Protein Inference: A Protein Quantification Perspective

Article *in* Computational biology and chemistry · February 2016

DOI: 10.1016/j.compbiolchem.2016.02.006

6 authors, including:

Zengyou He
Dalian University of Technology
87 PUBLICATIONS **1,834** CITATIONS
SEE PROFILE

Ting Huang
Dalian University of Technology
9 PUBLICATIONS **64** CITATIONS
SEE PROFILE

Peijun Zhu
3 PUBLICATIONS **2** CITATIONS
SEE PROFILE

Ben Teng
Dalian University of Technology
6 PUBLICATIONS **14** CITATIONS
SEE PROFILE

Available online at www.sciencedirect.com
ScienceDirect
Computational Biology and Chemistry 00 (2015) 1–14
COMPUT BIOL CHEM

Protein Inference: A Protein Quantification Perspective

Zengyou He[a,b,*], Ting Huang[c], Xiaoqing Liu[a], Peijun Zhu[a], Ben Teng[a], Shengchun Deng[d]

[a]*School of Software, Dalian University of Technology, Dalian, China.*
[b]*Key Laboratory for Ubiquitous Network and Service Software of Liaoning, Dalian, China.*
[c]*College of Computer and Information Science, Northeastern University, USA.*
[d]*School of Computer Science and Engineering, Harbin Institute of Technology, China.*

Abstract

In mass spectrometry-based shotgun proteomics, protein quantification and protein identification are two major computational problems. To quantify the protein abundance, a list of proteins must be firstly inferred from the raw data. Then the relative or absolute protein abundance is estimated with quantification methods, such as spectral counting. Until now, most researchers have been dealing with these two processes separately. In fact, the protein inference problem can be regarded as a special protein quantification problem in the sense that truly present proteins are those proteins whose abundance values are not zero. Some recent published papers have conceptually discussed this possibility. However, there is still a lack of rigorous experimental studies to test this hypothesis.

In this paper, we investigate the feasibility of using protein quantification methods to solve the protein inference problem. Protein inference methods aim to determine whether each candidate protein is present in the sample or not. Protein quantification methods estimate the abundance value of each inferred protein. Naturally, the abundance value of an absent protein should be zero. Thus, we argue that the protein inference problem can be viewed as a special protein quantification problem in which one protein is considered to be present if its abundance is not zero. Based on this idea, our paper tries to use three simple protein quantification methods to solve the protein inference problem effectively. The experimental results on six data sets show that these three methods are competitive with previous protein inference algorithms. This demonstrates that it is plausible to model the protein inference problem as a special protein quantification task, which opens the door of devising more effective protein inference algorithms from a quantification perspective. The source codes of our methods are available at: http://code.google.com/p/protein-inference/.

Keywords: Shotgun proteomics, Protein inference, Protein quantification, Spectral counting, Linear programming.

1. Introduction

Mass spectrometry (MS)-based shotgun proteomics is currently the most widely used method for the identification and quantification of proteins (Nesvizhskii et al., 2007). As shown in Figure 1, it first digests proteins in the sample into a mixture of peptides by enzymes such as trypsin. The resulting peptide mixtures are scanned by tandem mass spectrometry (MS/MS) to generate a set of MS/MS spectra. Then the peptide identification algorithm reports a set of peptide-spectrum matches (PSMs) by searching the MS/MS spectra

*Corresponding author. Tel.: +86 411 62274405. E-mail address: zyhe@dlut.edu.cn (Z. He)

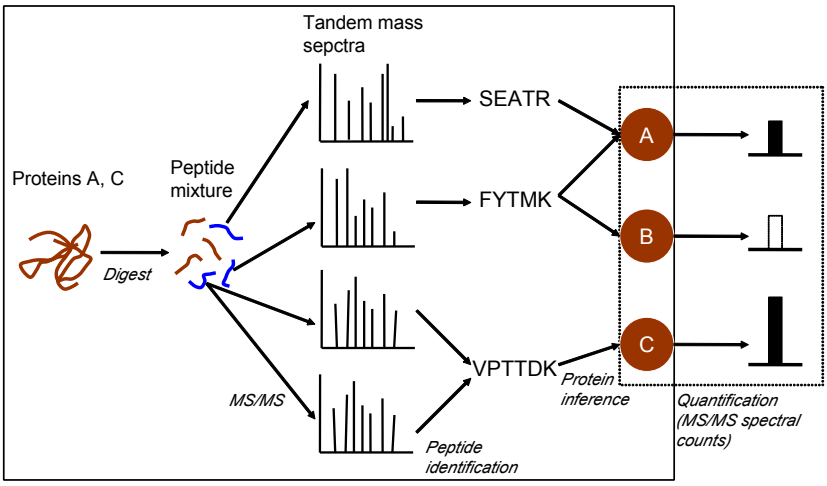

Figure 1: Protein identification and quantification using mass spectrometry in shotgun proteomics. There are three major computational problems: peptide identification, protein inference and protein quantification.

against a protein database. From these peptide identifications, we infer the existence of proteins with protein inference algorithms and calculate the relative or absolute abundances of proteins with protein quantification approaches.

Until recently, people tackle the identification and quantification of proteins as two individual and subsequent tasks: first select a subset of proteins that are truly present and then determine the quantities of these proteins. For both problems, many elegant approaches have been developed in the past decades. The readers can refer to two recent reviews Huang et al. (2012) and Nikolov et al. (2012) for details.

The starting point of this paper is fact that protein inference can be regarded as a special case of protein quantification. In protein inference, the objective is to generate a binary presence indicator value (1 or 0) for each candidate protein. In this regard, "protein existence inference" is probably more accurate for describing the original protein inference task. In protein quantification or "protein abundance inference", the goal is to determine the abundance of each protein. Clearly, if one protein is not present, its abundance value should be 0. Hence, the protein inference problem can be investigated from the perspective of protein quantification: present proteins are those proteins whose abundance values are not zero. In other words, we can adopt available protein quantification methods directly to solve the protein inference problem. This new angle may enable a better understanding of the protein inference problem and help in devising improved or hybrid protein inference methods by borrowing the power from protein quantification.

The possibility of exploiting protein quantification methods to solve the protein inference problem has been conceptually discussed in several papers (Dost et al., 2012; Li and Radivojac, 2012). Dost et al. (2012) used a simple example to show that it is feasible to obtain more accurate protein identifications with protein quantification methods than traditional parsimonious approaches. Li and Radivojac (2012) also pointed out that the protein inference problem can be regarded as a special protein quantification problem. However, they argued that existing protein quantification methods have not yet reached the accuracy needed for the wide dynamic range of quantities observed in cellular proteomics. As a result, solving the more general and difficult quantification problem may not provide a more accurate solution for the protein inference problem.

Although people have realized the potential of solving the protein inference problem from a quantification perspective, there are still no rigorous and extensive experimental studies to test this hypothesis. To fulfill this void, we empirically demonstrate the feasibility of solving the protein inference problem with existing protein quantification methods in the context of label-free proteomics. In the label-free quantitative proteomics studies, quantification methods which are based on peak ion intensities (from MS data) (Neilson et al., 2011) and spectral counting (from MS/MS data) (Lundgren et al., 2010; Choi et al., 2008) have been widely used.

Spectral counting measures the abundance of each protein based on the number of MS/MS spectra that match its constituent peptides. Given the peptide identification result, we can directly obtain spectral counting information since we just need to count the number of MS/MS spectra. In this paper, we use spectral counting as the quantification approach for solving the protein inference problem.

We first try two simple spectral counting methods in the literature. In both methods, the protein abundance is calculated as the sum of peptide abundance values. Their difference lies in how to handle the shared peptide. If the abundance of one shared peptide is b and it has k parent proteins, then b is used as its abundance value in the first method while b/k is used as its abundance value in the second method. These two methods assume that all the candidate proteins are present in the sample. As a result, the abundance value of each candidate protein will not be zero. However, this assumption contradicts the objective of protein inference: distinguishing present proteins (abundance$\neq$0) from absent proteins (abundance=0). Thus, we extend the second linear programming model in (Dost et al., 2012) to distribute the abundance values of shared peptides automatically in order to shrink the abundance values of absent proteins to zero.

To our knowledge, our paper is the first rigorous study with extensive experiments to demonstrate the feasibility of using protein quantification methods for solving the protein inference problem. Such an attempt connects two important computational problems that have long been investigated separately. The experimental results show that we can obtain better performance in most data sets even when the most simple version of spectral counting is utilized. Hence, the advance in protein quantification studies will promote the development of more effective protein inference algorithms.

In Section 2, we describe the details of three methods. Section 3 shows the experimental results on six data sets. Section 4 presents some discussions and Section 5 concludes the paper.

2. Methods

As shown in the left side of Figure 2, the input of the protein inference problem can be represented as a tripartite graph $G = (X \cup Y \cup Z, E_1 \cup E_2)$, where X, Y and Z are the set of l MS/MS experimental spectra, m identified peptides and n candidate proteins, respectively. For all $x_i \in X$, $y_j \in Y$, there is an edge $(x_i, y_j) \in E_1$ if and only if the spectrum x_i matches the peptide y_i in the peptide identification results. Similarly, $(y_j, z_k) \in E_2$ means that the peptide y_j is one part of the protein z_k . Each MS/MS spectrum corresponds to one and only one identified peptide whereas some peptides may have more than one matching spectrum, such as the peptides y_2 and y_3 in Figure 2. The relationship between peptides and proteins is more complex: one candidate protein may have several identified peptides and each peptide can be shared by multiple proteins. How to correctly distribute these shared peptides is one of the most challenging problem in protein inference.

We first formulate the protein inference problem as a special protein quantification problem. The objective of protein inference is to determine whether each candidate protein is present in the sample. The aim of protein quantification is to estimate the abundance value of each identified protein. Clearly, if one protein is not present in the sample, its abundance value should be 0. In this paper, the protein inference problem is re-visited from the perspective of protein quantification through seeking those proteins whose abundance values are not zero.

To obtain the protein abundance, we start with calculating the peptide abundance. Let b_j denote the abundance value of the peptide y_j, which can calculated as the sum of PSM probabilities (or scores):

$$b_j = \sum_{(x_i, y_j) \in E_1} a_i, \tag{1}$$

where a_i is the probability that the spectrum x_i matches the peptide y_j. Notice that a_i can be also viewed as the weight of edge $(x_i, y_j) \in E_1$, which can be obtained from peptide identification algorithms such as Mascot (Perkins et al., 1999) or post-processing tools such as PeptideProphet (Keller et al., 2002). In the traditional spectral counting methods, the peptide abundance is simply the number of MS/MS spectra identified for each peptide. Here, we generalize this spectral counting method to account for the quality of PSMs. More precisely, the contribution of each spectrum to the peptide abundance becomes a quantitative

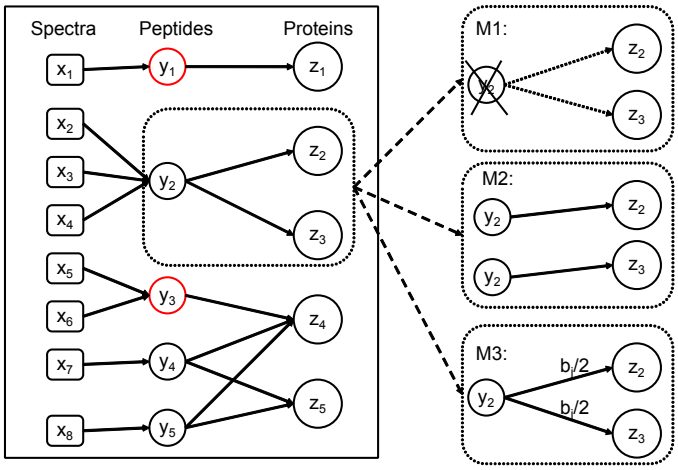

Figure 2: Three approaches for solving the shared peptide problem. y_1 and y_3 are unique peptides while y_2, y_4 and y_5 are shared peptides. The abundance of peptide y_j is represented by b_j. We use the peptide y_2 as an example to explain how these three approaches work.

value between 0 and 1 rather than a fixed value of 1. Such an extension is extremely important for protein inference since it may help us to distinguish between the proteins with the same number of PSMs.

To calculate the protein abundance, we need to distribute the abundance of each peptide to its parent proteins. The main difficulty is how to deal with the degenerate peptide that is shared by more than one protein since such a peptide can be generated by any subset of its parent proteins (Yang et al., 2013).

There are several approaches for solving the shared peptide problem in protein quantification (Zhang et al., 2010), as shown in the right side of Figure 2. The first approach is to simply discard the shared peptides and only use the unique peptides to calculate the protein abundance. But this approach has one disadvantage: it causes the loss of information, especially for the proteins whose identified peptides are all shared peptides. In Figure 2, if we delete the shared peptide y_2, then the proteins z_2 and z_3 do not have any identified peptides and they would be considered absent in the sample. In fact, at least one of these two proteins must be present if we assume the existence of peptide y_2. Alternatively, we can use both unique and shared peptides to estimate the protein abundance. In the second approach, the abundance of the shared peptide is utilized to calculate the abundance of all its parent proteins. In other words, each peptide is counted multiple times so that the abundance values of some proteins may be over-estimated. We call this method "multiple counting" in this paper. For example, the peptide y_2 in Figure 2 is counted twice in the second approach, which means that we artificially increase the abundance of peptide y_2 from b_2 to $2 * b_2$. The third approach divides the abundance of the shared peptide into different parts and then distributes each part to one of its parent proteins. This approach ensures that each peptide is "counted" only once. One typical representative in this category is the "equal division" method, which partitions the peptide abundance into k equal parts (k is the number of proteins that share this peptide).

Since both multiple counting and equal division are the most popular and simple quantification approaches based on spectral counting, we first try these two methods and test their performance for the protein inference task. Note that these two methods have an implicit assumption that the abundance value of each candidate protein is not zero. However, this assumption does not hold in the context of protein inference since the abundance values of some absent proteins should be zero. Thus, a new linear programming model is proposed, which can automatically distribute the peptide abundance so as to shrink the abundance values of some proteins to zero.

2.1. Multiple Counting

In this method, the shared peptides are used in the same way as the unique peptides and receive no special treatment. The abundance of a protein is simply the sum of abundance values from all its identified peptides:

$$c_k = \sum_{(y_j,z_k)\in E_2} b_j, \tag{2}$$

where c_k is the abundance of protein z_k. If the peptide y_j has q_j parent proteins, then it is counted q_j times and its actual abundance value used in the calculation is $q_j \cdot b_j$.

2.2. Equal Division

Different from the above method that counts shared peptides multiple times, the equal division method counts each peptide only once. It equally distributes the abundance of each shared peptide to its parent proteins:

$$c_k = \sum_{(y_j,z_k)\in E_2} \frac{b_j}{q_j}, \tag{3}$$

where q_j is the number of candidate proteins sharing the peptide y_j. If the peptide y_j is a unique peptide, then $q_j = 1$.

2.3. Linear Programming Model

Shared peptides play an important role in both protein inference and protein quantification. Dost et al. (2012) presented a linear programming (LP) model which used shared peptides to estimate the relative protein abundance. Kim (2012) modified this LP model to qualify the absolute protein abundance. On the basis of these attempts, we further extend the LP model and apply it to infer the identities of proteins.

For each identified peptide y_j, the peptide abundance can be computed as:

$$b_j = \sum_{\{k|(y_j,z_k)\in E_2\}} det_{jk} \cdot c_k = \sum_{\{k|(y_j,z_k)\in E_2\}} d_{jk}, \tag{4}$$

where $det_{jk} \in (0, 1)$ is the detectability of the peptide y_j and it represents the probability that the peptide y_j can be identified in a standard experiment if its parent protein z_k is present (Tang et al., 2006). In order to simplify the model, we introduce a new variable d_{jk} to replace the product between the peptide detectability det_{jk} and the protein abundance c_k. Then, d_{jk} is interpreted as the abundance that the protein z_k contributes to the peptide y_j. The variable d_{jk} can serve as the bridge between the peptide abundance and the protein abundance. On one hand, we can use d_{jk} to explain the known peptide abundance. On the other hand, we can calculate the unknown protein abundance through d_{jk}. Therefore, the protein quantification problem is equivalent to finding an optimal matrix $D = (d_{jk})$.

According to the above analysis, we propose a new LP model to solve the protein quantification problem:

$$\min_D \sum_{k=1}^{n} t_k, \tag{5}$$

$$\forall j : b_j - \sum_{\{k|(y_j,z_k)\in E_2\}} d_{jk} = 0, \tag{6}$$

$$\forall j,k : d_{jk} \le t_k, \tag{7}$$

$$\forall j,k : d_{jk} \sim \begin{cases} = 0 & if\ (y_j, z_k) \notin E_2 \\ \ge 0 & else \end{cases}. \tag{8}$$

Constraint (6) forces the predicted peptide abundance to be equal to the observed abundance value. Constraint (7) is to find the maximum value in each column vector d_k (the kth column of the matrix D).

Then, minimizing the objective function (the sum of maximum peptide abundance value from each protein) will shrink the abundance values of some proteins to 0.

After obtaining the matrix D, it is still a non-trivial task to recover the protein abundance value c_k since the peptide detectability value det_{jk} is unknown. If we assume that $\sum_{\{j|(y_j,z_k)\in E_2\}} det_{jk} = 1$, then the protein abundance c_k can be calculated as:

$$c_k = \sum_{\{j|(y_j,z_k)\in E_2\}} det_{jk} \cdot c_k = \sum_{\{j|(y_j,z_k)\in E_2\}} d_{jk}. \quad (9)$$

Notice that the above assumption on the sum of peptide detectability values is generally not true. Therefore, the calculated value according to Equation (9) is only an estimated value of the true protein abundance.

Previously, we have introduced a linear programming method, ProteinLP (Huang and He, 2012), to solve the protein inference problem. The LP model presented in this paper is essentially different from ProteinLP at least in the following ways:

- Our paper is based on the idea that the protein inference problem can be solved as a special protein quantification problem. Here we want to show the possibility of using protein quantification methods to address the protein inference problem. Thus, the LP method in this paper is actually a special protein quantification method, which mainly deals with peptide/protein abundance values. While ProteinLP focuses on calculating the protein existence probability based on the peptide identification probability values.

- These two methods have different assumptions. ProteinLP assumes that one peptide will be absent if all its parent proteins are not present in the sample. The LP model in this paper is based on the assumption that the abundance value of a peptide is equal to the sum of the abundance values from all its parent proteins.

- The variables in these two LP models are different. The variable of ProteinLP is a mathematical transformation of the joint probability that both a protein and its constituent peptide are present in the sample. The variable in this paper is the abundance that one parent protein contributes to its constituent peptide.

- The outputs of these two methods are different. The output of ProteinLP is the probability that one protein is present while that of our method is the protein abundance.

- The new LP model does not need any parameters while ProteinLP has to specify a threshold parameter manually. In order to find the proper parameter automatically, ProteinLP still needs to run an additional parameter selection procedure.

2.4. Converting Scores into Probabilities

After knowing the protein abundance, it is beneficial to convert the abundance into well-calibrated probability. The main reason is that the probability estimation allows us to select the appropriate threshold for reporting a set of confident proteins. In fact, the problem of converting ranking scores into estimated probabilities has been widely investigated in different domains (e.g., Gao and Tan (2006)). In this paper, we use the method proposed in (Gao and Tan, 2006) to fulfill this task.

We first estimate the probability p_k that the protein z_k is present in the sample given its abundance c_k:

$$\begin{aligned} & Pr(z_k = 1|c_k) \\ &= \frac{Pr(c_k|z_k = 1)Pr(z_k = 1)}{Pr(c_k|z_k = 1)Pr(z_k = 1) + Pr(c_k|z_k = 0)Pr(z_k = 0)} \\ &= \frac{1}{1 + \exp(-f_k)}, \end{aligned} \quad (10)$$

where

$$f_k = \log \frac{Pr(c_k|z_k = 1)Pr(z_k = 1)}{Pr(c_k|z_k = 0)Pr(z_k = 0)}. \tag{11}$$

f_k can be considered as a discriminant function which has a Gaussian distribution with equal covariance matrices (Bishop, 1995). Then, Equation (10) becomes

$$p_k = \frac{1}{1 + \exp(Ac_k + B)}. \tag{12}$$

Now, we need to estimate the parameters, A and B. Let r_k be a binary variable whose value is 1 if the protein z_k is present in the sample and 0 otherwise. Then, $R = (r_1, r_2, \cdots, r_n)$ is the presence indicator vector of n candidate proteins. If we assume that the existence of each protein is independent of other proteins, the probability of observing R given C is:

$$Pr(R|C) = \prod_{k=1}^{n} p_k^{r_k}(1 - p_k)^{1-r_k}, \tag{13}$$

where $C = \{c_1, c_2, \cdots, c_n\}$. The optimal parameter values should maximize $Pr(R|C)$, i.e., minimize the following negative log likelihood function:

$$LL(R|C) = \sum_{k=1}^{n} [(1 - r_k)(-Ac_k - B) + \log(1 + \exp(Ac_k + B))]. \tag{14}$$

Equation (14) is based on the assumption that we have already known the indicator vector R. However, we do not know such information in the protein inference process. Thus, we consider r_ks as hidden variables and employ the EM algorithm to simultaneously estimate A, B and R.

The EM algorithm utilizes an iterative procedure to estimate the parameter value $\theta = \{A, B\}$. The procedure includes two steps: set $r_k^{s+1} = E(r_k^s|C, \theta^s)$ (E-step) and compute $\theta^{s+1} = \arg\min_\theta LL(R^{s+1}|C)$ (M-step), where s is the iteration index. During the E-step, the unknown vector R is replaced by its expected value R^{s+1} under the current estimated parameter value θ^s. Since θ^s is fixed, $LL(R|C)$ is minimized by setting $r_k = 0$ if $Ac_k + B > 0$ or $r_k = 1$ if $Ac_k + B \leq 0$. During the M-step, a new parameter estimation θ^{s+1} is computed by minimizing $LL(R|C)$ given the vector R^{s+1} calculated by the first step. Since $R^s = [r_k^s]$ is fixed, minimizing $LL(R|C)$ with respect to A and B is a two-parameter optimization problem, which can be solved using the model-trust algorithm described in (Platt, 2000).

In the above score transformation procedure, all proteins share the same set of model parameters. In fact, the estimated abundance values from different proteins are generally not comparable since longer proteins may tend to have more matched mass spectra than shorter proteins even they have the same quantities. Therefore, a new model that takes into account more factors such as the length and ionization properties of proteins should be developed in the future.

3. Experimental Results

To test the performance of quantification-based protein inference methods, we have compared our methods with ProteinProphet (Nesvizhskii et al., 2003) and ProteinLP (Huang and He, 2012) on the six datasets.

3.1. Data sets

We choose six publicly available data sets to validate the performance of our methods. The names and URLs of these data sets are given in Table 1. These six data sets are divided into two categories: three data sets with reference sets and the other three data sets without reference sets. The first three data sets, 18 mixtures (Klimek et al., 2008), Sigma49 (Tabb et al., 2007) and yeast (Ramakrishnan et al., 2009a), have their corresponding reference sets that contain the ground-truth proteins. The another three data sets, DME (Brunner et al., 2007), HumanMD (Ramakrishnan et al., 2009b) and HumanEKC (Ramakrishnan et al.,

2009a), do not have such reference sets. For the data sets without reference sets, a target-decoy strategy is used instead to assess the performance. This strategy searches MS/MS spectra against a hybrid protein database which is composed of target protein sequences from the original database and the same number of decoy sequences (Teng et al., 2014). Thus, an identified protein is considered as a true positive if it is present in the protein reference set or comes from the target protein database.

Mixture of 18 Purified Proteins (18 mixtures) and Sigma49 data set. These two data sets are both generated from the sample of synthetic proteins mixtures. The protein database used for the 18 mixtures data set consists of 1,819 protein sequences, which includes 18 ground-truth proteins and some contaminant proteins. The database for the Sigma49 data set contains 15,682 Swiss-Prot human protein sequences.

Yeast data set. Its reference set is available at http://www.marcottelab.org/MSdata/gold/yeast.html. The protein database includes 6,714 protein sequences.

D. melanogaster data set (DME). The DME data set is produced from the embryonal Kc 167 cell line of D. melanogaster. We use Flybase (release 5.2) as the protein database, which contains 20,726 entries.

HumanMD data set and HumanEKC data set. The HumanMD data set is generated from medulloblastoma Daoy cell line and the HumanEKC data set is produced from human embryonic kidney T293 cell line. We use Ensembl (version 49.36k) as the protein database, which has 22,997 entries.

Table 1: **The data sets used in the experiment and their URLs.**

Data Set	The URL of Raw Data
Mixture of 18 Purified Proteins (Klimek et al., 2008)	http://regis-web.systemsbiology.net/PublicDatasets/
Sigma49 Data Set (Tabb et al., 2007)	https://proteomecommons.org/dataset.jsp?i=71610
Yeast Data Set (Ramakrishnan et al., 2009a)	http://www.marcottelab.org/users/MSdata/Data_02/
D. melanogaster Data Set (Brunner et al., 2007)	http://www.peptideatlas.org/repository/ (PAe001349)
HumanMD Data Set (Ramakrishnan et al., 2009b)	http://www.marcottelab.org/MSdata/Data_05/
HumanEKC Data Set (Ramakrishnan et al., 2009a)	http://www.marcottelab.org/MSdata/Data_07/

3.2. Peptide Identification

We use X!Tandem (v2010.10.01.1) (Craig and Beavis, 2004) for peptide identification with default search parameters. For the data sets with the reference sets, the MS/MS spectra are only searched against the target protein databases. For the data sets without the reference sets, the spectra are searched against both target and decoy protein databases. The peptide identification results are post-processed with PeptideProphet (Trans-Proteomic Pipeline v4.5) to obtain the presence probability for each peptide.

3.3. Protein Inference

We choose ProteinProphet and ProteinLP as the competing methods. ProteinProphet is the most popular method for protein inference so far. ProteinLP is one representative method that is also based on linear programming. We run ProteinProphet with its default parameter setting and run ProteinLP with parameter $\epsilon = 0$. Since some distinct proteins may have the same set of identified peptides, we cannot distinguish these proteins from each other without further evidence. Therefore, all the protein inference methods in the experiments will put these indistinguishable proteins into the same group. Each group of indistinguishable proteins is treated as a single protein during the protein inference procedure. When we evaluate the performance of different methods, we count all proteins in each group and use the presence probability of each group as the identification probability for proteins in that group.

3.4. Results

We use the curve that shows the number of true positives as a function of the q-value to assess the performance of different methods. Given a certain probability threshold t, the q-value is the minimal false discovery rate (FDR) that is reported for a protein: $q_t = \min_{t' \leq t} FDR_{t'}$. The FDR is estimated

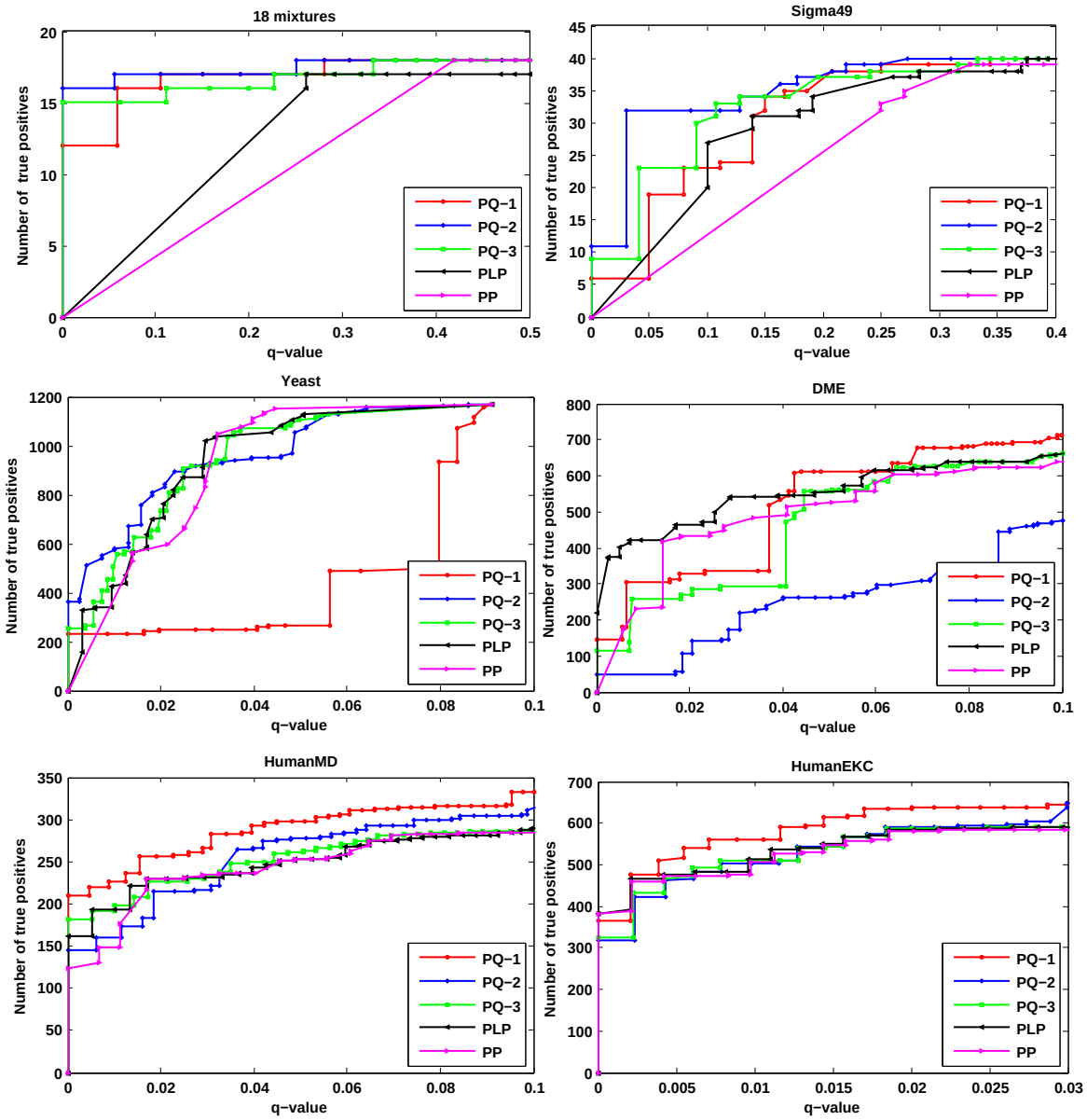

Figure 3: The comparison of identification performance among ProteinLP (PLP), ProteinProphet (PP) and our own three methods: multiple counting (PQ-1), equal division (PQ-2) and linear programming (PQ-3). If some proteins have the same probability in the ordered protein list, we skip these proteins with the same probability and calculate the q-value at the first encountered protein with a different probability.

as $FDR_{t'} = F_{t'}/(F_{t'} + T_{t'})$, where there are $T_{t'}$ true positives (TPs) and $F_{t'}$ false positives (FPs) with probabilities $\geq t'$.

Figure 3 displays the number of TPs reported by the five methods at different q-values. It shows that our methods are competitive with available protein inference algorithms. Throughout the six data sets, our three methods can always achieve zero FPs among the highest ranking proteins while the other two algorithms do not have such a property. This fact indicates that our methods have more strong distinction power than existing methods. More specifically, we have the following important observations.

First, the multiple counting method is the best performer on the HumanMD and HumanEKC data sets. For the HumanMD data set, it reports the largest number of TPs when the q-value is equal to 0. For the HumanEKC data set, it just identifies 17 fewer true positives than ProteinProphet at q-value=0. Even though the multiple counting method does not keep such excellent performance on the 18 mixtures, Sigma49 and DME data sets, it never performs the worst.

Second, equal division is the best performer (or tied with other algorithms) on the 18 mixtures, Sigma49 and yeast data sets. Similarly, when the q-value is equal to 0, it identifies the more TPs than other methods on the 18 mixtures, Sigma49 and yeast data sets. For the HumanMD data set, equal division does not have the worst performance. For the HumanEKC data set, the curve of equal division is almost tied with the curve of our LP model, ProteinProphet and ProteinLP and the gaps among these four methods are very small.

Third, our LP model exhibits the most stable identification performance among these five methods. More precisely, it does not perform the worst across all six data sets. ProteinLP also has such a property, but its performance is worse than three algorithms on the 18 mixtures and Sigma 49 data sets. In contrast, there is only one time that the performance of our LP model is worse than three algorithms (on the DME data set). The other three methods perform the worst on at least one data set. The number of data sets is 1, 2, 3 for multiple counting, equal division and ProteinProphet, respectively.

In the calculation of protein abundance, we generalize the number of MS/MS spectra to the sum of PSM probabilities. We wish such an extension may help us to distinguish between proteins with the same number of PSMs and further improve the identification performance. Figure 4 describes the performance gain when the generalized spectral counting is used instead of the traditional spectral counting. The experimental results of these three methods on the six datasets agree with our expectation: using the sum of PSM probabilities actually performs better than using the number of PSMs. Overall, there are 18 comparison results since we run our three methods on the six data sets. In these comparisons, the generalized spectral counting method performs obviously better than traditional spectral counting in 13 comparisons and performs as well as traditional spectral counting method in the remaining 5 comparisons.

The LP model in this paper is expected to be able to shrink the abundance values of some proteins to zero. Table 2 shows the effect of shrinkage on the six data sets. We record the number of total candidate proteins, the number of the proteins whose abundance values are zero and their rate. For the first two data sets generated from simple protein mixtures, there are around 4% proteins with abundance=0 while the proportion becomes $7\% \sim 8\%$ for the remaining four data sets generated from real samples.

Table 2: **The effect of shrinkage.** The percentage of proteins with abundance=0 is defined as the quotient between the number of proteins with abundance=0 and the number of total candidate proteins.

	18 mixtures	Sigma49	Yeast	DME	HumanMD	HumanEKC
Number of total candidate proteins	49	105	1285	907	414	669
Number of proteins with abundance=0	2	4	91	66	34	50
Percentage of proteins with abundance=0	4.1%	3.8%	7.1%	7.3%	8.2%	7.5%

After obtaining the protein abundance, we use an EM algorithm to convert the abundance score into a well-calibrated probability. Alternatively, we can just normalize the protein abundance by dividing the maximum of all calculated protein abundance values. The second strategy also gives us a protein score between 0 and 1 and keeps the holistic distribution of the original protein abundance unchanged. Figure 5 shows the reason why we adopt the more complex probability estimation approach. In this figure, the distributions of new scores generated from these two transformation methods are depicted. It is clearly visible that the probability estimation method is capable of generating a score distribution that is more close to the uniform distribution than the simple normalization method. This means that the probability estimation method allows for distinction between different proteins on a fine level.

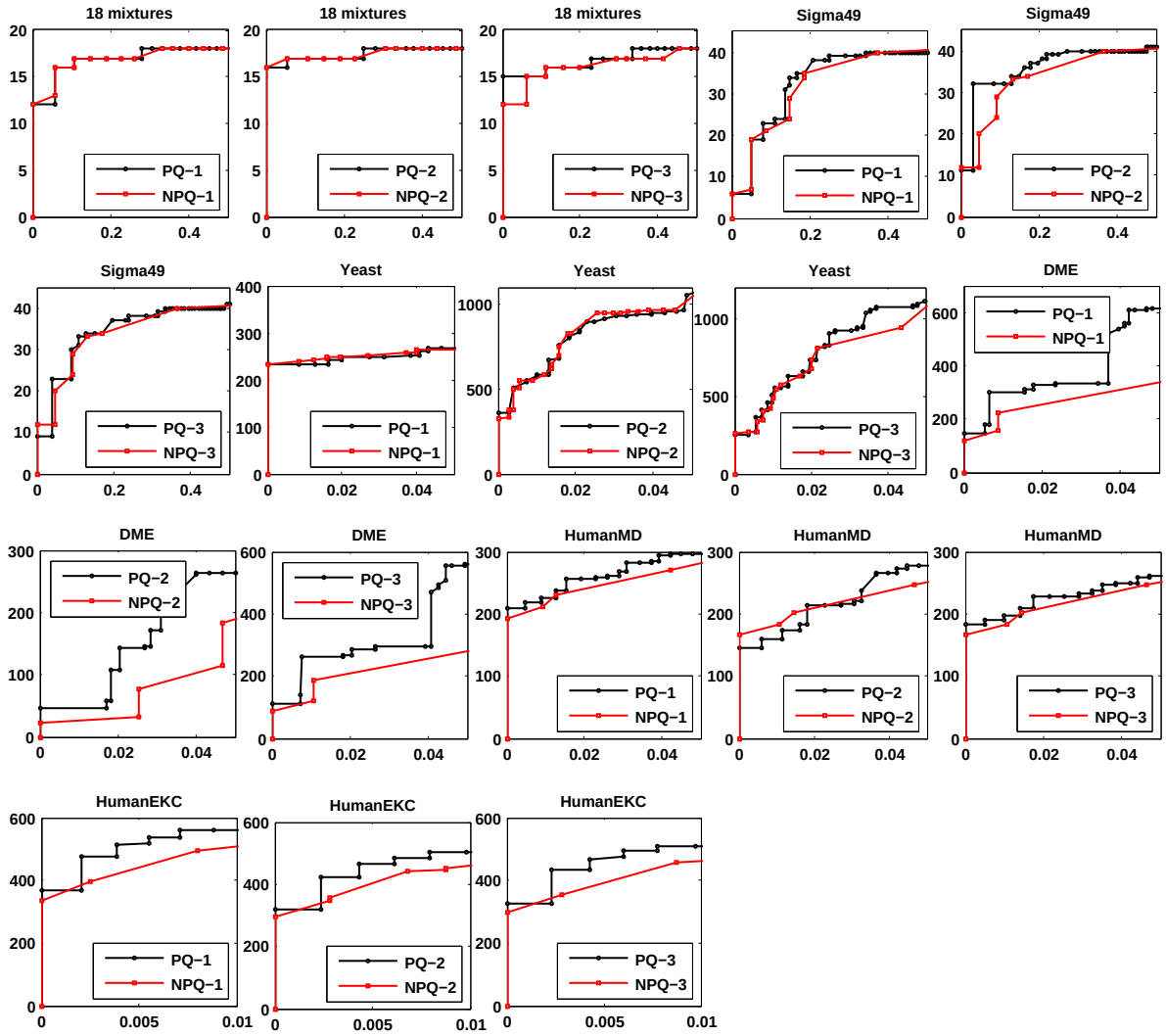

Figure 4: The comparison of identification performance between the generalized spectral counting methods (PQ-1, PQ-2, PQ-3) and the traditional spectral counting methods (NPQ-1, NPQ-2, NPQ-3). The y-axis is the number of true positives and x-axis is the corresponding q-value (the minimum FDR to report these proteins). The abbreviations for different methods are the same as those in Figure 3.

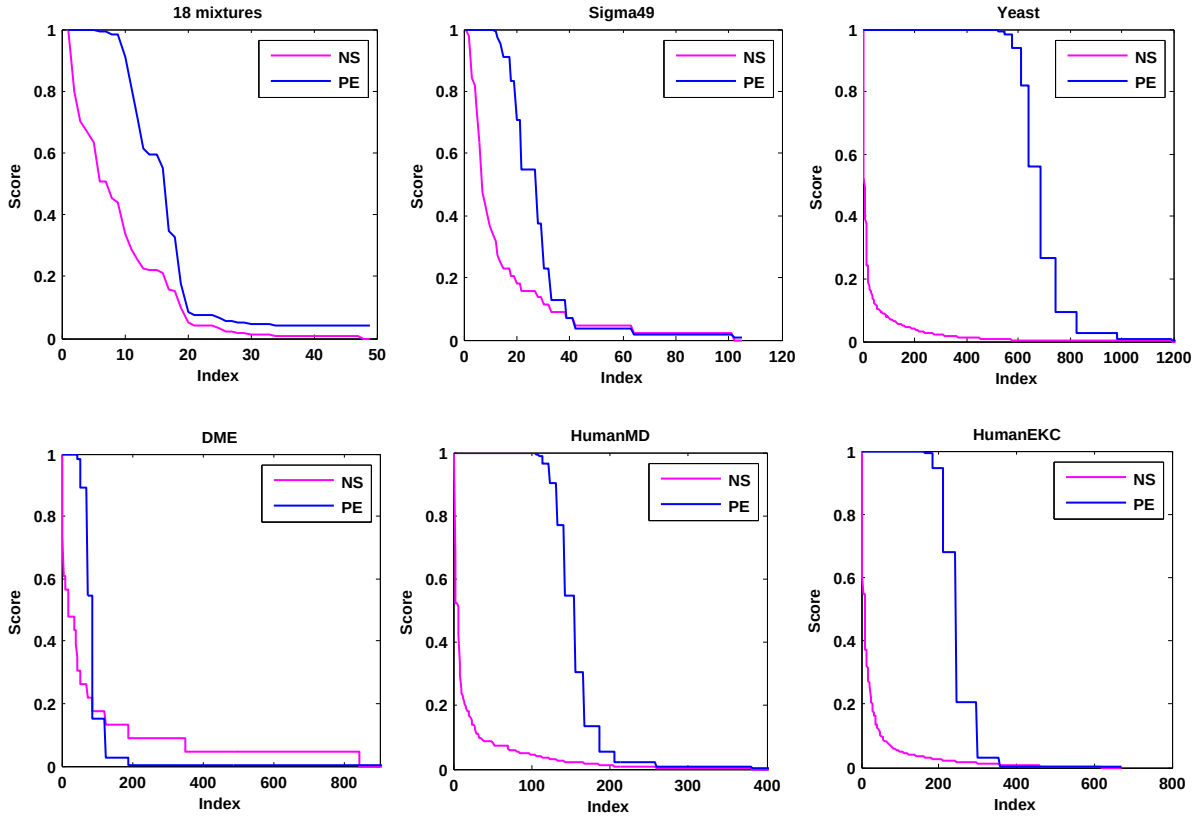

Figure 5: The comparison of the score distribution between normalized score (NS) and probability estimation (PE) when the protein abundance value is generated with the LP model. The scores of all the identified proteins are sorted by descending order.

4. Discussions

There have been already more than 20 protein inference algorithms in the literature, whose details are summarized in several reviews (Huang et al., 2012; Li and Radivojac, 2012; Claassen, 2012). Here we only discuss two inference methods that are most closely related with our work.

Based on the observation that peptides belonging to the same protein will show a good correlation with respect to their quantification patterns, Lukasse and America (2014) used the correlation of these patterns to validate peptide to protein matches. BagReg (Zhao et al., 2015) is a learning-based method for protein inference, which built a classification model based on several features such as the number of matched spectra for each protein. Overall, both methods utilized the quantification information in their algorithms rather than modeled the protein inference problem as a protein quantification problem.

The correct assignment of shared peptides to their parent proteins is one of most challenging problems in protein inference. However, it is generally very difficult to fulfill this task since the information included in the peptide-protein bipartite graph is insufficient for distinguishing correct peptide-protein matches from incorrect ones. Yang et al. (2013) mathematically investigated the ambiguity that will be induced by the uncertainty on the assignment of shared peptides. They derived a lower bound and an upper bound on the protein existence probability. Roughly speaking, all statistical protein inference methods will deliver a probability value between the lower bound and the upper bound. This partially explains why no methods can always perform the best in our experiments since all these methods cannot completely resolve the shared peptide assignment problem. In other words, all existing methods have already reached their theoretical limitation in protein inference if no supplementary data are provided for facilitating the inference. Therefore, it is unlikely that one can further improve the identification performance by only digging more on the mathematical formulation of the protein inference problem based on standard input data.

In fact, many researchers have already realized the aforementioned problem and begun to seek solutions by including supplementary information in the protein inference process. That is, in addition to the standard input data, supplementary data and information such as the single-stage MS data (He et al., 2010, 2011), peptide detectabilities (Li et al., 2009b; Huang et al., 2013) and protein-protein interactions (Ramakrishnan et al., 2009a; Li et al., 2009a) are utilized in the protein inference model as well. The use of extra information from other data sources may overcome the limitation of currently available protein inference algorithms.

5. Conclusions

Protein inference problem can be regarded as a special protein quantification problem. In this paper, we investigate the feasibility of solving the protein inference problem with existing protein quantification methods in the context of label-free proteomics. The experimental results show that such a new angle enables us to obtain better identification performance even with some simple quantification approaches.

We have tested three protein quantification methods for solving the protein inference problem. These three methods can achieve good performance but none of them are consistently the best method on all the data sets. Thus, it is still necessary to develop better algorithms. In the future work, we plan to try more quantification methods to check if we can further improve the identification performance.

Acknowledgements

This work was partially supported by the Natural Science Foundation of China under Grant No. 61572094 and the Fundamental Research Funds for the Central Universities of China (DUT14QY07).

References

Bishop, C. M., 1995. Neural Networks for Pattern Recognition. Oxford University Press, USA.

Brunner, E., Ahrens, C. H., Mohanty, S., Baetschmann, H., Loevenich, S., Potthast, F., Deutsch, E. W., Panse, C., de Lichtenberg, U., Rinner, O., Lee, H., Pedrioli, P. G. A., Malmstrom, J., Koehler, K., Schrimpf, S., Krijgsveld, J., Kregenow, F., Heck, A. J. R., Hafen, E., Schlapbach, R., Aebersold, R., 2007. A high-quality catalog of the drosophila melanogaster proteome. Nature Biotechnology 25 (5), 576–583.

Choi, H., Fermin, D., Nesvizhskii, A. I., 2008. Significance analysis of spectral count data in label-free shotgun proteomics. Molecular & Cellular Proteomics 7 (12), 2373–2385.

Claassen, M., 2012. Inference and validation of protein identifications. Molecular & Cellular Proteomics 11 (11), 1097–1104.

Craig, R., Beavis, R. C., 2004. Tandem: matching proteins with tandem mass spectra. Bioinformatics 20 (9), 1466–1467.

Dost, B., Bandeira, N., Li, X., Shen, Z., Briggs, S., Bafna, V., 2012. Accurate mass spectrometry based protein quantification via shared peptides. Journal of Computational Biology 19 (4), 337–348.

Gao, J., Tan, P.-N., 2006. Converting output scores from outlier detection algorithms into probability estimates. In: IEEE International Conference on Data Mining. Hong Kong, China, pp. 212–221.

He, Z., Yang, C., Yang et al, C., 2010. Optimization-based peptide mass fingerprinting for protein mixture identification. Journal of Computational Biology 17 (3), 221–235.

He, Z., Yang, C., Yu, W., 2011. A partial set covering model for protein mixture identification using mass spectrometry data. IEEE/ACM Transactions on Computational Biology and Bioinformatics 8 (2), 368–380.

Huang, T., Gong, H., Yang, C., He, Z., 2013. Proteinlasso: A lasso regression approach to protein inference problem in shotgun proteomics. Computational Biology and Chemistry 43, 46–54.

Huang, T., He, Z., 2012. A linear programming model for protein inference problem in shotgun proteomics. Bioinformatics 28 (22), 2956–2962.

Huang, T., Wang, J., Yu, W., He, Z., 2012. Protein inference: A review. Briefings in Bioinformatics 13 (5), 586–614.

Keller, A., Nesvizhskii, A. I., Kolker, E., Aebersold, R., 2002. Empirical statistical model to estimate the accuracy of peptide identifications made by MS/MS and database search. Analytical Chemistry 74 (20), 5383–5392.

Kim, D. H., 2012. Deconvolution of PPI networks: Approximation algorithms and optimization techniques. Ph.D. thesis, McGill University.

Klimek, J., Eddes, J. S., Hohmann, L., 2008. The Standard Protein Mix Database: A diverse data set to assist in the production of improved peptide and protein identification software tools. Journal of Proteome Research 7 (1), 96–103.

Li, J., Zimmerman, L. J., Park, B.-H., Tabb, D. L., Liebler, D. C., Zhang, B., 2009a. Network-assisted protein identification and data interpretation in shotgun proteomics. Molecular Systems Biology 5, 303.

Li, Y. F., Arnold, R. J., Li, Y., Radivojac, P., Sheng, Q., Tang, H., 2009b. A Bayesian approach to protein inference problem in shotgun proteomics. Journal of Computational Biology 16 (8), 1–11.

Li, Y. F., Radivojac, P., 2012. Computational approaches to protein inference in shotgun proteomics. BMC Bioinformatics 13 (Suppl 16), S4.

Lukasse, P. N., America, A. H., 2014. Protein inference using peptide quantification patterns. Journal of proteome research 13 (7), 3191–3199.

Lundgren, D. H., Hwang, S.-I., Wu, L., Han, D. K., 2010. Role of spectral counting in quantitative proteomics. Expert Review of Proteomics 7 (1), 39–53.

Neilson, K. A., Ali, N. A., Muralidharan, S., Mirzaei, M., Mariani, M., Assadourian, G., Lee, A., van Sluyter, S. C., Haynes, P. A., 2011. Less label, more free: Approaches in label-free quantitative mass spectrometry. Proteomics 11 (4), 535–553.

Nesvizhskii, A. I., Keller, A., Kolker, E., Aebersold, R., 2003. A statistical model for identifying proteins by tandem mass spectrometry. Analytical Chemistry 75 (17), 4646–4658.

Nesvizhskii, A. I., Vitek, O., Aebersold, R., 2007. Analysis and validation of proteomic data generated by tandem mass spectrometry. Nature Methods 4 (10), 787–797.

Nikolov, M., Schmidt, C., Urlaub, H., 2012. Quantitative mass spectrometry-based proteomics: an overview. Methods in Molecular Biology 893, 85–100.

Perkins, D. N., J.Pappin, D., M.Creasy, D., Cottrell, J. S., 1999. Probability-based protein identification by searching sequence databases using mass spectrometry data. Electrophoresis 20 (18), 3551–3567.

Platt, J. C., 2000. Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In: Advances in Large Margin Classifiers. MIT Press, pp. 61–74.

Ramakrishnan, S. R., Vogel, C., Kwon, T., Penalva, L. O., Marcotte, E. M., Miranker, D. P., 2009a. Mining gene functional networks to improve mass-spectrometry based protein identification. Bioinformatics 25 (22), 2955–2961.

Ramakrishnan, S. R., Vogel, C., Prince, J. T., Wang, R., Li, Z., Penalva, L. O., Myers, M., Marcotte, E. M., Miranker, D. P., 2009b. Integrating shotgun proteomics and mRNA expression data to improve protein identification. Bioinformatics 25 (11), 1397–1403.

Tabb, D. L., Fernando, C. G., Chambers, M. C., 2007. Myrimatch: highly accurate tandem mass spectral peptide identification by multivariate hypergeometric analysis. Journal of Proteome Research 6 (2), 654–661.

Tang, H., Arnold, R. J., Alves et al, P., 2006. A computational approach toward label-free protein quantification using predicted peptide detectability. Bioinformatics 22 (14), 481–488.

Teng, B., Huang, T., He, Z., 2014. Decoy-free protein-level false discovery rate estimation. Bioinformatics 30 (5), 675–681.

Yang, C., He, Z., Yu, W., 2013. A combinatorial perspective of the protein inference problem. IEEE/ACM Transactions on Computational Biology and Bioinformatics 10 (6), 1542–1547.

Zhang, Y., Wen, Z., Washburn, M. P., Florens, L., 2010. Refinements to label free proteome quantitation: How to deal with peptides shared by multiple proteins. Molecular & Cellular Proteomics 82 (6), 2272–2281.

Zhao, C., Liu, D., Teng, B., He, Z., 2015. BagReg: Protein inference through machine learning. Computational biology and chemistry 57, 12–20.