Diagnose the Premalignant Pancreatic Cancer Using High Dimensional Linear Machine

Yifeng Li and Alioune Ngom

School of Computer Sciences, 5115 Lambton Tower, University of Windsor,
401 Sunset Avenue, Windsor, Ontario, N9B 3P4, Canada
{li11112c,angom}@uwindsor.ca
http://cs.uwindsor.ca/uwinbio

Abstract. High throughput mass spectrometry technique has been extensively studied for the diagnosis of cancers. The detection of the pancreatic cancer at a very early stage is important to heal patients, but is very difficult due to biological and computational challenges. This paper proposes a simple classification approach which can be applied to the premalignant pancreatic cancer detection using mass spectrometry technique. Computational experiments show that our method outperforms the benchmark methods in accuracy and sensitivity without resorting to any biomarker selection, and the comparison with previous works shows that our method can obtain competitive performance.

Keywords: mass spectrometry, pancreatic cancer, classification, high dimensional linear machine.

1 Introduction

Proteomic mass spectrometry technique has great potential to be applied for clinical diagnosis and biomarker identification. The mass spectrometry data of a patient are obtained through measuring the ion intensities of tens of thousands of mass-to-charge (m/z) ratios of proteins and peptides. Analysis of such high throughput data is promising, but also difficult [1]. Some of the problems challenging the bioinformatics community include: 1) The data are quite noisy and subject to high variability. 2) Though the data are redundant, the amounts of useful and redundant information are not clear. 3) There are tens of thousands of features (m/z ratios), while there is only tens up to hundreds of samples, which is the well known large number of features versus small number of samples (LFSS) problem. This problem results in intolerable computational burden when using some prediction models, for example decision tree. Some models can not be applied on such data, because the number of their parameters grows exponentially as the number of dimensions increases, and therefore it is impossible to estimate these parameters using available training data. These problems are the notorious "curses of dimensionality" [2]. Due to the above problems, some models are easily subject to overfitting and hence have poor generalization. A lot of computational approaches dealing with the above problems have been proposed in two directions last decade. First of all, efforts of biomarker (and peak) identification and dimension reduction have been extensively taken for the clinical diagnosis, pathological, and computational purposes.

T. Shibuya et al. (Eds.): PRIB 2012, LNBI 7632, pp. 198–209, 2012.

As it is impossible to enumerate all works in this direction, we only give two highly cited examples in the following. Levner [3] tested many popular feature selection and feature extraction methods for biomarker identification coupled with nearest shrunken centroid classifier, and found that some state-of-art methods actually performs poorly on mass spectrometry data using consistent cross-validation. [4] is an excellent review on feature selection for mass spectrometry data. Second, kernel approaches have been invented [5]. These approaches are able to represent complex patterns and their optimization is dimension-free. Also, they are often robust to noise and redundant. Kernel approaches often have good capability generalization.

Patients with pancreatic cancer has a very high death rate. If the pancreatic cancer can be detected before the cancer develops, treated patients at the preinvasive stage can have a chance to survive. Unfortunately, there is no effective premalignant pancreatic cancer detection method by now [6]. In the precancerous stage, the proteins may have been developed differential signals. Proteomic mass spectrometry technique provides an insight into patient's protein profile, and therefore is quite promising to be applied to this area. Ge *et al.* [7] presented a framework of using ensemble of decision trees coupled with feature selection methods. Decision tree is very slow when learning on high dimensional data. Thus, in order to use decision tree as classifier, three feature selection methods (Student t-test, Wilcoxon rank sum test, and genetic algorithm) are used to reduce the dimension before classification. The performances of decision tree and its different ensembles were investigated in [7]. It was claimed that classifier ensembles generally have better prediction accuracy than single decision tree. However, most of the methods used in [7] still have low accuracies and low sensitivity. Another issue is that the candidate biomarkers selected by different methods are not consistent.

As we mentioned above, the curses of dimensionality actually imply, to a great extent, the difficulty of model selection due to LFSS in practice. Also the statement that"only very few features of mass spectrometry data are informative and the rest are redundant" is only an assumption in many circumstances. On another hand, the large number of features provides us with huge amount of information. Although the target informative knowledge hides in the data, we should have a chance of taking advantage of this using some data mining techniques. We can call this as one of the less-known "blessings of dimensionality" [8]. Although this principle has not yet been well understood theoretically, some studies based on this principle in computer vision have demonstrated prodigious results [9] [10]. In the high dimensional setting, real-world data points usually reside in manifolds. *Support vector machine* (SVM) [11] can be viewed as an example of taking advantage of high dimensionality. Its essential idea is that data points are mapped from the original low dimensional space to a very high (even infinite) dimensional space where the data points are likely to be linearly separable, and therefore a separating hyperplane could be implicitly learned through margin maximization.

In this paper, we shall prove that, in the case of LFSS, the mass spectrometry data are much likely to be linearly separable and we propose a high dimensional linear machine for such case to detect the premalignant pancreatic cancer at an early stage. The contributions of this study include

1. we bring the principle of blessings of dimensionality to the horizon of researchers in mass spectrometry data analysis;
2. we propose the high dimensional linear machine and show that it is a specific case of the general linear models for classification;
3. we propose a threshold adjustment method based on receiver operation curve.

The paper is organized as follows. In the next section, we first prove the linearity of the mass spectrometry in the case of LFSS, under some condition, and then describe our proposed method. The computational experiments and comparison results are then shown. After that related discussion are delivered. Finally, the paper is completed by some conclusions.

2 Methods

Suppose $\boldsymbol{D}_{m\times n}$ is a training set with m features (m/z ratios) and n samples. These samples are from two groups: the premalignant pancreatic cancer group (denoted by +1) and the normal group (denoted by -1). The class labels of these n training samples are in the column vector $\boldsymbol{c}$, and matrix $\boldsymbol{S}_{m\times p}$ represents p unknown samples. Each of these p samples is either from premalignant pancreatic cancer class or normal class. The computational task is to predict the class labels of these p samples.

Linear models for classification, such as *linear Bayesian classifier*, *Fisher discriminative analysis* (FDA), and the state-of-art SVM, try to find a hyperplane between two groups of the training set. This hyperplane can be formulated as

$$g(x) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} = 0, \tag{1}$$

where $\boldsymbol{w}, \boldsymbol{x} \in \mathbb{R}^{m+1}$. w_0 is the bias and the corresponding $x_0 = 1$. $\boldsymbol{w}$ and $\boldsymbol{x}$ in this form are thus *augmented*.

In our case, the hyperplane should separate the two classes in $\boldsymbol{D}$, that is

$$\boldsymbol{w}^{\mathrm{T}}[\mathbf{1}; \boldsymbol{D}] = \boldsymbol{c}^{\mathrm{T}}, \tag{2}$$

where the boldface $\mathbf{1}$ is a column vector accommodating n ones. $[\mathbf{1}; \boldsymbol{D}]$ uses MATLAB notation meaning the concatenation of $\mathbf{1}$ and $\boldsymbol{D}$ in row-wise direction. Using matrix transposition, we have

$$\boldsymbol{A}^{\mathrm{T}}\boldsymbol{w} = \boldsymbol{c}, \tag{3}$$

where $\boldsymbol{A} \in \mathbb{R}^{(m+1)\times n}$, $\boldsymbol{A} = [\mathbf{1}; \boldsymbol{D}]$. Each column of $\boldsymbol{A}$ is an augmented training sample.

As $n < m$, this system of linear equations is underdetermined. The condition of existing a solution $\boldsymbol{w}$ is $rank(\boldsymbol{A}^{\mathrm{T}}) = rank([\boldsymbol{A}^{\mathrm{T}}, \boldsymbol{c}])$. For rich high dimensional mass spectrometry data, this condition is not difficult to hold. In practice, due to biological complexity, it is much likely that the data is of full rank, that is $R(\boldsymbol{A}^{\mathrm{T}}) = n$, in which case case, $rank(\boldsymbol{A}^{\mathrm{T}}) = rank([\boldsymbol{A}^{\mathrm{T}}, \boldsymbol{c}])$ holds as $[\boldsymbol{A}^{\mathrm{T}}, \boldsymbol{c}]$ is also of full rank. Thus, we can state that it is much likely that mass spectrometry data are linearly separable. As long as the data are linearly separable, there are infinite solutions $\boldsymbol{w}$s, and therefore there are infinite hyperplanes separating the two groups in $\boldsymbol{A}$ perfectly. That is we can obtain zero training error. The learning of a linear classifier should consider the trade-off between two efforts: minimizing the training error and maximizing the generalization capability. In this linear separable case, we need to focus on the second one. For any positive training sample, $\boldsymbol{x}^+$, we have $g(\boldsymbol{x}^+) = +1$, and for any negative training sample $\boldsymbol{x}^-$, we have $g(\boldsymbol{x}^-) = -1$. Since the distance of $\boldsymbol{x}^+$ and $\boldsymbol{x}^-$ to the hyperplane $g(\boldsymbol{x}) = 0$ is $d(\boldsymbol{x}^+) = \frac{|g(\boldsymbol{x}^+)|}{\|\boldsymbol{w}\|_2} = \frac{1}{\|\boldsymbol{w}\|_2}$ and $d(\boldsymbol{x}^-) = \frac{|g(\boldsymbol{x}^-)|}{\|\boldsymbol{w}\|_2} = \frac{1}{\|\boldsymbol{w}\|_2}$, the margin between the two classes is $m_\pm = d(\boldsymbol{x}^+) + d(\boldsymbol{x}^-) = \frac{2}{\|\boldsymbol{w}\|_2}$. For the generalization purpose, this margin should be as wide as possible, that is the effort should be maximizing $\frac{2}{\|\boldsymbol{w}\|_2}$ which is equivalent to minimizing $\|\boldsymbol{w}\|_2$. Now let us summarize our task formally as below,

$$\begin{aligned} &\min_{\boldsymbol{w}} \frac{1}{2}\|\boldsymbol{w}\|_2, \\ &\text{s.t. } \boldsymbol{A}^{\mathrm{T}}\boldsymbol{w} = \boldsymbol{c}, \end{aligned} \tag{4}$$

where the objective is to maximize the generalization capability and the constraint is to keep zero training error. We coin this method as *high dimensional linear machine* (HDLM). Equation 4 is the well-known least l_2-norm problem, and therefore has analytical optimal solution: $\boldsymbol{w}^* = (\boldsymbol{A}^{\mathrm{T}})^\dagger \boldsymbol{c}$ where $(\boldsymbol{A}^{\mathrm{T}})^\dagger = \boldsymbol{A}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})^{-1}$ is the Moore-Penrose pseudoinverse [12]. Therefore, the hyperplane is

$$g(\boldsymbol{x}) = \boldsymbol{w}^{*\mathrm{T}}\boldsymbol{x} = (\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{x} = 0. \tag{5}$$

Since $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$ might be singular, its inverse can be computed by *singular value decomposition* (SVD) [13].

After obtaining $\boldsymbol{w}^*$, the learning step is finished. The second step is the prediction step. Given a unknown sample, $\boldsymbol{s}$ (augmented), the class label of $\boldsymbol{s}$ is predicted through the relation of $\boldsymbol{s}$ and the hyperplane learned. That is the decision rule is defined as

$$d(\boldsymbol{s}) = \begin{cases} +1 & g(\boldsymbol{s}) > 0 \\ -1 & g(\boldsymbol{s}) < 0 \\ rand\{-1, +1\} & g(\boldsymbol{s}) = 0 \end{cases}, \tag{6}$$

where $rand\{-1, +1\}$ returns either -1 or +1 with equal probabilities (suppose equal priors).

2.1 General Linear Models for Classification

HDLM looks similar with hard-margin SVM which is expressed as

$$\begin{aligned} &\min_{\boldsymbol{w}} \frac{1}{2}\|\boldsymbol{w}\|_2, \\ &\text{s.t. } \boldsymbol{A}^{\mathrm{T}}\boldsymbol{w} \geq \boldsymbol{c}. \end{aligned} \tag{7}$$

In fact, both HDLM and SVM are the special cases of the following general linear model for classification:

$$\min \frac{1}{2}\|\boldsymbol{w}\|_2 + \lambda l(\boldsymbol{A}, \boldsymbol{c}, \boldsymbol{w}), \tag{8}$$

where the second term is a loss function of training, and parameter λ controls the trade-off between the capability of generalization and training precision. For SVM, $l(\boldsymbol{A}, \boldsymbol{c}, \boldsymbol{w}) = \sum_{i=1}^{n} \max(0, 1 - c_i \boldsymbol{a}_i^{\mathrm{T}} \boldsymbol{w})$, where $\boldsymbol{a}_i^{\mathrm{T}}$ is the i-th row of $\boldsymbol{A}$. This is the well-known hinge loss. The loss function of HDLM is essentially square loss which is expressed as $l(\boldsymbol{A}, \boldsymbol{c}, \boldsymbol{w}) = \sum_{i=1}^{n} (c_i - \boldsymbol{a}_i^{\mathrm{T}} \boldsymbol{w})^2 = \|\boldsymbol{c} - \boldsymbol{A}\boldsymbol{w}\|_2^2$. From this we can see that the optimization of HDLM is essentially rigid regression. The advantage of HDLM over SVM is that HDLM makes use of the specific assumption of linear separability of high-dimensional mass spectrometry data, and has analytical solution which is fast to compute.

2.2 Kernel HDLM

Most of the linear models can be kernelized due to the fact that their training and prediction step only require the inner products between samples. This is also indeed true for HDLM. From Equation 5, we can see that the prediction of a unknown sample $\boldsymbol{s}$ only needs the inner products $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$ and $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{s}$. Therefore HDLM can be kernelized via replacing the inner products by kernel matrices $k(\boldsymbol{A}, \boldsymbol{A})$ and $k(\boldsymbol{A}, \boldsymbol{s})$. Because of this, we can find that the computation of HDLM is dimension-free, and the kernelization provides HLDM a flexible choice of representing complex patterns and dealing with noise and redundancy.

2.3 Increase the Performance

The classification performance can be measured by sensitivity ($sen. = \frac{TP}{TP+FN}$), specificity ($spec. = \frac{TN}{TN+FP}$), accuracy ($acc. = \frac{TP+TN}{TP+FN+TN+FP}$), and balanced accuracy ($BACC = \frac{sen.+spec.}{2}$), where TP, TN, FP, and FN are defined as the numbers of true positive, true negative, false positive, and false negative samples, respectively. Due to unbalanced group sizes and distributions of the groups, the sensitivity and specificity may be unbalanced, and therefore the accuracy may not reflect the true discriminative capability of the classifier. As a linear classifier, HDLM use the default threshold 0 in the decision rule (Equation 6). Thus, we need to adjust threshold, which is a variable that can be denoted by t. Our threshold learning method is described as below. As t increases from a reasonable value, the sensitivity increases to 1 while the specificity decreases to 0. Therefore, the sensitivity and specificity are functions with respect to t, respectively. The sensitivities and the corresponding specificities can be described by a *receiver operating characteristic* (ROC) curve [14]. An example of a ROC curve is shown in Figure 1. The far the ROC curve is away the line passing $(0, 0)$ and $(1, 1)$, the better a classifier is. The general quality of a classifier can be measured by area under the ROC curve (AUC). For application, we are also interested in choosing a threshold parameter of a specific classifier which leads to better performance than other thresholds. The distance between a point on the ROC curve to the straight line passing $(0, 0)$ and $(1, 1)$ is denoted by $d(t)$. We define the optimal pair of sensitivity and specificity as the one corresponding to the optimal $d(t)$, that is $d(t^*)$. t^* is the optimal threshold to learn. From Figure 1, we can easily find the relation between $d(t)$ and sensitivity and specificity: $d(t) = \frac{1}{\sqrt{2}}(Sen. - (1 - Spec.)) = \frac{1}{\sqrt{2}}(2BACC - 1)$. Practically, we can obtain t^* through measuring the mean BACC of k-fold CV of a binary linear classifier taking threshold t over the training set. The mean BACC can be denoted by function $MBACC(t, k, trainingset)$. Formally, $t^* = \arg_t \max MBACC(t, k, trainingset)$, where $t = -1 : 0.01 : 1$ (a MATLAB notation that generating a vector through increasing -1 to 1 by step 0.01). For narrative convenience, we coin this threshold adjusted HDLM as TA-HDLM.

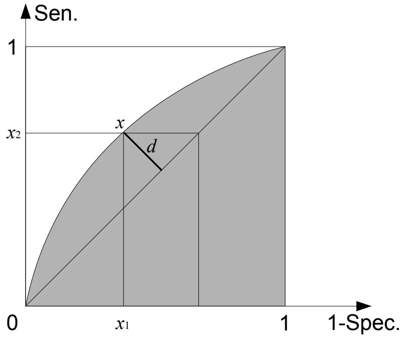

Fig. 1. ROC Curve

Other techniques such as feature selection [4], sample selection [15], classifier ensemble [16], and transductive learning [11, 17] are studied in machine learning to increase the performance of a classification approach. But for HDLM the most direct way is to tune the threshold in the decision rule. Since HDLM works in high dimensional setting, feature selection for the dimension reduction purpose is not applicable here. The linearity of samples in the high dimension discourages us to apply sample selection. Classifier ensemble is effective for weak classifiers, HDLM, however, does not fall into such class. Therefore we do not employ the bagging and boosting strategies. Transductive learning is a good choice when there are few labeled training samples, and many unlabeled training samples and testing samples. In our bioinformatics application of this study, it is unlikely to have a large number of unknown samples at once waiting for being diagnosed. For this reason we do not choose transductive learning.

2.4 Benchmark Methods

In order to be aware of the classification performance of the HDLM classifier, it is necessary to compare with other benchmark classification approaches. We includes two categories of classifiers as benchmark methods.

The first category is composed of the instance learning methods: *1-nearest neighbor* (1-NN), and two sparse representation methods. *Sparse representations* [18] [19] are novel and effective methods in the filed of pattern recognition. The fundamental idea is that an unknown sample can be represented by a linear combination of all the training samples for all of the classes. The sparse combination coefficients are obtained by minimizing the l_1-norm. Then the coefficients are partitioned according to classes.

The linear regression residual can be computed for each class using such corresponding coefficients. The unknown sample is assigned to the class which obtains the smallest regression residual. The implementation of *sparse representation classifier* (SRC) proposed in [18] works well in the case that the number of training samples is equal to or greater than the number of features, while it is difficult to control the regression error in the constraint. Therefore SRC is not applicable in the classification of mass spectrometry without any aid of feature selection. With the specific purpose to classify data of LFSS, the *non-negative least squares* (NNLS) classifier is recently proposed in [20]. The main idea of the NNLS classification method is that any new sample with unknown class label is assumed to be a sparse non-negative linear combination of the training samples. The combination coefficient is the non-negative least squares solution. And the training sample with the dominantly largest coefficient should reside in the same class as this new sample. Bootstrap NNLS (BNNLS) is also proposed in [20] to improve the prediction performance. NNLS and BNNLS are included in our benchmark methods in this study.

The second category consists of two SVMs and the recently proposed *extreme learning machine* (ELM) [21]. As state-of-art method, SVM is studied intensively and has been successfully applied in various fields, for example bioinformatics. The idea of SVM is to map the samples to a higher dimensional space where samples are likely to be linearly separable, and then to maximize the (hard or soft) margin between two groups. The kernel trick avoids the direct mapping and does optimization in the original space. Two kernel functions, *radial basis function* (rbf) and linear kernels, are utilized for SVM in the study. In fact, the linear kernel does not conduct any mapping. As we point out above, the mass spectrometry data are likely to be linearly separable in the original space. Therefore, the linear SVM may be enough instead of using any other kernel trick. ELM, as a variant of single layer feed-forward neural network, is claimed to be competitive with SVMs even outperform SVMs. As generalization of rbf neural network, ELM randomly assigns the weights, connecting the input to the hidden layer, instead of learning them. And then the weights connecting the hidden layer to the output are obtained as least squares or minimum norm optimizations.

3 Computational Experiments and Discussions

Our proposed methods are evaluated and compared with other benchmark approaches over a pancreatic cancer dataset: PanIN (human pancreatic intraepithelial neoplasia) [22]. This dataset was obtain from mice with premalignant pancreatic cancerous and normal statuses. The dataset is shortly described in Table 1. This dataset contains 101 normal samples and 80 premalignant pancreatic samples. 6771 m/z ratios compose the feature list. Readers are referred to [22] for more description about the collection of the data. The data is downloadable from [23]. Let matrix $\boldsymbol{A}_{6772 \times 181}$ represent the data with each column is an augmented sample. The rank of $\boldsymbol{A}^{\mathrm{T}}$ is estimated to be 181, which means this data is of full rank. Of course in such case, the rank of $[\boldsymbol{A}^{\mathrm{T}}, \boldsymbol{c}]$, which is a matrix concatenating $\boldsymbol{A}^{\mathrm{T}}$ and the class labels (column vector $\boldsymbol{c}$) in column-wise direction, is also 181. Any cancerous sample has the class label +1, and -1 for any normal sample. $Rank(\boldsymbol{A}^{\mathrm{T}}) = Rank([\boldsymbol{A}^{\mathrm{T}}, \boldsymbol{c}])$ indicates that the data are linearly separable. Therefore a subset of $\boldsymbol{A}$ is also linearly separable.

Table 1. Datasets

Data	#Classes	#Features	#Samples	$Rank(\boldsymbol{A}^{\mathrm{T}})$	$Rank([\boldsymbol{A}^{\mathrm{T}}, \boldsymbol{c}])$
PanIN [22]	2	6771	101+80=181	181	181

We used 10-fold cross-validation (CV) to split the data into training and test sets. The classifiers (NNLS, BNNLS, 1-NN, rbf-SVM, linear-SVM, ELM, and HDLM) learn on the training set, and predict the class labels of the test set. The classification performance was measured by sensitivity, specificity, accuracy, and balanced accuracy. The cancer samples are defined as positive samples, while normal samples negative. 10-fold CV reran for 20 times and the averaged result are shown in Table 2. We have the following observations. First, we can observe that the instance learning methods including the sparse representation methods do not perform well. The accuracies are just slightly better than random assignment. This may be because the data are very noisy and the distributions of the cancer and normal groups overlap largely. Second, the well known SVM with rbf kernel loses its power in the premalignant cancer diagnosis, sensitivity of 0 is obtained. But the linear-SVM classifier performs better in such high dimensional data because the data are linearly separable. Third, although it was claimed that ELM has similar even better performance than SVM [21], it performs poorly on this data. Forth, we can see that the performance of HDLM is significantly better than the benchmark approaches. It obtained a specificity of 0.758 and a sensitivity of 0.662. As we have stated above, the sensitivity is much crucial than specificity in disease diagnoses. Though the specificity and sensitivity are still unbalanced, this can be tackled through threshold adjustment. As we can be seen at the last row of Table 2, The sensitivity is increased to 0.710. Although this sacrifice some specificity, the accuracy and BACC do not degenerate dramatically.

The running time, including the training and test time for each pair of training and test sets, was recorded for each classifier. The averaged result is also listed in the last column of Table 2. We can observe that HDLM is much faster than SVMs and NNLSs. Although 1-NN and ELM are much efficient than HDLM, their accuracies are not competitive with HDLM. The fastness of HDLM is because it only needs to solve linear equations, while methods such as decision tree and neural network would be very intolerantly slow to learn over such high dimensional data. This is why feature selection or feature extraction have to be done when using decision tree and neural networks. Due to the threshold adjustment, TA-HDLM has the highest computational cost. This cost is still clinically acceptable to learn on about 163 training samples and predict about 18 unknown samples. unlike instance-based learning, once the learning of the HDLM model finishing, the prediction is actually very fast.

Next, we compared our methods with the performance reported in [7]. Readers should be aware that we find that the normal samples are incorrectly defined as positive samples while cancer samples negative in [7]. This can be proved as follows.

Table 2. Classification Performance

Method	Spec.(STD)	Sen.(STD)	Acc.(STD)	BACC(STD)	Time (CPU sec.)
NNLS	0.573(0.036)	0.491(0.039)	0.536(0.028)	0.532(0.028)	0.312
BNNLS	0.580(0.040)	0.484(0.033)	0.538(0.029)	0.532(0.028)	12.276
1-NN	0.583(0.030)	0.479(0.040)	0.537(0.020)	0.531(0.021)	0.056
rbf-SVM	**1**(0)	0(0)	0.558(0)	0.500(0)	1.393
linear-SVM	0.802(0.025)	0.424(0.040)	0.635(0.021)	0.613(0.022)	1.395
ELM	0.513(0.043)	0.488(0.056)	0.501(0.030)	0.500(0.031)	0.053
HDLM	0.758(0.020)	0.662(0.033)	**0.716**(0.021)	**0.710**(0.022)	0.193
TA-HDLM	0.704(0.031)	**0.710**(0.040)	**0.707**(0.021)	**0.707**(0.021)	75.633

Suppose $\frac{TP+FN}{TN+FP} = \alpha$. According to the definitions of sensitivity and specificity, we have $TP = Sen.(TP + FN) = Sen.\alpha(TN + FP)$ and $TN = Spec.(TN + FP)$. According to the definition of accuracy, we further have

$$\begin{aligned} Acc. &= \frac{TP + TN}{TP + FN + TN + FP} \\ &= \frac{Sen.\alpha(TN + FP) + Spec.(TN + FP)}{(1 + \alpha)(TN + FP)} \\ &= \frac{Sen.\alpha + Spec.}{1 + \alpha}. \end{aligned} \tag{9}$$

Therefore we have $\alpha = \frac{Spec.-Acc.}{Acc.-Sen.}$. Take C4.5 in Table 4 in [7] for example, $\alpha = \frac{0.21-0.6444}{0.6444-0.99} = 1.2569$ which approximates to $\frac{101}{80} = 1.2625$ (the ratio of the number of normal samples to the number of cancer samples in the whole data) or $\frac{10}{8} = 1.25$ (the ratio of number of normal samples to the number of cancer samples in a test set). Readers can verify this using more results from [7]. Therefore we need to swap the sensitivity and specificity in the results of [7] and compare them with the results of our methods. Now back to our comparison. The comparison result is shown in Table 3. 10-fold CV was also used in [7]. The first 3 blocks in this table are top results from [7] with respect to accuracy. $+S$, $+W$, and $+G$ mean the combinations with Student t-test feature ranking, Wilcoxon rank test, and genetic feature selection, respectively. It can be seen that HDLM outperforms these methods, except Logistic+S, in [7], in accuracy and BACC. TA-HDLM obtained the highest sensitivity (0.71) among these methods. The Multiboost+W as one of the classifier ensemble methods only obtained a sensitivity of 0.660. Logistic+S and Neural Network+S achieved the sensitivity of 0.700 which is slightly lower than TA-HDLM.

It has to be noted that we did not conduct any preprocessing for our proposed methods and benchmark methods applied in this study, except that, for the cases of SVM and ELM, the ion intensities of each m/z ratio in the training set are normalized to have mean 0 and standard deviation 1. The normalization parameters estimated from the training set are used to normalize the test set. Our normalization is different from the one in the preprocessing stage of [7] where the whole dataset are normalized and

Table 3. Classification Performance

Method	Spec.	Sen.	Acc.	BACC
Logistic+S	0.790	**0.700**	0.750	0.745
Neural Network+S	0.700	**0.700**	0.700	0.700
Random Forest+W	0.790	0.590	0.700	0.690
Multiboost+W	0.730	0.660	0.700	0.695
SVM+G	0.720	0.530	0.633	0.625
Logitboost+G	0.680	0.540	0.617	0.610
Adaboost+G	0.670	0.550	0.617	0.610
linear-SVM	0.802	0.424	0.635	0.613
HDLM	0.758	0.662	0.716	0.710
TA-HDLM	0.704	**0.710**	0.707	0.707

scaled. Care has to be taken in this and other preprocessing in [7], because the test samples should keep intact before the test stage of inductive learning. If the preprocessing of the training set is influenced by the test set, the predicting ability of a classifier (and a feature selection method) is inflated, because the more or less information in the test set has been divulge in the learning stage. It is not to say that the information in the test set should not be used in the learning stage. We have to discuss this in two aspects. For inductive learning of a feature selection and a classifier, the test set should never be touched during learning in order to have a fair evaluation of the capability of feature selection and classifier. One common mistake is that feature selection, that is biomarker or peak identification in the study of mass spectrometry data analysis, is done over the whole data (training set and test set), after that a classifier learns over the training set, and the prediction accuracy of the test set is reported as the evaluation of quality of the feature selection. This actually overestimates the capability of the feature selection. However, if the purpose is not to evaluate a feature selection or a classifier, but is the prediction accuracy, the information of the unlabeled testing samples can be used during learning. This falls into the category of transductive learning [11] and semi-supervised learning [17]. Actually the prediction accuracy obtained using transductive learning is often higher than that using inductive learning. Since all samples are utilized during the preprocessing including baseline correction, sample scaling, and smoothing in [7], though the prediction accuracy is acceptable for the purpose of classification, the performances of feature selection and classifiers are more or less overestimated, and the biomarkers reported more or less overfit the whole data as well. Overfitting can lead poor capability of generalization. A suggested way of biomarker identification is that the performances of feature selection and classifier are evaluated over k-fold CV without using test information during preprocessing, and once such confidence is obtained about the feature selection and classifier (no biomarker is reported as they vary from fold to fold), the biomarkers are selected over the whole data and reported (because the confidence of such feature selection method and the quality of the selected features has already established before). All in all, purposes must be clear when design computational experiments. And special care has to be taken that the class labels of a test set should never be used in any model for any purpose.

4 Conclusion and Future Work

It is crucial to diagnose premalignant pancreatic cancer in a very early stage in order to increase the survival rate of patients. However, it is clinically and computationally difficult. This paper propose to apply fast HDLM as computational model to predict the cancer samples obtained through high resolution mass spectrometry. HDLM can avoid overfitting through maximizing margin and kernelization. Its computation is dimension-free. Experiments show that our HDLM methods achieve competitive performance. Comparison with reported performance shows that our approaches significantly outperform most of the benchmark and proposed approaches. Due to high performance and simplicity of implementation, it will be beneficial to use our methods to the diagnosis of premalignant pancreatic cancer which is suffering low accuracy and sensitivity. And our approaches, combining with the mass spectrometry protein profiling technique, can also be applied to the prediction of other premalignant cancers at an early stage. As future work, our methods will be tested on more protein mass spectrometry data. The performance of different loss functions on high-dimensional mass spectrometry data is still unknown. We will statistically and experimentally compare the performance of HDLM with other liner models of different loss functions on more data. It is also worth investigating suitable kernels for mass spectrometry data.

Acknowledgments. This research has been supported by IEEE CIS Walter Karplus Summer Research Grant 2010, Ontario Graduate Scholarship 2011-2012, and Canadian NSERC Grants #RGPIN228117-2011.

References

1. Ma, B.: Challenges in Computational Analysis of Mass Spectrometry Data for Proteomics. Journal of Computer Science and Technology 25(1), 107–123 (2010)
2. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, New York (2006)
3. Levner, I.: Feature Selection and Nearest Centroid Classification for Protein Mass Spectrometry. BMC Bioinformatics 6, e68 (2005)
4. Saeys, Y., Inza, I., Larrañaga, P.: A Review of Feature Selection Techniques in Bioinformatics. Bioinformatics 23(19), 2507–2517 (2007)
5. Shawe-Taylor, J., Cristianini, N.: Pattern Recognition and Machine Learning. Cambridge University Press, Cambridge (2004)
6. Pawa, N., Wright, J.M., Arulampalam, T.H.A.: Mass Spectrometry Based Proteomic Profiling for Pancreatic Cancer. JOP. J Pancreas. 11(5), 423–426 (2010)
7. Ge, G., Wong, G.W.: Classification of Premalignant Pancreatic Cancer Mass-Spectrometry Data Using Decision Tree Ensembles. BMC Bioinformatics 9, 275 (2008)
8. Donoho, D.L.: High-Dimensional Data Analysis: The Curses and Blessings of Dimensionality. Lecture in Math Challenges of the 21st Century, pp. 1–32 (2000)
9. Knies, R.: Yi Ma and the Blessing of Dimensionality. Microsoft Research Featured Story (May 28, 2010), http://research.microsoft.com/en-us/news/features/dimensionality-052810.aspx
10. Kroeker, K.L.: Face Recognition Breakthrough. Communications of the ACM 52(8), 18–19 (2010)

11. Vapnik, V.: Statistical Learning Theory, pp. 339–371. Wiley, New York (1998)
12. Chong, E.K.P., Żak, S.H.: An Introduction to Optimization, 3rd edn., pp. 211–246. Wiley, New York (2008)
13. Golub, G.H., Van Loan, C.F.: Matrix Computations, 3rd edn., pp. 206–274. Johns Hopkins, Baltimore (1996)
14. Fawcett, T.: An Introduction to ROC Analysis. Pattern Recognition Letters 27, 861–874 (2006)
15. Mundra, P.A., Rajapakse, J.C.: Gene and Sample Selection for Cancer Classification with Support Vectors Based t-statistic. Neurocomputing 73(13-15), 2353–2362 (2010)
16. Rokach, L.: Ensemble-Based Classifiers. Artificial Intelligence Review 33(1-2), 1–39 (2010)
17. Chapelle, O., Schölkopf, B., Zien, A.: Semi-Supervised Learning, pp. 453–472. MIT Press, Cambridge (2006)
18. Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust Face Recognition via Sparse Representation. IEEE Transactions on Pattern Analysis and Machine Intelligence 31(2), 210–227 (2010)
19. Wright, J., Ma, Y., Mairal, J., Sapiro, G., Hung, T.S., Yan, S.: Sparse Representation for Computer Vision and Pattern Recognition. Proceedings of The IEEE 98(6), 1031–1044 (2010)
20. Li, Y., Ngom, A.: Classification Approach Based on Non-Negative Least Squares. Technical Report. No. 12-010, School of Computer Science, University of Windsor (2012)
21. Zhang, R., Huang, G.B., Sundararajan, N., Saratchandran, P.: Multicategory Classification Using an Extreme Learning Machine for Microarray Gene Expression Cancer Diagnosis. IEEE/ACM Transactions on Computational Biology and Bioinformatics 4(3), 487–495 (2007)
22. Hingorani, S.R., et al.: Preinvasive and Invasive Ductal Pancreatic Cancer and Its Early Detection in The Mouse. Cancer Cell 4, 437–450 (2003)
23. http://home.ccr.cancer.gov/ncifdaproteomics/ppatterns.asp