J. Eng. Technol. Sci., Vol. 53, No. 1, 2021, 210109

Gene Family Abundance Visualization based on Feature Selection Combined Deep Learning to Improve Disease Diagnosis

Hai Thanh Nguyen, Tai Tan Phan, Tinh Cong Dao, Thao Minh N. Phan, Phuc Vinh D. Ta, Cham Ngoc T. Nguyen, Ngoc Huynh Pham & Hiep Xuan Huynh*

College of Information Communication and Technology, Can Tho University
Campus II, 3/2 Street, Ninh Kieu District, Can Tho city, 900000, Viet Nam
*E-mail: hxhiep@ctu.edu.vn

Highlights:

- Introduction of a compact way to visualize datasets with a very large number of features using feature selection algorithms and a visualization approach.
- The method provides a solution for visualizing gene data in 2D images with selected features representing important features that greatly influence the classification of patients.
- The proposed *geneFS2Img* method improves not only prediction accuracy but also inference time.
- The performance with the selected features in this study approximated that of state-of-the-art methods using the species abundance feature.

Abstract. Advancements in machine learning in general and in deep learning in particular have achieved great success in numerous fields. For personalized medicine approaches, frameworks derived from learning algorithms play an important role in supporting scientists to investigate and explore novel data sources such as metagenomic data to develop and examine methodologies to improve human healthcare. Some challenges when processing this data type include its very high dimensionality and the complexity of diseases. Metagenomic data that include gene families often have millions of features. This leads to a further increase of complexity in processing and requires a huge amount of time for computation. In this study, we propose a method combining feature selection using perceptron weight-based filters and synthetic image generation to leverage deep-learning advancements in order to predict various diseases based on gene family abundance data. An experiment was conducted using gene family datasets of five diseases, i.e. liver cirrhosis, obesity, inflammatory bowel diseases, type 2 diabetes, and colorectal cancer. The proposed method provides not only visualization for gene family abundance data but also achieved a promising performance level.

Keywords: *deep learning; disease prediction; feature selection; gene family abundance; metagenomic; personalized medicine.*

Received March 22nd, 2020, Revised September 1st, 2020, Accepted for publication November 2nd, 2020.
 ISSN: 2337-5779,
DOI: 10.5614/j.eng.technol.sci.2021.53.1.9

1 Introduction

Currently, one of the trends in disease treatment is 'precision medicine', or 'personalized medicine' [1-3], which is a new term for healthcare combining genetic data with big data and public health to offer a powerful new approach for the right treatment method for the right patient at the right time with the right dose. Advancements in DNA sequencing technology have provided great support for the development of personalized medicine approaches. In such approaches, a patient's DNA is analyzed to find individual characteristics that are probable causes of the patient's disease on the basis of which appropriate individual treatment can be proposed. Thanks to the advancement of genetics technology, precision medicine is becoming more accurate. This method will replace traditional methods to improve the effectiveness of diagnosis and treatment.

The gene expression of each individual combined with general treatment methods have high unsuccessful treatment risk. By applying a personalized treatment, the patient's healing performance can be improved. Moreover, identifying the cause of the disease in each patient can reveal the best cure when traditional methods may not be suitable for their individual constitution. Researchers have discovered hundreds of genes that harbor variations contributing to human illness and identified genetic variability in patient responses to different treatments, and from there began to target genes as molecular causes of diseases. In addition, scientists are developing and using diagnostic tests based on genetics or other molecular mechanisms to better predict patient responses to targeted therapies [4].

In recent years, metagenomics (environmental genomics, eco-genomics, or community genomics) [5] has been increasingly used as part of a personalized medicine methodology group that attempts to the enhance effectiveness of human health care. From the results surveyed through many studies, the potential of the analysis of metagenomic data for the diagnosis of human diseases has been proven. Metagenomics can help to solve problems, including finding the cause of the disease, the essential bacterial species that change their density affecting human health or the disease status of a patient. In addition, given the enormous potential benefits of these data for improving human health care, many scientists have conducted experiments and proposed methods and tools based on the application of information technology to support data analysis for personalized medicine effectively.

One of the remarkable achievements is the progress made in the detection of the novel SARS-CoV-2 (COVID19) virus in a short time [6]. Thanks to new generation sequencing technology (NGS) [7], the entire profile of the genome of the SARS-CoV-2 virus was quickly revealed. The genome of the new virus can be compared with a gene bank to find genetic characteristics that differentiate

new species from known species in the world. It is estimated that the genome is more than 85% identical to the SARS virus genome in the case of SARS-CoV-2[8]. It is worth noting that genetic analysis has many advantages, such as making diagnostication and preparation of drugs and vaccines faster. Viewed from a different perspective, genetic sequencing techniques have changed the accessibility of medicine, especially in the case of dealing with diseases such as SARS-CoV-2 because the virus strains show a tendency to mutate or spread by other mechanisms.

Data visualization is an essential part of data analysis and plays a significant role in novel discoveries. We propose visualization of metagenomic data because of their high dimensionality and complexity. Using visualization of metagenomic data gives a complete overview of the information in the data and makes thorough exploitation of personalized medicine possible. It is necessary to clearly represent systematic, functional and other properties with enough detail to explain the structures and biological knowledge. One of the most popular programming languages is R, which we recommend for developing a graphical interface in a code-based environment.

In this study, a novel method is proposed, called *geneFS2Img*, to generate an image of gene family abundance after feature selection (FS) processing. A very large number of features, which can be up to nearly two million features, are compacted in images of 24 x 24 pixels with a new dimension of 576 (compressed more than 3000 times). Then, these images are input into a deep-learning architecture to make predictions. This study makes several contributions. Firstly, we introduce a compact way to visualize datasets with a very large number of features using a feature selection algorithms and the Fill-up approach [9].

The method provides a solution for visualizing gene data in 2D images with selected features, representing important features that have a large influence on the classification of patients. Secondly, *geneFS2Img* improves not only prediction accuracy but also inference time. Analytical results revealed that gene family abundance is potentially advantageous for predicting colorectal cancer and inflammatory bowel disease. Thirdly, although we only performed prediction on 576 selected features out of close to two million features, the results were comparable to most datasets using species abundance. We note that the number of selected features in this study approximated the number of species features in [9].

2 Related Work

Machine learning in general and deep learning in particular have a significant impact on many fields of science and technology, especially in medicine. We

believe that human-machine collaboration is essential in order to achieve ambitious goals in medicine, clinical disease diagnosis, disease prevention, personalized medicine, personalized prognosis and drug development. Thanks to the rapid development of information technology, many technology-based tools are becoming more and more popular for data analysis.

Other studies used various different machine learning classification methods, including a decision tree-based method, random forest (RF) [10], naive Bayes (NB) [11], and support vector machine (SVM) [12]. The most successful model is the neural network model in supervised learning algorithms that are widely used to predict disease genes. On the other hand, clustering is regularly done using unsupervised learning. A comparison of classification-based methods can be found in Le, *et al.* [13]. Moreover, deep learning uses different processing layers with linear and nonlinear transformations that are a high-level abstraction. Isolating the problem by applying biological knowledge, the neural network approach was developed inspired by the structure and function of neurons.

In other studies, researchers used algorithms such as mRMR (Min Redundancy Max Relevance) [14], Lasso (Least Absolute Shrinkage and Selection Operator) [15], Elastic Net [16]. In addition, Hilal, *et al.* applied a number of methods such as Conditional Mutual Information Maximization (CMIM), Fast Correlation Based Filter (FCBF), mRMR and eXtreme Gradient Boosting (XGBoost) [17]. Because the analysis of high-dimension data requires hundreds or thousands of variables, feature selection (FS) was seen as a priority. Therefore, Hilal, *et al.* investigated filter feature selection approaches for informative feature detection in gene expression microarray (GEM) analysis, also called differentially expressed gene (DEG) discovery, for gene prioritization and biomarker discovery [18]. Some of the most commonly used tools were developed with four classification methods (SVM, RF, Lasso and ENet) in various domains. Gene family abundance datasets are very complex and highly dimensional so that data visualization is difficult. The authors of [9] presented techniques for generating synthetic images by a supervised method to visualize metagenomic data based on comparison of linear discriminant analysis (LDA) with t-SNE [9].

3 Data Description

The advantages of metagenomics have been explained in Section 1. Applying machine learning to metagenomic data is a basic approach in the bioinformatics domain. One of the publicly available datasets that we used is provided by the HMP Unified Metabolic Analysis Network (HUMAnN2) [19] and has millions of features. The main source of the dataset was downloaded from *curatedMetagenomicData* [20] in R.

The six gene family abundance datasets provide vast knowledge of the causes of diseases such as inflammatory bowel disease (IBD) [21], liver cirrhosis (CIR) [22], colorectal cancer (COL) [23], obesity (OBE) [24] and type 2 diabetes (T2D) [25]. We also evaluated a dataset with data from 96 European women; 53 WT2 patients and 43 women who were not affected by the disease. Through the investigations, we have generalized and summarized the genetic diversity in Table 1.

Table 1 Description of detailed information on six gene family abundance datasets.

Dataset name	CIR	COL	IBD	OBE	T2D	WT2
#features	1,747,534	1,796,274	1,730,384	1,519,375	1,690,774	1,415,610
#samples	232	121	110	253	344	96
#patients	118	48	25	164	170	53
#controls	114	73	85	89	174	43
Ratio of patients	0.51	0.40	0.23	0.65	0.49	0.55
Ratio of controls	0.49	0.60	0.77	0.35	0.51	0.45

D is the set of six considered gene family abundance datasets, so we have:

D = $\{d_1, d_2, d_3, d_4, d_5, d_6\}$, with d_1 = CIRgene, d_2 = COLgene, d_3 = IBDgene, d_4 = OBEgene, d_5 = T2Dgene, d_6= WT2gene, d = 1 ... 6.

Let's say that: S_i= $\{s_1, s_2, \dots, s_n\}$ includes n samples in d_i; F_i= $\{f_1, f_2, \dots, f_m\}$ reveals m features corresponding to d_i; P_i= $\{p_1, p_2, \dots, p_k\}$ represents k patients who were affected by the disease corresponding to d_i, C_i= $\{c_1, c_2, \dots, c_k\}$ consists of k controls/healthy individuals that belong to d_i.

$$Matrix(C) = \begin{pmatrix} \text{CIR} & 232 & 1747534 & 118 & 114 \\ \text{COL} & 121 & 1796274 & 48 & 73 \\ \text{IBD} & 110 & 1730384 & 25 & 85 \\ \text{OBE} & 253 & 1519375 & 164 & 89 \\ \text{T2D} & 344 & 1690774 & 170 & 174 \\ \text{W T2} & 96 & 1415610 & 53 & 43 \end{pmatrix}$$

All abundance features in one sample sum up to 1:

$$\sum_{i=1}^{k} f_i = 1$$

where k indicates the number of features of a sample and f_i denotes the value of the $i-th$ feature.

4 Method

Figure 1 describes the 6 steps of our method in detail. The original gene family data are filtered by feature selection and after that images are generated from the new set of selected features before inputting them into a learning architecture such as Linear Regression or CNN2D. Because of the massive number of features in this dataset, we propose the Perceptron algorithm (Perceptron Weight Based Filter) to reduce irrelevant features from a training set mentioned in step 2. After filtering in step 3, informative features are collected by using feature selection. We propose to use image generation in step 4, making parallel data visualization increasingly accurate when we reduce the size of the images equal to 24 x 24 pixels. Last but not least, Linear Regression and Convolutional Neural Network play an important role in training in step 5. Finally, we use performance metrics to evaluate our used models.

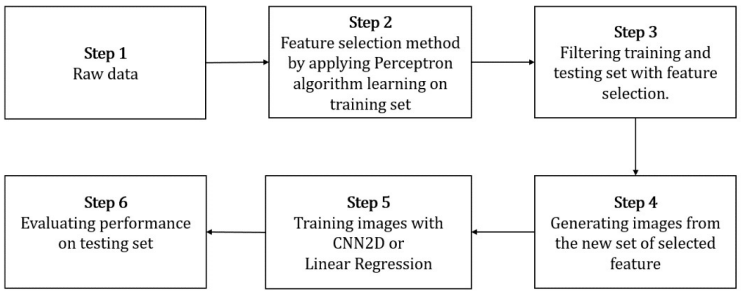

Figure 1 The overall framework of geneFS2Img.

4.1 Feature Selection for Gene Family

The main motivation for proposing a feature selection method [26] was to improve classification accuracy by employing appropriate feature selection to achieve better performance. In this study, we identified informative genes to construct a good model and exclude irrelevant genes from millions of genomes in the dataset without much loss of information. The reason why addressing large-scale data can reduce high dimensionality to avoid many challenges if costly measurement as well as decreasing the number of unneeded features that unfortunately impact the result.

The number of features was reduced to simplify and better comprehend the model. Applying feature selection is essential to achieve accurate predictions for various diseases in view of personalized medicine. After investigating several methods, we propose to use feature selection following the feature-weighting [27] method with a perceptron neural network.

As stated above, the proposed method aims to select a finite number of features (much less than the number of available features) that can be related to the disease. The main idea is that, at first, we want to determine the 'weight' of each feature by applying the perceptron algorithm on the selected training dataset, followed by removing features that have a low 'weight', only keeping ***n*** features from the training set as well and the test set for the training stage. Theoretically, ***n*** is a reasonable positive number; we set *n* = 576 to generate images with a size of 24 x 24 for image classification. The following section elaborates the main idea and the related algorithm to show how selection is executed.

Now, we introduce how the perceptron algorithm works. Firstly, we should understand the definition of weight sum, the activation function, the loss function and the optimization algorithm.

Weight sum:

$$f(x) = \sum_{i=0}^{n} w_i \, x_i$$

where: w_i is the weight of feature $i (i > 0), x_0 = 1$

Activation function: *g(u)* results in perceptron decisions. In this study we used the following Heaviside step function:

$$g(u) = \begin{cases} 1, & if \ u > 0 \\ 0, & otherwise \end{cases}$$

Activation result: $g(f(x))$

Loss function:

$$h(w) = \frac{1}{2} \sum_{i=1}^{n} \left[g\left(f(x_i)\right) - y_i \right]^2$$

where $w = (w_0, w_1, w_2, \ldots, w_m)$ represents the weights of m features, w_0 is the bias. Our goal is to minimize $h(w)$. We update weights (w_j) using the following approach:

$$w_j = w_j + \eta \sum_{i=1}^{n} x_{ij} . \left(g\left(f(x_i)\right) - y_j\right)$$

where:

1. n is the number of samples
2. w_j is the weight of feature j, w_0 is the bias
3. $x_i = (x_{i0}, x_{i1}, x_{i2}, \ldots x_{im})$ is the vector representing sample $i - th, x_{i0} = 1$
4. η is the learning rate; in this study we consider n equals 0.01

Computing w_i to minimize $h(w)$ is done following the pseudocode shown in Algorithm 1.

Algorithm 1 Perceptron

- Function: $Perceptron(x)$
- Input: x is the dataset, where x_i denotes the sample $i-th$ and x_{ij} is feature $j-th's$ value of sample $i-th$
- Output: Weights after optimizing $h(w)$

Pseudo code:

function $Perceptron(x)$:
begin
$i \leftarrow 0$
While $i \leq n$ **do**
$w_i \leftarrow 0$
$i = i + 1$
end while
$\eta = 0.01$
$H \leftarrow \frac{1}{2}\sum_{i=1}^{n}[g(f(x_i)) - y_i]^2$
$count \leftarrow 1$
While $count \leq 1000$ **do**
$newW \leftarrow w$
$j \leftarrow 1$
While $j \leq m$ **do**
$sum \leftarrow 0$
$i \leftarrow 1$
While $i \leq n$ **do**
$sum \leftarrow 0$
$i \leftarrow 1$
While $i \leq n$ **do**
$sum \leftarrow sum + \eta . x_{ij} . (g(f(x_i)) - y_j))$
$i \leftarrow i + 1$
end while
$j \leftarrow j + 1$
$newW_i \leftarrow newW_i + sum$
end while
$w \leftarrow newW$
$NewH \leftarrow 0$
$i \leftarrow 1$
While $i \leq n$ **do**
$NewH \leftarrow NewH + \frac{1}{2}[g(f(x_i)) - y_i]^2$
$i \leftarrow i + 1$
end while
if $NewH - H \geq 0$ **then**
break
end if
$count \leftarrow count + 1$
end while
return W
end

The next step of our method takes advantage of the perceptron algorithm shown in Algorithm 2. The function *Select*$(x, number)$ will eventually return indices of selected features as a set.

Algorithm 2 Perceptron weight-based filters

- **Function** $Select(x, number)$
- **Input**: x_i is the dataset $0 \leq i \leq number\ of\ features$, the number is the desired number of features that will be selected
- **Output**: Indices of selected features

Pseudo code:
function $Select(x, number)$:
begin
$weights \leftarrow Perceptron(x)$
$W \leftarrow \{\}$
$i \leftarrow 1$
$m \leftarrow number\ of\ feature\ in\ x$
While $i \leq m$ **do**
  $append\ tuple\ (|weights_i|, i)$ to W
 $i \leftarrow i + 1$
end while
Sort tuples in W using the following criteria: $tuple\ a \geq tuple\ b\ iff\ a_0 \geq b_0$
$indices \leftarrow \{\}$
$i \leftarrow 1$
while $i \leftarrow \min(number, m)$ **do**
  $append\ W_{i,1}\ to\ indices$
  $i \leftarrow i + 1$
end while
return $indices$
end

4.2 Visualization of Gene Family Abundance

Deep learning provides numerous robust algorithms to improve prediction tasks. One of them is convolutional neural networks (CNN), which has shown very promising performance, especially in image classification. We leverage the power of deep learning in image classification by converting gene abundance to 2D images using Fill-up [9]. The gene families were chosen from feature selection processing. The new set of features is presented in a square matrix with the size depending on the number of features. With 576 selected features, the size can be 24 x 24 pixels to cover the whole new set of features.

Initially, at the top of the image, the features are arranged from the first feature to the 24th feature from right to left. The second row is organized the same way, starting from the 25th feature. This procedure is repeated for the *i*-th row until the last feature is filled in the image. In the examples shown in Figure 2, the obtained features are presented by color and gray images.

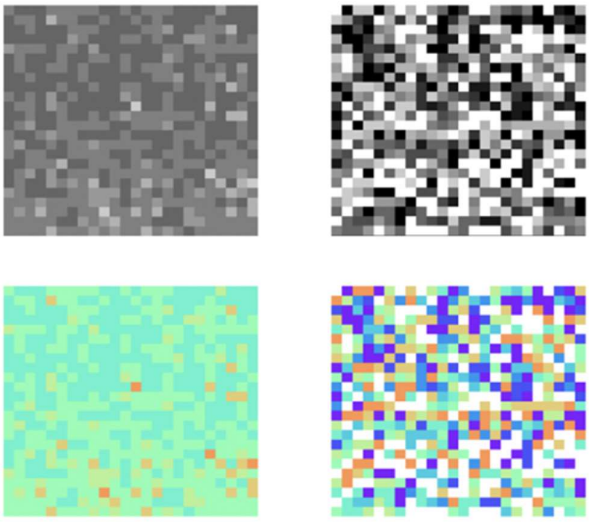

Figure 2 Gene family abundance visualization by feature selection and Fill-up. The first row presents gray images while the other row shows images with a rainbow colormap. The global maps (in the first column) show all coordinates of the features with the average value of all features in the training set. We also show a sample with the colormap given in the second column.

4.3 Deep-learning Architecture

Convolutional neural network was mainly used in this research to classify which situations are under control. The CNN architecture for images (CNN2D) presented in [9] is rather simple, but it showed promising results on metagenomic images. Since our method chooses 576 features from the gene dataset, the input of the network includes images of size 24 x 24 pixels with three colors channel (or one channel for gray images). The filter size of the convolutional layer is 3 x 3 using 64 filters, followed by a max pooling layer with a kernel size of 2 x 2 with a stride of 2. The convolutional layers were then flattened and connected to a neural network layer, which results in one output (as shown in Figure 3). With respect to FC, 2D samples were received as input, then using a sigmoid function to produce an output, which determines the probability of suffering from the disease. For all models used in the experiments, we used the Adam optimizer function with a default learning rate of 0.001 and a batch size of 16. Because we foresaw overfitting, we used the early stopping technique with an patience epoch of 5. The proposed method was developed based on the *deepmg* framework [9] in Python. The method leverages modules in the *scikit-learn* library [28], Matplotlib [29], Keras [30], Tensorflow [31].

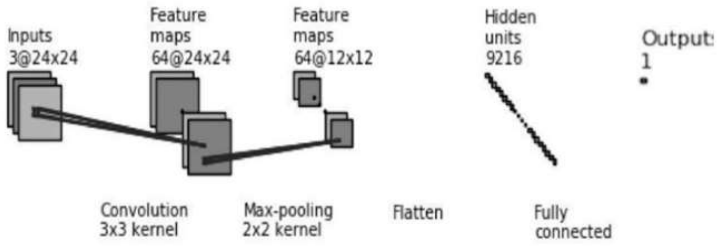

Figure 3 Example of a shallow CNN architecture for metagenomics [9].

5 CNN2D Architecture for Prediction Tasks Conducted From [9] Results

5.1 Performance Metrics

Evaluating the model or the algorithm used is an important step in determining the performance of the model and evaluating more exact conclusions. There are many measurement metrics to evaluate model performance. In this study, we used the following three indicators: accuracy, p-value and time consumption (in seconds) to compare classifiers. Accuracy is the ratio between the number of correct predictions and the total number of input samples. The p-value is the probability of obtaining results at least as extreme as the observed results of a statistical hypothesis test. Consumed time is the time required to perform the calculations of the proposed method. The performance was expressed as average accuracy based on 10-fold cross-validation repeated 10 times.

5.2 Experimental Results

5.2.1 Proposed Method Speeds Up Learning

In Figure 4 we can see clearly that consumed time was reduced significantly by hundreds of times in comparison between raw data and the Fill-up method with feature selection using the FC model. The consumed time was high on the T2D dataset, reaching more than 20 hours on average to process raw data, while for the smallest dataset, WT2, the least time was consumed, i.e. about 6 hours.

The amount of time spent was strikingly similar between COL and IBD on raw data. On the other hand, using feature selection dramatically reduced the consumed time for almost all used datasets, where the least time was required for the WT2 dataset. After using the FC model for feature selection, the most time was required for the T2D dataset, but there was still a considerable time reduction.

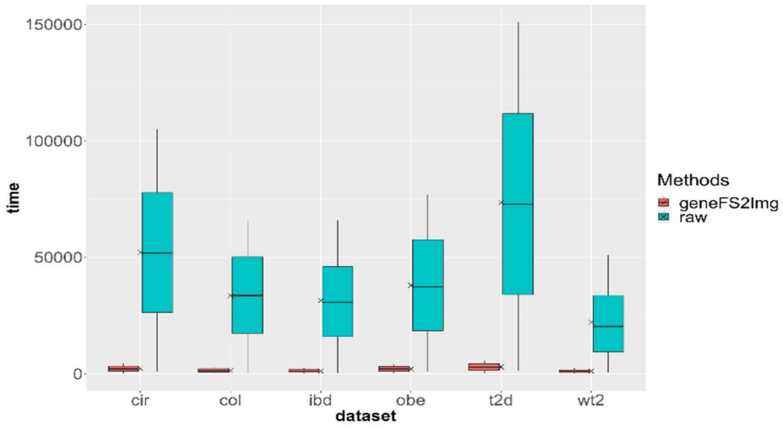

Figure 4 Boxplot showing total run time (in seconds) of the FC model running various datasets, compared between raw data and data visualized by *geneFS2Img*.

5.2.2 GeneFS2Img Outperforms State-of-the-Art Methods

Table 2 presents a comparison of the performance of the proposed approach with the state-of-the-art methods in [9]. Although [9] using PCA and RN_PRO for dimensionality reduction performed poorly, it is worth noting that *geneFS2Img* achieved significant improvements compared to [9]. CIR, COL, IBD all had outstanding results with gene family abundance in [9], but the authors did not present standard deviation values for performance, so when using FC on raw data we used 10-fold cross-validation repeated 10 times to get the t-test statistic value.

Table 2 Performance comparison between the proposed method and FC model[9].

Method	CIR	COL	IBD	OBE	T2D	WT2	AVG
PCA* [9]	0.547	0.604	0.775	0.648	0.514	0.540	0.605
RD_PRO*[9]	0.555	0.605	0.775	0.648	0.496	0.530	0.602
Raw [9]	0.761	0.628	0.775	0.648	0.655	0.620	0.681
geneFS2Img (FC with gray image)	**0.872**	**0.811**	**0.852**	0.635	0.610	**0.645**	0.738
geneFS2Img (FC with color image)	**0.855**	**0.785**	**0.852**	0.624	0.619	**0.635**	0.728
geneFS2Img (CNN with gray image)	**0.847**	**0.813**	**0.852**	0.642	0.633	**0.647**	0.739
geneFS2Img (CNN with color image)	**0.841**	**0.779**	**0.838**	0.627	0.618	**0.657**	0.727

Note: * denotes the performance in [9] using dimensionality reduction by PCA (principal component analysis) and RD_PRO (random projection)

The p-value was used for investigating meaningful improvements (p-value < 0.05) when comparing between our method using *geneFS2Img* and raw data. In addition, we obtained 5 out of 6 significant results on the CIR, COL, IBD, T2D and WT2 datasets with p-values of 2.2E-16, 2.2E-16, 8.585E-15, 0.00007934 and 0.00615, respectively, while we also gained a p-value of 0.061 on the OBE dataset by using FC model. For the diseases considered in this study (liver cirrhosis, colorectal cancer and IBD) we obtained the best prediction while the prediction model performed poorly on obesity and type 2 diabetes.

5.2.3 Results Analysis of Diseases Diagnosis

As shown in Figure 5 we calculated the sensitivity by dividing the number of actual patient predictions with the total number of patient predictions. It is noteworthy that the prediction on CIR had high reliability. It could accurately predict whether the patient's situation was under control or under threat with a sensitivity value of approximately 89%. Applying the same method on COL and IBD also resulted in models with acceptable sensitivity, fluctuating in the range of 75% to 86%. It is noteworthy that predicting samples to know whether patients did not get IBD was more accurate than predicting samples affected by the disease, with the accuracy of each prediction being 75.44% and 86.98%, respectively. With a small decrease in difference but the same trend as IBD, COL also revealed a higher accuracy in prediction of a healthy status for this disease. In contrast, obesity predictions with a positive result were more accurate than negative results.

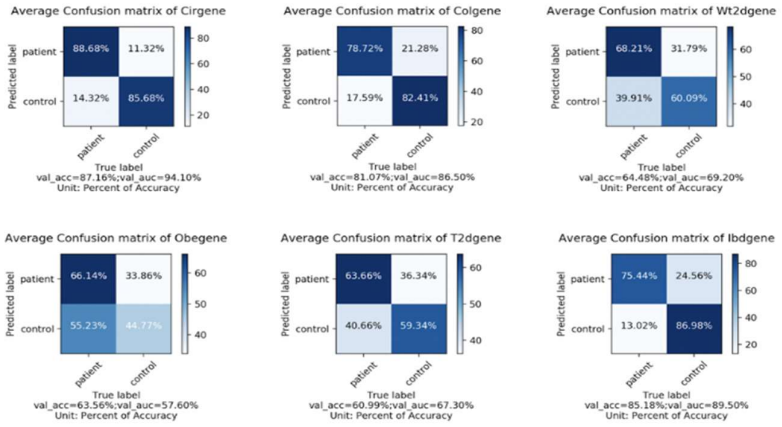

Figure 5 Average confusion matrix for 6 gene family abundance datasets.

5.2.4 Selected Features are Meaningful for Predicting Diseases

The presented bar charts shown in Figure 6 delineate the number of features that appeared after 10 runs with 10-fold cross-validation and concentrate on the frequency of selected times from 0 to 100 for CIR and COL as examples. The graphs provide information about the proportion of significant selected features after 100 runs. All of the datasets had 576 features after reducing dimensionality. The CIR and COL datasets had the highest proportion of features that appeared 100 times, i.e. 0.239 and 0.255 respectively. The charts show a trend of mild fluctuations of features appearing 0-80 times. Furthermore, the number of features had a marked rise between 90 and 99 runs and had a record high at 100 runs for all of the datasets. The detailed information shown by CIR's chart clearly shows that the figures remained constant for about 10 features and more at considered milestones such as 60, 80 runs. In addition, COL's chart had a period of instability along 0-90 runs but then the figures climbed back up again. Other datasets also revealed similar patterns.

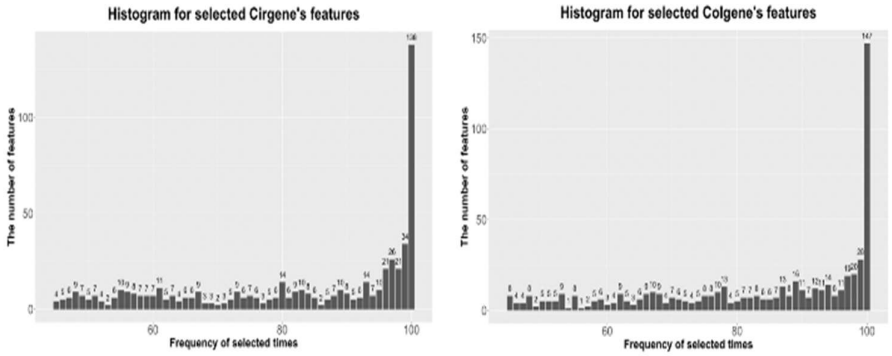

Figure 6 Filtering important selected features by using geneFS2Img on gene family datasets.

Identifying and selecting critical features from high-dimensional data in prediction models is essential because using datasets containing millions of features can affect the accuracy. Many different algorithms could be applied for feature selection, depending on the case at hand and the preference of the user. After careful consideration from different perspectives, *geneFS2Img* was chosen in this study. The promising results of this study open up possibilities for future research in the personalized medicine domain.

6 Conclusion

We presented *geneFS2Img* to illustrate gene family abundance in very compact images based on a number features reduced through a selection process. The performance shows promising results that reveal that the selection algorithms retained relevant features. The results and modules for *geneFS2Img* are available at https://github.com/hainguyenct/geneFS.

Compared to the results with raw data, we obtained significant improvements in both prediction accuracy and inference time. The consumed time was reduced remarkably with the sets of selected features compared to the original sets with all features.

As revealed by the results, the proposed method appears to be an appropriate tool for predicting liver cirrhosis, colorectal cancer, and IBD. Some results using gene family abundance data even outperformed the results for bacterial species abundance. We note that the reduced set of features was approximate to the number of bacterial species in the comparison of this study.

References

[1] Academy of Medical Sciences, *Stratified, Personalised or P4 Medicine: a New Direction for Placing the Patient at the Centre of Healthcare and Health Education (Technical Report)*, 2015.

[2] Smith, R., *Stratified, Personalised, or Precision Medicine*, British Medical Journal, 15 October 2012.

[3] Dudley, J.T. & Karczewski, K.J., *Exploring Personal Genomics*, Oxford : Oxford University Press, 2014. DOI: 10.1093/acprof:oso/9780199644483.001.0001.

[4] Meiliana, A., *Personalize Medicine: The Future of Health Care,* Indonees Biomed J., **8**(3), pp. 127- 146, 2016, DOI: 10.18585/inabj. v8i3.271.

[5] Handelsman, J., *Metagenomics: Application of Genomics to Uncultured Microorganisms*, Microbiol Mol. Biol. Rev., **68**, pp. 669-684, 2004.

[6] McCall, B.,*COVID-19 and Artificial Intelligence: Protecting Health-Care Workers and Curbing the Spread,* The Lancet Digital Health, **2**(4), e166-E167, 2020. DOI: 10.1016/ S2589-7500(20)30054-6.

[7] Behjati, S. & Tarpey, P.S., *What is Next Generation Sequencing,* **98**, pp. 236-238, 2013. DOI:10.1136/archdischild-2013-304340.

[8] TH, N., *Disease Prediction Using Synthetic Image Representations of Metagenomic Data and Convolutional Neural Networks,* The 13th IEEE-RIVF International Conference on Computing and Communication Technologies 2019, Da Nang 20-22/03/2019; pp. 231-236, 2019.

[9] Breiman, L., *Random Forests.* Mach Learn, **45**, pp. 5-32, 2001. DOI: 10.1023/A:1010933404324.

[10] Lewis, D.D. *Naïve (Bayes)at forty: The Independence Assumption in Information Retrieval,* in Nédellec C., Rouveirol C. (eds) Machine Learning: ECML-98, ECML 1998, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics*), **1398**, pp. 4-15, 1998. DOI: 10.1007/BFb 0026666.

[11] Vapnik, V., *Support Vector Machine*. Mach Learn, **20**, pp. 273-297, 1995.

[12] Le, D., Hoai, N.X. & Kwon, Y., *Knowledge and Systems Engineering*, **326**, pp. 577-588, 2015.

[13] Cai, L., Wu, H., Li, D., Zhou, K., & Zou, F., *Type 2 Diabetes Biomarkers of Human Gut Microbiota Selected Via Iterative Sure Independent Screening Method,* PloS one, **10**(10), e0140827, 2015.

[14] Pasolli, E., *Machine Learning Meta-Analysis of Large Metagenomic Datasets: Tools and Biological Insights,* PLOS Computational Biology, **12**(7), e1004977, 2016. DOI: 10.1371/journal.pcbi.1004977.

[15] Zou, Hui, & Hastiem, T., *Regularization and Variable Selection via the Elastic Net,* Journal of the Royal Statistical Society: Series B (Statistical Methodology), **67**(2), pp. 301-320, 2005.

[16] Hacılar, Hilal., *Inflammatory Bowel Disease Biomarkers of Human Gut Microbiota Selected via Ensemble Feature Selection Methods,* 2020.

[17] Lazar, C., Taminau, J., Meganck, S., Steenhoff, D., Coletta, A., Molter, C., de Schaetzen, V., Duque, R., Bersini, H. & Nowe, A., *A Survey on Filter Techniques for Feature Selection in Gene Expression Microarray Analysis,* **9**(4), pp. 1106-1119, 2012. DOI: 10.1109/TCBB.2012.33

[18] Abubucker, S., *Metabolic Reconstruction for Metagenomic Data and Its Application to the Human Microbiome*, **8**, p. e1002 358. ISSN 1553-7358. http://journals.plos.org/ploscompbiol/article?id=10.1371/journal .pcbi.1002358.

[19] Pasolli, E., *Accessible, Curated Metagenomic Data Through ExperimentHub,* **14**, pp. 1023-1024, 2017.

[20] Xu, J., *Systematic Comparison of Two Animal-to-Human Transmitted Human Coronaviruses: SARS-CoV-2 and SARS-CoV,* Viruses 2020, **12**(2), pp. 244, 2020, DOI: 10.3390/v12020244. 2020.

[21] Qin, J., *A Human Gut Microbial Gene Catalogue Established by Metagenomic Sequencing,* **464**(7285), pp. 59-65, 2010, DOI: 10.1038/nature08821 PMID: 20203603.

[22] Qin, N., *Alterations of the Human Gut Microbiome in Liver Cirrhosis.* Nature, **513**(7516), pp. 59-64, 2014, DOI: 10.1038/nature13568.

[23] Zeller, G., Tap, J., Voigt, A.Y., Sunagawa, S., Kultima, J.R., Costea, P.I., Amiot, A., Böhm, J., Brunetti, F., Haberman, N., Hercog, R., Koch, M., Luciani, A., Mende, D.R., Schneider, M.A., Schrotz-King, P., Van Nhieu,

J.T., Yamada, T., Zimmerman, J., Benes, V., Kloor, M., Ulrich, C.M., Doeberitz, M.v.K., Sobhani, I. & Bork, P., *Potential of Fecal Microbiota for Early-Stage Detection of Colorectal Cancer*, Mol Syst Biol, **10**, 766, 2014. DOI: 10.15252/msb.20145645.

[24] Le Chatelier, E, Nielsen, T., Qin, J., Prifti, E., Hildebrand, F. & Falony, G., *Richness of Human gut Microbiome Correlates with Metabolic Markers.* Nature, **500**(7464), pp. 541-546, 2013. DOI: 10.1038/nature12506 PMID: 23985870.

[25] Qin, J., Li, Y., Cai, Z., Li, S., Zhu, J. & Zhang, F., *A Metagenomewide Association Study of Gut Microbiota in Type 2 Diabetes*. Nature 2012; **490**(7418), pp. 55-60, 2012, DOI: 10.1038/nature11450 PMID: 23023125.

[26] Ditzler, G., Morrison, J.C., Lan, Y. & Rosen, G.L., *Feature Subset Selection for Metagenomics*, **16**, 358, 2015. DOI: 10.1186/s12859- 015-0793-8

[27] Blum, A.L. & Langley, P., *Selection of Relevant Features and Examples in Machine Learning*, **97**(1-2), pp. 245-271, 1997. DOI: 10.1016/S0004-3702(97)00063-5.

[28] Garreta, R. & Moncecchi, G., *Learning Scikit-learn: Machine Learning in Python*, Birmingham, United Kingdom, Packt Publishing Ltd, 2013.

[29] Hunter, J.D., *Matplotlib: A 2D Graphics Environment*, Computing in Science & Engineering, **9**(3), pp. 90-95, 2007.

[30] Chollet, F., *Keras*, https://keras.io (2015).

[31] Abadi, M., *TensorFlow: Large-scale Machine Learning on Heterogeneous Systems*, software available from tensorflow.org. (15 February, 2020)