Proteomics Seminar 2014SS

Xiao Liang

April 29, 2014

Data and text mining

A linear programming model for protein inference problem in shotgun proteomics

Ting Huang and Zengyou He*

School of Software, Dalian University of Technology, Dalian 116621, China

Advance Access publication September 6, 2012
Part I

Protein Inference Problem
Protein inference in shotgun proteomics experiment

Figure: Protein inference using mass spectrometry data.

Goal: Find a subset of proteins that are truly present in the sample.

---

What input do we need for protein inference?

- A list of identified peptides.
  1. Database-driven approach
  2. de novo algorithm

- Peptide probabilities (detectabilities). <- rigorous statistical validation
  PeptideProphet$^2$ estimates $Pr(+|S)$: the probability that the peptide assignment with discriminant score $S$ is correct.

- A list of candidate proteins.

- Expected output: a set of proteins accompanying protein probabilities.

Challenge: Peptide degeneracy

Peptide degeneracy: a single peptide mapped to multiple proteins.

<table>
<thead>
<tr>
<th>Peptides</th>
<th>Proteins</th>
<th>Peptide probabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASTSSSSSSSSNQQTEKETNTPK</td>
<td>P51965</td>
<td>UB2E1_HUMAN</td>
</tr>
<tr>
<td>YEWRSTILGPPGSVY</td>
<td>P51965</td>
<td>UB2E1_HUMAN</td>
</tr>
<tr>
<td>YEWRSTILGPPGSVY</td>
<td>Q96LR5</td>
<td>UB2E2_HUMAN</td>
</tr>
<tr>
<td>YEWRSTILGPPGSVY</td>
<td>Q969T4</td>
<td>UB2E3_HUMAN</td>
</tr>
<tr>
<td>VLLSICSLTDCNPADPLVGSIATQYMTNR</td>
<td>P51965</td>
<td>UB2E1_HUMAN</td>
</tr>
</tbody>
</table>

Figure: Peptide identifications (Sigma49 data)

- Shared peptides should belong to all proteins that they can match.
Challenge: Peptide degeneracy

Peptide degeneracy: a single peptide mapped to multiple proteins.

<table>
<thead>
<tr>
<th>Peptides</th>
<th>Proteins</th>
<th>Peptide probabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASTSSSSSSSNQQTEKETNTPK</td>
<td>P51965</td>
<td>UB2E1_HUMAN</td>
</tr>
<tr>
<td>YEWRSTILGPPGSVY</td>
<td>P51965</td>
<td>UB2E1_HUMAN</td>
</tr>
<tr>
<td>YEWRSTILGPPGSVY</td>
<td>Q96LR5</td>
<td>UB2E2_HUMAN</td>
</tr>
<tr>
<td>YEWRSTILGPPGSVY</td>
<td>Q969T4</td>
<td>UB2E3_HUMAN</td>
</tr>
</tbody>
</table>
| VLLSICSLTDNCNPA
DPLVGSIA
TQYMTNR      | P51965|UB2E1_HUMAN         | 1.0000                |

Figure: Peptide identifications (Sigma49 data)

- Shared peptides should belong to all proteins that they can match.
- Conditional probability: model the conditional probability of
  - one protein being present given a peptide,
  - one peptide being present given a protein.
Existing protein inference algorithms

- ProteinProphet\(^3\) calculates the conditional probability. Given peptides \(i, i = 1 \cdots n\), with probabilities \(Pr(+|S_i)\) corresponding to a protein, the probability \(p\) that this protein is present:

\[
p = 1 - \prod_{i}^{n} [1 - Pr(+|S_i)].
\] (1)

- Fido\(^4\) estimates the protein posterior error probability.

\[
p = Pr(+|\text{protein}).
\] (2)

---


Part II

Protein linear programming (ProteinLP)
Model

Table 1. Notations and definitions

<table>
<thead>
<tr>
<th>Notations</th>
<th>Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>((1, \ldots, i, \ldots, n))</td>
<td>All (n) peptides identified by peptide identification algorithms</td>
</tr>
<tr>
<td>((1, \ldots, j, \ldots, m))</td>
<td>All (m) proteins that might have generated these (n) peptides</td>
</tr>
<tr>
<td>((y_1, \ldots, y_i, \ldots, y_n))</td>
<td>Peptide vector: indicator variables of peptides’ presences if peptide (i) is present, (y_i = 1); otherwise (y_i = 0)</td>
</tr>
<tr>
<td>((x_1, \ldots, x_j, \ldots, x_m))</td>
<td>Protein vector: indicator variables of proteins’ presences</td>
</tr>
<tr>
<td>((z_1, \ldots, z_i, \ldots, z_n))</td>
<td>The probabilities of peptides’ presences estimated by peptide identification algorithms or PeptideProphet</td>
</tr>
</tbody>
</table>

- \(Pr(x_j = 1)\): the probability that protein \(j\) is present in the sample.
- \(Pr(y_i = 1, x_j = 1)\): the probability that peptide \(i\) and protein \(j\) are present in the sample.

\[
Pr(x_j = 1) = 1 - \prod_{i=1}^{n}[1 - Pr(y_i = 1, x_j = 1)]
\]  

(3)
Model

- From Eq. 3:

\[
Pr(x_j = 1) = 1 - \prod_{i=1}^{n}[1 - Pr(y_i = 1, x_j = 1)] = 1 - \prod_{i=1}^{n} e^{\ln[1 - Pr(y_i = 1, x_j = 1)]}.
\] (4)

The protein probability is rewritten as:

\[
Pr(x_j = 1) = 1 - \prod_{i=1}^{n} e^{p_{ij}},
\] (5)

where \( p_{ij} := \ln[1 - Pr(y_i = 1, x_j = 1)] \leq 0. \)

- The peptide probability:

\[
Pr(y_i = 1) = 1 - \prod_{j=1}^{m}[1 - Pr(y_i = 1, x_j = 1)] = 1 - \prod_{j=1}^{m} e^{p_{ij}}.
\] (6)

\[
z_i = 1 - \prod_{j=1}^{m} e^{p_{ij}}
\] (7)
**LP formulation**

**Objective:**

Maximize the number of proteins with zero probabilities,
while peptide probabilities from joint probabilities should be as close to the input value as possible.

Maximize: \[ \sum_{j=1}^{m} t_j, \] \hspace{1cm} (8)

Subject to: \[ \forall i: t_j \leq p_{ij} \leq 0, \] \hspace{1cm} (9)

\[ \forall i: \ln(1 - z_i - \epsilon) \leq \sum_{j=1}^{m} p_{ij} \leq \ln(1 - z_i + \epsilon), \] \hspace{1cm} (10)

\[ p_{ij} = 0, \quad \text{if protein j doesn't contain peptide i.} \] \hspace{1cm} (11)
LP formulation

\[ P = (p_{ij})_{m \times m} = \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1m} \\ p_{21} & p_{22} & \cdots & p_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ p_{m1} & p_{m2} & \cdots & p_{mm} \end{pmatrix} \]

Column constraints \( \forall j, i: p_{ij} \geq t_j \)

Row constraints \( \forall i: \)
\[
\begin{align*}
\ln(1-z_i - \varepsilon) & \leq \sum_{j=1}^{m} p_{ij} \\
\ln(1-z_i + \varepsilon) & \geq \sum_{j=1}^{m} p_{ij}
\end{align*}
\]

\[ \text{Constraint (11):} \]
\[ p_{ij} = 0 \text{ if } Pr(y_i = 1, x_j = 1) = 0. \]
LP formulation

\[ P = (p_{ij})_{m \times m} = \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1m} \\ p_{21} & p_{22} & \cdots & p_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ p_{m1} & p_{m2} & \cdots & p_{mm} \end{pmatrix} \]

\[ \ln[1 - \Pr(y_i = 1, x_j = 1)] \]

\[ \text{Column constraints } \Rightarrow \forall j, i: p_{ij} \geq t_j \]

\[ \text{Row constraints } \Rightarrow \forall i: \begin{cases} \ln(1 - z_i - \epsilon) \leq \sum_{j=1}^{m} p_{ij} \\ \ln(1 - z_i + \epsilon) \geq \sum_{j=1}^{m} p_{ij} \end{cases} \]

- **Constraint (11):**
  - \( p_{ij} = 0 \) if \( \Pr(y_i = 1, x_j = 1) = 0 \).

- **Constraint (10) peptide probability:**

\[ z_i \pm \epsilon = 1 - \prod_{j=1}^{m} e^{p_{ij}} \quad (12) \]

\[ \Rightarrow \ln(1 - z_i \pm \epsilon) = \sum_{j=1}^{m} p_{ij} \quad (13) \]
LP formulation solved with GLPK

A standard LP:

Maximize: \[ c^T x + c_0, \] (14)
Subject to: \[ Ax = b, \] (15)
\[ Ax \leq b, \] (16)
\[ LB \leq x \leq UB. \] (17)

\[ x = (p_{11} \cdots p_{1m} p_{21} \cdots p_{2m} \cdots p_{nm} t_1 \cdots t_m)^T \]

\[ c^T = (0 | 1_{1,m}), \quad c_0 = 0 \]

\[ A = \begin{pmatrix} 1_{1,m} & 0 & \cdots & 0 \\ 0 & 1_{1,m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1_{1,m} \end{pmatrix} \]
Using results of $P$ from Glpk, joint probability matrix $1 - e^P$ is computed:

<table>
<thead>
<tr>
<th></th>
<th>Protein1</th>
<th>Protein2</th>
<th>Protein3</th>
<th>...</th>
<th>Protein$_m$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peptide 1 (0.9)</td>
<td>0.9</td>
<td>0</td>
<td>0</td>
<td>...</td>
<td>0</td>
</tr>
<tr>
<td>Peptide 2 (0.85)</td>
<td>0.7</td>
<td>0.5</td>
<td>0</td>
<td>...</td>
<td>0</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Peptide n (0.9)</td>
<td>0</td>
<td>0.5</td>
<td>0</td>
<td>...</td>
<td>0.8</td>
</tr>
</tbody>
</table>

Protein Probabilities: $1 - (1 - 0.9)(1 - 0.7)$
Peptide degeneracy

- **ProteinLP**: joint probability $Pr(x_j = 1, y_i = 1)$. e.g. if peptide $i$ present in more than one protein: $m$, $n$, $r$:

  $$Pr(x_m = 1, y_i = 1) \cdot Pr(x_n = 1, y_i = 1) \cdot Pr(x_r = 1, y_i = 1) > 0$$  (18)

- **ProteinProphet**: taking a weight $w_i^n$ into account, if peptide $i$ corresponds to $N$ different proteins.

  $$p_n = 1 - \prod_{i}^{n}(1 - w_i^j Pr(+/S_i)) \quad j = 1 \cdots N.$$  (19)

Combining with Number of Sibling Peptides (NSP): $NSP_i = \sum_{\{m|m \neq i\}} p(+/D_m)$.

$$Pr(+/S, NSP) = \frac{Pr(S+)/Pr(NSP+)/Pr(S+/NSP) + Pr(S-/NSP)}{Pr(S+)/Pr(NSP+)/Pr(S-/NSP)}.$$  (20)
Part III

Results
Datasets

- Ground-truth data: 18 mixtures (Klimek et al., 2008), Sigma49 and yeast (Ramakrishnan et al., 2009a)

- Data without reference sets: DME (Brunner et al., 2007), HumanMD (Ramakrishnan et al., 2009b) and HumanEKC (Ramakrishnan et al., 2009a).
Data obtained from http://www.marcottelab.org/MSdata/.

Peptide identification: X!Tandem (v2010.10.01.1) (David and Cottrell, 2004).

GLPK (LPWrapper in OpenMS)

Proteinlists:

```xml
<PeptideIdentification score_type="XTandem" higher_score_better="true" significance_threshold="0" MZ="667.96337890625" RT="901.678">
  <PeptideHit score="20.8" sequence="DQQKDAEGELSAATLLPK" charge="3" aa_before="K" aa_after="L" protein_refs="PH_4025">
    <UserParam type="float" name="E-Value" value="1.1"/>
  </PeptideHit>
</PeptideIdentification>

<PeptideIdentification score_type="XTandem" higher_score_better="true" significance_threshold="0" MZ="408.515991210938" RT="902.077">
  <PeptideHit score="23.2" sequence="SPPPSPTTQRR" charge="3" aa_before="R" aa_after="L" protein_refs="PH_432 PH_429 PH_428 PH_4628 PH_5036">
    <UserParam type="float" name="E-Value" value="0.65"/>
  </PeptideHit>
</PeptideIdentification>
```
Validation

With setting a threshold $t$ on the protein probabilities, only positive proteins remain.

False positives can be determined:
- Ground truth datasets.
- Datasets without references - using target-Decoy Analysis.
  - Protein database contaminated with a set of shuffled unreal sequences (decoy database).
  - Protein from decoy database is false one.
Validation

Given a certain probability threshold $t$, $F_t$ is the number of false positives,

- False Discovery Rate (FDR):
  \[
  FDR_t = \frac{F_t}{F_t + T_t}.
  \]

- q-values:
  \[
  q_t = \min_{t' \leq t} FDR_{t'}.
  \]

- Posterior error probability (PEP):
  \[
  PEP = Pr(+|p).
  \]
Comparison of q-values

MSB is MSBayespro\textsuperscript{5}.

Comparison of q-values
Comparison: the number of degenerate peptides

<table>
<thead>
<tr>
<th></th>
<th>PP</th>
<th>ProteinLP</th>
<th>Fido</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TP</td>
<td>FP</td>
<td>TP</td>
</tr>
<tr>
<td>Simple proteins</td>
<td>17</td>
<td>8</td>
<td>17</td>
</tr>
<tr>
<td>Degenerate Proteins</td>
<td>1</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>18 mixtures Sigma49</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simple proteins</td>
<td>27</td>
<td>1</td>
<td>30</td>
</tr>
<tr>
<td>Degenerate Proteins</td>
<td>5</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>18 mixtures HumanMD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simple proteins</td>
<td>70</td>
<td>0</td>
<td>64</td>
</tr>
<tr>
<td>Degenerate Proteins</td>
<td>54</td>
<td>0</td>
<td>60</td>
</tr>
</tbody>
</table>

*Table:* Accuracy on proteins containing shared peptides with q-value threshold 0.3 for Sigma49 and 0.01 for HumanMD.
Part IV

Conclusions
Conclusions

- Joint probabilities provide the degeneracy information.
- Joint probabilities simplify the optimization problem.
- To do:
  - Integrate supplementary information, e.g. protein-protein interaction, by adding linear constraints.
  - Considering the parameter $\epsilon$ for different peptide probabilities and protein information.
Thanks for listening.

Questions?