized datasets, a trend observed in both "all genes" and "reduced gene" datasets.

Due to the consistent performance (accuracy) of XGBoost, we structured our next experiment into three distinct stages using only XGBoost outcomes: (i) Identifying common genes between the normalized and unnormalized versions, (ii) Identifying common genes comparing z-score normalization and min-max normalization, and (iii) Identifying common genes between the 'all genes dataset' and the 'reduced genes dataset.' The results of the three scenarios are shown in Fig. 4 and Fig. 5.

In Stage (i), we identified patient-specific shared genes between normalized and unnormalized data for both the 'all genes' and 'reduced genes' datasets shown in Fig. 4(a) and Fig. 4(b), respectively. It becomes evident that only a few genes overlapped across different runs in this analysis. For instance, when examining the patient TCGA-3C-AAAU-01A in run 1, we found that 17 genes were shared between the two datasets (unnormalized and z-score normalized when all genes are considered) shown in Fig. 4(a). A similar result was noticed when 'reduced genes' were considered, as shown in Fig. 4(b). In Stage (ii), we observed that all the top 100 genes were overlapped between the two datasets (z-score normalized and min-max normalized), as depicted in Fig. 4(c) and 4(d). *This finding suggests that the choice of different normalization techniques did not impact the identification of the top genes*.

The result of Stage (iii) is shown in Fig. 5. It shows similar scenarios to Stage (i) that there are few overlapped genes among the 'all genes dataset' and 'reduced genes dataset' with normalization. Interestingly, the number of common genes for all the patients at every run is the same for the two scenarios shown in Fig. 5(a) and Fig. 5(b). The scenarios in Fig. 5 are generated using two different normalization techniques, but we see the same number of common genes for every patient. This observation again shows that there is no impact of z-score and min-max normalization on identifying significant genes.

Based on the three scenarios mentioned above, the choice of normalization methods does not impact the selection of

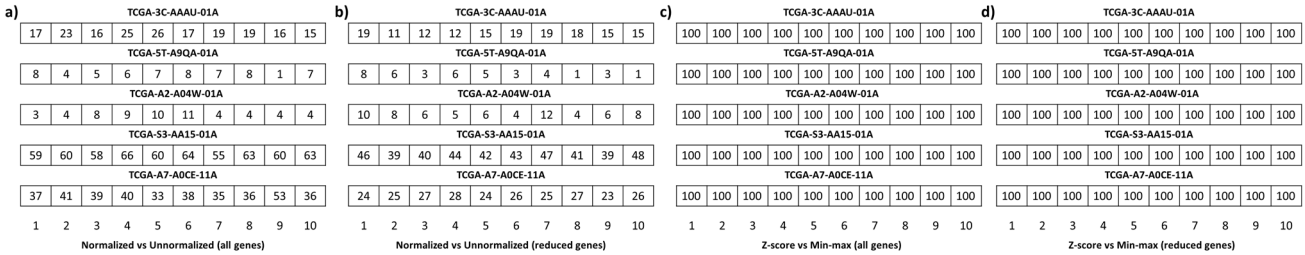

Fig. 4. **Effect of normalization on identifying patient-specific biomarkers using XGBoost**. Shared genes across 10 separate runs (seeds) within five distinct samples. Each cell represents the number of common genes in two runs with the same seed but different normalization. In each run, patient-specific top 100 genes based on SHAP scores were isolated. Two sets of patient-specific genes derived from two runs are compared to find the shared genes. Shared genes between (a) z-score normalized data and unnormalized data with all genes, (b) z-score normalized data and unnormalized data with reduced genes, (c) z-score normalization and min-max normalization with all genes, and (d) z-score normalization and min-max normalization with reduced genes, in ten runs.

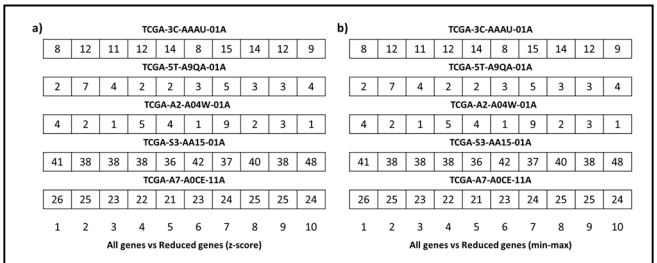

Fig. 5. **Effect of filtering on identifying patient-specific biomarkers.** Shared genes across 10 separate runs (seeds) within five distinct samples. Each cell represents the number of common genes in two runs (all genes vs. reduced genes) with the same seed. In each run, patient-specific top 100 genes based on SHAP scores were isolated. Two sets of patient-specific genes derived from two runs are compared to find the shared genes. Shared patient-specific genes using (a) z-score normalization and (b) min-max normalization.

significant genes for XGBoost. However, differences in selecting significant genes arise when comparing unnormalized and normalized data, even though the accuracies are the same. Also, we found discrepancies in identifying significant genes when the two different datasets (all genes vs. reduced genes) were used.

IV. Discussion

Most of the prior machine learning and deep learning models were predominantly used as "black box" tools without transparent explanations. However, recent advancements have introduced explainable machine learning algorithms such as SHAP, LIME, ANCHOR, and DeepLIFT to explain the model prediction. These approaches offer the capability to provide local and global-level explanations for model predictions. This interpretability is particularly valuable for applications in precision medicine. One critical aspect of using these explainable machine learning models is the robustness of their outputs. To address this question, our study delves into the consistency and reliability of results generated by machine learning and deep learning models.

Our findings indicate that shallow machine learning algorithms like Logistic Regression (LR) consistently produce the same accuracies across multiple runs, resulting in consistent feature importance scores for all features in each run. In contrast, when it comes to more complex models like Multi-Layer Perceptron (MLP) and XGBoost, while the overall accuracies remain similar across runs, there is significant variation in the selection of top genes. This variability poses a challenge for precision medicine, as different runs can yield distinct sets of significant genes, leading to varying outputs.

Variations observed in MLP outcomes can be attributed to several factors. MLPs are initialized with random weights, meaning that the initial conditions for the training process differ in each run. This variance in initialization can lead to diverse convergence paths, ultimately resulting in varying outcomes. In this study, MLP implementation utilized the Stochastic Gradient Descent (SGD) optimization technique, chosen after hyperparameter tuning. This method introduces randomness by employing a subset of the training data in each iteration, adding an additional layer of variability to the process. Furthermore, the optimization process in deep neural networks can converge to distinct local minima, even with consistent random initialization and data. This behavior is due to the highly non-convex nature of the optimization landscape in deep networks, where different runs may find dissimilar local minima.

Similarly, the variations in XGBoost can be attributed to the inherent randomness in the subsampling of both data and features. This variability can result in differences in the model's learned parameters and predictions. Factors such as the model's initializations, including the starting point of the decision trees and feature selection, can diverge between runs, thereby impacting the model's convergence and the structure of the learned trees. A specific random seed was set in our study to mitigate this randomness. While this approach did produce consistent and robust outcomes, it's important to note that the results obtained with a fixed random seed may not necessarily represent the best set of genes for a particular patient. The choice of random seed can influence the results, and as such, variations may arise when different random seeds are employed.

On the other hand, logistic regression (LR) produces the same result in every run because it is a simple and deterministic machine learning algorithm. Logistic regression does not involve randomness in its learning process. The model is built through a deterministic optimization procedure, typically using techniques like gradient descent. For a particular set of hyperparameters and a given dataset, the model's coefficients (weights) will converge to the same values each time.

V. Conclusion

Many previous studies have predominantly concentrated on identifying important genes at the cohort or population level, often overlooking the personalized nature of disease treatment based on individual patient's unique genetic characteristics. This study focused on identifying patient-specific key genes for precision medicine, employing the explainable machine learning algorithm SHAP, and utilizing it for local interpretation. Furthermore, our study underscores the significance of ensuring the robustness and consistency of SHAP explainability in precision medicine applications. It also provides insights into the reasons behind variations in significant gene selection by MLP and XGBoost models across different runs. Additionally, our study highlights the suitability of shallow machine learning algorithms, such as logistic regression, for precision medicine due to their consistent and robust output across different runs. It suggests that logistic regression (LR) may be a superior choice for identifying key genes in terms of robustness.

The results presented in this study are computational outcomes derived from machine learning and deep learning methods. To establish their reliability, validating these findings through wet lab experiments is essential. Successful validation in the wet lab would underscore the potential of our pipeline in identifying crucial genes applicable to various disease types.

In the future, we intend to explore different explainable machine learning methods that can provide local feature interpretation and conduct a comparative analysis of the results. While our current study concentrated on transcriptomic profile data, our upcoming research will encompass various omics data types, including copy number variation (CNV), DNA methylation, and single nucleotide polymorphism (SNP) data to comprehensively assess their roles in disease progression for individual patients. This expansion can potentially unveil valuable biological insights within precision medicine.

Acknowledgment

The work has been supported by the NSF CAREER Award (#1901628) to AMM.

References

[1] D. Singh and B. Singh, “Investigating the impact of data normalization on classification performance,” *Appl. Soft Comput.*, vol. 97, p. 105524, Dec. 2020.

[2] S. Aksoy and R. M. Haralick, “Feature normalization and likelihood-based similarity measures for image retrieval,” Pattern Recognit. Lett., vol. 22, no. 5, pp. 563–582, Apr. 2001.

[3] J. Sola and J. Sevilla, “Importance of input data normalization for the application of neural networks to complex industrial problems,” IEEE Trans. Nucl. Sci., vol. 44, no. 3 PART 3, pp. 1464–1468, 1997.

[4] A. Gupta, F. Nelwamondo, S. Mohamed, C. M. Ennett, and M. Frize, "Statistical Normalization and Back Propagation for Classification."

[5] J. Pan, Y. Zhuang, and S. Fong, "The impact of data normalization on stock market prediction: Using SVM and technical indicators," Commun. Comput. Inf. Sci., vol. 652, pp. 72–88, 2016.

[6] W. Wu, E. P. Xing, C. Myers, I. S. Mian, and M. J. Bissell, "Evaluation of normalization methods for cDNA microarray data by k-NN classification," BMC Bioinformatics, vol. 6, no. 1, pp. 1–21, Jul. 2005.

[7] W. Li and Z. Liu, "A method of SVM with Normalization in Intrusion Detection," Procedia Environ. Sci., vol. 11, no. PART A, pp. 256–262, Jan. 2011.

[8] J. Yu and K. Spiliopoulos, "Normalization Effects on Deep Neural Networks," Found. Data Sci., vol. 5, no. 3, pp. 389–465, 2023.

[9] X. Liu et al., "Normalization Methods for the Analysis of Unbalanced Transcriptome Data: A Review," Front. Bioeng. Biotechnol., vol. 7, p. 501340, Nov. 2019.

[10] M. Sultan et al., "A global view of gene activity and alternative splicing by deep sequencing of the human transcriptome," Science, vol. 321, no. 5891, pp. 956–960, Aug. 2008.

[11] D. Bottomly et al., "Evaluating Gene Expression in C57BL/6J and DBA/2J Mouse Striatum Using RNA-Seq and Microarrays," PLoS One, vol. 6, no. 3, p. e17820, 2011.

[12] I. V. Deyneko, O. N. Mustafaev, A. Tyurin, K. V. Zhukova, A. Varzari, and I. V. Goldenkova-Pavlova, "Modeling and cleaning RNA-seq data significantly improve detection of differentially expressed genes," BMC Bioinformatics, vol. 23, no. 1, pp. 1–14, 2022.

[13] O. Yersal and S. Barutca, "Biological subtypes of breast cancer: Prognostic and therapeutic implications," World J. Clin. Oncol., vol. 5, no. 3, pp. 412–424, Aug. 2014.

[14] M. Abu-Helalah et al., "BRCA1 and BRCA2 genes mutations among high risk breast cancer patients in Jordan," Sci. Rep., vol. 10, no. 1, Dec. 2020.

[15] X. Dai et al., "Breast cancer intrinsic subtype classification, clinical use and future trends," Am. J. Cancer Res., vol. 5, no. 10, p. 2929, 2015, Accessed: Oct. 22, 2023. [Online]. Available: /pmc/articles/PMC4656721/.

[16] K. H. ; Lau, A. M. ; Tan, Y. Shi, K. H. Lau, A. M. Tan, and Y. Shi, "New and Emerging Targeted Therapies for Advanced Breast Cancer," Int. J. Mol. Sci. 2022, Vol. 23, Page 2288, vol. 23, no. 4, p. 2288, Feb. 2022.

[17] J. Shen et al., "Deep learning approach for cancer subtype classification using high-dimensional gene expression data," BMC Bioinformatics, vol. 23, no. 1, pp. 1–17, Dec. 2022.

[18] S. Rajpal, V. Kumar, M. Agarwal, and N. Kumar, "Deep Learning Based Model for Breast Cancer Subtype Classification," Nov. 2021, Accessed: Oct. 22, 2023. [Online]. Available: https://arxiv.org/abs/2111.03923v2.

[19] R. Mendonca-Neto et al., "Classification of breast cancer subtypes: A study based on representative genes," J. Brazilian Comput. Soc., vol. 28, no. 1, pp. 59–68, Dec. 2022.

[20] T. Shibahara et al., "Deep learning generates custom-made logistic regression models for explaining how breast cancer subtypes are classified," PLoS One, vol. 18, no. 5, p. e0286072, May 2023.

[21] J. M. Choi and H. Chae, "moBRCA-net: a breast cancer subtype classification framework based on multi-omics attention neural networks," BMC Bioinformatics, vol. 24, no. 1, pp. 1–15, Dec. 2023.

[22] Y. Ektefaie et al., "Integrative multiomics-histopathology analysis for breast cancer classification," npj Breast Cancer 2021 71, vol. 7, no. 1, pp. 1–6, Nov. 2021.

[23] S. Rakshit, I. Saha, S. S. Chakraborty, and D. Plewczyski, "Deep Learning for Integrated Analysis of Breast Cancer Subtype Specific Multi-omics Data," IEEE Reg. 10 Annu. Int. Conf. Proceedings/TENCON, vol. 2018-October, pp. 1917–1922, Jul. 2019.

[24] R. B. Tanvir, M. M. Islam, M. Sobhan, D. Luo, and A. M. Mondal, "MOGAT: An Improved Multi-Omics Integration Framework Using Graph Attention Networks," bioRxiv, p. 2023.04.01.535195, Apr. 2023.

[25] Z. N. Kesimoglu and S. Bozdag, "SUPREME: multiomics data integration using graph convolutional networks," NAR Genomics Bioinforma., vol. 5, no. 2, Mar. 2023.

[26] F. Lüönd, S. Tiede, and G. Christofori, "Breast cancer as an example of tumour heterogeneity and tumour cell plasticity during malignant progression," Br. J. Cancer 2021 1252, vol. 125, no. 2, pp. 164–175, Apr. 2021.

[27] R. B. Tanvir, A. Al Mamun, M. Sobhan, and A. M. Mondal, "Quantifying Intratumor Heterogeneity by Key Genes Selected using Concrete Autoencoder," bioRxiv, p. 2021.09.06.459161, Jun. 2023.

[28] D. Song and X. Wang, "DEPTH2: an mRNA-based algorithm to evaluate intratumor heterogeneity without reference to normal controls," J. Transl. Med., vol. 20, no. 1, pp. 1–17, Dec. 2022.

[29] H. Guo, X. Lv, Y. Li, and M. Li, "Attention-based GCN integrates multi-omics data for breast cancer subtype classification and patient-specific gene marker identification," Brief. Funct. Genomics, Apr. 2023.

[30] H. Chereda et al., "Explaining decisions of graph convolutional neural networks: patient-specific molecular subnetworks responsible for metastasis prediction in breast cancer," Genome Med., vol. 13, no. 1, pp. 1–16, Dec. 2021.

[31] M. Sobhan and A. M. Mondal, "Explainable Machine Learning to Identify Patient-specific Biomarkers for Lung Cancer," Proc. - 2022 IEEE Int. Conf. Bioinforma. Biomed. BIBM 2022, pp. 3152–3159, 2022.

[32] S. M. Lundberg and S. I. Lee, "A Unified Approach to Interpreting Model Predictions," Adv. Neural Inf. Process. Syst., vol. 2017-December, pp. 4766–4775, May 2017.

[33] "GDC." https://portal.gdc.cancer.gov/ (accessed Oct. 22, 2023).

[34] C.-Y. J. PENG, K. L. LEE, and G. M. INGERSOLL, "Sample Data for Gender and Recommendation for Remedial Reading Instruction."

[35] S. Haykin, Neural networks: a comprehensive foundation. Prentice Hall PTR, 1994.

[36] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., vol. 13-17-August-2016, pp. 785–794, Mar. 2016.