Li *et al. BMC Bioinformatics* (2019) 20:549
https://doi.org/10.1186/s12859-019-3170-1

BMC Bioinformatics

METHODOLOGY ARTICLE Open Access

Quality control of imbalanced mass spectra from isotopic labeling experiments

Tianjun Li[1], Long Chen[1*] and Min Gan[2]

Abstract

Background: Mass spectra are usually acquired from the Liquid Chromatography-Mass Spectrometry (LC-MS) analysis for isotope labeled proteomics experiments. In such experiments, the mass profiles of labeled (heavy) and unlabeled (light) peptide pairs are represented by isotope clusters (2D or 3D) that provide valuable information about the studied biological samples in different conditions. The core task of quality control in quantitative LC-MS experiment is to filter out low-quality peptides with questionable profiles. The commonly used methods for this problem are the classification approaches. However, the data imbalance problems in previous control methods are often ignored or mishandled. In this study, we introduced a quality control framework based on the extreme gradient boosting machine (XGBoost), and carefully addressed the imbalanced data problem in this framework.

Results: In the XGBoost based framework, we suggest the application of the Synthetic minority over-sampling technique (SMOTE) to re-balance data and use the balanced data to train the boosted trees as the classifier. Then the classifier is applied to other data for the peptide quality assessment. Experimental results show that our proposed framework increases the reliability of peptide heavy-light ratio estimation significantly.

Conclusions: Our results indicate that this framework is a powerful method for the peptide quality assessment. For the feature extraction part, the extracted ion chromatogram (XIC) based features contribute to the peptide quality assessment. To solve the imbalanced data problem, SMOTE brings a much better classification performance. Finally, the XGBoost is capable for the peptide quality control. Overall, our proposed framework provides reliable results for the further proteomics studies.

Keywords: Mass Spectra, Proteomics, Imbalanced Data, Quality Control, Gradient Boosting

Background

Computational methods in proteomics are mainly designed to improve the analysis performance of MS. There are many well designed methods, like the molecular formulas predicting [1], the linear regression for overlapped $^{18}O/^{16}O$ ratio estimation [2], the statistical methods for corresponding feature identification [3], the self-boosted percolator for peptide prophet enhancing [4, 5] and the peptide identification for mixture spectra [6] to name a few.

In the Stable Isotope Labeling with Amino Acids in Cell Culture (SILAC) based proteomics experiments, the amino acids with *light* and *heavy* labels are metabolized into peptides [7]. Then the identified peptides show only a fixed shift in mass in different conditions of the spectra. However, the pairs of heavy-light peptide and some other features in SILAC data are often influenced by some biological, experimental or chemical errors. Since these errors may affect the later quantitative analysis, and the errors are hard to handle manually, there is a great need to have some computational methods to assess the spectral quality [8].

Some software or platforms have been provided to handle the whole peptide analysis workflow, such as OpenMS [9], MaxQuant [10] and Trans-Proteomic Pipeline (TPP) [11–13]. These software can be used to convert the raw data into the readable files with analysis results. The whole peptide analysis workflow is defined as follows: "raw data converting → sequence database identification → validation → quantification" [14]. Currently, researchers pay more attention to the validation or quality control of

*Correspondence: longchen@um.edu.mo
[1]Department of Computer and Information Science, University of Macau, Taipa, Macau, China
Full list of author information is available at the end of the article

quantification part in the workflow, and many methods have been proposed to this end. For example, the signal-to-noise ratio is proven to be an essential factor in ratio estimations for the isotope labeling based experiments [15, 16]. In addition, the preceding peak is demonstrated to be useful when compared with the target peak in the same scan of the peptide [17]. These signal-to-noise ratios and preceding peak ratios are some mass profiles, and many studies have demonstrated the importance of controlling the quality of mass profiles [16] in quantification analysis.

Many methods have been conducted to control the spectral quality for better quantification results. These methods can be mainly divided into naive methods [18], classification methods [19, 20] and statistical methods [21], while the classification approach is the most widely used ones [19]. For details, some features are extracted from few LC-MS raw data, and then the corresponding quality tags are generated manually. The features and associated tags are formed together into a data set. We then divide the data set into a training set and a validation set, where the training set is used for classifier training and the validation set is used to ensure the classifier's performance. Finally, we can evaluate the quality of other spectra by extracting the same features from related spectra and then passing through the trained classifier. The mass profiles classified as the high quality ones are retained for further analysis. The general diagram for the design of quality control is illustrated in Fig. 1.

There are pairs of heavy and light peptide peak clusters in LC-MS[2]. Most spectral feature extracting methods focus on related clusters in one scan. However, based on the extracted ion chromatogram (XIC), the information in the nearby scans also helps to quantitation [22]. This motivates us to derive four new features from the corresponding neighbor scans to construct the classifier. Combining nine features extracted from single scan, we totally release thirteen features as the inputs of classifiers.

To comprehensively consider all the features related to the quality of quantification, machine learning methods can be used for quality assessment of spectra, including the support vector machine (SVM) based models [19, 20] and the tree-structured models [23–25]. Note that proteomics also has some models based on deep neural network (DNN) [26–28] , but DNN is time-consuming and requires a lot of training data to deal with the overfitting problem, which is not suitable for quantification. So in this paper, we do not discuss deep models.

When training a classification model, by default most machine learning algorithms treat the training set as the balanced one. However, because learning system has difficulty in deriving concepts from the minority class, imbalanced data has become one of the major challenges affecting the performance of machine learning algorithms [29, 30]. The re-sampling method is one of the most important methods for dealing with data imbalance problem. In the field of re-sampling, there are two well-known approaches: the under-sampling one and the over-sampling one. But the under-sampling method may discard potentially relevant information, while the over-sampling method may increase the likelihood of overfitting and the complexity of the model training [31]. Therefore, we need decent ways to deal with unbalanced data problems. SMOTE is one of the mature methods [29]. This method re-samples new data point by combining random factors from zero to one with its k nearest neighbors.

Considering the fact that there are only a few problematic spectra in LC-MS analysis, we carefully addressed the important problem of classifying imbalanced spectral dataset in this paper. We suggest using the SMOTE to handle the unbalanced data, and employing the famous XGBoost[32] as the classifier, which achieves outstanding performance without having high overhead in computation time. XGBoost is a kind of tree-structured model. The basic idea of the tree-structured model is to design an ensemble approach for several rule-based binary trees [33, 34]. In the past decades the *Gradient Boosting* [35] is the most famous tree ensemble method, and this technique led to the renowned Gradient Boosting Decision Tree (GBDT). XGBoost is a variant of GBDT, and it has gained the popularity by winning many machine learning competitions since its availability.

We evaluate the classifiers by the data with different heavy-light ratios. The SMOTE technique shows its capability in improving the performance of classifiers by re-balancing the data, and the SMOTE XGBoost shows

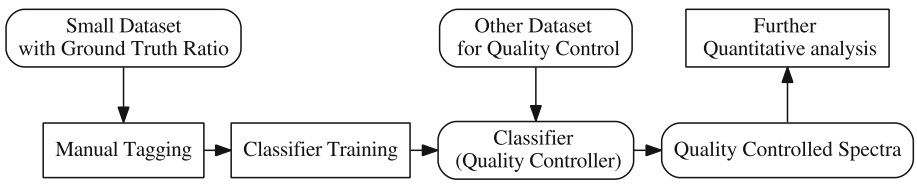

Fig. 1 The design of quality controller. The spectra after peptide quality control can be used for further studies, such as quantification

its reliability in assessing the quality of mass profiles in LC-MS data.

Results and discussion

In SILAC technique, two populations of cells are cultivated in cell culture at first. Then the growth medium with normal amino acids is fed to one population. On the contrary, the growth medium containing labeled heavy isotopes amino acids is fed to another one. The labeled amino acids are usually the lysine (K, +8.014199) and arginine (R, +10.008269). This population of cells would replace the heavy-labeled-isotopes into their proteins, so that the combined normal (light)-heavy cell populations can be analyzed together by LC-MS. The produced mass spectra can reflect the abundance ratios for the peptides and proteins in concern. In this study, the raw data is the combination of SILAC labeled yeast (S. cerevisiae) and unlabeled ones that mixed at various light/heavy ratios (1:2, 1:1, 1.5:1 and 2:1), and we analyzed these data by TPP in a web-based distributed system. The PeptideProphet [5] and the ASAPratio [36] are the TPP built-in validation and quantitation methods, respectively. We also called the TPP derived peptide ratios as the ASAPRatios, which will be used to tag the training data and evaluate the quality controller.

Feature extraction and training data preparation

Considering the strong correlations between the spectral features and the quality of the spectra, we extract thirteen features to design the quality classifiers. These features are mass deviation (MassDev), signal to noise ratio (S/N), preceding peak ratio (PPR), six isotope deviations (IsoDevs) and four scan isotope pattern deviations (SIDs). These features are discussed in "Methods" section. The final processed data is formed by these features with the size of $n \times 13$ for one sample, where n denotes the number of spectra.

To train a classifier, the data and corresponding training labels are required. The training label is intended to indicate the relationship between the target and the class. But in this study, there are two types of labels, one for training labels and the other for isotope labeling. Therefore, to avoid confusion, we adopt *tags* to represent the training labels.

During the whole procedure of LC-MS based proteomics experiments, the errors or mistakes are unavoidable, so it is desirable for us to filter out the low quality spectra (or corresponding peptides). If we have some historical spectral data with low or high quality tags, then we can use them directly to train the classifiers for quality control. However, we usually only have some LC-MS data with fixed mix ratios, and the samples in the data do not have quality tags like good (positive) or bad (negative). To prepare the quality tags, we assume that we can have some data with prefixed quantitative ratios, and we followed the work of [16, 20] to manually labeled the data for the training of classifiers via peptide quantitative ratios and mass accuracy (deviation). It should be clarified that the classifiers trained by the manually tagged data only accept the 13 features designed in this paper and they will not use the quantitative ratio as the input feature.

In details, we use ASAPRatio to denote the estimated quantitative ratio, and the high quality peptides should have ASAPRatios be close to the ground truth. Figure 2 shows the details about the distribution of ASAPRatio in this 1:1 sample data. Figure 2a is the MA plot for the sample. The rest of the sub-figures (b, c and d) are generated by the binary logarithmic sample data ($\log_2$(Sample)), which are used for the probability density function (PDF) and cumulative distribution function (CDF). The MA plot for other samples are also included [see Additional file 1]. The histogram and the corresponding Gaussian kernel density estimation (KDE) in Fig. 2b represent the PDF of the sample. The empirical CDF and the Chebyshev's Inequality[37] are shown in Fig. 2c. Known by Chebyshev's Inequality, the data located in the range $[\mu-3\sigma, \mu+3\sigma]$ would keep at least 8/9 of the whole data. The empirical CDF illustrates that our sample has almost the same cumulative probability just in the locations that refer to 3σ when compared to the N$\sim$(0.22, 1) one. Figure 2d shows the CDF details with the interval $[\mu - 3\sigma, \mu + 3\sigma]$ in our data distribution. Hence, three-sigma of Chebyshev's Inequality is applied to group the logarithm transformed ASAPRatios manually, and its result is marked as *ratio-tag*.

The *ratio-tag* based on ASAPRatio narrows down the scope of high quality data, but it is not accurate enough because the distorted low quality spectrum may also produce a correct ratio. However, there is an intuition that the identified high quality peak should adopt the mass value close to the theoretical one. Thus, the mass deviation can be used to group the data further by this intuition. Figure 3 shows the distribution of standardized $((x - \mu)/\sigma)$ mass deviation and the corresponding histogram and density map as well. It follows from the histogram and density map in Fig. 3b that the standardized mass deviation value has a global maximum of about -0.13, which is the systematic bias of mass measurement. So a spectrum is tagged as positive if the standardized mass deviation value fell in the interval $[-0.13 - \text{threshold}, -0.13 + \text{threshold}]$. According to the distribution of mass deviations, here the threshold is set to 0.5 to exclude all the outliers. The corresponding threshold is also plotted as a short red vertical line in the density map. This tag result based on mass deviation is marked as *mass-tag*.

For the final tags of the 1:1 sample, the spectra with both positive *ratio-tag* and positive *mass-tag* are marked as positive, while the others are regarded as negative ones.

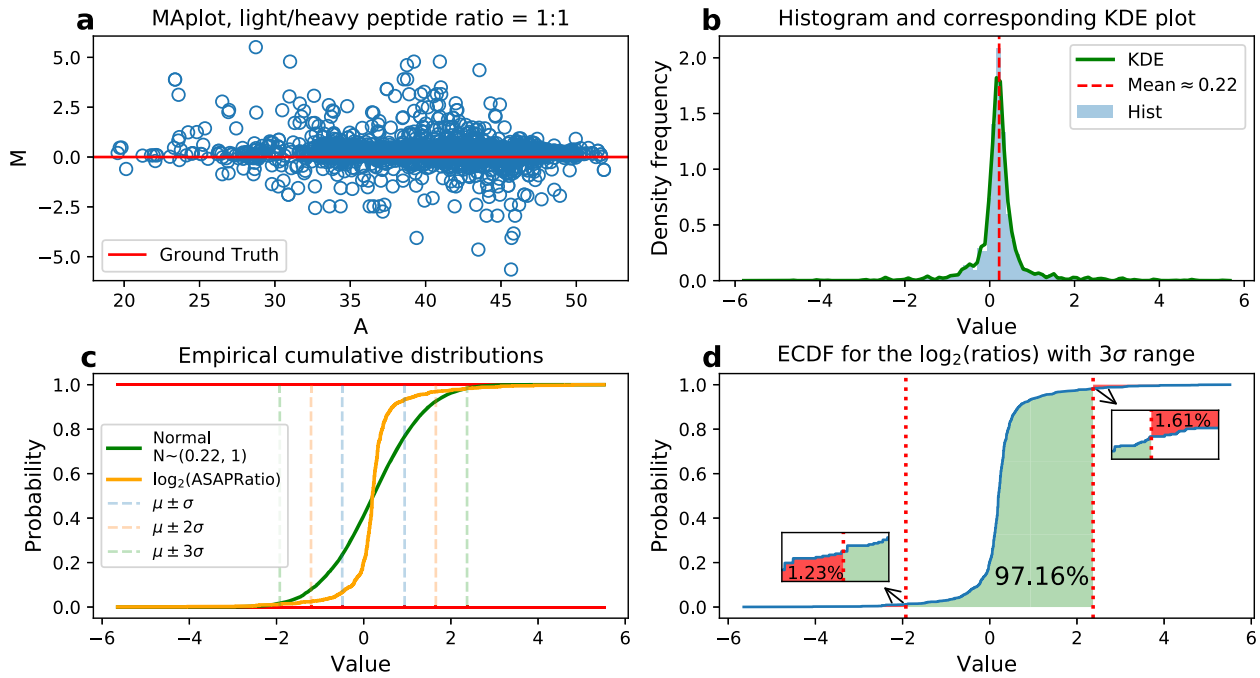

Fig. 2 Data distribution in peptide 1:1 sample. **a**. MA plot of ASAPRatio. **b**. Histogram and corresponding KDE plot for the binary logarithmic ASAPRatio. **c**. Empirical CDF for the binary logarithmic ASAPRatio and one normal distribution with k-σ range given by Chebyshev's Inequality, k=1,2,3. **d**. Cumulative probability within 3σ range of the binary logarithmic ASAPRatio, the red dotted lines stand for the boundaries of [$\mu \pm 3\sigma$]

Now we have a training data set for the optimization of the classifier. Since the quantitative ratios are not used as input feature, the trained classifiers will not be in favor of them. For the feature of mass deviation, the bias to it may be a problem. But according to the experimental results, such a biasing does not happen. The features other than mass deviation used in this paper play important roles in the classifiers (See Table 2 for details), and this makes the trained models much better than simple mass deviation based quality controller (Please refer Fig. 3a).

Over-sampling for the imbalanced data

Since there would be some inevitable contaminants in the process of culturing amino acids or errors occurred in the data collection, the final data would have some errors or unreliable parts. More importantly, these contaminants or errors are rare and unpredictable. But for the whole data, there is only few spectra influenced by these unreliable parts, theoretically. So the quantity of the high quality peptides (Positive) has an imbalance ratio with the low quality ones (Negative). By checking the tags of the training set (75% of total training data), we find that the number of positive tags is six times larger than that of the negative ones. The domination of positive tags highlights the data imbalance problem, which may lead to result the classifier does not learn enough from the minority class when training.

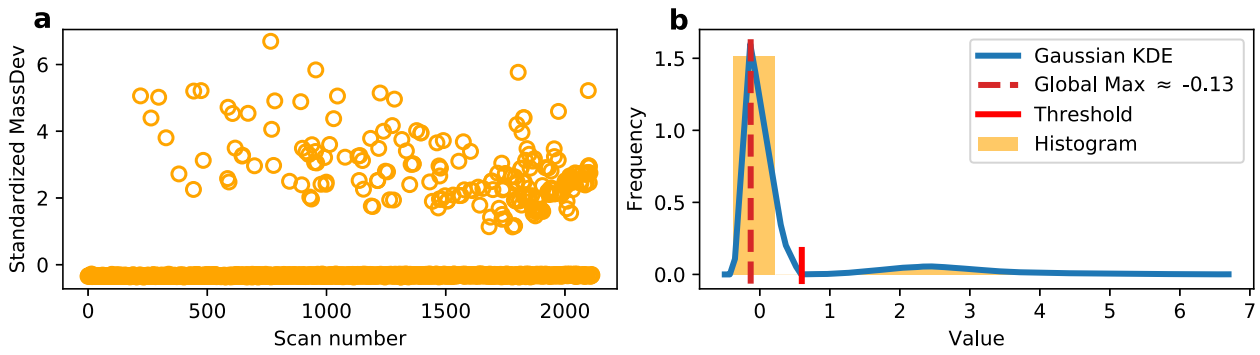

Fig. 3 Distribution of mass deviation and corresponding histogram and density map. **a**. Distribution of mass deviation. **b**. Histogram and the density map of standardized mass deviation, the red vertical line refers the threshold

We apply the SMOTE to re-balance the training set. Table 1 shows the details of the quantity changes in this training sample.

We also adopt the "gain" as the indicator of feature importance for the trained XGBoost models in Table 2. The gain is the most relevant attribute to interpret the relative importance of each feature in XGBoost. It implies the relative contribution of the corresponding feature to the model and is calculated by taking each feature's contribution for each tree in the model. Higher gain means this feature contributes more for prediction, and the gain with very small value usually means that the contribution of this feature is not significant.

Generally, a classifier can be considered appropriate if all meaningful training features contribute to the classifier. In this study, all the features in the training set were extracted based on the nature of the peptide, which is valuable for the peptide quantification. However, it follows from the Table 2 that the model generated from original data has many very small feature gains, so we think that the model may not be learned sufficiently because there are some features that contribute only little to it.

Classifier training and validation

There are many parameter tuning methods, and randomized search or grid search may be the most basic automatic methods for the models without deep architecture. However, these methods are really time-consuming and limited by the predefined set of parameter grids. In this paper, we employ a Bayesian optimization tool[38] to tune the XGBoost model, and the XGBoost model is implemented by the xgboost python package[39]. In details, we randomly divide the total training data into one training set (75%) and one testing set (25%), and then apply the training set to the optimization tool with XGBoost model for training. The optimization aims to find out the parameters that have the maximum mean value of the 10-fold cross validation evaluation scores under different parameters. We set the "roc_auc" value as the evaluation score of the cross validation. The bounds of the parameters in the optimization tool are set as follows:

- "learning_rate": (0.01, 0.3),
- "n_estimators": (10, 2000),

Table 1 Quantity changes with or without resample methods for the training set

	Positive Number	Negative Number	Total Number (for training)
Original Data	1585	529	2114
After Split (75%)	1367	218	1585
SMOTE	1367	**1367**	**2734**

The bold ones are the ones changed by the resample methods

Table 2 Gain as the feature importance for the XGBoost model with or without SMOTE

Feature Name	Gains without SMOTE	Gains with SMOTE
MassDev	11.54877	141.50188
PPR	0.52670	3.73862
S/N	0.39289	7.49105
IsoDev_Light1	1.30760	30.78152
IsoDev_Light2	2.51584	6.70143
IsoDev_Light3	3.52582	2.30519
IsoDev_Heavy1	0.64070	4.80937
IsoDev_Heavy2	0.49069	4.60379
IsoDev_Heavy3	0.69931	7.06721
SID_{sum}	0.25840	3.94553
SID_0	0.60187	5.18034
SID_1	0.62181	5.15330
SID_2	0.54873	4.74491

- "max_depth": (3, 10),
- "gamma": (0, 0.05),
- "colsample_bytree": (0.7, 1),
- "subsample": (0.7, 1).

The first three parameters denote the structure of the model, and the last three parameters solve the over-fitting problem by controlling the complexity and robustness of the model. Note that the values in "n_estimators" and "max_depth" are set as integer values.

After 30 iterations of the Bayesian optimization, we have got the following parameters:

- "learning_rate": 0.197,
- "n_estimators": 11,
- "max_depth": 10,
- "gamma": 0.04,
- "colsample_bytree": 0.97,
- "subsample": 0.96.

The trained classifiers can be applied to other samples without manual tagging. If the ground truth peptide ratio is provided, then the favorable classifiers should find high quality spectra that possess estimated heavy-light peptide ratios compactly close to the ground truth ones. Therefore, estimated peptide ratios are used to validate and evaluate classifiers.

To validate the performance of the classifier, the receiver operating characteristics (ROC) curve is employed. Specifically, since we applied the over-sampling in training, the 10-fold cross validation in parameter tuning is slightly different. The traditional cross validation method divides the training data evenly into k folds, and then enumerates k times as follows: each time we select one fold as the testing fold, and the other $k-1$ folds are

used as the training folds. Then the classifier is trained by the training folds, and we evaluate the classifier using the testing fold. For this special cross validation, we added the over-sampling for the $k-1$ training folds in the enumeration, and then the classifier is trained by the over-sampled training folds. Note that the testing fold is not over-sampled. By this manner, we validated the classifier by 10-fold cross validation with ROC curve and area under ROC curve (AUC) values in Fig. 4.

Quality control results

The XGBoost classifier is trained by the SMOTE re-balanced data with the parameters tuned above. For comparison approach, we added the SVM based quality control framework[20] as the baseline method. The features and the parameters in this SVM based framework are the same as the ones mentioned in [20]. Specially, a class weight parameter is declared in SVM for the imbalanced problem, and the imbalance ratio in their case is 2.2. For our situation, the imbalance ratio is about 6. Hence, we changed the default weight parameter 2.2 to 6. The SVM baseline method is implemented by the svm package in python scikit-learn[40].

We evaluated our classifiers on four SILAC yeast samples (1:2, 1:1, 1.5:1 and 2:1) [see Additional file 2]. Note that only a part of the 1:1 sample was used for training, and here the entire 1:1 one was used for evaluation. Furthermore, we adopted the mean, the mode and the coefficient of variation (CV) as the evaluation criteria. The criteria are calculated by the binary logarithmic peptide ratios. Typically, CV is defined as $CV = \sigma/\mu$. But for the logarithmic data, the way to calculate CV should be changed to make sense [41, 42], that is

$$\begin{aligned} CV &= \sqrt{\exp\left([\ln(\text{base})]^2\,\sigma^2\right) - 1} \\ &= \sqrt{\text{base}^{[\ln(\text{base})]\sigma^2} - 1}. \end{aligned} \tag{1}$$

While in our study, since the base of the logarithm is 2, we use this to calculate CV: $CV = \sqrt{2^{(\ln 2)\sigma^2} - 1}$.

Figure 5 illustrates the overall performance of our XGBoost models and SVM baseline model using CV as the indicator. It is clear that the quality of the peptide controlled by XGBoost model is quite concentrated compared to the SVM baseline approach, and this concentration is very useful for the quantitative analysis. Moreover, the classifier trained by the re-balanced data set provides better performance. It also can be seen from Figure 5 that it is more concentrated when the spectra are filtered out by about 30%. So we display the statistical details for the spectra that are filtered out by 30% in Table 3.

It should be noted that the evaluation of control results just based on mean or mode is not reliable because the ASAPRatio is only the estimated peptide ratio and there is an unknown systematic bias in such kind of estimation. For example, one classifier A results in many ratios close to 1.1, and the other classifier B results in most ratios close to 1.2. Even we know the ground truth ratio is 1, we cannot conclude that the classifier A is better because the unknown systematic bias may be 0.2 which makes the estimated ratio more accurate when it is close to 1.2. So we mainly evaluate the control results based on the variance.

In another point of view, what we needed in quantitation is a more accurate quantitative result. For a set of quantified peptide ratios, we believe that the results are accurate if the ratios are well concentrated and distributed around a certain value. So the means and the modes in Table 3 refer the "certain value", while the CVs imply the concentration, and the CV value close to zero indicates more

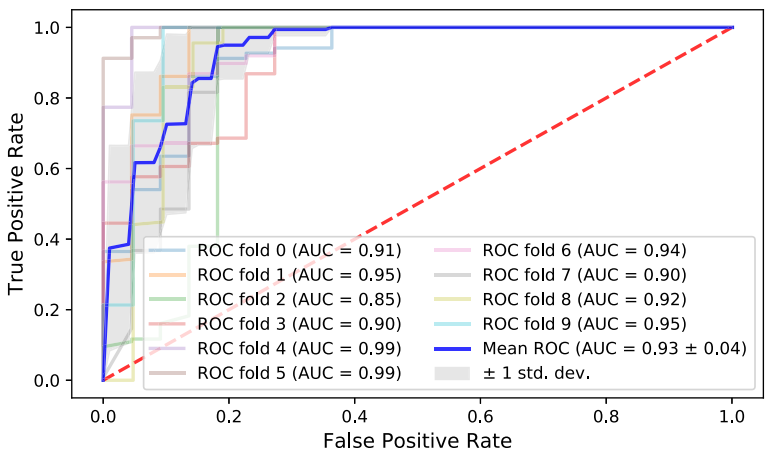

Fig. 4 Cross validation with ROC curves and AUC values

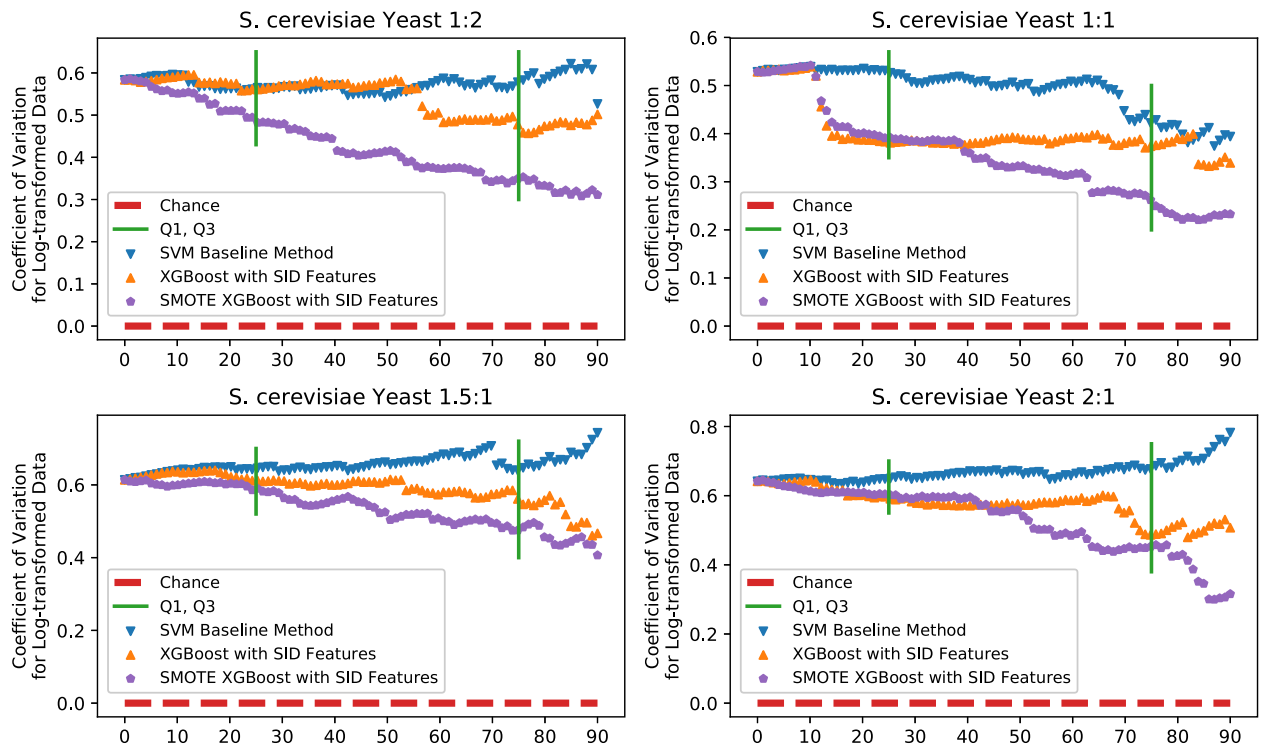

Fig. 5 Coefficient of ratio variation as a function of the percentage of spectra filtered by SVM baseline method (blue triangle), XGBoost without SMOTE method (orange triangle) and proposed XGBoost method (purple pentagon) for four yeast samples: 1(L):2(H), 1(L):1(H), 1.5(L):1(H) and 2(L):1(H). The green vertical lines denote for Q1 and Q3. The improvement of the CV's obtained by using XGBoost becomes fairly reliable

concentration. The means and the modes in the table are all close to the ground truth, and the CVs with XGBoost model are closer to zero than others. This means that the spectra quality controlled by XGBoost are more reliable. This method also provides reliable results for higher ratio samples [see Additional file 3].

Quality control for protein level quantification

Furthermore, the protein level quantification can benefit from the quality control of spectra. There is a basic idea that one protein should contain many peptides, the ratios in protein may vary with or without peptide quality assessment. The protein ratio should be compactly close to the ground truth when only using high quality spectra.

We conduct a simple experiment on this basic idea and show the ratio changes of four proteins in two samples in Table 4. Figure 6 shows the box plot of ratios for the four proteins. The results show that the quality control method makes the estimated protein ratios close to the ground truth with smaller variances.

Conclusion

For better quantitative analysis in LC-MS based proteomics experiments, this paper introduces some new approaches to construct a reliable quality assessor of spectra for isotopic labeled samples. There are mainly four types of variation have been associated with ratio estimation [43], and this work mainly focuses on reducing the artificial variation.

Table 3 Number of spectra, means, modes, and coefficients of variation for peptide ASAPRatios derived from four yeast samples (before filtering, after filtering by SVM base method, and after filtering by XGBoost model)

	Before filtering/After filtering by SVM baseline method/After filtering by XGBoost model			
Peptide ratio ($\log_2$(ratio))	Number of spectra	Mean	Mode	Coefficient of Variation
1:2 (-1)	2062/1444/1444	-0.77/-0.77/-0.80	-0.86/-0.89/-0.84	0.58/0.57/0.47
1:1 (0)	2114/1480/1480	0.22/0.22/0.21	0.19/0.19/0.19	0.53/0.51/0.39
1.5:1 (0.58)	2441/1709/1709	0.72/0.72/0.74	0.82/0.82/0.82	0.61/0.64/0.56
2:1 (1)	2110/1477/1477	1.09/1.07/1.12	1.30/1.29/1.30	0.64/0.65/0.59

Table 4 Comparison of protein ratio estimations with or without peptide quality control

Protein	YCR012W 1:1 Sample		YPL061W 1:1 Sample		YBR118W 1.5:1 Sample		YPL240C 1.5:1 Sample	
Assessor	Pep. Num.	Pep. Ratio	Pep. Num.	Pep. Ratio	Pep. Num.	Pep. Ratio	Pep. Num.	Pep. Ratio
Without assessor	153	1.7109±3.6754	27	0.9459±0.3516	68	2.6947±7.4300	33	1.6364±1.0504
XGBoost assessor	131	1.1922±0.3274	25	0.9484±0.3636	28	1.7054±0.3479	4	1.4675±0.5568

We find that the peptide quantification may be influenced by the XIC, so we introduce new features based on nearby LC scans for better classification. We also notice that the unbalanced data may affect the results of the assessment. For this problem, we re-sample the unbalanced spectral features using SMOTE technique and train the classifiers using the SMOTE set. The trained classifiers are tested on SILAC labeled samples. The results show that SMOTE XGBoost classifier is the state-of-the-art and capable of the quality assessment for mass spectra.

The recently proposed new re-sample methods [44] can be considered in future work. Furthermore, the feature extraction functions and the pre-trained classifiers of this method can be easily embedded into the LC-MS based quantitative proteomics analysis pipeline.

Methods

Spectral features

Mass deviation

Theoretically, the mass of one peptide is a definite value by the components of its amino acid. We marked this definite value as the neutral peptide mass (M_t). However, the experimental mass value would be different from the theoretical one due to the isotope. Meanwhile, we also marked the experimental peptide mass as M_e, and typically this M_e is the precursor neutral mass (monoisotopic mass).

The mass deviation (MassDev) is defined as the deviation level of the mass value, which is shown by the following Eq. 2.

$$\text{MassDev} = \frac{M_t - M_e}{M_t} \times 10^6, \tag{2}$$

where 10^6 is the unit parts per million (PPM), and smaller MassDev value refers better quality of the peptide.

Preceding peak ratio

The preceding peak of current peak has been explained to be an essential factor in evaluating the profile of a peptide in [17]. In details, the estimation of the ratio is unreliable when the preceding peak (M_{pp}) is comparable to the mono-isotopic peak (M_{mp}) in peak intensity. Hence, the reliability can be represented by the rate of M_{pp} and M_{mp}, and this is the preceding peak ratio (PPR). PPR is defined in Eq. 3.

$$\text{PPR} = \frac{M_{pp}}{M_{mp}}. \tag{3}$$

Note that the values would be set to zero if there are rare preceding peaks that cannot be identified, and the close to one PPR is a signal of unreliable spectra.

Signal to noise ratio

In signal processing, what we expected is the pure true peak signal. But noise is unavoidable now. In this way, the signal with less noise should be better, and one simple way to denote this is signal-to-noise ratio (S/N). The S/N is known as an important factor for evaluating the estimation ratio accuracy [15, 16], because the peptide

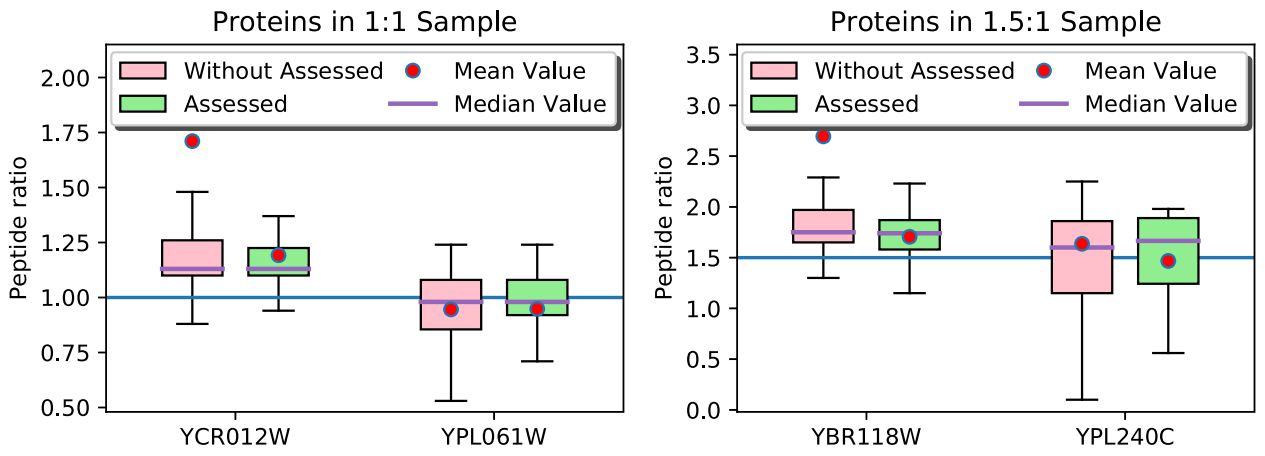

Fig. 6 Box plot for ratio estimations of four proteins with or without quality assessor

with better quality usually has lower noise. Generally, the median value of peak intensities is set to the noise level, and the mono-isotopic peak intensity value indicates the signal level [45].

Isotope deviations

Isotope is the key to the labeling technique. The theoretical isotopic pattern is a set of values associated with the relative abundance of the isotopes, but the experimental one may deviate from it. So the isotope deviation (IsoDev) is another critical feature for the SILAC spectra and can be obtained by both light and heavy labeled peptides.

Suppose that TP represents the theoretical isotopic pattern and EP stands for the experimental one, then the definition of the isotope deviation is given by the following Eq. 4 [20],

$$\mathrm{IsoDev}_i = \frac{\mathrm{TP}_i}{\mathrm{TP}_0} - \frac{\mathrm{EP}_i}{\mathrm{EP}_0}, \tag{4}$$

where $i = 1, 2, 3$ are the different deviations in each pattern. Note that the TP_0 and EP_0 represent the abundance of theoretical peak and mono-isotopic abundance in experiment, respectively.

Scan isotope pattern deviations

As far as we know, the accuracy of the estimated ratios is also influenced by the nature of corresponding peptides. While in a mass spectrometer the target peptide is identified at one corresponding LC scan, due to the continuity of LC and XIC, the neighboring scans of the target scan would also have valuable information, which is shown in Fig. 7.

The isotope patterns in neighboring scans are of great importance because they should be similar to the theoretical pattern of identified peptide. By considering the ratio between the first and the second peaks in the isotope clusters for the heavy and light peptides, we define a group of features named Scan Isotope Pattern Deviations (SIDs) to show the deviation between the mono first-second peak ratio of the target scan (M_0) and the integration of the experimental first-second peak ratios of neighboring scans by (5),

$$\mathrm{SID}_i = \frac{\mathrm{E}_i - \mathrm{M}_0}{\mathrm{M}_0}, \tag{5}$$

where

$$\mathrm{E}_i = \frac{\mathrm{L2}_i + \mathrm{H2}_i}{\mathrm{L1}_i + \mathrm{H1}_i}.$$

Here $i = 0, 1, 2$ refer the target and neighboring scans: scan0, scan1 and scan2. The $\mathrm{L1}_i$ and $\mathrm{L2}_i$ stand for the corresponding scan's first and second light peak intensities (blue and orange peaks in Figure 7), respectively. Similarly, $\mathrm{H1}_i$ and $\mathrm{H2}_i$ are the heavy ones (green and purple peaks in Fig. 7), respectively.

In addition, the $\mathrm{SID}_{\mathrm{sum}}$ is designed to show the ratio between the summarized first and second peaks in the isotopic cluster from neighboring scans in (6),

$$\mathrm{SID}_{\mathrm{sum}} = \frac{\mathrm{E}_{\mathrm{sum}} - \mathrm{M}_0}{\mathrm{M}_0}, \tag{6}$$

here

$$\mathrm{E}_{\mathrm{sum}} = \frac{\sum_{i=0}^{2} (\mathrm{L2}_i + \mathrm{H2}_i)}{\sum_{i=0}^{2} (\mathrm{L1}_i + \mathrm{H1}_i)}.$$

Due to the similarity among the isotope patterns in neighboring scans, the SIDs designed above should be close to each other and close to zero in high quality spectra. Similar to the PPR, there will be some unidentified peaks,in which case the SIDs would be set to -1. Therefore, the four SIDs are inserted into the feature set for training the classifiers.

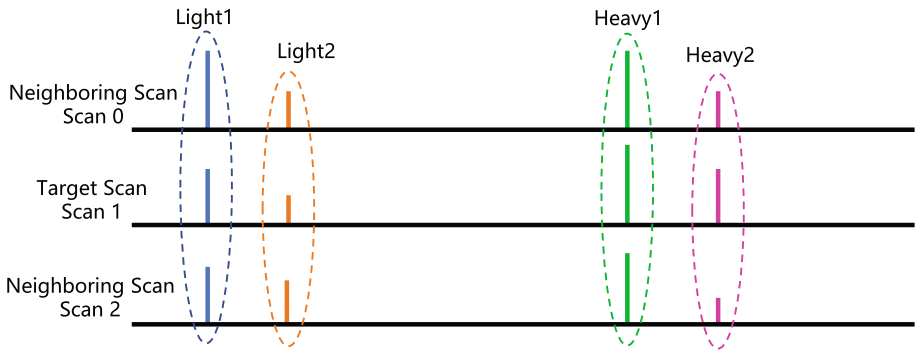

Fig. 7 Example for the target scan and its corresponding neighboring scans. The blue lines indicate the first light peak intensities (Light1), the orange lines stand for the second light peak intensities (Light2), the green lines are the first heavy peak intensities (Heavy1) and the purple ones represent the second heavy peak intensities (Heavy2)

Synthetic minority over-sampling technique

SMOTE is designed as a kind of over-sampling technique. Traditional over-sampling methods randomly repeat the minority samples as the newly-generated ones. But SMOTE calculates the nearest k^{th} neighbors by some distance methods at first, and then adds new sample between a data and it neighbors. More specifically, SMOTE adds a new data point by Eq. 7,

$$x_{new} = x + rand(0,1) \times ||\hat{x} - x||, \tag{7}$$

where x denotes the minority class sample, $||.||$ is the distance function and $\hat{x}$ represents the neighbors of x. Figure 8 illustrates the sampling procedure of SMOTE. We use the SMOTE in python environment by package *imbalanced-learn*[46].

Extreme gradient boosting machine

In the field of supervised learning, gradient boosting[35] has been empirically verified to be effective. XGBoost is a kind of gradient boosting method with tree ensemble approach. This method has become very famous in Kaggle since 2014 and is known for its high performance and excellent results. In this algorithm, the following Eq. 8 gives the definition of K additive function ensemble model (K trees),

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i),\ f_k \in \mathcal{F}, \tag{8}$$

where x_i stands for the i^{th} sample, $\mathcal{F}$ is the space that containing all regression trees and f_k refers the k^{th} function in the functional space $\mathcal{F}$.

To train the ensemble model, the objective in (9) needs to be minimized,

$$\mathcal{L}(\phi) = \sum_{i=1}^{n} loss(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k). \tag{9}$$

Here $loss$ is a loss function that measures the difference between target y_i and prediction $\hat{y}_i$. The Ω penalizes the complexity and is defined in [32] in (10).

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} ||w||^2. \tag{10}$$

The number of leaves in tree is defined as T, γ stands for minimum loss reduction, λ is the weight of regularization, $||w||$ represents the corresponding leaves' score (L2 norm), and the Eq. 11 can be used here to define the tree $f(x)$,

$$f_t(x) = w_{q(x)},\ w \in R^T,\ q : R^d \rightarrow 1, 2, \ldots, T, \tag{11}$$

here w denotes the score function, and q is the function that assigns each data point to the corresponding leaf (tree structure).

The objective in (9) is optimized by training the tree ensemble model in an additive (boosting) manner. Suppose that $\hat{y}_i^{(t)}$ is the prediction of the i^{th} instance at t^{th} training round, in additive manner a new f_t should be added to minimize the following objective,

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} loss\left(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t). \tag{12}$$

Taylor expansion is applied to (12) to quickly optimize the objective in general setting [47], obtaining Eq. 13 here,

$$\mathcal{L}^{(t)} \simeq \sum_{i=1}^{n} \left[loss\left(y_i, \hat{y}_i^{(t-1)}\right) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t), \tag{13}$$

where $g_i = \partial_{\hat{y}^{(t-1)}} loss\left(y_i, \hat{y}^{(t-1)}\right)$ denotes the statistics of first order gradient on the loss function and $h_i = \partial^2_{\hat{y}^{(t-1)}} loss\left(y_i, \hat{y}^{(t-1)}\right)$ is the second order ones.

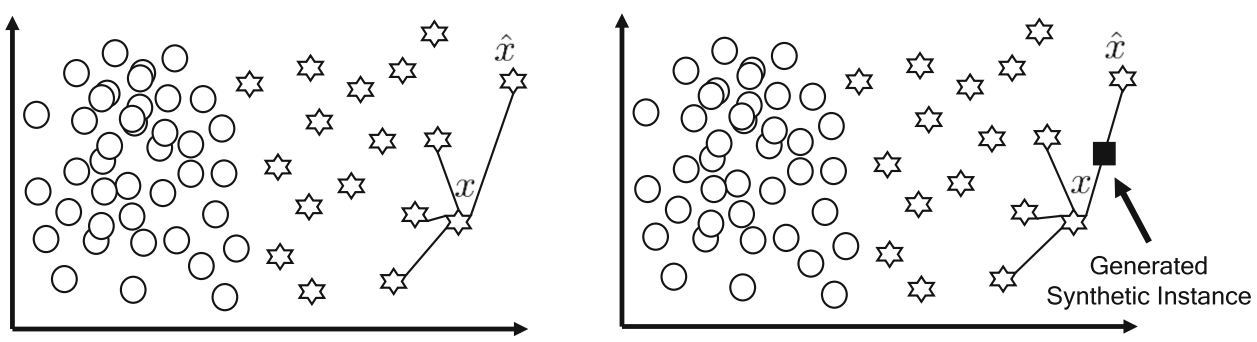

Fig. 8 Example for SMOTE. The circles are the majority class samples and the hexagrams are the minority ones. The rectangle is the new sample generated by SMOTE

The constant can be removed for simplifying the objective function at step t, and we get

$$\tilde{\mathcal{L}}^{(t)} = \sum_{i=1}^{n}\left[g_i f_t(x_i) + \frac{1}{2}h_i f_t^2(x_i)\right] + \gamma T + \frac{1}{2}\lambda\sum_{j=1}^{T}||w||^2. \quad (14)$$

Let $I_j = \{i|q(x_i) = j\}$ be the instance set of leaf j, then the Eq. 14 can be expanded to obtain

$$\tilde{\mathcal{L}}^{(t)} = \sum_{j=1}^{T}\left[\left(\sum_{i\in I_j} g_i\right)w_j + \frac{1}{2}\left(\sum_{i\in I_j} h_i + \lambda\right)w_j{}^2\right] + \gamma T. \quad (15)$$

Suppose that $q(x)$ is a fixed structure, then the best weight w_j^* of leaf j is calculated as follows,

$$w_j^* = -\frac{G_j}{H_j + \lambda}, \quad (16)$$

where $G_j = \sum_{i\in I_j} g_i$ and $H_j = \sum_{i\in I_j} h_i$, and the corresponding optimal value can be calculated by

$$\tilde{\mathcal{L}}^{(t)}(q) = -\frac{1}{2}\sum_{j=1}^{T}\frac{G_j^2}{H_j + \lambda} + \gamma T. \quad (17)$$

The quality of the tree structure can be evaluated by the score function (17), but it is impossible to enumerate all structures. So a greedy approach[35] is employed here. More specifically, we grow the tree from a single leaf and try to optimize (17) by splitting one leaf into two iteratively. For one split, the instance set is partitioned into left nodes(IL) and right nodes(IR) with $InstanceSet = IL \cup IR$. So the gain $\mathcal{G}$ after this split is given by Eq. 18.

$$\mathcal{G}_{split} = \frac{1}{2}\left[\frac{G_{IL}^2}{H_{IL} + \lambda} + \frac{G_{IR}^2}{H_{IR} + \lambda} - \frac{(G_{IL} + G_{IR})^2}{H_{IL} + H_{IR} + \lambda}\right] - \gamma. \quad (18)$$

The splitting procedure will continue until the split gain $\mathcal{G}$ no longer positive.

Supplementary information

Supplementary information accompanies this paper at https://doi.org/10.1186/s12859-019-3170-1.

Additional file 1: This is a pdf file (233KB) containing all samples' MA plots and corresponding distribution plots. The related peptide ratios are shown in the title of each plot.

Additional file 2: This is a four-sheet xls file (7180KB) containing the TPP analysis results and extracted features, each sheet refers to one dataset with special ratio in its sheet name, e.g., 1_1 is the ratio 1:1 sample data.

Additional file 3: This is a pdf file (69KB) containing an example for controlling the quality of the spectrum with high peptide ratio.

Abbreviations

AUC: Area under ROC curve; DNN: Deep neural network; GBDT: Gradient boosting decision tree; IsoDev: Isotope deviation; LC-MS: Liquid chromatography-mass spectrometry; MassDev: Mass deviation; PPM: Parts per million; PPR: Preceding peak ratio; ROC: Receiver operating characteristics; S/N: Signal to noise ratio; SID: Scan isotope pattern deviation; SILAC: Stable Isotope Labeling with Amino Acids in Cell Culture; SMOTE: Synthetic minority over-sampling technique; SVM: Support vector machine; TPP: Trans-Proteomic Pipeline; XGBoost: Extreme gradient boosting machine; XIC: Extracted ion chromatogram

Acknowledgements

The authors acknowledge and thank the anonymous reviewers for their suggestions that allowed the improvement of our manuscript.

Authors' contributions

LC conceived, designed, and supervised this study and edited the manuscript. TL conducted the simulations, interpreted the results, wrote and edited the first draft of the manuscript. MG advised, reviewed and edited the manuscript. All authors read and approved the manuscript.

Funding

This paper is supported by Science and Technology Development Fund, Macao S.A.R (097/2015/A3, 196/2017/A3), University of Macau RC (MYRG2015-00148-FST, MYRG2018-00132-FST), and the National Nature Science Foundation of China under Grant No.: 61673405.

Availability of data and materials

The extracted data supporting the conclusions of this article is included within the article and its additional files. The raw data and codes used during the current study are available from the corresponding author on reasonable request.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

[1]Department of Computer and Information Science, University of Macau, Taipa, Macau, China. [2]College of Mathematics and Computer Science, Fuzhou University, Fuzhou, Fujian, China.

Received: 1 February 2019 Accepted: 22 October 2019
Published online: 06 November 2019

References

1. Zhang J, Gao W, Cai J, He S, Zeng R, Chen R. Predicting molecular formulas of fragment ions with isotope patterns in tandem mass spectra. IEEE/ACM Trans Comput Biol Bioinformatics. 2005;2(3):217–30. https://doi.org/10.1109/TCBB.2005.43.
2. Chen L, Petritis K, Tegeler T, Petritis B, Haskins WE, Zhang J. Improved quantification of labeled lc-ms. In: 2011 IEEE International Conference on Bioinformatics and Biomedicine; 2011. p. 299–303. https://doi.org/10.1109/BIBM.2011.75.
3. Cui J, Ma X, Chen L, Zhang J. Scfia: a statistical corresponding feature identification algorithm for lc/ms. BMC Bioinformatics. 2011;12:439–9. https://doi.org/10.1186/1471-2105-12-439. 1471-2105-12-439[PII].
4. Yang P, Ma J, Wang P, Zhu Y, Zhou BB, Yang YH. Improving x!tandem on peptide identification from mass spectrometry by self-boosted percolator. IEEE/ACM Transactions on Computational Biology and Bioinformatics. 2012;9(5):1273–80. https://doi.org/10.1109/TCBB.2012.86.
5. Keller A, Nesvizhskii AI, Kolker E, Aebersold R. Empirical statistical model to estimate the accuracy of peptide identifications made by ms/ms and database search. Analytical Chemistry. 2002;74(20):5383–92. https://doi.org/10.1021/ac025747h.
6. Liu Y, Ma B, Zhang K, Lajoie G. An approach for peptide identification by de novo sequencing of mixture spectra. IEEE/ACM Transactions on Computational Biology and Bioinformatics. 2017;14(2):326–36. https://doi.org/10.1109/TCBB.2015.2407401.

7. Ong S-E, Blagoev B, Kratchmarova I, Kristensen DB, Steen H, Pandey A, Mann M. Stable isotope labeling by amino acids in cell culture, silac, as a simple and accurate approach to expression proteomics. Mol Cell Proteome. 2002;1:376–86.
8. Bittremieux W, Tabb DL, Impens F, Staes A, Timmerman E, Martens L, Laukens K. Quality control in mass spectrometry-based proteomics. Mass Spectrom Rev. 2017. https://doi.org/10.1002/mas.21544.
9. Kohlbacher O, Reinert K, Gröpl C, Lange E, Pfeifer N, Schulz-Trieglaff O, Sturm M. Topp – the openms proteomics pipeline. Bioinformatics. 2007;23(2):191. https://doi.org/10.1093/bioinformatics/btl299.
10. Cox J., Mann M. Maxquant enables high peptide identification rates, individualized p.p.b.-range mass accuracies and proteome-wide protein quantification. Nat Biotech. 2008;26(12):1367–72. https://doi.org/10.1038/nbt.1511.
11. Keller A, Eng J, Zhang N, Li X-j, Aebersold R. A uniform proteomics ms/ms analysis platform utilizing open xml file formats. Mol Syst Biol. 2005;1(1):. https://doi.org/10.1038/msb4100024. https://www.embopress.org/doi/pdf/10.1038/msb4100024.
12. Deutsch EW, Mendoza L, Shteynberg D, Farrah T, Lam H, Tasman N, Sun Z, Nilsson E, Pratt B, Prazen B, Eng JK, Martin DB, Nesvizhskii A, Aebersold R. A guided tour of the trans-proteomic pipeline. Proteomics. 2010;10(6):1150–9.
13. Pedrioli PGA. Trans-Proteomic Pipeline: A Pipeline for Proteomic Analysis. Totowa: Humana Press; 2010, pp. 213–238.
14. Deutsch EW, Lam H, Aebersold R. Data analysis and bioinformatics tools for tandem mass spectrometry in proteomics. Physiol Genomics. 2008;33(1):18–25. https://doi.org/10.1152/physiolgenomics.00298.2007. https://www.physiology.org/doi/pdf/10.1152/physiolgenomics.00298.2007.
15. Pan C, Kora G, Tabb DL, Pelletier DA, McDonald WH, Hurst GB, Hettich RL, Samatova NF. Robust estimation of peptide abundance ratios and rigorous scoring of their variability and bias in quantitative shotgun proteomics. Anal Chem. 2006;78(20):7110–20. https://doi.org/10.1021/ac0606554.
16. Bakalarski CE, Elias JE, Villén J, Haas W, Gerber SA, Everley PA, Gygi SP. The impact of peptide abundance and dynamic range on stable-isotope-based quantitative proteomic analyses. Journal of Proteome Research. 2008;7(11):4756–65. https://doi.org/10.1021/pr800333e.
17. Sadygov R. G., Zhao Y., Haidacher S. J., Starkey J. M., Tilton R. G., Denner L. Using power spectrum analysis to evaluate ^{18}o-water labeling data acquired from low resolution mass spectrometers. J Proteome Res. 2010;9(8):4306–12. https://doi.org/10.1021/pr100642q.
18. Silva JC, Gorenstein MV, Li G-Z, Vissers JP, Geromanos SJ. Absolute quantification of proteins by lcmse: a virtue of parallel ms acquisition. Mol Cell Proteomics. 2006;5(1):144–56.
19. Anderson D, Li W, Payan DG, Noble WS. A new algorithm for the evaluation of shotgun peptide sequencing in proteomics: support vector machine classification of peptide ms/ms spectra and sequest scores. J Proteome Res. 2003;2(2):137–46.
20. Nefedov AV, Gilski MJ, Sadygov RG. Svm model for quality assessment of medium resolution mass spectra from ^{18}o-water labeling experiments. J Proteome Res. 2011;10(4):2095–103. https://doi.org/10.1021/pr1012174.
21. Chang C, Zhang J, Han M, Ma J, Zhang W, Wu S, Liu K, Xie H, He F, Zhu Y. Silver: an efficient tool for stable isotope labeling lc-ms data quantitative analysis with quality control methods. Bioinformatics. 2014;30(4):586–7. https://doi.org/10.1093/bioinformatics/btt726.
22. Cui J, Petritis K, Tegeler T, Petritis B, Ma X, Jin Y, Gao S-JS, Zhang JM. Accurate lc peak boundary detection for 16o/18o labeled lc-ms data. PloS one. 2013;8(10):72951.
23. IZMIRLIAN G. Application of the random forest classification algorithm to a seldi-tof proteomics study in the setting of a cancer prevention trial. Ann N Y Acad Sci. 2004;1020(1):154–74. https://doi.org/10.1196/annals.1310.015.
24. Lin X, Wang Q, Yin P, Tang L, Tan Y, Li H, Yan K, Xu G. A method for handling metabonomics data from liquid chromatography/mass spectrometry: combinational use of support vector machine recursive feature elimination, genetic algorithm and random forest for feature selection. Metabolomics. 2011;7(4):549–58. https://doi.org/10.1007/s11306-011-0274-7.
25. Swan AL, Mobasheri A, Allaway D, Liddell S, Bacardit J. Application of machine learning to proteomics data: classification and biomarker identification in postgenomics biology. OMICS. 2013;17(12):595–610. https://doi.org/10.1089/omi.2013.0017.
26. Ma C. Deepquality: Mass spectra quality assessment via compressed sensing and deep learning. arXiv preprint arXiv:1710.11430. 2017.
27. Kim M, Eetemadi A, Tagkopoulos I. Deeppep: Deep proteome inference from peptide profiles. PLOS Comput Biol. 2017;13(9):1–17. https://doi.org/10.1371/journal.pcbi.1005661.
28. Zimmer D, Schneider K, Sommer F, Schroda M, Mühlhaus T. Artificial intelligence understands peptide observability and assists with absolute protein quantification. Front Plant Sci. 2018;9:1559.
29. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. Smote: Synthetic minority over-sampling technique. J Artif Int Res. 2002;16(1):321–57.
30. He H, Garcia EA. Learning from imbalanced data. IEEE Trans Knowl Data Eng. 2009;21(9):1263–84.
31. Wang S, Yao X. Multiclass imbalance problems: Analysis and potential solutions. IEEE Trans Syst Man Cybern B (Cybernetics). 2012;42(4):1119–30. https://doi.org/10.1109/TSMCB.2012.2187280.
32. Chen T, Guestrin C. Xgboost: A scalable tree boosting system. In: Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16. New York: ACM; 2016. p. 785–794. https://doi.org/10.1145/2939672.2939785.
33. Breiman L. Random forests. Mach Learn. 2001;45(1):5–32.
34. Liaw A, Wiener M. Classification and regression by randomforest. R news. 2002;2(3):18–22.
35. Friedman JH. Greedy function approximation: A gradient boosting machine. Ann Statist. 2001;29(5):1189–232. https://doi.org/10.1214/aos/1013203451.
36. Li X-j, Zhang H, Ranish JA, Aebersold R. Automated statistical analysis of protein abundance ratios from data generated by stable-isotope dilution and tandem mass spectrometry. Analytical Chemistry. 2003;75(23):6648–57. https://doi.org/10.1021/ac034633i.
37. Ross SM. Chapter 4 - random variables and expectation. In: Ross SM, editor. Introduction to Probability and Statistics for Engineers and Scientists. Fifth edition. Boston: Academic Press; 2014. p. 89–140. https://doi.org/10.1016/B978-0-12-394811-3.50004-6. http://www.sciencedirect.com/science/article/pii/B9780123948113500046.
38. Nogueira F. A Python implementation of bayesian global optimization with gaussian processes. https://github.com/fmfn/BayesianOptimization.
39. Chen T, He T, Khotilovich V, Xu B, Benesty M, Tang Y. dmlc XGBoost eXtreme Gradient Boosting. https://github.com/dmlc/xgboost.
40. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, Duchesnay E. Scikit-learn: Machine learning in Python. J Mach Learn Res. 2011;12:2825–30.
41. Wikipedia contributors. Coefficient of variation — Wikipedia, The Free Encyclopedia. 2019. https://en.wikipedia.org/w/index.php?title=Coefficient_of_variation.
42. Canchola J, Tang S, Hemyari P, Paxinos E, Marins E. Correct use of percent coefficient of variation (cv) formula for log-transformed data. MOJ Proteomics Bioinform. 2017;6(4):316–7.
43. Bantscheff M, Schirle M, Sweetman G, Rick J, Kuster B. Quantitative mass spectrometry in proteomics: a critical review. Anal Bioanal Chem. 2007;389(4):1017–31. https://doi.org/10.1007/s00216-007-1486-6.
44. Ma L, Fan S. Cure-smote algorithm and hybrid algorithm for feature selection and parameter optimization based on random forests. BMC Bioinformatics. 2017;18:169. https://doi.org/10.1186/s12859-017-1578-z.
45. Horn DM, Zubarev RA, McLafferty FW. Automated reduction and interpretation of high resolution electrospray mass spectra of large molecules. Journal of the American Society for Mass Spectrometry. 2000;11(4):320–32.
46. Lemaître G, Nogueira F, Aridas CK. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. J Mach Learn Res. 2017;18(17):1–5.
47. Friedman J, Hastie T, Tibshirani R. Additive logistic regression: a statistical view of boosting. Ann Stat. 2000;28:337–407. https://doi.org/10.1214/aos/1016218223.

Publisher's Note