Genetic Algorithm Based Feature Selection for Mass Spectrometry Data

Yifeng Li, Yihui Liu, Li Bai

***Abstract*—Mass spectrometry technique is a revolutionary tool for diagnosing early stage cancer by analyzing protein mass spectra, and for detecting biomarkers. But because of the high dimensionality of the data, feature selection is a necessary procedure before classification and analysis. In this paper we present a genetic algorithm for feature selection for prostate protein mass spectrometry data. An elitism coupled with rank based stochastic universal sampling selection strategy, uniform crossover operation, and a uniform mutation with adaptive mutation rate strategy are used. Two fitness functions are defined for the genetic algorithm: one is a multivariate filter measurement and the other is a wrapper measurement. Our experiments show that the wrapper-based genetic algorithm outperforms all the other feature selection methods presented here. The multivariate filter-based genetic algorithm also yields better performance than transformed methods, sequential selection methods, and univariate filter methods.**

I. Introduction

MASS spectrometry (MS) technique is a powerful tool to research proteomics. It can be applied to high-throughput biomarker identification and disease diagnosis. It can also assist medical experts to explore the pathology in addition to clinical observation. The mass spectra obtained by mass spectrometer can be depicted as a histogram with mass-to-charge ratio (M/Z) on horizontal axis, and ion intensity on vertical axis. The M/Z ratio is called feature in this paper, and the ion intensity is called feature value of corresponding M/Z ratio. The high-dimensionality and small sample size of the data pose a challenge its analysis, in despite of its promising applications. The protein mass spectra contains abundant useful and complex information, we should mine the specific information relating to the specific case and remove the redundant information. For the case of classification of cancer and healthy samples, several features are sufficient for it; thousands of features not only slow the classifiers seriously but also degrade the prediction accuracy. So feature selection is a necessary phase for sample classification and searching for biomarkers.

Feature selection aims to select a discriminative feature subset from the original features. This subset is the optimal with respect to an evaluation criterion, hence feature selection is an optimisation process. There are three categories of feature selection methods: filter, wrapper, and embedded methods. Filter methods are independent of the classification algorithm. Wrapper methods use classification accuracy as the evaluation criterion of feature subset under examination, so this technique depends on the classifier adopted. Embedded methods are also classifier dependent techniques, as the feature selection process is done within the classification algorithm itself. Additionally, filter methods are grouped into two classes in term with the number of selected features and the dependence among features. Univariate filter methods evaluate each feature separately ignoring the feature dependencies. Multivariate filter methods consider the feature dependencies and combination effects, and thereby can evaluate more than one feature each time.

Many feature selection methods have been used in the domain of protein mass spectrometry [1]. For univariate filter methods, *t*-test [2], *F*-test [3], *P*-test [4], *KS*-test [4], Peak Probability Contrast [5], and etc. are used. For mutivariate filter methods, CFS [2] and Relief-F [6] are used. For wrapper methods, genetic algorithms (GAs) [7], nature inspired [8], SFS [4], SBS [4], and etc. are used. For embedded methods, Random forest/decision tree [9], weight vector of SVM [10], neural network [11], nearest shrunken centroid [4], boosting [4], etc. are used. Liu and Bai use wavelet detail [12] and wavelet approximation [13] to characterize the features of mass spectra. Significant biomarkers are then detected based on optimized features selected by genetic algorithm.

A good feature subset not only helps to improve the prediction accuracy but also aids in finding and analyzing the underlying significant biomarkers of the mass spectra. In this paper, we define the problem of feature selection for protein mass spectra and present a genetic algorithm to search optimal feature subset. We present an adaptive mutation operation, and build one multivariate filter method and one wrapper method. The category of a feature selection method is determined by the evaluation criterion for a feature subset. Univariate filter methods are widely used in the domain of mass spectrometry, but multivariate filter methods are seldom used because developing effective evaluation criteria is still a problem. In this paper we present an evaluation criterion for

Manuscript received July 5, 2008.

This work was supported by international collaboration project of Shandong Province Education Department, China and research funds of Shandong Institute of Light Industry (12041653).

Yifeng Li, IEEE Student Member, is with the Institute of Intelligent Information Processing, School of Information Science and Technology, Shandong Institute of Light Industry, Jinan, Shandong 250353 China (bolirenyifeng@yahoo.com.cn).

Yihui Liu is with the Institute of Intelligent Information Processing, School of Information Science and Technology, Shandong Institute of Light Industry, Jinan, Shandong 250353 China (corresponding author, phone/fax: 86-531-89631256/86-531-89631251; yihui_liu_2005@yahoo.co.uk).

Li Bai is with School of Computer Science, University of Nottingham, UK, NG8 1BB (bai@cs.nott.ac.uk).

the multivariate filter method. This criterion employs a combination of scatter matrices and Bhattacharyya distance. For wrapper method we build the evaluation function using combination of classification error rate and posterior probability. The evaluation criteria are the fitness function in GA. In order to achieve good generalization for small sample size we use the linear discriminant analysis classifier.

II. Theories

A. Linear Discriminant Analysis

In this study, we use the combination of the empirical error rate of linear discriminant analysis (LDA) classifier and the posterior probability of the classifier as the optimality criterion of the wrapper-based feature selection method. LDA is stemmed from R. A. Fisher's classical and pioneering paper [14]. The key idea of LDA is that it aims to obtain an optimal projection direction, under which the *m*-dimensional feature vectors of samples can be projected onto a lower dimensional subspace (generally one-dimensional space). In this new space, the distances of samples from different classes are enlarged whereas the distances of samples in the same class are shortened to the best.

LDA considers maximizing the following objective:

$$J(w) = \left(w^T S_B w\right) / \left(w^T S_W w\right) \tag{1}$$

where w denotes the projection direction represented by a column vector, w^T is the transpose of w, S_B and S_W stand for the between-class scatter matrix and the within-class scatter matrix, respectively. They are:

$$S_B = \sum_{i=1}^{c} \left[P(c_i)(\mu_i - \mu)(\mu_i - \mu)^T \right] \tag{2}$$

$$S_W = \sum_{i=1}^{c} \left\{ P(c_i) E\left[(x - \mu_i)(x - \mu_i)^T \mid c_i \right] \right\} \tag{3}$$

where $P(c_i)$ is the prior probability of class c_i, c is the number of classes, μ_i is the mean of class c_i, μ is the mean of all the data. For binary-class problem, the projection direction is

$$w^* = \arg\max_w \left[J(w) \right] = S_W^{-1}(\mu_1 - \mu_2). \tag{4}$$

It can be used as the weight vector of linear discriminant function

$$g(x) = \left(w^*\right)^T x + w_0 \tag{5}$$

where x is a sample and w_0 is known as threshold.

LDA classifier is computationally fast, and because the VC dimension of LDA classifier for *m*-dimension input is *m*+1, LDA classifier takes on good classification generalization on small-sized dataset.

B. Bhattacharyya Distance

In this paper, we employ Bhattacharyya distance [15] to build the class separability measurement of multivariate filter-based feature selection method. Bhattacharyya distance is a distance metric of two probability density functions. Bhattacharyya distance is defined as

$$J_B = -\ln \int \left[p(x \mid \omega_1) p(x \mid \omega_2) \right]^{1/2} dx \tag{6}$$

where $p(x \mid \omega_1)$ and $p(x \mid \omega_2)$ are class conditional probability density functions of class ω_1 and class ω_2, respectively. Bhattacharyya Distance have the following properties: $J_B \geq 0$; If $p(x \mid \omega_1) = p(x \mid \omega_2)$, $J_B = 0$; If $p(x \mid \omega_1)$ and $p(x \mid \omega_2)$ never overlap each other, then $J_B = MAX$.

For Gaussian distributions $N(\mu_1, \Sigma_1)$, $N(\mu_2, \Sigma_2)$

$$J_B = \frac{1}{8}(\mu_1 - \mu_2)^T \left[(\Sigma_1 + \Sigma_2)/2 \right]^{-1} (\mu_1 - \mu_2) + \frac{1}{2} \ln \frac{\left| (\Sigma_1 + \Sigma_2)/2 \right|}{\left[|\Sigma_1| |\Sigma_2| \right]^{1/2}} . \tag{7}$$

C. Scatter Matrices [15]

We use a simple measurement to evaluation the class separability of two classes. This measurement does not consider the distribution of data under investigation and just uses the between-class scatter matrix and the within-class matrix. It is defined as

$$J_S = trace(S_M) / trace(S_W) \tag{8}$$

where $S_M = S_B + S_W$. S_B, S_W are defined in (2) and (3), respectively.

D. Measurements for the Classification Performance

In this paper, we employ five measurements for representing the performance of classification:

$$\begin{aligned} &Accuracy = (TP+TN)/(TP+TN+FP+FN) \\ &Sensitivity = TP/(TP+FN) \\ &Specificity = TN/(TN+FP) \\ &Balanced\ Accuracy\ (BACC) = (Sensitivity + Specificity)/2 \\ &Positive\ Predictive\ Value\ (PPV) = TP/(TP+FP) \end{aligned} \tag{9}$$

where *TP*, *TN*, *FP*, and *FN* stand for the numbers of true positive (cancer), true negative (normal), false positive, and false negative samples, respectively.

III. Mathematical Modeling

A. Problem

Each sample is represented as a feature vector of high dimensionality, of which each component is a signal intensity value (feature value) of the ions corresponding to an M/Z ratio (feature). Assume each original sample is a point of *n*-dimension feature space. Our task is to develop and run GA to search an *m*-sized class-discriminative feature subset from the *n* features. The samples after feature selection are points in the *m*-dimension subspace. And the subsequent classification and analysis are performed on this *m*-dimension subspace.

B. Encoding and Decoding

An individual of a population represents a feature subset which can be encoded into an integer valued vector of length m. Each component of the vector is a gene. Each gene is an index of the original feature set. The decoding is easy. In term of the indices, we can reconstruct a feature subset for the best individual.

C. Fitness Functions

The fitness function is related to the classification accuracy. We employ a combined multivariate filter method and a wrapper method both with GA as the search engine to select discriminative feature subsets. The filter-based measurement is a passive method, as class separability measurements are used instead of classification accuracy. We use a combination of scatter matrices and Bhattacharyya distance to define the fitness function. As we view the GA as a minimization approach, we create the fitness function as follow:

$$f_{SB}(x) = 1/J_S + t_0 - J_B \tag{10}$$

where J_S and J_B are defined in (8) and (7), respectively, t_0 is a bias, and x is an individual representing a feature subset.

A reasonable way to evaluate the classification ability of an individual is to use the classification accuracy as a part of a fitness function. In this paper, a linear combination of the empirical error of the LDA classifier and the a-posteriori probability is employed to estimate the quality of the feature subset under examination. Assume that two subsets s_1 and s_2 yield the same empirical classification error rate, but s_1 obtains a-posteriori probability of p_1, and s_2 gets p_2. If $p_1 > p_2$, then we favor s_1 as the fitter individual. This fitness function is

$$f_{EP}(x) = 100e_c + e_p \tag{11}$$

where e_c is the empirical classification error rate and e_p is the defined as

$$e_p = 1 - \frac{1}{n_{train}} \left\{ \sum_{i=1}^{n_{train}} \max \left[P(c_1 \mid x_i), \Lambda \, , P(c_c \mid x_i) \right] \right\} \tag{12}$$

where n_{train} is the number of training samples and P ($c_j|x_i$) is the posteriori probability of sample x_i belonging to class c_j.

D. Selection and Reproduction

We apply an elitism coupled with rank based stochastic universal sampling selection strategy.

Fitness-proportional selection does not guarantee the selection of any particular individual, including the fittest. The best-so-far individual may not be selected to survive in the next generation, which make the outstanding genes thrown away, therefore the individual corresponding to the high-quality solution, even the global optimum solution, can not exist in the final population. In order to remove the effort of the above, the top several fittest elites in the current population are copied to the next generation without being disrupted by crossover and mutation, which is called elitism. Then the rest participating in genetic operations are chosen by the rank based stochastic universal sampling selection strategy.

If GAs use raw fitness obtained by the fitness function to perform fitness-proportional selection and the fitnesses differ greatly, some super-fit individuals can predominate in the population after several generations as they are more likely to survive, while the other individuals may be wiped out in spite of having the potential to be the optimal solution. In order to avoid trapping in the suboptimal point and encourage exploration in a wider space, we use rank [16] strategy to scale the fitness of each individual based on their rank before conducting the selection of the individuals for participating in crossover and mutation. The rank of an individual is its position in the sorted raw fitnesses of all the individuals in a population. The fittest individuals have scaled fitness 1, the second fittest 2 etc, and the worst have scaled fitness N, N is the population size. Stochastic universal sampling is then conducted based on the scaled fitness. It allocates parent individuals using a roulette wheel with N slots sized according to expectation which is defined as

$$e(i) = \frac{1/\sqrt{fitness_{scaled}(i)}}{\sum_{j=1}^{N} 1/\sqrt{fitness_{scaled}(j)}} \cdot N_{parents} \tag{13}$$

where $e(i)$ is the expectation of the ith individual, $fitness_{scaled}(i)$ is the scaled fitness of the ith individual based on rank, N is the population size, and $N_{parents}$ is the numbers of parent individuals selected to do crossover and mutation. There are $N_{parents}$ equal-sized pointers above the wheel. When sampling, these pointers are spun only once to allocate each parent from the slots the pointers land on. This stochastic universal sampling strategy is first introduced by Baker [17], and exhibits no bias and minimal spread.

E. Uniform Crossover

In this paper, uniform crossover is used to recombine the genes of parent chromosomes. Unlike one-point crossover and two-point crossover, uniform crossover is one child version genetic operation which produces one offspring given each pair of parents, and each gene of the offspring is randomly selection from the corresponding genes of its parents. The parameter crossover fraction R_c, population size N and elites count N_e determine the number of offspring produced by crossover operation and the number of their parents in the mating pool. They are $N_c = R_c(N - N_e)$ and $2\ N_c$ respectively.

F. Uniform Mutation with Adaptive Mutation Rate

The goal of mutation is to increase exploration. One parent is altered to form one offspring through mutation operation. The number of parents selected for mutation is $N - N_e - N_c$. Uniform mutation model is the simplest and commonly used in GA. For integer valued vector encoding, this method replaces each component in a uniform low probability that is called mutation rate represented as R_m. For many instance of

GA, R_m is assumed as a constant. But in the natural world, the mutation rate is not always constant [18]. Some fine mutation rate tuning methods have been presented in [19]-[21]. We use a uniform mutation with adaptive mutation rate strategy, which changes the mutation parents in each generation, and adjust the mutation rate once for each five generation.

$$R_m = \begin{cases} R_{ms} & if\ G_c \bmod \Delta = 0 \quad (G_c \neq 0, G_c \neq G_{\max}) \\ R_{mc} & else \end{cases} \quad (14)$$

where R_{mc} is a very small constant value, G_c is the current generational counter, $G_{\max}$ is the maximum number of generations, and Δ is an interval. Assume R_{mc}=0.02, $\Delta = 5$, the mutation rate is R_{ms} for generation 5, 10, 15,…, otherwise is R_{mc}, namely 0.02, for the other generations. R_{ms} is defined in the followings.

This strategy automatically tunes R_{ms} based on the population diversity or similarity. The similarity of chromosome *i* and chromosome *j* is defined as

$$s_{ij} = 2 - \frac{L_{ij}}{m} \quad (15)$$

where *m* is the number of genes of a chromosome, L_{ij} is the number of different genes in chromosome *i* and chromosome *j*. For example, if *i*=(20, 257, 698, 710), and *j*=(45, 332, -419, 698), then L_{ij}= 7. The domain of s_{ij} is $0 \le s_{ij} \le 1$. The similarity of chromosome *i* to the population is defined as

$$s_i = \sum_{j=1}^{N} s_{ij} \ . \quad (16)$$

The scope of s_i is $1 \le s_i \le N$. The similarity of a population is defined as

$$s = \sum_{j=1}^{N} s_j \quad (17)$$

The range of *s* is $N \le s \le N^2$. The mutation rate is defined as

$$R_{ms} = \frac{R_{\max} - R_{\min}}{N^4 - N^2} s^2 + \frac{R_{\min} N^2 - R_{\max}}{N^2 - 1} \quad (18)$$

where $R_{\max}$ and $R_{\min}$ are the man-defined upper bound and lower bound, respectively. Generally we designate $R_{\max}$=0.5, $R_{\min}$=0.02. R_{ms} is a monotone increasing function for *s*. If the similarity of the last population is high, then the mutation rate R_{ms} is increased in the current generation, so the diversity of the next generation is increased, and so the similarity of the next generation is decreased. Higher diversity means more search potential. Hence, this adaptive mutation rate model tends to keep the population diversity and encourages GA to explore more potential areas. In the intervals when R_m= R_{mc}, GA can sufficiently search these areas exploited by R_{ms}.

G. Population Size

Large population size can improve the search capability of GA. However, the larger the population size, the longer the genetic algorithm takes to compute each generation. So we make the number of the overall genes in a population approximately equal to the number of the original candidate features by compromise.

H. Termination Criteria

The simplest and widely used criterion is that if the current generation counter meets the maximum number generation, GA will stop after the last genetic operations. Additionally, we use an adaptive criterion to halt the evolution when the weighted average change in a sliding window is less than a predefined threshold constant.

$$wavg = \sum_{i=1}^{i=w} \left(\frac{1}{2}\right)^{w-i} \frac{\left|best(i+1) - best(i)\right|}{best(i)+1} \quad (19)$$

where *w* is the width of the window, *best*(*w*+1) is the fittest chromosome of the current generation, *best*(1) is the fittest chromosome of the farthest generation among the window to the current generation. The biased weights are used because the changes near the current generation are more important than those far from the current generation.

IV. Experiments, Results, and Analysis

We used the GA on a prostate protein mass spectra dataset acquired from FDA-NCI Clinical Proteomics Program Databank. The spectra were collected utilizing the H4 protein chip, prepared manually using the recommended protocol, and a Ciphergen PBS1 SELDI-TOF mass spectrometer. This dataset is named PC-H4 dataset in [4] and [22]. PC-H4 contains 322 total samples. Each sample is composed of 15154 feature values corresponding to 15154 features. 26 samples with prostate cancer with prostate-specific antigen (PSA) levels 4-10 and 43 samples with prostate cancer with PSA levels greater than 10 were combined into cancer class, while 190 samples with benign prostate hyperplasia with PSA levels greater than 4 and 63 samples with no evidence of disease with PSA level less than 1 into normal class.

We divided the experiment into two parts: feature selection and classification, as shown in Fig. 1. We first run GA to select discriminative 20-sized features subsets. For multivariate filter methods, we used (10) to define the fitness functions. For wrapper method, we employed (11) to evaluate the fitness of chromosomes. In each generation, elitism selection is performed to copy top two of the fittest chromosome of the current generation to the next without change, then parents for crossover and mutation are selected by the rank based stochastic universal sampling selection strategy. Each pair of parents participating in uniform crossover produce one child for the next generation. We use uniform mutation operation with adaptive mutation rate strategy to alter the parents selected for mutation.

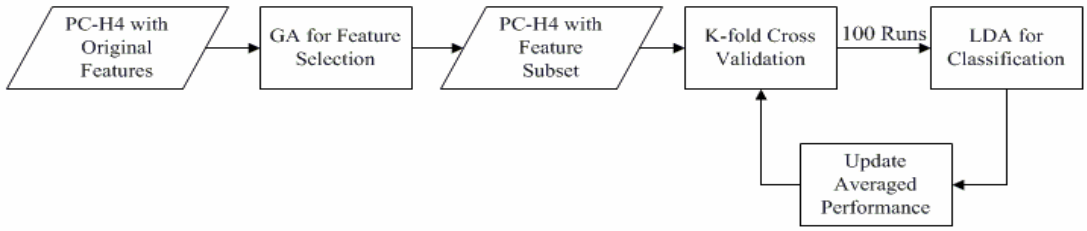

Fig. 1. The procedures of feature selection and classification for mass spectra.

TABLE I

THE PERFORMANCES OF CLASSIFICATION FOR 20-SIZED FEATURE SUBSETS SELECTED BY GA AND THE OTHER FEATURE SELECTIONS EVER USED ON PC-H4. STD STANDS FOR STANDARD DEVIATION.

	Accuracy(*std*)	Sensitivity(*std*)	Specificity(*std*)	PPV(*std*)	BACC(*std*)
Filter GA	0.9270(0.0174)	0.8792(0.0287)	0.94(0.019)	0.8025(0.0492)	0.9096(0.0189)
Wrapper GA	0.9775(0.0084)	0.9616(0.0225)	0.9819(0.0063)	0.9355(0.0218)	0.9717(0.0131)
PCA [4]	0.530 (0.20)	0.493 (0.21)	0.540 (0.24)	0.248 (0.11)	0.516 (0.18)
PCA/LDA [4]	0.692 (0.15)	0.623 (0.33)	0.710 (0.22)	0.431 (0.17)	0.667 (0.14)
SFS [4]	0.885 (0.05)	0.725 (0.36)	0.929 (0.03)	0.728 (0.03)	0.827 (0.17)
SBS [4]	0.773 (0.03)	0.652 (0.27)	0.806 (0.11)	0.498 (0.07)	0.729 (0.09)
P-test [4]	0.813 (0.02)	0.580 (0.28)	0.877 (0.08)	0.572 (0.07)	0.728 (0.11)
T-test [4]	0.816 (0.04)	0.522 (0.31)	0.897 (0.05)	0.575 (0.07)	0.709 (0.14)
KS-test [4]	0.826(0.04)	0.710 (0.35)	0.857 (0.08)	0.579 (0.05)	0.784 (0.14)
NSC(20) [4]	0.850 (0.06)	0.638 (0.31)	0.833 (0.12)	0.529 (0.07)	0.736 (0.10)
BoostedFE [4]	0.960 (0.01)	0.812 (0.07)	1.000 (0.00)	1.000 (0.00)	0.906 (0.03)

TABLE II

SOME FREQUENTLY EMERGED FEATURES AND THEIR NEIGHBORS (*N*(*)) FOR FILTER-BASED GA 20 RUNS. *F* IS THE NUMBER OF TIMES THE FEATURES EMERGE IN THE 20 FEATURE SUBSETS.

N(65.4754)	*f*	*N*(68.0681)	*f*	*N*(81.1128)	*f*	*N*(125.6354)	*f*	*N*(501.2661)	*f*	*N*(6909.6737)	*f*
65.3244	4	67.1473	1	81.1128	10	125.0085	4	500.0132	1	6908.1223	2
65.4754	5	67.4535	1	81.4493	2	125.2173	14	500.8483	2	6909.6737	2
66.3847	3	67.9142	6	81.9555	1	125.4262	12	501.2661	6		
		68.0681	6			125.6354	14	502.5206	4		
		68.2221	1			125.8447	2	502.9391	1		
		68.9952	1			126.0541	3				
						126.2638	3				
						127.3146	5				
						127.5253	7				
						127.7362	1				

TABLE III

SOME FREQUENTLY EMERGED FEATURES AND ITS NEIGHBORS (*N*(*)) FOR WRAPPER-BASED GA 20 RUNS. *F* IS THE NUMBER OF TIMES THE FEATURES EMERGE IN THE 20 FEATURE SUBSETS.

N(125.6354)	*f*	*N*(362.8249)	*f*	*N*(478.9542)	*f*	*N*(501.2661)	*f*	*N*(4013.4641)	*f*	*N*(6132.2857)	*f*
125.0085	2	362.4694	1	475.6919	2	497.096	1	4090.3226	1	6132.2857	3
125.2173	1	362.8249	6	478.5458	3	497.5122	2	4092.7104	3		
125.4262	1			478.9542	5	497.9286	3	4096.2934	3		
125.6354	12			479.3628	2	498.3452	1	4098.6829	1		
126.0541	4					500.4307	1	4099.8779	1		
126.2638	2					500.8483	3	4101.0731	3		
127.3146	6					501.2661	15	4102.2685	1		
127.5253	2					501.6841	1	4103.4641	4		
128.1584	3					502.1022	1	4104.6598	2		
						502.5206	1	4105.8557	1		
						502.9391	1				

TABLE IV

THE ACCURACIES OF SOME DISCRIMINATIVE FEATURES USING LDA CLASSIFIER

Feature	68.0681	81.1128	125.6354	362.8249	478.9542	501.2661	4013.4641	6132.2857	6909.6737
Accuracy	0.7789	0.7762	0.8315	0.6509	0.6416	0.8547	0.6076	0.6590	0.6162

After feature selection, 3-fold cross validation is executed 100 runs for LDA classifier. For each run, LDA classifier is performed 3 times for different training and testing splits. The averaged classification performance over the 100 runs is used to represent the performance of the feature subset selected by GA. To evaluate the performance of GA for feature selection more impartially we rerun the above two steps, as depicted in Fig. 1, for 20 times and show the result in the top two rows of Table I. The performance of GA for feature selection is determined by the fitness measurement. The wrapper-based GA is better than the multivariate filter-based GA for feature selection on PC-H4 dataset.

Other methods and their performances using 3-fold cross validation on PC-H4 listed in Table I are from [4], we find

that transform methods like PCA and PCA/LDA have poor performances. Furthermore these methods cannot harvest concrete features from the original features but transformed features, which are not competent for searching biomarkers. Sequential selection methods like SFS and SBS work better than transformed methods, but are still not effectively enough. Univariate filter methods can only achieve an accuracy of 0.800. Multivariate filter method as presented in this paper obtains an accuracy of 0.927 which is better than the transformed methods, sequential selection methods, and univariate filter methods. The performance of wrapper-based GA outperforms all of the other methods in accuracy, sensitivity, and BACC respects. GA achieves higher sensitivity than others. Table II and III listed some frequently emerged features and their neighbors. The accuracy of these features is shown in Table IV. We observed that 125.6354, 501.2661 are two significantly discriminative features.

V. Conclusion And Future Works

In this paper, we present a genetic algorithm for mass spectra feature selection. We use elitism section coupled with rank based stochastic universal sampling strategy, and a uniform mutation with adaptive mutation rate strategy. We develop a multivariate filter measurement and a wrapper measurement to build the fitness function. The multivariate filter-based GA achieves an accuracy of 0.9270, and wrapper-based GA 0.9775 which outperforms the other methods. We find two features frequently emerging in the subset selected by the GA: 125.6354 gives an accuracy of 0.8315, and 501.2661 yields 0.8547. In future research, we aim to develop more effective fitness functions for the GA, and introduce more intelligent search algorithms into feature selection for protein mass spectra. The discriminative features selected by these algorithms should be investigated in the context of proteomics.

References

[1] Y. Saeys, I. Inza, and P. Larranaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23(19), pp. 2507-2517, 2007.

[2] H. Liu, J. Li, and L. Wong, "A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns," *Genome Inform*., vol. 13, pp.51–60, 2002.

[3] G. Bhanot, G. Alexe, B. Venkataraghavan, and A. J. Levine, "A robust meta classification strategy for cancer detection from MS data," *Proteomics*, vol. 6(2), pp. 592–604, 2006.

[4] I. Levner. (2005, March). Feature selection and nearest centroid classification for protein mass spectrometry. *BMC Bioinformatics* [Online].6. Available: http://www.biomedcentral.com/1471-2105/6/68

[5] R. Tibshirani, T. Hastie, B. Narasimhan, S. Soltys, G. Shi, A. Koong, and Q. Le, "Sample classification from protein mass spectrometry, by 'peak probability contrast'," *Bioinformatics*, vol. 20, pp. 3034–3044, 2004.

[6] J. Prados, A. Kalousis, J. Sanchez, L. Allard, O. Carrette, and M. Hilario, "Mining mass-spectra for diagnosis and biomarker discovery of cerebral accidents," *Proteomics*, vol. 4, pp. 2320-2332, 2004.

[7] N. O. Jeffries. (2004, November). Performance of a genetic algorithm for mass spectrometry proteomics. *BMC Bioinformatics* [Online]. 5. Available: http://www.biomedcentral.com/1471-2105/5/180

[8] H. Ressom, R. S. Varghese, M. Abdel-Hamid, S. A. Eissa, D. Saha, L. Goldman, E. F. Petricoin, T. P. Conrads, T. D. Veenstra, C. A. Loffredo, and R. Goldman, "Analysis of mass spectral serum profiles for biomarker selection," *Bioinformatics*, vol. 21(21), pp. 4039-4045, Sep. 2005.

[9] B. Wu, T. Abbott, D. Fishman, W. McMurray, G. Mor, K. Stone, D. Ward, K. Williams, H. Zhao, "Comparison of statistical methods for classification of ovarian cancer using mass spectrometry data," *Bioinformatics*, vol. 19, pp. 1636-1643, 2003.

[10] K. Jong, E. Marchiori, M. Sebag, A. van der Vaart, "Feature selection in proteomic pattern data with support vector machines", in *Proceedings of the IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, La Jolla, 2004, pp. 41-48.

[11] G. Ball, S. Mian, F. Holding, R. O. Allibone, J. Lowe, S. Ali, G. Li, S. McCardle, I. O. Ellis, C. Creaser, and R. C. Rees, "An integrated approach utilizing artificial neural networks and SELDI mass spectrometry for the classification of human tumours and rapid identification of potential biomarkers," *Bioinformatics*, vol. 18(3), pp. 395-404, 2002.

[12] Y. Liu and L. Bai, "Find key m/z values in predication of mass spectrometry Cancer Data," in *ICIC 2008*, Shanghai, Sep. 2008, to be published.

[13] Y. Liu and L. Bai, "Salient information of mass spectra of prostate cancer dataset," in *IEEE Grc 2008*, Hangzhou, Aug. 2008, to be published.

[14] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, pp.179-188, 1936

[15] S. Theodoridis and K. Koutroumbas, *Pattern Recognition, Third Edition*, Academic Press, 2006, pp. 213-262.

[16] D. Whitley, "The GENITOR algorithm and selection pressure: why rank-based allocation of reproductive trials is best," in *ICGA3*, San Mateo, 1989, pp.116-121.

[17] J. E. Baker, "Reducing bias and inefficiency in the selection algorithm," in *Proceedings of the Second International Conference on Genetic Algorithms and their Application*, Hillsdale, 1987, pp. 14-21.

[18] G. C. Mills, "The molecular evolutionary clock: a critique," *Perspectives on Science and Christian Faith*, vol. 46, pp. 159-168, 1994.

[19] R. Tanse, "Distributed genetic algorithms", in *ICGA3,* San Mateo, 1989, pp. 434-439.

[20] L. Davis, "Adapting operator probabilities in genetic algorithms", in *ICGA3*, San Mateo, 1989, pp. 61-69.

[21] T. Back, "Optimal mutation rates in genetic search," in *ICGA5*, San Mateo, 1993, pp. 2-8.

[22] R. H. Lilien, H. Farid, and B. R. Donald, "Probabilistic disease classification of expression-dependent proteomic data from mass spectrometry of human serum," *Computational Biology*, vol. 10(6), pp. 925-946, Oct. 2003.