Computational Methods for Analysis of MALDI-TOF Spectra to Discover Peptide Serum Biomarkers

Running head: Computational Methods for Analysis of MALDI-TOF Spectra

Authors:

Habtom W. Ressom PhD

Rency S. Varghese MS

Radoslav Goldman PhD

Georgetown University, Department of Oncology, Lombardi Comprehensive Cancer Center, Washington, DC, USA

Corresponding author:

Habtom W. Ressom

Department of Oncology

Georgetown University Medical Center

Building D, Room 174

Washington, DC 20057

Tel: 202-687-2283

Fax: 202-687-0227

EMAIL: hwr@georgetown.edu

1. Introduction

The characterization of peptides in serum and plasma by mass spectrometry (MS) is one of the promising strategies for biomarker discovery (1, 2). In addition to careful experimental design, improved mass spectrometric technology and sample preparation methods, innovative computational methods are needed to facilitate data interpretation and extract useful information from high dimensional MS data. In this chapter, we describe computational methods for analysis of raw mass spectra generated by MALDI-TOF/TOF instrument (Bruker Daltonics, Billerica, MA) and exported as text files by the Flex Analysis software. We bin the text files using ‘Protein Group Report Generator 1.6’, which is a Visual Basic 6 application, and import the binned data into MATLAB, where we apply spectral preprocessing methods as well as statistical and machine learning methods to extract candidate peptide biomarkers.

The purpose of data preprocessing is to correct intensity and m/z values in order to: (1) reduce noise, (2) reduce amount of data, and (3) make the spectra comparable to each other. For example, binning reduces the dimension of a spectrum by grouping intensity measurements at adjacent m/z values into bins. Its aim is to preserve raw data information while performing a dimensional reduction for subsequent processing and mining phases. Smoothing is a process by which data points are averaged with their neighbors as in a time-series of data. Its main aim is to reduce noise, i.e., to increase signal to noise ratio. Baseline correction flattens the base profile of a spectrum to minimize the impact of varying baseline caused by the chemical noise in the matrix or by ion overloading; drifting baseline introduces serious distortion of ion intensities without adequate correction. Normalization reduces systematic variation that may be caused by varying amounts of protein, sample degradation over time, or variation in the sensitivity of the MS ion detector. Normalization enables the comparison of different samples since the absolute peaks of different spectra could be incomparable. Peak detection deals with the identification of peaks that display a reasonable intensity compared to those that may be just noise. Hence, the main task of peak detection is separating real peaks (e.g. corresponding to peptides) from peaks representing noise. Although this task can be done visually by mass spectrometry experts, algorithms that do not require human intervention are needed for rapid and repeatable quantitative processing of spectra that often contain hundreds of discrete peaks. The simple peak finding (SPF) provides the locations of all local maximum peaks and their associated left-hand and right-hand bases (3). Peak calibration allows correction of drifts that do not reflect any real sample variation. Without peak calibration, the same peak (e.g. the same peptide) can have different m/z values across samples. To allow an easy and effective comparison of different spectra, peak alignment methods find a common set of peak locations (i.e. m/z values) in a set of spectra, in such a way that all spectra have common m/z values for the same biological entities. For example, dynamic programming can be used to align peaks across spectra (4). However, this approach and other peak detection tools such as Proteome Quest (Correlogics Systems, Bethesda, MD) deal with exact mass points (single peak). Hence, they do not address the existence of isotopes representing the same peptide, where the same peptide could have multiple peaks. It is important to take into consideration that each detected m/z value is affected by noise and isotopes causing the presence of a window in which the m/z ratio can be shifted. A window is defined here to indicate the range of potential m/z shifting for each peak location (5). To automate the creation of windows, neighboring peaks are coalesced, if they fall within a pre-specified mass separation (3). A mass separation cutoff is determined based on the instrument's tolerance.

Finally, peaks that best discriminate the subjects within a subgroup are selected via a feature selection algorithm. In mass spectral data analysis, the peak dimensionality is usually larger than or comparable to the sample size. This makes many standard pattern classification algorithms fail to address the high risk of overfitting. Thus, there is an algorithmic need for peak selection in addition to the biological need of discovering a manageable set of key disease biomarkers (6). A commonly used approach in peak selection is to apply statistical analyses such as t-test and weighting factor, employed in the weighted voting algorithm (7) that recognize differentially abundant peaks between two groups with multiple subjects. These peaks are then used as inputs to a pattern classification algorithm such as support vector machine (SVM). This approach has the following limitations: (a) it selects peaks based on "relevance criterion", not on the basis of their "usefulness" (i.e., prediction capability); (b) redundant peaks can exist; and (c) peaks that have strong discriminant power jointly, but are weak individually are ignored. The SVM recursive feature elimination (SVM-RFE) algorithm recursively classifies samples with SVM and select features according to their SVM weights (8). Benefiting from the good performance of SVMs in high dimensional gene-expression data, SVM-RFE is often considered as one of the best feature selection algorithms in the literature. SVM-RFE ranks the features once using all samples, and uses the top ranked features in the subsequent cross-validation. This will generate a biased estimation of errors and limits the search space by allowing only the top ranked features as candidate features. The goal is to search for a manageable combination of useful features from the entire set of peaks. However, due to the large number of peaks, a systematic method is required to select the best combination of peaks without examining all possible combinations. Stochastic global optimization methods such as genetic algorithms (GAs), simulated annealing, and swarm intelligence (SI) methods are ideal candidates for selecting features from a high-dimensional search space. The recent release of ClinProTools^{TM} uses GAs to determine features for SVM classifiers. In this chapter, we provide codes that can be used to select the optimal peak set by combining SVM with a special type of SI method, ant colony optimization (ACO). ACO allows the integration of features selected on the basis of both “relevance” and “usefulness” criteria (9).

This chapter is organized as follows: Section 2 lists the software tools needed for MALDI-TOF MS data analysis including codes written by the authors of this chapter. Section 3 highlights the methodologies for spectral preprocessing and peak selection. Also, this section outlines the procedure to run the codes, in which the methodologies are implemented. Finally, Section 4 provides additional notes and alternative procedures.

2. Materials

2.1 FlexAnalysis software

FlexAnalysis software is provided by Bruker Daltonics (Billerica, MA). Raw mass spectra generated by MALDI-TOF/TOF instrument can be viewed and saved using this software**.**

2.2 Binning software tool

1. Protein Group Report Generator 1.6 (PGRG) (http://microarray.georgetown.edu/files/msdat.zip)
2. VB6 Run-time and VB6 Service Pack available at Microsoft website.

2.3 MATLAB tools

1. MATLAB software (Windows version)
2. MATLAB Bioinformatics Toolbox
3. MATLAB Statistics Toolbox
4. OSU SVM Toolbox for MATLAB (http://sourceforge.net/projects/svm/)
5. MATLAB codes (http://microarray.georgetown.edu/files/msdat.zip). These codes include m-files the authors of this manuscript wrote and additional m-files from The Cromwell Package (http://bioinformatics.mdanderson.org/cromwell.html) and SVM-RFE files from (http://www.bo.infn.it/~masotti/software.html)

3. Methods

Figure 1 illustrates our methodology for peak selection. The spectra in the labeled set are used for peak selection. The resulting peaks and the associated SVM classifier is used to predict the disease state of the spectra in the blinded set. Spectral preprocessing (i.e., binning, baseline correction, normalization, peak detection, and peak calibration) and peak selection are performed on the labeled set by subjecting the entire process to cross-validation.

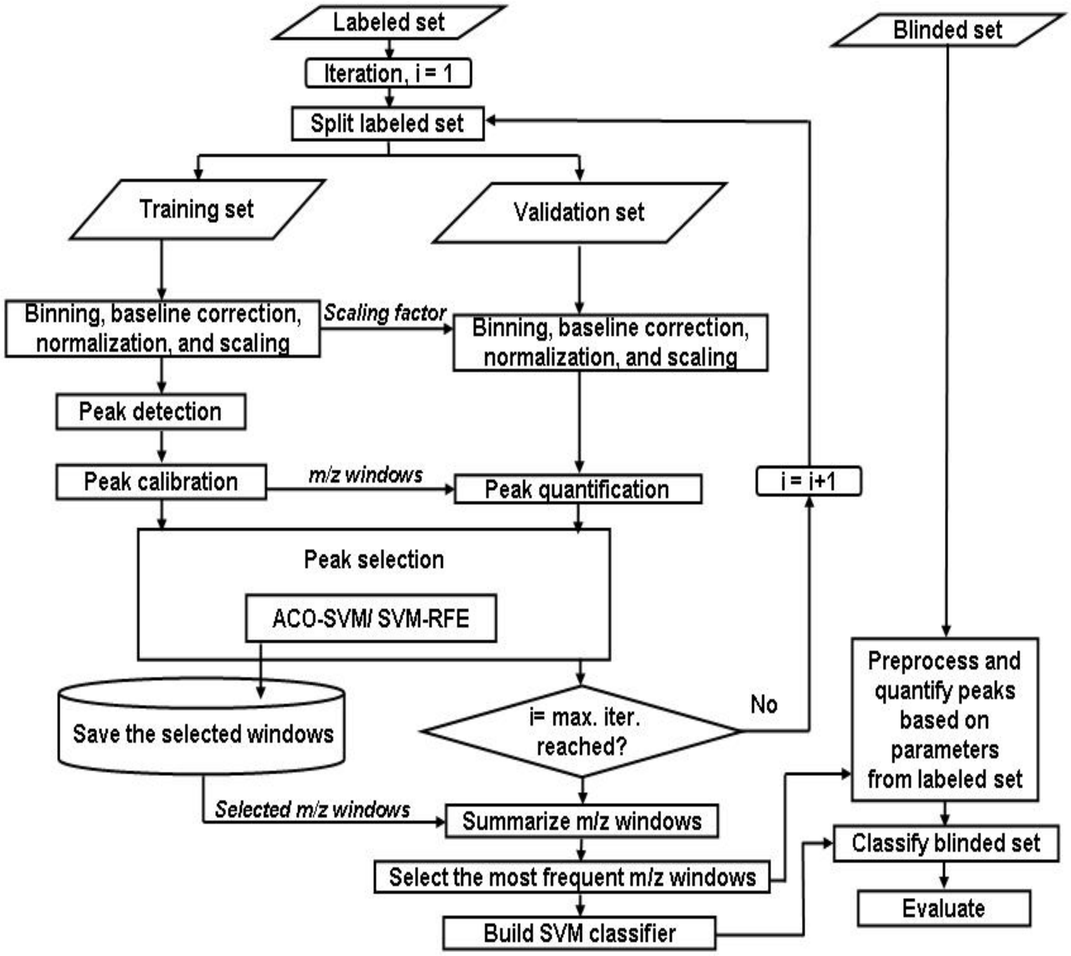

Figure 1. Methodology for peak detection.

As illustrated in Figure 1, the spectra in the labeled set are split into training and validation sets. The performance of the peaks selected from the training spectra are evaluated via validation spectra to guide the search for the optimal peak set. In the following, we describe briefly the specific methodologies we implemented for spectral preprocessing and peak selection.

3.1 Spectral Preprocessing

Binning reduces the dimension of a spectrum by grouping intensity measurements at adjacent m/z values into bins. Its aim is to preserve raw data information while performing a dimensional reduction for subsequent processing and mining phases

We estimate the baseline of a binned spectrum by obtaining the minimum value within a shifting window size of 50 bins and a step size of 50 bins of each spectrum. Spline approximation is applied to regress the varying baseline. The regressed baseline is smoothed using the lowess smoothing method. The resulting baseline is subtracted from the spectrum.

Each spectrum is normalized by dividing by its total ion current. After scaling the peak intensities of the normalized training spectra to an overall maximum intensity of 100, local maximum peaks above a specified threshold are identified and peaks that fall within a prespecified mass separation are coalesced into a single m/z window to account for drift in m/z location and to represent isotopic clusters by a single peak. The maximum intensity in each window is used as the variable of interest.

3.2 Peak selection

We apply the hybrid ACO-SVM algorithm to search for a peak set that consists of a pre-specified number of peaks. ACO-SVM selects a peak set on the basis of its ability to distinguish the case and control spectra in the validation set. Note that the spectra in the validation set are not involved in the spectral preprocessing. They are binned, baseline corrected, normalized, and scaled on the basis of the parameter used to preprocess the spectra in the training set. These parameters include scaling factor that standardizes the peaks in the training set to have a maximum of 100. The peaks in the validation set are quantified at the selected m/z windows and are presented to SVM classifier previously trained using the peaks from the training set. The performance of the SVM classifier in predicting the disease state of the subjects in the validation set is used by ACO-SVM to guide its search for the optimal peak set.

Alternatively, the user can choose the SVM-RFE algorithm, which uses the labeled set to determine the most useful set of peaks.

The above steps (spectral preprocessing and peak selection) are repeated multiple times by randomly splitting the labeled spectra into training and validation sets with resubstitution. The peaks selected in multiple iterations (i=1,2,…) are summarized by merging overlapping m/z windows to determine the most frequently selected m/z windows. Note that the number of peaks detected and the size of the m/z windows could vary due to the change in the population set at each iteration.

To evaluate the peak selection process, we quantify the peak intensities at the m/z windows of the final peak set in both the labeled and blinded sets. Note that the blinded set is not used during the entire peak selection process, thus it serves as an independent set to evaluate the generalization capability of the selected peaks. Alternatively, if the disease state of the spectra in the blinded set is unknown, an SVM classifier built via the labeled set will be used for prediction. The spectra in the blinded set are binned, baseline corrected, normalized, and scaled on the basis of parameters used to preprocess the spectra in the labeled set.

3.3. Procedure

1. Place the provided *PGRG.exe* in any directory and create the following folders: "INPUT_GRP1", "INPUT_GRP2", and “OUTPUT"
2. Place input files (i.e., .TXT files from FlexAnalysis software) into either INPUT_GRP1 or INPUT_GRP2. Alternatively, split the TXT files into two phenotypic groups and place them in the two folders.
3. Run the PGRG.exe
4. Provide values for all the form fields

   Note: Choose the CSV option for Output file. The Output File name should not contain characters like / \: etc.
5. Choose either “Sum Intensity Values per Reported ID” or “Calculate Average Intensity per Reported ID.”
6. Press the Generate Report button.

   The application generates an Output File in the OUTPUT folder. PGRG uses data from all the TXT files stored in the INPUT_GRP1 or INPUT_GRP2 or both directories to generate the report. If only one of the INPUT folders contains input files, then a simple report will be generated. If both input folders contain input files, the generated report contains five more columns for group comparison purposes.
7. Copy the EXCEL file that has the binned data into a folder where the MATLAB files are copied.
8. Start MATLAB
9. Run the m-file *ProteomicAnalysis.m*, which will prompt you to provide the following information sequentially:

9.1 Enter the name of the EXCEL file which has the training data. After it reads the data into MATLAB.

9.2 Choose a peak selection method; the options are “SVM-RFE” (default) and ACO-SVM.

9.3 Enter the number of features to be selected. The default is set to three. It is advised to use at least three features.

9.4 Enter the number of times (iterations) the reshuffling algorithm is to be done. The code will run through the baseline correction and normalization process, peak detection, and peak selection process.

9.5 Select the desired number of peaks. (Note that if the reshuffling is done only once, the number of peaks will be equal to the number entered in Procedure 9.3. However, for multiple iterations, a frequency plot will be provided to assist the user to select the number of desired peaks).

9.6 Enter a filename you wish to save the results as a MAT file. The result includes parameters used for normalization, along with the normalized train data, the SVM classifier built for the training data and the selected m/z windows. These will be saved as a MAT file in the name you have provided when prompted for.

10. To predict the disease state of spectra in a blinded set, bin the blinded spectra following Steps 1 to 6. Run the M-file *PredictResults.m*. The m-file will prompt you to provide the following information sequentially:

10.1 Enter the name of the EXCEL file which has the blinded data

10.2 Enter the name of the saved MAT file (which has parameters used to analyze the labeled set) from the labeled data.

The code will perform baseline correction and normalization in the blinded dataset and then quantify the peaks/windows based on the selected peaks from the labeled data. It also uses the saved SVM classifier information and then gives you labels for the blinded dataset.

11. As an alternative to Step 10, the m-file *SummarizeResults.m* allows you to choose a new set of summarized peaks from the previously trained spectra. Running this code prompts you to provide the following information sequentially:

    11.1 Enter the name of the EXCEL file which has the binned blinded data

    11.2 Enter the name of the saved MAT file that has parameters used by the labeled set. If the labeled set was shuffled more than once, a frequency plot will be displayed for the summarized peaks.

    11.3 Choose the number of peaks based on the frequency plot. These peaks will be used to design an SVM classifier which will then predict the labels for the blinded set.

4. Notes

1. As pointed out in Procedures 2 and 6, our binning tool can read .TXT files generated by the FlexAnalysis software either from a single input folder or two folders. If a single input folder is used, the output file will have one column per subject. If the

files are placed into two input folder, then the output file will provide the following additional five columns for group comparison purposes:

[GRP1] "Avrg Intensity" = an Average GRP1 intensity per row

[GRP1] "Val Count" = number of GRP1 intensity values per row

[GRP2] "Avrg Intensity" = an Average GRP2 intensity per row

[GRP2] "Val Count" = number of GRP2 intensity values per row

"GRP |Dif|" = the absolute difference between the [GRP1] and [GRP2]

2. If 'ACO-SVM' is chosen in Procedure 9.2, the algorithm will need a long time to perform the feature selection for multiple generations. The number of ants in this algorithm is set by default to 50 and the number of generations is set to 500. To change these options, the user needs to edit the m-files *PeakSelection_ACOSVM.m* and *peak_ACO_S2N.m*. However, the default peak selection algorithm in Procedure 9.2 (SVM-RFE) requires relatively less computational time.
3. If the feature selection algorithm is run only once (i.e., if number of reshuffling is set to 1 in Procedure 9.4), then frequency plot will not be provided. We suggest that the user performs multiple reshuffling (10-100 iterations) to determine the most frequently selected peaks.
4. If the feature selection algorithm is run for multiple iterations by reshuffling the labeled spectra, then a summarized set of peaks will be provided in Procedure 9.5. The summarization is needed because, when the samples are reshuffled, the number of peaks found for each run and the width of each peak may vary. Summarization allows us to merge overlapping peaks and to provide a frequency plot of the number of occurrence of each summarized peak in multiple iterations. The number of peaks

needed for classification of a blinded set can be chosen based on a frequency plot, which assists users to estimate the optimal number of peaks. The frequency plot presents a bar plot with the number of occurrences versus the peaks sorted in the order of decreasing frequency. A commonly used approach is to select all peaks starting from the first until the frequency curve becomes flat (i.e. the change in frequency becomes low).

5. The MAT file saved following Procedure 9.6 consists of various training parameters including scaling factor, selected peaks, normalized labeled spectra, and an SVM classifier built based on the spectra in the labeled set. If multiple iterations are used in Procedure 9.4, the SVM classifier uses the peaks selected by the user on the basis of the frequency plot (Procedure 9.5). Otherwise, the number of peaks will be equal to the value entered in Procedure 9.3. The training parameters are used by *PredictResults.m* to predict the disease state of the spectra in the blinded set. If the user wants to change the number of peaks without rerunning the peak selection process, previously saved training parameters from Procedure 9.6 can be used by *SummarizeResults.m* (Procedure 11) to display the frequency plot and to allow the selection of the number of peaks needed. Once the number of peaks are selected (Procedure 11.3), *SummarizeResults.m* predicts the disease state of the spectra in the blinded set.

Acknowledgments

This work was supported in part by NSF grant IIS-0812246, Prevent Cancer Foundation grant, NCI grants R21 CA130837, R03 CA119313, and EDRN Associate Membership awarded to HWR, and NCI grant R01 CA115625 to RG.

References

1. Tammen, H., Schulte, I., Hess, R., Menzel, C., Kellmann, M., Mohring, T., and Schulz-Knappe, P. (2005) Peptidomic analysis of human blood specimens: comparison between plasma specimens and serum by differential peptide display. *Proteomics* **5,** 3414-22.
2. Villanueva, J., Shaffer, D. R., Philip, J., Chaparro, C. A., Erdjument-Bromage, H., Olshen, A. B., Fleisher, M., Lilja, H., Brogi, E., Boyd, J., Sanchez-Carbayo, M., Holland, E. C., Cordon-Cardo, C., Scher, H. I., and Tempst, P. (2006) Differential exoprotease activities confer tumor-specific serum peptidome patterns. *J Clin Invest* **116,** 271-84.
3. Coombes, K. R., Tsavachidis, S., Morris, J. S., Baggerly, K. A., Hung, M. C., and Kuerer, H. M. (2004) *in* "Technical Report UTMDABTR-001-04 (http://www.mdanderson.org/pdf/biostats_utmdabtr-001-04.pdf)", The University of Texas M.D. Anderson Cancer Center.
4. Sauve, A. C., and Speed, T. P. (2004) Normalization, baseline correction and alignment of high-throughput mass spectrometry data *Proceedings of the Genomic Signal Processing and Statistics workshop, Baltimore, MD, USA.*
5. Yasui, Y., McLerran, D., Adam, B. L., Winget, M., Thornquist, M., and Feng, Z. (2003) An automated peak identification/calibration procedure for high-dimensional protein measures from mass spectrometers. *J Biomed Biotechnol* **2003,** 242-48.
6. Zhang, X., Lu, X., Shi, Q., Xu, X. Q., Leung, H. C., Harris, L. N., Iglehart, J. D., Miron, A., Liu, J. S., and Wong, W. H. (2006) Recursive SVM feature selection and sample classification for mass-spectrometry and microarray data. *BMC Bioinformatics* **7,** 197.
7. Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A., Bloomfield, C. D., and Lander, E. S. (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science* **286,** 531-7.

8. Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002) Gene Selection for cancer classification using support vector machines. *Machine learning* **46,** 389-422.

9. Ressom, H. W., Varghese, R. S., Drake, S. K., Hortin, G. L., Abdel-Hamid, M., Loffredo, C. A., and Goldman, R. (2007) Peak selection from MALDI-TOF mass spectra using ant colony optimization. *Bioinformatics* **23,** 619-26.