Protein Mass Spectra (SELDI) Data Processing and Classification with "caMassClass" library

By
Jarek Tuszynski (SAIC)

1 License of caMassClass

The caMassClass Software License, Version 1.0

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the disclaimer of Article 3, below. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

2. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

   "This product includes software developed by the SAIC and the National Cancer Institute."

3. If no such end-user documentation is to be included, this acknowledgment shall appear in the software itself, wherever such third-party acknowledgments normally appear.

4. The names "The National Cancer Institute", "NCI" and "SAIC" must not be used to endorse or promote products derived from this software.

5. This license does not authorize the incorporation of this software into any third party proprietary programs. This license does not authorize the recipient to use any trademarks owned by either NCI or SAIC-Frederick.

6. THIS SOFTWARE IS PROVIDED "AS IS," AND ANY EXPRESSED OR IMPLIED WARRANTIES, (INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE) ARE DISCLAIMED. IN NO EVENT SHALL THE NATIONAL CANCER INSTITUTE, SAIC, OR THEIR AFFILIATES BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

2 Discussion of Protein Mass Spectra (SELDI) Data Processing and Classification

2.1 Introduction

The purpose of this task is to build a tool that applies classification algorithms to proteomics data (especially SELDI data). Main intended use of those algorithms is distinguishing cancerous samples from normal samples; however they can be used for other classification problems as well.

Developed tools will be used to extend caWorkbench to allow researchers to perform standard classification operations on protein (SELDI) data collected and stored as a part of caARRAY. Most of the classification methods described in this document can be easily applied for other types of data.

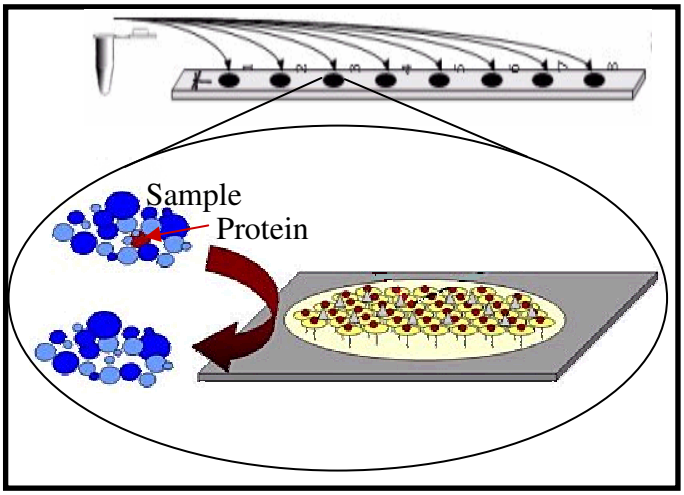

Figure 1: SELDI chip preparation. Drawing adapted from East Virginia Medical School – Virginia Prostate Center[10] website.

2.2 Background

SELDI is a relatively new process which adapted existing mass spectrometry methodology to protein study. An excellent overview this process can be found at [10][21]. The following summary of the process mostly bases on their account.

Chip Array surface is selected from variety of chips specialized in capturing different types of protein samples. Available types are:

- Hydrophobic for reversed-phase chromatography
- Normal phase chromatography
- Anion or cation exchange surface
- Metal binding (IMAC)
- Etc.

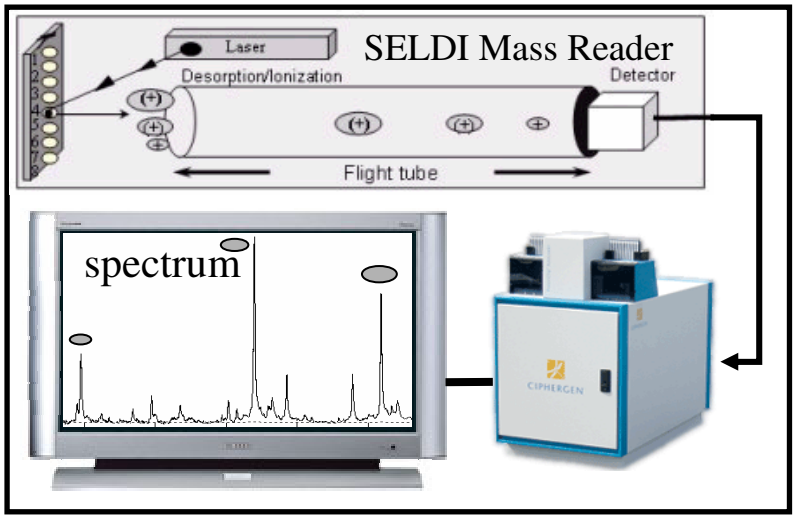

Figure 2: SELDI sample processing. Parts of drawing adapted from EVMS[10] website.

Chip goes through several preparation steps performed either manually or by robotic workstation. Sample is applied onto chip array, where targeted subset of protein binds to the chip surface, while the rest of the sample is washed away. Afterwards, energy absorbing molecule (EAM) is added in order to ionize the proteins and the chip with the sample is

send to “Surface –Enhanced Laser Desorption / Ionization” (SELDI) mass reader. There laser is used to free ionize proteins and allow them to be pulled by magnetic field through vacuum “time-of-flight” tube, where they are separated based on their mass to charge ratio. On the other end of the flight tube their arrival time is recorded by a detector in form of a spectrum. The rest of this paper is concerned with studying those SELDI spectra. This process was first commercialized by Ciphergen Inc. However at present other manufacturers produce competing products.

2.3 Methods

Before SELDI or other protein data can be classified it has to go through several steps of what I will call pre-processing.

Many different researchers used very different methods in order to process and classify SELDI data. However generalized approach is as follows:

I. Data Input
II. Pre-processing
    1. Baseline subtraction
    2. Normalization
    3. Mass-drift Adjustment
    4. Peak finding and alignment (finding biomarkers, feature extraction)
    5. Merging of multiple sample copies
III. Classification
    1. Feature Selection
    2. Building Classifier

Many of the above steps are optional and were skipped by some or most of the researchers.

Data Input

In case of SELDI data exported from Ciphergen software, data at different stages of processing comes in different format.

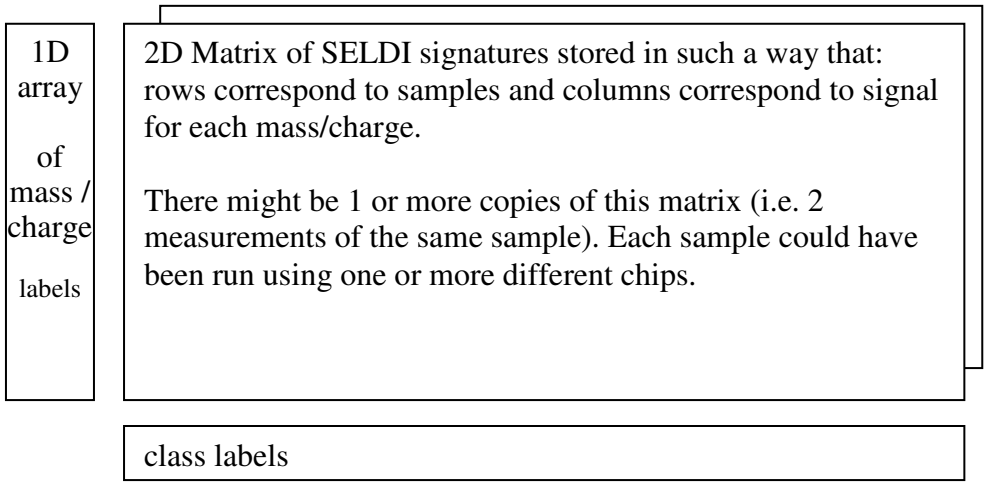

Figure 1: Conceptual view of most common format of SELDI data

1. Raw spectra, baseline subtracted and/or normalized spectra come in the form of separate Excel CSV ("comma separated values") file for each spectrum. Each file contains mass (M/Z) column and intensity column.
2. One could also export data after peak finding step. Those files are also in the CSV format. They store one row per peak, and can contain many different columns describing different aspects of each peak. Among them one should find: spectrum name and/or number, intensity (peak height), "Substance.Mass" (peak mass / position).
3. After peak alignment all data can be exported in a single CSV file that contains one row per spectrum with each column representing different cluster of peaks (biomarker, feature).

Other non vendor-specific formats will likely be developed and used in the future, but for time being those three seem to be most common.

Pre-Processing

Ciphergen SELDI machine comes with its own software which is able to perform some or most of the preprocessing steps. It is up to a user to decide at which point to export their data from Ciphergen environment and start processing it by themselves. However in case of data available on the web, which accompany many papers, researchers that want to reproduce published results have no choice of data format.

If input data is not base-line subtracted this step should be always performed. Base-line is a smooth line that follows local minimum without rising into peaks. Subtracting that line makes "valleys" of the spectrum rest at the zero line. This step is usually done by Ciphergen software, but codes to perform it are also available in PROcess library[2] and Cromwell package[3]. PROcess library bslnoff function divides spectrum into number of unequal sections, finds a minimum (or a quantile corresponding to given probability) of each section, replaces each intensity by that minimum and fits a smooth curve through all

the points. Cromwell's waveletSmoothAndBaselineCorrect function uses wavelets to smooth the spectrum and than uses cumulative (monotone) minimum as a baseline.

The second step of preprocessing is normalizing the multiple spectra. It usually involves cutting first low mass spectra where there is a lot of high frequency high volume noise, which can skew normalization. Afterwards, one finds mean intensity of each spectrum and scales all spectra in such a way as to match all mean intensities. Other ways of normalizing the data also exist for example Petricoin/Liotta study normalized the data by matching minimum and maximum of each signature [4][5][6][7].

At this stage an optional step of mass drift adjustment can be performed. Mass drift adjustment attempts to shift the whole spectrum one or more time steps forward or backward if that is going to improve that spectrum correlation with other samples. This step is especially useful in case of multiple copies of the same sample, which should have very high correlation. The process is done in the following way:

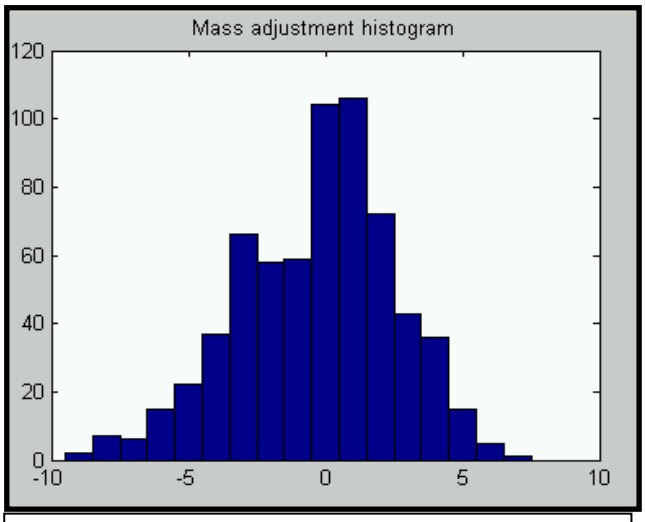

Figure 4: Example of a histogram of shifts to the right (+) or left (-) calculated during mass drift adiustment

1. First we extract peak regions from all the spectra. That is done by finding a mean spectrum and identifying peak regions as the ones where mean spectrum is above average (in Matlab: peaks = S(:,mean(S,1)>mean(S(:))).
2. Then we create procedure for matching 2 spectra. First spectrum is not moved and the second is shifted one time step to the right or to the left as long as the correlation between two spectra improves
3. Finally we use the above procedure: first to match all spectra to their copies (if present) and than to match each spectrum to mean spectrum. Since mean spectrum will be changing due to those shifts, the procedure will probably have to be done two or three times before stabilizing. Most of the shifts are assumed to be a few time steps.

I did not found any references to other teams using mass drift adjustment, but it does seem to improve quality of the data.

The next step would be peak finding and alignment in order to find biomarkers. However it seems like at this stage there are two major different approaches related to SELDI data classification: some research teams use peak alignment to reduce size of the data before classification [8][9][6] and other teams apply classification techniques to the raw data [4] . If the first approach is used than the steps are as follows:

1. in each spectrum separately peaks are detected using variety of different methods [1][2][3][6]
2. peaks from different spectra are aligned into single matrix [11][8][2][3] (see figure 4)

If the second approach is taken than we skip the above 2 steps and use feature selection approach to lower dimensionality of the data. This step however requires use of class labels and by definition is not a part of pre-processing.

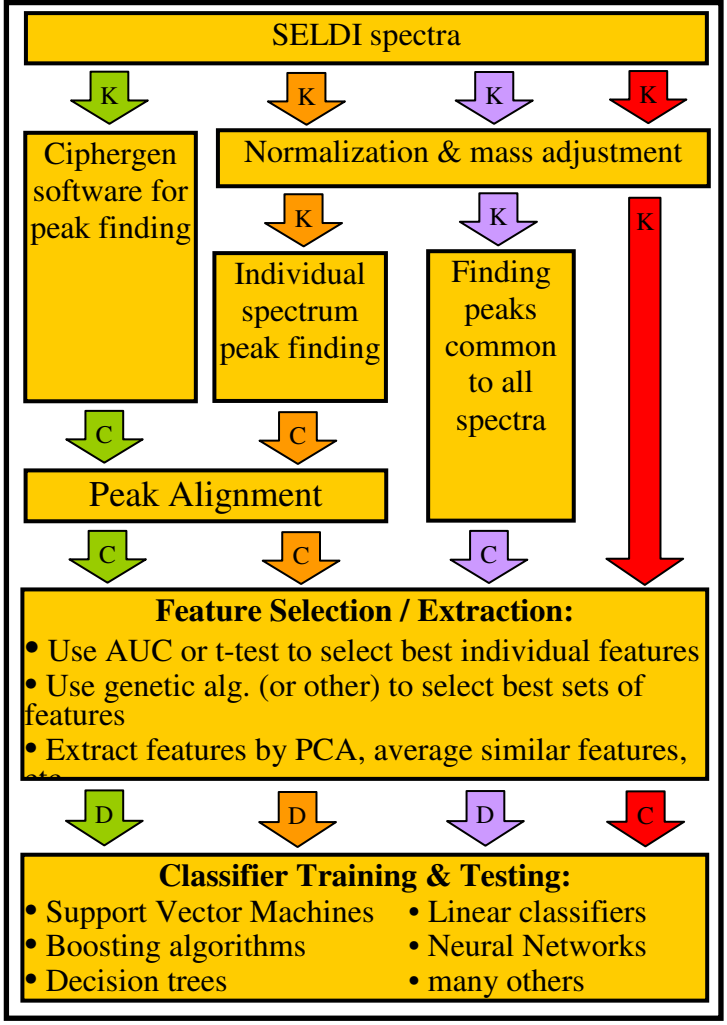

Figure 5: Algorithm families for classification of SELDI signatures show different approaches used by different teams: green: EVMS [8] & CPDR [9], brown: Baggerly [6], purple: an algorithm which combines peak finding and peak alignment, red - Pettricoin/Liotta **[4].**

The final pre-processing step is to merge multiple copies of each sample that could have been provided into single uniform set of features associated with each sample. If only a single copy is collected of each sample than this step should be skipped. There are two types of sample copies:

1. Equivalent copies that were taken under the same conditions and should be identical [8]
2. Copies are taken under different conditions (different chips, different hardware settings, etc.) are assumed to be different. [9]

In order to merge the equivalent copies one can:

1. Average them in order to get a signature with much better signal to noise ratio. That is especially true for the test set.
2. If more than 2 copies are collected than one can average only two (or more) copies that are most similar to each other and discard the outliers.
3. Even with 2 copies one can average the copies if they are highly correlated to each other and discard one if they are not. The discarded copy should be the one that resembles the least other samples.

4. In case of the train set one can also keep all the copies and treat them as separate training samples. That choice gives smaller sample purity, but creates larger train set.
5. Another possibility is to keep all the samples and their averages for the even larger number of train samples.

In case of copies taken under different conditions there seem to be only one merging strategy: merge them end-to-end creating samples with twice the number of features.

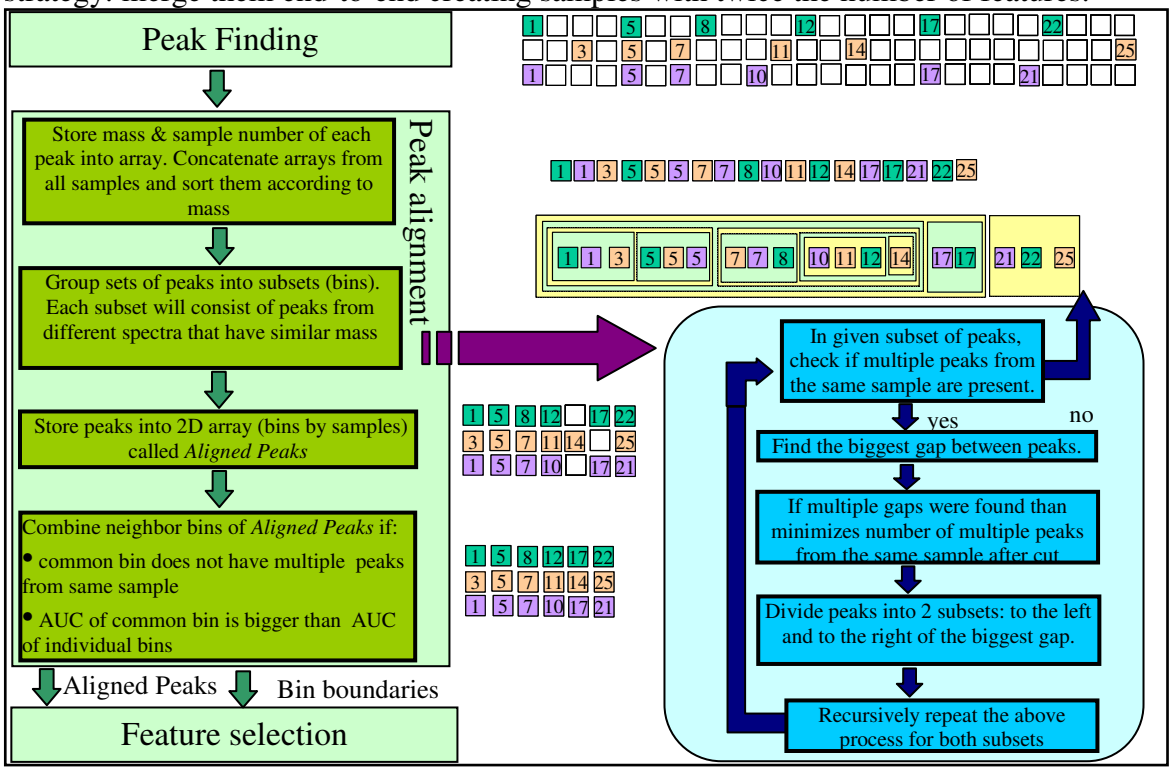

Figure 2: Peak alignment algorithm that follows method from [11].

Building classifiers

There are many classifiers that can be used. So the main purpose of this section is to provide framework to compare them to each other in order to choose the best one. The process is called cross-validation [14] and the general steps are as follow:

- For each classification method:
  - Repeat multiple times (10s – 100s)
    - Split train set of labeled samples into 2 groups: temporary train and test sets
    - Train each classifier on train set and test it on the test set
  - Collect statistics on each classification method: mean and variance of accuracy, or sensitivity/specificity
- Choose classification method with the best performance
- Apply this method to the full set of labeled samples

In this framework the step of training each classifier could be preceded or combined with feature selection, for example:

1. In order of dropping dimensionality of the data one can use t-test or Wilcoxon-test (equivalent to area under ROC curve) to rank each feature according to its individual strength of separating the data into two or more classes. Features with rank below certain threshold could be eliminated. [7][8]
2. Another approach is to look for very similar neighbor features (with high correlation between them) and keep only one of them – the one with higher separation capabilities, as measured by Wilcoxon-test. That approach is especially useful if no peak finding is used during preprocessing.
3. Feature selection can be also performed with goal of finding a good set of features instead of sets of good features. That is more time consuming approach but with potential high rewards. For example exhaustive search [6][7], genetic programming [4], or other methods, can be used to find the best set of features according to some criteria (statistical distance, accuracy of classifiers that could be build using those features (see figure 2), etc.)

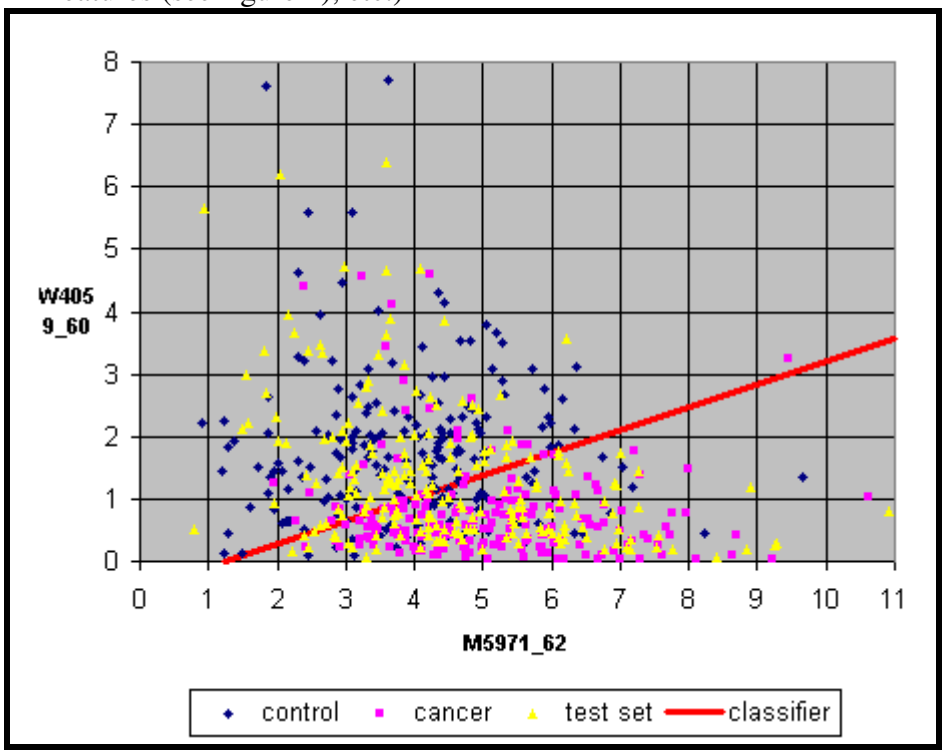

Figure 3: Example of feature selection. Among 95 features 2 were found that allow separation of 2 clusters with average accuracy of 78%, in this case test set was also predicted with 79% accuracy.

The classifiers particularly useful for working on SELDI classification problem have to be able to work with data sets that have usually much more features than samples. I have the best results with:

1. Fisher linear classifier when combined with feature selection [6][7]
2. Support vector machine classifiers

3. Neural network classifiers (we got the best results with a feed-forward neural network classifier with back-propagation)
4. Decision tree classifiers[8][9]
5. Boosting methods based on decision stump classifiers (AdaBoost[13], LogitBoost[12])

Pettricoin/Liotta team [4][21] also reported good results with self organizing maps (SOM) approach.

Table 1: Error rates using different approaches on different data sets. Some classification is between cancer and normal samples, other is between different stages of cancer.

	Peak finding by Ciphergen software & alignment	**Individual peak finding & alignment**	**Common peak finding**	**Selection of smoothed features without peak finding**	**Feature Selection without peak finding**
CPDR-1 Cancer/Normal	23±7% -my peak finding 17±6% - peak finding by CPDR	~25%		~25%	
CPDR-2 Cancer/Cancer		Very poor results	Very poor results	Very poor results	
CPDR-3 Cancer/Normal			48%	Very poor results	7% (but ~50% in blind test)
CPDR-3a Cancer/Normal	~20% in blind test	33±10%	30±8%		6% (20% in blind test)
CPDR-4 Cancer/Normal	21% in blind Cancer/ normal test				~30% in cross validation of cancer/ cancer
EVMS-VPC Cancer/Benign/Normal	2.5±2 % (Peak finding and alignment done by EVMS)	~25%	19.2±5%	10 feat. & ldc - 18.3±4% 200 feat. & svm - 24±4%	10 feat. & ldc - 19.3±4% 200 feat. & svm - 28±6%
NCI-Prostate Data Cancer/Benign/Normal	No data in Ciphergen format	17 ±5%	13±4%	10 feat. & ldc - 12±4% 200 feat. & svm - 15±4%	9 feat. & ldc - 13±4% 200 feat. & svm - 19±4%
NCI-Ovarian Data Set I (Lancet set)	No data in Ciphergen format	Poor results	Poor results	~20%	0% - Benign/Normal & Cancer

Cancer/Benign/Normal					using Eigen vectors; Normal/Cancer 5 feat. & ldc - 8±4%
NCI-Ovarian Data Set II Cancer/Benign/Normal	No data in Ciphergen format				0.05% error can be achieved with 3 point linear classifier
NCI-Ovarian Data Set III Cancer/Normal	No data in Ciphergen format	Impossible because data was not base-line corrected	Impossible because data was not base-line corrected		0% - using 2 sets of 2 features and linear classifier

2.4 Concerns

In SELDI spectra classification problem there are two major potential problems inherit to the process:

- Low data reproducibility – SELDI instruments are very sensitive to minute changes in machine settings and sample preparation protocol. As a result, two machines or the same machine at different times will likely give the different results for the same samples. That, among other problems, creates a potential for introducing bias into the data samples. For example if cancer samples are processed in a separate batch from normal samples, than there is a potential that the best classifier found to distinguish between them will not relay on biological differences but rather on differences in machine settings. That or similar scenario possibly happen to one of ovarian cancer datasets listed in [5] as suggested by [6][7]. Another example would be CPDR-3 data set where train dataset was collected at different time than test dataset. As a result my best classifiers which had 7% error rate during cross validation procedure had close to 50% error rate in during blind test.
- Possibility of false discovery – is a second potential problem, which is shared with other types of data that have much larger number of features (10 000s) than samples (100s), for example microarrays [22]. Many of the standard classification algorithms are so good at what they are doing, that they can find patterns even in the random data. For example if we use exhaustive search to find the best two-feature linear classifier (see figure 5) using data with N=50 000 features, than theoretically we have $N^2/2 = 10^9$ different data sets to evaluate, so even very unlikely events with probability of 1 in 10^9 should happen once. Also, the smaller number of samples the higher the chance that random data will arrange itself into desired pattern.

2.5 Conclusions and Recommendations

Our main conclusion is that there is no single best approach for classifying SELDI data, but rather several competing algorithms that have to be tried in order to find the optimal one. The choice of the algorithm depends on many factors, some of them listed in the next section. My recommendation would be to implement multiple algorithms for every step of data processing and implement them as a single cohesive R library that would provide a common interface to allow researchers to experiment with different method.

The Methods sections list a multitude of different approaches and algorithms. Some of the factors that could affect choice of the algorithm:

1. Data size vs. time and memory available – some methods are more appropriate for smaller data sets since they take too long to process, also some algorithms take too much memory. My machine has 1.5 GB and it is no uncommon for me to run out of memory on larger data sets.
2. Source of data and stage of processing – when working with data posted on the web by different research teams, one does not have a choice of the level of preprocessing done on the data. For example some data sets will be baseline subtracted and normalized other will be raw, yet another set will contain only extracted biomarkers. So the pre-processing steps to be chosen will have to match the data itself.
3. Number of copies collected – the choice of data merging techniques performed on the end of preprocessing will mostly depend on number and type of copies available.
4. Number of different categories – there is often a difference between two-way and multiple way classification, since some classification algorithms do not always support more than two classes. For example if one performs cancer/non-cancer classification than his choice of classifiers might be different than if one performs prostate cancer/non-cancer/BPH (benign prostatic hyperplasia) classification, and different again if one wants to study distinguishing features between four stages of cancer.
5. Purpose of the study – the choice of the classifier could be different if the final goal of the study is to find the best possible classifier vs. to find the best classifier as a way of identifying a limited set of features with the greatest power of distinguishing between multiple classes. One might want to find distinguishing features, since they have to correspond to different protein which we might want to identify. For example: decision tree, boosting and some feature selection algorithms work by finding limited sub-set of features and operating only on those, while neural networks, SVM, fisher and many other algorithms always use all the features provided. Because of that, the first set of algorithms gives you two results: a classifier and best set of features, while the second gives you only the classifier
6. Availability of different algorithms in any particular language – software and algorithms for SELDI data processing [1][2] [3], mass spectrometry data processing and classification [15-20] are available in R, Matlab, and C (C++) codes. This project is to be written in R language, what means that some libraries are available, while other will have to be rewritten.

The final library would follow the basic course of operation that contains following steps:

1. User inputs Process Parameters, which will uniquely describe the rest of the flow. The parameters are saved into Parameter Store, which will be retrieved by remaining processes.
2. Data is pre-processed according to user specifications retrieved from Parameter Store, and then stored in Pre-processed Data Store
3. Classifiers are built using pre-processed data and class labels. The steps of the process are specified by Parameter Store.
4. Classifier is verified by a User or applied by a Clinical Manager. That is done by running the classifier on unlabeled pre-processed data in order to predict the class labels.

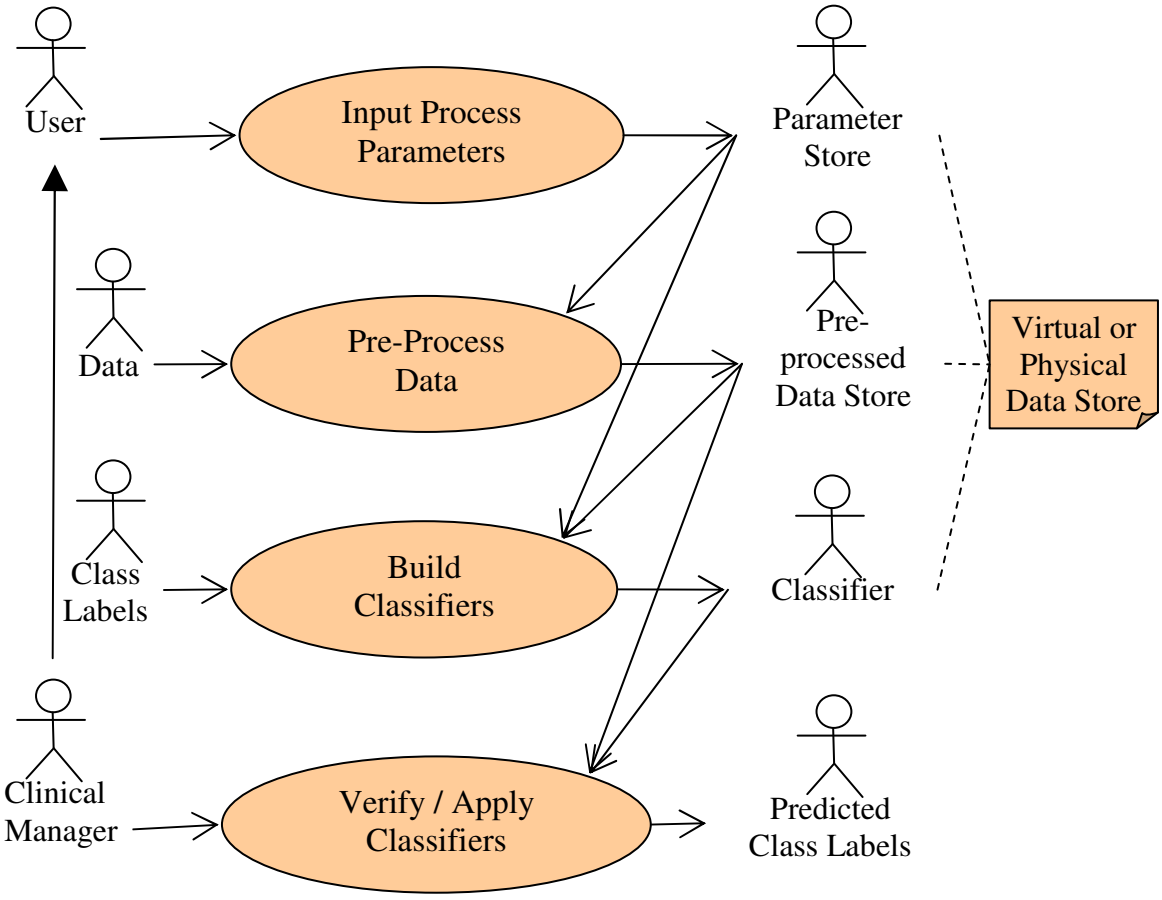

Figure 4: Basic course of *Pattern Recognition for Diagnosis and Treatment* use case

The above steps are common to all of described classification algorithms, and the choice of the actual algorithm will have to be saved in Parameter Store.

References

[1] Cyphergen's ProteinChip Software 3.0 User Manual.

[2] PROcess R library by Xiaochun Li
http://bioconductor.org/repository/devel/package/Source/PROcess_0.9.tar.gz

[3] University of Texas - M.D. Anderson Cancer Center – Cromwell Matlab package
http://bioinformatics.mdanderson.org/cromwell.html

[4] Petricoin EF, Ardekani AM, Hitt BA, Levine PJ, Fusaro VA, Steinberg SM, Mills GB, Simone C, Fishman DA, Kohn EC, Liotta LA: Use of proteomic patterns in serum to identify ovarian cancer. Lancet 2002, 359:572-577.

[5] Clinical Proteomics Program Databank website at www.ncifdaproteomics.com (unfortunately this website changes often).

[6] Baggerly KA, Morris JS, Coombes KR. Reproducibility of SELDI-TOF protein patterns in serum: comparing data sets from different experiments. Bioinformatics. 2004 Jan 29.

[7] Sorace JM, and Zhan, M. A data review and re-assessment of ovarian cancer serum proteomic profiling BMC Bioinformatics 2003, 4:24.

[8] Bao-Ling Adam, Yinsheng Qu, John W. Davis, Michael D. Ward, Mary Ann Clements, Lisa H. Cazares, O. John Semmes, Paul F. Schellhammer, Yutaka Yasui, Ziding Feng, and George L. Wright, Jr.. Serum Protein Fingerprinting Coupled with a Pattern-matching Algorithm Distinguishes Prostate Cancer from Benign Prostate Hyperplasia and Healthy Men Cancer Res 2002 62: 3609-3614.

[9] Bañez LL, Prasanna P, Sun L, Ali A, Zou Z, Adam B-L, McLeod DG, Moul JW and Srivastava S: Diagnostic Potential of Serum Proteomic Patterns in Prostate Cancer. J. Urol. (in press), 2003.

[10] Virginia Medical School – Virginia Prostate Center - Overview of the SELDI System; http://www.evms.edu/vpc/seldi/seldiprocess/index.html .

[11] Virginia Medical School – Virginia Prostate Center - Overview of the PeakMiner Software http://www.evms.edu/vpc/seldi/peakminer.pdf .

[12] M. Dettling & P. Bühlmann; Boosting for Tumor Classification with Gene Expression Data; Bioinformatics, June 12, 2003

[13] Yinsheng Qu, Bao-Ling Adam, Yutaka Yasui, Michael D. Ward, Lisa H. Cazares, Paul F. Schellhammer, Ziding Feng, O. John Semmes, and George L. Wright, Jr.;Boosted Decision Tree Analysis of Surface-enhanced Laser Desorption/Ionization Mass Spectral

Serum Profiles Discriminates Prostate Cancer from Noncancer Patients; Clin Chem 2002 48: 1835-1843.

[14] Wagner M, Naik DN, Pothen A, Kasukurti S, Devineni RR, Adam BL, Semmes OJ., Wright GL; Computational protein biomarker prediction: a case study for prostate cancer BMC Bioinformatics 2004, 5:26 (11 March 2004)

[15] SOM (self organizing maps) Toolbox, Matlab - http://www.cis.hut.fi/projects/somtoolbox/

[16] PRTools; pattern recognition and classification toolbox, Matlab - http://www.ph.tn.tudelft.nl/~bob/PRTOOLS.html , http://prtools.org/prtools.html

[17] rpart; Recursive partitioning and regression tree package; R - http://cran.r-project.org/src/contrib/Descriptions/rpart.html

[18] nnet, neural networks; R - http://www.math.mcgill.ca/sysdocs/R/library/nnet/html/nnet.html

[19] svm, support vector machines, R - http://www.maths.lth.se/help/R/.R/library/e1071/html/svm.html

[20] MATLAB Support Vector Machine Toolboxes - http://theoval.sys.uea.ac.uk/~gcc/svm/toolbox/ , http://asi.insa-rouen.fr/~arakotom/toolbox/index , http://www.ece.osu.edu/~maj/osu_svm/

[21] Liotta LA, Petricoin E. SELDI-TOF-based serum proteomics pattern diagnostics for early detection of cancer. Current Opinion in Biotechnology 2004, **15**:24-30

[22] Downey, Tom; With Microarrays, Pitfalls of false discovery; Genome Technology; 01/2003

[23] David G. Stork and Elad Yom-Tov; Computer Manual in MATLAB to accompany Pattern Classification; Wiley Interscience; ISBN: 0-471-42977-5