Journal of proteome research

Article

Cite This: J. Proteome Res. 2019, 18, 1477−1485

pubs.acs.org/jpr

Spectral Clustering Improves Label-Free Quantification of Low-Abundant Proteins

Johannes Griss,*[,†,‡] Florian Stanek,[§,∥] Otto Hudecz,[§,∥] Gerhard Dürnberger,[§,∥,⊥] Yasset Perez-Riverol,[‡] Juan Antonio Vizcaíno,[‡] and Karl Mechtler[§,∥]

[†]Department of Dermatology, Medical University of Vienna, Währinger Gürtel 18-20, 1090 Vienna, Austria

[‡]European Molecular Biology Laboratory, European Bioinformatics Institute (EMBL-EBI), Wellcome Trust Genome Campus, CB10 1SD Hinxton, Cambridge, United Kingdom

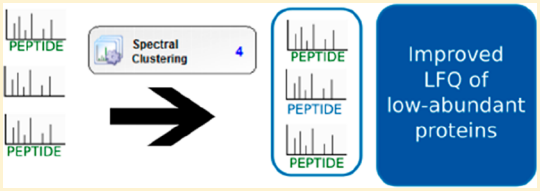
[§]Research Institute of Molecular Pathology (IMP), Vienna Biocenter (VBC), Campus-Vienna-Biocenter 1, 1030 Vienna, Austria

[∥]Institute of Molecular Biotechnology of the Austrian Academy of Sciences (IMBA), Vienna Biocenter (VBC), Dr. Bohr-Gasse 3, 1030 Vienna, Austria

[⊥]Gregor Mendel Institute of Molecular Plant Biology (GMI), Vienna Biocenter (VBC), Dr. Bohr-Gasse 3, 1030 Vienna, Austria

Supporting Information

ABSTRACT: Label-free quantification has become a common-practice in many mass spectrometry-based proteomics experiments. In recent years, we and others have shown that spectral clustering can considerably improve the analysis of (primarily large-scale) proteomics data sets. Here we show that spectral clustering can be used to infer additional peptide-spectrum matches and improve the quality of label-free quantitative proteomics data in data sets also containing only tens of MS runs. We analyzed four well-known public benchmark data sets that represent different experimental settings using spectral counting and peak intensity based label-free quantification. In both approaches, the additionally inferred peptide-spectrum matches through our *spectra-cluster* algorithm improved the detectability of low abundant proteins while increasing the accuracy of the derived quantitative data, without increasing the data sets' noise. Additionally, we developed a Proteome Discoverer node for our *spectra-cluster* algorithm which allows anyone to rebuild our proposed pipeline using the free version of Proteome Discoverer.



KEYWORDS: *label-free quantification, spectral counting, spectral clustering, Proteome Discoverer, benchmarking study, IMP free nodes, Proteome Discoverer node, bioinformatics, mass spectrometry, proteomics*

INTRODUCTION

Label-free quantification (LFQ) has become a core method to derive quantitative data from mass spectrometry (MS)-based proteomics data. Especially in clinical settings, the number of samples often surpasses the number of reagents available in isobaric tag-based experiments, such as iTRAQ, TMT, and/or SILAC. Additionally, at the start of clinical studies, the total number of patients might not be foreseeable while initial results are required as quickly as possible to potentially adapt working hypotheses. Another advantage of using LFQ approaches is the reduced cost, as no expensive tagging reagents are needed.

Two main methods exist for LFQ in MS-based proteomics. Peak-intensity based quantification estimates peptide abundance based on the MS1 precursor ion intensity while spectral counting based approaches simply use the number of spectra identified per protein (peptide spectrum matches, PSMs) to subsequently perform protein-based quantification. Spectral counting-based methods have the disadvantage that they can only utilize *identified* MS2 spectra. To achieve reproducible results, the mass spectrometer must fragment the same precursor ions in every MS run and subsequently record MS2 spectra of sufficient quality in order for them to be identified by the search engine. Many label-free peak intensity based algorithms circumvent this downside by employing the so-called "match-between-runs" (MBR) approach.[1] Here, the retention times of all MS runs are aligned and the peptide identifications are propagated between runs based on matching MS1 features. This greatly increases the number of usable events for quantification, while decreasing the number of missing values and thereby potentially improving the overall reproducibility.

Among the other existing applications of spectral clustering,[2] we and others have recently shown that it can be used as a first step to help in the identification of unidentified spectra by inferring identifications from consistently identified spectra included in the same spectral cluster.[3,4] If, for example, two experiments measure the same protein in high and low abundance, respectively, search engines will more likely identify the high quality spectra from the first one. Through clustering, low quality spectra can be clustered with high quality spectra and thus can be identified. At the time of writing, four main clustering algorithms exist for MS/MS based proteomics data

Received: May 28, 2018
Published: March 12, 2019

DOI: 10.1021/acs.jproteome.8b00377
J. Proteome Res. 2019, 18, 1477−1485

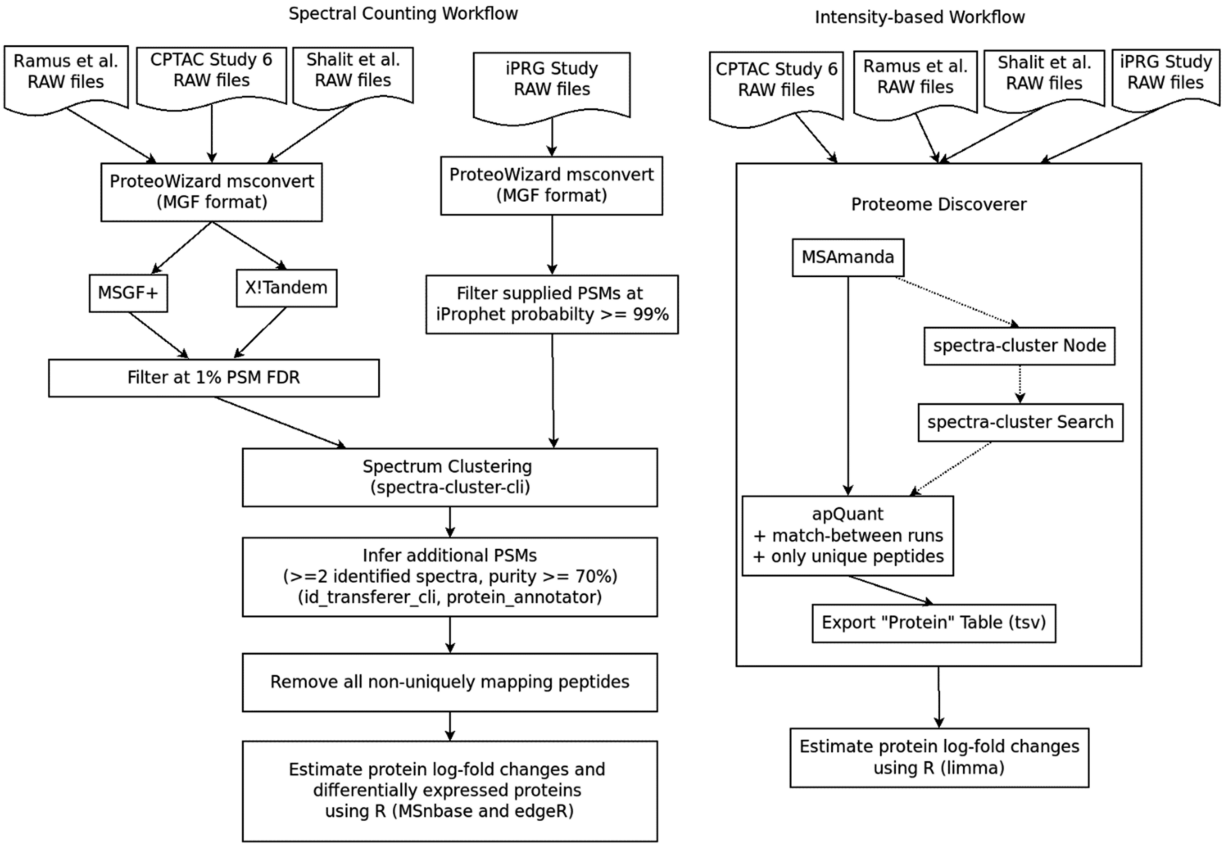


Figure 1. Overview of used workflows to assess the influence of additionally inferred PSMs through clustering on LFQ, for both spectral counting and intensity-based approaches.

that are capable of processing millions of MS/MS spectra: msCluster,[4] MaRaCluster,[5] *spectra-cluster*,[3] and msCRUSH.[6]

Originally, we developed our *spectra-cluster* algorithm to process repository sized data sets, such as the data sets submitted to the PRIDE Archive.[3,7] There, it allowed us to identify correctly and incorrectly identified spectra, as well as millions of consistently unidentified spectra. In short, the *spectra-cluster* algorithm is a greedy clustering algorithm merging the first spectra that pass the set threshold. To reduce the risk of incorrect matches, clustering is performed in different rounds with decreasing thresholds. The current default setting (used in this manuscript) is to start with a target accuracy of 100% (only nearly identical spectra are merged) and over 5 rounds reach a final accuracy of 99% (1% incorrectly clustered spectra). The similarity of two spectra is assessed using a probabilistic score where the similarity of the m/z values is assessed using a hypergeometric distribution and the similarity of the intensity values, using the Kendall Tau correlation. The two p-values are combined using Fisher's method (for a detailed description see ref 3).

Here, we propose to use the additional identifications inferred through spectrum clustering to improve the accuracy of LFQ-based methods. We show that spectral clustering can increase the accuracy of both spectral counting based and peak-intensity based methods by benchmarking our approach using four established published data sets. Finally, we present a novel Proteome Discoverer (PD, Thermo Fisher Scientific) node that incorporates our *spectra-cluster* algorithm[3] into the widely used PD software suite, making it easily available to the wider proteomics community.

EXPERIMENTAL SECTION

Test Data Sets

We used four established public benchmark data sets to enable a direct comparison with previous results: (i) a data set coming from the Clinical Proteomic Technologies Assessment for Cancer (CPTAC) study 6;[8] (ii) a data set from the Association of Biomolecular Research Facilities Proteome Informatics Research Group (iPRG) 2015 study;[9] (iii) a benchmark data set run on a Orbitrap QExactive Plus generated by Shailt et al.[10] (PXD001385); and (iv) a benchmark data set run on a LTQ-Orbitrap Velos generated by Ramus et al.[11] (PXD001819). In all cases we downloaded the original RAW files (CPTAC study: https://cptac-data-portal.georgetown.edu; iPRG study: ftp://iprg_study@ftp.peptideatlas.org/, password "ABRF329"; Shailt et al. and Ramus et al., from PRIDE Archive[12]) and converted them to MGF format using ProteoWizard's msconvert tool (version 3.0.9393)[13] and the corresponding vendor libraries.

From the CPTAC study, we only evaluated the Sigma Universal Proteomics Standard (UPS) spiked-in samples into a yeast background. These consist of replicate measurements of six samples containing 60 ng/μL yeast lysate spiked with various concentrations of the 48 UPS 1 proteins (0.25−20 fmol/L, samples A−E in ascending concentrations). These samples were sent to different laboratories for analysis. For this benchmark, we used three of the four Orbitrap data sets: two from site_65 (LTQ-XL-OrbitrapP − "site_65-OrbiP" throughout the manuscript - and LTQ-OrbitrapW) and the one from site_86.

The iPRG study was purely focused on the evaluation of bioinformatic algorithms and approaches. The study partic-

DOI: 10.1021/acs.jproteome.8b00377
J. Proteome Res. 2019, 18, 1477−1485

ipants were provided with a set of RAW files, search results, and extracted precursor ion intensities. It was up to the study participants which data to use. Four samples were measured in triplicate, each containing 6 spiked-in proteins in different concentrations into a background of 200 ng of yeast digest (see Table 1 in ref 9). To make our results as comparable as possible to the iPRG study results, we chose to use the provided search results for our analysis in the spectral counting pipeline but reanalyzed the RAW files when using the PD pipeline (Figure 1).

Shalit et al. spiked four different amounts of *E. coli* lysate (3 ng, 7.5 ng, 10 ng, 15 ng) into a HeLa cell background and analyzed three replicates per concentration value. The analysis then represented concentration changes of 5:1, 2:1, and 1.5:1 where all amounts were compared against the 15 ng one.

Ramus et al. spiked nine different concentrations of UPS1 proteins into a yeast background (from 50 amol to 50,000 amol) and analyzed three replicates per concentration value. In the original manuscript, three comparisons were performed: 5,000 amol vs 50,000 amol (1:10), 500 amol vs 50,000 amol (1:100), and 12,500 amol vs 25,000 amol (1:2).

Spectral Clustering and Spectral Counting Based Quantification

The complete workflow is summarized in Figure 1. Searches were performed using X!Tandem (version 2017.2.1.2),[14] and MSGF+ (version 10089)[15] against a combined version of the UniProtKB/SwissProt Yeast database (version October 2016) and the UPS 1 and 2 sequences and a combined version of the UniProtKB/SwissProt human database (version October 2016) and the UniProtKB/SwissProt *E. coli* database (version September 2018). Carbamidomethylation was set as fixed modification, and oxidation of methionine and N-terminal acetylation as variable modifications. The enzyme was set to trypsin with 2 missed cleavages allowed. Reversed decoy sequences were appended to the database. The precursor ion tolerance was set to 20 ppm (CPTAC, Ramus et al.) or 10 ppm (Shalit et al.), the fragment ion tolerance was set to 0.5 m/z (CPTAC, Ramus et al.), and 0.05 m/z (Shalit et al.) units for X! Tandem, and in MSGF+ the instrument was set to "Orbitrap". Search results were filtered at 1% FDR at the PSM level using the target-decoy approach. Search results from the different search engines were not merged but processed individually (see below). The input MGF files used in the clustering process were annotated with the results of the search using the "mgf_search_result_annotator.py" tool from the "spectra-cluster-py" Python package (version 1.0, http://www.github.com/spectra-cluster/spectra-cluster-py).

For the iPRG data, the original search results provided by the authors were used. PSMs were filtered at 1% posterior-error probability (iProphet probability ≥99%), and MGF files were annotated accordingly before the clustering process, using a custom Python script.

All MGF files from the same site and instrument (CPTAC study) or from one study were clustered using the *spectra-cluster-cli* tool (version 1.1.2, http://github.com/spectra-cluster/spectra-cluster-cli). The precursor and fragment tolerance were set to match the search engine settings. The "mz_150" peak filter was enabled. All other settings were left at their default values.

The PSMs were transferred to unidentified spectra using the "id_transferer_cli" tool from the "spectra-cluster-py" Python package (version 1.0) requiring at least 2 identified spectra in the same cluster and a minimum ratio (proportion of spectra identified as the same peptide sequence in the same spectral cluster) of ≥70%.[3,7] When processing PRIDE Archive, we originally set the minimum number of identified spectra to 3. As the data sets processed in this study are homogeneous with known data set wide FDR values for the search results, we decreased this value to 2. Protein inference was done using the "protein_annotator.py" tool from the "spectra-cluster-py" Python package (version 1.0). Only unique peptides were used for quantification (peptides mapping to a single protein).

Label-free spectral counting-based quantification was done using R (version 3.4.3) and the Bioconductor packages MSnbase[16] (version 2.4.1) and edgeR[17] (version 3.20.7), on the raw spectral counts. Result files from the "id_transferer_cli" and "protein_annotator" tool were converted into MSnSet objects using the "msnbase_adapter.r" script from the "spectra-cluster-py" package. The complete R workflow can be found in Supplementary File 1.

PD Workflow (Intensity-Based Label-Free Quantification)

The *spectra-cluster* PD node was created using C# and serves as a wrapper for the spectral clustering algorithm. It is freely available at http://ms.imp.ac.at/?goto=spectra-cluster. RAW files were loaded directly into PD (version 2.1.21). Spectra were identified using MSAmanda (node version 2.1.5.4882) and clustered using the *spectra-cluster* node. All parameters were identical to the settings explained above. Quantification was performed with IMP-apQuant[18] (version 3.1.0.20387) using the iBAQ method[19] and all available peptides. Shared peptides were not used for quantification. All other parameters were left at their default values. The results ("Protein" table) were exported to tab separated values (TSV) files, filtered for "master" proteins, normalized using the MSnbase package,[16] and the log-transformed values processed using R and the Bioconductor package limma[20] (version 3.34.5). Missing values were imputed using the 5% quantile if the protein was detected in all replicates of the other concentration. The PD and R workflows can be found in Supplementary File 1.

Software Availability

The complete workflow used to generate the spectral counting based data is freely available as a nextflow (http://www.nextflow.io) workflow at https://github.com/bigbio/nf-workflows. The ProteomeDiscoverer workflow used to generate the intensity-based quantification data and Jupyter notebooks (https://jupyter.org) containing the complete R code used to process the results are available in Supplementary File 1. All tools of the spectra-cluster toolsuite, including the Proteome Discoverer node, are available at https://spectra-cluster.github.io.

RESULTS AND DISCUSSION

We used four public data sets for the benchmarking of our algorithm: the CPTAC study 6 (subset data sets from two sites: two different ones from site 65, and an additional one from site 86), the iPRG 2015 study, and benchmark data sets published by Shalit et al. and Ramus et al. (Figure 1). In all but the iPRG data set, we used two different search engines (X!Tandem and MSGF+) to identify spectra. In the iPRG data set, we used the original identification results provided for the spectral counting pipeline. The nextflow (https://www.nextflow.io) workflows containing the complete pipeline are available at https://github.com/bigbio/nf-workflows.

We filtered all search results of the spectral counting pipeline based on a PSM FDR level of 1%. Most studies currently employ

DOI: 10.1021/acs.jproteome.8b00377
J. Proteome Res. 2019, 18, 1477−1485

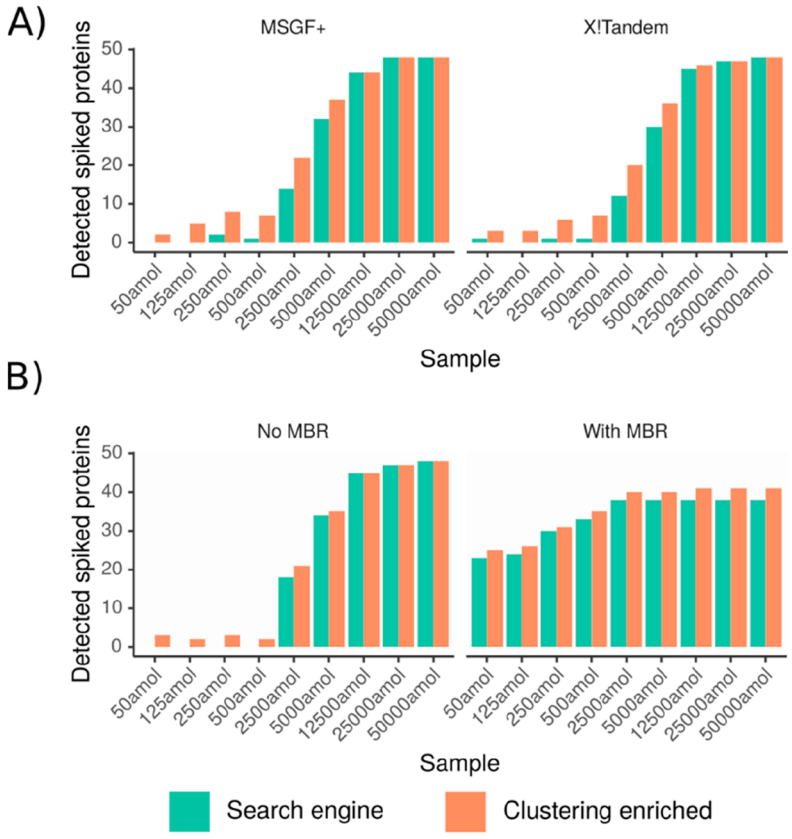


Figure 2. Number of detected spiked UPS proteins (n = 48) in the Ramus et al. data set from the (A) spectral counting pipeline (two search engines used, X!Tandem and MSGF+) and (B) the intensity-based pipeline (with and without MBR enabled).

both peptide and protein level FDRs which generally result in considerably lower PSM FDR values.[9] We deliberately used this higher PSM FDR to represent the worst-case scenario for our proposed pipeline. This higher FDR leads to a higher rate of estimated incorrectly clustered spectra and thereby negatively influences the assessment of the clustering accuracy (Figure S1). Thereby, fewer additionally inferred identifications will be assigned to unidentified spectra, as fewer spectral clusters will reach the required minimum purity (at the peptide sequence level) of 70%. Finally, a higher PSM level FDR increases the number of PSMs retrieved from the identification analysis and thereby reduces the proportion of additional PSMs inferred through spectral clustering.

Intensity-based quantification was performed using the IMP-apQuant PD node.[18] Spectral clustering was performed using the new *spectra-cluster* PD node (see below). Results were assessed with and without IMP-apQuant's MBR feature.

Spectra-Cluster PD Node

The *spectra-cluster* PD node integrates our open source spectra-cluster algorithm into one of the most commonly used analysis tools for proteomics data (Figure S2). The *spectra-cluster* node can handle both identified and unidentified spectra as input for the algorithm. If identifications are provided, these are directly incorporated into the clustering results and can be used to transfer identifications to unidentified spectra using the "Spectral Cluster Search" node. This node again creates identifications as output which can be used by any other PD node. Finally, clustering results are available through a "Clusters" table in the PD results. This table contains one entry per cluster showing basic statistics such as the number of spectra, the number of identified spectra, and the frequency of sequences within a cluster, as well as the number of spectra per sample.

As with other PD tables, the "Clusters" table can easily be exported to other formats for further analysis, using, for example, the R programming language. Since the "Spectra Cluster Search" node's PSMs are incorporated into the PD workflow, the additionally inferred identifications are visible throughout all other PD result tables. Thereby, a spectral counting based workflow can easily be created by exporting the PD's peptide/PSM table.

To our knowledge, this is the first time that a MS/MS clustering algorithm is available in a widely used proteomics software pipeline. Despite many potential applications, spectral clustering algorithms are currently not widely used. We believe that this is due to the computational expertise currently required to use them. We expect that the integration of our *spectra-cluster* algorithm into PD will make this approach readily available to the community and will considerably decrease the effort required to use spectral clustering.

Improved Detection of Low-Abundant Proteins

Throughout all test data sets, spectral clustering allowed us to quantify more low abundant proteins (Figure 2, S3). In the Ramus et al. data set, which spans the widest range of concentrations, clustering inferred PSMs allowed us to consistently identify spiked-in proteins already from the 50 amol concentration (Figure 2). As expected, this effect decreased with increasing protein concentrations in all data sets. For example, in the CPTAC study's samples A, using the spectral counting pipeline, the total number of detected UPS

DOI: 10.1021/acs.jproteome.8b00377
J. Proteome Res. 2019, 18, 1477−1485

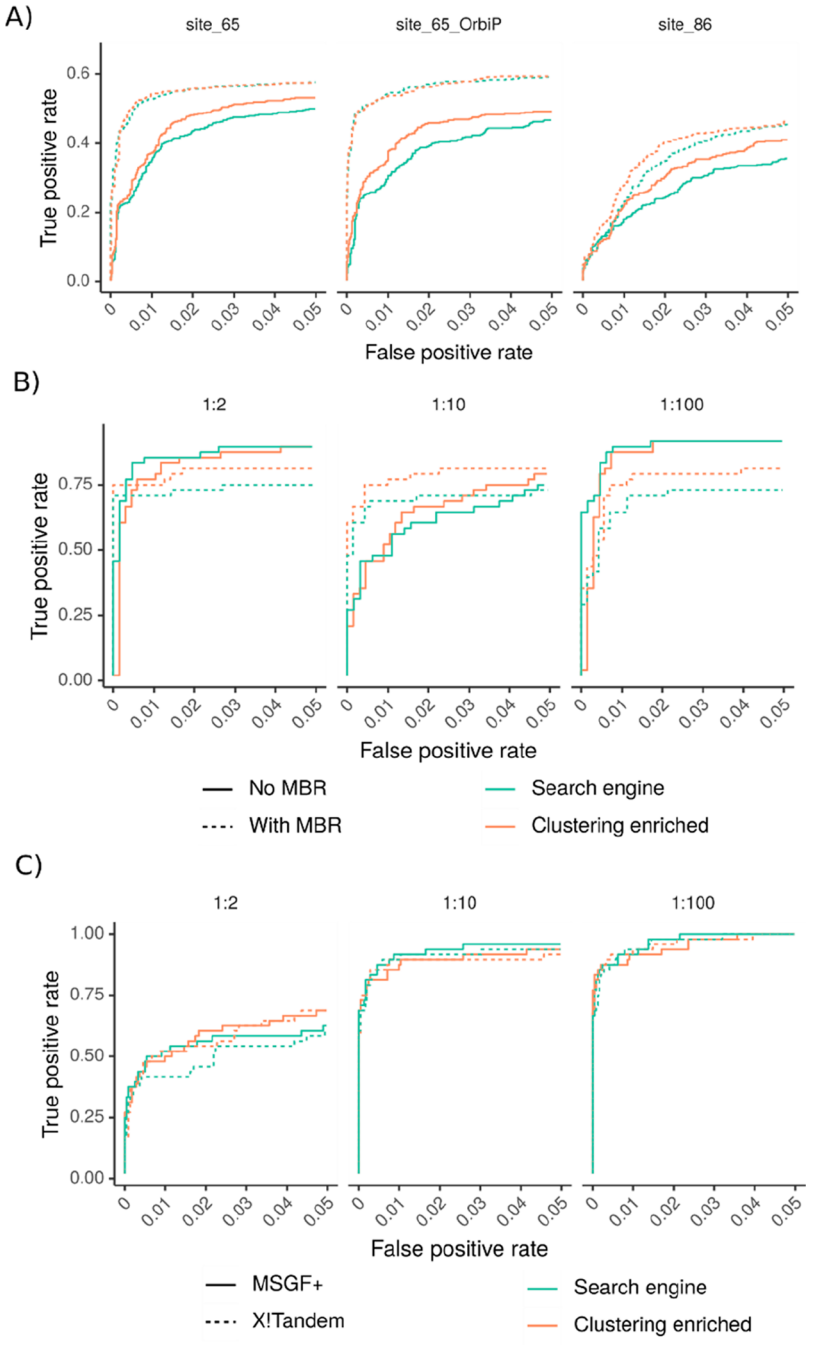


Figure 3. Results of the statistical analysis using limma for the intensity-based pipeline (A, B) and edgeR for the spectral counting based pipeline (C), as true versus false positive rates. (A) Combined result for the three CPTAC data sets using the intensity-based pipeline. (B) Result for the Ramus et al. data set from the intensity- and (C) from the spectral counting-based pipeline.

proteins over the three replicates increased from 28 to 49 (75%) for MSGF+ and from 26 to 33 (27%) for X!Tandem. For samples B, the proportion of additionally detected proteins was 17% for MSGF+ and 11% for X!Tandem (Figure S3). Finally, for samples E, which contained the highest protein concentrations, no additional proteins were detected through clustering. These results are in line with our original expectation that clustering increases the detectability of low abundant proteins. Low abundant proteins generally result in lower quality spectra. We and others have shown that clustering can be used to identify them through clustering with samples where the proteins were observed in higher quantity.

As expected, apQuant's MBR function quantified considerably more low-abundant proteins than clustering alone (Figure 2B, S3). In the CPTAC data set, for example, MBR without and with clustering quantified 93 and 96 proteins respectively, in the lowest concentration (summarized across all three sites, Figure S3). Without the MBR function only 36 and 50 proteins without and with clustering, respectively, were quantified. Again, this effect decreased with increasing concentrations. In the highest concentration samples, all approaches quantified 138 proteins (48 proteins in three sites). Nevertheless, clustering increased the number of detected proteins in three of the five CPTAC concentrations and all

DOI: 10.1021/acs.jproteome.8b00377
J. Proteome Res. 2019, 18, 1477−1485

samples of the Ramus et al. data set on-top of the MBR function (Figures 2 and S3).

apQuant is one of the first quantification algorithms able to estimate an FDR for quantified features. Once the quantified features fall below the set threshold (5% by default and in this study), they are discarded. Surprisingly, the MBR feature led to fewer quantified proteins in the highest concentration samples in both the Ramus et al. and the Shalit et al. data sets (Figures 2B and S3). In the latter data set, clustering further decreased the number of quantified proteins in this setup. This result indicates potential risks of using the MBR function in complex samples such as the Shalit et al. data set, or when larger numbers of samples are included as is the case for the Ramus et al. data set (3 replicates for 9 concentrations). Most existing quantification tools do not report feature FDR values but quantify any feature available. Therefore, the user will not notice if feature detection and quantification quality are deteriorating. apQuant's reliability assessment is able to highlight this issue. In our tests, the highest number of spiked proteins was identified in the samples containing the highest concentrations, but, interestingly, only when disabling the MBR function and using clustering. This indicates clustering without the MBR function produces more comprehensive quantification results in certain setups.

Overall, clustering enabled us to quantify more proteins in all of the three test data sets. This is consistent with previous studies that showed that spectral clustering can be used to increase the number of identified spectra.[4,7] It also highlights that low-abundant proteins result in lower-quality spectra that can still be identified using clustering. Since clustering algorithms use similar metrics as spectral library search engines, a combination of a classical search engine with a spectral library search engine might potentially lead to a similar improvement. In this respect, clustering has the advantage that it does not need to rely on a spectral library but can directly use the data set's spectra which may lead to a higher sensitivity.

Improved Detection of Regulated Proteins

We used the R Bioconductor packages edgeR[17] to assess differentially expressed proteins in the spectral counting pipeline, and limma[20] in the intensity-based pipeline (Supplementary File 1). Spiked proteins were considered as true positives and any regulated background proteins as false positives. We did not include a statistical analysis of the iPRG data set, as it only contains six spiked proteins.

Using the intensity-based pipeline, clustering improved the results in four of the five evaluated data sets. In the CPTAC data sets this was most pronounced in site_86, which overall showed the worst results (Figure 3A). In site_65 and site_65_OrbiP the results were nearly ideal when MBR was enabled and we did not observe any changes through clustering. Without MBR, clustering did improve the results in both data sets but could not reach the level of the analyses performed with MBR enabled (Figure 3A). This is expected as the MBR function can use the always present MS1 data to improve quantification results instead of the stochastic MS2 spectra.

In the Ramus et al. data set, clustering considerably improved the statistical results when using MBR and only improved the results in the 1:10 comparison without MBR (Figure 3B). This was the only data set where the statistical results were better without MBR in two comparisons. In this data set, MBR seemed to produce unreliable results most likely due to the considerably larger number of samples (see above). In these conditions, clustering improved the results considerably.

When using the spectral counting pipeline, clustering did not change the accuracy of the statistical results when combining all comparisons of the data sets (Figure S4) but did improve specific comparisons. In the CPTAC data sets, clustering improved the results when comparing the lower concentration samples (A-B, A-C, A-D, B-C) and led to worse results in the highest concentrations samples, in the site_65 and site_65_OrbiP data set (C-E, D-E, Figure S5). Here, clustering did not infer any PSMs in the "E" samples and therefore decreased the fold change. Consistently, the results improved in all comparisons where both samples equally profited from the additional PSMs.

In the Ramus et al. data set, clustering clearly improved the results of the "1:2" comparison while leading to marginally worse results in the "1:10" and "1:100" comparisons. Similar to the CPTAC data set, in these two comparisons the highest concentration sample was compared with two low concentration ones. Here, clustering only increased the number of PSMs in the low-concentration samples which led to lower fold changes. In contrast, in the "1:2" comparison both samples equally profited from the additional PSMs and the results improved. Moreover, clustering ameliorated the originally worse performance of X! Tandem, when compared to MSGF+ (Figure 3C).

In the Shalit et al. data set, both pipelines detected considerably fewer proteins than the 228 *E. coli* proteins reported in the original study[10] (Figure S6). Clustering did not change these results in both pipelines. This is most likely caused by the different methods of analyzing regulated proteins: Shalit et al. defined regulated proteins only based on the observed relative deviation from the expected fold change but did not use any statistical method to detect regulated proteins. We used the linear modeling provided by limma and edgeR, which takes the background distributions into consideration as well as apQuant's estimate of quantification reliability. Therefore, no clear conclusions can be drawn from this result.

Overall, clustering improved the statistical results in both pipelines. The cases found in the spectral counting pipeline where clustering led to slightly worse results were all caused by only one sample profiting from the additionally inferred PSMs. In these cases, protein abundances were already high enough for the search engine to identify all of the protein's corresponding mass spectra. Therefore, the spectral counting based quantitation reached its upper-limit of detection. When using clustering, these cases can be detected, as no additional PSMs can actually be inferred. This could be used to develop methods to correct for such cases and thereby increase the dynamic range of spectral counting based workflows.

The lower overall improvement of the statistical results in the spectral counting pipeline compared to the intensity-based one are counterintuitive when looking at the increased number of observed proteins through clustering. This apparent contradiction is caused by the way additional PSMs are "used". In the spectral counting pipeline these directly increase the estimated abundance of the respective protein. If a protein was originally not detected in a low-concentration sample, this protein is considered to have zero expression in the subsequent statistical analysis which leads to a large fold change. In the intensity-based pipeline, the additional PSM's precursor intensity value is used to calculate the protein's abundance based on the integrated intensity of all PSMs. Moreover, missing values for undetected proteins were imputed using the 5% quantile of all intensities (see Methods and Supplementary File 1). Therefore, the effect of having additional PSMs resulted more likely in the improvement of the accuracy of protein abundance estimates

DOI: 10.1021/acs.jproteome.8b00377
J. Proteome Res. 2019, 18, 1477−1485

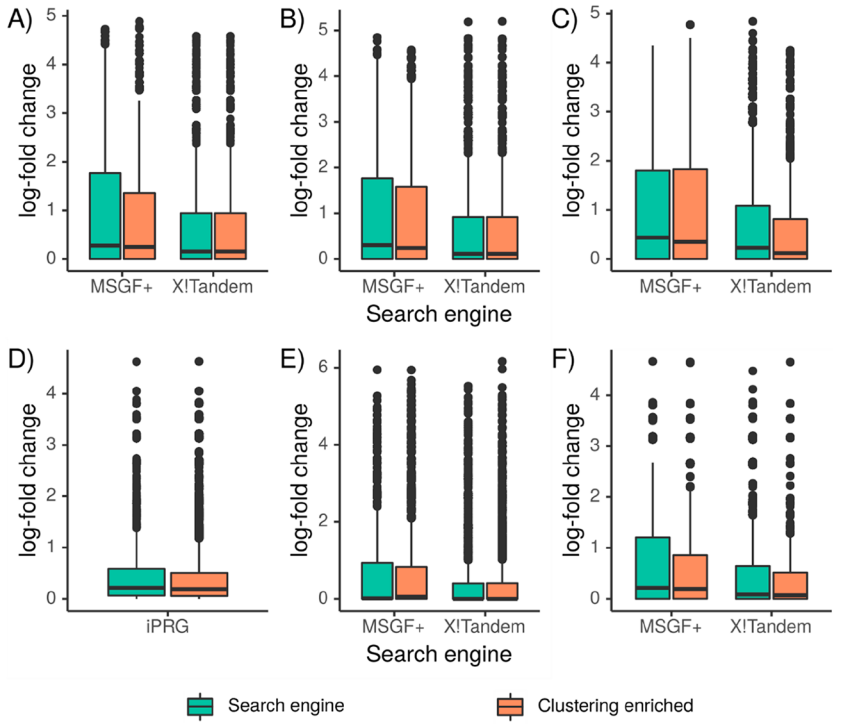


Figure 4. Logarithmic fold change of background proteins from all comparisons using the spectral counting pipeline. In all analyzed data sets, the estimated fold change of background proteins came close to 0 through the clustering of inferred identifications. Panels show the data for the CPTAC data sets site_65 (A), site_65_OrbiP (B), site_86 (C), and the iPRG (D), the Ramus et al. (E), and the Shalit et al. data sets (F).

in the intensity-based pipeline, irrespective of the protein's abundance in the respective sample.

Stable Quantification Accuracy

Any method that increases the number of PSMs is at risk of simultaneously increasing the corresponding FDR. A common method to assess clustering accuracy is to compare the clustering results with results coming from a search engine (Figure S1). Benchmark data sets for quantitative proteomics offer additional methods to assess our method's accuracy: (1) by estimating the error of the derived quantitative values; (2) by analyzing the error of quantified replicate measurements; and (3) by analyzing the estimated change of unchanged background proteins.

In all data sets and approaches used we found no significant changes in the squared error of the estimated log-fold changes of spiked proteins. Using the spectral counting pipeline, the median squared error decreased from 0.79 to 0.78 (MSGF+), from 0.87 to 0.86 (X!Tandem) in the Shalit et al. data set, from 1 to 0.5 in the iPRG data set, did not change in the Ramus et al. data set (10 MSGF+, 11 X!Tandem), and increased from 3.5 to 4 (MSGF+) and from 5.1 to 5.3 (X!Tandem) across the CPTAC data sets. Using the intensity-based pipeline, the median squared error decreased from 2.5 to 2.2 (no MBR) and remained at 0.4 (with MBR enabled) in the CPTAC data sets, decreased from 3.0 to 2.4 (no MBR) and increased from 1.4 to 1.7 (with MBR) in the iPRG data set, decreased from 1.7 to 1.6 (no MBR) and from 0.03 to 0.02 (with MBR) in the Ramus et al. data set, and did not change in the Shalit et al. data set (0.5 no MBR, 0.1 with MBR). Overall, clustering inferred PSMs led to a reduced median error in most cases and never led to a significant increase. This additionally indicates that clustering did not increase the number of incorrect identifications.

We estimated the precision of the quantitative results by using the coefficient of variation (CV) of replicate measurements. Again, we found no significant differences between both approaches. In the spectral counting pipeline clustering decreased the CV in all data sets but the Ramus et al. one, where it led to a slight increase (22% to 26%, Figure S7). Using the intensity-based pipeline, clustering never increased the CV. In the CPTAC data set, the median CV decreased from 3.4% to 3.2% (no MBR) and did not change with MBR enabled (2.1%). In the iPRG data set we did not observe any change. In the Ramus et al. data set the median CV decreased from 0.8% to 0.7% (no MBR) and did not change with MBR (0.3%). Similarly, in the Shalit et al. data set the CV decreased from 1.5% to 1.4% (without MBR) and did not change with MBR (0.3%). Clustering inferred PSMs generally reduced the CV, although this improvement was marginal. More importantly, the data showed no sign that clustering increased incorrect PSMs.

The considerably higher CVs in the spectral counting pipeline compared to the intensity of one is primarily related to the way PSMs are aggregated on the protein data. If a protein was only identified through two spectra in one replicate and with four in the other, this results in a difference of 100%. In the intensity-based pipeline protein abundance is estimated based on the aggregated spectra' precursor intensities which reduces the observed variation. These two very different approaches of estimating protein abundances are reflected in the different statistical pipelines used.

We additionally assessed the estimated fold change of the background proteins. Using the spectral counting pipeline, in all data sets but the Ramus et al. one., clustering reduced the median absolute log-fold change of the background proteins closer to the correct value of 0 (Figure 4). In the Ramus et al. data set we observed a marginal increase for MSGF+ from 0.03 to 0.07. Using the intensity-based pipeline, the median absolute log-fold change of background proteins did not change in the

DOI: 10.1021/acs.jproteome.8b00377
J. Proteome Res. 2019, 18, 1477−1485

iPRG and Shalit et al. data sets, decreased in the CPTAC data sets, and increased marginally in the Ramus et al. from 0.11 to 0.13 (with MBR enabled) and did not change without MBR (Figure S8).

Overall, clustering inferred PSMs did not have a negative impact on the accuracy or precision of the derived quantitative estimates. In our view, this is providing additional evidence that spectral clustering can infer additional correct PSMs at a stable FDR, as random incorrect identifications would also hinder the accuracy of the quantitative results.

Outlook

Since the *spectra-cluster* algorithm was originally developed to process highly heterogeneous repository sized data sets such as those available in the PRIDE Archive, we deliberately chose small data sets for this study in order to show that clustering accuracy did not decrease with smaller data sets. In previous work we already showed that *spectra-cluster* could accurately process millions of MS/MS spectra.[3,7] Therefore, this pipeline seems highly suited for large-scale (clinical) studies. In such cases, clustering could even be used prior to the identification step to reduce the search and processing times by directly identifying the resulting consensus spectra. Additionally, differential expression analysis could be performed before the identification step altogether to then target these spectra of interest with potentially computationally more expensive methods. This approach has recently been suggested in a preprint using MaRaCluster.[21] Such workflows are already fully supported by our PD node. Spectral counts can easily be exported through the respective results table and imported into R. Then, any spectral counting workflow can be used on this data.

Here we also showed that the additionally inferred PSMs can improve the accuracy of spectral counting based LFQ, as well as the detectability of low abundant proteins. This improvement is most likely also applicable to related approaches such as the Normalized Spectral Index[22] which are readily available through the MSnbase R Bioconductor package.[16]

Even though the intensity-based workflow already used apQuant's MBR functionality, spectral clustering still improved the quantification results in several cases. In this study, this was mostly linked to a reduction of apQuant's quantification reliability assessment through the MBR function. This highlights that MBR approaches are not suitable for every experimental setup. We observed that clustering did improve quantification results in all of these cases.

In addition to our *spectra-cluster* algorithm, MSCluster[4] and MaRaCluster[5] are other dedicated MS/MS clustering algorithms. Since MaRaCluster and *spectra-cluster* are similar in performance,[23] it is very likely that both MS/MS clustering algorithms will result in similar improvements.

CONCLUSIONS

We and others have shown that additional PSMs can be inferred using spectral clustering.[3,4] Nevertheless, we are unaware of any subsequent working application of these results to improve proteomics LFQ analysis workflows in practice.

Here we showed that clustering can indeed improve LFQ results and increase the detectability of low-abundant proteins without increasing the data set's FDR. Additionally, to our knowledge, this is the first time that a spectral clustering algorithm has been integrated into a commonly used proteomics software pipeline. With this integration of our *spectra-cluster* algorithm into PD and its subsequent ease-of-use, we therefore believe that clustering can be used to improve any LFQ pipeline.

ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acs.jproteome.8b00377.

Figure S1 - Estimated clustering accuracy. Figure S2 - Proteome Discoverer workflow showing the new spectra-cluster node. Figure S3 - Number of detected proteins. Figure S4 - True positive vs false positive quantified proteins from the spectral-counting pipeline. Figure S5 - True vs false positive quantified proteins from the spectral-counting pipeline in the CPTAC data sets. Figure S6 - True vs false positive quantified proteins in the Shalit et al. data set. Figure S7 - CV of the spectral-counting pipeline. Figure S8 - Fold change of background proteins estimated by the intensity-based pipeline. (PDF)

Supplementary File 1 - Proteome Discoverer workflow and Jupyter notebooks containing the R analysis code as well as the complete input data. (ZIP)

AUTHOR INFORMATION

Corresponding Author

*Johannes Griss E-mail: johannes.griss@meduniwien.ac.at, Tel: +43 1 40400 77020.

ORCID

Johannes Griss: 0000-0003-2206-9511

Karl Mechtler: 0000-0002-3392-9946

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

This work received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme under grant agreement No 788042, from the Austrian Science Fund (FWF) [grant number P 30325-B28], from the Wellcome Trust [grant numbers WT101477MA and 208391/Z/17/Z], and from EMBL core funding.

ABBREVIATIONS

CPTAC, Clinical Proteomic Technologies Assessment for Cancer; CV, Coefficient of Variation; iPRG, Proteome Informatics Research Group; LFQ, Label-free Quantification; MBR, Match-Between Runs; MS, Mass Spectrometry; PSM, Peptide Spectrum Match; PD, Proteome Discoverer; TSV, Tab Separated Values; UPS, Universal Proteomics Standard

REFERENCES

(1) Cox, J.; Hein, M. Y.; Luber, C. A.; 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.

(2) Perez-Riverol, Y.; Vizcaino, J. A.; Griss, J. Future Prospects of Spectral Clustering Approaches in Proteomics. *Proteomics* **2018**, *18* (14), e1700454.

(3) Griss, J.; Perez-Riverol, Y.; Lewis, S.; Tabb, D. L.; Dianes, J. A.; Del-Toro, N.; Rurik, M.; Walzer, M. W.; Kohlbacher, O.; Hermjakob, H.; Wang, R.; Vizcaino, J. A. Recognizing millions of consistently

DOI: 10.1021/acs.jproteome.8b00377
J. Proteome Res. 2019, 18, 1477−1485

unidentified spectra across hundreds of shotgun proteomics datasets. *Nat. Methods* **2016**, *13* (8), 651−656.
(4) Frank, A. M.; Monroe, M. E.; Shah, A. R.; Carver, J. J.; Bandeira, N.; Moore, R. J.; Anderson, G. A.; Smith, R. D.; Pevzner, P. A. Spectral archives: extending spectral libraries to analyze both identified and unidentified spectra. *Nat. Methods* **2011**, *8* (7), 587−91.
(5) The, M.; Kall, L. MaRaCluster: A Fragment Rarity Metric for Clustering Fragment Spectra in Shotgun Proteomics. *J. Proteome Res.* **2016**, *15* (3), 713−20.
(6) Wang, L.; Li, S.; Tang, H. msCRUSH: fast tandem mass spectral clustering using locality sensitive hashing. *J. Proteome Res.* **2019**, *18* (1), 147−158.
(7) Griss, J.; Foster, J. M.; Hermjakob, H.; Vizcaino, J. A. PRIDE Cluster: building a consensus of proteomics data. *Nat. Methods* **2013**, *10* (2), 95−6.
(8) Tabb, D. L.; Vega-Montoto, L.; Rudnick, P. A.; Variyath, A. M.; Ham, A. J.; Bunk, D. M.; Kilpatrick, L. E.; Billheimer, D. D.; Blackman, R. K.; Cardasis, H. L.; Carr, S. A.; Clauser, K. R.; Jaffe, J. D.; Kowalski, K. A.; Neubert, T. A.; Regnier, F. E.; Schilling, B.; Tegeler, T. J.; Wang, M.; Wang, P.; Whiteaker, J. R.; Zimmerman, L. J.; Fisher, S. J.; Gibson, B. W.; Kinsinger, C. R.; Mesri, M.; Rodriguez, H.; Stein, S. E.; Tempst, P.; Paulovich, A. G.; Liebler, D. C.; Spiegelman, C. Repeatability and reproducibility in proteomic identifications by liquid chromatography-tandem mass spectrometry. *J. Proteome Res.* **2010**, *9* (2), 761−76.
(9) Choi, M.; Eren-Dogu, Z. F.; Colangelo, C.; Cottrell, J.; Hoopmann, M. R.; Kapp, E. A.; Kim, S.; Lam, H.; Neubert, T. A.; Palmblad, M.; Phinney, B. S.; Weintraub, S. T.; MacLean, B.; Vitek, O. ABRF Proteome Informatics Research Group (iPRG) 2015 Study: Detection of Differentially Abundant Proteins in Label-Free Quantitative LC-MS/MS Experiments. *J. Proteome Res.* **2017**, *16* (2), 945−957.
(10) Shalit, T.; Elinger, D.; Savidor, A.; Gabashvili, A.; Levin, Y. MS1-based label-free proteomics using a quadrupole orbitrap mass spectrometer. *J. Proteome Res.* **2015**, *14* (4), 1979−86.
(11) Ramus, C.; Hovasse, A.; Marcellin, M.; Hesse, A. M.; 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.
(12) Perez-Riverol, Y.; Csordas, A.; Bai, J.; Bernal-Llinares, M.; Hewapathirana, S.; Kundu, D. J.; Inuganti, A.; Griss, J.; Mayer, G.; Eisenacher, M.; Perez, E.; Uszkoreit, J.; Pfeuffer, J.; Sachsenberg, T.; Yilmaz, S.; Tiwary, S.; Cox, J.; Audain, E.; Walzer, M.; Jarnuczak, A. F.; Ternent, T.; Brazma, A.; Vizcaino, J. A. The PRIDE database and related tools and resources in 2019: improving support for quantification data. *Nucleic Acids Res.* **2019**, *47* (D1), D442−D450.
(13) Adusumilli, R.; Mallick, P. Data Conversion with ProteoWizard msConvert. *Methods Mol. Biol.* **2017**, *1550*, 339−368.
(14) Craig, R.; Beavis, R. C. TANDEM: matching proteins with tandem mass spectra. *Bioinformatics* **2004**, *20* (9), 1466−7.
(15) Kim, S.; Pevzner, P. A. MS-GF+ makes progress towards a universal database search tool for proteomics. *Nat. Commun.* **2014**, *5*, 5277.
(16) Gatto, L.; Lilley, K. S. MSnbase-an R/Bioconductor package for isobaric tagged mass spectrometry data visualization, processing and quantitation. *Bioinformatics* **2012**, *28* (2), 288−9.
(17) McCarthy, D. J.; Chen, Y.; Smyth, G. K. Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation. *Nucleic Acids Res.* **2012**, *40* (10), 4288−97.
(18) Doblmann, J.; Dusberger, F.; Imre, R.; Hudecz, O.; Stanek, F.; Mechtler, K.; Durnberger, G. apQuant: Accurate Label-Free Quantification by Quality Filtering. *J. Proteome Res.* **2019**, *18* (1), 535−541.
(19) Schwanhausser, B.; Busse, D.; Li, N.; Dittmar, G.; Schuchhardt, J.; Wolf, J.; Chen, W.; Selbach, M. Global quantification of mammalian gene expression control. *Nature* **2011**, *473* (7347), 337−42.
(20) 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* (7), e47.
(21) The, M.; Kall, L. Focus on the spectra that matter by clustering of quantification data in shotgun proteomics. *bioRxiv* **2018**, 488015.
(22) Griffin, N. M.; Yu, J.; Long, F.; Oh, P.; Shore, S.; Li, Y.; Koziol, J. A.; Schnitzer, J. E. Label-free, normalized quantification of complex mass spectrometry data for proteomic analysis. *Nat. Biotechnol.* **2010**, *28* (1), 83−9.
(23) Griss, J.; Perez-Riverol, Y.; The, M.; Kall, L.; Vizcaino, J. A. Response to ″Comparison and Evaluation of Clustering Algorithms for Tandem Mass Spectra″. *J. Proteome Res.* **2018**, *17* (5), 1993−1996.

DOI: 10.1021/acs.jproteome.8b00377
J. Proteome Res. 2019, 18, 1477−1485