2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

# A New GP-based Wrapper Feature Construction Approach to Classification and Biomarker Identification

Soha Ahmed and Mengjie Zhang
School of Engineering and
Computer Science
Victoria University of Wellington
PO Box 600, Wellington 6014, New Zealand
Email:{soha.ahmed, mengjie.zhang}@ecs.vuw.ac.nz

Lifeng Peng
School of Biological Sciences
Victoria University of Wellington
PO Box 600, Wellington 6014, New Zealand
Email: lifeng.peng@vuw.ac.nz

***Abstract*—Mass spectrometry (MS) is a technology used for identification and quantification of proteins and metabolites. It helps in the discovery of proteomic or metabolomic biomarkers, which aid in diseases detection and drug discovery. The detection of biomarkers is performed through the classification of patients from healthy samples. The mass spectrometer produces high dimensional data where most of the features are irrelevant for classification. Therefore, feature reduction is needed before the classification of MS data can be done effectively. Feature construction can provide a means of dimensionality reduction and aims at improving the classification performance. In this paper, genetic programming (GP) is used for construction of multiple features. Two methods are proposed for this objective. The proposed methods work by wrapping a Random Forest (RF) classifier to GP to ensure the quality of the constructed features. Meanwhile, five other classifiers in addition to RF are used to test the impact of the constructed features on the performance of these classifiers. The results show that the proposed GP methods improved the performance of classification over using the original set of features in five MS data sets.**

## I. Introduction and Background

Mass Spectrometry (MS) is a powerful analytical tool for proteomics or metabolomics. The mass spectrometer measures the molecular weight and the relative abundance of compounds in samples [1]. The samples pass through an ionizer to facilitate the detection of compounds, and, therefore, each mass value is accompanied by a charge ratio. There are two common methods of ionization in the mass spectrometer which are the matrix-assisted laser desorption ionization (MALDI) and surface-enhanced laser desorption ionization (SELDI). MALDI ionizes the enzymatic peptides of proteins through co-crystallization of peptides and a weak acid matrix followed by laser shooting on the peptide-matrix mixture [2]. SELDI ionizes proteins that are captured on a chemically modified surface with the addition of a matrix solution on the surface followed by laser desorption ionization [2]. The MALDI and SELDI are typically combined with a time of flight (TOF) mass spectrometer [2]. The MS spectrum contains the mass to charge ratios ($m/z$) and intensities of compounds which can be used for identification and quantification of these compounds [1]..Each spectrum produced from the mass spectrometer is composed of thousands of features ($m/z$ values and their corresponding intensities) but at the same time the number of spectra is typically very small. The set of features that can discriminate between the different classes of spectra are typically referred to as the *biomarkers* [3]. Biomarkers are the indicators of biological processes which can predict a benefit or harm based on epidemiological, therapeutic, physiological evidence [3]. Biomarkers are useful for monitoring therapeutic interventions and early diagnosis of cancer [4].

Feature reduction refers to the process of removing irrelevant or redundant features through feature selection or construction. Feature construction is the process of transforming the original input features into new features [5]. The new features have potential to improve the classification performance and also reduce the dimensionality of the input space. Feature construction can be performed through a wrapper, filter or an embedded approach. In the wrapper approach, the evaluation of the constructed features is done through a classifier while the filter approach does not use a classifier for evaluation. In the embedded approach, there is no interaction between the classifier and the feature construction process. The construction of features is a complex process as it is based on discovering the hidden relationships between features. Hence, a powerful method should be used for feature construction.

Genetic programming (GP) is an evolutionary algorithm that evolves computer programs [6]. GP searches for a solution of a user defined problem by optimizing a population of computer programs. It measures the goodness of the solutions through an objective function that evaluates the program through its ability to perform a specific computational task [6]. Similar to other evolutionary algorithms, GP starts with a random initial population of individuals as candidate solutions and then selects the fittest individuals, produces new individuals through its genetic operators. Each individual is assigned a fitness value according to the fitness function. Finally, the program with the best fitness is taken as a solution for the problem [6]. There are several ways of encoding the GP programs such as tree-based GP [6], linear GP [7], cartesian GP [8] and grammar-based GP [9]. The focus here is on the tree-based GP as it is the most commonly used method and it is easier to construct features from it. The program in the tree-based GP is represented by a set of nodes where each node is either a terminal or a function. On the one hand, the function

978-1-4799-1488-3/14/$31.00 ©2014 IEEE

node, which is an operation, has children that are arguments of that operation. On the other hand, the terminal nodes are inputs to the programs and they do not have children.

Feature construction aims to find a set of new features which when used improve the predictive power of the classifiers. GP can be a good choice for feature construction due to automatic ability to form classification models and select features [10]. The success of GP in evolving classifiers and feature selection has been a strong motivation for its use for feature construction.

There have been two main scenarios for the use of GP for feature construction. The first scenario is to use GP for the construction of a single feature. The single constructed feature is used for classification of the data, which is not normally enough for improving the classification performance. Another solution is to use this single constructed feature with the original set of features (augmentation), and this normally increases the dimensionality. The second scenario is to use several GP trees to construct multiple features (a single feature for each class) [5]. None of the existing methods tested the construction of multiple features from a single GP tree using a wrapper approach. This paper represents such an attempt.

### A. Research Goals

This paper aims to propose two methods for feature construction GPWFC$_1$ and GPWFC$_2$, two GP wrapper multiple feature construction systems with the following properties.

1) They take a wrapper approach, which uses a classification algorithm to evaluate the features, to have the advantage of the increasing performance more than a filter approach.
2) They propose fitness measures which ensure the significant discrimination between the samples from different classes.
3) They propose an approach to construction of multiple features from a single GP tree, and, therefore, augmentation is no longer needed.

The main goal of this paper is to test the performance of the wrapper-based GP systems (methods) to construct multiple features. Specifically the following questions are going to be investigated.

1) Will the classification accuracy of using the constructed features be better than using the original features?
2) How can a single GP tree constructs multiple high-level features?
3) What is a good evaluation measure of the wrapped classifier?
4) How can the constructed features be generalised to multiple classifiers and unbiased to a specific classifier?
5) Can GP identify the biomarkers of the data sets?

### B. Organisation

The rest of the paper is organised as follows. Section II explains briefly the related work of GP for feature construction. The previous research on classification of MS data is also explained in this section. The new proposed GP methods are described in section III. The experimental set-up which includes the MS data sets, GP settings and parameters are described in section IV. Section V gives the results and discussion. Finally, the conclusions and the future work are given in section VI.

## II. Related Work

### A. Feature Construction using GP

GP has been successfully used for feature construction in two trends which are: 1) feature construction using attribute values for classification problems and 2) feature construction using raster graphics for object and edge detection problems. In the former trend, the constructed features are the results of scalar functions of original features [11]. In the latter trend which acts on images, the constructed features are the filters which operate on the raw images' raw pixel values [12]–[15].

Feature construction using GP has been done by taking a filter, a wrapper or an embedded approach. In the filter approach, some sort of information-theoretic measure such as information gain, information gain ratio or entropy gain has been used as fitness functions [5], [10], [16]. In the wrapper approach, the fitness is evaluated according to the performance of a specific classifier. During the fitness evaluation, the constructed feature along with the original features are fed into the classifier [11]. Some wrapper methods construct a single feature where each chromosome encodes a constructed feature, whereas other approaches use multiple trees to construct multiple features [17]. The third approach uses GP for constructing features by taking an embedded approach. In the embedded approach, there is no direct interaction between the classifier and GP. An example of using an embedded GP approach is in [18] where Fisher criterion has been used as a fitness function.

The wrapper approach usually has a better performance than the filter and embedded approaches, but its drawback is the search process, which is computationally intensive [18]. Therefore, if the time cost is not put into consideration, the wrapper approach is usually a better choice.

### B. Biomarker Detection and Classification of MS Data

MS has been successfully used to identify and quantify proteins and smaller molecules which leads to biomarker discovery [4].The analysis of MS data includes several critical preprocessing steps which include peak extraction, baseline adjustment, spectrum normalisation and alignment. Finally, the most important step is the classification of the MS data to distinguish diseases.

There have been several studies on the use of statistical methods and machine learning algorithms for biomarker detection and classification of MS data. The methods used to select biomarkers of MS data include T-statistics [19], machine learning methods (classification methods) such as decision trees [3], genetic algorithms and self-organizing-maps (SOM) [20]. Mostly, the majority of previous studies were focused on the ranking of individual features, ignoring the interactions between those features. In our previous research [21]–[23], GP was used for feature selection and classification of MS data with promising results which encouraged us to use GP for feature construction.

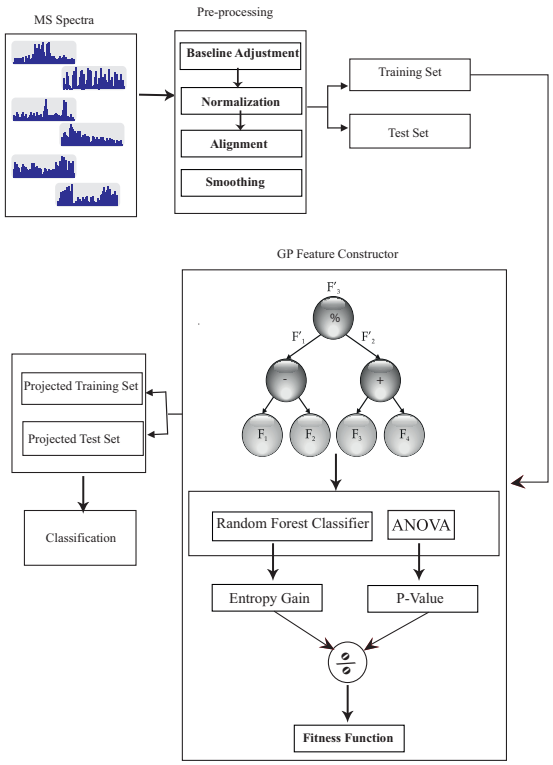


Fig. 1: Overview of the GP-based systems for feature construction.

## III. The New GP Methods

In this section, the two new GP methods for constructing multiple features are described. To construct multiple features, both the root and the subtrees of the evolved program are used.

Unlike the previous approaches which use multiple GP trees (runs) to construct many features [5], the construction of multiple features here is done from a single evolved GP tree (in a single run). The advantages of using a single tree are to reduce the computational cost and produce more new features (from the subtrees) which have the potential to increase the classification accuracy.

Figure 1 shows an overview of the two proposed GP methods. The process starts by performing the pre-processing of the data. The pre-processing consists of several steps which are baseline adjustment, spectrum normalisation, alignment and smoothing. The details of the pre-processing steps and parameters are to be explained in section IV. The pre-processed data is divided into training and test sets where 2/3 of the data are kept for training and 1/3 are held for the testing purpose. The training set is passed to GP to construct the features while the test set is held as the unseen data.

A random forest classifier is wrapped to the GP system to evaluate the goodness of the constructed features. Random forest is an ensemble method in which a group of decision trees are used together to form a forest of classifiers. The idea of random forest is that weak learners can be strong learners when they work together. The reason for selecting random forest is its good performance for both binary and multiple class classification problems [24]. The first proposed method (GPWFC$_1$) uses the classification accuracy of RF as a fitness function. The second proposed method (GPWFC$_2$) is a modification of the fitness function to be the entropy gain of RF divided by the p-value of an ANOVA test done on the features selected by GP. The p-value of the features from different classes is used to ensure significant discrimination between the classes.

An example of the process of constructing features is shown in the figure. The two original features $F_1$ and $F_2$ are combined with the addition operator to form the new feature $F'_1$, while the two features $F_3$ and $F_4$ construct $F'_2$ through the division operator. The root of the tree is used as final constructed feature $F'_3$. At the end of training, the constructed features from the best evolved program of GP (root and subtrees) are used to project the training and test sets. Finally, the projected data is used for classification.

### A. Fitness Functions

A wrapper fitness measure to feature selection and construction can usually achieve better classification performance than a filter measure [25]. Therefore, for both methods the wrapper evaluation measure is used.

**GPWFC$_1$:** The first proposed method GPWFC$_1$ uses the classification accuracy of RF as a fitness function, which is the number of correctly classified instances divided by the total number of instances.

$$\text{Fitness Fn}_1 = \frac{\text{\# Correctly classified instances}}{\text{Total number of instances}} \quad (1)$$

**GPWFC$_2$:** The second method (GPWFC$_2$) proposes a new fitness measure. An information-based evaluation of classification, which is the entropy gain [26], is used instead of using the classification accuracy. The classification accuracy is based only on a single prediction of the instance which excludes the information provided by class distribution from the classifier. The entropy gain measures the savings in bits when the classifier prediction is used to classify the instances, as opposed to using a naive method which does not have prior knowledge of the training data [26]. The entropy gain is given by:

$$EG = \sum_{i=1}^{N} entropy_{naive}(c_i|x_i) - entropy_{classifier}(c_i|x_i) \quad (2)$$

where $N$ is the total number of instances, $x$ is the instance belonging to class $c$. The entropy is

$$entropy = -log_2(P(c|x)) \quad (3)$$

where $P$ is the probability distribution provided by the classifier. In order to ensure that the constructed features can achieve significant discrimination between the classes, a one way ANOVA test is adopted for the features of the different classes. The less p-value of the ANOVA test, the more the distance between the classes. The p-value of the ANOVA test is also used in the fitness fitness function to maximise the between-class distance.

The fitness function of GPWFC$_2$ is the following:

$$\text{Fitness Fn}_2 = \frac{EG}{\textit{p-value}} \quad (4)$$

where $EG$ is the value of the entropy gain of the random forest classifier given in equation (2). The fitness function used is multi-objective where the $EG$ is maximised and the p-value is minimised.

TABLE I: Data sets summary details

| Data set | No. of spectra | No. of features | $m/z$ interval | Reference |
|---|---|---|---|---|
| Ovarian cancer (OVA) | 253(162+91) | 15,154 | 0-20,000 | [20] |
| Detection of drug-induced toxicity (TOX) | 62(28+34) | 45,200 | 799.115-12,000 | [27] |
| Premalignant pancreatic cancer (PAN) | 181(80+101) | 6771 | 800-11992.91 | [28] |
| Hepatocellular carcinoma (HCC) | 150 (78+72) | 36,802 | 799.73-10,000 | [29] |
| Detection of glycan biomarkers (DGB) | 128(78+25+25) | 16,075 | 1,499.8-5,518.3 | [2] |

## IV. Experiment set-up

This section explains the details of the experimental set-up which include MS data sets used, the pre-processing details and parameters, evaluation process and the GP settings.

### A. MS Data Sets

Five different MS data sets are used to test effectiveness of the proposed GP methods. Three are from a SELDI mass spectrometer, whereas the remaining two are from a MALDI mass spectrometer. Table I summarizes the details of the data sets. The data sets used in the experiments are the following.

- Ovarian cancer (OVA) [20]: Serum samples of female patients with ovarian cancer and unaffected women were analysed using a SELDI mass spectrometer. This data set has been one of the most analysed benchmarks because it is one of the pioneering works on MS data profiling. The objective is to detect the small set of proteomic biomarkers which has the potential to classify the two classes. The data set is composed of 253 spectra where 162 spectra are cancerous samples and 91 spectra are control samples. The range of the $m/z$ values is from 0 to 20,000 with a total of 15,154 values per spectrum.
- Detection of drug-induced toxicity (TOX) [27]: In this work, the aim is to distinguish between anthracycline- and anthracenedione-induced cardiotoxicity and control samples. Rat serum samples were analysed using a SELDI-TOF device. The number of samples in the different classes were highly unbalanced. Therefore, from this data set, we picked the definite positive and the definite negative classes to avoid the imbalance. The $m/z$ values range from 799.115 to 12,000. The data set consists of 62 spectra (28 in the definite positive class and 34 samples in the definite negative class) where each spectrum has 45,200 $m/z$ readings.
- Premalignant pancreatic cancer (PAN) [28]: Serum from cancerous and healthy samples were analysed using SELDI-TOF technology. The data set contains 80 spectra from the cancer group and 101 samples from the healthy group. The number of features ($m/z$ readings) is 6771 and the $m/z$ values range from 800 to 11992.91.

TABLE II: Pre-processing running parameters

| | OVA | TOX | PAN | HCC | DGB |
|---|---|---|---|---|---|
| Window size for baseline removal | 500 | 200 | 200 | 50 | 200 |
| Smoothing frame size | 5 | 6 | 3 | 6 | 6 |
| Maximum intensity after normalisation | 300 | | | | |

- Hepatocellular carcinoma (HCC) [29]: This data set is generated using a MALDI-TOF mass spectrometer. The samples are from patients suffering from hepatocellular carcinomas and from healthy individuals. The number of samples in this data set is 150 (78 are affected patients and 72 are non-affected persons). The number of $m/z$ readings is 36,802 and the $m/z$ values range from 799.73 to 10,000.
- Detection of glycan biomarkers (DGB) [2]. This data set contains three groups of samples (78 healthy control samples, 25 hepatocellular carcinoma and 25 chronic liver samples). The samples were generated from a MALDI-TOF device and the aim was to select glycan structures in order distinguish the different groups. The $m/z$ values are in the interval of 1499.8 to 5518.3 with a total number of readings of 16,075.

### B. Pre-Processing

The pre-processing of MS data is an elementary and a critical stage of the analysis framework. The pre-processing converts the data from the raw form to an homogeneous matrix which constitutes the input for the feature selection and classification algorithms [2]. As discussed earlier, the pre-processing steps used in our experiments include baseline adjustment, spectrum normalisation, alignment and smoothing. The Matlab bioinformatics toolbox [30] is used to perform the pre-processing. The baseline removal is used to remove the low-range noise. The baseline is estimated by passing a window on the spectra and the minimum $m/z$ values are calculated. A piecewise linear interpolation method is used for regression of the baseline. In order to make the intensity values range the same, normalisation is performed. The normalisation of the spectra is done through calculating the area under the curve [27] and rescaling the spectra to have a maximum intensity value of 300. This is done by using the $msnorm$ function in the Matlab toolbox. After normalisation is performed, alignment of the peaks is performed to match the similar peaks across all the spectra. Finally, smoothing of the spectra is done to remove the low signal fluctuation. Smoothing is done through using a Savitzky-Golay filter. Table II shows the running parameters of the pre-processing steps used with each of the data sets. The parameters are selected based on the original papers of the data sets [2], [20], [27]–[29].

### C. Evaluation process

To evaluate the constructed features, the projected training and test sets are used to train and test five different classifiers in addition to random forest (RF). The aim of using the five other classifiers is to test the generalisability of the proposed methods. Since the imbalance ratio is small, the evaluation is done through the classification accuracy of the test data. The classification accuracy is the percentage of the correctly classified instances to the total number of instances in the test set.

The Waikato Environment for Knowledge Analysis (WEKA) package [31] is used to run the classification algorithms.

The five benchmark classification algorithms used are the following:

1) Multi-Layer Perceptron (MLP) classifier: this is also known as Neural networks classifier. MLP consists of layers of networks. The number of nodes in the last layer is equal to the number of classes in which the node with the maximum output indicates the predicted class.
2) Naive Bayes (NB): a probabilistic classifier based on the Bayes theorem.
3) Naive Bayes Tree (NB-tree): this classifier is a hybridization of Naive Bayes and decision tree classifiers. The leaf nodes of the decision tree use Naive Bayes as a decision stump.
4) Decision Table (DT): a set of decision tables is constructed using a possible set of features. The instances of the test set are mapped to the decision tables' cells.
5) One Rule (OneR): OneR generates a rule for each predictor in the data and selects one of these rules as the . The selected rule is the final with the minimum error.

### D. GP Running Parameters

As stated earlier, the tree-based GP is used in our experiments, which produces a single floating point as a result of the fitness evaluation [6]. The ramped half-and-half method is used to generate the initial population [10]. The function set consists of the four standard mathematical operators $\{+, -, \%, \times\}$ in addition to $max$, $min$, *If-Then-Else*, $tanh$ operators. The $\%$ is a protected division where the division of zero returns zero. The $tanh$ takes one argument while the *If-Then-Else* takes three arguments and returns the second one if the first is negative or returns the third argument otherwise.

The terminal set is composed of the intensity features of the data and a randomly generated constant between the range of [-10,10]. The crossover and mutation rates used are 0.8 and 0.19, respectively. An elitism approach is also taken with a rate of 0.01 to make sure that the performance is monotonically increasing. The number of individuals in the population is set to 400. In order to avoid bloating, the tree depth is set between 8-10. The method of selection is tournament selection and the size of the tournament used is 7. The process terminates at a maximum number of 20 generations. This number is selected as there was no further improvement when increasing the number of generations. Every training is repeated for 30 independent runs for each dataset with 30 different seeds. The total number of runs performed for both $GPWFC_1$ and $GPWFC_2$ on the five data sets is 300 (2*30*5). The best program generated from the last generation of the training data is used to construct the features for both the projected training and test sets. The GP implementation used in the experiments is the Evolutionary Computing Java-based (ECJ) package [4].

### E. Methods for Comparison

The full original set of features is applied to the same five data sets for comparison. The original features are compared with the two proposed methods using random forest (RF) classifier and also using another five classifiers namely MLP, NB, NB-Tree, DT and OneR.

To test the significance of the results, a statistical significance test (Z-test), is performed between the classification performances of the three methods (original, $GPWFC_1$, $GPWFC_2$). The confidence interval in the Z-tests is set to 95%.

For all the methods, a machine with an Intel(R) Core(TM) i7-3770 CPU @ 3.40GHz, running Ubuntu 4.6 and Java 1.7.0_25 with a total memory of 8GByte is used to run the experiments.

## V. Results and Discussion

Table III shows the experimental results of the two proposed methods, where RF is used as the classification algorithm. In the table, "All" means all the original set of features are used with RF. "Avg.No. Features" shows the average number of features constructed by $GPWFC_1$, $GPWFC_2$ and the total number of original features for each of the data sets. "Best", "Mean" and "StdDev" represent the best, the average and the standard deviation of the classification accuracy. "T-test" shows the results of T-test, where the "+" ("-") means that either $GPWFC_1$ or $GPWFC_2$ are significantly better (worse) than using all the original features. "=" means there is no significance difference between them. Moreover, the second sign, in the $GPWFC_2$ row, "$\star$" ("-" ) means that $GPWFC_2$ is significantly better (worse) than the $GPWFC_1$ , whilst "=" means there is no significance difference between them.

### A. Experimental Results of $GPWFC_1$ and $GPWFC_2$

It can be noticed from Table III that $GPWFC_1$ achieved significant improvement over using the original set of features with almost all the data sets. $GPWFC_1$ constructed new features from the original features which succeeds in achieving the two objectives of reducing the dimensionality and increasing the classification accuracy. The number of features constructed are much smaller than the original features. The average reduction of number features ranges from 96.55% to 99.43% in all the data sets. In addition, the significant increase in the classification performance ranges from 1.42% to 7.21%.

Concerning the second proposed method, $GPWFC_2$, it is mostly better than $GPWFC_1$. According to Table III, $GPWFC_2$ achieved significantly better classification performance than using all the features in nearly all the cases. For example, the average of the 30 runs of $GPWFC_2$ is better than using all the features by 5.22% in the *OVA* data set. For the *TOX* data set, the mean of $GPWFC_2$ is better by 7.21%. For *HCC* and *DGB* data sets, the classification performance of $GPWFC_2$ is significantly better by 13.88% and 10.25 %, respectively. Another important advantage of $GPWFC_2$ is reducing the dimensionality (the number of features). $GPWFC_2$ significantly reduces the number of features from thousands to less than 300 and at the same time significantly improves the classification performance.

The results suggests that the proposed methods can discover hidden information and relations between the low-level features. Both methods can select small number of features and, through the operators of the function set, constructs a smaller number of high-level features. GP constructs high-level features which contain better information than the original

TABLE III: Results of RF classification algorithm.

| Dataset | Method | RF | | | Avg. No. Features | T-test |
|---|---|---|---|---|---|---|
| | | Best | Mean | StdDev | | |
| OVA | All | 87.95 | | | 15,154 | |
| | $GPWFC_1$ | 97.59 | 91.21 | 3.86 | 202.60 | + |
| | $GPWFC_2$ | 97.59 | 93.17 | 2.48 | 257.63 | + ⋆ |
| TOX | All | 70.00 | | | 45,200 | |
| | $GPWFC_1$ | 100.0 | 77.21 | 12.15 | 190.43 | + |
| | $GPWFC_2$ | 95.00 | 77.21 | 11.12 | 188.34 | + = |
| PAN | All | 61.01 | | | 6771 | |
| | $GPWFC_1$ | 72.88 | 59.44 | 5.66 | 233.33 | = |
| | $GPWFC_2$ | 69.49 | 58.53 | 4.42 | 279.60 | -= |
| HCC | All | 59.18 | | | 36,802 | |
| | $GPWFC_1$ | 85.71 | 72.11 | 5.58 | 206.17 | + |
| | $GPWFC_2$ | 83.67 | 73.06 | 6.1 | 149.20 | + = |
| DGB | All | 60.97 | | | 16,075 | |
| | $GPWFC_1$ | 72.50 | 61.42 | 5.46 | 190.56 | + |
| | $GPWFC_2$ | 75.00 | 62.25 | 5.96 | 61.33 | + = |

features and at the same time removes the redundant and irrelevant features. Therefore, the new constructed features' average (best) classification accuracy is better than using the all the features.

The only exception is with the *PAN* data set where there is no significant difference between $GPWFC_1$ and the original features. Moreover, using all the original features with *PAN* data set is slightly better than the average performance of the 30 runs of $GPWFC_2$ but the best performance of $GPWFC_2$ is better by 8.48%. This is perhaps because RF has a feature selection capability, RF only succeeds in selecting better features with *PAN* data set which is the data set with the smallest number of features. However, it can not handle data sets with huge number of features (such as *HCC* or *TOX* data sets).

### B. The Performance with Other Classifiers

To test the generalisability of the proposed methods, the features constructed by both $GPWFC_1$ and $GPWFC_2$ are tested with other classifiers namely MLP, NB, NB-tree, DT and OneR. Table IV shows the results of using the $GPWFC_1$'s constructed features, $GPWFC_2$'s constructed features and all the original features with those classifiers. As shown in the table, for all the data sets using the original features with MLP and NB-tree was running out of memory and did not manage to produce the results.

In the case of NB, the average classification accuracy of the two new methods increases by 10.21% and 11.86%, respectively, compared to using all the features in the *OVA* data set. For the rest of the data sets, the performance of the two GPWFC methods makes an improvement between the range of 1.31%-15.22%.

Using a DT classifier, the average performance of $GPWFC_2$ is significantly better than the original features for three data sets and they are similar for two data sets. However, $GPWFC_1$ is better for three data sets and worse with the remaining two data sets. Finally, using the OneR classifier, the proposed approaches has a better average classification performance than all the features in three and four data sets, respectively. $GPWFC_1$ is worse than using the original features in *OVA* data set when used with OneR. The t-test results shows that $GPWFC_2$ is never significantly worse than using the original set of features. The previous results prove that the new methods can be generalised to any classifier and not biased to RF. However, using the classification accuracy as a fitness function can have a less chance of generalisation with other classifiers.

### C. $GPWFC_1$ vs $GPWFC_2$

As shown in Table III, the classification performance of $GPWFC_2$ succeeds in either improving the performance or maintaining the good performance of $GPWFC_1$. For example in the *OVA* data set, the average performance of $GPWFC_2$ is better than $GPWFC_2$ by 1.96%. In many cases (more than half) $GPWFC_2$ reduces the number of constructed features more than $GPWFC_1$. An example of this is, in *DGB* data set, $GPWFC_1$ constructs an average number of features of 190.56 while the average number of features constructed using $GPWFC_2$ is 61.33. Therefore, $GPWFC_2$ reduced the dimensionality by 67.82% more than $GPWFC_1$.

This suggests that the information-based evaluation of the classifier is more effective than the classification accuracy. This is due to the fact that when the classifier gives multiple decisions for an instance, it will be typically recognised as misclassified when calculating the accuracy. However, using the entropy gain evaluation means that if the classifier produces the correct class label as a second choice will not necessarily decrease the entropy gain.

Comparing Table III and Table IV, the second method ($GPWFC_2$) has more generalisation ability than the first one ($GPWFC_1$). In many cases $GPWFC_2$ with other classifiers is better than using RF. However, the performance of $GPWFC_1$ sometimes decreases when used with other classifiers. The possible reason for this is the use of the p-value of the features from different classes. Minimising the p-value helps in selecting features which achieve more significant discrimination between the classes. The selected features are then combined through the operators of GP to form a set of new features which has the potential to increase the classification performance more the selected features.

According to the t-test results in Table IV, the second proposed method is either significantly better or similar to the first one and never significantly worse. This suggests that using the information-based evaluation of classification along with the filter evaluation (p-value) has the potential to increase the performance and to generalise better than using the accuracy of the classifier for evaluation.

### D. Analysis of the Detected and the Constructed Biomarkers

In this subsection, an analysis of the detected biomarkers (features) is performed. As $GPWFC_2$ achieves better performance in all cases, the analysis is focused more on it. The amount of overlap between the two methods is also discussed here. The original selected features are analysed here in order to test which are the original $m/z$ biomarkers and how much is the reduction of dimensionality by feature construction. Examples of the detected features by $GPWFC_2$ for each data set are shown in Figure 2. The selected features in the nodes of the evolved GP program are visualised for each class. These selected features are used afterwards for constructing the new features. In Figure 2, the x-axis represents the values of the m/z ratios of the selected feature while y-axis represents the intensity values. The m/z ratios are the features identities and the intensities are the features values. The red solid line gives an example of a spectrum from the healthy group, while the blue dash line gives an example of a spectrum from the patient

TABLE IV: Results of using MLP, NB-Tree, DT, NB and OneR as the classification algorithm.

| Dataset | Method | MLP | | | | NB | | | | NB-tree | | | | DT | | | | OneR | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Best | Mean | StdDev | T-test | Best | Mean | StdDev | T-test | Best | Mean | StdDev | T-test | Best | Mean | StdDev | T-test | Best | Mean | StdDev | T-test |
| OVA | All | – | | | | 74.69 | | | | – | | | | 89.15 | | | | 85.60 | | | |
| | $GPWFC_1$ | 100.0 | 98.64 | 2.21 | | 93.97 | 84.90 | 4.30 | + | 96.38 | 89.03 | 4.25 | | 95.18 | 85.26 | 5.95 | - | 90.36 | 81.40 | 6.49 | - |
| | $GPWFC_2$ | 100.0 | 99.30 | 1.39 | = | 92.77 | 86.55 | 3.07 | +⋆ | 98.79 | 92.35 | 3.42 | ⋆ | 96.38 | 87.88 | 4.28 | =⋆ | 97.59 | 83.26 | 4.46 | += |
| TOX | All | – | | | | 75.00 | | | | – | | | | 75.00 | | | | 75.00 | | | |
| | $GPWFC_1$ | 90.00 | 66.83 | 10.95 | | 95.00 | 85.00 | 7.88 | + | 95.00 | 74.16 | 12.39 | | 100.0 | 83.33 | 14.06 | + | 100.0 | 82.16 | 11.57 | + |
| | $GPWFC_2$ | 95.00 | 70.16 | 11.41 | = | 95.00 | 81.22 | 8.68 | += | 100.0 | 76.83 | 12.00 | = | 100.0 | 83.22 | 11.18 | += | 100.0 | 78.83 | 10.67 | += |
| PAN | All | – | | | | 50.84 | | | | – | | | | 49.15 | | | | 52.54 | | | |
| | $GPWFC_1$ | 67.79 | 58.98 | 5.42 | | 55.93 | 52.15 | 1.87 | + | 69.49 | 56.44 | 6.39 | | 71.18 | 57.68 | 4.72 | + | 59.32 | 51.42 | 5.74 | = |
| | $GPWFC_2$ | 71.18 | 58.85 | 5.67 | = | 59.32 | 53.67 | 1.84 | +⋆ | 71.18 | 55.56 | 6.57 | = | 72.88 | 56.67 | 5.24 | += | 64.40 | 53.16 | 5.35 | =⋆ |
| HCC | All | – | | | | 48.18 | | | | – | | | | 51.02 | | | | 53.06 | | | |
| | $GPWFC_1$ | 81.63 | 71.97 | 8.20 | | 73.46 | 63.40 | 5.05 | + | 83.67 | 71.08 | 6.37 | | 77.55 | 67.68 | 6.21 | + | 77.55 | 63.87 | 5.55 | + |
| | $GPWFC_2$ | 83.67 | 73.33 | 4.85 | = | 75.51 | 63.13 | 6.65 | += | 85.71 | 71.97 | 7.03 | = | 79.59 | 67.75 | 6.4 | += | 79.59 | 65.00 | 6.99 | += |
| DGB | All | – | | | | 43.90 | | | | – | | | | 65.85 | | | | 48.78 | | | |
| | $GPWFC_1$ | 72.50 | 60.75 | 6.95 | | 57.50 | 47.33 | 5.12 | + | 72.50 | 53.58 | 7.24 | | 75.00 | 59.33 | 8.20 | - | 75.00 | 60.67 | 8.95 | + |
| | $GPWFC_2$ | 75.00 | 62.50 | 8.86 | = | 57.50 | 47.83 | 4.63 | += | 72.50 | 54.92 | 7.78 | = | 72.50 | 62.16 | 6.49 | == | 77.50 | 56.00 | 7.76 | += |

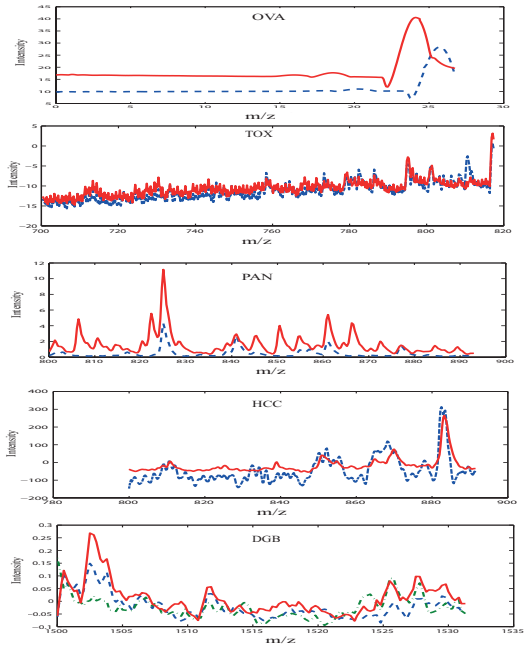


Fig. 2: Visualisation of the $GPWFC_2$ detected biomarkers for the five data sets.

group. In the DGB data set, the green dash-dotted line is an example from the chronic liver group.

It can be noticed that the $GPWFC_2$ selects features with different levels of intensity between the different classes. This means that the proposed method succeeds in detecting the biomarkers that have the potential to discriminate between the different groups. It can be noticed that the detected peaks are mostly higher in the healthy group. The first part of Figure 2, which is an example of the detected biomarkers for the OVA data set, shows that the selected peaks are significantly different between the $m/z$ values ranging from 0.002221 to 27.026. The accuracy of test set (RF classifier) using the evolved features from these original features is 92.35%. The number of constructed features from this program is 291, whilst the number of selected features is 556. This means the constructed features are less than the selected ones by 47.66%. For the rest of the data sets, the spectrum seems to be quite different in the levels of intensity. The $m/z$ range of the biomarkers in TOX data set is from 700.62 to 819.32. The detected features are 470, while the constructed features are 229. For the PAN and HCC data sets the ranges' of the $m/z$ ratios detected are from 800.0 to 893.45 and 799.73 to 893.23, respectively. The number of selected features from

TABLE V: The Percentage of overlap of the biomarkers between $GPWFC_1$ and $GPWFC_2$.

| | OVA | TOX | PAN | HCC | DGB |
|---|---|---|---|---|---|
| Overlap (%) | 81.29 | 20.95 | 27.87 | 35.77 | 41.77 |

those evolved programs in PAN and HCC are 227 and 368, respectively, whereas the number of constructed features are 193 and 177, respectively. In the DGB data set, which has three groups, the two patient groups (blue dashed line and green dashed-dotted line) are near to each other than with the healthy group. This mainly because the two patients groups have more between-class distance and it is more difficult to differentiate them The $m/z$ values range detected is from 1500 to 1537. The difference between the number selected and constructed features is 69. In summary, $GPWFC_2$ is capable of detecting the biomarkers and at the same time reduces the number of those biomarkers through feature construction.

Notice that, there is an overlap between $GPWFC_1$'s and $GPWFC_2$'s selected features which is shown in Table V. This overlap indicates that the common features can be trusted as true biomarkers of the data sets. This suggests that the important biomarkers can be detected by both GPWFC methods, and those important biomarkers can be used more reliably for classification.

## VI. Conclusions and Future Work

The main goal of this paper was to develop two new GP methods for construction of multiple feature. Specifically, the objective was to make use of the good performance of the wrapper approach for feature construction and biomarker detection and also ensure the generalisation with multiple classifiers. The goal was successfully achieved by developing two new GP approaches ($GPWFC_1$ and $GPWFC_2$). The fitness function used for $GPWFC_1$ is simply the classification accuracy of RF, which can be normally biased to RF. Therefore, for $GPWFC_2$ a new fitness function is proposed. $GPWFC_2$ is maximising the entropy gain of RF classifier and the distance between the classes and minimising the p-value of the features. The two $GPWFC$s are applied to five different MS data sets and compared to using all the original features. $GPWFC_1$ and $GPWFC_2$ achieved significant improvement over using all the original features. On the one hand, the number of features constructed are much smaller than the original set. On the other hand, the classification performance is significantly better. This suggests that GP succeeds in finding a better amount of information from the original set by constructing the new high-level features. Moreover, $GPWFC_2$ is better than $GPWFC_1$ in terms of the generalisation ability and number of

features. The results also show that GP selects features which are the biomarkers of the data sets.

Our future works include the following: we will investigate the arithmetic simplification of the evolved GP tree to reduce the number of constructed features and also the computational time. Another future direction is to extend GPWFC to perform classification, in addition to feature construction, for both binary and multi-class classification of MS data. This will help in reducing the computational cost by making GP as the classifier, and, therefore, overcome the disadvantage of the high cost of the wrapper approach.

## REFERENCES


[1] S. Datta, "Feature Selection and Machine Learning with Mass Spectrometry Data," in *Mass Spectrometry Data Analysis in Proteomics*, R. Matthiesen, Ed. Humana Press, 2013, vol. 1007, pp. 237–262.

[2] R. Armaanzas, Y. Saeys, I. Inza, M. Garcia-Torres, C. Bielza, Y. van de Peer, and P. Larranaga, "Peakbin selection in mass spectrometry data using a consensus approach with estimation of distribution algorithms," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 8, no. 3, pp. 760–774, 2011.

[3] B.-L. Adam, Y. Qu, J. W. Davis, M. D. Ward, M. A. Clements, L. H. Cazares, O. J. Semmes, P. F. Schellhammer, Y. Yasui, Z. Feng, and G. L. Wright, "Serum protein fingerprinting coupled with a pattern-matching algorithm distinguishes prostate cancer from benign prostate hyperplasia and healthy men," *Cancer Research*, vol. 62, no. 13, pp. 3609–3614, 2002.

[4] S. Luke, *Essentials of Metaheuristics*, 2nd ed. Lulu, 2013, http://cs.gmu.edu/∼sean/book/metaheuristics/.

[5] K. Neshatian, M. Zhang, and P. Andreae, "A filter approach to multiple feature construction for symbolic learning classifiers using genetic programming," *IEEE Transactions of Evolutionary Computation*, vol. 16, no. 5, pp. 645–661, 2012.

[6] J. R. Koza, "Introduction to genetic programming: tutorial," in *Genetic and Evolutionary Computation Conference,GECCO (Companion)*, 2008, pp. 2299–2338.

[7] W. Banzhaf, F. D. Francone, R. E. Keller, and P. Nordin, *Genetic Programming: An Introduction: on the Automatic Evolution of Computer Programs and Its Applications*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998.

[8] J. Miller, "Cartesian Genetic Programming," in *Cartesian Genetic Programming*, ser. Natural Computing Series, J. F. Miller, Ed. Springer Berlin Heidelberg, 2011, pp. 17–34.

[9] R. McKay, N. Hoai, P. Whigham, Y. Shan, and M. O'Neill, "Grammar-based Genetic Programming: a survey," *Genetic Programming and Evolvable Machines*, vol. 11, no. 3-4, pp. 365–396, 2010.

[10] K. Neshatian, M. Zhang, and M. Johnston, "Feature construction and dimension reduction using genetic programming," in *Proceeding of 20th Australian Conference on Artificial Intelligence*, 2007, pp. 160–170.

[11] H. Firpi, E. Goodman, and J. Echauz, "On Prediction of Epileptic Seizures by Computing Multiple Genetic Programming Artificial Features," in *Genetic Programming*, ser. Lecture Notes in Computer Science, M. Keijzer, A. Tettamanzi, P. Collet, J. Hemert, and M. Tomassini, Eds. Springer Berlin Heidelberg, 2005, vol. 3447, pp. 321–330.

[12] M. Smith and L. Bull, "Feature Construction and Selection Using Genetic Programming and a Genetic Algorithm," in *Genetic Programming*, ser. Lecture Notes in Computer Science, C. Ryan, T. Soule, M. Keijzer, E. Tsang, R. Poli, and E. Costa, Eds. Springer Berlin Heidelberg, 2003, vol. 2610, pp. 229–237.

[13] K. Krawiec and B. Bhanu, "Visual learning by coevolutionary feature synthesis," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 35, no. 3, pp. 409–425, 2005.

[14] W. Fu, M. Johnston, and M. Zhang", "Genetic programming for automatic construction of variant features in edge detection," in *Proceedings of the 16th European Conference on the Applications of Evolutionary Computation, EvoApplications*, 2013, pp. 354–364.

[15] W. Fu, M. Johnston, and M. Zhang, "Automatic construction of invariant features using genetic programming for edge detection," in *Proceedings of 25th Australasian Conference on Artificial Intelligence*, 2012, pp. 144–155.

[16] M. Muharram and G. D. Smith, "Evolutionary Constructive Induction," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 11, pp. 1518–1528, 2005.

[17] K. Krawiec, "Genetic programming-based construction of features for machine learning and knowledge discovery tasks," *Genetic Programming and Evolvable Machines*, vol. 3, no. 4, pp. 329–343, 2002.

[18] H. Guo and A. Nandi, "Breast cancer diagnosis using genetic programming generated feature," in *2005 IEEE Workshop on Machine Learning for Signal Processing*, 2005, pp. 215–220.

[19] G. Chen, T. G. Gharib, C.-C. Huang, D. G. Thomas, K. A. Shedden, J. M. G. Taylor, S. L. R. Kardia, D. E. Misek, T. J. Giordano, M. D. Iannettoni, M. B. Orringer, S. M. Hanash, and D. G. Beer, "Proteomic analysis of lung adenocarcinoma: Identification of a highly expressed set of proteins in tumors," *Clinical Cancer Research*, vol. 8, no. 7, pp. 2298–2305, 2002.

[20] Petricoin, A. M. Ardekani, B. A. Hitt, P. J. Levine, V. A. Fusaro, S. M. Steinberg, G. B. Mills, C. Simone, D. A. Fishman, E. C. Kohn, and L. A. Liotta, "Use of proteomic patterns in serum to identify ovarian cancer," *The Lancet*, vol. 359, no. 9306, pp. 572–577, 2002.

[21] S. Ahmed, M. Zhang, and L. Peng, "Genetic Programming for Biomarker Detection in Mass Spectrometry Data," in *Proceeding of the 25 th Australasian Conference on Artificial Intelligence*, 2012, pp. 266–278.

[22] S. Ahmed, M. Zhang and L. Peng, "Feature Selection and Classification of High Dimensional Mass Spectrometry Data: A Genetic Programming Approach," in *Proceedings of the 11th Europian Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics (EvoBio)*, 2013, vol. 7833, pp. 43–55.

[23] S. Ahmed, M. Zhang, and L. Peng, "Enhanced feature selection for biomarker discovery in lc-ms data using GP," in *Proceedings of 2013 IEEE Congress on Evolutionary Computation*, 2013, pp. 584–591.

[24] X. Guan, M. Chance, and J. Barnholtz-Sloan, "Splitting random forest (SRF) for determining compact sets of genes that distinguish between cancer subtypes," *Journal of Clinical Bioinformatics*, no. 1, pp. 1–13, 2012.

[25] P. Yang, W. Liu, B. Zhou, S. Chawla, and A. Zomaya, "Ensemble-based wrapper methods for feature selection and class imbalance learning," in *Advances in Knowledge Discovery and Data Mining*, ser. Lecture Notes in Computer Science, J. Pei, V. Tseng, L. Cao, H. Motoda, and G. Xu, Eds. Springer Berlin Heidelberg, vol. 7818, pp. 544–555. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-37453-1_45

[26] L. Trigg, *An Entropy Gain Measure of Numeric Prediction Performance*, ser. Working paper series (University of Waikato. Dept. of Computer Science), 1998.

[27] E. F. Petricoin, V. Rajapaske, E. H. Herman, A. M. Arekani, S. Ross, D. Johann, A. Knapton, J. Zhang, B. A. Hitt, T. P. Conrads, T. D. Veenstra, L. A. Liotta, and F. D. Sistare, "Toxicoproteomics: Serum proteomic pattern diagnostics for early detection of drug induced cardiac toxicities and cardioprotection," *Toxicologic Pathology*, pp. 122–130, 2004.

[28] S. R. Hingorani, E. F. P. III, A. Maitra, V. Rajapakse, C. King, M. A. Jacobetz, S. Ross, T. P. Conrads, T. D. Veenstra, B. A. Hitt, Y. Kawaguchi, D. Johann, L. A. Liotta, H. C. Crawford, M. E. Putt, T. Jacks, C. V. Wright, R. H. Hruban, A. M. Lowy, and D. A. Tuveson, "Preinvasive and invasive ductal pancreatic cancer and its early detection in the mouse," *Cancer Cell*, vol. 4, no. 6, pp. 437–450, 2003.

[29] H. Ressom, R. S. Varghese, E. Orvisky, S. Drake, G. Hortin, M. Abdel-Hamid, C. A. Loffredo, and R. Goldman, "Ant colony optimization for biomarker identification from maldi-tof mass spectra," in *Proceedings ofthe 28th IEEE Annual International Conference in Engineering in Medicine and Biology Society*, 2006, pp. 4560–4563.

[30] MATLAB, *version 7.10.0 (R2010a)*. Natick, Massachusetts: The MathWorks Inc., 2010.

[31] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *SIGKDD Explorer Newsletter*, pp. 10–18, 2009.