Published in final edited form as:

J Proteome Res. 2018 March 02; 17(3): 1314–1320. doi:10.1021/acs.jproteome.7b00786.

StPeter: Seamless Label-Free Quantification with the Trans-Proteomic Pipeline

Michael R. Hoopmann^{†,iD}, Jason M. Winget^{†,‡}, Luis Mendoza, and Robert L. Moritz^{*,iD}
Institute for Systems Biology, Seattle, Washington 98109, United States

Abstract

Label-free quantification has grown in popularity as a means of obtaining relative abundance measures for proteomics experiments. However, easily accessible and integrated tools to perform label-free quantification have been lacking. We describe StPeter, an implementation of Normalized Spectral Index quantification for wide availability through integration into the widely used Trans-Proteomic Pipeline. This implementation has been specifically designed for reproducibility and ease of use. We demonstrate that StPeter outperforms other state-of-the-art packages using a recently reported benchmark data set over the range of false discovery rates relevant to shotgun proteomics results. We also demonstrate that the software is computationally efficient and supports data from a variety of instrument platforms and experimental designs. Results can be viewed within the Trans-Proteomic Pipeline graphical user interfaces and exported in standard formats for downstream statistical analysis. By integrating StPeter into the freely available Trans-Proteomic Pipeline, users can now obtain high-quality label-free quantification of any data set in seconds by adding a single command to the workflow.

Graphical abstract

*Corresponding Author. rmoritz@systemsbiology.org.

ORCID

Michael R. Hoopmann: 0000-0001-7029-7792

Robert L. Moritz: 0000-0002-3216-9447

‡Present Address

J.M.W.: Procter & Gamble, 8700 Mason Montgomery Road, Mason, OH, 45040.

†M.R.H. and J.M.W. contributed equally to this work.

Author Contributions

M.R.H. wrote the final program implementation and devised the distributed algorithm. J.M.W. prototyped the software program. L.M. developed the user interface integrated with the TPP. J.M.W. and M.R.H. performed the data analysis and benchmarking. J.M.W., M.R.H., and R.L.M. drafted and edited the manuscript.

The authors declare no competing financial interest.

Keywords

label-free quantification; open-source software; quantitative proteomics; data analysis pipeline; automation; trans-proteomic pipeline

INTRODUCTION

Label-free quantification (LFQ) of shotgun proteomics data is attractive because it requires no additional sample modification, is inexpensive to implement, and does not require a list of targets a priori. There are two broad categories of LFQ approaches based on whether precursor ion (MS^1) or fragment ion (MS^2) data are used as the basis of data required by a processing algorithm. Many reviews have compared the merits of each approach (for example^{1,2}), with the general finding that several methods perform well when applied to either approach for protein quantification.

The Trans-Proteomic Pipeline (TPP) is a popular and widely used data-processing pipeline for shotgun proteomics data.³ A signature feature of the TPP is its usage of open file formats to interchange data between processing modules⁴ and support for all major instrument platforms via mass spectrometer (MS) raw data file conversions using the ProteoWizard package.⁵ While the TPP has supported label-based quantification since its inception,⁶ it has until now not included a label-free algorithm. Although users could manually integrate third party LFQ algorithms in the TPP, novice users may struggle with such advanced tasks, thus creating a need to provide an LFQ algorithm and workflow as a standard feature in the widely used TPP.

Of the many published LFQ approaches¹ to implement in the TPP, we compared several techniques and chose to use the MS^2 -based Normalized Spectral Index (SI_N) described by Griffin et al.⁷ This method strikes a balance of accuracy and sensitivity while remaining computationally efficient compared to other label-free techniques. The Normalized Spectral Index approach has been previously implemented in other community tools, such as SINQ and the Crux toolkit;^{8,9} however, these implementations are not compatible with the analytical methods, statistical validation approaches, and additional open formats used in the TPP. An algorithm, named StPeter, was written to implement SI_N analysis within the TPP and provide the community with a well-incorporated and easy to use LFQ method in this pipeline.

With any shotgun-based proteomic quantification technique, accuracy is impeded by many factors, including peptides assigned to multiple proteins. Alternative shotgun LFQ techniques have proposed methods, such as dividing degenerate peptide signals among inferred proteins,¹⁰ to avoid quantification errors. Griffin et al. made no consideration for degenerate peptides in their published method. Therefore, we added a novel modification to the SI_N approach that distributes the portion of the SI_N from shared peptides among their constituent proteins, similar to strategies developed for spectral counting methods.¹⁰ To the best of our knowledge, this distributed method (d SI_N) has never been previously applied to normalized spectral index-based analyses. We evaluate our strategy and algorithm and

demonstrate its use within the TPP for rapid and user-friendly quantification of shotgun data sets.

EXPERIMENTAL PROCEDURES

Algorithm

The spectral index (SI) for a protein is the cumulative fragment ion intensity for each spectrum of each peptide giving rise to a protein, which can be represented using the equation from Griffin et al.

$$SI = \sum_{k=1}^{pn} \left(\sum_{j=1}^{sc} i_j \right)_k \quad (1)$$

where i is the fragment ion intensity of peptide k , j is the j th spectral count of sc total spectral counts for peptide k and pn is the number of peptides identified for that protein.

Further normalization is required to improve the accuracy of spectral index, using the following Normalized Spectral Index (SI_N) equation from Griffin et al.

$$SI_N = \left[\sum_{k=1}^{pn} \left(\sum_{j=1}^{sc} i_j \right)_k \right] / \left[\sum_{x=1}^n SI_x \right] / L \quad (2)$$

where SI is the total spectral index for all identified proteins, n , and L is the protein length (number of amino acids) for the protein.

The observed spectral index of a peptide shared among multiple proteins represents the combined signal contribution from each of those proteins. To approximate the contribution of shared peptides to each of their respective proteins, the signal is divided among the fractional contribution of unique peptides for each protein. Thus the protein spectral index from peptides unique to that protein can be defined as

$$SI_u = \sum_{k=1}^{pu} \left(\sum_{j=1}^{sc} i_j \right)_k \quad (3)$$

where i is the fragment ion intensity of peptide k , j is the j th spectral count of sc total spectral counts for peptide k , and pu is the number of peptides identified that are unique for that protein. Similarly, the summed fragment ion intensity for a shared peptide sequence (FI_s) can be defined as

$$FI_s = \sum_{j=1}^{sc} i_j \quad (4)$$

where i is the fragment ion intensity and j is the j th spectral count of sc total spectral counts for the peptide sequence that is shared among multiple proteins. Therefore, the distributed spectral index (dSI) can be defined as the spectral index of unique peptides plus the fractional spectral index of shared peptides for a protein

$$dSI = SI_u + \sum_{k=1}^{ps} \left(\frac{SI_u}{\sum_{m=1}^M (SI_u)_m} \times FI_s \right)_k \quad (5)$$

where for each shared peptide, k , among the set of shared peptides, ps , the fractional spectral index is the fragment ion intensity for the peptide multiplied by the ratio of the spectral index of unique peptides for a protein divided by the sum of spectral indexes for all proteins, M , that share the same peptide. Finally, the distributed spectral index can be normalized using the following equation

$$dSI_N = dSI / \left[\sum_{j=1}^n dSI_j \right] / L \quad (6)$$

where the distributed spectral index for a protein is divided by the sum of dSI for all identified proteins, n , and L is the protein length (number of amino acids) for the protein.

Implementation

StPeter is written in the C++ programming language and is compiled and installed along with the TPP. When run, the program effectively traverses proteomic search results in reverse (Figure 1). First, the protein results (protXML) are parsed to yield inferred protein identifications, probabilities of correctness, evidential peptide sequences, and protein lengths as well as the source peptide results file or files (pepXML). This pepXML is then parsed to yield peptide-spectrum matches (PSMs), post-translational modifications, and the raw data files in mzML format. Finally, the raw mzML files are processed to retrieve the spectral information with peak mass/charge ratios and intensities. Once these data are compiled, the program generates theoretical 1+ and 2+ charged (for 3+ and greater precursor charges) fragment ion series (e.g., b and y ions for collision induced dissociation) for each peptide of interest and matches these to the experimental spectrum according to the defined mass tolerance. The associated peak intensities are used to compute the spectral indexes, which are then normalized across all observed protein spectral indexes, as described above. The spectral index values are then exported to the original protXML file in a nested quantification element for each protein. Additionally, the user may request protein quantities exported to a summary comma-separated file. Because of shared approaches with other quantification strategies, the software also reports alternative quantification metrics such as raw spectral counting and Normalized Spectral Abundance Factor,¹¹ and can be quickly modified to support alternate quantification or analysis pipelines, such as MSstats.¹²

Several technical considerations were made in designing StPeter to maximize its versatility. Efficient file parsing was used to manage analysis of large data sets: The individual pepXML and mzML files in a typical shotgun analysis can easily exceed 1 GB each. After extracting protein information from the protXML file, peptide and PSM targets are cataloged, sorted, and iteratively extracted from their respective data files. This approach ensures a low memory footprint despite the exceptional size of many proteomics data sets. Quantification occurs for all detected peptides and their inferred proteins at or below the user-defined false discovery rate (FDR) threshold. For each of these proteins, quantification is determined only from PSMs with the same FDR threshold. This filtering step is necessary to avoid quantifying proteins using PSMs that have very poor probabilities but were used as corroborating evidence of protein identification with other high-scoring PSMs. Protein quantities are written as elements to the input protXML file. If an analysis is repeated with different parameters, then the new results replace the original analysis. Because the protXML file after StPeter analysis is identical, save for the supplemented protein quantities, it remains compatible with all other tools that use the format, including the TPP. Therefore, additional downstream analysis and visualization of the results are seamless.

Benchmarking

Data from Ramus et al. were downloaded from the PRIDE repository using the data set identifier PXD001819.¹³ The data contain the 48 UPS1 human proteins set at various concentrations in a constant yeast lysate background. Raw MS data were converted to mzML format using msconvert,¹⁴ then searched using Comet 2016.01 rev. 2¹⁵ and MS-GF+ v9949¹⁶ search engines against a database containing all yeast proteins, UPS1 spike-in proteins, and trypsin plus randomized decoy sequences. Search engine parameters were tuned for similarity, with 25 ppm mass tolerance, methionine oxidation set as a variable modification, cysteine carbamidomethylation as a fixed modification, up to two missed cleavages, and two peptide tryptic termini required. MS-GF+ output was converted to pepXML using the idconvert function in ProteoWizard.⁵ Resulting pepXML files were processed with PeptideProphet,¹⁷ iProphet,¹⁸ and ProteinProphet¹⁹ in TPP 5.0 rc9³ to produce protXML result files of inferred proteins. These protXML files were used as input to StPeter to yield quantitative information.

For statistical analysis, we converted SI_N and dSI_N values for each protein to nanogram estimations using the RPQ method.⁷ In brief, each protein SI_N is divided by the sum of all proteins' SI_N and multiplied by the protein load in nanograms. Protein quantitation comparisons used MSstats v.3.5.3¹² with Tukey's median polish, after extracting StPeter values from the protXML files and formatting them for MSstats using an in-house script. MSstats normalization was disabled and missing values were replaced with the minimum estimated protein quantity from all runs. Comparison of true-positive rate (TPR) and false-positive rate (FPR) were made as described in Ramus¹³ and Veit²⁰ for consistency. In brief, the combined results of three UPS1 comparisons (50 vs 0.5 fmol/ μ g, 50 vs 5 fmol/ μ g, and 25 vs 12.5 fmol/ μ g) were merged into a single list and filtered for a $\log_2$ -fold change of 1 or -1. The list was sorted by p value, and TPR was calculated as the fraction of UPS1 proteins observed from the expected total ($3 \times 48 = 144$) and FPR was calculated as the fraction of yeast proteins observed among the total proteins at a given p value.

Data from Zhang et al.^{10,21} containing known quantities of homologous proteins were analyzed to assess the dSI_N method. In brief, the data are 12 replicate MudPIT²² analyses on a linear ion trap mass spectrometer, consisting of tryptic digests of six purified serum albumin proteins (mouse, rat, rabbit, pig, bovine, and human) in a yeast lysate background, as previously described.¹⁰ To facilitate analysis using StPeter inside the TPP, the raw data were converted to mzML and searched using Comet 2016.01 rev. 2.¹⁵ Comet parameters replicated those used in the original study, namely, a precursor mass tolerance of 3.0 Da, fixed cysteine carbamidomethylation modification, differential methionine oxidation modification, and a fragment bin tolerance of 1.0005 Da with an offset of 0.4 Da. Following the database search, peptide-spectrum matches (PSMs) were validated with PeptideProphet, and proteins were inferred with ProteinProphet. The resulting protXML files were analyzed twice with StPeter, once computing SI_N on proteotypic peptides only and again computing dSI_N from all identified peptides.

Data from Cox et al.²³ containing thousands of protein differences were used to assess the dSI_N method on large-scale whole proteome analyses. These data consisted of a three-fold difference in *Escherichia coli* protein (K12 lysate) mixed with a constant background of human protein (HeLa S3 lysate), fractionated, and analyzed in triplicate by shotgun mass spectrometry. The same workflow was used as the Zhang et al. analysis, with the exception of a 25 ppm precursor mass tolerance when searching with Comet, and only dSI_N was computed. Additionally, dSI_N values were only used for proteins observed in at least two of the three replicates for each condition.

Additional benchmarking of the StPeter algorithm performance was conducted using publicly available data sets from the PeptideAtlas repository. These data sets differed in size, instrumentation, and complexity. Species names, instrument platforms, and data set size are described in Table 1. Efficiency of StPeter computation was assessed.

RESULTS

StPeter was benchmarked against a recently published data set composed of a yeast lysate with varied spike-in levels of an equimolar mixture of 48 human proteins.¹³ We then performed a statistical comparison as described in the original publication to permit direct comparison of the results.

Comparisons of human protein quantities were performed for three different ratios of UPS1 concentrations (Figure 2A). StPeter dSI_N protein quantities were analyzed with MSstats, as described in the methods, and visualized by plotting the computed log₁₀(*p*-value) against the protein log₂(fold-change). As expected, most human proteins clustered apart from the background yeast proteins, particularly for the sets with the largest UPS1 ratios. Although low fold-changes (red) were more difficult to discriminate than large fold-changes (blue and green), these results are similar and consistent with previously published observations on the same data set.¹³ In the typical FPR range applied to shotgun proteomics data (1–5%), StPeter outperforms other benchmarked algorithms, with sensitivity from 72 to 83% over this range (Figure 2B). StPeter performed well with and without use of the distributed

approach, although for this particular data set, the use of only proteotypic peptides in computing SI_N performed the best.

To assess the dSI_N approach, a data set of six similar albumin proteins was analyzed.^{10,21} Twelve replicates of a shotgun mixture of homologous albumins from six different species at 2.5-fold concentration differences were analyzed using the TPP with StPeter, as described in the methods. For each replicate, linear correlation of computed quantities with known spiked-in concentrations were computed (RSQ in Excel), as previously described.¹⁰ Only proteotypic peptides were used when computing SI_N , producing an average correlation of 0.904 ± 0.065 across the 12 replicates. Including degenerate peptides to compute dSI_N improved the average correlation to 0.920 ± 0.055 . This improvement resembles the same improvement observed when computing dNSAF instead of NSAF, as reported by the authors of the data set.¹⁰ When using average protein values across the 12 replicates, correlation using dSI_N was 0.987 versus 0.963 for SI_N (Figure 3), showing strong linearity in performance with either method over three orders of magnitude.

The dSI_N approach was then used to analyze a large-scale whole proteome analysis in which *E. coli* lysate was mixed with HeLa lysate, separated into 24 fractions, and analyzed in triplicate by shotgun mass spectrometry.²³ Two conditions were observed, either 10 or 30 μ g of *E. coli* lysate mixed with 60 μ g of human lysate. Ratios of dSI_N were computed for ~4500 proteins observed under both conditions (Figure 4). Comparisons were made to methods using SI_N , NSAF,²⁴ and dNSAF.¹⁰ Both the *E. coli* and human proteins formed distinct clouds, with *E. coli* proteins showing an increase in fold-change and human proteins showing a decrease in fold-change, as expected (Figure 4A). The mean of the fold-change distribution was used to compute a $\log_2$ ratio of 1.73 between the two distributions when using dSI_N or approximately the 3-fold expected increase in *E. coli* protein concentration (Figure 4B). Comparisons between all quantification methods showed only slight differences in mean fold-change and standard deviations among the protein populations. All methods produced a distinct cone-shaped distribution among protein quantities where the ratios of the lowest abundant proteins are most influenced by slight differences in spectral observations. Although dSI_N and SI_N produced the same fold-change ratio, the standard deviations were slightly better for dSI_N . A similar trend was observed when comparing standard deviations of dNSAF to NSAF. All spectral counting methods presented here outperformed the previously reported spectral counting method, which had produced a mean fold-change of 1.95 (approximately four-fold) with human and *E. coli* protein standard deviations of 0.69 and 0.86, respectively.²³ These results more closely resemble those obtained using precursor signals and MaxLFQ (mean fold-change of 1.74, 0.46 human and 0.51 *E. coli* protein standard deviations), on the same data set,²³ and illustrate the improvement over simple spectral counting when using spectral indexes, as previously reported.⁷ These results also agree with more extensive comparisons between MS^1 -based and MS^2 -based quantification methods.²⁵

To profile hardware performance as well as to demonstrate its capability to process data from a variety of sources, we ran StPeter on several publicly available data sets from the PeptideAtlas repository, summarized in Table 1. These data were from a variety of species, instrument platforms, and experimental scales and included both low- and high-resolution

MS² experiments. As shown in the Table, StPeter is extremely fast and efficient, with all analyses finishing in seconds with a low memory footprint.

Interface and Further Analysis of LFQ Results

StPeter has both command-line and graphical interfaces. The graphical user interface is integrated with the Petunia interface used for all TPP components (Figure 5A). The user selects the protXML for quantification and can set FDR and MS² match tolerances. Once quantification is complete the user is provided a link to the protXML viewer, which displays the results in graphical or text format (Figure 5B). The values are rewritten to the protXML so that the results can be re-extracted from the protXML in the future without rerunning the quantification. StPeter results can also be exported to simplified tab-separated text files using the export function of the protXML viewer invoked within the Petunia interface.

Statistical analysis of proteomics data is an active area of research, and approaches vary on experimental design and desired comparisons to be made; therefore, this portion of the interpretation of StPeter results is left to the user. We currently recommend using MSstats¹² on protein-level normalized spectral indices with internal MSstats normalization disabled. Additional information, instructions, and tutorials on how to use StPeter to replicate the presented analyses can be found at <http://tools.proteomecenter.org/wiki/index.php?title=Software:StPeter>

CONCLUSIONS

Here we have described StPeter, an efficient implementation of the Normalized Spectral Index algorithm for label-free quantification of LC–MS/MS data. Additionally, we have described modifications to the algorithm, described as the distributed normalized spectral index, for the label-free quantification of proteins that includes peptides shared among multiple proteins. Because of its integration with open data formats, StPeter can be applied to data of any experimental design collected on all MS instrument platforms that have been processed and searched via any MS interpretation method, which ultimately yields a protXML output. StPeter is itself free and open source, released within the TPP. It can be downloaded with the current release of the TPP at <http://sourceforge.net/p/sashimi>

Acknowledgments

We thank Dr. Eric W. Deutsch for constructive review of this manuscript. This work was funded in part by National Institutes of Health from the National Institute of General Medical Sciences under Grant Nos. 2P50 GM076547/Center for Systems Biology and R01 GM087221; the National Heart, Lung and Blood Institute Grant No. R01 HL133135; and Procter & Gamble, Inc.

ABBREVIATIONS

FDR false discovery rate

TPR true-positive rate

FPR false-positive rate

LC–MS/MS liquid chromatography tandem mass spectrometry

LFQ	label-free quantification
NSAF	normalized spectral abundance factor
PSM	peptide-spectrum match
SI_N	normalized spectral index
dSI_N	distributed normalized spectral index
TPP	Trans-Proteomic Pipeline

References

1. Blein-Nicolas M, Zivy M. Thousand and one ways to quantify and compare protein abundances in label-free bottom-up proteomics. *Biochim. Biophys. Acta, Proteins Proteomics*. 2016; 1864(8):883–95.
2. Dowle AA, Wilson J, Thomas JR. Comparing the Diagnostic Classification Accuracy of iTRAQ, Peak-Area, Spectral-Counting, and emPAI Methods for Relative Quantification in Expression Proteomics. *J. Proteome Res*. 2016; 15(10):3550–3562. [PubMed: 27546623]
3. Deutsch EW, Mendoza L, Shteynberg D, Slagel J, Sun Z, Moritz RL. Trans-Proteomic Pipeline, a standardized data processing pipeline for large-scale reproducible proteomics informatics. *Proteomics: Clin. Appl*. 2015; 9(7–8):745–54. [PubMed: 25631240]
4. Keller A, Eng J, Zhang N, Li XJ, Aebersold R. A uniform proteomics MS/MS analysis platform utilizing open XML file formats. *Mol. Syst. Biol*. 2005; 1:E1.
5. Kessner D, Chambers M, Burke R, Agus D, Mallick P. ProteoWizard: open source software for rapid proteomics tools development. *Bioinformatics*. 2008; 24(21):2534–6. [PubMed: 18606607]
6. Li XJ, Zhang H, Ranish JA, Aebersold R. Automated statistical analysis of protein abundance ratios from data generated by stable-isotope dilution and tandem mass spectrometry. *Anal. Chem*. 2003; 75(23):6648–57. [PubMed: 14640741]
7. Griffin NM, Yu J, Long F, Oh P, Shore S, Li Y, Koziol JA, Schnitzer JE. Label-free, normalized quantification of complex mass spectrometry data for proteomic analysis. *Nat. Biotechnol*. 2010; 28(1):83–9. [PubMed: 20010810]
8. McIlwain S, Tamura K, Kertesz-Farkas A, Grant CE, Diamant B, Frewen B, Howbert JJ, Hoopmann MR, Kall L, Eng JK, MacCoss MJ, Noble WS. Crux: rapid open source protein tandem mass spectrometry analysis. *J. Proteome Res*. 2014; 13(10):4488–91. [PubMed: 25182276]
9. Trudgian DC, Ridlova G, Fischer R, Mackeen MM, Ternette N, Acuto O, Kessler BM, Thomas B. Comparative evaluation of label-free SINQ normalized spectral index quantitation in the central proteomics facilities pipeline. *Proteomics*. 2011; 11(14):2790–7. [PubMed: 21656681]
10. Zhang Y, Wen Z, Washburn MP, Florens L. Refinements to label free proteome quantitation: how to deal with peptides shared by multiple proteins. *Anal. Chem*. 2010; 82(6):2272–81. [PubMed: 20166708]
11. Pavelka N, Fournier ML, Swanson SK, Pelizzola M, Ricciardi-Castagnoli P, Florens L, Washburn MP. Statistical similarities between transcriptomics and quantitative shotgun proteomics data. *Mol. Cell. Proteomics*. 2008; 7(4):631–44. [PubMed: 18029349]
12. Choi M, Chang CY, Clough T, Broudy D, Killeen T, MacLean B, Vitek O. MSstats: an R package for statistical analysis of quantitative mass spectrometry-based proteomic experiments. *Bioinformatics*. 2014; 30(17):2524–6. [PubMed: 24794931]
13. Ramus C, Hovasse A, Marcellin M, Hesse AM, Mouton-Barbosa E, Bouyssie D, Vaca S, Carapito C, Chaoui K, Bruley C, Garin J, Cianferani S, Ferro M, Van Dorssaeler A, Burlet-Schiltz O, Schaeffer C, Coute Y, Gonzalez de Peredo A. Benchmarking quantitative label-free LC-MS data processing workflows using a complex spiked proteomic standard dataset. *J. Proteomics*. 2016; 132:51–62. [PubMed: 26585461]

14. Holman JD, Tabb DL, Mallick P. Employing ProteoWizard to Convert Raw Mass Spectrometry Data. *Curr. Protoc Bioinformatics*. 2014; 46:13.24.1–9.
15. Eng JK, Jahan TA, Hoopmann MR. Comet: an open-source MS/MS sequence database search tool. *Proteomics*. 2013; 13(1):22–4. [PubMed: 23148064]
16. Kim S, Pevzner PA. MS-GF+ makes progress towards a universal database search tool for proteomics. *Nat. Commun*. 2014; 5(5277)
17. 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):5383–92. [PubMed: 12403597]
18. 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.
19. Nesvizhskii AI, Keller A, Kolker E, Aebersold R. A statistical model for identifying proteins by tandem mass spectrometry. *Anal. Chem*. 2003; 75(17):4646–58. [PubMed: 14632076]
20. Veit J, Sachsenberg T, Chervov A, Aicheler F, Urlaub H, Kohlbacher O. LFQProfiler and RNP(xl): Open-Source Tools for Label-Free Quantification and Protein-RNA Cross-Linking Integrated into Proteome Discoverer. *J. Proteome Res*. 2016; 15(9):3441–8. [PubMed: 27476824]
21. Zhang Y, Wen Z, Washburn MP, Florens L. Improving label-free quantitative proteomics strategies by distributing shared peptides and stabilizing variance. *Anal. Chem*. 2015; 87(9):4749–56. [PubMed: 25839423]
22. Florens L, Washburn MP. Proteomic analysis by multidimensional protein identification technology. *Methods in molecular biology*. 2006; 328:159–75. [PubMed: 16785648]
23. Cox J, Hein MY, Luber CA, Paron I, Nagaraj N, Mann M. Accurate proteome-wide label-free quantification by delayed normalization and maximal peptide ratio extraction, termed MaxLFQ. *Mol. Cell. Proteomics*. 2014; 13(9):2513–26. [PubMed: 24942700]
24. Zybilov B, Mosley AL, Sardi ME, Coleman MK, Florens L, Washburn MP. Statistical analysis of membrane proteome expression changes in *Saccharomyces cerevisiae*. *J. Proteome Res*. 2006; 5(9):2339–47. [PubMed: 16944946]
25. Bubis JA, Levitsky LI, Ivanov MV, Tarasova IA, Gorshkov MV. Comparative evaluation of label-free quantification methods for shotgun proteomics. *Rapid Commun. Mass Spectrom*. 2017; 31(7): 606–612. [PubMed: 28097710]

Figure 1.

Flowchart describing the StPeter algorithm. The workflow starts with protXML input (red oval). Note that all results are appended to the protXML file that was used as input, and thus no new files are generated. Iteration is not required but can be performed to generate new results using different parameters, for example, using a different FDR threshold.

Figure 2.
 (A) Volcano plot of $-\log_{10}(p\text{-value})$ versus protein $\log_2(\text{fold change})$ of StPeter protein quantities processed with MSstats. Human proteins from the different comparisons are represented in color, contrasted with the background yeast proteins in gray. Dashed lines indicating two-fold changes in quantity (vertical line) and $p < 0.01$ (horizontal line) are provided for reference. (B) ROC curves of benchmarked LFQ tools. StPeter (magenta and cyan lines) demonstrates superior sensitivity over the 1–5% FPR range.

Figure 3.

Linear regressions of SI_N and dSI_N values and known protein concentrations. The six quantified proteins are pig, mouse, rabbit, human, rat, and cow serum albumins at concentrations of 0.1, 0.25, 0.63, 1.6, 4, and 10 pmol, respectively. Average dSI_N (red circles) and SI_N (blue triangle) values and their standard deviations were computed from the 12 replicate analyses.

Figure 4.

Large-scale protein quantification of whole proteomes. (A) Computed log₂(ratios) of observed proteins between the two conditions plotted log₂(abundance) values from the sample of lower *E. coli* concentration for four different MS²-based quantification methods.

(B) Histograms of the ratio distributions of the observed proteins.

Figure 5. Petunia graphical interface for StPeter. (A) User selects a protXML to process and optionally sets the FDR and MS2 match tolerance values. (B) Resulting quantification is displayed graphically in the protXML viewer. $\text{Log}_2(\text{SI}_N)$ values are presented to the user when sample load estimates are not provided in the parameters, as shown in panel A.

Table 1

Performance Metrics of StPeter on a Variety of Community Datasets

ID	A	B	C	D	E
name	Bendixen Milk	Hahne Placenta	Pig Retina	Vialas Candida	Wu Tcell
PMID	22837157	N/A	N/A	23811046	17519225
accession	PAe002015	PAe003818	PAe004683	PAe002111	PAe001756
instrument	ABI QStar Elite	Bruker amaZon	Thermo QExactive	Thermo Orbitrap Velos	Thermo LTQ
mzML files	11	58	13	4	17
mzML size (GB)	1.60	2.93	7.26	28.0	2.04
PSMs (1% FDR)	649	9308	114804	31030	4000
proteins	149	1535	4689	1758	1056
run time (s) ^a	2	30	36	10	15
peak memory (MB)	136	422	228	68	420

^aRun times were computed from cached files on a magnetic storage device to best evaluate actual CPU run time.