Journal of proteome research

pubs.acs.org/jpr

Technical Note

MSstatsShiny: A GUI for Versatile, Scalable, and Reproducible Statistical Analyses of Quantitative Proteomic Experiments

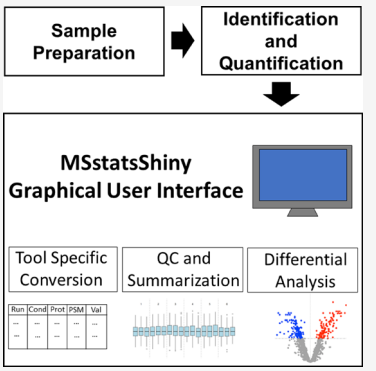
Devon Kohler, Maanasa Kaza, Cristina Pasi, Ting Huang, Mateusz Staniak, Dhaval Mohandas, Eduard Sabido, Meena Choi, and Olga Vitek*

Cite This: *J. Proteome Res.* 2023, 22, 551−556

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: Liquid chromatography coupled with bottom-up mass spectrometry (LC-MS/MS)-based proteomics is a versatile technology for identifying and quantifying proteins in complex biological mixtures. Postidentification, analysis of changes in protein abundances between conditions requires increasingly complex and specialized statistical methods. Many of these methods, in particular the family of open-source Bioconductor packages *MSstats*, are implemented in a coding language such as R. To make the methods in *MSstats* accessible to users with limited programming and statistical background, we have created *MSstatsShiny*, an R-Shiny graphical user interface (GUI) integrated with *MSstats*, *MSstatsTMT*, and *MSstatsPTM*. The GUI provides a point and click analysis pipeline applicable to a wide variety of proteomics experimental types, including label-free data-dependent acquisitions (DDAs) or data-independent acquisitions (DIAs), or tandem mass tag (TMT)-based TMT-DDAs, answering questions such as relative changes in the abundance of peptides, proteins, or post-translational modifications (PTMs). To support reproducible research, the application saves user's selections and builds an R script that programmatically recreates the analysis. *MSstatsShiny* can be installed locally via Github and Bioconductor, or utilized on the cloud at www.msstatsshiny.com. We illustrate the utility of the platform using two experimental data sets (MassIVE IDs MSV000086623 and MSV000085565).



KEYWORDS: *bioinformatics, proteomics, post-translational modifications, mass spectrometry, differential analysis, software, graphical user interface*

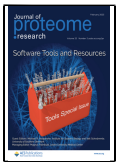
INTRODUCTION

Quantitative bottom-up mass spectrometry-based proteomic experiments face a multitude of challenges of statistical analysis. These challenges are due to a wide variety of experimental designs and data acquisition methods,[1,2] which in turn introduce numerous sources of variation, uncertainty, and missing quantitative information. A common currency in the analysis of quantitative bottom up proteomic experiments are spectral features. These are peptide ions for data-dependent acquisitions (DDAs), peptide transitions for selected reaction monitoring (SRM), and peptide ions and fragment ions for data-independent acquisitions (DIAs). Many tools, such as MaxQuant[3] and Skyline,[4] have been developed to extract, identify, and quantify features from acquired spectra. These tools make many different assumptions and decisions while processing the data, including choices of grouping spectral features into proteins, filtering out poor quality features, and reporting missing data and outliers. Statistical analyses take as input the output of these tools, and derive conclusions regarding changes in protein abundance between conditions. Therefore, statistical analysis methods and implementations must be both versatile enough to account for differences in statistical properties of data from various experimental designs, and robust enough to handle multitudes of experimental and tool-specific data processing artifacts.

The *MSstats* family of R packages has been developed for such analysis tasks,[5,6] and is widely used by the proteomics community. *MSstats* takes as input the output of tools that identify and quantify acquired spectra, and performs downstream analysis to derive protein-level conclusions. The methods in *MSstats* automatically adjust to the specific experimental designs, including designs with group comparisons or repeated measures designs, differing numbers of biological replicates, technical replicates, and fractionation. The *MSstats* family is applicable to a wide variety of data acquisitions, including DDA, DIA, SRM, and parallel reaction monitoring (PRM) experiments (*MSstats*), as well as DDA acquisitions labeled with tandem mass tags (TMTs) (*MSstatsTMT*). The packages help answer question regarding relative changes in the abundance of peptides and proteins

Special Issue: Software Tools and Resources 2023

Received: September 29, 2022
Published: January 9, 2023



(*MSstats*/*MSstatsTMT*), or relative changes in post-translational modifications (PTMs) (*MSstatsPTM*).[5−7]

The *MSstats* package family is broadly separated into a three step workflow: data conversion and preprocessing, feature-level summarization, and differential analysis, as shown in Figure 1.[8]

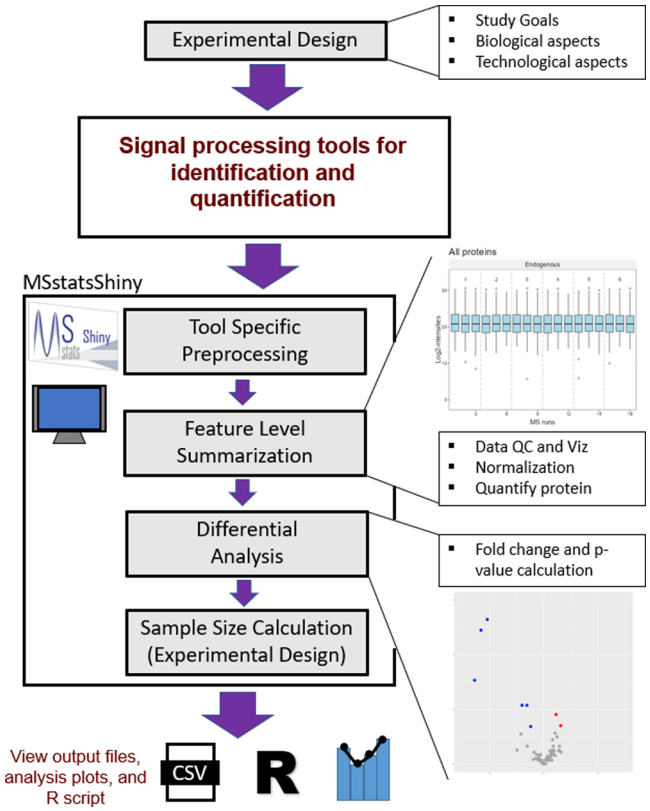


Figure 1. Experimental analysis workflow of a typical proteomic experiment. *MSstatsShiny* takes as input the output of spectral processing tools used for identification and quantification. The functionalities of the application are separated into tabs, including data preprocesssing, feature summarization, differential analysis, and sample size calculation. Visualizations can be created at each step of the analysis to review the results. The GUI outputs the results of the analysis in a variety of formats, including raw CSV data files, PDF files of plots, and an R script to recreate the analysis.

The packages take as input a list of identified and quantified peak intensities produced by a spectral processing tool (as opposed to a list of ratios of peak intensities between groups). To this end, the packages include a broad range of converters for different spectral processing tools. Currently converters are available for a wide variety of tools, including MaxQuant, Skyline, and Spectronaut,[9] with a full list available in the Design and Implementation section. The converters transform the data into the format required for *MSstats* and include additional preprocessing steps, such as quality control and feature filtering, to ensure only high quality observations are included. The converters output a list of $\log_2$-intensities used to characterize the identified protein in the required format for the data summarization step.

The summarization and differential analysis steps are both part of the statistical modeling and inference procedure employed by *MSstats*. In the summarization step, the feature $\log_2$-intensities are first modeled to determine a threshold for feature intensities below a limit of confident quantification. A subset of the intensities below the threshold, which do not correspond to reference standards, and which are acquired in a run where another confident quantification for this protein is present, are imputed using a two-way additive Accelerated Failure Time Model.[10] After the imputation, the feature intensities are summarized into a single value per protein per run using Tukey's median polish (TMP),[11] a simple and robust procedure which iteratively fits a two-way additive model.

In the differential analysis step, the summarized values are used to fit a linear mixed-effects model appropriate for the experimental design.[12−14] From this model, for any comparison of conditions of interest, *MSstats* derives model-based estimates of the $\log_2$fold changes, and the corresponding standard errors and p-values. In the special case of a balanced designs with no outliers and no missing values, the approach to summarization and differential analysis employed by MSstats is equivalent to the analysis of split-plot designs.[15] It has been proven to be more robust and broadly applicable compared to using one linear mixed effects model with all features.[8]

Additionally, *MSstatsTMT* includes optional functionality to adjust model variance through Empirical Bayes Moderation,[16] and *MSstatsPTM* includes statistical methods to remove confounding between changes in abundance of the PTM and changes in abundance of the unmodified protein.[7] The *MSstats* packages have been proven to effectively analyze the output of many different experimental designs, e.g., within the MassIVE.quant platform.[17]

While *MSstats* is an effective tool for quantitative proteomic analysis, so far the methods have only been implemented as coding packages in R, requiring users to have coding skills in order to utilize them effectively. This often limits the users to bioinformaticians or experimentalists with strong computational skills.[18] Additionally, since the tools are dispersed into separate packages, the user must choose the package that best fits their specific experiment.

Several applications have been developed to increase the accessibility of postsearch quantitative proteomic analysis methods to the wider proteomic community. The *Galaxy* project has recently implemented *MSstats* and *MSstatsTMT* directly into their GUI, https://usegalaxy.eu/.[19,20] Users have access to the core functionalities of the packages, including converters for several spectral processing tools, missing value imputation, feature summarization, and differential analysis. While the core functionalities are available, many spectral processing converters are currently missing, including tools for Progenesis (Nonlinear Dynamics/Waters), Proteome Discoverer,[21] Spectronaut, DIA-Umpire,[22] SpectroMine (Biognosys), and Philosopher,[23] meaning that users of these tools need to manually convert the data into the correct format before using Galaxy. Additionally, there is no functionality for processing experiments targeting PTMs (using MSstatsPTM).

Beyond Galaxy, there are a number of GUI-based applications which utilize non-*MSstats*-based methods for downstream postidentification proteomic analysis. *Amica*[24] is a shiny-based GUI for proteomic experiments that includes converters for MaxQuant and Fragpipe,[25] and utilizes *DEqMS*[26] and *Limma*[27] to perform differential analysis. *LFQ-Analyst*[28] is designed specifically to analyze and visualize proteomics data output from MaxQuant, and includes differential analysis options using *Limma*. *ProTIGY*[29] is a general analysis tools for omics data and supports any data that can be organized as a $p \times n$ matrix with p features and n samples. *ProTIGY* contains converters for Spectrum Mill

https://doi.org/10.1021/acs.jproteome.2c00603
J. Proteome Res. 2023, 22, 551−556

(Agilent) and MaxQuant, and uses methods from *Limma* for differential analysis. *SQuAPP*[30] is a shiny-based application which includes a variety of vizualizations, quality control functionality, as well as data processing and differential analysis using *Limma*. The *START App*,[31] originally designed for RNaseq-based analysis, includes a customized instance to visualize and perform differential analysis on proteomic experiments targeting phosphorylation (phosphoproteomics). While there are a multitude of applications that can be used to analyze proteomic data, they are limited in the spectral processing tools they can handle, the types of experimental designs that can be analyzed, and most of them only use variations of the *Limma* implementation that are not specifically designed for quantitative proteomics.[16,32]

To make the full scope of methods in the *MSstats* family accessible to users with limited programming and statistical background, we have developed the R-shiny-based application *MSstatsShiny* (Figure 1). The application is general, including functionality for a variety of acquisition types, experimental designs, and spectral processing tools. The application supports reproducible research by saving selections for future processing. It is scalable, with options for handling large data sets. *MSstatsShiny*'s integration with the *MSstats* family of packages gives users access to high-quality statistical analysis methods and visualizations without having to code in R. The application improves on the Galaxy integration with *MSstats* through adding PTM analysis, all available converters to the workflow, and including unique functionality such as code generation and peptide level analysis. *MSstatsShiny* is open source, available in both a cloud-based environment online, http://www.msstatsshiny.com, and locally downloadable via Github, https://github.com/Vitek-Lab/MSstatsShiny, or Bioconductor, https://bioconductor.org/packages/release/bioc/html/MSstatsShiny.html

DESIGN AND IMPLEMENTATION

MSstatsShiny utilizes a tab-based design to organize and facilitate the analysis workflow (Figure 2). There are six main tabs in the platform, including "Homepage", "Data Uploading", "Data Processing", "Statistical Inference", "Future Experiments", and "Help". The "Homepage" gives users an overview of how the GUI works, updates on new features and information on where to access help. The "Data Uploading" page requires the user to select the experimental design and the spectral processing tool used. Additionally, this tab lets users select if they would like to analyze their data on a protein or peptide level. In general *MSstatsShiny* is not built for peptide level analysis due to low feature counts creating different methodological challenges; however, this option is available if required. The "Data Processing" page provides options for feature summarization and missing value imputation. Additionally, this step includes a variety of visualization options to illustrate the quantitative data after data summarization and quality control of MS runs. The "Statistical Inference" tab provides the user with the results of the statistical analysis, and gives them the options to download the results. Visualizations of the differential analysis, such as volcano plots and heatmaps, are also included here. The Future Experiments tab is used for power analysis and sample size calculations for future experiments. Throughout all steps in the platform there are explanations of the processing options, and the "Help" tab provides links to more in depth documentation.

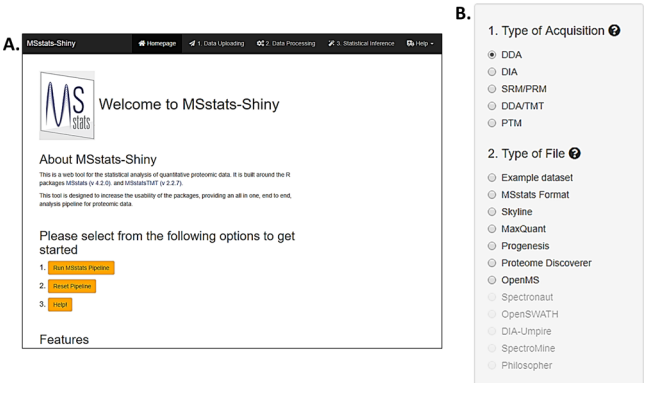


Figure 2. Example tabs of the *MSstatsShiny* GUI. A) GUI Homepage showing an overview of the platform, how to get started, and the main processing tabs. B) Data Uploading tab, including all available experimental design selections.

Integration with *MSstats* Package Family Provides General Analysis Workflow for a Variety of Experiments

MSstatsShiny is directly integrated with *MSstats* (version 4.0+), *MSstatsTMT* (version 2.0+), and *MSstatsPTM* (version 2.0+). Functions from each package are automatically run in the background depending on the experimental design and options input by the user. The workflows for all three packages are executed in the same way, encompassing the entire analysis pipeline, from preprocessing, to summarization and differential analysis. Integration with the packages gives users the ability to analyze a wide variety of experiments. Figure 2B shows the options available in the data upload step. First the user must select the appropriate experimental design, which involves the type of sample preparation, data acquisition method, and biological question. For label-free experiments targeting relative protein quantification the user should select either DDA, DIA, or SRM. For TMT sample preparation targeting relative protein quantification, the user should select the TMT option. Finally, for any experiment targeting relative PTM quantification, the user should use the PTM option. The GUI will automatically call functions from the appropriate *MSstats* package depending on the user's selections.

MSstatsShiny includes DDA converters for Skyline, MaxQuant, Progenesis, Proteome Discoverer, and OpenMS,[33] DIA converters for Skyline, Spectronaut, OpenSWATH,[34] and DIA-Umpire, and a SRM/PRM converter for Skyline. TMT converters include MaxQuant, Proteome Discoverer, OpenMS, SpectroMine, and Philosopher. Finally, PTM includes a converter for MaxQuant. The application seamlessly integrates the functionalities from three separate R packages, providing users with a general GUI that can be used to process many types of experiments.

Cloud-Based and Local Installation Environments Provide Scalable Options

MSstatsShiny is available both as a downloadable application and in a cloud-based environment online. The cloud-based option provides users with a quick and easy solution with all the features of the local install. However, due to processing limits on the cloud environment, this option should be exclusively used for small data sets, under 250 MB. Meanwhile, when downloading and using the application locally, the size limit will depend on the memory available on the local machine. Additionally, the local option may be better for users

https://doi.org/10.1021/acs.jproteome.2c00603
J. Proteome Res. 2023, 22, 551−556

Table 1. Case Study Overview Including Computational Time and Memory Used during Analysis[a]

Study	Acquisition	No. of proteins	No. of bio. replicates	Upload	Summarization	Model	RAM Usage	MSV ID
Data set 1: Rat - Global Protein - DIA Label-free (GUI)	DIA	281	10	14 min 02 s	52 min 35 s	1 min 14 s	20 GB	MSV000086623
Data set 1: Rat - Global Protein - DIA Label-free (code)	DIA	281	10	10 min 48 s	52 min 22 s	23 s	16 GB	MSV000086623
Data set 2: Mouse - Phosphorylation -2mix-TMT (GUI)	TMT	9542	4 or 3	15 s	15 min 52 s	9 min 12 s	5 GB	MSV000085565

[a]"Study" denotes the experiment code name and processing type (either GUI or using R). "No. of proteins" is the total number of proteins in the dataset. "No. of bio. replicates" shows the number of biological replicates per condition. "Upload", "Summarization", and "Model" show the amount of time each step took to complete. "RAM Usage" shows the maximum amount of RAM utilized during the analysis. "MSV ID" is the ID of the MassIVE.quant analysis container, containing the data needed to reprocess the case study.

with data privacy concerns. As new features get added to the platform, the cloud-based application will be updated automatically. Meanwhile in the local version any new updates will need to be downloaded manually from Github or Bioconductor.

Detailed install instructions for the local install are available in Bioconductor and the Github repository.

Code Generation Makes Analysis Reproducible and Scalable

To facilitate both reproducibility and scalability, *MSstatsShiny* includes functionality to generate the fully executable R script recreating the user's analysis. As the user goes through the application, the application automatically tracks the selected parameters. Once the user completes the modeling step, the application provides a button for downloading the R script version of the analysis. The application creates a full analysis script, complete with all user selections and *MSstats* functions, as well as scripts for easy data visualization. It identifies the versions of the software used in the analysis. The output of this code will exactly reproduce the results shown in *MSstatsShiny*. This code can then be shared and used to recreate the analysis in the future.

In addition to making analysis easier to reproduce, the code functionality also increases scalability. When experimental files become large, the GUI can suffer performance degradation. This is mainly due to a baseline amount of RAM required to display the GUI, a limitation not present with command line analyses. For large data sets, greater than 5 GB, we recommend analyzing a small subset of data through the GUI platform, and selecting the processing options. Once the subset analysis is complete, the user can download the R script file and run the full data set using the script. Using the script file will reduce the usage of RAM, allowing larger files to be analyzed. This method allows users to scale up their analysis without the need for advanced coding methods. For data sets that do not fit into memory, the downloaded code can be combined with more advanced methods, such as using R packages that implement MapReduce functionality.[35−38]

CASE STUDIES

We illustrate the utility of the *MSstatsShiny* platform using two experimental data sets. The data sets were chosen to exhibit the generalizability of the GUI. The data for both studies is available in the MassIVE data repository (IDs listed below) .[17] An overview of each data set and their performance is provided in Table 1. All analyses were performed using a local installation of *MSstatsShiny* on a Windows 10 computer with an Intel i7−7700K processor running at 4.2 GHz using 32 GB of RAM. The R code generated by *MSstatsShiny* for each case study is available as Supporting Information.

Data set 1: Rat - Global Protein - DIA Label-free

Experimental Overview. Stark et al.[39] analyzed the impact of RIP1 kinase deficiency on the cell death pathway in rats. RIP1 kinase-dead (KD) rats and baseline wild type (WT) rats were compared in multiple tests to determine if there were statistical differences in protein abundances. The data were acquired using DIA, and the resulting DIA mass spectrometric data were identified and quantified using Spectronaut Pulsar X software. The instrument used, search parameters, and detailed data acquisition are described in the manuscript.[39] Statistical analysis in the original publication was performed using the command line version of *MSstats*. The raw data were available in MassIVE ID MSV000086623.

MSstatsShiny Processing. A local installation of *MSstatsShiny* was required to process this experiment, as the Spectronaut files totalled over 7 GB and were too large for the cloud version. Two data files were input into the *MSstatsShiny* platform: a raw Spectronaut file and an MSstats annotation file (both available in MassIVE under "Quantification Results"). The DIA and Spectronaut options were chosen on the data uploading page. The data was summarized using the top 200 features via Tukey's Median Polish. Custom comparisons were made using a manually created contrast matrix in the modeling step, resulting in 16 comparisons across conditions. Due to this file being large, it took a large amount of RAM and a long processing time to complete the analysis. To exhibit the scalability of the code download option, we also ran this study using R code generated by the platform. While both the GUI and code produced the same results, the code was significantly faster in the data upload step and utilized 20% less RAM.

Results. The investigators originally utilized the command line version of *MSstats* requiring technological expertise in order to analyze the experiment. *MSstatsShiny* allowed us to completely skip the command line and run the entire analysis in a point and click GUI environment. All the functionality utilized in the original analysis was available, including recreating the contrast matrix so the same comparisons could be made. Using *MSstatsShiny* resulted in a simplified analysis workflow which those without knowledge of R code could recreate.

Data set 2: Mouse - Phosphorylation -2mix-TMT

Experimental Overview. Masculins et al.[40] investigated primary murine macrophages infected with *Shigella flexneri* (*S. flexneri*). The experiment included both a total proteome (i.e., global profiling run) and a phosphopeptide enrichment run.

https://doi.org/10.1021/acs.jproteome.2c00603
J. Proteome Res. 2023, 22, 551−556

Both enriched and total proteome abundances were measured in WT and ATG16L1-deficient (cKO) samples at three points: uninfected, early infection, and late infection. Twenty-two samples were allocated to two 11-plex TMT mixtures in an unbalanced design. The instrument used, search parameters, and detailed data acquisition are described in the paper.[40] The original statistical analysis was performed using the command line version of *MSstatsTMT*. The raw data is available in MassIVE ID MSV000085565 and the input data for *MSstatsShiny* is available in reanalysis container RMSV000000357.2.

MSstatsShiny Processing. Identification and quantification of the acquired spectra and conversion into *MSstats* format was performed with a proprietary tool, so conversion could not be performed in *MSstatsShiny*. More information on converting data into *MSstats* format without using a dedicated converter can be found in the *MSstats* user guide. The data in *MSstats* format was uploaded to the platform using the "MSstats Format" option. The experimental data files were under 250 MB, and could be processed using the cloud version of the application. Both the phosphorylated and global profiling data sets were uploaded to *MSstatsShiny* with "PTM" selected on the data upload page. Data summarization and modeling were performed separately for each data set. *MSstatsShiny* automatically applied functionality from *MSstatsPTM* to combine the resulting models to eliminate the confounding between changes in PTM abundance and the unmodified protein. The GUI analyzed the results quickly and with minimal RAM usage.

Results. Masculins et al. utilized *MSstatsTMT* to model the phosphorylation and global profiling. This required knowledge of R and the command line version of *MSstatsTMT*. Additionally, they did not adjust the PTM model to remove confounding with the total proteome. Using the *MSstatsShiny* application, we were able to model the data sets without directly coding in R, and we leveraged internal functionalities from *MSstatsPTM* to remove confounding with the modification and the underlying protein. Additionally, the platform automatically generated the R script for this analysis, allowing future users to recreate the exact result.

CONCLUSIONS

MSstatsShiny is a general, reproducible, and scalable data analysis platform. It allows users to utilize the *MSstats* family of packages without needing any prior knowledge of coding. It is applicable to a wide variety of acquisition methods, spectral processing tools, and experimental design, compiling all methods into one easy to use tool. The application opens up reproducible, robust analysis methods to the entire proteomics community, enabling all researchers to perform quality data analysis and encouraging collaboration.

ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jproteome.2c00603.

R code produced by *MSstatsShiny* for reanalysis of case studies: Case_Study1_R_code.R, and Case_Study2_R_code.R (ZIP)

AUTHOR INFORMATION

Corresponding Author

Olga Vitek − *Khoury College of Computer Science, Northeastern University, Boston, Massachusetts 02115, United States;* orcid.org/0000-0003-1728-1104; Email: o.vitek@northeastern.edu

Authors

Devon Kohler − *Khoury College of Computer Science, Northeastern University, Boston, Massachusetts 02115, United States;* orcid.org/0000-0001-7301-2596

Maanasa Kaza − *Khoury College of Computer Science, Northeastern University, Boston, Massachusetts 02115, United States*

Cristina Pasi − *Universitat Oberta de Catalunya, Barcelona 08018, Spain*

Ting Huang − *Khoury College of Computer Science, Northeastern University, Boston, Massachusetts 02115, United States*

Mateusz Staniak − *University of Wrocław, Wrocław 50-137, Poland*

Dhaval Mohandas − *EXL Service, New York, New York 10022, United States*

Eduard Sabido − *Center for Genomic Regulation, Barcelona Institute of Science and Technology, Barcelona 08003, Spain; Universitat Pompeu Fabra, Barcelona 08002, Spain;* orcid.org/0000-0001-6506-7714

Meena Choi − *MPL, Genentech, South San Francisco, California 94080, United States*

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jproteome.2c00603

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

This work was supported in part by NSF DBI-1759736 and the Chan-Zuckerberg Essential Open-Source Software Award to O.V. The CRG/UPF Proteomics Unit is part of the Spanish Infrastructure for Omics Technologies (ICTS OmicsTech) and it is member of ProteoRed PRB3 consortium which is supported by grant PT17/0019 of the PE I+D+i 2013-2016 from the Instituto de Salud Carlos III (ISCIII), ERDF, and "Secretaria d'Universitats i Recerca del Departament d'Economia i Coneixement de la Generalitat de Catalunya" (2017SGR595). We also acknowledge support of the Spanish Ministry of Science and Innovation to the EMBL partnership, the Centro de Excelencia Severo Ochoa, the CERCA Programme/Generalitat de Catalunya, and the European Union's Horizon 2020 research and innovation program (GA 823839; EPIC-XS).

REFERENCES

(1) Kumar, C.; Mann, M. Bioinformatics analysis of mass spectrometry-based proteomics data sets. *FEBS Lett.* **2009**, *583*, 1703−1712. Prague Special Issue: Functional Genomics and Proteomics

(2) Schubert, O. T.; Röst, H. L.; Collins, B. C.; Rosenberger, G.; Aebersold, R. Quantitative proteomics: challenges and opportunities in basic and applied research. *Nature Protocol* **2017**, *12*, 1289−1294.

(3) Cox, J.; Mann, M. MaxQuant enables high peptide identification rates, individualized p.p.b.-range mass accuracies and proteome-wide protein quantification. *Nat. Biotechnol.* **2008**, *26*, 1367−1372.
(4) Pino, L.; Searle, B.; Bollinger, J.; Nunn, B.; MacLean, B.; MacCoss, M. The Skyline ecosystem: Informatics for quantitative mass spectrometry proteomics. *Mass Spectrom. Rev.* **2020**, *39*, 229−244.
(5) Choi, M.; Chang, C.; 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*, 2524−2526.
(6) Huang, T.; Choi, M.; Tzouros, M.; Golling, S.; Pandya, N. J.; Banfai, B.; Dunkley, T.; Vitek, O. MSstatsTMT: statistical detection of differentially abundant proteins in experiments with isobaric labeling and multiple mixtures. *Mol. Cell. Proteomics* **2020**, *19*, 1706−1723.
(7) Kohler, D.; Tsai, T.; Verschueren, E.; Huang, T.; Hinkle, T.; Phu, L.; Choi, M.; Vitek, O. MSstatsPTM: Statistical relative quantification of post-translational modifications in bottom-up mass spectrometry-based proteomics. *Mol. Cell. Proteomics* **2022**, 100477.
(8) Choi, M. A flexible and versatile framework for statistical design and analysis of quantitative mass spectrometry-based proteomic experiments. Ph.D. Thesis, Purdue University, 2016.
(9) Martinez-Val, A.; Bekker-Jensen, D. B.; Hogrebe, A.; Olsen, J. V. In *Proteomics Data Analysis*; Cecconi, D., Ed.; Springer US: New York, NY, 2021; pp 95−107.
(10) Wei, L. The accelerated failure time model: a useful alternative to the Cox regression model in survival analysis. *Statistics in medicine* **1992**, *11*, 1871−1879.
(11) Tukey, J. W. *Exploratory data analysis*; Addison-Wesley, 1977.
(12) McLean, R. A.; Sanders, W. L.; Stroup, W. W. A unified approach to mixed linear models. *Am. Statistician* **1991**, *45*, 54−64.
(13) Faraway, J. J. *Extending the linear model with R: generalized linear, mixed effects and nonparametric regression models*; CRC Press, 2006.
(14) Bates, D.; Mächler, M.; Bolker, B.; Walker, S. Fitting linear mixed-effects models using lme4. *J. Stat. Soft.* **2015**, *67*, 1−48.
(15) Kutner, M.; Nachtsheim, C.; Neter, J.; Li, W. *Applied linear statistical models*, 5th ed.; McGraw-Hill Irwin: Boston, MA, USA, 2005.
(16) Smyth, G. K. Linear models and empirical bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology* **2004**, *3*, No. Article 31.
(17) Choi, M.; et al. MassIVE.quant: a community resource of quantitative mass spectrometry-based proteomics datasets. *Nat. Methods* **2020**, *17*, 981−984.
(18) Vincent, A. T.; Charette, S. J. Who qualifies to be a bioinformatician? *Front. Genetics* **2015**, *6*. DOI: 10.3389/fgene.2015.00164
(19) Giardine, B.; Riemer, C.; Hardison, R.; Burhans, R.; Elnitski, L.; Shah, P.; Zhang, Y.; Blankenberg, D.; Albert, I.; Taylor, J.; Miller, W.; et al. Galaxy: a platform for interactive large-scale genome analysis. *Genome Res.* **2005**, *15*, 1451−1455.
(20) Pinter, N.; Glätzer, D.; Fahrner, M.; Fröhlich, K.; Johnson, J.; Grüning, B. A.; Warscheid, B.; Drepper, F.; Schilling, O.; Föll, M. C. MaxQuant and MSstats in galaxy enable reproducible cloud-based analysis of quantitative proteomics experiments for everyone. *J. Proteome Res.* **2022**, *21*, 1558−1565.
(21) Orsburn, B. Proteome Discoverer - a community enhanced data processing suite for protein informatics. *Proteomes* **2021**, *9*, 15.
(22) Tsou, C.-C.; Avtonomov, D.; Larsen, B.; Tucholska, M.; Choi, H.; Gingras, A.-C.; Nesvizhskii, A. I. DIA-Umpire: comprehensive computational framework for data-independent acquisition proteomics. *Nat. Methods* **2015**, *12*, 258−264.
(23) Da Veiga Leprevost, F.; Haynes, S. E.; Avtonomov, D. M.; Chang, H.-Y.; Shanmugam, A. K.; Mellacheruvu, D.; Kong, A. T.; Nesvizhskii, A. I. Philosopher: a versatile toolkit for shotgun proteomics data analysis. *Nat. Methods* **2020**, *17*, 869−870.
(24) Didusch, S.; Madern, M.; Hartl, M.; Baccarini, M. amica: an interactive and user-friendly web-platform for the analysis of proteomics data. *BMC Genomics* **2022**, *23*, 817.
(25) Kong, A.; Leprevost, F.; Avtonomov, D.; Mellacheruvu, D.; Nesvizhskii, A. MSFragger: ultrafast and comprehensive peptide identification in mass spectrometry-based proteomics. *Nat. Methods* **2017**, *14*, 513−520.
(26) Zhu, Y.; Orre, L.; Tran, Y.; Mermelekas, G.; Johansson, H.; Malyutina, A.; Anders, S.; Lehtiö, J. DEqMS: a method for accurate variance estimation in differential protein expression analysis. *Mol. Cell. Proteomics* **2020**, *19*, 1047−1057.
(27) Ritchie, M. E.; Phipson, B.; Wu, D.; Hu, Y.; Law, C. W.; Shi, W.; Smyth, G. K. limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Res.* **2015**, *43*, No. e47.
(28) Shah, A. D.; Goode, R. J. A.; Huang, C.; Powell, D. R.; Schittenhelm, R. B. LFQ-Analyst: An easy-to-use interactive web platform to analyze and visualize label-free proteomics data preprocessed with MaxQuant. *J. Proteome Res.* **2020**, *19*, 204−211.
(29) Krug, K.; Clark, N. ProTIGY, https://github.com/broadinstitute/protigy.
(30) Ergin, E. K. SQuAPP, https://github.com/LangeLab/SQuAPP.
(31) Nelson, J. W.; Sklenar, J.; Barnes, A. P.; Minnier, J. The START App: a web-based RNAseq analysis and visualization resource. *Bioinformatics* **2016**, *33*, 447−449.
(32) Smyth, G. K. In *Bioinformatics and Computational Biology Solutions Using R and Bioconductor*; Gentleman, R., Carey, V., Huber, W., Irizarry, R., Dudoit, S., Eds.; Springer New York: New York, NY, 2005; pp 397−420.
(33) Röst, H. L.; et al. OpenMS: a flexible open-source software platform for mass spectrometry data analysis. *Nat. Methods* **2016**, *13*, 741−748.
(34) Röst, H. L.; Rosenberger, G.; Navarro, P.; Gillet, L.; Miladinović, S. M.; Schubert, O. T.; Wolski, W.; Collins, B. C.; Malmström, J.; Malmström, L.; Aebersold, R. OpenSWATH enables automated, targeted analysis of data-independent acquisition MS data. *Nat. Biotechnol.* **2014**, *32*, 219−223.
(35) Dean, J.; Ghemawat, S. MapReduce: Simplified Data Processing on Large Clusters. *Communications of the ACM* **2008**, *51*, 107−113.
(36) Pradeep, S.; Moy, P. Handling large data sets in R, https://rpubs.com/msundar/large_data_analysis.
(37) Sethia, K. Learn to Write MapReduce in R Step-by-Step, https://summerofhpc.prace-ri.eu/learn-to-write-mapreduce-in-r-step-by-step/.
(38) Lee Yung Rowe, B. From functional programming to MapReduce in R, https://www.r-bloggers.com/2015/09/from-functional-programming-to-mapreduce-in-r/.
(39) Stark, K.; Goncharov, T.; Varfolomeev, E.; Xie, L.; Ngu, H.; Peng, I.; Anderson, K.; Verschueren, E.; Choi, M.; Kirkpatrick, D.; Easton, A.; Webster, J.; McKenzie, B.; Vucic, D.; Bingol, B. Genetic inactivation of RIP1 kinase activity in rats protects against ischemic brain injury. *Cell Death Dis.* **2021**, *12*, 379.
(40) Maculins, T.; et al. Multiplexed proteomics of autophagy-deficient murine macrophages reveals enhanced antimicrobial immunity via the oxidative stress response. *eLife* **2021**, *10*, No. e62320.

https://doi.org/10.1021/acs.jproteome.2c00603
J. Proteome Res. 2023, 22, 551−556