Accurate Label-Free Quantification by directLFQ to Compare Unlimited Numbers of Proteomes

Authors

Constantin Ammar, Julia Patricia Schessner, Sander Willems, André C. Michaelis, and Matthias Mann

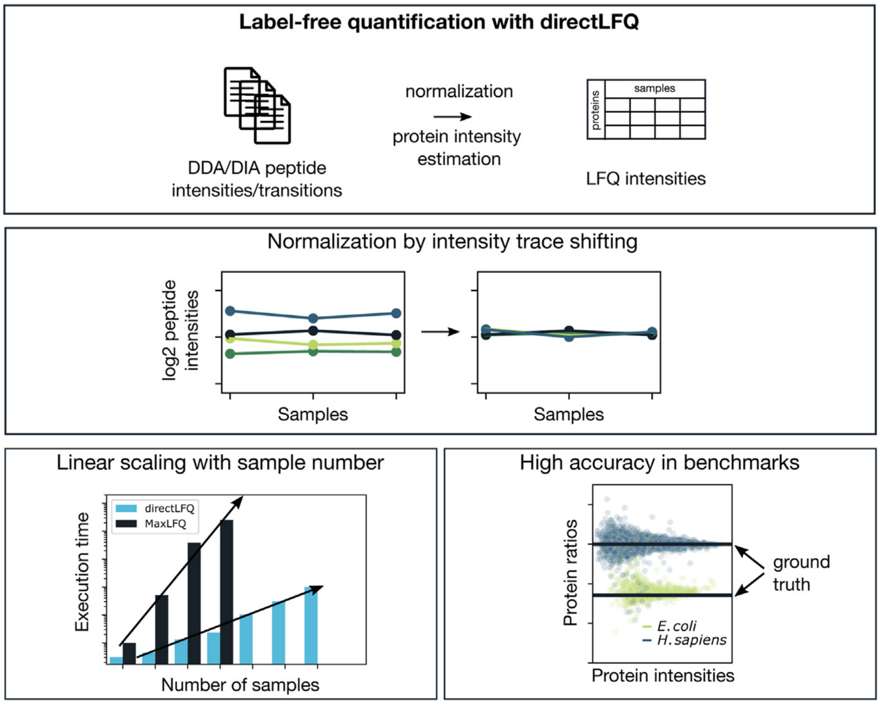
Correspondence

mmann@biochem.mpg.de

In Brief

Quantification of protein intensities is a crucial and challenging task in mass spectrometry–based proteomics. With directLFQ, we introduce an efficient algorithm that massively decreases execution time for large sample numbers while maintaining high accuracy. As it scales linearly with the number of samples, directLFQ unlocks high-quality quantification for arbitrarily large sample sizes. This is especially important for clinical proteomics as well as the rising field of single cell proteomics.

Graphical Abstract


Label-free quantification with directLFQ
DDA/DIA peptide intensities/transitions
normalization
protein intensity estimation
samples
proteins
LFQ intensities
Normalization by intensity trace shifting
log2 peptide intensities
Samples
Samples
Linear scaling with sample number
directLFQ
MaxLFQ
Execution time
Number of samples
High accuracy in benchmarks
Protein ratios
ground truth
E. coli
H. sapiens
Protein intensities

Highlights

- Novel algorithmic concept for label-free quantification.
- Linear dependence of execution time on number of samples.
- Quantifies up to 100,000 samples in less than 2 h.
- Higher accuracy and robustness than state-of-the-art MaxLFQ.

2023, Mol Cell Proteomics 22(7), 100581

https://doi.org/10.1016/j.mcpro.2023.100581

Accurate Label-Free Quantification by directLFQ to Compare Unlimited Numbers of Proteomes

Constantin Ammar, Julia Patricia Schessner, Sander Willems, André C. Michaelis, and Matthias Mann*

Recent advances in mass spectrometry–based proteomics enable the acquisition of increasingly large datasets within relatively short times, which exposes bottlenecks in the bioinformatics pipeline. Although peptide identification is already scalable, most label-free quantification (LFQ) algorithms scale quadratic or cubic with the sample numbers, which may even preclude the analysis of large-scale data. Here we introduce directLFQ, a ratio-based approach for sample normalization and the calculation of protein intensities. It estimates quantities *via* aligning samples and ion traces by shifting them on top of each other in logarithmic space. Importantly, directLFQ scales linearly with the number of samples, allowing analyses of large studies to finish in minutes instead of days or months. We quantify 10,000 proteomes in 10 min and 100,000 proteomes in less than 2 h, a 1000-fold faster than some implementations of the popular LFQ algorithm MaxLFQ. In-depth characterization of directLFQ reveals excellent normalization properties and benchmark results, comparing favorably to MaxLFQ for both data-dependent acquisition and data-independent acquisition. In addition, directLFQ provides normalized peptide intensity estimates for peptide-level comparisons. It is an important part of an overall quantitative proteomic pipeline that also needs to include high sensitive statistical analysis leading to proteoform resolution. Available as an open-source Python package and a graphical user interface with a one-click installer, it can be used in the AlphaPept ecosystem as well as downstream of most common computational proteomics pipelines.

Mass spectrometry (MS)-based proteomics is the method of choice for global analysis of the proteome (1), including applications in clinical (2, 3), single cell (4, 5) and spatial (6) proteomics. Modern MS-proteomics workflows are increasingly quantitative, meaning that the major insights derived from the experiments are gleaned from observed changes in protein abundances (7). Appropriate computational processing is therefore key for obtaining unbiased quantitative protein values from the raw ion intensities acquired by the instrument (8, 9). Conceptually, the computational processing of MS proteomics data comprises identification and quantification. In the identification step, the ion signals acquired by the MS instrument are assigned to their most probable peptides and proteins using statistical models. In the quantification step, the intensities of the ion signals are used to derive meaningful proxies for protein (or peptide) abundances.

In this study, we focus on the quantification step, which again consists of two essential parts: the normalization of systematic biases between samples and the generation of peptide and protein intensities from the underlying ion intensities. A variety of methods have been established for quantification (10–19), of which the MaxLFQ approach is one of the most widely used. It is implemented in most of the current computational proteomics pipelines (20–25). One reason for the popularity of MaxLFQ is that it addresses major pitfalls in proteomics quantification by accounting for the fact that different peptides belonging to the same protein can have very different base intensities, for example, due to differing ionization efficiencies (26). In addition, it is robust against missing values for normalization (differing sample depths) and can reliably estimate protein intensities. MaxLFQ achieves this by solving equation systems for both steps (10). However, the number of terms and equations scales quadratically with the number of samples, leading to challenges in execution time as well as high overall complexity of the approach. Although this can be alleviated by faster implementations of the algorithms for quadratic optimization (22, 27), the issue of quadratic increase remains, imposing an upper limit for feasible sample numbers.

To mitigate these issues, we have introduced directLFQ as a simple and direct method for sample normalization and protein intensity estimation. We reframe the normalization problem by using the concept of “intensity traces,” which are

From the Department of Proteomics and Signal Transduction, Max Planck Institute of Biochemistry, Martinsried, Germany
*For correspondence: Matthias Mann, mmann@biochem.mpg.de.
Present address for Sander Willems: Research and Development, Bruker Belgium nv., Kontich, Belgium.

https://doi.org/10.1016/j.mcpro.2023.100581

aligned by a single scaling factor per intensity trace. By reframing the problem in such a way, we reduce the quadratic to linear scaling (*i.e.*, ten times as many samples only takes ten times longer), for any number of samples. Comparing directLFQ with MaxLFQ for several quantitative benchmarking sets, we show an overall better performance of directLFQ, for both data-dependent acquisition (DDA) and data-independent acquisition (DIA) datasets. In a complex spatial proteomics dataset, we observe that several aspects of the biology are better resolved with directLFQ.

The code of directLFQ is openly available on GitHub (https://github.com/MannLabs/directlfq), with easy access for all types of users (graphical user interface [GUI] for end users, as well as a command line interface and a Python application programming interface). Supported input formats include AlphaPept (22), MaxQuant (16), Spectronaut (23), DIA-NN (24), and IonQuant/FragPipe (25). The underlying algorithm allows for easy adaptation to other software pipelines.

EXPERIMENTAL PROCEDURES

directLFQ Analysis of MaxQuant Files

As examples of DDA, MaxQuant result files were (re)processed in this study by using the evidence.txt output file as an input to directLFQ. The proteinGroups.txt file was used as an additional input for protein group mapping.

Dynamic Organellar Maps Analysis

For analysis of the dynamic organellar maps (DOM) data, the original MaxQuant output files from the study of Schessner *et al.* (28) acquired on an Orbitrap Exploris 480 mass spectrometer were used. directLFQ was called on the MaxQuant files as described above. Subsequently, in order to fulfill the formatting requirements of the DOM-QC analysis tool, an adapted directLFQ file was created, where columns from the proteinGroups.txt file were added to the directLFQ file. The adapted directLFQ file as well as the original proteinGroups.txt file were both uploaded to the DOM-QC tool (https://domqc.bornerlab.org/QCtool) and a .yaml file containing the aligned comparison results was downloaded. Plots were created on a local machine based on the DOM-QC code available on GitHub (https://github.com/JuliaS92/SpatialProteomicsQC). The two-sample *t* test was carried out using the “scipy.stats.ttest_ind” function of the scipy package v. 1.9.3 with default parameters.

DDA Mixed Species Benchmark

For the DDA mixed species benchmark described below, the MaxQuant output files of the “HeLa–*Escherichia coli*” dataset acquired by our group (Meier *et al.* (29)) on a Q Exactive HF mass spectrometer were downloaded from the respective PRIDE (30) repository PXD006109. directLFQ was called on the MaxQuant files as described above. For benchmarking, a subset of six files was used that had been acquired in standard DDA mode (not in BoxCar mode). Three files contained six times the amount of *E. coli* as compared with the other three. For analysis, the median log2 transformed LFQ intensity of the high *E. coli* set was obtained and subtracted from the median log2 transformed LFQ intensity of the low *E. coli* set. This resulted in one log2 fold change per protein. For analysis, these fold changes were plotted as violin plots or scattered against the mean of the log2 transformed LFQ intensities over all samples.

Consistency Analysis of 200 HeLa Files

To assess the performance of directLFQ on larger datasets, the MaxQuant output files of the “200 HeLa” dataset by Bian *et al.* (31) acquired on a Q Exactive HF-X mass spectrometer was downloaded from the corresponding PRIDE repository PXD015087. directLFQ was called on the MaxQuant files as described above. The directLFQ output file and the proteinGroups.txt file were used for further analysis. The coefficient of variation (CV) was calculated for every protein over the 200 HeLa files, and the resulting distributions of CVs were compared.

DIA Mixed Species Benchmark

For the DIA mixed species benchmark, six.raw files corresponding to the “large-FC” dataset by Huang *et al.* (32) acquired on a Q Exactive HF mass spectrometer were downloaded from the PRIDE repository PXD016647. The raw files were analyzed using Spectronaut 15 in directDIA mode based on the UniProt databases UP000000625, UP000005640, UP000002311, and UP000001940.

The Spectronaut report was exported in the format specified in the spectronaut_tableconfig_fragion.rs file available on the directLFQ GitHub repository and accessible *via* the directLFQ GUI. directLFQ was called on the exported Spectronaut results file, and the protein intensities were estimated based on a combination of the MS1 isotope intensities as well as the fragment ion intensities (*i.e.*, each fragment ion and MS1 isotope corresponded to one independent intensity trace). The iq package was called on the same report file using the “iq::process_long_format” command.

The raw files mapped to two conditions S1 and S2, with three samples in S1 and three in S2. *Saccharomyces cerevisiae* and *Caenorhabditis elegans* proteins had different abundances in S1 and S2 (ratios 2 and 0.77, respectively), while *Homo sapiens* remained constant. As described above for the DDA data, the median log2 transformed intensity of each protein was obtained and these intensities were subtracted between S1 and S2. The intensity estimate for each protein was the mean log2 transformed intensity over all conditions.

Consistency Analysis of Clinical DIA Study

The DIA-NN results file of the study of Demichev *et al.* (33) (acquired on a TripleTOF 6600 mass spectrometer) was downloaded from the PRIDE repository PXD029009. directLFQ was called on the output file, using the “MS1.Area” as well as the “Fragment.Quant.Raw” columns for creating ion intensities. Each fragment ion as well as the MS1.Area corresponded to one independent intensity trace. The DIA-NN implemented MaxLFQ results were obtained *via* the “Genes.MaxLFQ” column. The iq package was called on the appropriately reformatted report using the “iq::process_long_format” command.

The dataset contained several types of quality control (QC) samples. As a quality measure the CV for each type of QC sample was obtained for every protein. The distributions of all the CVs together were then used for comparing the different LFQ approaches.

Normalization Analysis

To test normalization on a challenging dataset, the MaxQuant result files of the tissue dataset by Wang *et al.* (34) acquired on an Orbitrap Fusion Lumos mass spectrometer were downloaded from the PRIDE repository PXD010154. Precursor (sequence and charge) intensities were extracted from the evidence.txt file. directLFQ normalization was performed by calling the directLFQ normalization class on the precursor table. Median normalization was performed using R code for normalization provided together with the iq package (https://cran.r-project.org/web/packages/iq/vignettes/iq.html).

The dataset consisted of several different tissue measurements, from which “lung” was chosen as the reference tissue. The precursor intensities were log2 transformed, and the precursor intensity of the lung tissue was subtracted from the precursor intensities of the other conditions. This resulted in a distribution of ratios (one ratio per precursor) relative to lung for each tissue. The boxplots in the main text below contain the set of precursors that do not have any missing values, while normalization was performed on the complete dataset. We chose the precursors with no missing values for visualization because they are more likely to reflect the true abundance changes between the tissues.

Execution Time Analysis

Execution time comparison was performed on a medium-strength computer cluster (128 GB RAM, 64 logical processors @2.4 GHz). Execution times of up to 10,000 samples on this cluster were comparable with execution times on a state-of-the-art MacBook (MacBook Pro 16-inch, 2021, M1 Max, 64 GB RAM). The MaxLFQ reference runs were performed in the scope of an interactomics study (35) on a comparable cluster (512 GB RAM, 40 logical processors @2.2 GHz). directLFQ runs on CPUs with multiprocessing enabled per default and the option to manually set the number of cores to use.

Different sample sizes were simulated by using a template dataset that contained N samples. To simulate M > N samples, we duplicated the N samples as often as was necessary to reach M. In the case of M < N, we left out as many samples as necessary, by taking the first M columns in the template.

The template dataset for DDA data was based on the interactomics study, and the template dataset for DIA data was based on an example dataset provided together with the iq package (https://github.com/tvpham/iq/releases/download/v1.1/Bruderer15-DIA-longformat-compact.txt.gz).

We called directLFQ on the templates using the "lfq_benchmark.LFQTimer" class. For iq, we used the system.time command on the "iq::fast_preprocess" and the "iq::fast_MaxLFQ" commands. The data were adjusted to adhere to the format necessary for the "iq::fast_preprocess" command.

For the 100,000 sample datasets, the memory limit of 128 GB was surpassed, which is why we decreased the number of proteins in the template file by a factor of 4 and multiplied the resulting execution time by 4. As the proteins are processed independently of each other this should be a realistic estimation of the true execution time.

RESULTS

Reframing the Normalization Problem with Intensity traces

The underlying idea of the directLFQ approach is to frame the set of samples or ions to be normalized as a set of intensity traces. In the case of sample normalization, such a trace consists of all the ion intensities measured in a sample (Fig. 1*A*). Each point in the trace has the coordinates: (ion identifier, log2(intensity) of ion). For example, the trace representing sample 1 might be precursors VTTHPLAK_2+ with log2(intensity) of 23, VTVAGLAGK_3+ with log2(intensity) of 25, and so on. In sample 2, these same peptides will have differing intensities, resulting in a slightly different trace. There are as many traces as there are samples in the dataset.

In the case of protein intensity estimation, each trace consists of specific precursor ions (DDA) or transitions and precursor ions (DIA) and each trace contains all the intensities of these ions over the different samples (Fig. 1*B*). For example, the protein containing the precursors VTTHPLAK_2+, VTTHPLAK_3+, DGLILTSR_2+, and VTVAGLAGK _3+ would have four ion intensity traces with each point in a trace defined by the coordinates (sample number, log2(intensity)).

Note that the concept of intensity traces requires log transformation, because the shape of the intensity trace is then determined by the relative changes between samples independent of base intensity. For simplicity, we denote the *x*-axis components of the intensity traces as *features*, which could be either samples, precursor ions, or transitions, depending on the context.

Shifting Intensity traces to Compare Relative Changes

For the directLFQ approach, we frame sample normalization and protein intensity estimation as problems of intensity traces that need to be shifted (Fig. 1*C*). We do not alter the shape of each intensity trace but instead make the shapes of the intensity traces comparable with each other. This is achieved by adding one scaling factor to each intensity trace (corresponding to multiplication before the log transformation). The scaling factors are chosen to minimize the distance of the intensity traces from each other (see section below). This way the shapes of the intensity traces are preserved, but the systematic shifts between the traces are corrected for. In the case of "between sample normalization" this means that systematic intensity shifts between samples are corrected without changing the relative intensities of ions within each sample. In the case of "protein intensity estimation" this means that systematic biases between ions, due to differing ionization efficiencies of precursors in case of protein intensity estimation, are corrected for, again preserving the relative abundance changes of the ions between samples.

A Systematic Approach to Shifting Intensity traces

As mentioned, the core concept of directLFQ is to shift intensity traces on top of each other. A wide variety of methods can implement this shifting. Most simply, one original or an average/median trace could be selected and all traces could be shifted toward that trace. Alternatively, one could treat this as a minimization problem, *e.g.*, using a quadratic solver. Based on our previous work with the MS-EmpiRe (36) package, we here chose to use pair-wise comparisons for shifting using an adapted single linkage approach. This means that we compare all pairs of traces with each other as follows: each of the n traces is considered a vector of log2(intensities) with as many elements as features (Fig. 1*C*, upper right). In this vector, missing values are encoded as NaN (Not a Number). When subtracting both vectors, this results in a distribution of fold changes. From this distribution we then extract the median and the variance. The median estimates any systematic shift between the intensity traces, whereas the statistical variance reflects the overall divergence of the traces. We shift the whole set of traces using an iterative procedure as shown at the bottom of Figure 1*C*: given n traces, we first collect two $n \cdot n$ (half-) matrices, one containing the variances of all pairs of intensity traces and one containing the medians. From the

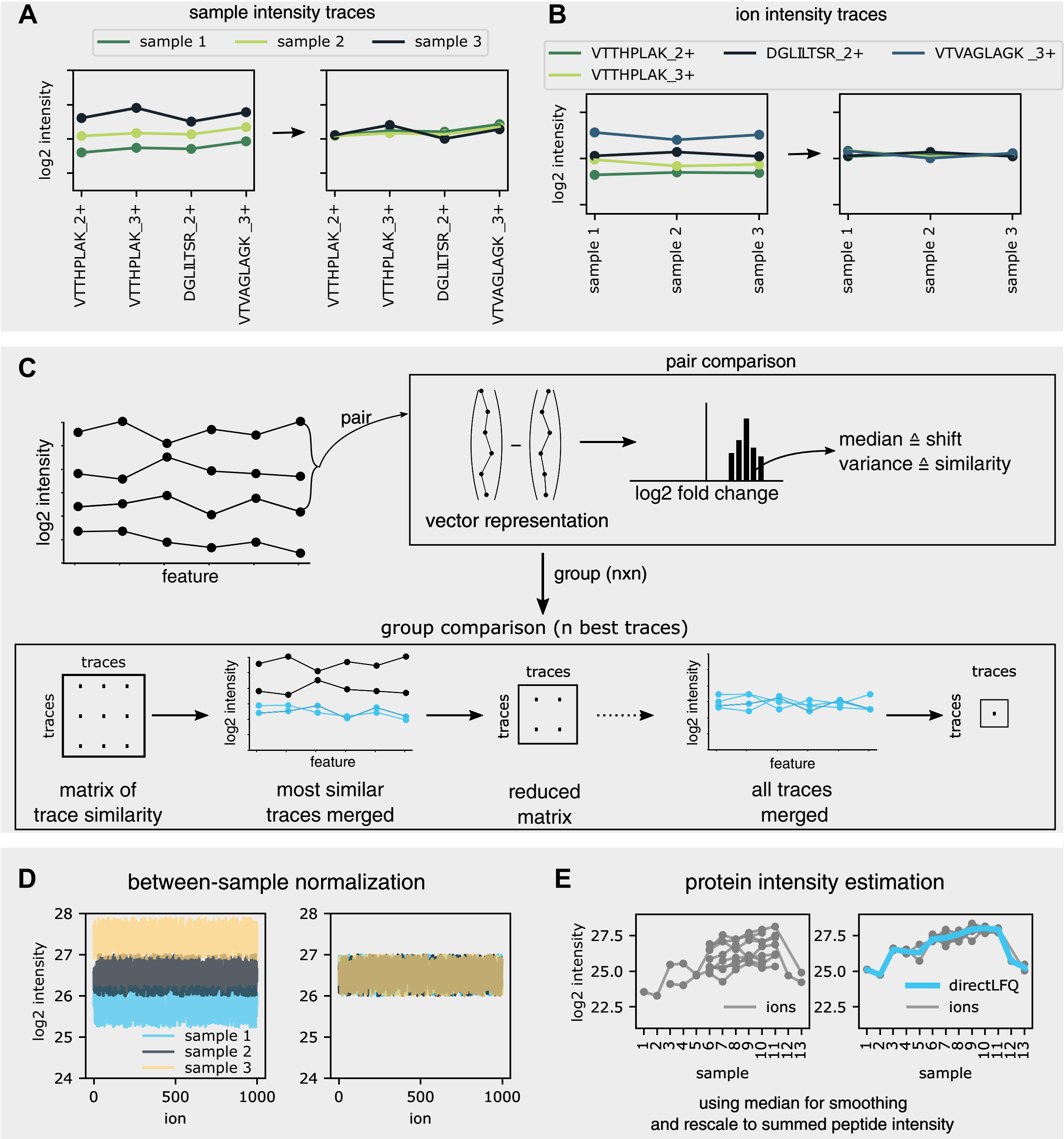


FIG. 1. **The directLFQ approach.** Objects to be normalized are intensity traces that can be shifted. *A*, between-sample normalization, where each trace represents a sample and each element of the trace is a peptide's log2 intensity. Traces are shifted on top of each other (*blue*) as described below. *B*, protein intensity estimation, where each trace belongs to a peptide and each element of the trace is a sample's log2 intensity. *C*, the shifting process. Traces are compared in a pairwise fashion by subtracting the intensities and extracting the median and variance of the resulting difference distribution (*top*). The most similar samples are shifted (indicated in *blue*) and a merged sample is created. The process is repeated on a now smaller similarity matrix until all traces are shifted (*bottom*). A more realistic example for between-sample normalization is given in (*D*) and for subsequent protein intensity estimation in (*E*).

variance matrix, we extract the pair of intensity traces that is most similar, *i.e.*, the smallest element in the matrix. One of the intensity traces in the pair is rescaled to the other by adding the corresponding median shift from the median half matrix. After scaling, the pair of traces is combined, creating a new and more stable “averaged trace” that replaces the pair.

After also recalculating the affected elements of the median and the variance matrices, the procedure is repeated on the resulting $(n-1)\cdot(n-1)$ matrices until all n intensity traces are merged. Each time an averaged trace is shifted, the corresponding scaling factors need to be propagated to all samples underlying the averaged trace. To this end, the scaling factors are tracked throughout the procedure and then used to shift the original intensity traces on top of each other.

With this approach, the most similar samples are merged first, decreasing the possible error in each shift. Creating the average intensity trace in this way has two conceptual benefits.

First, it should be more stable than the single-intensity traces, because it is an average of multiple traces. Second, creating the average trace mitigates the missing value problem. If an intensity at a certain position is missing in one vector, but present in another, the averaged vector will automatically be filled with the intensity that is not missing.

As this comparison scales quadratically with n, we define upper limits n_{max} ($n_{max} = 50$ for sample normalization and $n_{max} = 10$ for protein intensity estimation). If there are more than n_{max} intensity traces, we construct an average from the n_{max} intensity traces with the lowest numbers of missing values and shift all the remaining intensity traces toward this average. The reasoning behind this is that an average constructed from n_{max} traces should be sufficiently stable and complete to enable precise sample shifting. We tested this assumption for a case of sample normalization, where the number of samples (1600) is much larger than n_{max}. This showed that the variance between samples did not depend on n_{max}, from 5 to 200, indicating that the most complete traces picked first by the algorithms are already an excellent basis for robust normalization (supplemental Fig. S1). After having created the average, the further shifting steps are trivial and computationally inexpensive. We name the combination of the ion trace concept and the trace shifting algorithm "directLFQ," because it addresses possible quantification biases most directly.

Between-sample Normalization

With the directLFQ algorithm in hand, we perform between-sample normalization as the first step of the quantification workflow. Here each sample is rescaled as described above and visualized in Figure 1*D*, adjusting for biases such as due to differences in sample loading or different performance of the mass spectrometer. The underlying assumption is that the majority of proteins are not regulated between samples, because we use the median between intensity traces as a scaling factor. This is a common assumption based on biological observations of proteome regulation and implicit in many normalization algorithms, including MaxLFQ (10).

In cases where the majority of proteins are regulated, one could use the mode instead of the median as the metric. The mode is the most often observed change between the samples. However, this is more challenging to estimate in noisy data and is therefore often less stable than the median. The most common biological reason that our assumption of an unchanging majority of proteins does not hold would be that a particular group of "uninteresting" proteins such as contaminants or extracellular matrix, for instance, are present in a subset of samples. In such a case these could be excluded from the intensity traces in directLFQ by providing a subset of "housekeeping proteins" (37, 38) to perform normalization on. directLFQ offers the option to pass a set of such proteins, and normalization will be performed using this subset.

Protein Intensity Estimation

After sample normalization, directLFQ estimates the most likely protein profile as illustrated in Figure 1*E*. After the ion intensity traces are shifted on top of each other, the median intensity of each sample is an estimate for the relative protein intensity. The protein intensity profile is then transformed back from log2 transformed intensities into linear space and multiplied by a single factor representing the overall protein abundance. This is the sum of all linear ion intensities over all samples for the given protein divided by the sum of all linear protein intensities over all samples. In this way, the overall peptide intensity is retained, as in the MaxLFQ algorithm (10).

Handling DDA, DIA, and Other Types of Acquisition Data

We have tested and benchmarked directLFQ on DDA as well as DIA data, with DDA having quantitative data only on the MS1 level and DIA at the MS1 and MS2 levels. For DDA data processing, we build ion intensity traces based on the MS1 intensities of the charged precursors. For DIA data, we build intensity traces based on the MS1 intensities of the charged precursors as well as on the MS2 level fragment ion intensities. Thus DIA multiplies the number of data points available for quantification, which we find to stabilize the protein intensity estimation (supplemental Fig. S2). The directLFQ algorithm can also readily be applied to different types of quantitative proteomics, including isotope label–based methods such as tandem mass tags (39) (TMT). For TMT experiments contained in a single plex, the algorithm can be applied as is. In case there are more conditions than TMT channels, the corresponding experiments should be "channel normalized" before using directLFQ.

Timing directLFQ on up to 100,000 Samples

As described above, the directLFQ algorithm scales linearly with the number of samples, which should in principle allow quantification of arbitrarily large numbers of samples in a reasonable time. To test this in a controlled manner and in the absence of extremely large datasets, we used real datasets as templates and simulated datasets with increasing numbers of samples by replication (Experimental Methods). For any of the methods that we compared, we used the elapsed execution times for normalization and protein intensity estimation for benchmarking, without data loading and data output.

Figure 2*A* depicts the timing results for the very large dataset of our recent yeast interactome project (35) as processed by MaxQuant. The option "fastLFQ" (10) was enabled in processing with MaxLFQ. Its execution times increased quickly and reached 2 weeks of processing time at around 2000 samples after which we broke off the computation. In contrast, directLFQ took around 2 min for the same number of samples. To keep execution times reasonable, we omitted larger samples with MaxLFQ and further scaled up the number of samples for directLFQ to 10,000 and 100,000 by replication, which took 10 min and 100 min, respectively (Experimental Methods).

For DIA data, we instead compared against the fast C++-based implementation of the popular R package iq, as recommended for processing DIA data (Fig. 2*B*). Owing to its very fast implementation of the quadratic LFQ algorithm, it took only 2 s on its small 10-sample dataset, compared with 30 s for directLFQ. While this difference is inconsequential in routine proteomics practice, we do observe the expected nonlinear increase in execution time as a function of the number of samples and 10,000 samples already required a processing time of 2.5 days, while directLFQ takes around 16 min. We further scaled directLFQ up to 100,000 samples, which took only around 160 min. Note that directLFQ is only parallelized for the CPU, whereas implementation on GPUs would further drastically shrink processing time.

The increased processing time of directLFQ in DIA compared with DDA is due to the fact that we have one intensity trace for each fragment ion, increasing the total number of traces to be processed.

Applying directLFQ to Benchmarking Datasets

Next, we benchmarked directLFQ on several public datasets that are meant to directly assess quantification performance. We first applied directLFQ to a mixed-species (*H. sapiens* and *E. coli*) dataset acquired in DDA mode by our group (29). A total of six samples were measured, containing identical amounts of *H. sapiens* cell lysate, while three of them had 6-fold more *E. coli* than the others. Comparing the two groups with three samples each, a ratio around one would be expected for the *H. sapiens* proteins and a ratio of six for the *E. coli* proteins. The better the quantification is, the closer the proteins should be to the expected ratios. The directLFQ-derived protein intensities align well around the expected ratios, with substantially reduced outliers and lower standard deviation (0.57 instead of 0.86 for *E. coli*) as compared with MaxLFQ. In addition, the *H. sapiens* proteins are centered on the expected ratio in directLFQ, while we see systematic deviation in MaxLFQ (Fig. 3*A*). The number of proteins was near identical in both methods (supplemental Fig. S3*A*), and the outliers were indeed closer to the expected value for directLFQ compared with MaxLFQ (supplemental Fig. S3*B*).

To test the performance of directLFQ on larger experiments, we applied it to a published 200-sample HeLa technical dataset (34) (Fig. 3*B*). As the samples should be identical and each protein should therefore not significantly change between samples, we used the CV of each protein as a quality measure. Our results show CV distributions of directLFQ and MaxLFQ that are nearly identical.

To test directLFQ on DIA data, we applied it to a three-species dataset that was acquired by Huang *et al.* (32) that we processed with Spectronaut (23) (*S. cerevisiae*, *H. sapiens*, and *C. elegans*) (Experimental Methods). It consists of six samples, split into two conditions, with expected ratios 0.77, 1, and 2 (yeast, human, and *C. elegans*). We compared directLFQ against the above-mentioned iq package and the MaxLFQ implementation of Spectronaut. Both the expected ratios and the spread around them were better for directLFQ. Standard deviations for *S. cerevisiae* were 0.21, 0.26, and 0.31

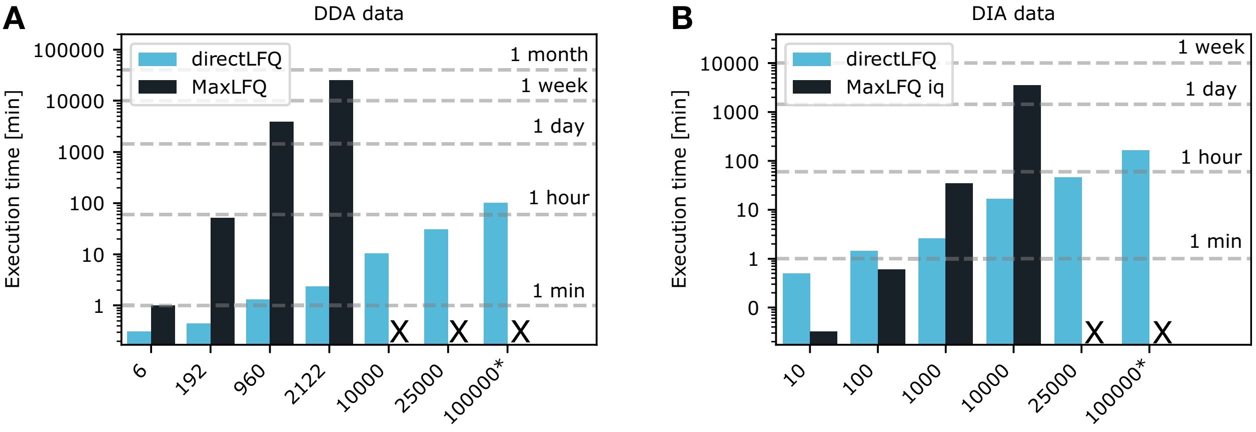


FIG. 2. **Processing times for different methods and sample sizes.** directLFQ scales (sub-) linearly for (*A*) data-dependent acquisition (DDA) data and (*B*) data-independent acquisition (DIA) data, resulting in more than 1000-fold faster execution times than MaxLFQ. X in the plot marks instances with prohibitive calculation times. *Extrapolated times, as the templates for the 100,000 sample set had to be shortened to adhere to 128 GB memory. See Experimental procedures and discussion section for details.

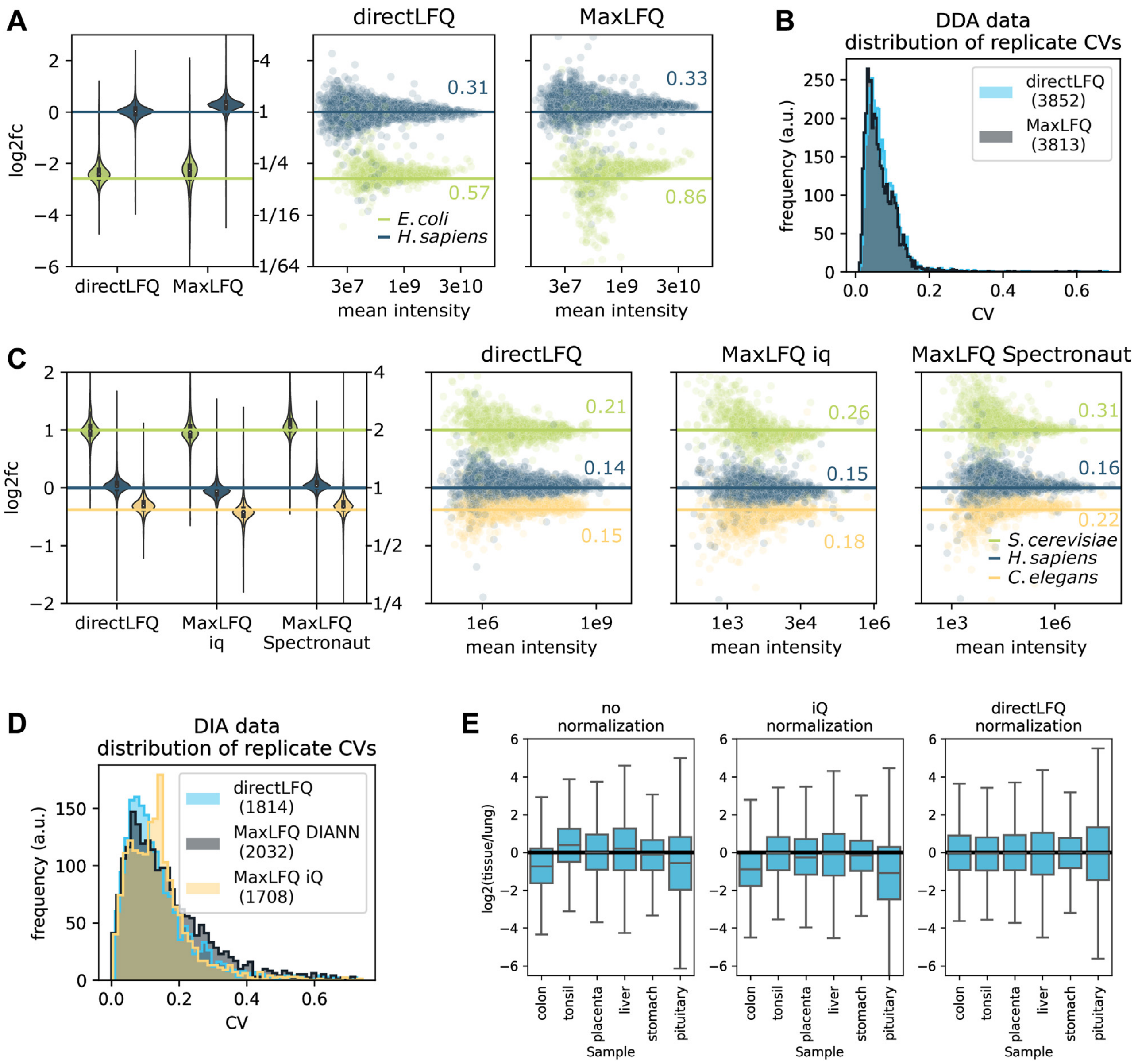


FIG. 3. **Applying directLFQ to different benchmarking datasets.** *A*, mixed-species data-dependent acquisition (DDA) dataset processed with directLFQ and MaxLFQ. *E. coli* proteins should align along a log2 ratio of –2.59 (*blue line*), and *H. sapiens* proteins should align along a log2 ratio of 0. Median values are indicated by white dots in the violin plots, standard deviations are indicated in the respective color. *B*, distribution of coefficient of variation (CV) values on a DDA dataset with 200 replicate HeLa samples, with very similar results for directLFQ and MaxLFQ. *C*, mixed-species data-independent acquisition (DIA) dataset processed with directLFQ and two MaxLFQ implementations (iq and Spectronaut). Expected log2 ratios for *S. cerevisiae*, *H. sapiens*, and *C. elegans* proteins are –0.38, 0, and 1, respectively. *D*, distribution of CV values between technical repeat samples from a ~900-sample clinical DIA dataset, processed with directLFQ and two MaxLFQ implementations (iq and DIA-NN) with comparable results for all approaches. *E*, testing directLFQ precursor normalization on a challenging tissue dataset and comparing against standard median normalization. After normalization, all boxes should be aligned around 0, which is the case for directLFQ but not for the median normalization approach.

for directLFQ, iq, and Spectronaut, respectively. iq showed a systematic offset for the *H. sapiens* proteins, which is not visible for directLFQ and Spectronaut (Fig. 3*C*). The number of proteins was again near identical (supplemental Fig. S3*C*).

We next tested performance on a clinical DIA dataset consisting of almost 1000 Covid 19 plasma proteomes (33), which had been processed with the software DIA-NN (24). This included many QC samples, which ideally should show no variability between runs, allowing us to use the CVs on the QC samples as a quality measure of quantification. Comparison of directLFQ against iq and DIA-NN revealed that the distributions of CVs are in a similar range, with iq showing an anomalous peak at a CV of 0.18 and directLFQ and DIA-NN being comparable (Fig. 3*D*).

Lastly, we tested the sample normalization algorithm of directLFQ on a tissue dataset. We chose tissues because they often have strong differences in their proteome composition and therefore constitute a challenging normalization benchmark. As a first step, we used a lung proteome as a reference and calculated the ratio to lung for each precursor in each tissue. This results in a set of ratios for each of the tissues, with systematic shifts between them (Fig. 3*E*). The objective of a normalization function is then to assign constant scaling factors to each tissue, such that they are optimally aligned. Clearly, a simple median normalization, as, for example, implemented in the iq package, does not suffice to align the datasets (Fig. 3*E*, middle), while directLFQ normalization results in well-centered distributions (Fig. 3*E*, right). This demonstrates that the directLFQ normalization strategy is effective in correcting systematic biases between samples. As noted above, such normalization is only recommended if less than half of the quantified proteins are substantially regulated. If this is not the case directLFQ provides the option to specify a protein subset to normalize on.

Applying directLFQ to Organellar Maps Data

To test directLFQ in a sophisticated cell biological situation involving proteomic separation at different time points, we analyzed a DOM dataset from Schessner *et al.* (28). In this dataset, cells were mechanically lysed and the resulting cellular compartments were separated using differential centrifugation (28). Different *fractions* of the separated sample were measured with DDA. In these organellar maps, proteins are most abundant in the fractions that correspond to their localization (for instance, Golgi apparatus). Furthermore, proteins belonging to the same protein complex or organelle are expected to have similar intensity profiles. This DOM dataset is a good benchmark for the performance of the directLFQ algorithm because this is a complex biological dataset with many distinct outcomes. In addition, the paper provides the DOM-QC benchmarking tool that is meant to assess quantification performance on this type of data, which we use in the evaluations shown.

Principal component analysis by DOM-QC of the directLFQ processed data visibly separates different organellar parts of the cell (Fig. 4*A*). Comparing the directLFQ results to the MaxLFQ results originally used in the publication suggests a more consistent clustering of the directLFQ data. This is especially visible for the endosome, peroxisome, or Golgi proteins. The overall spread of the MaxLFQ data is higher, which could either have biological or algorithmic reasons. However, investigating the proteins at the extremes of the principal component analysis indicates that directLFQ accurately reflects the changes visible at the precursor level, while MaxLFQ appears to overestimate protein ratios (supplemental Fig. S4).

To further explore the differences in protein cluster consistencies, we used the DOM-QC tool to quantify the consistency of protein profiles for proteins belonging to the same complex (Fig. 4*B*). On the default list of complexes provided by DOM-QC, the overall distance within clusters of directLFQ is consistently better (lower values) with few exceptions.

The protein intensity profiles of the minichromosome maintenance (MCM) complex had the best results in directLFQ compared with MaxLFQ and those of the proteasome complex were best in MaxLFQ (Fig. 4*C* and indicated by arrows in Fig. 4*B*). In the MCM complex, where directLFQ compared favorably, we see that the traces of directLFQ are more tightly aligned, with almost identical profiles. For the Proteasome, where MaxLFQ compared favorably, we see that directLFQ has a deviating trace. Zooming in further, we examined the most deviating protein MCM7 of the MCM complex in the MaxLFQ data. This indicated that the algorithm might overestimate the ratios for this protein, as for example visible for the 1K and Cyt fractions of the MCM7 trace, which was not the case for directLFQ (Fig. 4*D*). Conversely, to investigate the cause of the discrepancy for the proteasome, we inspected all aligned precursor intensity traces that underlie the protein intensity estimation, which is only possible at the protein level in MaxLFQ. Note that the shapes of each intensity trace are untouched, therefore accurately reflecting the underlying data. This revealed that the protein intensity estimation is indeed consistent with the precursor data for both MCM7 and PSMB2 in directLFQ. For PSMB2 there are several datapoints supporting the variation in the shape of the protein intensity profile in the 6K fraction. Thus, despite the higher variance, the data clearly validate PSMB2 as a member of the proteasome complex, while at the same time indicating underlying technical or even biological reasons such as peptide modifications for the deviating shape.

DISCUSSION

Here, we have introduced a simple yet effective algorithm for normalization and protein intensity estimation for DDA as well as DIA proteomics data. Its central concept is the shifting of peptide intensity traces and sample intensity traces on top of each other. In computer science terms, it is of linear order O(n) where n can be the number of samples or the proteins. In practice this allows the quantification of extremely large sample sizes of hundreds of thousands. As with any other algorithm, there are hardware requirements that have to be fulfilled and, in particular, directLFQ is currently memory limited, with a rough estimate of around 30 GB of memory necessary for 10,000 samples. While not a practical problem now, in the future this could be alleviated by using standard out-of-memory computing approaches. The underlying code is openly available on GitHub under the Apache License facilitating improvements and contributions from the community. The package is easily accessible *via* Python Package

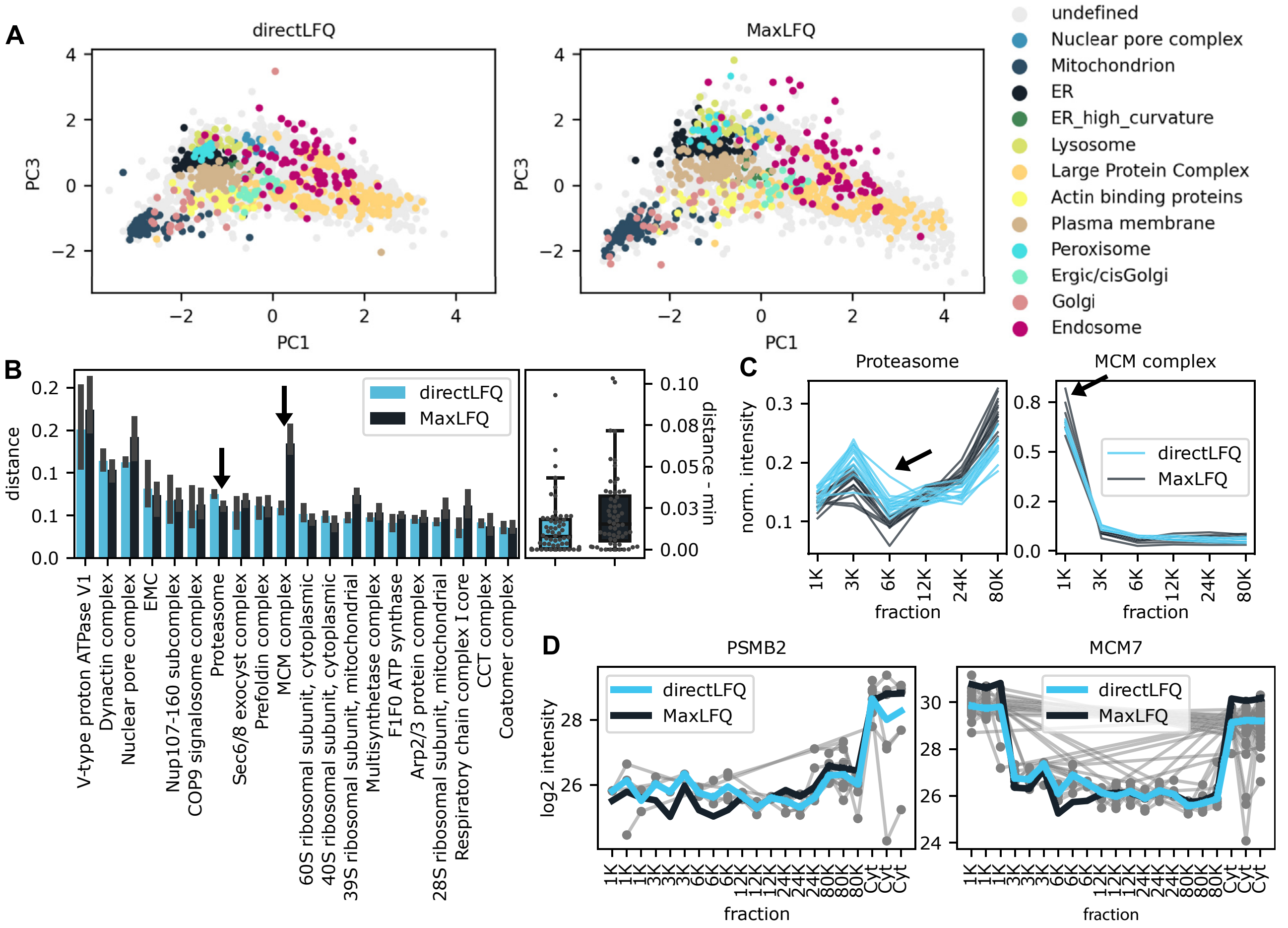


FIG. 4. **Applying directLFQ to dynamic organellar maps data from ref. (28) and comparing with MaxLFQ.** *A*, principal component analysis maps of the dynamic organellar maps data in which protein clusters are color coded. Several clusters such as Golgi and Mitochondrion are separated more clearly with directLFQ. *B*, quantitative assessment of the similarity of the intensity profiles of protein clusters (lower distance means better consistency). On the left, the distances with error bars are displayed for each tested protein cluster. The arrows indicate the two clusters minichromosome maintenance (MCM) complex and Proteasome where directLFQ and MaxLFQ perform best, respectively. On the right, the normalized distances are compared with each other as boxplots; directLFQ has significantly lower distance ($p = 0.014$, two-sided t test). *C*, protein intensity profiles of these two clusters. One outlier trace in each cluster is marked by an arrow. *D*, visualization of the protein profiles over all replicates together with the underlying ion data. The traces show that directLFQ faithfully represents the underlying data.

Index (PyPI) as well as through one-click installers coupled to a GUI. As the concepts underlying directLFQ are relatively straightforward, we hope for wider use of the algorithm within the community.

The aim of directLFQ is to provide a scalable alternative to MaxLFQ with equal or better performance. While MaxLFQ is the most popular and most widely used quantification algorithm, it should be noted that a wide variety of other LFQ quantification approaches exist. The “topN” approach is simple yet widely used, in which the N “best” peptides (typically the most intense one) are selected for quantification (40). While this approach is fast, it does not take into account issues such as missing intensities. A further approach is based on linear models, where effect sizes of proteins are fit toward underlying peptide data, for example, implemented in the MSstats (15) or the MSqRob (13, 19) packages. These models can improve downstream analysis of proteomics data such as differential expression analysis. However, they depend on further inputs about the experimental design (“experimental design template”) and are solved with linear regression, which does not scale linearly along all axes. Furthermore, a variety of computational methods have been proposed that focus on accurate feature selection, especially with the advent of DIA. Tools such as mapDIA (17) or DIA-NN (24) use correlation-based metrics between transitions for selection of well-quantified precursors, while other solutions ((18), diffacto package and (12), part of MSstats) present statistical approaches for feature selection. Avant-Garde (16) uses a genetic algorithm and integrates several subscores about fragment transitions for confident extraction of precursor intensities. Many of these approaches are likely to improve quantification performance as compared with using directLFQ on the unprocessed data. In these cases, it is possible to feed the selected or refined features from the upstream tools into directLFQ.

We show that directLFQ compares favorably in biological and technical benchmarks to the state-of-the-art implementations of MaxLFQ algorithms. In both DDA and the DIA spike-in datasets (Fig. 3, *A* and *C*), the overall outcomes are similar on the qualitative level, but notable differences in the quantification for directLFQ and MaxLFQ remain. While the MaxLFQ and the directLFQ approach both have the same goal of calculating accurate protein intensities given the data, the algorithms are different. MaxLFQ does not have an equivalent to shifting peptide intensity traces on top of each other, an inherently stable method. Instead, it is based on minimizing ratios. This seems to be the underlying reason why directLFQ can be more stable than MaxLFQ in some situations.

The directLFQ approach effectively deals with quantification challenges such as differing sample loadings, differing ionization efficiencies between peptides, as well as missing values. It also provides normalized peptide or fragment-ion intensities, which allows retracing the protein intensity estimation, enabling inspection of individual peptides.

Nevertheless, as in the other current protein intensity estimation approaches, some challenges remain.

First, directLFQ performs linear per-sample normalization, meaning that one scaling factor per sample is derived. If shifts occur in a nonlinear manner, for example, due to distortions in the chromatographic elution profile, this is not corrected by directLFQ. However, the occurrence or severity of the distortion depends on the individual case. If single precursor elution profiles suffer from distortion, this is often compensated by having multiple precursors quantified per protein. As we use the median to estimate the protein intensity, this is robust against such outliers. More systematic effects such as increased peak tailing or peak broadening throughout the whole run are ideally corrected for already, whereas directLFQ automatically corrects the “linear components” of such effects.

Second, directLFQ solves the ionization efficiency problem (different intensity levels of peptides of the same proteins) by using relative instead of absolute quantification. This is done by (implicitly) comparing the relative quantification of all samples. This stabilizes the protein intensity estimation and also means that each individual sample can influence the protein intensity estimation of all other samples. In other words, actions like adding or removing samples may slightly alter the protein intensity estimations of all other samples. In practice, this means that all comparisons have to be done on a dataset that has been processed in the same directLFQ analysis.

Third, recall that the aim of directLFQ is to estimate protein intensities. By definition, this reduces multiple data points that could describe the protein to a single data point that is the “best guess” protein intensity. This reduces overall complexity and is useful for many tasks, such as globally analyzing protein behavior. However, this aspect of quantification only represents a subset of the tasks and algorithms required for comprehensive analysis of quantitative proteomics data. In particular, it is usually necessary to perform statistical analyses such as differential expression analysis in order to determine the regulation between biological conditions. For these, it can be more informative to retain the ion-level information (precursor or transition intensities) instead of working with protein intensity estimates (19, 36). For this, resolving peptide level information more deeply is an outstanding challenge in quantitative proteomics. In particular, interferences, systematic biases, and assessment of consistency from the basic ion-level to the protein or gene level will still need to be better addressed, a topic that we are working on already.

Fourth, directLFQ uses the standard output tables of common proteomics search engines as an input. These output tables all contain a protein group mapping, where peptides are assigned to a “best guess” group of protein isoforms, usually based on Occam’s Razor principle. The division into groups of isoforms, *e.g.*, due to splicing or also handling of peptides mapping to multiple homologous genes, is therefore left entirely to the search engine. In the case of a false mapping, this will result in nonoptimal protein intensities. It should be noted, however, that alignment of intensity traces with directLFQ allows comparison of the relative quantitative behavior of peptides. In the case of differential alternative splicing, such peptides are expected to behave differently (41, 42) and it is therefore possible to visualize such diverging peptide intensity traces using directLFQ. We are currently addressing these issues in follow-up work.

The possibility to quantify increasingly large numbers of samples becomes more and more important with the advent of high-throughput proteomics approaches, such as measuring very large cohorts of patients. In addition, the emerging field of single cell proteomics will drastically increase the number of cells (and therefore samples) that will be measured and quantified. With directLFQ we provide an approach that allows fast quantification for all such scenarios. Furthermore, directLFQ can already be used on other MS data types such as TMT and we see no reason why it should not be useful for RNA sequencing data, too.

DATA AVAILABILITY

All code to reproduce the results shown in this study is available under https://github.com/MannLabs/directlfq with downloaders for the underlying data provided there. The identifiers of the PRIDE repositories used to download the original data are provided in the respective Experimental Results sections. The MaxQuant results tables used for the organellar maps data can be downloaded under the PRIDE reviewer account provided with the submission.

Supplemental data—This article contains supplemental data.

Acknowledgments—We thank our colleagues in the department of Proteomics and Signal Transduction and the Clinical Proteomics group at the NNF Center for Protein Research for help and fruitful discussions, especially Patricia Skowronek, Marvin Thielert, Maximilian Strauss, and Jakob Bader. This study was supported by The Max-Planck Society for the Advancement of Science.

Funding and additional information—C. A., S. W., and A. C. M. acknowledge funding from the Bavarian State Ministry of Health and Care as a part of “DigiMed Bayern” (grant No: DMB-1805-0001).

Author contributions—C. A. and M. M. conceptualization; C. A. methodology; C. A., J. P. S., and A. C. M. formal analysis; S. W. resources; C. A. and M. M. writing – original draft; C. A. and M. M. writing – review & editing.

Conflict of interest—The authors declare no competing interests.

Abbreviations—The abbreviations used are: CV, coefficient of variation; DDA, data-dependent acquisition; DIA, data-independent acquisition; DOM, dynamic organellar maps; GUI, graphical user interface; LFQ, label-free quantification; MCM, minichromosome maintenance; MS, mass spectrometry; TMT, tandem mass tag; QC, quality control.

Received February 22, 2023, and in revised form, May 16, 2023
Published, MCPRO Papers in Press, May 22, 2023, https://doi.org/10.1016/j.mcpro.2023.100581

REFERENCES

1. Aebersold, R., and Mann, M. (2016) Mass-spectrometric exploration of proteome structure and function. *Nature* **537**, 347–355
2. Niu, L., Thiele, M., Geyer, P. E., Rasmussen, D. N., Webel, H. E., Santos, A., *et al*. (2022) Noninvasive proteomic biomarkers for alcohol-related liver disease. *Nat. Med.* **28**, 1277–1287
3. Bader, J. M., Geyer, P. E., Müller, J. B., Strauss, M. T., Koch, M., Leypoldt, F., *et al*. (2020) Proteome profiling in cerebrospinal fluid reveals novel biomarkers of Alzheimer’s disease. *Mol. Syst. Biol.* **16**, e9356
4. Brunner, A., Thielert, M., Vasilopoulou, C., Ammar, C., Coscia, F., Mund, A., *et al*. (2022) Ultra-high sensitivity mass spectrometry quantifies single-cell proteome changes upon perturbation. *Mol. Syst. Biol.* **18**, e10798
5. Specht, H., Emmott, E., Petelski, A. A., Huffman, R. G., Perlman, D. H., Serra, M., *et al*. (2021) Single-cell proteomic and transcriptomic analysis of macrophage heterogeneity using SCoPE2. *Genome Biol.* **22**, 50
6. Mund, A., Coscia, F., Kriston, A., Hollandi, R., Kovács, F., Brunner, A.-D., *et al*. (2022) Deep visual proteomics defines single-cell identity and heterogeneity. *Nat. Biotechnol.* **40**, 1231–1240
7. Schubert, O. T., Röst, H. L., Collins, B. C., Rosenberger, G., and Aebersold, R. (2017) Quantitative proteomics: challenges and opportunities in basic and applied research. *Nat. Protoc.* **12**, 1289–1294
8. Nesvizhskii, A. I., Vitek, O., and Aebersold, R. (2007) Analysis and validation of proteomic data generated by tandem mass spectrometry. *Nat. Methods* **4**, 787–797
9. Zhang, F., Ge, W., Ruan, G., Cai, X., and Guo, T. (2020) Data-independent acquisition mass spectrometry-based proteomics and software tools: a Glimpse in 2020. *Proteomics* **20**, 1900276
10. Cox, J., Hein, M. Y., Luber, C. A., Paron, I., Nagaraj, N., and Mann, M. (2014) Accurate proteome-wide label-free quantification by delayed normalization and maximal peptide ratio extraction, termed MaxLFQ. *Mol. Cell. Proteomics* **13**, 2513–2526
11. Weisser, H., Nahnsen, S., Grossmann, J., Nilse, L., Quandt, A., Brauer, H., *et al*. (2013) An automated pipeline for high-throughput label-free quantitative proteomics. *J. Proteome Res.* **12**, 1628–1644
12. Tsai, T.-H., Choi, M., Banfai, B., Liu, Y., MacLean, B. X., Dunkley, T., *et al*. (2020) Selection of features with consistent profiles improves relative protein quantification in mass spectrometry experiments. *Mol. Cell. Proteomics* **19**, 944–959
13. Sticker, A., Goeminne, L., Martens, L., and Clement, L. (2020) Robust summarization and Inference in proteome-wide label-free quantification. *Mol. Cell. Proteomics* **19**, 1209–1219
14. David, F. N., and Tukey, J. W. (1977) Exploratory data analysis. *Biometrics* **33**, 768
15. Choi, M., Chang, C.-Y., Clough, T., Broudy, D., Killeen, T., MacLean, B., *et al*. (2014) MSstats: an R package for statistical analysis of quantitative mass spectrometry-based proteomic experiments. *Bioinformatics* **30**, 2524–2526
16. Vaca Jacome, A. S., Peckner, R., Shulman, N., Krug, K., DeRuff, K. C., Officer, A., *et al*. (2020) Avant-garde: an automated data-driven DIA data curation tool. *Nat. Methods* **17**, 1237–1244
17. Teo, G., Kim, S., Tsou, C.-C., Collins, B., Gingras, A.-C., Nesvizhskii, A. I., *et al*. (2015) mapDIA: preprocessing and statistical analysis of quantitative proteomics data from data independent acquisition mass spectrometry. *J. Proteomics* **129**, 108–120
18. Zhang, B., Pirmoradian, M., Zubarev, R., and Käll, L. (2017) Covariation of peptide abundances accurately reflects protein concentration differences. *Mol. Cell. Proteomics* **16**, 936–948
19. Goeminne, L. J. E., Gevaert, K., and Clement, L. (2016) Peptide-level robust ridge regression improves estimation, sensitivity, and specificity in data-dependent quantitative label-free shotgun proteomics. *Mol. Cell. Proteomics* **15**, 657–668
20. Sinitcyn, P., Hamzeiy, H., Salinas Soto, F., Itzhak, D., McCarthy, F., Wichmann, C., *et al*. (2021) MaxDIA enables library-based and library-free data-independent acquisition proteomics. *Nat. Biotechnol.* **39**, 1563–1573
21. Cox, J., and Mann, M. (2008) MaxQuant enables high peptide identification rates, individualized p.p.b.-range mass accuracies and proteome-wide protein quantification. *Nat. Biotechnol.* **26**, 1367–1372
22. [preprint] Strauss, M. T., Bludau, I., Zeng, W.-F., Voytik, E., Ammar, C., Schessner, J., *et al*. (2021) AlphaPept, a modern and open framework for MS-based proteomics. *bioRxiv*. https://doi.org/10.1101/2021.07.23.453379
23. Bruderer, R., Bernhardt, O. M., Gandhi, T., Miladinović, S. M., Cheng, L.-Y., Messner, S., *et al*. (2015) Extending the limits of quantitative proteome profiling with data-independent acquisition and application to acetaminophen-treated three-dimensional liver microtissues. *Mol. Cell. Proteomics* **14**, 1400–1410
24. Demichev, V., Messner, C. B., Vernardis, S. I., Lilley, K. S., and Ralser, M. (2020) DIA-NN: neural networks and interference correction enable deep proteome coverage in high throughput. *Nat. Methods* **17**, 41–44
25. Yu, F., Haynes, S. E., and Nesvizhskii, A. I. (2021) IonQuant enables accurate and sensitive label-free quantification with FDR-controlled match-between-runs. *Mol. Cell. Proteomics* **20**, 100077
26. Mann, M. (2019) The ever expanding scope of electrospray mass spectrometry—a 30 year journey. *Nat. Commun.* **10**, 3744
27. Pham, T. V., Henneman, A. A., and Jimenez, C. R. (2020) iq: an R package to estimate relative protein abundances from ion quantification in DIA-MS-based proteomics. *Bioinformatics* **36**, 2611–2613
28. [preprint] Schessner, J. P., Albrecht, V., Davies, A. K., Sinitcyn, P., and Borner, G. H. H. (2021) Deep and fast label-free dynamic organellar mapping. *bioRxiv*. https://doi.org/10.1101/2021.11.09.467934
29. Meier, F., Geyer, P. E., Virreira Winter, S., Cox, J., and Mann, M. (2018) BoxCar acquisition method enables single-shot proteomics at a depth of 10,000 proteins in 100 minutes. *Nat. Methods* **15**, 440–448

30. Perez-Riverol, Y., Bai, J., Bandla, C., García-Seisdedos, D., Hewapathirana, S., Kamatchinathan, S., *et al*. (2022) The PRIDE database resources in 2022: a hub for mass spectrometry-based proteomics evidences. *Nucleic Acids Res.* **50**, D543–D552
31. Bian, Y., Zheng, R., Bayer, F. P., Wong, C., Chang, Y.-C., Meng, C., *et al*. (2020) Robust, reproducible and quantitative analysis of thousands of proteomes by micro-flow LC–MS/MS. *Nat. Commun.* **11**, 157
32. Huang, T., Bruderer, R., Muntel, J., Xuan, Y., Vitek, O., and Reiter, L. (2020) Combining precursor and fragment information for improved detection of differential abundance in data independent acquisition. *Mol. Cell. Proteomics* **19**, 421–430
33. Demichev, V., Tober-Lau, P., Nazarenko, T., Lemke, O., Kaur Aulakh, S., Whitwell, H. J., *et al*. (2022) A proteomic survival predictor for COVID-19 patients in intensive care. *PLoS Digit Health* **1**, e0000007
34. Wang, D., Eraslan, B., Wieland, T., Hallström, B., Hopf, T., Zolg, D. P., *et al*. (2019) A deep proteome and transcriptome abundance atlas of 29 healthy human tissues. *Mol. Syst. Biol.* **15**, e8503
35. [preprint] Michaelis, A. C., Brunner, A.-D., Zwiebel, M., Meier, F., Strauss, M. T., Bludau, I., *et al*. (2021) The social architecture of an in-depth cellular protein interactome. *bioRxiv*. https://doi.org/10.1101/2021.10.24.465633
36. Ammar, C., Gruber, M., Csaba, G., and Zimmer, R. (2019) MS-EmpiRe utilizes peptide-level noise distributions for ultra-sensitive detection of differentially expressed proteins. *Mol. Cell. Proteomics* **18**, 1880–1892
37. Wiśniewski, J. R., and Mann, M. (2016) A proteomics approach to the protein normalization problem: selection of unvarying proteins for MS-based proteomics and Western Blotting. *J. Proteome Res.* **15**, 2321–2326
38. Wiśniewski, J. R., Hein, M. Y., Cox, J., and Mann, M. (2014) A "proteomic ruler" for protein copy number and concentration estimation without spike-in standards. *Mol. Cell. Proteomics* **13**, 3497–3506
39. Thompson, A., Schäfer, J., Kuhn, K., Kienle, S., Schwarz, J., Schmidt, G., *et al*. (2003) Tandem mass tags: a novel quantification strategy for comparative analysis of complex protein mixtures by MS/MS. *Anal. Chem.* **75**, 1895–1904
40. Silva, J. C., Gorenstein, M. V., Li, G.-Z., Vissers, J. P. C., and Geromanos, S. J. (2006) Absolute quantification of proteins by LCMSE. *Mol. Cell. Proteomics* **5**, 144–156
41. Ammar, C. (2020) Context-based Analysis of Mass Spectrometry Proteomics Data. Ph.D. dissertation, LMU Munchen.
42. Bludau, I., Frank, M., Dörig, C., Cai, Y., Heusel, M., Rosenberger, G., *et al*. (2021) Systematic detection of functional proteoform groups from bottom-up proteomic datasets. *Nat. Commun.* **12**, 3810