Improved Mass Spectrometry Peak Intensity Prediction by Adaptive Feature Weighting

Alexandra Scherbart[1], Wiebke Timm[1,2], Sebastian Böcker[3], and Tim W. Nattkemper[1]

[1] Biodata Mining & Applied Neuroinformatics Group, Faculty of Technology, Bielefeld University
{ascherba,wtimm,tnattkem}@techfak.uni-bielefeld.de
[2] Intl. NRW Grad. School of Bioinformatics & Genome Research, Bielefeld University
[3] Bioinformatics Group, Jena University
boecker@minet.uni-jena.de

Abstract. Mass spectrometry (MS) is a key technique for the analysis and identification of proteins. A prediction of spectrum peak intensities from pre computed molecular features would pave the way to a better understanding of spectrometry data and improved spectrum evaluation. The goal is to model the relationship between peptides and peptide peak heights in MALDI-TOF mass spectra, only using the peptide's sequence information and the chemical properties. To cope with this high dimensional data, we propose a regression based combination of feature weightings and a linear predictor to focus on relevant features. This offers simpler models, scalability, and better generalization. We show that the overall performance utilizing the estimation of feature relevance and re-training compared to using the entire feature space can be improved.

1 Introduction

Mass spectrometry (MS) is a key technique for the analysis and identification of proteins and peptides. Matrix-assisted laser desorption ionization (MALDI) is one of the most often used technique for the analysis of whole cell proteomes in high-throughput experiments. In quantitative proteomics, approaches are desired to allow a comparison of protein abundances in cells. Proteins are quantitatively characterized in a complex sample or protein abundances across different samples are compared. Opposed to labeling methods, label-free methods directly use peak heights (referred to as intensities) to estimate peptide abundances. There are different applications of MALDI-MS where the prediction of peak intensities in spectra are needed for further improvements. The identification is commonly done by comparing the peak's masses from a spectrum (protein mass finger print, PMF), to theoretical PMFs in a data base, generating a score for each comparison. Different tools are available for this purpose. For an overview see [1]. These tools rarely use peak intensities, because there is no model to calculate the theoretical PMFs directly. The use of peak intensities could improve the reliability of protein identification without lowering the error rate, as was shown by [2] for tandem MS.

M. Köppen et al. (Eds.): ICONIP 2008, Part I, LNCS 5506, pp. 513–520, 2009.

For the prediction of MALDI PMF there has been one study so far by [3] who applied different regression and classification algorithms. [4] used multi-layer neural networks to predict peptide detectabilities (i.e. the frequency with which peaks occur in spectra) in LC/MS ion trap spectra which is a related problem.

An algorithmic approach for peak intensity prediction is a non-trivial task because of several obstacles: The extraction of PMF from spectra is a signal processing task which can not be done perfectly. Data from this domain is always very noisy and contains errors introduced by preprocessing steps in the wet lab as well as in signal processing. Misidentifications may even lead to wrong sequences. Intensity values can be distorted due to the unknown scale of spectra. It is nearly impossible to come by a large enough dataset from real proteins where the content is known, i.e. there is no perfect gold standard, because of the not reproducible and non-unique peptide/intensity relation.

To model the relationship of peptide sequences and the peak intensities, as regression task, numerical feature vectors have to be calculated building the feature space as input for the learning architectures. Intensities for different peptide sequences differ even if they have the same abundance, because of their different chemical properties. We propose to use chemical properties of the peptides derived from the peptide sequences. Since these high-dimensional vectors are supposed to contain redundant and interdependent information, a method is needed, that can cope with this data and is able to reduce dimensionality of feature space. The task of finding a suitable subset of features is well-known as feature selection and is one of the central problems in machine learning [5]. Focusing on relevant features that contribute to model the peptide/peak intensity relation offers simpler models, scalability, and better generalization [6]. Additionally, it reduces computational costs.

To overcome these obstacles to predict peak intensities in MALDI-TOF spectra based on a training set of peptide/peak intensity pairs, we considered an artificial neural net architecture, namely the Local Linear Map [7], since it combines unsupervised (a) and supervised (b) learning principles, with comparable results to those obtained by ν-Support Vector Regression (SVR) [8]. The LLM can learn global non-linear regression functions by fitting a set of *local* linear functions to the training data. It is very efficient (i.e. fast training and adaptation to addition data, and low memory-usage) and offers transparency. Other than for example SVR it can be used for data mining once adapted in a straight forward manner. Both architectures have been proposed to model the non-linear relationship between peptide and peak intensities. In previous experiments, we have modelled the relationship between with a positive correlation of $r^2 = 0.46$. But the reduction to a set of 18 features chosen in an ad-hoc forward feature selection brought no significant performance gain.

In this paper, the regression approach is extended by combining feature weightings with a linear predictor. We propose a two-step regression approach, which includes adaptive regression based feature weighting by m learning architectures individually, to ensure different kind of regression behavior and abstraction levels. Subsequently, the estimated feature weightings of d features are

used for scaling the input vectors in a retraining step with an independent predictor. The performance when utilizing a weighting of features and subsequent re-training is compared to an approach integrating weighting and filtering.

2 Materials and Methods

2.1 Data

In this study we use two datasets **A** and **B** of peptides of MALDI mass spectra. The first one, **A**, consists of 66 spectra of 29 different proteins, with 16 of these proteins being present in multiple spectra, whereas **B** consists of 200 spectra of 137 different proteins with 39 of these proteins occurring multiple times.

Peak extraction steps include soft filtering, baseline correction, peak picking and isotopic deconvolution in the corresponding raw spectra. The resulting list of peaks is matched against masses derived from a theoretical tryptic digestion. These steps for **A** (and **B** respectively) result in 857 (1631) matched peaks corresponding to 415 (1135) different peptides. In the remainder, the intensities refer to scaling the original intensity by "mean corrected ion current", where I_p^{orig} is the original intensity after peak extraction, I_i is the raw value at index i, B_i the baseline, and N_i the noise determined in the denoising step: $I_p^M = \ln\left(\frac{I_p^{\text{orig}}}{\sum_{i=1}^{N} I_i - B_i - N_i} + 1\right)$.

Feature Sets. We derive numerical representations of the peptides from the peptide sequences to contribute different properties of peptides. The feature vectors are built by amino acid frequencies (20 monomers, *mono*), typically used in bioinformatics, and by physico-chemical information about the amino acids constituting the peptide. The latter feature vectors are attributes taken from the amino acid index database [9](*aaindex*) extended by peptide length, mass, and numbers and fractions of acidic, basic, polar, aliphatic and arginine residues, yielding 531-dimensional vectors.

Most of the peptides in the dataset occur multiple times in different spectra with different intensity values. Due to limitation of training data, we eliminate outliers (potential noisy peptides) by mapping each peptide to one unique value, the α-trimmed mean of all intensities per distinct peptide with $\alpha = 50\%$. The α-trimmed mean is defined as the mean of the center 50% of an ordered list. In the case of less than 4 peptides in the list a simple mean is taken.

2.2 Feature Selection/Weighting

Classic feature selection steps include heuristic search (forward selection, backward elimination, stepwise selection) and successively adding or eliminating attributes by an adequate strategy. The approaches can be divided into filters and wrappers. Filter approaches use general characteristics (e.g. correlation) of the data provided to select a subset of features, independently of the chosen learner. Wrappers score subsets of features by a metric according to the estimated accuracy of a given learning machine.

To reflect feature importance as a numerical value, an estimation of the feature relevance is done by different regression models which are trained on entire feature space. Each of the applied architectures comprises an internal model dependent metric as measure of quality such that the feature weights are set proportionally to assessed change in accuracy (via correlation, MSE) e.g. to decrease in error when permuting features. The greater the decrease in performance when leaving a certain feature out, the higher is the assigned degree of relevance of the feature. The model dependent metrics evaluation as offered by the R [10] package [11] and yield m estimations interpreted as feature weightings $Z \in \mathbb{R}^{m \times d}$ and are calculated according to:

- Linear Models (**LM**): absolute value of t-statistic for each model parameter.
- Random Forest (**RF**): [12]: "the average increase in squared OOB (out-of-bag) residuals when the variable is permuted".
- Partial Least Squares (**PLS**): weights are proportionally to decrease in the sums of squares.
- Bagged Trees (**BT**): total importance over all bootstrapped trees.
- Boosted Trees (**GBM**): total sum of importance over all boosting iterations.

2.3 Local Linear Map

Motivated by the Self-Organizing Maps (SOM) [13], an LLM consists of a set of n_l regular ordered nodes $\mathbf{v}_i, i = 1, \ldots, n_l$, which are connected to each other via a two-dimensional grid structure, defining a neighborhood between the nodes and a topology in feature space. Each node consists of a triple $\mathbf{v}_i = (\mathbf{w}_i^{\text{in}}, \mathbf{w}_i^{\text{out}}, \mathbf{A}_i)$. The vectors $\mathbf{w}_i^{\text{in}} \in \mathbb{R}^{d_{\text{in}}}$ are used to build prototype vectors adapting to the statistical properties of the input data $\mathbf{x}_\xi \in \mathbb{R}^{d_{\text{in}}}$. The vectors $\mathbf{w}_i^{\text{out}} \in \mathbb{R}^{d_{\text{out}}}$ approximate the distribution of the target values $\mathbf{y}_\xi \in \mathbb{R}^{d_{\text{out}}}$. The matrices $\mathbf{A}_i \in \mathbb{R}^{d_{\text{in}} \times d_{\text{out}}}$ are locally trained linear maps from the input to the output space. In the unsupervised training phase, the prototype vectors $\mathbf{w}_i^{\text{in}}$ are adapted following the SOM learning rule: the vectors $\mathbf{w}_i^{\text{in}}$ are pulled towards the input pattern $\mathbf{x}_\xi$ according to the distance between the input pattern and the corresponding closest prototype in input space $\mathbf{w}_\kappa^{\text{in}}$, with $\kappa = \underset{i}{\operatorname{argmin}} \left\{ \|\mathbf{x}_\xi - \mathbf{w}_i^{\text{in}}\| \right\}$. After unsupervised adaptation and tessellation of the input space, an input feature vector is mapped to an output by the corresponding local expert: $\mathcal{C}(\mathbf{x}) = \mathbf{w}_\kappa^{\text{out}} + \mathbf{A}_\kappa \left(\mathbf{x}_\xi - \mathbf{w}_\kappa^{\text{in}} \right)$ .The weights $\mathbf{w}_i^{\text{out}}$ and the linear map $\mathbf{A}_i$ are changed iteratively by the gradient descent learning rules.

The concept of approximating nonlinear functions by fitting simple models to localized subsets of the data is related to other regression approaches like Locally-Weighted Regression [14] and to radial basis functions [15].

2.4 Evaluation

Performance assessment is done by 10-fold cross-validation (CV) for each regression model. It was ensured that peptides from one spectrum as well as peptides occurring in more than one spectrum are found in only one of the portions.

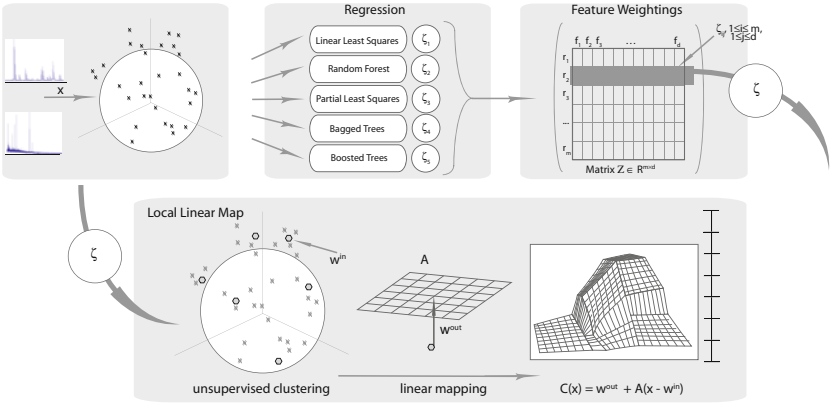

Fig. 1. Graph depicting the proposed architecture

Model selection: Grid search over the parameter space is performed to determine optimal parameters for learning. For every point in the parameter space the prediction accuracy for every training/test set is determined by squared pearson-correlation coefficient r^2 and root mean square error of the test set. The choice of the best parameter set is made by the best mean r^2 over all 10 test sets.

Model assessment: The final model is built by retraining the whole dataset with the optimal parameters chosen in the previous step. To validate its generalization error on new data, the other dataset respectively is used.

Feature Weighting: The final models $\{\mathbf{LM}, \mathbf{RF}, \mathbf{PLS}, \mathbf{BT}, \mathbf{GBM}\}$ are evaluated by the corresponding model specific metric for estimating the contribution of each feature.

Filtering of Feature Weightings: A filtering approach discards feature weightings below a certain threshold. The cut-off value is determined by the $t\%$ most important features (AUC), where $t \in \{70, 90, 95\}$ and hence these features are chosen $\max_i \arg T_i \leq t$ for a fixed method k and $\omega_{k\cdot}$ sorted in descending numerical order, where $T_j = \sum_{i=1}^{j} \omega_{ki}, 1 \leq j \leq d$.

Retraining of LLM model: These feature weightings Z (scaled up to be in range of $[0, 1]$) are then used to scale the input variables to $\tilde{\mathbf{x}} = \mathbf{w} \cdot \mathbf{x}$. A new model of LLM-type is built based on the optimal parameters determined in the model selection step with the input vectors $\tilde{\mathbf{x}}$.

3 Results

In this study, we compare the performance of a predictor when applying different approaches of weighting and filtering of features. The feature importance or degree of feature relevance is estimated by 5 regression architectures. We evaluate the peak intensity prediction on two different datasets $(\mathbf{A}, \mathbf{B})$ for two

different feature sets (**aaindex** (531-dimensional vectors) and **mono** (20-dim.)). To make the results comparable, exactly one predictor is used as reference learning architecture, namely the LLM, for retraining on the derived feature space. Performance and generalization performance is assessed via across dataset prediction as described in 2.4. The presented results are restricted to these of dataset **A**, trained, parameter-tuned (CV) and validated on **B** (GV).

3.1 Non-weighted Feature Space

The regression models of the learning architectures are evaluated in terms of prediction and generalization performance (r^2) as well as their estimation of the contribution of features to the corresponding final model taking the entire, non-weighted feature space as input. The reference results are summarized in Tab. 1 for the **aaindex** and the **mono** feature space and the regression architectures **LM**, **LLM** and **SVR** given in the last three columns.

It can be observed that the applied learning architectures differ strongly in their performance. This trade-off may be due to two main reasons: First of all, the number of variables (531) even exceeds the number of available peptides in the dataset (415). Best prediction and generalization performance is observed for the **SVR**, while the **LLM** shows only slight worse accuracy regarding *mono* feature space. The linear model (**LM**) shows a clear overfitting and lacks of generalization on new peptides across datasets. A comparison between the two peptide features representations shows better performance for the small **mono** feature space in general. These observations suggest a non-linear relationship between peptide feature vectors and the intensity target values.

3.2 Weighted Feature Space

We utilize the estimated contributions of features from the model specific metrics and perform a retraining of the LLM models by replacing the input variables: $\tilde{\mathbf{x}} = \mathbf{w} \cdot \mathbf{x}$. The model of LLM-type is rebuilt based on the optimal parameters determined in the model selection step with the input vectors $\tilde{\mathbf{x}}$.

A comparison in Tab. 1 of the resulting performances when introducing a feature weighting shows clearly the increase in accuracy for all derived estimations of variable importance applied to the **LLM** and *aaindex* feature space.

While an improvement of prediction performance can be observed for the **LLM**. compared to standard **LLM** trained on the entire *aaindex* feature space in all cases, the same holds for generalization on new data, despite of $\mathbf{LLM_{LM}}$ and $\mathbf{LLM_{RF}}$ models. Though using a linear predictor as a feature weighting method, the $\mathbf{LLM_{LM}}$ keeps up to extract and separate the underlying characteristics of the peptides in CV. The feature weighted-LLM trained on the high-dimensional *aaindex* feature space beats the non-weighted models of **LLM**-type trained on *aaindex* feature space as well as for the *mono* feature space in general. It outperforms the SVR in terms of prediction performance and it is only slightly worse in generalization case. The $\mathbf{LLM_{RF}}$ and $\mathbf{LLM_{PLS}}$ models trained on *aaindex* feature space yield the major improvement of prediction performance.

Table 1. Prediction accuracy of regression models of **LLM**-type - retrained on **aaindex** feature space (531 dim) incorporating input vectors scaled by feature weightings. The three last columns are given as reference accuracy in feature spaces without scaling. Results are also given for the *mono* feature space (20 dim).

aaindex	$\mathbf{LLM_{LM}}$	$\mathbf{LLM_{GBM}}$	$\mathbf{LLM_{RF}}$	$\mathbf{LLM_{PLS}}$	$\mathbf{LLM_{TB}}$	**LLM**	**SVR**	**LM**
CV	0.33	0.48	0.46	0.47	0.49	0.27	0.44	0.36
GV	0.18	0.26	0.01	0.39	0.35	0.27	0.42	0.02
mono	$\mathbf{LLM_{LM}}$	$\mathbf{LLM_{GBM}}$	$\mathbf{LLM_{RF}}$	$\mathbf{LLM_{PLS}}$	$\mathbf{LLM_{TB}}$	**LLM**	**SVR**	**LM**
CV	0.42	0.46	0.44	0.42	0.45	0.41	0.46	0.27
GV	0.29	0.30	0.31	0.30	0.30	0.33	0.44	0.20

3.3 Weighted and Filtered Feature Space

There is a small number of features considered as highly relevant, while a few features were estimated to contribute to the predictor model to very low degree. Hence, they might be overweighted or this is due to the number of other features slightly correlated to each other. It seems reasonable to apply a filtering on the resulting estimated feature weightings prior to retraining the **LLM**. A filtering approach following discards feature weightings below a certain threshold. The cut-off value is determined by the $t\%$ most important features (AUC).

With a filtering of the feature weightings the performance is not increased for all applied feature weighting methods. With a rising filtering threshold, prediction performance keeps constant, while the generalization performance decreases. The corresponding results are given exemplarily regarding the $\mathbf{LLM_{TB}}$ for the different values of in terms of CV $\{0.49, 0.47, 0.46\}$ and in terms of GC $\{0.31, 0.3, 0.3\}$. The corresponding number of features excluded were $\{225, 274, 384\}$. The precedent filtering of feature weights leads to a slight worse prediction performance of the subsequent applied **LLM**.

4 Conclusions

The learning architectures of Local Linear Map-type [16] and ν-Support Vector Regression (SVR) [8] have been proposed to model the non-linear relationship between peptide and peptide peak heights in MALDI-TOF mass spectra. High-dimensional numerical feature vectors are derived from the peptide sequence building the feature space as input for the learning architectures. These features are supposed to differ in relevance. For this purpose, we focus on the issue of using relevant features in modelling the non-linear relationship between peptides and peptide peak heights. The regression architecture of **LLM**-type is extended by assigning features degrees of relevance according to their estimated contributions to different predictor models. We propose a regression based combination of estimated feature weightings and a linear predictor offering simpler models, better generalization and reduced computational costs. A comparison between the two peptide feature representations shows better performance for the high-dimensional *aaindex* feature space in general. We got the major improvement in

performance of regression models when retraining with weighted features based on estimated feature relevance by Partial Least Squares and Bagged Trees. These model dependent feature weightings methods perform a skillful scoring of the features in combination with the **LLM**. Though many features were supposed by the applied methods to be relevant to low degree, integrating a filtering of the feature weightings prior to retraining the LLM led to decrease of performance. The most relevant *aaindex* features found amongst others were estimated gas-phase-basicity, fractions of arginine residues, acidic, basic and polar.

References

1. Shadforth, I., Crowther, D., Bessant, C.: Protein and peptide identification algorithms using MS for use in high-throughput, automated pipelines. Proteomics 5(16), 4082–4095 (2005)
2. Elias, J.E., Gibbons, F.D., King, O.D., Roth, F.P., Gygi, S.P.: Intensity-based protein identification by machine learning from a library of tandem mass spectra. Nat. Biotechnol. 22(2), 214–219 (2004)
3. Gay, S., Binz, P.A., Hochstrasser, D.F., Appel, R.D.: Peptide mass fingerprinting peak intensity prediction: extracting knowledge from spectra. Proteomics 2(10), 1374–1391 (2002)
4. Tang, H., et al.: A computational approach toward label-free protein quantification using predicted peptide detectability. Bioinformatics 22(14), 481 (2006)
5. Blum, A., Langley, P.: Selection of relevant features and examples in machine learning. Artificial Intelligence 97(1-2), 245–271 (1997)
6. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. J. Mach. Learn. Res. 3, 1157–1182 (2003)
7. Ritter, H.: Learning with the self-organizing map. In: Kohonen, T., et al. (eds.) Artificial Neural Networks, pp. 379–384. Elsevier Science Publishers, Amsterdam (1991)
8. Timm, W., Böcker, S., Twellmann, T., Nattkemper, T.W.: Peak intensity prediction for pmf mass spectra using support vector regression. In: Proc. of the 7th International FLINS Conference on Applied Artificial Intelligence (2006)
9. Kawashima, S., Ogata, H., Kanehisa, M.: AAindex: Amino Acid Index Database. Nucleic Acids Res. 27(1), 368–369 (1999)
10. R Development Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Stat. Comp., Austria (2008) ISBN 3-900051-07-0
11. Kuhn, M.: caret: Classification and Regression Training, R package v. 3.16 (2008)
12. Liaw, A., Wiener, M.: Classification and regression by randomforest. R News 2(3), 18–22 (2002)
13. Kohonen, T.: Self-organized formation of topologically correct feature maps. In: Biological Cybernetics, vol. 43, pp. 59–69 (1982)
14. Cleveland, W.S., Devlin, S.J.: Locally-weighted regression: An approach to regression analysis by local fitting. J. of the American Stat. Assoc. 83, 596–610 (1988)
15. Millington, P.J., Baker, W.L.: Associative reinforcement learning for optimal control. In: Proc. Conf. on AIAA Guid. Nav. and Cont., vol. 2, pp. 1120–1128 (1990)
16. Scherbart, A., Timm, W., Böcker, S., Nattkemper, T.W.: Som-based peptide prototyping for mass spectrometry peak intensity prediction. In: WSOM 2007 (2007)