REPRISAL: Protein Inference in Python

Joseph Isaac Bloom[1], Giuseppe Infusini [1,2,3], Andrew Webb[1,2,3]

[1] Mass Dynamics, Melbourne, Australia [2] The Walter and Eliza Hall Institute of Medical Research, Melbourne, Victoria, Australia [3] Department of Medical Biology, University of Melbourne, Melbourne, Victoria, Australia

INTRODUCTION

Problem Statement: The Protein Inference (PI) Problem is the task of inferring which proteins are present in a given sample based on peptide identifications where proteins may share peptides[1].

Background: During LC-MS/MS bottom-up proteomics workflows, we digest proteins into peptides and then need to "put humpty together again" after observing peptides. This affects quantification and can be difficult to interpret.

Our Approach: Our approach is has two layers:

1. A novel PI algorithm, REPRISAL (REcursive PRotein Inference and Scoring ALgorithm) algorithm.
2. An open-source python package, "protein-inference", to run REPRISAL (and other PI algorithms).

METHODS

The PI Python Package (Figure 1):

- Provides a generic workflow for Protein Inference using Target Decoy Scoring.
- Provides utilities for annotation and visualization.

Package: *https://github.com/MassDynamics/protein- inference*

REPRISAL Algorithm:

REPRISAL operates on each problem network*, using the following algorithm:

1. Unscored proteins are assigned scores equal to the sum of associated peptide scores.
2. The protein with the highest provisional score is assigned the peptides currently associated with it. (These peptides are excluded from further consideration.)
3. If any protein now lacks any possible peptides that are unassigned, it is assigned a "subset" protein to the protein which was last assigned peptides (and scored).
4. Repeat steps 1-3 until all peptides are assigned and all proteins scored.

Benchmarking (Figure 2):

We have compared the performance of REPRISAL, FIDO and Percolator "picked protein" on iPRG2016[2] and Schriek et al 2021[3].

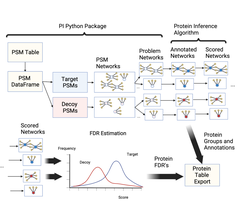


Figure 1: Conceptual diagram of protein inference workflow.

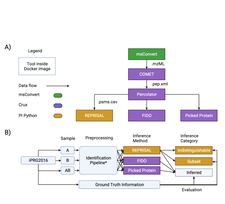


Figure 2: Benchmarking Workflow A) Computation[4,5,6,7]. B) Datasets

RESULTS

REPRISAL Network Annotation and Protein Grouping

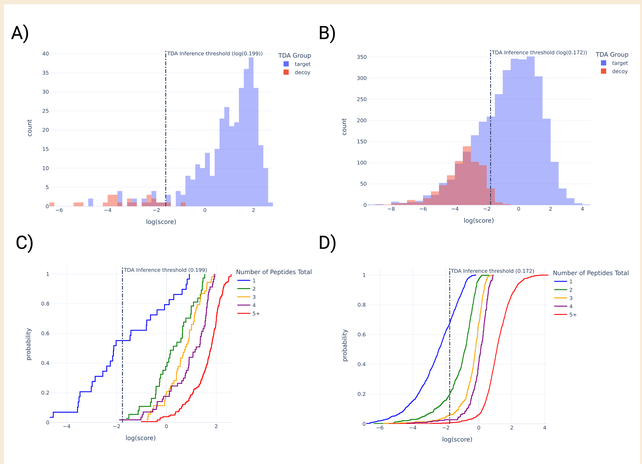


Figure 3: Target Decoy Scores. A) iPRG2016 Sample AB, B) Schriek et al 2021
C) Targets from A, by peptides in network. D) Targets from B, by peptides in Network

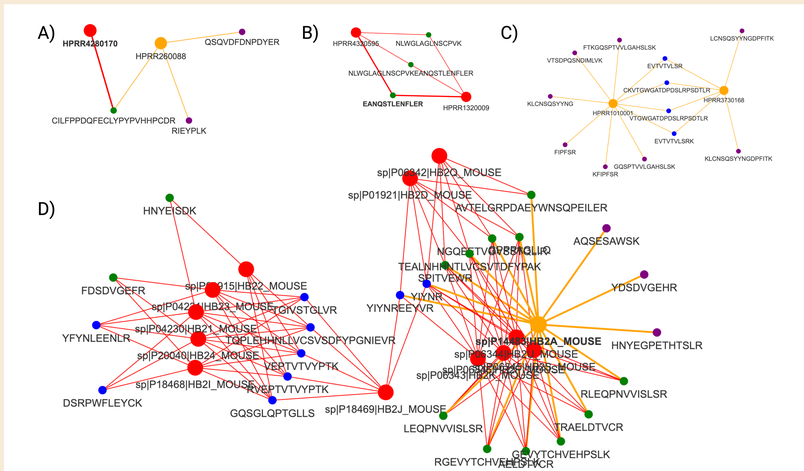


Figure 4: Example *problem networks with coloring by annotation A, B, C) iPRG2016. D) Schriek et al 2021 Study. Key: Yellow - Proteins with proteotypic peptides (purple). Red - Proteins without proteotypic peptides. Green - razor peptides. Blue - non-unique peptides.

iPRG2016 Benchmark

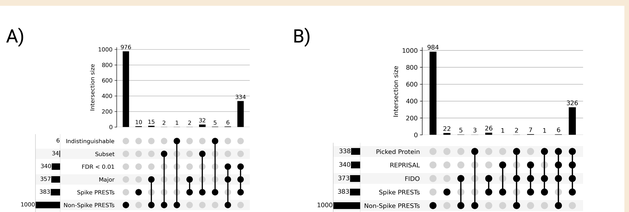


Figure 5: Upset Plots show benchmarking results for iPRG2016 Sample AB
A) Ground Truth and Inference Categories B) Ground Truth and other PI algorithms

Schriek et al 2021 - Inference and Quantitation

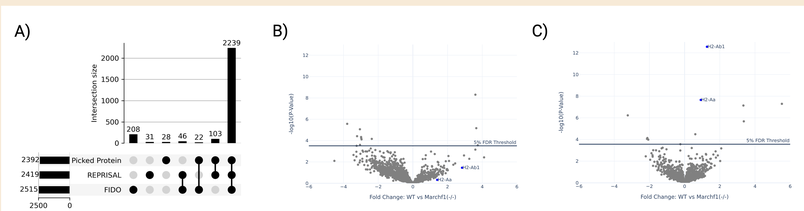


Figure 6: Upset Plots for Schriek et al study (processed on Mass Dynamics with/without REPRISAL and Razor based Quantification)
A) Identitifcations by Method , Volcano plot for Schriek et al 2021 B) with percolator picked protein, D) with REPRISAL

DISCUSSION

REPRISAL

- Good Target Decoy separation (Figure 3, Figure 6).
- Inference categories provide detail (Figure 6, Figure 7b).
- REPRISAL returns lower false positives on iPRG2016. (Table 1).

Note: iPRG2016 has very simple problem networks compared to Schriek et al 2021. FIDO was run with default parameters.

Protein Inference in Python

- Visualizations provide interpretability (Figure 4, 5)
- Streamlit app enables exploration of problem networks (Figure 8)

Protein Inference and Quantitation

- Proteotypic/Razor peptides are essential.
- Razor peptides can also used for Quantification (Figure 7)

CONCLUSION

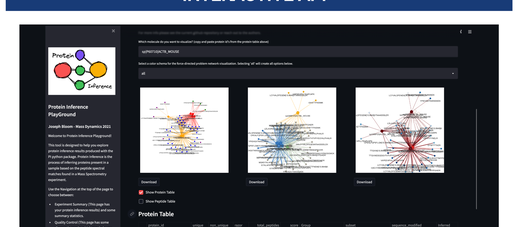
REPRISAL is a PI algorithm, which scores, groups and annotates proteins.

The PI Python package provides a framework for performing PI and analysing identification results in python.

Future work may include:

- Benchmarking on more complex datasets and in highly homologous networks with ground truth quantitative data.
- Further algorithmic development of REPRISAL
- Further development of the PI playground app for results investigation.

INTERACTIVE APP



CITATIONS

1. Li YF, Radivojac P. Computational approaches to protein inference in shotgun proteomics. BMC Bioinformatics. 2012;13 Suppl 16: S4.
2. Lee J-Y, Choi H, Colangelo CM, Davis D, Hoopmann MR, Käll L, et al. ABRF Proteome Informatics Research Group (iPRG) 2016 Study: Inferring Proteoforms from Bottom-up Proteomics Data. J Biomol Tech. 2018;29: 39–45.
3. Schriek P, Liu H, Ching AC, Hoang P, Gupta N, Wilson KR, et al. Physiological substrates and ontogeny-specific expression of the ubiquitin ligases MARCH1 and MARCH8. Curr Res Immunol. 2021;2: 218–229.
4. Kong AT, Leprevost FV, Avtonomov DM, Mellacheruvu D, Nesvizhskii AI. MSFragger: ultrafast and comprehensive peptide identification in mass spectrometry–based proteomics. Nature Methods. 2017;14: 513–520.
5. Eng JK, Jahan TA, Hoopmann MR. Comet: an open-source MS/MS sequence database search tool. Proteomics. 2013;13: 22–24.
6. Käll L, Canterbury JD, Weston J, Noble WS, MacCoss MJ. Semi-supervised learning for peptide identification from shotgun proteomics datasets. Nat Methods. 2007;4: 923–925.
7. Serang O, MacCoss MJ, Noble WS. Efficient marginalization to compute protein posterior probabilities from shotgun mass spectrometry data. J Proteome Res. 2010;9: 5346–5357.

Thanks to Mark Condina for his assistance and feedback when preparing this poster.