Li and Radivojac *BMC Bioinformatics* 2012, **13**(Suppl 16):S4
http://www.biomedcentral.com/1471-2105/13/S16/S4

BMC Bioinformatics

REVIEW Open Access

Computational approaches to protein inference in shotgun proteomics

Yong Fuga Li*, Predrag Radivojac

Abstract

Shotgun proteomics has recently emerged as a powerful approach to characterizing proteomes in biological samples. Its overall objective is to identify the form and quantity of each protein in a high-throughput manner by coupling liquid chromatography with tandem mass spectrometry. As a consequence of its high throughput nature, shotgun proteomics faces challenges with respect to the analysis and interpretation of experimental data. Among such challenges, the identification of proteins present in a sample has been recognized as an important computational task. This task generally consists of (1) assigning experimental tandem mass spectra to peptides derived from a protein database, and (2) mapping assigned peptides to proteins and quantifying the confidence of identified proteins. Protein identification is fundamentally a statistical inference problem with a number of methods proposed to address its challenges. In this review we categorize current approaches into rule-based, combinatorial optimization and probabilistic inference techniques, and present them using integer programing and Bayesian inference frameworks. We also discuss the main challenges of protein identification and propose potential solutions with the goal of spurring innovative research in this area.

Introduction

The main objective of mass spectrometry-based proteomics is to provide a molecular snapshot of the form (e.g. splice isoforms, post-translational modifications), abundance level, and functional aspects (e.g. protein-protein interactions, protein localization) of each protein in a biological sample [1-3]. Among proteomics strategies, bottom-up or shotgun proteomics has emerged as a high-throughput technology capable of characterizing hundreds of proteins at the same time. In this scenario, proteins in a sample are first digested into peptides, typically using site-specific proteolytic enzymes (e.g. trypsin). Peptides are then separated by liquid chromatography (LC) and analyzed by tandem mass-spectrometry (MS/MS) resulting in a set of MS/MS spectra [4]. In contrast to the top-down proteomics strategy, where intact proteins are directly analyzed through mass spectrometers, shotgun proteomics is characterized by high separation efficiency and mass spectral sensitivity. At the same time, it places higher demands on the computational and statistical techniques necessary for peptide identification, protein identification, and label-free quantification.

School of Informatics and Computing, Indiana University, Bloomington 150 S. Woodlawn Avenue, Bloomington, Indiana, 47405, USA

In a standard computational pipeline, MS/MS spectra from a mass spectrometer are searched against spectral libraries [5-8] and/or *in silico* spectra [9-14] corresponding to peptides from a protein database in order to provide *peptide-spectrum matches* (PSMs). Such a database search, depending on the parameters of the search and the MS/MS platform, can result in a large number of PSMs that are assigned scores indicating the confidence level of correct identification of the respective peptide. The next step is to assemble a list of *identified proteins* from all, or a subset of, PSMs and provide statistical confidence levels for each protein.

Protein identification is a special case of label-free protein quantification because, in an ideal scenario, each protein with a correctly inferred non-zero quantity (abundance) would be considered identified. However, label-free quantification has not yet reached the accuracy needed for the wide dynamic range of quantities observed in cellular or extracellular proteomes [15]. In addition, in many practical situations it suffices to only consider the existence of proteins in the sample and not their exact quantity. Thus, solving the more general and significantly more difficult problem of quantification to provide a solution to its subproblem may result in less accurate solutions to protein identification.

Obtaining a list of identified proteins from a set of peptide sequences with identification scores may seem straightforward. However, there are several factors that combine to challenge such intuition: (1) Usually only a small number of peptide identifications, mostly unreliable, are available for each protein [16]. This is because only the top-scoring PSMs for each peptide are typically included into the candidate set for peptide identifications, and among those candidates only a small subset are considered to be confident identifications. This leads to difficulties in providing confident protein identifications, e.g. if only a single peptide is identified from a protein. (2) Peptides, even those from the same protein, are not equally likely to be identified in a proteomics experiment [17-19]. The probability that a peptide is identified in a standard proteomics experiment has been referred to as *peptide detectability* [19], see additional file 1. (3) Many peptide sequences encountered in a typical proteomics workflow can be mapped to more than one protein in a database. These are referred to as *degenerate* or *shared peptides* [20,21]. It is a common situation that a eukaryotic sample contains more degenerate than *unique peptides*, i.e. peptides that can be mapped to only one protein. (4) It is non-trivial to estimate the false discovery rates (FDRs) of identified peptides and proteins. Some approaches to estimating peptide-level FDRs involve construction of decoy databases or use unsupervised estimation of class-conditional distributions (distributions of PSM scores given correct and false identifications, respectively). However, a large number of low-scoring PSMs may create difficulties in determining the certainty of both peptide and protein identification. While methods for the estimation of peptide-level FDRs have been well-characterized, computing protein-level FDRs remains an open problem [22,23].

The process of identifying proteins that are present in a biological sample is now widely framed as a statistical inference problem, and has been referred to as the *protein inference problem* [20,21]. To date, a number of approaches have been proposed to address this problem [20,35-37]. We categorize those approaches into three broad groups, noting that a particular method may exploit more than one strategy:

1. Rule-based strategies - methods that rely on a relatively small set of confidently identified (unique) peptides that are subsequently assigned to proteins.
2. Combinatorial optimization algorithms - methods that rely on constrained optimization formulations of the protein inference problem resulting, for example, in minimal protein lists that cover some or all confidently identified peptides.
3. Probabilistic inference algorithms - methods that formulate the problem probabilistically and assign identification probabilities for each protein in a database.

In the following sections, we provide justification for the development of advanced protein inference algorithms and then review the major computational strategies. All combinatorial optimization techniques are presented using a framework of integer programming; on the other hand, probabilistic algorithms are summarized using Bayesian inference principles. Our focus is also on the intuition behind the algorithms, the types of solutions generated, and the strengths and limitations of each method. We believe this information is essential in order to understand commonalities among the algorithms as well as their principal differences. It is also important for the proper interpretation of outputs from the various protein inference tools already applied in bottom-up proteomics.

Notation

Before discussing algorithmic details, it is important to introduce notation that will be used throughout this paper. Let us consider a set of tandem mass spectra $\mathcal{S}$ from a proteomics experiment and let $\mathcal{P} = \{P_1, P_2, \ldots\}$ be a database of proteins that the spectra are searched against. Let also $\mathcal{p} = \{p_1, p_2, \ldots\}$ be the set of all peptides in the database and, similarly, $\mathcal{p}_i$ be the set of peptides that belong to protein P_i. We now define two sets of indicator variables as follows

$$t_j = \begin{cases} 1 \text{ if peptide } p_j \text{ is confidently identified} \\ 0 \text{ otherwise} \end{cases}$$

and

$$x_j = \begin{cases} 1 & \text{if peptide } p_j \text{ is present in the sample} \\ 0 & \text{otherwise} \end{cases}$$

Confident peptide identifications can be determined in several ways, typically by using strict FDR thresholds on the top-scoring PSMs (per peptide) and are estimated using a decoy database [22] or tools such as PeptideProphet [38], which calculate the posterior probability of a correct peptide identification. When posterior probabilities are available, stringent thresholds (e.g. 0.90) can be applied directly to those probabilities. Alternatively, sufficiently high scores from various search engines [9,39-42] are sometimes used to select confident identifications.

It is important to mention that variables t_i and x_i are different. For example, a peptide p_j that is confidently identified, e.g. using an FDR threshold of 0.01, will result in setting $t_j = 1$. On the other hand, x_i can be seen as a hidden variable that is to be inferred. Accordingly, $P(y_i = 1|\mathcal{S})$ refers to the probability that peptide j is present in the sample given all the data from the mass

spectrometer. A set of confidently identified peptides, using any of the above-mentioned approaches will be denoted as $\mathcal{C} = \{p_j \mid t_j = 1\}$.

In some situations it will be necessary to consider peptides with explicit designations of their parent proteins. In those cases, the j-th peptide derived from protein P_i will be denoted as p_{ij}. Two or more such peptides will be allowed to have identical amino acid sequences. For example, peptides p_{ij} and p_{kl} $(i \neq k)$ with identical amino acid sequences will be referred to as degenerate peptides. In the context of protein inference, peptides that occur multiple times only within a single protein will not be considered degenerate. Finally, we define

$$y_i = \begin{cases} 1 \text{ if protein } P_i \text{ is present in the sample} \\ 0 \text{ otherwise} \end{cases}$$

Variable y_i can be seen as an equivalent of x_i at the protein level. Thus, $P(y_i = 1|\mathcal{S})$ is the posterior probability that protein P_i is present in the sample. The summary of notation and abbreviations is shown in Table 1.

Protein inference: significance and algorithms

Our first goal is to investigate the influence of degenerate peptides and to show that their presence is often a major factor contributing to the challenges in protein inference. We analyze several cellular and serum samples and characterize the peptide identification process. The data include cell line and plasma samples from *Homo sapiens* [16], a tissue sample from *Mus musculus* [43], as well as samples from *Saccharomyces cerevisiae* [44] and *Deinococcus radiodurans* [24]. The sets of spectra were searched using MASCOT [39] against the human IPI database (v3.35), mouse IPI database (v3.35), Saccharomyces Genome Database (R63, 05-Jan-2010), and *D. radiodurans* proteins extracted from GenBank (27-Aug-2009), respectively.

Figure 1A shows the percentage of identified peptides per protein for an FDR of 0.01 (on the unique peptide level) when using a reversed database as decoy. We observe that 32-63% of proteins are covered by only one confidently identified peptide, while 5-36% of proteins are covered by five peptides or more. Figure 1B shows the percentage of degenerate peptides in each sample. The results indicate that 57-68% of peptides in human and mouse samples are degenerate, regardless of the type of biological sample (e.g. cell line vs. tissue vs. plasma). On the other hand, the yeast and *D. radiodurans* data sets contain only 18% and 1% of degenerate peptides, respectively. Figure 1C provides the percentage of candidate proteins hit by unique peptides. In mouse and human samples more than 80% of candidate proteins are identified only with degenerate peptides. This percentage decreases to 23% for yeast and 3% for *D. radiodurans*. Finally, in Figure 1D we provide the percentage of protein groups of a particular size, where a group is formed from the set of proteins that are hit by exactly the same peptides. In accordance with previous results, most of the yeast and *D. radiodurans* candidate proteins are distinguishable; however, for human and mouse samples, between 30% and 50% of protein groups contain multiple proteins.

This analysis provides evidence that protein inference is a non-trivial problem, especially for multicellular eukaryotes that are known to contain large numbers of paralogous proteins. It also emphasizes the importance of developing sophisticated protein inference algorithms.

Rule-based approaches

With a typical LC-MS/MS experiment resulting in a potentially large number of protein identifications, concerns were raised regarding the impact of misidentified proteins on biomedical science [45]. In response to this, several guidelines were proposed regarding the standards for publishing proteomics results [46-49]. The so-called "two-peptide rule" or two-hit rule, requiring two or more confidently identified peptides to define a confident protein identification, was advocated [46,48]. The same guidelines also recommended the parsimony principle (see next Section) as an explanation for the confident peptide identifications, and suggested that "protein family" - proteins with similar sequences due to single amino acid variants, homologs, splicing variants, or annotation mistakes - should be reported as one group if the proteins share the same identified peptides.

There is a good rationale for using the two-peptide rule. In principle, one correct unique peptide should be sufficient to correctly identify a protein. However, even for the low FDR associated with a set of peptides, many individual peptides in a large data set are incorrectly identified. Furthermore, proteins identified by single peptide hits are more likely to be incorrectly identified than proteins with higher peptide coverage [45]. It was reported that FDRs for single-hit proteins can be over 10 times higher than FDRs at the PSM level [50], likely due to the clustering of correct peptide identifications to the correct proteins and the lack of clustering behavior for the incorrect peptides [50,51].

However, the two-peptide rule has been challenged [51,52]. First, while including single-hit proteins without stringent quality control can compromise specificity, ignoring such proteins will certainly compromise sensitivity [52]. Second, controlling the confidence (FDR) at the peptide level and then deducing the proteins using heuristic rules leads to undefined FDRs at the protein level [27,50-52]. On the other hand, controlling FDR directly at the protein level may rescue some of the confident single-hit proteins. Indeed, Gupta and Pevnzer

Table 1 Summary of notation and abbreviations used throughout this paper.

Notation	Description
$\mathcal{S}$	Set of all fragmentation spectra outputted by mass spectrometer
$\mathcal{S}_j$	Set of spectra identified for peptide *j*
s	A single fragmentation spectrum, $s \in \mathcal{S}$
P_i or *i*	Protein *i*
p_j or *j*	Peptide *j*
p_{ij}	Peptide *j* derived from protein *i*; used to explicitly indicate the parent protein for peptide *j*
$\mathcal{P} = \{P_1, P_2, \dots\}$	Protein database, a set of proteins used for peptide and protein identification
$\mathcal{p} = \{p_1, p_2, \dots\}$	Peptide database, the set of all (tryptic) peptides derived from $\mathcal{P}$
$\mathcal{p}_i$	Set of peptides derived from protein P_i
t_i	Indicator variable, set to 1 if peptide is p_j confidently identified
$\mathcal{C} = \{p_j \mid t_j = 1\}$	Set of peptides that are confidently identified
x_j	Indicator variable, set to 1 if $p_j \in \mathcal{p}$ is present in the sample
y_i	Indicator variable, set to 1 if $P_i \in \mathcal{P}$ is present in the sample
$x = (x_1, \dots, x_j, \dots)$	Indicator vector representing all peptides in $\mathcal{p}$
$y = (y_1, \dots, y_i, \dots)$	Indicator vector representing all proteins in $\mathcal{P}$
N(*i*)	Set of peptides mapped to protein P_i
N(*j*)	Set of proteins that contain peptide p_j
$x_{\mathcal{p}_i}$	Indicator vector representing peptides in $\mathcal{p}_i$
$\hat{P}(x_j = 1 \mid \mathcal{S}_j)$	Peptide identification probability, the probability that peptide *j* is present in the sample given the spectra identified for peptide *j*
$P(x_j = 1 \mid s)$	The probability of the PSM matching to be correct when peptide *j* is the top-scoring match of spectrum
$P(y_i = 1 \mid \mathcal{S})$	Protein posterior probabilities, the probability that protein *i* is present in the sample given all spectra
$d_{ij}(q)$	Detectability of peptide p_{ij} at some specified quantity *q*; effective detectability
$d_{ij}^0 = d_{ij}(q^0)$	Detectability of peptide p_{ij} at standard quantity q^0 ; standard detectability
d_{ij}	Detectability of peptide p_{ij}, effective detectability
NSP_{ij}	The estimated number of (identified) sibling peptides of peptide p_{ij}, used by ProteinProphet to adjust the peptide identification probability
PSM	Peptide-spectrum match; when it is clear from the context, we use PSM to also refer to the top-scoring PSM per spectrum
FDR	False discovery rate; the fraction of incorrect peptide identifications in $\mathcal{C}$ or the fraction of incorrect protein identifications in a given list outputted by a protein inference algorithm. FDR should be distinguished from the false positive rate (FPR), the fraction of all peptides (proteins) from the database that are not present in the sample but are predicted to be present (at a particular threshold).

demonstrated that using the "single-peptide rule" results in 10-40% more protein identifications compared with the two-peptide rule at a fixed FDR level [52]. The single-peptide rule simply uses the highest scoring peptide from a protein as a score for that protein, and then directly estimates FDR at the protein level (rather than at the peptide level) using decoy databases. Thus, any protein that has one or more peptides with a score above a certain threshold is deemed confident. This statement seems problematic because proteins hit by single peptides should not be reliable. However, two mediocre peptides are not necessarily better than one good peptide; thus, many proteins hit by a single peptide can be rescued with more stringent score thresholds. Since a significant portion of such proteins are correct [53], it is not surprising that the single-peptide rule leads to more protein identifications.

With the help of protein-level FDR estimation (using a decoy database), better and more complex rules may be devised to achieve even higher sensitivity. For example,

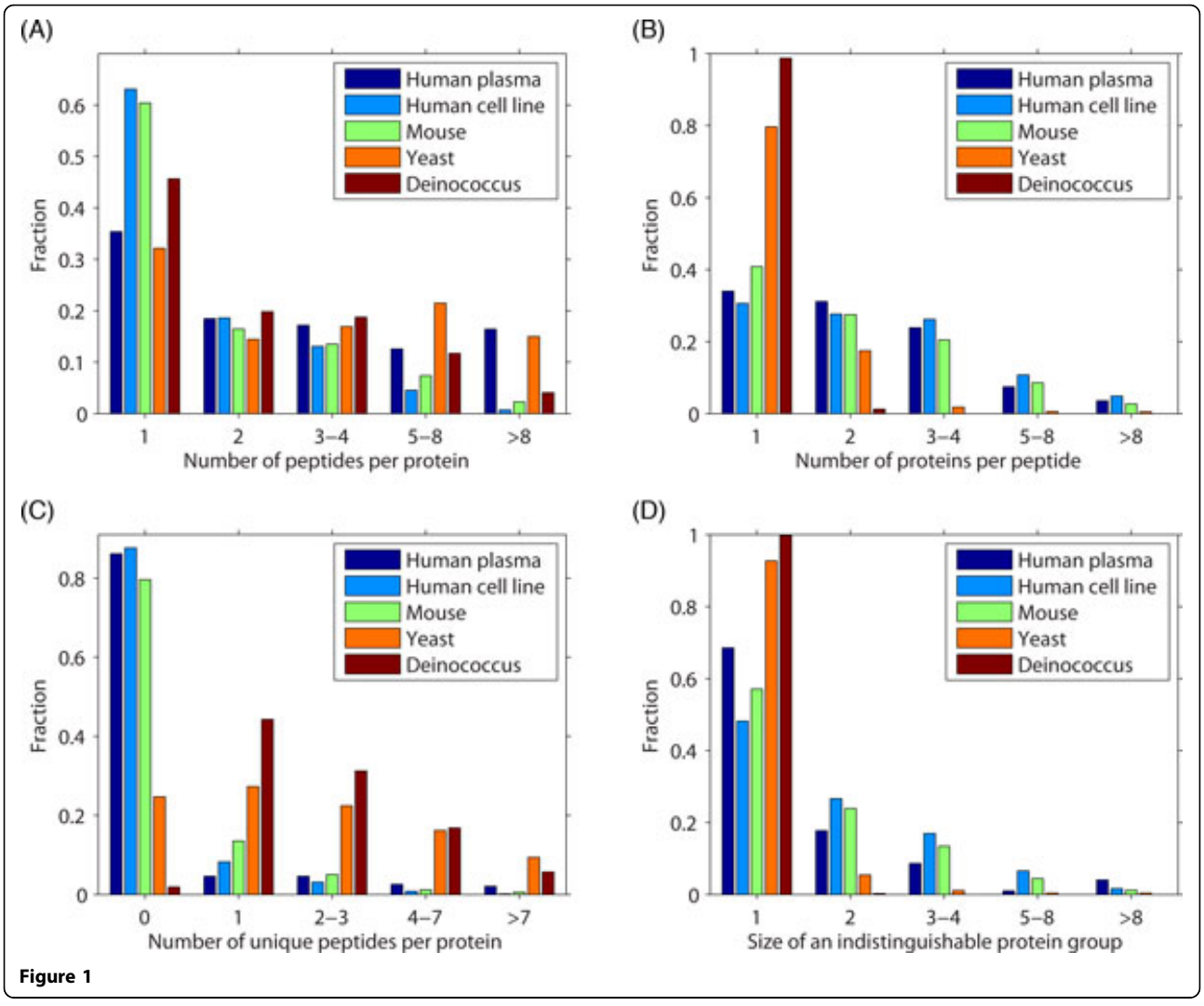


Figure 1

Weatherly et al. proposed setting separate score thresholds for proteins with different number of confident peptide identifications [51]. They reported that gradually lower score thresholds were needed for proteins with increasingly higher coverage. For the coverage of 1 (i.e. proteins hit by single peptides), a MASCOT score of 44 was required, while for coverage of 6, a MASCOT score as low as 11 was necessary for the same FDR [51].

Despite the relative simplicity of rule-based approaches, the performance of heuristic rules is fundamentally limited by the lack of rigorous treatment and proper combination of the peptide identification scores and prior knowledge.

Combinatorial optimization algorithms

The input to this class of algorithms typically consists of a set of confidently identified peptides $\mathcal{C} = \{p_j | t_j = 1\}$ and a protein database $\mathcal{P}$. The objective is to provide a list of proteins that optimizes certain criteria. In one way or another, all such formulations result in NP-hard problems and are usually solved using approximation algorithms.

The minimum set cover formulation

Minimum set cover (MSC) problem: Given a set of confident peptide identifications $\mathcal{C}$ and protein data-base $\mathcal{P}$, find a smallest protein list $\mathcal{L} \subseteq \mathcal{P}$ such that each peptide from $\mathcal{C}$ is assigned to at least one protein from $\mathcal{L}$. More formally,

$$\text{minimize} \sum_i y_i$$

$$\text{subject to} \sum_{i:p_j \in \mathcal{P}_i} y_i \geq 1 \quad (\forall p_j \in \mathcal{C}),$$

This protein inference formulation is identical to the classical computer science problem of minimum set cover, where given a set of elements (peptides) $\mathcal{U}$ and a set of subsets (proteins) over $\mathcal{U}$, the goal is to find a

smallest (not necessarily unique) set of subsets that contain all elements in $\mathcal{U}$. It is convenient to visualize the MSC formulation using bipartite graphs (Figure 2A). Using graph representation, it is relatively easy to see that an optimal solution to the MSC problem can also be provided if the original graph is divided into connected components and an optimal MSC solution provided separately for each component.

The MSC approach has been implemented in the IDPicker software [54,55]. IDPicker, however, also contains several heuristics that further simplify the solution and its interpretation. The algorithm starts by collapsing the peptide-protein bipartite graph such that all peptides/proteins connected to the same proteins/peptides form group nodes containing multiple peptides or proteins. It then finds a set of disconnected subgraphs within a bipartite graph using a depth-first search. Finally, it performs a MSC optimization in each of those subgraphs. IDPicker extends beyond algorithmic implementations, e.g. it contains modules for calculating confidently identified peptides (using an FDR-based approach), modules for combining scores from multiple search engines, as well as visualization of results.

The minimum set cover formulation is one of the most commonly encountered strategies in protein inference, and is recommended by the guidelines for publishing proteomics results [46,48]. Its intuition is to select the smallest among many possible solutions (Occam's razor, parsimony principle), which can be justified by considering the number of possible solutions when protein list consists of exactly n proteins. Assuming $n \ll |\mathcal{P}|$, the solutions of smaller sizes are selected from a smaller solution space and are therefore less likely to be spurious findings. In many practical situations, including protein inference, the MSC formulation leads to natural and acceptable solutions. However, it is not obvious that a minimalist formulation should apply to biological samples in which multiple paralogous proteins or protein isoforms may be present at the same time. This approach also ignores other available information, e.g. peptides that are not identified (all dashed edges in Figure 2B), gene functions [56] or mRNA expression levels [57].

The partial set cover formulation

Although the MSC formulation relies on a set confidently identified peptides, a subset of such peptides are expected to be incorrect identifications. This fact provides motivation for the partial set cover approaches where the goal is to find the minimum protein list that covers at least $100 \cdot c\%$ of the identified peptides, where $0 < c \leq 1$ is a user specified parameter.

Minimum partial set cover (MPSC) problem: Given a set of confident peptide identifications $\mathcal{U}$, protein database $\mathcal{P}$, and parameter c $(0 < c \leq 1)$, find a protein list $\mathcal{L}$ of minimal size such that at least $100 \cdot c\%$ of identified peptides are assigned to the proteins from $\mathcal{L}$. More formally,

$$\text{minimize} \sum_i y_i$$

$$\text{subject to} \quad z_j + \sum_{i:p_j \in \mathcal{P}_i} y_i \geq 1 \quad (\forall p_j \in \mathcal{C})$$

$$\sum_j z_j < (1-c) \cdot |\mathcal{C}|,$$

where $z_j \in \{0, 1\}$ indicates whether peptide $p_j \in \mathcal{C}$ is excluded $(z_j = 1)$ from the list of assigned peptides. Both MSC and MPSC problems are NP-hard in general. Thus, optimal solutions cannot be guaranteed in situations with a large number of identified peptides (note that each peptide from $\mathcal{C}$ adds a constraint in the problem formulation). A number of approximation algorithms have been proposed ranging from greedy algorithms to integer

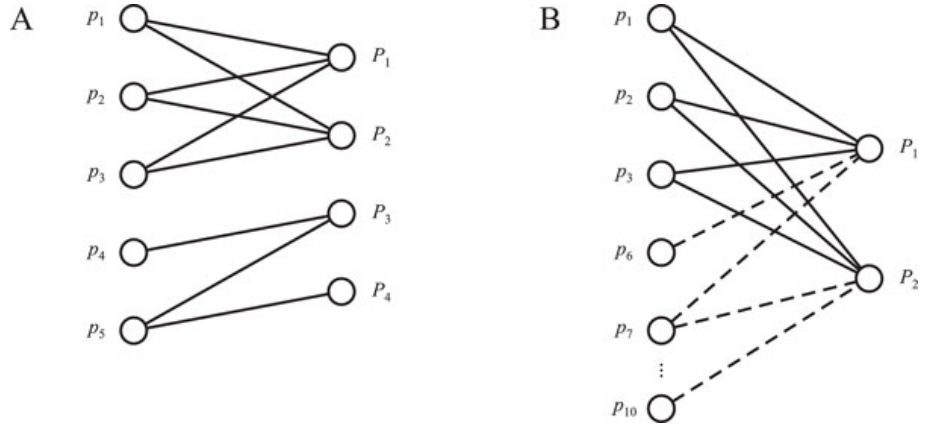


Figure 2

programming, and several such algorithms have been tested in protein inference [58].

Both the MSC and MPSC problem formulations result in situations where it is not possible to distinguish among proteins identified exclusively by degenerate peptides (e.g. proteins P_1 and P_2 in Figure 2). Nesvizhskii and Aebersold have identified several such classes of proteins, naming them indistinguishable proteins, subset proteins, subsumable proteins, etc. [20]. Because such situations are common for eukaryotes or samples containing multiple closely related organisms, different problem formulations are necessary to provide appropriate tie resolutions.

The minimum missed peptide formulation

The MSC-based formulations of the protein inference problem rely only on peptides that were confidently identified ($\mathcal{C}$) and thus ignore all unidentified peptides from the proteins containing at least one peptide from $\mathcal{C}$, see dashed edges in Figure 2B. In addition, these methods implicitly assume that each peptide is equally likely to be observed in an MS/MS experiment. The first combinatorial approach addressing these aspects was the minimum missed peptide (MMP) formulation [59]. This approach relies on the concept of peptide detectability (Box 1).

To provide intuition for the MMP approach, let us consider the example in Figure 3, which itself corresponds to the bipartite graph from Figure 2B. When considering only peptides in $\mathcal{C}$ (solid lines in Figure 2B), proteins P_1 and P_2 would be classified as indistinguishable [20]; however, given detectabilities of all peptides, it can be inferred that protein P_1 is more likely to be present in the sample than protein P_2. Specifically, the three identified peptides (shaded) are the most detectable peptides in protein P_1. On the other hand, these peptides are among the least expected peptides to be observed if protein P_2 was in the sample. Thus, protein P_1 is more likely to be a correct identification than protein P_2. Note that the tie resolution was provided by considering unidentified peptides.

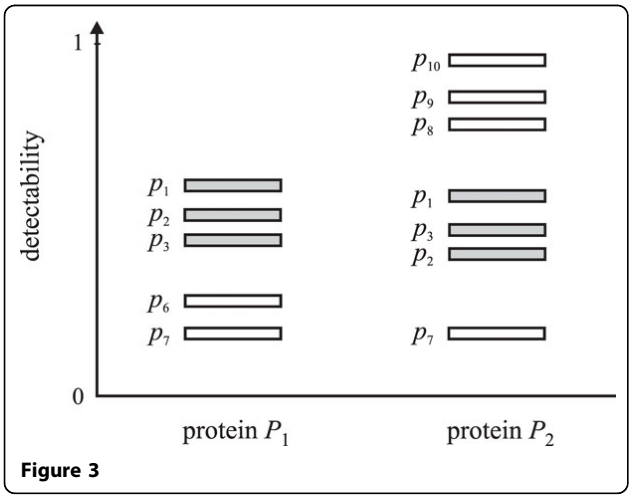


Figure 3

Before formalizing the MMP approach, let us consider a particular *solution* to the protein inference problem in which different peptides from $\mathcal{C}$ are *assigned* to protein P_i. Note that some peptides $p_{ij} \in \mathcal{C}$ may not be assigned to P_i $(x_{ij} = 0)$ although their sequence can be mapped to the protein and the peptide is confidently identified $(t_{ij} = 1)$. Peptide p_{ij} is defined as *missed* if $p_{ij} \notin \mathcal{C}$ and

$$d_{ij} \geq \min_k \{d_{ik} \mid p_{ik} \in \mathcal{C} \text{ and } x_{ik} = 1\},$$

where d_{ij} is detectability of peptide p_{ij}. In other words, a peptide is missed if in a particular inference solution (1) it is not confidently identified and (2) a peptide with lower detectability from the same protein is identified and assigned to that protein. We emphasize that the peptides with detectabilities lower than the minimum detectability of assigned peptides for protein P_i are not considered missed due to the fact that protein quantity influences effective detectability of all peptides in P_i. Thus, for effective detectability below a certain threshold, no peptides are expected to be observed. The MMP approach can now be formalized as follows.

Minimum missed peptide (MMP) problem: Given a set of confident peptide identifications $\mathcal{C}$, protein database $\mathcal{P}$, and peptide detectability for each peptide $p_j \in \mathcal{P}$, find a set of proteins $\mathcal{L} \subseteq \mathcal{P}$ that covers all peptides in $\mathcal{C}$ and minimizes the number of missed peptides. More formally,

$$\text{minimize} \sum_{i,j} z_{ij} \cdot (1 - t_j)$$

$$\text{subject to } (z_{ij} - z_{ik}) \cdot (d_{ij} - d_{ik}) \geq 0 \quad (\forall i, j \in N(i), k \in N(i))$$

$$\sum_{i \in N(j)} z_{ij} \geq t_j \quad (\forall p_j \in \mathcal{C}),$$

where $z_{ij} \in \{0, 1\}$ indicates whether detectability d_{ij} for peptide $p_{ij} \in \mathcal{C}$ is above or equal to $(z_{ij} = 1)$ or below $(z_{ij} = 0)$ the minimum detectability of peptides assigned to protein P_i and $N(i)$ is a set of peptides connected to P_i in the expanded bipartite graph (see Figure 2B). A set of identified proteins can now be determined as

$$\gamma_i = \begin{cases} 0 & \text{if } \sum_{j \in N(i)} z_{ij} \cdot t_j = 0 \\ 1 & \text{if } \sum_{j \in N(i)} z_{ij} \cdot t_j > 0 \end{cases}$$

Alves et al. have shown that the minimum cover set problem can be reduced to the minimum missed peptide formulation [59]. Thus, the MMP problem is NP-hard and approximation algorithms are needed for large-scale problems. Alves et al. proposed an efficient greedy approximation algorithm that provides a good solution [59-61]. Alternative formulations and algorithmic

approaches are also possible. For example, this algorithm can be generalized in a relatively straightforward manner to a partial set formulation or to a version that minimizes the overall probability of unidentified peptides.

Although the MMP formulation was the first protein inference technique capable of resolving indistinguishable proteins, it generally shares the limitations of other approaches based on combinatorial optimization techniques. That is, these algorithms do not provide probabilities for identified proteins, unless post-processing statistical models are used [62].

Probabilistic inference algorithms

Similarly to the previous classes of algorithms, probabilistic approaches to protein inference generally consist of two steps. First, PSM scores are converted to PSM probabilities using algorithms such as PeptideProphet [38]. After this pre-processing step, protein inference is performed based on an assumed probabilistic model. In probabilistic terms, protein inference involves computing protein posterior probabilities $P(y_i = 1|\mathcal{S})$ for every protein in $\mathcal{P}$.

Several classes of probabilistic algorithms have been proposed so far [21,24,60,61,63-71], with different strategies and levels of rigor in addressing protein groups and different run-time performance. Some probabilistic algorithms do not address degenerate peptides [63,65,68,70], while some such as ProteinProphet [21] combine probabilistic inference with the parsimony principle (for degenerate peptides) and protein grouping (for indistinguishable proteins). In the following subsections, we provide an in-depth discussion of the three major probabilistic methods: ProteinProphet [21], MSBayesPro [61], and Fido [71], and briefly mention several other methods. We use the same notation for all models and, when possible, provide new interpretations of the algorithms. We aim to reveal inherent connections and principal differences among the methods. For original derivations and interpretations, readers are referred to the original publications.

ProteinProphet

ProteinProphet is the first and most widely used probabilistic protein inference approach [21], with importance comparable to the first automated peptide identification tool, SEQUEST [9]. ProteinProphet consists of four major steps; together, they convert the original PSM probabilities from PeptideProphet to peptide identification probabilities and then combine the peptide identification probabilities to infer proteins.

Pre-processing In order to obtain protein identification probabilities, peptide identification probabilities are needed as input. Here, the difficulty is to obtain one peptide identification probability from typically multiple spectra matched to a peptide. The solution used in ProteinProphet is to simply take the maximum value among the peptide-spectrum matching probabilities for peptide j (step 1, Figure 4A), i.e.

$$P(x_j = 1|\mathcal{S}_j) = \max_{s\in\mathcal{S}_j} P(x_j = 1|s),$$

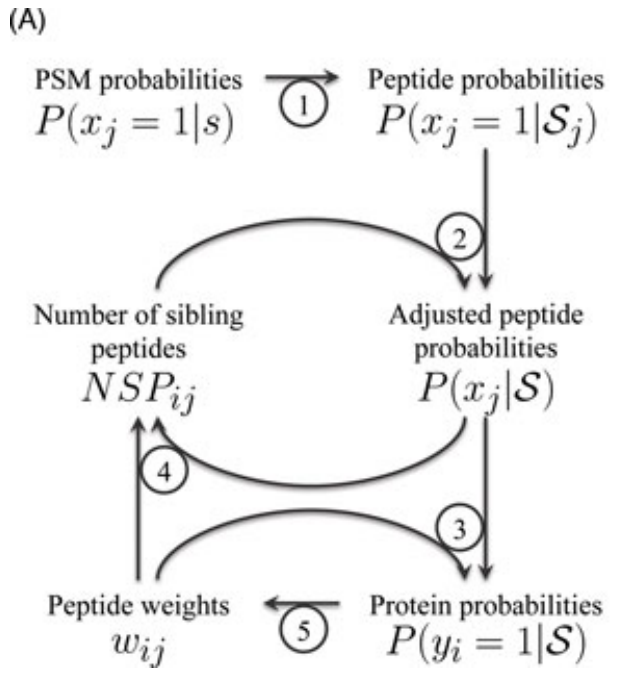
where $\mathcal{S}_j$ is the set of spectra identified for peptide j. If no spectrum is matched to the peptide, i.e. if $\mathcal{S}_j = \emptyset$, then $P(x_j = 1|\mathcal{S}_j) = 0$. Recently, the iProphet algorithm was proposed to improve this approach [72].

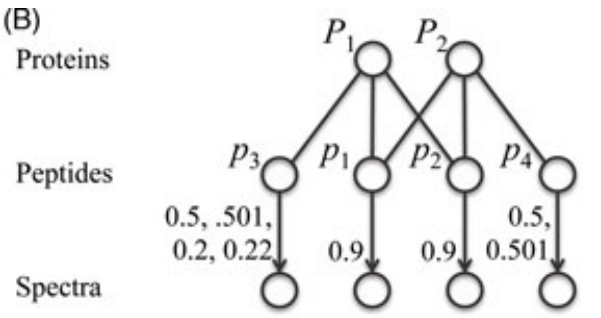
Combining peptide probabilities A key feature of ProteinProphet is that protein probabilities are computed by assuming peptide identifications to be independent pieces of evidence for the presence of protein i in the sample, i.e.

$$P(y_i = 1|\mathcal{S}) = 1 - \prod_{j\in N(i)} (1 - P(x_j = 1|\mathcal{S}_j)),$$

where $N(i)$ is the set of peptides mapped to protein i. This assumption, however, is not easy to justify because peptide identifications are not statistically independent. That is, if one peptide from the protein is confidently identified, the chance is higher that another peptide from the same protein will also be identified. Another problem with this assumption is that each degenerate peptide is counted toward all proteins it maps to. These issues are addressed via the following two adjustment steps.

Adjustment for peptide identification probability To address the limitation due to the independence assumption, ProteinProphet replaces $P(x_j = 1|\mathcal{S}_j)$ in the equation above by $P(x_j = 1|\mathcal{S})$; step 2, Figure 4A. The difference between the adjusted peptide identification probability $P(x_j = 1|\mathcal{S})$) and the original peptide identification probability $P(x_j = 1|\mathcal{S}_j)$ comes from the presence of other spectra (peptides) mapped to the same protein as peptide j. They are expected to change the confidence of peptide identification. However, it is not straightforward to estimate $P(x_j = 1|\mathcal{S})$. Nesvizhskii et al. defined the expected number of sibling peptides (NSP), i.e. the number identified peptides (other than peptide p_j) weighted by the adjusted peptide identification probability $P(x_j = 1|\mathcal{S})$), from the same protein. Specifically, $NSP_{ij} = \sum_{j'\in N(i),j'\neq j} P(x_{j'} = 1|\mathcal{S})$, where i indexes a parent protein of peptide j (step 4, Figure 4A). ProteinProphet then approximates $P(x_j = 1|\mathcal{S}) \approx P(x_j = 1|\mathcal{S}_j, NSP_{ij})$, which is computed from $P(x_j|\mathcal{S}_j)$) and $P(NSP_{ij}|x_j)$) by using the Bayes rule. Since computing NSP_{ij} requires $P(x_j = 1|\mathcal{S})$, and computing $P(x_j = 1|\mathcal{S})$ requires NSP_{ij}, iterative updating is used until convergence (steps 2, 4; Figure 4A).





Data set	Noisy input				Output		
	p_1	p_2	p_3	p_4	P_1	P_2	group
A	0.9	0.9	0.5	0.5	N/A	N/A	0.990
B	0.9	0.9	0.501	0.5	0.913	0.500	0.998
C	0.9	0.9	0.5	0.501	0.500	0.913	0.998
D	0.9	0.9	0.501	0.501	0.849	0.849	N/A
E	0.9	0.9	0.2	0.5	N/A	N/A	0.990
F	0.9	0.9	0.2	0.501	0	0.996	0.996
G	0.9	0.9	0.22	0.501	0.207	0.968	0.997

Figure 4

Adjustment for peptide degeneracy In order to address degenerate peptides, a weighting scheme is used to modify protein probabilities to

$$P(y_i = 1|\mathcal{S}) = 1 - \prod_{j \in N(i)} (1 - w_{ij} \cdot P(x_j = 1|\mathcal{S})),$$

where w_{ij} is the "proportion" of peptide j assigned to protein i (step 3, Figure 4A). Nesvizhskii et al. defined that $w_{ij} = P(y_i = 1|\mathcal{S}) / \sum_{i' \in N(j)} P(y_{i'} = 1|\mathcal{S})$, where $N(j)$ is the set of proteins that contain peptide j (step 5, Figure 4A). This adjustment step is in accordance with the parsimony principle cause $\sum_{i \in N(j)} w_{ij} = 1$, i.e. one peptide is ensured to come from only one protein in total. Note that $w_{ij} = 1$ for all unique peptides and that $w_{ij} = 0$ if peptide j cannot be mapped to protein i, i.e. when $i \notin N(j)$. Since the calculations of w_{ij} and $P(y_i = 1|\mathcal{S})$ are mutually dependent, another iterative updating procedure is used until convergence.

By combining these four steps, with a minor modification to include weights w_{ij} for peptides in the NSP adjustment step, i.e. $NSP_{ij} = \sum_{j' \in N(i), j' \neq j} w_{ij} \cdot P(x_{j'} = 1|\mathcal{S})$, protein identification probability $P(y_i = 1|\mathcal{S})$ can be approximated through a variant of the expectation-maximization (EM) iterative process (steps 2-5; Figure 4A). Since indistinguishable proteins remain indistinguishable in ProteinProphet, the grouping strategy is adopted by treating the indistinguishable proteins as one protein. Therefore, a "group probability", i.e. the probability that any one of the proteins in the group is identified, is reported.

As the first probabilistic inference method for protein identification, ProteinProphet has been very successful and, as part of the Trans-Proteomic Pipeline [73], remains the most widely used protein inference tool. Although the degenerate peptides are handled by a parsimony-driven weighting procedure, an iterative method by ProteinProphet is used to obtain those weights and ultimately results in reasonable probabilities for proteins. Recently, the tool has been improved, mainly at the preprocessing step, due to iProphet [72]. By using the same computational strategy as in the NSP adjustment step of ProteinProphet, iProphet obtains one identification probability for each peptide by aggregating the PSM probabilities of the peptide from multiple search engines, spectra, experiments, charge states, and PTM states.

Limitations Because ProteinProphet relies on certain strong assumptions, e.g. the parsimony-driven weighting (step 5, Figure 4A), its outputs are not always sensible from a statistical perspective. One such scenario was noticed by the authors [21], that for a set of proteins with shared peptides, a protein with a unique peptide, no matter how small the identification probability is, always dominates the protein(s) without unique peptides. In other words, the algorithm assigns score 1 to the protein with a random but unique peptide identification and score 0 to other proteins. This is undesirable, since there are always a large number of random peptide identifications with close to 0 probabilities in real proteomics data sets.

To address the issue, only peptides with probabilities ≥0.2 are used to compute protein probabilities. Similarly, we observed that the inference outcome of ProteinProphet is sensitive to minor changes in peptide probabilities. This can be illustrated by a simple example shown in Figure 4B. Consider two homologous proteins P_1 and P_2 with identified peptides $\{p_1, p_2, p_3\}$ and $\{p_1, p_2, p_4\}$, respectively. Suppose p_1 and p_2 are reliable identifications, but that p_3 and p_4 are not, with small identification probabilities. In the seven toy datasets (A-G) in Figure 4B, we varied the identification probability of peptides p_3 and p_4, and computed the protein probability using ProteinProphet. In data sets A and E, when the probabilities of unique peptides are not larger than 0.5, ProteinProphet considers proteins P_1 and P_2 indistinguishable, and only reports a group probability; in data set B, when probability of peptide p_3 is slightly larger than p_4 (which has probability 0.5 or less), ProteinProphet considers protein P_1 as much more reliable than P_2; in data sets C and G, when probability of peptide is (slightly) larger than p_3 (which has probability 0.5 or less), ProteinProphet considers protein P_2 as much more reliable than P_1; in data set D, when the probabilities are both larger than 0.5, ProteinProphet considers both proteins to be reliable; while in data set F, when the probability of peptide p_3 is 0.2 or less, ProteinProphet suggests that only protein P_2 can be the true protein, despite the significant probability that peptide p_4 is a random identification. This non-continuity of the inference results is counterintuitive. Naturally, one would expect the probability of protein P_2 (P_1) decreases (increases) gradually as the probability of peptide p_3 decreases.

Although ProteinProphet applies the parsimony principle to the issue of shared peptides, it uses a probabilistic model and an EM-like algorithm. Thus, ProteinProphet distinguishes itself from the other parsimony principle-driven methods, such as the combinatorial approaches discussed earlier. However, it is not clear how often ProteinProphet actually leads to the same solutions as other various combinatorial approaches regarding proteins with shared peptides. In addition, with the presence of degenerate peptides, the inference problem is difficult; thus, it would be interesting to compare the EM-like iterative algorithm used by ProteinProphet with the heuristics used by the combinatorial approaches to examine how efficiently they handle large data sets.

MSBayesPro

MSBayesPro [61] is defined as a full probabilistic protein inference method which provides "perhaps the most rigorous existing treatment of the peptide degeneracy problem" [71]. The MSBayesPro model includes peptide detectability in the probabilistic model; thus it can, to some degree, distinguish among "indistinguishable" proteins.

Model structure MSBayesPro is a Bayesian network (Figure 5) serving as a generative model for the data. The high level structure of the network is simple: Proteins → Peptides → Spectra, which mimics the experimental protocol in proteomics where proteins are first digested into peptides, from which spectra are generated. Hence,

$$P(\gamma, x, \mathcal{S}) = P(\gamma)P(x|\gamma)P(\mathcal{S}|x) \propto P(\gamma)P(x|\gamma)P(x|\mathcal{S}),$$

where y is a vector of random indicator variables for all candidate proteins, x is a vector of random indicator variables representing *all* peptides from those proteins, and $\mathcal{S}$ represents the data, i.e. all the spectra generated in the experiment. The Peptides → Spectra associations are defined by the available PSM scores (or probabilities). The Proteins → Peptides connections, however, are determined by the sequences of the peptides and

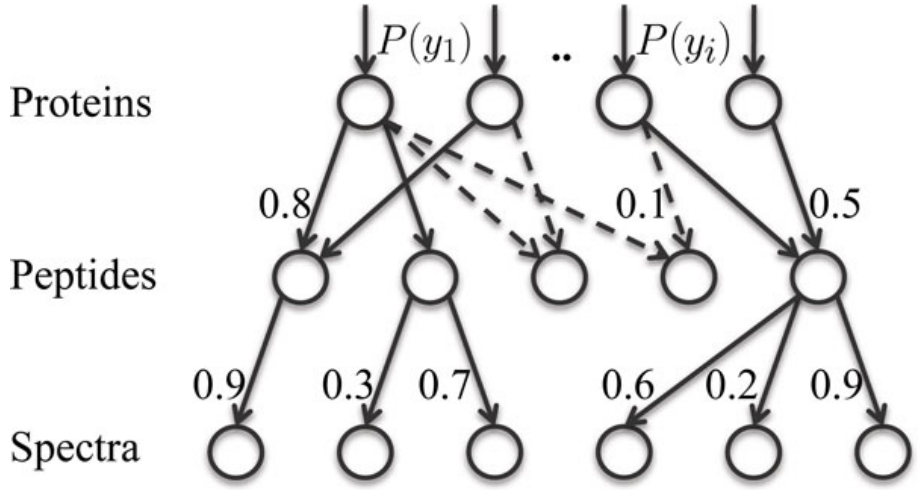


Figure 5

candidate proteins. If the sequence of peptide p_j can be exactly mapped to protein P_i, there will be an edge pointing from the protein node i to peptide node j in the network. This is similar to the structure of the model used in ProteinProphet, although the latter is not a Bayesian network. However, there is an important difference between MSBayesPro and ProteinProphet, i.e. all peptides, identified and unidentified, are included in the network structure in MSBayesPro. In contrast, the unidentified peptides are ignored in ProteinProphet and other Bayesian network models [69,71] proposed subsequently. Other than the simplification of the model structure, we believe there is no legitimate justification for excluding unidentified peptides from a probabilistic model. Such peptides will have the identification probability $P(x_j = 1|\mathcal{S}_j) = 0$; thus $x_j = 0$ is guaranteed in the inference step. We note that it is these unidentified peptides that, together with the peptide detectability information, will lead to tie resolution between grouped proteins and improve the scoring of proteins hit by single peptides.

The MSBayesPro model has an important property in that the peptide identifications are conditionally independent given the presence of the parent proteins (Figure 5). This is not to be confused with the independence assumption of peptide identification used in ProteinProphet. Actually, the conditional independence assumption in MSBayesPro will lead to marginally dependent peptide identifications if two peptides share parent proteins directly or indirectly through other peptide/protein nodes (that is, if the two peptides are in a connected component of the graph). Furthermore, the conditional independence assumption aligns with the LC-MS/MS experiment. Consider a protein P_i that is in the sample at some known abundance q_i. Then, further knowing the information that one peptide is already identified from this protein does not inform whether another peptide from the same protein should be identified in MS/MS or not. With conditional independence, we can expand the joint probabilities of the set of peptides $N(i)$ (both the identified ones and those that are not) from protein i as

$$P(x_{N(i)}|y_i = 1, y_{i'\neq i} = 0, q_i = q) = \prod_{j\in N(i)} P(x_j|y_i = 1, y_{i'\neq i} = 0, q_i = q).$$

where q_i is the abundance of protein P_i.

Model inputs and parameters MSBayesPro requires peptide identification likelihood ratios and a set of peptide detectabilities. The former is a required input to the method, and the latter, as required parameters of MSBayesPro, can be provided as an input, or ideally, peptide detectabilities should be estimated via a machine learning model from the same data set on which protein inference is carried out [24,61].

For peptide identifications, the input to MSBayesPro is the likelihood ratios $P(\mathcal{S}_i|x_i = 1)/P(\mathcal{S}_i|x_i = 0)$ rather than the peptide identification probabilities $P(x_j|\mathcal{S}_j)$ that implicitly include a uniform prior [60,61]. Here the original peptide-invariant class priors used to compute peptide identification probability are replaced in MSBayesPro by the peptide sequence and protein abundance dependent detectabilities, which are more informative priors. We note that this treatment in MSBayesPro is somewhat related to the NSP adjustment in ProteinProphet, which essentially changes the prior to incorporate information from the NSP values (interestingly, NSP values may roughly reflect protein abundances, in similar ways as effective detectability). Note that unlike detectability, NSP is not specific to the sequence of a peptide.

Using peptide detectability is an important distinguishing feature of MSBayesPro. Detectability is required to build the conditional distribution tables between the Protein and Peptide layers and subsequently to compute the posterior probabilities for the proteins. However, to use detectability properly it is important to consider the impact from protein quantity (Box 1). Li et al. [60] proposed a quantity adjustment formula to convert *standard peptide detectability* $d_{ij}^0 = P(x_j = 1|y_i = 1,\ q_i = q^0)$ to *effective detectability* $d_{ij}(q) = P(x_j = 1|y_i = 1, q_i = q)$, where q_i, the quantity of protein P_i, is estimated by the maximum likelihood or moment matching approaches. If a (degenerate) peptide p_j is shared by multiple proteins, the network structure requires combining detectabilities d_{ij} over all parent proteins of p_j. Here, MSBayesPro assumes that $d_j = 1 - \prod_{i\in N(j)} P(x_j = 0|y_i = 1,\ q_i) = 1 - \prod_{i\in N(j)} (1 - d_{ij})$. Alternative approaches in combining multiple detectabilities may also work, but the key intuition is the following: if, for a given peptide, there are multiple parent proteins all present in the sample, the detectability of the peptide should be larger than its detectability from any of the individual proteins alone. This treatment permits a non-parsimonious solution, because a degenerate peptide is allowed to come from more than one parent protein.

Inference algorithms With the Bayesian network model structure and parameters specified, it is in principle easy to exactly compute the joint posterior probability for the proteins, i.e. $P(y|\mathcal{S}) = \sum_x P(y, x|\mathcal{S})$. An optimal solution for the presence of all proteins (the maximum *a posteriori* configuration) is computed as $y_{MAP} = \text{argmax}_y\, P(y|\mathcal{S})$. The joint posterior probability can be further marginalized to compute $P(y_i|\mathcal{S})$ for the presence of each individual protein in the sample. In practice, this is not always possible due to the prohibitive time

complexity, i.e. the inference on Bayesian networks is NP-hard in general [74]. MSBayesPro uses Gibbs sampling instead of exact computation when a connected component in the Bayesian network is large (it is easy to show that connected components should be considered separately).

It is important to note that MSBayesPro also reports estimated protein quantities and the marginal posterior probabilities for peptides, which provide better scores for measuring peptide confidence [61]. Thus, in its core, MSBayesPro is also a label-free quantification algorithm. Further generalization of the MSBayesPro model has been suggested to unify the peptide and protein identification problems and perform higher-level inference on genes and pathways based on proteomics data [75].

Limitations The use of peptide detectability is both the strength and a limitation of MSBayesPro. The method requires good detectability predictions in order to achieve good performance [24]. However, prediction of detectability for non-tryptic peptides and post-translationally modified peptides is not a fully solved problem yet, which limits the applicability of MSBayesPro. In addition, detectabilities cannot be expected to provide tie resolution for proteins with nearly identical sequences. These cases, however, reveal the limits of shotgun proteomics experiments and should be addressed by follow-up experiments such as well-designed targeted proteomics experiments. Another limitation is related to the computational complexity: efficient approximation algorithms are necessary for MSBayesPro to work on very large data sets.

The Fido model

The Fido model [71,76] uses a Bayesian network, but was primarily designed for fast inference. The major contribution of this method consists of two graph transformations applied to each connected component: collapsing protein nodes that are connected to the identical sets of peptides and pruning of spectral nodes (with user specified parameters) that results in splitting of the connected components. Both transformations facilitate tradeoffs between the accuracy and speed of the inference step. Fido also allows an application of advanced probabilistic inference algorithms, e.g. the junction tree algorithm, which significantly improve protein inference on large graphs.

There are two major differences in the Bayesian network models used by Fido and MSBayesPro. First, unidentified peptides are ignored in Fido and a sequence-independent parameter is used as a replacement for peptide detectability (Figure 6). Hence, the resulting Bayesian network is simpler and inference is faster. Second, another parameter β is introduced to the model, which is the prior probability for a peptide to be identified from an artificial "noise" node. This addresses the situation where input peptide probabilities are not accurate (e.g. many incorrect peptides are assigned high probability). We believe this is a legitimate remedy for disasters that can happen during the peptide probability estimation. However, parameter β seems to be redundant given that $(1-\alpha)^{|N(j)|}$ is the probability for a peptide p_j to be identified from "noise". The authors indeed observed strong inverse correlation between the optimal values of α and β.

One limitation of the Fido model is that it requires a decoy (randomized) database to find the best values of the parameters (α, β, and γ- the prior for the presence of proteins) by combining an ROC optimization (in a supervised manner) with FDR estimation. Some versions of this approach may lead to overly optimistic performance estimates. Decoy database-independent maximum likelihood approach may be an alternative to fit the parameters. Finally, the parameter optimization step dramatically

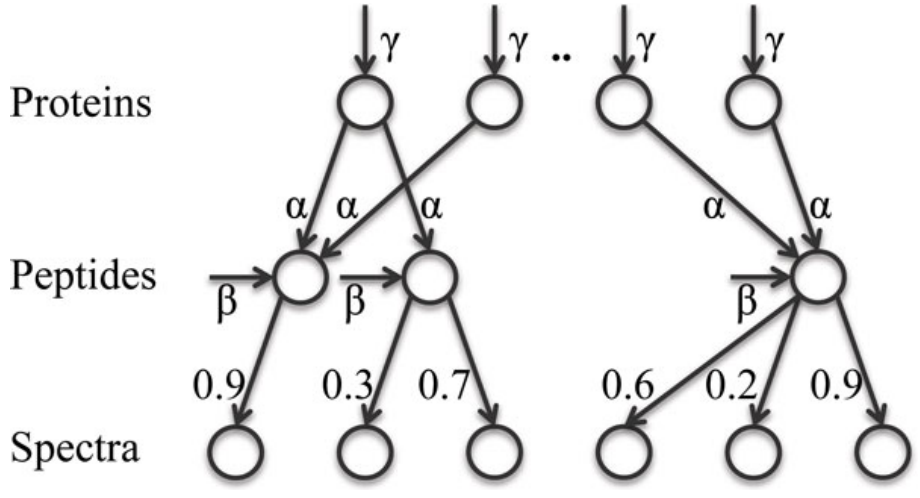


Figure 6

increases the run time of the algorithm (up to 2000 times), which compromises the overall speed of the method [71].

Other probabilistic approaches

Yang et al. recently investigated protein inference from an information retrieval (IR) point of view [68]. This work is interesting because it leverages methods in the IR field to the protein identification problem in proteomics. The authors found that the Prob-OR score, which is similar to ProteinProphet without the two adjustment steps, is dramatically worse than Prob-AND score, which is related to the protein posterior probabilities computed by MSBayesPro if degenerate peptides were treated as unique to each parent protein. We emphasize that the IR method proposed by Yang et al. is inherently a ranking approach rather than an inference approach; hence, it does not directly address the shared peptide issue as do the other probabilistic approaches discussed above.

Gerster et al. [69] recently reported a new probabilistic approach, Markovian Inference of Proteins and Gene Models (MIPGEM), that is similar to MSBayesPro and Fido. MIPGEM models peptide probabilities as random variables as in some previous approaches [66] and assumes conditional independence between peptide scores given their parent proteins (Markovian assumption). Similar to the Fido model, MIPGEM does not consider peptide detectability or unidentified peptides although the authors suggested that including detectability would be a future consideration. Table 2 provides a summary of the major probabilistic inference methods. Several other methods are reviewed in [35-37].

Discussion

Our main goal in this review was to present the challenges, intuition and proposed solutions to the protein inference problem. With increased throughput of proteomics experiments, the tools and approaches presented here will have increasingly more important applications to many problems in biology and biomedical sciences. These applications include inference and verification of gene models, identification of splice forms or post-translatioinally modified sites. Some of these problems can only be addressed using proteomics techniques and, as such, proteomics holds great promise in systems biology, biomarker discovery, diagnostics, prognostics and treatment monitoring.

Undoubtedly, there is a need for more sophisticated methodology for protein inference, unbiased performance evaluation of these techniques, as well as stand-alone tools with graphical user interface that will facilitate transition from research environments to practice in biomedical sciences. We conclude this paper by discussing the current issues in evaluating protein inference algorithms and then speculating on the ideal protein inference approaches.

Evaluation of protein identification methods

Despite the development of computational protein identification methods, objectively evaluating the performance of the methods remains a problem. Two strategies are currently available: the use of standard samples (mixtures of known proteins) and the use of decoy protein sequences to estimate FDR at the protein level. Both approaches have limitations.

To date, only a limited number of standard samples [78-80] containing 10-50 proteins have been used to facilitate evaluation of peptide/protein identification. The advantage of using standard samples is that the truth is known; thus, the accuracy measures, e.g. precision and recall, of protein identification can be directly computed. However, standard samples are frequently plagued by contaminant proteins and the boundary between true and false protein identification is blurred. Another limitation of standard samples is their small number of proteins, which leads to difficulties in assessing statistical significance in method comparisons.

The second approach estimates protein-level false discovery rates with the help of decoy databases. Although the approach has been used in several studies [51,52], two serious problems of the approach are generally ignored. We suggest that using decoy databases for evaluation of protein identification algorithms should be approached with these limitations in mind. First, unlike the decoy (e.g. reversed, randomized) database approach for peptides, the decoy database for proteins does not produce the correct estimation of the number of incorrect protein identifications when the correct proteins comprise a significant portion of the database. In an extreme scenario, when all proteins in the database are present in the sample, all the identified proteins from the forward database are correct despite many peptides being in-correct identifications. On the other hand, all identified proteins from a decoy database are incorrect. Thus, using a decoy directly will produce a non-zero FDR, while FDR = 0 is the correct answer.

This problem can be addressed by correcting for the bias due to the number of true proteins in the forward database. Let the number of identified forward and decoy proteins be n_F and n_D, and the total number of forward and decoy proteins in the databases be N_F and N_D, respectively. Let the protein level FDR in forward database be FDR_P and the rate of incorrect protein identifications from the forward and decoy database be

$$\gamma_F = \frac{FDR_P \cdot n_F}{N_F - (1 - FDR_P) \cdot n_F},$$

and

$$\gamma_D = \frac{n_D}{N_D},$$

Table 2 A comparison between different probabilistic protein inference algorithms.

Methods	ProteinProphet	MSBayesPro	Fido	MIPGEM
Underlying graph structure	Bipartite graph with identified peptides and matching proteins[1]	Bayesian network with all peptides from proteins with at least one identified peptide	Bayesian network with identified peptides and matching proteins	k-partite graph with identified peptides, matching proteins and (optionally) matching gene models[2]
Inference algorithm	EM (Expectation Maximization) like	1) Exact[3]; 2) Memorizing-Gibbs sampling	1) Exact[3] ; 2) Pruning approximation	1) Exact[3]; 2) Direct sampling
Input	Probabilities for peptides with user-defined cutoff for p (often $p > 0.05$ is used)	Likelihood ratios for peptides with $p > 0.05$ and peptide detectabilities	Likelihood ratios for peptides with $p > 0.05$	Probabilities for peptides with user-defined cutoff for p (often $p > 0.05$ is used; 0.9 for best performance)
Output	1) Protein probabilities; 2) Protein group probabilities; 3) NSP adjusted peptide probabilities	1) MAP solution, protein abundances and probabilities; 2) Protein group probabilities; 3) Posterior peptide probabilities	1) Protein probabilities; 2) Protein group probabilities	1) Protein probabilities; 2) Gene model probabilities
Protein prior estimation	No protein priors	Direct frequency estimation based on protein posterior probabilities in one run of MSBayesPro	Grid search optimizing cross-validation performance through multi-runs of Fido with different priors	Grid search optimizing model likelihood through multi-runs of the MIPGEM with different priors
Peptide probability adjustment by	NSP from a parent protein	Protein quantity adjusted peptide detectability	Two detectability-like parameters α, β	Treating peptide identifications as random variables
Protein grouping	Yes	No (indistinguishable proteins are resolved)	Yes	No (indistinguishable proteins are not resolved)
Peptide charge	Considered	Ignored	Considered	Considered
Novel aspects	1) First probabilistic protein inference algorithm; 2) Efficient EM algorithm	1) A Bayesian network; 2) Resolves indistinguishable proteins using unidentified peptides and peptide detectability; 3) Modified Gibbs sampling	1) Using a noise model to remedy inaccurate peptide probabilities; 2) Pruning algorithm, efficient inference	Gene model probabilities[4]
Availability	http://tools.proteomecenter.org	http://darwin.informatics.indiana.edu/yonli/	http://noble.gs.washington.edu/proj/fido	-

1. For ProteinProphet, the underlying bipartite graph does not correspond to a Bayesian Network although it guides the EM-like algorithm through inference.

2. MIPGEM uses a rule-based protein removal scheme to simplify the network structure;

3. Exact computation is used only for small connected components;

4. Gene centric proteomics was proposed in [77], and implemented earlier in a deterministic way in [67].

respectively. An assumption regarding a decoy database is that the rates of the false protein identifications are identical; hence, $\gamma_F = \gamma_D$. By solving this equation we find

$$FDR_P = \frac{n_D \cdot (N_F - n_F)}{n_F \cdot (N_D - n_D)}.$$

Note that there is a correction factor $(N_F - n_F)(N_D - n_D)$ in this equation compared to the FDR formula used for peptides. Also, when $N_F = n_F$, $FDR_P = 0$ as expected. A related correction is implemented in the MAYU approach [50] developed for FDR estimation from large proteomics data sets, i.e. the case when $n_F/N_F \gg 0$. Further corrections may be needed if the average lengths of the identified vs. non-identified proteins are different.

We would like to point out that, for probabilistic protein inference algorithms, theoretical protein FDR values can be computed based on the protein posterior probabilities. However, such theoretical FDR values are only accurate when the reported protein posterior probabilities are accurate. Hence, they need to be evaluated themselves, e.g. against the target/decoy-based empirical FDRs.

The second and more serious issue for applying the decoy approach is related to the existence of protein families. In fact, to our knowledge, no solution has yet been proposed. Simply speaking, a randomized database cannot serve as a good decoy for evaluating methods on data sets that contain many degenerate peptide identifications. The reason is that such peptides are typically shared among forward proteins, which could be similar to each other due to biological/annotation reasons, but not with decoy proteins. As a result, a randomized protein database cannot provide indications whether the identifications made among homologous proteins are

correct or not. For this reason, a randomized decoy database is expected to underestimate FDRs for eukaryotic samples, which have large number of shared peptides (Figure 1). The problem might be addressed using well-constructed non-random sequence database or using a closely related proteome database as decoy. Evaluating protein inference algorithms using such non-random decoys, however, remains a research problem.

We emphasize that both standard mixtures and the target/decoy approach for complex samples have their pros and cons in evaluating protein inference algorithms, and they are not mutually exclusive approaches. In fact, standard mixtures can be used to validate the target/decoy approach for protein FDR estimation. It is generally a good idea to use both strategies for a more complete and objective evaluation.

A need for guidelines for comparisons between methods

Due to the complexity of protein inference, fair evaluation of the proposed methods has been challenging. This is due to two major aspects. First, reliable and objective validation of the protein identification results is itself a challenging problem, as the FDR estimation is still unreliable. In addition, it is not even obvious how to compare models whose outputs are considerably different, e.g. those that provide protein groups and those that resolve ties between all proteins. Second, due to the lack of agreed upon guidelines, avoidable unfair comparisons are sometimes seen in the literature [69]. In other works, different peptide identification algorithms or scoring schemes are sometimes used as inputs to different protein inference methods, making the protein inference comparisons uninterpretable.

In order to address this situation, we tentatively propose the following principles for comparisons of protein inference algorithms. First, whenever possible, the same or equivalent peptide identification scores as input to different programs should be used. Second, effort should be made to provide inputs most appropriate to each algorithm considered. For example, algorithms that take all peptide identifications should be provided all scores, while programs that take only confident identifications should be provided such a subset. Third, at least one standard protein mixture data set should be used and all known proteins (whether they belong to "indistinguishable" protein groups or not) in such data sets should be included in the evaluation of the protein inference methods. This will allow the evaluation of protein inference algorithms on proteins identified without any unique peptides. Finally, and in an ideal scenario, large data sets from complex samples of unknown proteins should also be used to compare different programs; however, we caution that the current decoy database strategy may not provide reliable FDR estimates at the protein level (evaluation for protein data sets with significant fraction of degenerate peptides is a particular problem).

The ultimate protein inference approach

Despite the amount of published work, the protein inference problem is far from solved. We believe two aspects are crucial to the future approaches. First, the model should be probabilistic and with degenerate peptides treated in principled ways. Second, unidentified peptides should be exploited with peptide detectability incorporated into the model, perhaps adjusted to allow modeling peptide competition at the elution stage in a given sample. Despite the current limitations of peptide detectability predictions, especially for non-tryptic and modified peptides, it is believed that including detectability [24,35,69,71] or peptide-specific information for peptide probability adjustment [21] would improve the current methods for protein inference.

Furthermore, we believe that better estimation of peptide/protein quantity might also help protein inference by, for example, improving the quantity adjustment of peptide detectability [60,61], and provide additional input information for protein inference. As mentioned in the Introduction, protein inference can be viewed as a special case of protein label-free quantification. In fact, an ideal inference algorithm should automatically be a quantification algorithm, and vice versa. We believe much better performance can be achieved by combining the protein inference and quantification tasks into one statistical framework.

Algorithmic development is equally important for rigorous and yet practical probabilistic inference. Serang et al. [76] proposed an approximate solution by setting low peptide probabilities to zero and then applying the graph pruning procedure. In this way the complexity of the problem can be controlled at arbitrarily low levels with the price of potentially high error (i.e. the computed probability may greatly deviate from the exact values). The Gibbs sampling approach implemented in MSBayesPro can achieve arbitrarily high accuracy in probability estimation; however, the time required for the inference can be prohibitively long. A fast algorithm with controllable error bound is desirable. Applying well-established exact or approximate graph inference algorithms, e.g. the junction tree algorithm [76], is an important direction for further investigation.

Additional material

Additional file 1: Peptide detectability.

Acknowledgements

We thank Prof. Matthew Hahn, Prof. Haixu Tang and Dr. Sujun Li for the comments and help in writing this paper. We also thank the anonymous reviewers for their suggestions and criticisms that further improved the paper. This work was supported by the National Institutes of Health grants RR024236-01A1 and CA126480-01.

This article has been published as part of *BMC Bioinformatics* Volume 13 Supplement 16, 2012: Statistical mass spectrometry-based proteomics. The full contents of the supplement are available online at http://www.biomedcentral.com/1471-2105/13/S16.

Competing interests
The authors declare that they have no competing interests.

Published: 5 November 2012

References

1. Aebersold R, Mann M: **Mass spectrometry-based proteomics.** *Nature* 2003, **422(6928)**:198-207.
2. Cravatt BF, Simon GM, Yates JR: **The biological impact of mass-spectrometry-based proteomics.** *Nature* 2007, **450(7172)**:991-1000.
3. Choudhary C, Mann M: **Decoding signalling networks by mass spectrometry-based proteomics.** *Nat Rev Mol Cell Biol* 2010, **11(6)**:427-439.
4. Steen H, Mann M: **The ABC's (and XYZ's) of peptide sequencing.** *Nat Rev Mol Cell Biol* 2004, **5(9)**:699-711.
5. Craig R, Cortens JC, Fenyo D, Beavis RC: **Using annotated peptide mass spectrum libraries for protein identification.** *J Proteome Res* 2006, **5(8)**:1843-1849.
6. Frewen BE, Merrihew GE, Wu CC, Noble WS, MacCoss MJ: **Analysis of peptide MS/MS spectra from large-scale proteomics experiments using spectrum libraries.** *Anal Chem* 2006, **78(16)**:5678-5684.
7. Lam H, Deutsch EW, Eddes JS, Eng JK, King N, Stein SE, Aebersold R: **Development and validation of a spectral library searching method for peptide identification from MS/MS.** *Proteomics* 2007, **7(5)**:655-667.
8. Lam H, Deutsch EW, Eddes JS, Eng JK, Stein SE, Aebersold R: **Building consensus spectral libraries for peptide identification in proteomics.** *Nat Methods* 2008, **5(10)**:873-875.
9. Eng JK, McCormack AL, Yates JR: **An approach to correlate tandem mass spectral data of peptides with amino acid sequences in a protein database.** *J Am Soc Mass Spectrom* 1994, **5**:976-989.
10. Zhang Z: **Prediction of low-energy collision-induced dissociation spectra of peptides.** *Anal Chem* 2004, **76(14)**:3908-3922.
11. Zhang Z: **Prediction of low-energy collision-induced dissociation spectra of peptides with three or more charges.** *Anal Chem* 2005, **77(19)**:6364-6373.
12. Elias JE, Gibbons FD, King OD, Roth FP, Gygi SP: **Intensity-based protein identification by machine learning from a library of tandem mass spectra.** *Nat Biotechnol* 2004, **22(2)**:214-219.
13. Arnold RJ, Jayasankar N, Aggarwal D, Tang H, Radivojac P: **A machine learning approach to predicting peptide fragmentation spectra.** *Pac Symp Biocomput* 2006, 219-230.
14. Klammer AA, Reynolds SM, Bilmes JA, MacCoss MJ, Noble WS: **Modeling peptide fragmentation with dynamic Bayesian networks for peptide identification.** *Bioinformatics* 2008, **24(13)**:i348-356.
15. Anderson NL, Anderson NG: **The human plasma proteome: history, character, and diagnostic prospects.** *Mol Cell Proteomics* 2002, **1(11)**:845-867.
16. Resing KA, Meyer-Arendt K, Mendoza AM, Aveline-Wolf LD, Jonscher KR, Pierce KG, Old WM, Cheung HT, Russell S, Wattawa JL, *et al*: **Improving reproducibility and sensitivity in identifying human proteins by shotgun proteomics.** *Anal Chem* 2004, **76(13)**:3556-3568.
17. Le Bihan T, Robinson MD, Figeys D: **Definition and characterization of a "trypsinosome" from specific peptide characteristics by nano-HPLC-MS/MS and in silico analysis of complex protein mixtures.** *J Proteome Res* 2004, **3(6)**:1138-1148.
18. Kuster B, Schirle M, Mallick P, Aebersold R: **Scoring proteomes with proteotypic peptide probes.** *Nat Rev Mol Cell Biol* 2005, **6(7)**:577-583.
19. Tang H, Arnold RJ, Alves P, Xun Z, Clemmer DE, Novotny MV, Reilly JP, Radivojac P: **A computational approach toward label-free protein quantification using predicted peptide detectability.** *Bioinformatics* 2006, **22(14)**:e481-e488.
20. Nesvizhskii AI, Aebersold R: **Interpretation of shotgun proteomic data: the protein inference problem.** *Mol Cell Proteomics* 2005, **4(10)**:1419-1440.
21. Nesvizhskii AI, Keller A, Kolker E, Aebersold R: **A statistical model for identifying proteins by tandem mass spectrometry.** *Anal Chem* 2003, **75(17)**:4646-4658.
22. Elias JE, Gygi SP: **Target-decoy search strategy for increased confidence in large-scale protein identifications by mass spectrometry.** *Nat Methods* 2007, **4(3)**:207-214.
23. Nesvizhskii AI: **A survey of computational methods and error rate estimation procedures for peptide and protein identification in shotgun proteomics.** *J Proteomics* 2010, **73(11)**:2092-2123.
24. Li YF, Arnold RJ, Tang H, Radivojac P: **The importance of peptide detectability for protein identification, quantification, and experiment design in MS/MS proteomics.** *J Proteome Res* 2010, **9(12)**:6288-6297.
25. Alves P, Arnold RJ, Clemmer DE, Li Y, Reilly JP, Sheng Q, Tang H, Xun Z, Zeng R, Radivojac P: **Fast and accurate identification of semi-tryptic peptides in shotgun proteomics.** *Bioinformatics* 2008, **24(1)**:102-109.
26. Walsh CT: *Posttranslational modification of proteins: expanding nature's inventory* Englewood, CO: Roberts and Company Publishers; 2006.
27. Balgley BM, Laudeman T, Yang L, Song T, Lee CS: **Comparative evaluation of tandem MS search algorithms using a target-decoy search strategy.** *Mol Cell Proteomics* 2007, **6(9)**:1599-1608.
28. Lu P, Vogel C, Wang R, Yao X, Marcotte EM: **Absolute protein expression profiling estimates the relative contributions of transcriptional and translational regulation.** *Nat Biotechnol* 2007, **25(1)**:117-124.
29. Wedge DC, Gaskell SJ, Hubbard SJ, Kell DB, Lau KW, Eyers C: **Peptide detectability following ESI mass spectrometry: prediction using genetic programming.** *Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation (GECCO): 2007; New York, NY* 2007, 2219-2225.
30. Mallick P, Schirle M, Chen SS, Flory MR, Lee H, Martin D, Ranish J, Raught B, Schmitt R, Werner T, *et al*: **Computational prediction of proteotypic peptides for quantitative proteomics.** *Nat Biotechnol* 2007, **25(1)**:125-131.
31. Sanders WS, Bridges SM, McCarthy FM, Nanduri B, Burgess SC: **Prediction of peptides observable by mass spectrometry applied at the experimental set level.** *BMC Bioinformatics* 2007, **8(Suppl 7)**:S23.
32. Vogel C, Marcotte EM: **Calculating absolute and relative protein abundance from mass spectrometry-based protein expression data.** *Nat Protoc* 2008, **3(9)**:1444-1451.
33. Webb-Robertson BJ, Cannon WR, Oehmen CS, Shah AR, Gurumoorthi V, Lipton MS, Waters KM: **A support vector machine model for the prediction of proteotypic peptides for accurate mass and time proteomics.** *Bioinformatics* 2008, **24(13)**:1503-1509.
34. Bohrer BC, Li YF, Reilly JP, Clemmer DE, DiMarchi RD, Radivojac P, Tang H, Arnold RJ: **Combinatorial libraries of synthetic peptides as a model for shotgun proteomics.** *Anal Chem* 2010, **82(15)**:6559-6568.
35. Shi J, Wu F: **Protein inference by assembling peptides identified from tandem mass spectra.** *Curr Bioinformatics* 2009, **4(3)**:226-233.
36. Huang T, Wang J, Yu W, He Z: **Protein inference: a review.** *Brief Bioinform* 2012.
37. Serang O, Noble WS: **A review of statistical methods for protein identification using tandem mass spectrometry.** *Stat Interface* 2012, **5(1)**:3-20.
38. 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.** *Anal Chem* 2002, **74(20)**:5383-5392.
39. Perkins DN, Pappin DJ, Creasy DM, Cottrell JS: **Probability-based protein identification by searching sequence databases using mass spectrometry data.** *Electrophoresis* 1999, **20(18)**:3551-3567.
40. Tanner S, Shu H, Frank A, Wang LC, Zandi E, Mumby M, Pevzner PA, Bafna V: **InsPecT: identification of posttranslationally modified peptides from tandem mass spectra.** *Anal Chem* 2005, **77(14)**:4626-4639.
41. Tabb DL, Fernando CG, Chambers MC: **MyriMatch: highly accurate tandem mass spectral peptide identification by multivariate hypergeometric analysis.** *J Proteome Res* 2007, **6(2)**:654-661.
42. Li W, Ji L, Goya J, Tan G, Wysocki VH: **SQID: an intensity-incorporated protein identification algorithm for tandem mass spectrometry.** *J Proteome Res* 2011, **10(4)**:1593-1602.
43. Li S, Arnold RJ, Tang H, Radivojac P: **On the accuracy and limits of peptide fragmentation spectrum prediction.** *Anal Chem* 2011, **83(3)**:790-796.
44. Tabb DL, Vega-Montoto L, Rudnick PA, Variyath AM, Ham AJ, Bunk DM, Kilpatrick LE, Billheimer DD, Blackman RK, Cardasis HL, *et al*: **Repeatability and reproducibility in proteomic identifications by liquid chromatography-tandem mass spectrometry.** *J Proteome Res* 2010, **9(2)**:761-776.

45. Baldwin MA: **Protein identification by mass spectrometry: issues to be considered.** *Mol Cell Proteomics* 2004, **3(1)**:1-9.
46. Carr S, Aebersold R, Baldwin M, Burlingame A, Clauser K, Nesvizhskii A: **The need for guidelines in publication of peptide and protein identification data: Working Group on Publication Guidelines for Peptide and Protein Identification Data.** *Mol Cell Proteomics* 2004, **3(6)**:531-533.
47. Bradshaw RA, Burlingame AL, Carr S, Aebersold R: **Reporting protein identification data: the next generation of guidelines.** *Mol Cell Proteomics* 2006, **5(5)**:787-788.
48. Wilkins MR, Appel RD, Van Eyk JE, Chung MC, Gorg A, Hecker M, Huber LA, Langen H, Link AJ, Paik YK, *et al*: **Guidelines for the next 10 years of proteomics.** *Proteomics* 2006, **6(1)**:4-8.
49. Jones AR, Orchard S: **Minimum reporting guidelines for proteomics released by the Proteomics Standards Initiative.** *Mol Cell Proteomics* 2008, **7(10)**:2067-2068.
50. Reiter L, Claassen M, Schrimpf SP, Jovanovic M, Schmidt A, Buhmann JM, Hengartner MO, Aebersold R: **Protein identification false discovery rates for very large proteomics data sets generated by tandem mass spectrometry.** *Mol Cell Proteomics* 2009, **8(11)**:2405-2417.
51. Weatherly DB, Atwood JA, Minning TA, Cavola C, Tarleton RL, Orlando R: **A heuristic method for assigning a false-discovery rate for protein identifications from Mascot database search results.** *Mol Cell Proteomics* 2005, **4(6)**:762-772.
52. Gupta N, Pevzner PA: **False discovery rates of protein identifications: a strike against the two-peptide rule.** *J Proteome Res* 2009, **8(9)**:4173-4181.
53. Gupta N, Benhamida J, Bhargava V, Goodman D, Kain E, Kerman I, Nguyen N, Ollikainen N, Rodriguez J, Wang J, *et al*: **Comparative proteogenomics: combining mass spectrometry and comparative genomics to analyze multiple genomes.** *Genome Res* 2008, **18(7)**:1133-1142.
54. Zhang B, Chambers MC, Tabb DL: **Proteomic parsimony through bipartite graph analysis improves accuracy and transparency.** *J Proteome Res* 2007, **6(9)**:3549-3557.
55. Ma ZQ, Dasari S, Chambers MC, Litton MD, Sobecki SM, Zimmerman LJ, Halvey PJ, Schilling B, Drake PM, Gibson BW, *et al*: **IDPicker 2.0: Improved protein assembly with high discrimination peptide identification filtering.** *J Proteome Res* 2009, **8(8)**:3872-3881.
56. Ramakrishnan SR, Vogel C, Kwon T, Penalva LO, Marcotte EM, Miranker DP: **Mining gene functional networks to improve mass-spectrometry-based protein identification.** *Bioinformatics* 2009, **25(22)**:2955-2961.
57. Ramakrishnan SR, Vogel C, Prince JT, Li Z, Penalva LO, Myers M, Marcotte EM, Miranker DP, Wang R: **Integrating shotgun proteomics and mRNA expression data to improve protein identification.** *Bioinformatics* 2009, **25(11)**:1397-1403.
58. He Z, Yang C, Yu W: **A partial set covering model for protein mixture identification using mass spectrometry data.** *IEEE/ACM Trans Comput Biol Bioinform* 2011, **8(2)**:368-380.
59. Alves P, Arnold RJ, Novotny MV, Radivojac P, Reilly JP, Tang H: **Advancements in protein identification from shotgun proteomics using predicted peptide detectability.** *Pac Symp Biocomput* 2007, **12**:409-420.
60. Li YF, Arnold RJ, Li Y, Radivojac P, Sheng Q, Tang H: **A Bayesian approach to protein inference problem in shotgun proteomics.** *The 12th Annual International Conference on Research in Computational Molecular Biology, RECOMB 2008: 2008; Singapore* 2008, 167-180.
61. Li YF, Arnold RJ, Li Y, Radivojac P, Sheng Q, Tang H: **A Bayesian approach to protein inference problem in shotgun proteomics.** *J Comput Biol* 2009, **16(8)**:1183-1193.
62. Sadygov RG, Liu H, Yates JR: **Statistical models for protein validation using tandem mass spectral data and protein amino acid sequence databases.** *Anal Chem* 2004, **76(6)**:1664-1671.
63. Qian WJ, Liu T, Monroe ME, Strittmatter EF, Jacobs JM, Kangas LJ, Petritis K, Camp DG, Smith RD: **Probability-based evaluation of peptide and protein identifications from tandem mass spectrometry and SEQUEST analysis: the human proteome.** *J Proteome Res* 2005, **4(1)**:53-62.
64. Feng J, Naiman DQ, Cooper B: **Probability model for assessing proteins assembled from peptide sequences inferred from tandem mass spectrometry data.** *Anal Chem* 2007, **79(10)**:3901-3911.
65. Price TS, Lucitt MB, Wu W, Austin DJ, Pizarro A, Yocum AK, Blair IA, FitzGerald GA, Grosser T: **EBP, a program for protein identification using multiple tandem mass spectrometry datasets.** *Mol Cell Proteomics* 2007, **6(3)**:527-536.
66. Shen C, Wang Z, Shankar G, Zhang X, Li L: **A hierarchical statistical model to assess the confidence of peptides and proteins inferred from tandem mass spectrometry.** *Bioinformatics* 2008, **24(2)**:202-208.
67. Grobei MA, Qeli E, Brunner E, Rehrauer H, Zhang R, Roschitzki B, Basler K, Ahrens CH, Grossniklaus U: **Deterministic protein inference for shotgun proteomics data provides new insights into Arabidopsis pollen development and function.** *Genome Res* 2009, **19(10)**:1786-1800.
68. Yang Y, Harpale A, Ganapathy S: **Protein identification from tandem mass spectra with probabilistic language modeling.** *Machine Learning and Knowledge Discovery in Databases* 2009, 554-569.
69. Gerster S, Qeli E, Ahrens CH, Buhlmann P: **Protein and gene model inference based on statistical modeling in k-partite graphs.** *Proc Natl Acad Sci USA* 2010, **107(27)**:12101-12106.
70. Li Q, MacCoss M, Stephens M: **A nested mixture model for protein identification using mass spectrometryA nested mixture model for protein identification using mass spectrometry.** *Annals* 2010, **4(2)**:962-987.
71. Serang O, MacCoss MJ, Noble WS: **Efficient marginalization to compute protein posterior probabilities from shotgun mass spectrometry data.** *J Proteome Res* 2010, **9(10)**:5346-5357.
72. Shteynberg D, Deutsch EW, Lam H, Eng JK, Sun Z, Tasman N, Mendoza L, Moritz RL, Aebersold R, Nesvizhskii AI: **iProphet: multi-level integrative analysis of shotgun proteomic data improves peptide and protein identification rates and error estimates.** *Mol Cell Proteomics* 2011, **10(12)**: M111 007690.
73. Deutsch EW, Mendoza L, Shteynberg D, Farrah T, Lam H, Tasman N, Sun Z, Nilsson E, Pratt B, Prazen B, *et al*: **A guided tour of the Trans-Proteomic Pipeline.** *Proteomics* 2010, **10(6)**:1150-1159.
74. Cooper G: **Probabilistic inference using belief etworks is NP-hard.** *Artificial Intelligence* 1990, **42(2-3)**:393-405.
75. Li YF, Arnold RJ, Radivojac P, Tang H: **Protein identification problem from a Bayesian point of view.** *Stat Interface* 2012, **5(1)**:21-38.
76. Serang O, Noble WS: **Faster mass spectrometry-based protein inference: junction trees are more efficient than sampling and marginalization by enumeration.** *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 2012.
77. Rappsilber J, Mann M: **What does it mean to identify a protein in proteomics?** *Trends Biochem Sci* 2002, **27(2)**:74-78.
78. Keller A, Purvine S, Nesvizhskii AI, Stolyar S, Goodlett DR, Kolker E: **Experimental protein mixture for validating tandem mass spectral analysis.** *OMICS* 2002, **6(2)**:207-212.
79. Purvine S, Picone AF, Kolker E: **Standard mixtures for proteome studies.** *OMICS* 2004, **8(1)**:79-92.
80. Klimek J, Eddes JS, Hohmann L, Jackson J, Peterson A, Letarte S, Gafken PR, Katz JE, Mallick P, Lee H, *et al*: **The standard protein mix database: a diverse data set to assist in the production of improved Peptide and protein identification software tools.** *J Proteome Res* 2008, **7(1)**:96-103.

doi:10.1186/1471-2105-13-S16-S4
Cite this article as: Li and Radivojac: Computational approaches to protein inference in shotgun proteomics. *BMC Bioinformatics* 2012 **13**(Suppl 16):S4.