Assessment of false discovery rate control in tandem mass spectrometry analysis using entrapment

Bo Wen[1], Jack Freestone[2], Michael Riffle[1], Michael J. MacCoss[1], William S. Noble[*1,3], and Uri Keich[*2]

[1]Department of Genome Sciences, University of Washington
[2]School of Mathematics and Statistics, University of Sydney
[3]Paul G. Allen School of Computer Science and Engineering, University of Washington

Abstract

A pressing statistical challenge in the field of mass spectrometry proteomics is how to assess whether a given software tool provides accurate error control. Each software tool for searching such data uses its own internally implemented methodology for reporting and controlling the error. Many of these software tools are closed source, with incompletely documented methodology, and the strategies for validating the error are inconsistent across tools. In this work, we identify three different methods for validating false discovery rate (FDR) control in use in the field, one of which is invalid, one of which can only provide a lower bound rather than an upper bound, and one of which is valid but under-powered. The result is that the field has a very poor understanding of how well we are doing with respect to FDR control, particularly for the analysis of data-independent acquisition (DIA) data. We therefore propose a theoretical formulation of entrapment experiments that allows us to rigorously characterize the behavior of the various entrapment methods. We also propose a more powerful method for evaluating FDR control, and we employ that method, along with other existing techniques, to characterize a variety of popular search tools. We empirically validate our entrapment analysis in the fairly well-understood DDA setup before applying it in the DIA setup. We find that none of the DIA search tools consistently controls the FDR at the peptide level, and the tools struggle particularly with analysis of single cell datasets.

1 Introduction

In mass spectrometry-based proteomics, controlling the false discovery rate (FDR) among the set of reported proteins, peptides, or peptide-spectrum matches (PSMs) is easy to get wrong. Most widely used FDR control procedures in proteomics involve target-decoy competition (TDC) where the observed spectra are searched against a bipartite database comprised of real ("target") and shuffled or reversed ("decoy") peptides [8]. Ideally, these procedures would control the actual proportion of false positives among the reported set of discoveries, which is known as the "false discovery proportion" (FDP); however, in practice this is impossible because the FDP varies from experiment to experiment and cannot be directly measured [2]. Instead, we control the FDR, which is the expected value of the FDP, i.e., its theoretical average over all random aspects of the experiment and its analysis. Although the TDC procedure can be rigorously proven to control the FDR in a spectrum-centric search, subject to several reasonable assumptions [14], in practice many analysis pipelines implement variants of the procedure that potentially fail to control the FDR. For example, PSM-level control using TDC is inherently problematic [14, 27]. Similarly, most pipelines involve training a semi-supervised classification algorithm, such as Percolator [16] or PeptideProphet [18], to re-rank peptide-spectrum matches (PSMs), which in practice can compromise FDR control [11].

Failure to correctly control the FDR can have serious negative implications. Most obviously, if a given analysis pipeline tends to underestimate the FDP—i.e., if the pipeline claims that the FDR is controlled,

*william-noble@uw.edu, uri.keich@sydney.edu.au

say, at 1% but the actual average of the FDP is 5%—then the scientific conclusions drawn from those experiments may be invalid. Perhaps more insidiously, invalid FDR control can also impact our choice of analysis pipelines and make comparison of instrument platforms and proteomics workflows impossible. To see why this is the case, consider a hypothetical tool that consistently fails to control the FDR. In a benchmarking experiment, if we compare the number of proteins detected by a collection of analysis tools, all using a fixed FDR threshold, then the liberally biased tool will have a clear (and unfair) advantage.

To address this concern, therefore, it is important to have a rigorous procedure to evaluate the validity of the FDR control provided by a proteomics analysis pipeline. The standard way to carry out such an evaluation is via an "entrapment" procedure [13, 31], which involves expanding the tool's input dataset so that its search space includes verifiably false entrapment discoveries. Most commonly this is done by expanding the database with peptides taken from proteomes of species that are not expected to be found in the sample, so any such reported peptide is presumably a false discovery. Critically, the distinction between the original input and its entrapment expansion is hidden from the tool itself, so that the entrapment discoveries can subsequently be used to evaluate the tool's FDR control procedure.

Designing an entrapment experiment involves making two major decisions: how should the tool's input be expanded and how should the entrapment discoveries be used to evaluate the tool's FDR control procedure. In this work, we primarily focus on the second decision. This is motivated by a survey of published entrapment experiments, which suggests that, while conceptually simple, correctly carrying out an entrapment estimation can be tricky [3, 6, 24, 26, 28–30, 33–36, 38, 41, 43, 45, 47–49]. Indeed, our survey identified a variety of estimation methods, a few of which are invalid as either a lower bound or as an upper bound estimate, while others are often incorrectly used, drawing potentially incorrect conclusions in both cases.

In this work, we expose common errors in existing approaches and introduce a formal framework for entrapment experiments. This framework allows us to rigorously establish properties of existing estimators and to propose a novel entrapment method that allows more accurate evaluation of FDR control for mass spectrometry analysis pipelines. We used entrapment analysis of several popular data-dependent acquisition (DDA) tools to validate that our framework yields results consistent with the field's consensus that these tools generally seem to control the FDR. In contrast, a similar analysis of three popular data-independent acquisition (DIA) tools (DIA-NN [6], Spectronaut, and EncyclopeDIA [40]) finds that none of these search tools consistently controls the FDR at the peptide level across all the datasets we investigated. Furthermore, this problem becomes much worse when these DIA tools are evaluated at the protein level. These results suggest an opportunity for the field: insofar as existing methods yield results with unexpectedly high levels of noise, we anticipate that reducing this noise by accurately controlling the FDR has the potential to yield better statistical power in downstream analyses.

2 Results

2.1 Many published studies use entrapment incorrectly

Before describing various methods for entrapment analyses, it is important to distinguish between methods that provide estimated upper bounds versus lower bounds of the FDP, and understand their limitations. The primary output of an entrapment procedure can be summarized by plotting the entrapment-estimated FDP as a function of the FDR cutoff used (or reported as a q-value) by the evaluated tool. If the entrapment procedure provides an estimated upper bound on the FDP, then the entrapment analysis suggests that the actual FDP falls below the plotted curve. Conversely, the entrapment procedure may provide a lower bound, indicating that the actual FDP falls above the curve. Therefore, applying both an upper bounding and a lower bounding entrapment procedure to a given analysis tool can yield one of three outcomes (Figure 1): (1) if the upper bound falls below the line $y = x$, then we can take this as empirical evidence suggesting that the tool successfully controls the FDR; (2) conversely, if the lower bound falls above $y = x$, then we can use it as evidence suggesting that the tool fails to control the FDR; (3) if the estimated upper bound is above $y = x$ and the lower bound is below $y = x$, then the experiment is inconclusive.

An important caveat to all of these analyses is that an entrapment procedure aims to gauge the FDP among the discoveries reported by the tool; however, the tool itself is typically designed to control its expected value, the FDR. Hence, we should use reasonably large datasets in our entrapment analysis, where the FDP is typically close to the FDR (by the law of large numbers), or we should average the empirical FDP over

Citation	Tools analyzed	DIA	DDA	Entrapment method	Entrapment composition	Valid?
Peckner 2018 [35]	Specter	✓		Other	foreign	
Demichev 2020 [6]	DIA-NN	✓		Sample	foreign	
Lu 2020 [29]	O-Pair		✓	Combined*	foreign	✓
Sinitcyn 2021 [41]	MaxDIA	✓		Sample	foreign	
Lu 2021 [30]	DIAmeter	✓		Sample	shuffled	
The 2022 [45]	PPGF		✓	Combined*	shuffled	✓
Lee 2022 [26]	cTDS		✓	Sample	foreign	
Na 2022 [33]	Deephos		✓	Sample	foreign	
Ye 2023 [47]	DIA-NN, Spectronaut	✓		Lower bound	shuffled	
Lancaster 2023 [24]	Spectronaut	✓		Lower bound	foreign	
Scott 2023 [38]	GPS	✓		Lower bound	foreign	
Nowatzky 2023 [34]	Mistle		✓	Combined	foreign	✓
Yu 2023 [48]	MSFragger-DIA, DIA-NN	✓		Sample	foreign	
Penny 2023 [36]	Spectronaut	✓		Sample	foreign	
Zhang 2023 [49]	Mzion		✓	Lower bound	foreign	
Lou 2023 [28]	Benchmarking	✓		Lower bound	foreign	
Strauss 2024 [43]	AlphaPept		✓	Other	foreign	
Bubis 2024 [3]	Spectronaut	✓		Sample	shuffled	

Table 1: **Summary of previous entrapment analyses.** "PPGF" is the picked protein group FDR method. The final column indicates whether the entrapment method is deemed invalid to demonstrate FDR control. * Lu *et al.* 2020 and The *et al.* 2022 employed large entrapment databases for the combined method.

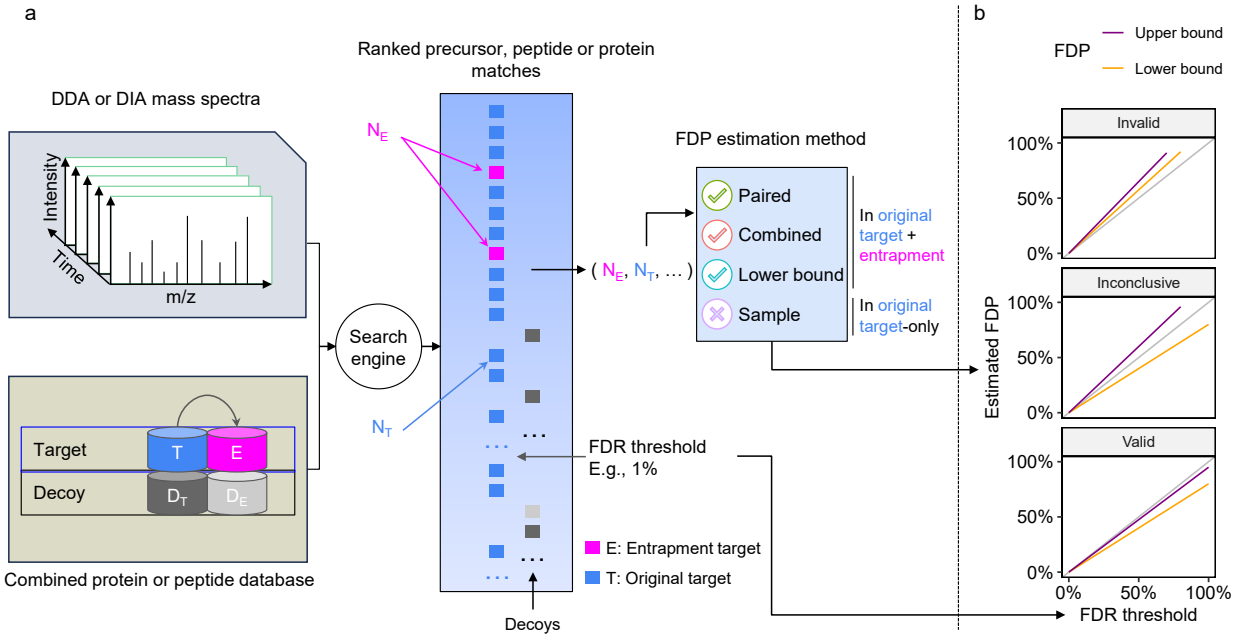

Figure 1: **Entrapment strategy for FDR control evaluation. (a)** Schematic of an entrapment method. The target database is augmented with entrapment sequences, and the augmented database is used by the search tool to produce a ranked list of peptides. In this example, we show the decoys that the tool is using to control the FDR, but the tool can use other methods to control the FDR. The target/entrapment labels are hidden from the search engine but are revealed to the entrapment method, allowing it to provide an estimated FDP. Note that some entrapment estimation methods require additional inputs besides the count of the number of original and entrapment targets. **(b)** Comparing the FDR reported or used by a given analysis tool (x-axis) to the estimated upper bound (blue) and lower bound (red) on the FDP produced by two different entrapment estimation methods allows us to conclude that the tool's FDR estimates are valid (bottom) or invalid (top). If the bounds fall on either side of the line $y = x$ (middle), then the analysis is inconclusive.

multiple entrapment sets, which ameliorates the problem. Even in those cases, the random nature of the FDP implies that some deviations above $y = x$ may be acceptable if they are small and rare.

With that caveat in mind, note that from a statistical perspective, the onus falls on the tool developer to establish that the method indeed controls the FDR, so "inconclusive" (scenario 3 above) is a tentative strike against the tool. Furthermore, FDR control should be universal; consequently, a valid FDR control procedure should achieve scenario (1) for any reasonably large dataset.

We next present the three main approaches in the literature to estimating the FDP. For concreteness, we describe entrapment procedures that expand the peptide database; however, our arguments apply to any type of entrapment expansion, as established rigorously in Supplementary Notes S1-S2. The first two methods aim to estimate the FDP in the combined, target+entrapment, list of original target and entrapment discoveries, whereas the third method aims to estimate the FDP in a stricter list, focusing only on the original target discoveries and excluding the entrapment discoveries.

The first method, which we refer to as the "combined" method, estimates the FDP among the target+entrapment ($\mathcal{T} \cup \mathcal{E}_\mathcal{T}$) discoveries while taking into account r, which is the effective ratio of the entrapment to original target database size:

$$\widehat{\mathrm{FDP}}_{\mathcal{T} \cup \mathcal{E}_\mathcal{T}} = \frac{N_\mathcal{E}(1 + 1/r)}{N_\mathcal{T} + N_\mathcal{E}}, \tag{1}$$

where $N_\mathcal{T}$ and $N_\mathcal{E}$ denote the number of target and entrapment discoveries respectively. In Supplementary Note S2.2 we prove that, under an assumption analogous to the equal-chance assumption that TDC relies on, Equation (1) provides an estimated upper bound, i.e., on average it overestimates the true FDP. Thus, the combined method can be used to provide empirical evidence that a given tool successfully controls the FDR among its discoveries (Figure 1b). Notably, with $r = 1$ Equation (1) reduces to Elias and Gygi's original estimation of the FDR in the concatenated target-decoy database, which in our case is the target+entrapment database [8]. This combined method has been used in several previous studies (Table 1); for example, The *et al.* used it to evaluate the "picked protein" method for FDR control [45]. It has also been used to evaluate the DDA analysis tool Mistle [34], as well as for evaluation of the O-Pair method for gycloproteomics data [29].

Unfortunately, the combined estimation is often applied incorrectly to establish FDR control after removing the $1/r$ term:

$$\underline{\widehat{\mathrm{FDP}}}_{\mathcal{T} \cup \mathcal{E}_\mathcal{T}} = \frac{N_\mathcal{E}}{N_\mathcal{T} + N_\mathcal{E}}. \tag{2}$$

The problem is that, as we prove in Supplementary Note S2.1, without the $1/r$ term and assuming that any entrapment discovery is indeed false, Equation (2) represents a lower bound on the FDP. As such, this method can only be used to indicate that a tool *fails* to control the FDR (Figure 1b), rather than as evidence of FDR control. In what follows we refer to Equation (2) as the "lower bound." Table 1 shows that multiple studies incorrectly used the lower bound to validate FDR control, including a recent benchmarking study to evaluate several widely used DIA tools for proteomics and phosphoproteomics DIA data analysis [28]. In that review, the lower bound was used both correctly to point out questionable FDR control, as well as incorrectly as evidence of FDR control.

The third approach, which we refer to as the "sample" estimation method, estimates the FDP only among the original target ($\mathcal{T}$) discoveries as

$$\widehat{\mathrm{FDP}}_{\mathcal{T}} = \frac{N_\mathcal{E} \cdot 1/r}{N_\mathcal{T}}. \tag{3}$$

This approach has been employed to evaluate DIA-NN [6], MSFragger-DIA [48], MaxDIA [41], and DIAmeter [30]. We argue that the sample entrapment method is inherently flawed because, while typically underestimating the FDP, it can also overestimate it in some unusual cases (Supplementary Note S3). Hence, the sample estimation method cannot be used to provide empirical evidence that a tool controls the FDR nor that the tool fails to control the FDR.

Our literature review, summarized in Table 1, indicates that many publications fail to correctly employ the entrapment estimation. Overall, we identified only three studies that correctly use entrapment estimation, and all three focused on DDA analysis [29, 34, 45]. A common mistake is to use the lower bound method, which cannot establish that a given method correctly controls the FDR [24, 28, 38, 47, 49], or to use the

problematic sample entrapment method [6, 30, 41, 48]. Further discussion of some of the studies in Table 1 is provided in Supplementary Note S5.

Note that, in addition to employing different methods to estimate the FDP, the above studies also differ with respect to the entrapment expansion, which in this case amounts to how the entrapment database is constructed. In the "shuffled entrapment" approach, the entrapment sequences are derived analogously to decoy sequences by shuffling the corresponding target sequences, whereas in the "foreign entrapment" approach they are taken from the proteome of some other species.

2.2 The paired estimation method yields a tighter upper bound on the FDP

As noted above, the combined estimation method provides an estimated upper bound. In practice, we observed that this method can often significantly overestimate the FDP, which motivated us to propose a complementary "paired estimation" approach. By taking advantage of sample-entrapment pairing information, this method allows us to reduce the conservative bias of the combined method while still retaining its upper bound nature. As such, the paired estimation method is more likely to provide evidence of proper FDR control than the combined method is. For this method to work, say, in peptide-level analysis, we require that each original target peptide be paired with a unique entrapment peptide (so in particular, $r = 1$). In practice, this means that the paired estimation method requires a shuffling or reversal to generate the entrapment peptides.

Given such paired entrapment peptides, and still considering peptide-level analysis, the paired method estimates the FDP in the list of target+entrapment discovered peptides by

$$\widehat{\mathrm{FDP}}^{*}_{\mathcal{T}\cup\mathcal{E}_{\mathcal{T}}} = \frac{N_{\mathcal{E}} + N_{\mathcal{E}\geq s>\mathcal{T}} + 2\cdot N_{\mathcal{E}>\mathcal{T}\geq s}}{N_{\mathcal{T}} + N_{\mathcal{E}}}, \tag{4}$$

where s is the discovery cutoff score; $N_{\mathcal{E}\geq s>\mathcal{T}}$ denotes the number of discovered entrapment peptides (scoring $\geq s$) for which their paired original target peptides scores $< s$; and $N_{\mathcal{E}>\mathcal{T}\geq s}$ is the number of discovered entrapment peptides for which the paired original target peptides scored lower but were still also discovered. In Supplementary Note S2.3 we recast Equation (4) so it can be applicable in our more general entrapment framework. In addition, we introduce the "k-matched" generalization of the paired method that, in the case of peptide-level analysis for example, relies on a larger entrapment database, where each target peptide is uniquely associated with k entrapment ones (so $r = k$). Finally, we prove that, under an assumption akin to TDC's equal chance assumption, both the paired method and its k-matched generalization are valid upper bounds in the same averaged sense that the combined method is.

2.3 Comparing the estimation methods using data from a controlled experiment

We first demonstrate the qualitative differences among the above estimation methods—lower bound, sample, combined, and paired—using the ISB18 dataset, which consists of DDA data generated from a known mixture of 18 proteins [21]. We used the Tide search engine [7] to carry out FDR control at the peptide level (Section 4.1.2). Due to the relatively small size of the ISB18 dataset, we averaged each entrapment method's estimated FDP over multiple applications, each with different randomly drawn decoy and entrapment databases. Accordingly, our entrapment methods here are reporting the empirical FDR, that is, the average of the FDP estimates over the 100 drawn decoys and entrapments.

In the first experiment, the original target database consists of the ISB18 peptides, and the entrapment part consists of shuffled sequences with $r = 1$. We first focus on the paired and combined methods, both of which are estimating the FDP in the same list of target+entrapment discoveries at the given FDR threshold. Notably, the paired method yields an estimated FDR curve that weaves very closely about the line $y = x$, which is firmly within the 95% coverage band of this estimate (Figure 2a). In contrast, the entire 95% coverage band of the combined estimation lies above the diagonal, demonstrating the upper bound nature of this method. In particular, in this example we can use the paired method to argue that the FDR seems to be controlled in this case (as we expect it to be), but we cannot make that argument using the combined estimate. As expected, the lower bound curve is below the diagonal but, as such, it is uninformative in this case. Finally, the fact that the sample method is also below the diagonal indicates that, like the lower bound, it is probably underestimating the true FDP here.

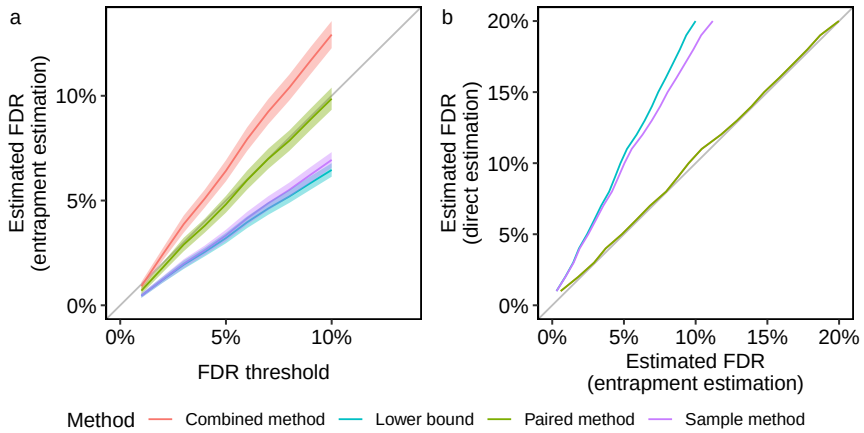

Figure 2: **Entrapment analysis using the ISB18 data.** The figure summarizes two different types of entrapment experiments, both using shuffled sequences with $r = 1$. The lists of discoveries for both panels were generated by searching with Tide, followed by peptide-level FDR control. **(a)** The entrapment-estimated FDR, or the estimated FDP averaged over 100 sets of decoys and entrapments, is plotted (with 95% coverage bands) as a function of the given FDR threshold. The FDP is estimated using four different entrapment methods with the original target database consisting only of the ISB18 peptides. **(b)** The estimated FDR, or the estimated FDP averaged over 100 sets of decoys and entrapments, is plotted against the corresponding castor-based estimates. This experiment uses an original target database consisting of the ISB18 and castor peptides at a ratio of 1:636. The combined and paired curves visually overlap due to the small proportion of native ISB18 peptides in the ISB18+castor database.

In the second experiment, we sought to obtain an independent evaluation of the entrapment methods themselves. We did this by taking advantage of the controlled nature in which the ISB18 dataset was generated to conduct a double entrapment experiment, which is designed to gauge how accurately the entrapment methods estimate the FDP. Specifically, we constructed an extended "original target" database that consisted of the ISB18 peptides augmented by a much larger set of peptides from a foreign species (the castor bean): the ratio of ISB18 to castor peptides was 1:636. We then applied the paired and sample entrapment methods using shuffled sequences ($r = 1$) to estimate the FDP among the reported peptides. The controlled nature of the ISB18 dataset implies that any reported castor peptide is a false discovery. At the same time, with a ratio of 1:636 ISB18-to-castor peptides it is reasonable to assume that any ISB18 reported peptide is a true discovery. Thus, we can directly estimate the FDP in each discovery list and compare it with the estimate produced by the entrapment procedures.

Given the very small proportion of native ISB18 peptides in the extended ISB18+castor target database, it is not surprising that the combined and paired methods essentially coincide in this setup (Figure 2b). Notably, though, both provide very accurate estimates of the FDP as confirmed by the independent, "direct" castor-based estimate, where the latter counts every entrapment or castor peptide as a false discovery. In contrast, the sample entrapment seems to substantially underestimate the castor-based estimate because it is trying to estimate the FDP in the wrong list of discoveries — the ISB18+castor ones — ignoring the fact that the tool was instructed to control the FDR in the larger set of ISB18+castor+shuffled entrapments. Finally, the lower bound also underestimates the FDP, but that is not surprising given its definition.

2.4 Entrapment analysis of DDA search engines is consistent with our theoretical analysis

We next set out to further validate our analysis of the entrapment methods by applying them to gauge the peptide-level FDR control of four well-established DDA search engines—Tide [7], Sage [25], MS-GF+ [20] and MSFragger [22]—using data generated from a complex sample, rather than from a controlled mixture. For MS-GF+ we carry out peptide-level FDR control using the primary search engine scores, whereas for the other three search engines we use a machine learning post-processor: Percolator-RESET for Tide, Sage's

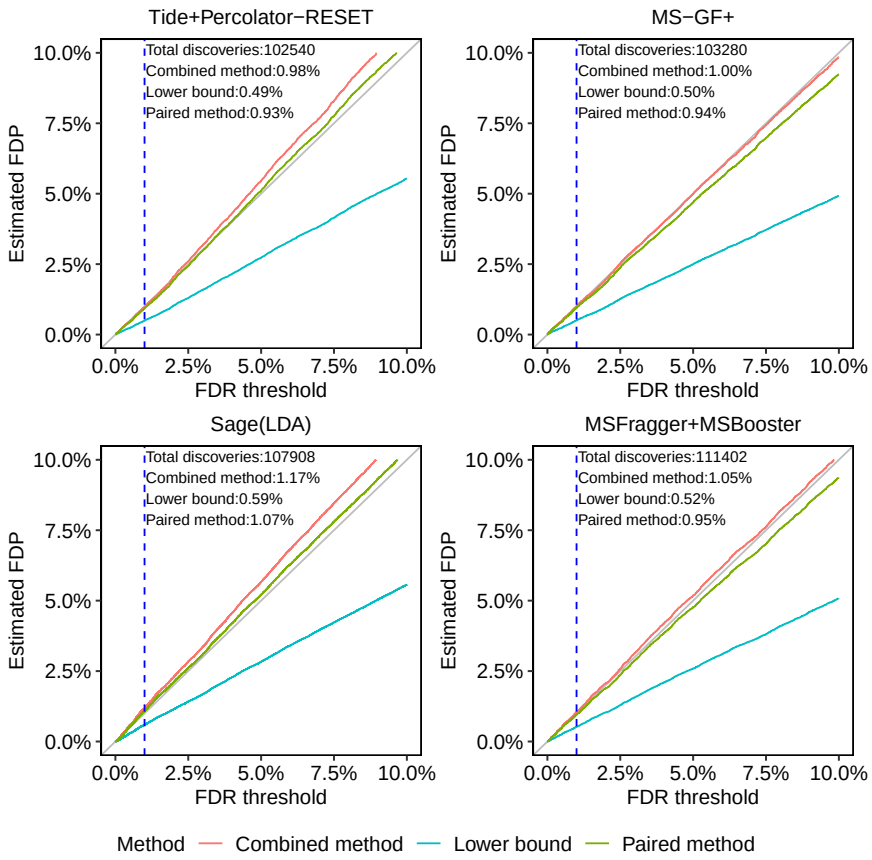

Figure 3: **Comparing entrapment procedures using HEK293 DDA data.** Each figure shows, for a given search procedure, the (shuffled with $r = 1$) entrapment-estimated FDP in the combined list of target peptides that was reported at the given FDR threshold. The dashed vertical line is at the 1% FDR threshold, as are the numbers reported in text in the figure.

built-in linear discriminant analysis, MSBooster [46] for MSFragger. For this analysis, we use data from 24 MS/MS analyses of the human cell line HEK293, searched against target+entrapment databases for which the human reference proteome was taken as the original target sequences, and was paired with shuffled entrapment sequences (so $r = 1$). Thus, in addition to being derived from a more complex sample, this dataset is substantially larger than the ISB18 dataset, as is the original target database. Having established that the sample estimation method is inherently problematic, from here onward we consider only the lower bound, combined, and paired estimation procedures.

A priori, given the established nature of DDA FDR analysis, we expect all of these tools to produce valid FDR estimates. Accordingly, the conclusions we draw from the analysis of the HEK293 data (Figure 3) largely mirror those we drew from the Figure 2a. Specifically, for all peptide-level analysis tools the paired method yields estimated FDPs that are quite close to the diagonal. The more conservative nature of the combined method is on display again: it is always above the paired estimation curve, with the latter already presenting an upper bound. More specifically, relying on the combined method we might have reservations on whether we have evidence that Tide+Pecolator-RESET and Sage control the FDR in this setup, but we can argue for such apparent evidence if we use the paired method. As expected, the lower bound seems to consistently significantly underestimate the FDP.

To gauge the possible impact of the inherent variability of the entrapment sequences, and where relevant, the decoy sequences, we again constructed 95% coverage bands for the estimates. Supplementary Figure S5 shows that, for both Tide and Sage (for which generating these plots was fairly straightforward), the effect of this variability is marginal.

We reach similar conclusions when comparing our entrapment estimation methods using protein level analysis in DDA data with MaxQuant [5]. Specifically, Supplementary Figure S6a demonstrates that the paired method provides us with evidence of MaxQuant's successful protein-level FDR control while the combined methods leaves a certain degree of uncertainty about it. At the same time, comparison of panels (a) and (b) shows that entrapment analyses based on shuffled and foreign sequences are in fairly good agreement with one another in this case as well.

Finally, we further validated our entrapment analysis by applying the combined, paired and lower bound estimates to peptide-level analysis procedures that do not rely on TDC to control the FDR. Specifically, we analyzed two such procedures in the spirit of the PSM-level one described in [17]. Both procedures rely on empirical p-values that are estimated from the decoys generated by shuffling the peptides in the target datbase. The list of discovered peptides is then determined by applying either the Benjamini-Hochberg (BH) [2] or Storey's procedure [42] to the list of empirical p-values (see Section 4.2 for details). To obtain further confidence in our analysis, each entrapment procedure was applied twice, once with shuffled and once with foreign (*Arabidopsis*) entrapment sequences. As expected, in all cases the combined method reported the highest estimated FDP, the lower bound reported the lowest estimate, and the paired method fell between those two (Supplementary Figure S7). The fact that for both the BH and Storey procedures, the combined estimate is well below the diagonal suggests that both are conservative in this case. This is further suggested by the substantially smaller number of discoveries those two procedures make compared with TDC.

2.5 DIA search tools fail to consistently control the FDR at the peptide level, and the problem is much worse at the protein level

Turning to tools for analysis of DIA data, we performed a more extensive evaluation. Specifically, we applied three different search engines—DIA-NN, Spectronaut and EncyclopeDIA—to the ten DIA datasets listed in Table 3. In the case of EncyclopeDIA, we only analyzed the four datasets for which we have gas phase fractionation runs. In each case, we applied all three entrapment estimation methods separately at the peptide level (for EncyclopeDIA) or precursor level (for DIA-NN and Spectronaut) and at the protein level for each search engine. Precursor-level analysis is analogous to peptide-level analysis except that each discovery is a (possibly modified) peptide and a corresponding charge state. In addition, because Spectronaut only estimates precursor-level FDR for each run, rather than for a full experiment, we only report precursor-level results for a single selected MS run for each DIA dataset.

The results of this experiment suggest that the precursor or peptide-level FDR control is frequently questionable, whereas protein-level FDR control is frequently invalid (Table 2). For example, for the human-lumos dataset (Figure 4) we observe that the precursor or peptide-level FDR control appears to be inconclusive,

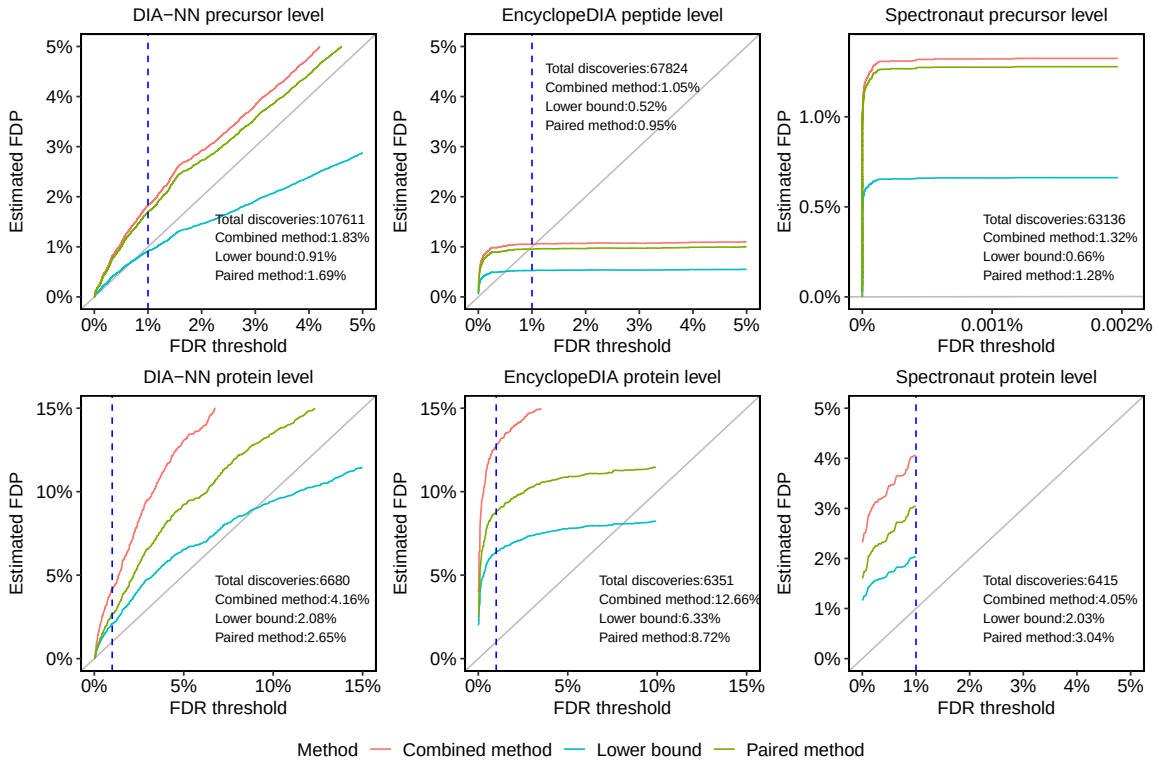

Figure 4: **Entrapment evaluation of the FDR control of DIA analysis tools.** DIA-NN, EncyclopeDIA and Spectronaut were applied to the human-lumos DIA dataset using shuffled entrapment with $r = 1$. The top row shows the precursor or peptide-level estimated FDPs, and the bottom row shows the corresponding protein-level analysis. In the Spectronaut precursor-level plot, the x axis was set to show the maximum FDR threshold reported by the tool, which was less than the 1% threshold set in the analysis. The dashed vertical lines are at the 1% FDR threshold, as are the numbers reported in text in the panels.

Dataset	DIA-NN		EncyclopeDIA		Spectronaut	
	Precursor	Protein	Peptide	Protein	Precursor	Protein
human-astral	0.7-1.3 (?)	1.5-2.1 (X)			0.8-1.5 (?)	1.6-2.3 (X)
human-qe	0.7-1.3 (?)	1.5-2.2 (X)	0.7-1.3 (?)	4.3-6.4 (X)	0.7-1.4 (?)	1.3-1.9 (X)
human-tripletof	0.6-1.1 (?)	1.0-1.5 (X)	0.8-1.5 (?)	2.5-3.7 (X)	0.7-1.4 (?)	1.5-2.8 (X)
yeast-lumos	0.6-1.0 (✓)	1.2-1.4 (X)	0.4-0.7 (✓)	3.5-4.1 (X)	0.8-1.5 (?)	1.3-1.6 (X)
mouse-qe	0.7-1.3 (?)	0.8-1.1 (?)			0.7-1.4 (?)	1.3-1.9 (X)
human-timstof2	0.6-1.1 (?)	0.8-1.2 (?)			0.7-1.5 (?)	1.4-2.1 (X)
human-timstof1	0.7-1.2 (?)	0.8-1.2 (?)			0.7-1.3 (?)	1.1-1.7 (X)
100cell-eclipse	0.9-1.6 (?)	1.3-1.8 (X)			0.9-1.7 (?)	1.8-2.9 (X)
1cell-eclipse	2.3-4.7 (X)	2.0-3.5 (X)			3.8-7.6 (X)	3.0-5.6 (X)
human-lumos	0.9-1.7 (?)	2.1-2.6 (X)	0.5-1.0 (✓)	6.3-8.7 (X)	0.7-1.3 (?)	2.0-3.0 (X)

Table 2: **Entrapment analysis of FDR control of three DIA analysis tools.** Each entry in the table lists the lower bound and paired-estimated upper bound on the empirical FDP among the target+entrapment discoveries reported by the DIA search engine at an FDR threshold of 1%. The entrapment procedures used shuffled entrapment sequences with $r = 1$ and were applied at both the peptide or precursor level and the protein-level. Each entry is followed by an indicator for whether the FDR control for this tool on this dataset is deemed valid (✓), invalid (X), or inconclusive (?). EncyclopeDIA results are provided only for the four datasets with gas phase fractionation data available. Note that the evaluation of each method is based on the full results presented in Figure 4 and Supplementary Figures S8–S13.

whereas the protein-level FDR control is apparently invalid: e.g., the lower bound on EncylopeDIA's FDP is above 6%. Indeed, keeping in mind that the true FDP is likely to be between the lower bound and the paired-estimated upper bound, we see that the protein-level 1% FDR control appears to be consistently invalid for EncyclopeDIA and Spectronaut and mostly invalid for DIA-NN. Although the peptide-level analysis indicate substantially better control of the FDR, there are still some datasets for which the results are inconclusive or worse. Notably, this is particularly the case for the single cell (1cell-eclipse) data, where the lower bound on DIA-NN's precursor-level FDP data is above 2.3% and Spectronaut's FDP is above 3.8%. It is worth noting that DIA-NN's and Spectronaut's protein-level estimated FDPs are also highest on that particular data set. Supplementary Figures S8–S13 complement these observations by providing for all datasets and DIA-tools the lower bound, as well as the combined and paired estimated FDPs for a wide range of FDR thresholds. In particular, the figures consistently demonstrate how the entrapment estimation methods compare with one another, with the combined method reporting the largest estimated FDP, the lower bound the smallest, and the paired method in between the other two.

We also performed complementary experiments in which we varied the entrapement-to-original-target ratio r. Consistent with Lou *et al.* [28], we find that the estimated FDP among DIA-NN's reported proteins generally increases with r (Supplementary Figure S14–S15). On the other hand, Lou *et al.* only increased r up to $r \approx 1$ and concluded that DIA-NN controls the FDR (because they were incorrectly using the lower bound, Section S5). In contrast, we explored higher values of r and found that even the lower bound is consistently substantially higher than the corresponding FDR thresholds, indicating quite clearly that DIA-NN apparently fails to control the FDR in those scenarios. For example, for $r = 8$ we see a lower bound of almost 7% (Supplementary Figure S14). The same phenomenon was observed in the case of Spectronaut as well: for example, the lower bound is above 4% for $r = 6$ (Supplementary Figure S16). While these observations do not imply that the FDP is as high when searching only the original target database, they do indicate that the tool struggles to control the FDR in some setups. In general, proper FDR control should be applicable to any realistic scenario.

3 Discussion

Overall, our work identifies a lack of consensus in the field about entrapment estimation methods. Accordingly, we introduce a formal framework within which we can rigorously study entrapment estimation methods. This framework is equally applicable to PSM, peptide and protein level analyses: provided the assumptions that we specify hold, the corresponding upper or lower bound nature of the estimates is guaranteed (though we generally caution against trying to control the FDR at the PSM level). Moreover, our framework is also applicable to what Madej and Lam referred to as "entrapment query protocol" [32], where foreign spectra are used to expand the dataset in lieu of the common expansion through a peptide entrapment database.

Applying our entrapment procedures to DIA analysis tools we demonstrate that their FDR control at the peptide or precursor level can be questionable, and it is often questionable at the protein level. Given the importance of protein-level analysis to mass spectrometry experiments, we believe that this result should serve as a call to further research into this problem. Keep in mind that this is not just a matter of rigorous statistics: often the detected proteins are then further used to identify differentially expressed proteins, so incorrectly calling the detected proteins can adversely affect our ability to successfully accomplish our ultimate goal.

In general, an upper bound estimate like the combined estimation can only be used to show valid FDR control, and the lower bound to highlight invalid FDR control. However, if r is large then the difference between Equations (1) and (2) becomes negligible, so each of the methods can be reasonably used for making both arguments. That said, keep in mind that using $r \gg 1$ creates a much larger combined target database, most of which is made of entrapment sequences. Thus, establishing FDR control in this somewhat atypical setup is not as convincing as establishing it for smaller values of r (e.g., $r = 1, 2$). Of course, when using smaller values of r we have to take into account that Equation (2) is a lower bound, and to establish FDR control we need to use the upper bound. Our paired entrapment estimation method, Equation (4), and its k-matched generalization, Equation (S3), provide tighter upper bounds than the combined method, particularly for smaller values of r and larger fractions of the native peptides in the original target database (cf. Section 2.3).

As mentioned, the validity of an estimate hinges on whether its underlying assumptions are expected to hold, which in turn depends on how the entrapment procedure expands the input dataset. The most common expansion approach in analyzing MS/MS tools is to enlarge the database by adding to it foreign entrapment sequences. In this work, we chose instead to focus on shuffled entrapments because, as we argue in detail in Supplementary Note S4 and Supplementary Figures S1–S4, the use of foreign species raises complex questions associated with the choice of those species; for example, what is the "right" evolutionary distance for the entrapment species, and whether the entrapment species coincides with a potential source of contamination, as we found in our analysis of the commonly used HEK293 dataset. Such questions will need to be addressed before we can objectively agree on a way to use such foreign entrapment sequences.

Granted, the use of shuffled entrapment is not without its controversy. In particular, Madej and Lam recently argued that using randomly shuffled entrapment sequences to validate tools that rely on shuffled decoys to control the FDR amounts to circular reasoning [32]. While there is merit to this claim, there are many cases where this is not exactly the case.

First, these tools often use the decoy sequences differently than they are used by the entrapment estimation methods. For example, Percolator's cross-validation scheme to improve the ranking of the PSMs can inadvertently misuse the target/decoy label when multiple spectra are generated from the same peptide species [11]. Because in this case the compromised FDR control stems from indirectly peeking at the target/decoy label but not at the target/entrapment label, the problem can be identified even when both the entrapment sequences and the decoys are shuffled. Similarly, there is no issue with applying shuffled-based entrapments to procedures such as BH and Storey that use shuffled decoys to compute empirical p-values.

Second, if we believe for example that Assumption 2a holds, then the combined estimation method is valid regardless of which type of decoys the analysis tool uses to control the FDR. Note that the field has largely been comfortable with applying TDC that relies on an even stronger assumption than the latter: one that also requires the independence of false discoveries. As previously pointed out, shuffled sequences cannot account for errors due to homologs or more generally due to neighbors (i.e., distinct peptides with similar spectra) [10]. However, our analyses here suggest that the same applies when using foreign species such as *Arabidopsis*) in the analysis of a human sample.

Indeed, utilizing a mixed foreign and shuffled entrapment approach we find perfect agreement between the shuffled-based combined and paired estimates and the foreign-based one in the controlled ISB18 set up (Figure 2b). Similarly, we find little difference when comparing the applications of the combined and lower bound methods to both Tide's and Sage's peptide-level analyses of the HEK293 data using shuffled sequences and using foreign arabidopsis sequences (Tide: bottom row of Supplementary Figure S7; Sage: Figure 3 bottom left, and Supplementary Figure S1, right panel). The same holds when comparing foreign and shuffled entrapment estimation methods applied to MaxQuant's protein-level analysis (Supplementary Figure S6), as well as applied to non-TDC based procedures (Supplementary Figure S7, first 2 rows).

Moreover, the lower bound method only requires that the entrapment sequences are not present in the sample (Assumptions 1a and 1b in Supplementary Note S2.1). As long as this holds, using this estimate to conclude that a tool apparently fails to control the FDR involves no circular reasoning. In particular, our analysis of DIA tools, which mostly relies on the lower bound estimate, is valid even though we used shuffled sequences.

Regardless, it is clear that the topic of entrapment expansion merits further research. Indeed, considering Madej and Lam's entrapment setup, originally proposed in [15], where foreign spectra are used to expand the dataset in lieu of a peptide entrapment database, one can consider a shuffled-based rather than a foreign-based expansion, for example by applying the spectrum shuffling protocol of [23]. Future research could consider all four possible expansions: shuffled versus foreign, and database versus spectrum set, including possibly mixing the strategies as in our double entrapment analysis of the ISB18, as well as using more than one expansion method.

Finally, to facilitate future entrapment analyses, we have produced an open source software tool, FDR-Bench, that provides two main functions: (1) build entrapment databases using randomly shuffled target sequences or using sequences from foreign species, and (2) estimate the FDP using the lower bound, combined, and paired methods.

References

[1] R. F. Barber and Emmanuel J. Candès. Controlling the false discovery rate via knockoffs. *The Annals of Statistics*, 43(5):2055–2085, 2015.

[2] Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society Series B*, 57:289–300, 1995.

[3] J. A. Bubis, T. N. Arrey, E. Damoc, B. Delanghe, J. Slovakova, T. M. Sommer, H. Kagawa, P. Pichler, N. Rivron, K. Mechtler, et al. Challenging the Astral mass analyzer-up to 5300 proteins per single-cell at unseen quantitative accuracy to study cellular heterogeneity. *bioRxiv*, pages 2024–02, 2024.

[4] M. C. Chambers, B. Maclean, R. Burke, D. Amodei, D. L. Ruderman, S. Neumann, L. Gatto, B. Fischer, B. Pratt, J. Egertson, K. Hoff, D. Kessner, N. Tasman, N. Shulman, B. Frewen, T. A. Baker, M. Y. Brusniak, C. Paulse, D. Creasy, L. Flashner, K. Kani, C. Moulding, S. L. Seymour, L. M. Nuwaysir, B. Lefebvre, F. Kuhlmann, J. Roark, P. Rainer, S. Detlev, T. Hemenway, A. Huhmer, J. Langridge, B. Connolly, T. Chadick, K. Holly, J. Eckels, E. W. Deutsch, R. L. Moritz, J. E. Katz, D. B. Agus, M. J. MacCoss, D. L. Tabb, and P. Mallick. A cross-platform toolkit for mass spectrometry and proteomics. *Nature Biotechnology*, 30(10):918–920, 2012.

[5] J. Cox and M. Mann. MaxQuant enables high peptide identification rates, individualized p.p.b.-range mass accuracies and proteome-wide protein quantification. *Nature Biotechnology*, 26:1367–1372, 12 2008.

[6] V. Demichev, C. B. Messner, S. I. Vernardis, K. S. Lilley, and M. Ralser. DIA-NN: neural networks and interference correction enable deep proteome coverage in high throughput. *Nature Methods*, 17(1):41–44, 2020.

[7] B. Diament and W. S. Noble. Faster SEQUEST searching for peptide identification from tandem mass spectra. *Journal of Proteome Research*, 10(9):3871–3879, 2011.

[8] J. E. Elias and S. P. Gygi. Target-decoy search strategy for increased confidence in large-scale protein identifications by mass spectrometry. *Nature Methods*, 4(3):207–214, 2007.

[9] J. Freestone, L. Käll, W. S. Noble, and U. Keich. Semi-supervised learning while controlling the fdr with an application to tandem mass spectrometry analysis. *bioRxiv*, 2023. https://www.biorxiv.org/content/10.1101/2023.10.26.564068v3.

[10] Jack Freestone, William S Noble, and Uri Keich. Analysis of tandem mass spectrometry data with conga: Combining open and narrow searches with group-wise analysis. *Journal of Proteome Research*, 23(6):1894–1906, 2024.

[11] Jack Freestone, William Stafford Noble, and Uri Keich. Reinvestigating the correctness of decoy-based false discovery rate control in proteomics tandem mass spectrometry. *Journal of Proteome Research*, 23(6):1907–1914, 2024.

[12] S. Gessulat, T. Schmidt, D. P. Zolg, P. Samaras, K. Schnatbaum, J. Zerweck, T. Knaute, J. Rechenberger, B. Delanghed, A. Huhmer, U. Reimer, H. Ehrlich, S. Aiche, B. Kuster, and M. Wilhelm. Prosit: proteome-wide prediction of peptide tandem mass spectra by deep learning. *Nature Methods*, 16(6):509, 2019.

[13] V. Granholm, W. S. Noble, and L. Käll. On using samples of known protein content to assess the statistical calibration of scores assigned to peptide-spectrum matches in shotgun proteomics. *Journal of Proteome Research*, 10(5):2671–2678, 2011.

[14] K. He, Y. Fu, W.-F. Zeng, L. Luo, H. Chi, C. Liu, L.-Y. Qing, R.-X. Sun, and S.-M. He. A theoretical foundation of the target-decoy search strategy for false discovery rate control in proteomics. *arXiv*, 2015. https://arxiv.org/abs/1501.00537.

[15] K. Jeong, S. Kim, and N. Bandeira. False discovery rates in spectral identification. *BMC Bioinformatics*, 13(Suppl. 16):S2, 2012.

[16] L. Käll, J. D. Canterbury, J. Weston, W. S. Noble, and Michael J MacCoss. Semi-supervised learning for peptide identification from shotgun proteomics datasets. *Nature Methods*, 4(11):923–925, 2007.

[17] L. Käll, J. D. Storey, M. J. MacCoss, and W. S. Noble. Assigning significance to peptides identified by tandem mass spectrometry using decoy databases. *Journal of Proteome Research*, 7(1):29–34, 2008.

[18] A. Keller, A. I. Nesvizhskii, E. Kolker, and R. Aebersold. Empirical statistical model to estimate the accuracy of peptide identification made by MS/MS and database search. *Analytical Chemistry*, 74:5383–5392, 2002.

[19] A. Kertesz-Farkas, F. L. Nii Adoquaye Acquaye, K. Bhimani, J. K. Eng, W. E. Fondrie, C. Grant, M. R. Hoopmann, A. Lin, Y. Y. Lu, R. L. Moritz, M. J. MacCoss, and W. S. Noble. The Crux Toolkit for Analysis of Bottom-Up Tandem Mass Spectrometry Proteomics Data. *Journal of Proteome Research*, 22(2):561–569, Feb 2023.

[20] S. Kim and P. A. Pevzner. MS-GF+ makes progress towards a universal database search tool for proteomics. *Nature Communications*, 5:5277, 2014.

[21] J. Klimek, J. S. Eddes, L. Hohmann, J. Jackson, A. Peterson, S. Letarte, P. R. Gafken, J. E. Katz, P. Mallick, H. Lee, A. Schmidt, R. Ossola, J. K. Eng, R. Aebersold, and D. B. Martin. The standard protein mix database: a diverse data set to assist in the production of improved peptide and protein identification software tools. *Journal of Proteome Research*, 7(1):96–1003, 2008.

[22] A. T. Kong, F. V. Leprevost, D. M. Avtonomov, D. Mellacheruvu, and A. I. Nesvizhskii. MSFragger: ultrafast and comprehensive peptide identification in mass spectrometry-based proteomics. *Nature Methods*, 14(5):513–520, 2017.

[23] H. Lam, E. Deutsch, and R. Aebersold. Artificial decoy spectral libraries for false discovery rate estimation in spectral library searching in proteomics. *Journal of Proteome Research*, 9(1):605–610, 2010.

[24] N. M. Lancaster, P. Sinitcyn, P. Forny, T. M. Peters-Clarke, C. Fecher, A. J. Smith, E. Shishkova, T. N. Arrey, A. Pashkova, M. L. Robinson, et al. Fast and deep phosphoproteome analysis with the orbitrap astral mass spectrometer. *bioRxiv*, pages 2023–11, 2023.

[25] M. R. Lazear. Sage: An open-source tool for fast proteomics searching and quantification at scale. *Journal of Proteome Research*, 22(11):3652–3659, 2023.

[26] S. Lee, H. Park, and H. Kim. False discovery rate estimation using candidate peptides for each spectrum. *BMC Bioinformatics*, 23(1):454, 2022.

[27] A. Lin, T. Short, W. S. Noble, and U. Keich. Improving peptide-level mass spectrometry analysis via double competition. *Journal of Proteome Research*, 21(10):2412–2420, 2022.

[28] R. Lou, Y. Cao, S. Li, X. Lang, Y. Li, Y. Zhang, and W. Shui. Benchmarking commonly used software suites and analysis workflows for dia proteomics and phosphoproteomics. *Nature Communications*, 14(1):94, 2023.

[29] L. Lu, N. M. Riley, M. R. Shortreed, C. R. Bertozzi, and L. M. Smith. O-Pair search with MetaMorpheus for O-glycopeptide characterization. *Nature Methods*, 17(11):1133–1138, 2020.

[30] Y. Y. Lu, J. Bilmes, R. A. Rodriguez-Mias, J. Villén, and W. S. Noble. DIAmeter: Matching peptides to data-independent acquisition mass spectrometry data. *Bioinformatics*, 37(Suppl 1):i434–i442, 2021.

[31] Jie Ma, Jiyang Zhang, Songfeng Wu, Dong Li, Yunping Zhu, and Fuchu He. Improving the sensitivity of mascot search results validation by combining new features with bayesian nonparametric model. *Proteomics*, 10(23):4293–4300, 2010.

[32] D. Madej and H. Lam. On the use of tandem mass spectra acquired from samples of evolutionarily distant organisms to validate methods for false discovery rate estimation. *Proteomics*, page 2300398, 2024.

[33] S. Na, H. Choi, and E. Paek. Deephos: predicted spectral database search for TMT-labeled phosphopeptides and its false discovery rate estimation. *Bioinformatics*, 38(11):2980–2987, 2022.

[34] Y. Nowatzky, P. Benner, K. Reinert, and T. Muth. Mistle: bringing spectral library predictions to metaproteomics with an efficient search index. *Bioinformatics*, 39(6):btad376, 2023.

[35] R. Peckner, S. A. Meyers, J. D. Egertson, R. S. Johnson, J. G. Abelin, S. A. Carr, M. J. MacCoss, and J. D. Jaffe. Specter: linear deconvolution as a new paradigm for targeted analysis of data-independent acquisition mass spectrometry proteomics. *Nature Methods*, 15(5):371–378, 2018.

[36] J. Penny, M. Arefian, G. N. Schroeder, J. A. Bengoechea, and B. C. Collins. A gas phase fractionation acquisition scheme integrating ion mobility for rapid diaPASEF library generation. *Proteomics*, 23(7-8):2200038, 2023.

[37] M. Picciani, W. Gabriel, V.-G. Giurcoiu, O. Shouman, F. Hamood, L. Lautenbacher, C. B. Jensen, J. Müller, M. Kalhor, A. Soleymaniniya, et al. Oktoberfest: Open-source spectral library generation and rescoring pipeline based on Prosit. *Proteomics*, page 2300112, 2023.

[38] A. M Scott, C. Karlsson, T. Mohanty, E. Hartman, S. T. Vaara, A. Linder, J. Malmström, and L. Malmström. Generalized precursor prediction boosts identification rates and accuracy in mass spectrometry based proteomics. *Communications Biology*, 6(1):628, 2023.

[39] B. C. Searl, K. E. Swearingen, C. A. Barnes, T. Schmidt, S. Gessulat, B. Küster, and M. Wilhelm. Generating high quality libraries for dia ms with empirically corrected peptide predictions. *Nature Communications*, 11(1):1–10, 2020.

[40] B. C. Searle, L. K. Pino, J. D. Egertson, Y. S. Tin, R. T. Lawrence, B. X. MacLean, J. Vill'en, and M. J. MacCoss. Chromatogram libraries improve peptide detection and quantification by data independent acquisition mass spectrometry. *Nature Communications*, 9:5128, 2018.

[41] P. Sinitcyn, H. Hamzeiy, F. Salinas Soto, D. Itzhak, F. McCarthy, C. Wichmann, M. Steger, U. Ohmayer, U. Distler, S. Kaspar-Schoenefeld, et al. MaxDIA enables library-based and library-free data-independent acquisition proteomics. *Nature Biotechnology*, 39(12):1563–1573, 2021.

[42] J. D. Storey. A direct approach to false discovery rates. *Journal of the Royal Statistical Society Series B*, 64:479–498, 2002.

[43] M. T. Strauss, I. Bludau, W.-F. Zeng, E. Voytik, C. Ammar, J. P. Schessner, R. Ilango, M. Gill, F. Meier, S. Willems, et al. AlphaPept: a modern and open framework for MS-based proteomics. *Nature Communications*, 15(1):2168, 2024.

[44] P. Sulimov and A. Kertész-Farkas. Tailor: A nonparametric and rapid score calibration method for database search-based peptide identification in shotgun proteomics. *Journal of Proteome Research*, 19(4):1481–1490, 2020.

[45] M. The, P. Samaras, B. Kuster, and M. Wilhelm. Reanalysis of ProteomicsDB using an accurate, sensitive, and scalable false discovery rate estimation approach for protein groups. *Molecular and Cellular Proteomics*, 21(12):100437, Dec 2022.

[46] K. L. Yang, F. Yu, G. C. Teo, K. Li, V. Demichev, M. Ralser, and A. I. Nesvizhskii. MSBooster: improving peptide identification rates using deep learning-based features. *Nature Communications*, 14(1):4539, 2023.

[47] Z. Ye, P. Sabatier, L. Hoeven, T. Phlairaharn, D. Hartlmayr, F. Izaguirre, A. Seth, H. Joshi, D. Bekker-Jensen, N. Bache, et al. High-throughput and scalable single cell proteomics identifies over 5000 proteins per cell. *bioRxiv*, pages 2023–11, 2023.

[48] F. Yu, G. C. Teo, A. T. Kong, K. Fröhlich, G. X. Li, V. Demichev, and A. I. Nesvizhskii. Analysis of DIA proteomics data using MSFragger-DIA and FragPipe computational platform. *Nature Communications*, 14(1):4154, 2023.

[49] Q. Zhang. Mzion enables deep and precise identification of peptides in data-dependent acquisition proteomics. *Scientific Reports*, 13(1):7056, 2023.

4 Methods

4.1 TDC protocols for controlling the FDR

4.1.1 TDC

Initially introduced by Elias and Gygi [8], the following variation of the original TDC procedure was subsequently proved to control the FDR [14]: the input is a list of pairs (W_i, L_i), where $L_i = +1$ for a target PSM, peptide, or protein, $L_i = -1$ for a decoy, and W_i is its corresponding score.

The pairs are ordered in decreasing order of their score W_i, and for each k we find D_k, the number of decoys in the top k pairs, and $T_k = k - D_k$ the number of targets. Given the desired FDR level α, TDC sets the rejection/discovery index at

$$K = K(\alpha) := \max\{k : \frac{D_k + 1}{\max\{T_k, 1\}} \leq \alpha\}. \tag{5}$$

Finally, TDC reports all targets among the top K pairs.

Assuming that a pair corresponding to an incorrect discovery is at least as likely to be a decoy as it is to be a target, i.e., that $P(L_i = -1) \geq 1/2$ independently of everything else (including of the score W_i), TDC rigorously controls the FDR [1, 14] among its reported discoveries.

4.1.2 Peptide-level TDC with PSM-and-peptide

PSM-and-peptide is a TDC-based procedure for peptide-level analysis introduced in [27]. This method involves a double competition: the first is a PSM-level competition where each spectrum is searched against the concatenated target-decoy database and the best matching peptide is assigned to it (where ties are broken randomly). This defines the (optimal) PSM associated with each spectrum, and then each target or decoy peptide is assigned a score, which is the maximum of the scores of all PSMs which this peptide is part of. PSM-and-peptide introduces a second level of competition to define the scores and labels by utilizing the pairing between each target and its shuffled decoy. Specifically, it keeps only the higher scoring peptide from each target-decoy pair. Any peptide with no matching PSM is assigned the lowest possible score, and all ties are randomly broken. This procedure defines the pair's winning score W_i and its label $L_i = \pm 1$ indicating whether the higher scoring peptide was the target or the decoy. TDC is then applied as above to this list of scores and labels (W, L).

4.2 Controlling the FDR using the BH and Storey

Our peptide-level FDR control using the BH and Storey's procedures began with a Tide target-decoy search without competition. For each target (an original or entrapment target) or decoy peptide, the highest scoring PSM was selected based on the Tailor score [44], and the peptide score was then defined as the score of the selected PSM. Next, a p-value was assigned to each target peptide score and was defined as the proportion of decoy peptides with a score as high as the assigned one. For FDR control using BH, the p-values of peptides were adjusted using the function `p.adjust` with the method parameter set as `BH` in R. The adjusted p-values were then taken as FDR thresholds for downstream analysis. For FDR control using Storey's procedure, the function `qvalue` from the R package `qvalue` (version 2.34) was used with its default parameters. The returned q-values were then taken as FDR thresholds for downstream analysis.

4.3 Datasets

The MS/MS data used in this study include a wide range of datasets from different vendors, different MS instruments, data acquisition strategies, and species. As shown in Table 3, a total of 12 datasets were used in the study, including two DDA datasets and 10 DIA datasets. The raw data were downloaded from PRIDE, MassIVE and jPOST. The raw data were converted to mgf or mzML format files using MSConvert in ProteoWizard (version 3.0.24031) [4]. This excludes the ISB18 data, where ms2 files were obtained from [27]. Among the 10 DIA datasets, four include gas phase fractionation DIA runs, which were used to build chromatogram libraries for EncyclopeDIA analysis. Two of the datasets were from a previous single-cell proteomics study (Dataset ID: PXD023325).

Dataset ID	Dataset name	MS instrument	MS runs	Species	Data type
ISB18	ISB18	LTQ Orbitrap	9	various	DDA
PXD001468	HEK293	Q-Exactive	24	human	DDA
PXD042704	human-astral	Orbitrap Astral	1	human	DIA
PXD034525	human-lumos	Orbitrap Fusion Lumos	16, 6 (GPF)	human	DIA
PXD028735	human-tripletof	TripleTOF 5600	3, 8 (GPF)	human	DIA
PXD028735	human-qe	Q Exactive HF-X	3, 6 (GPF)	human	DIA
PXD041421	human-timstof1	timsTOF Pro	4	human	DIA
PXD017703	human-timstof2	timsTOF Pro	3	human	DIA
PXD023325	100cell-eclipse	Orbitrap Eclipse	3	human	DIA
PXD023325	1cell-eclipse	Orbitrap Eclipse	3	human	DIA
PXD012988	mouse-qe	Q Exactive HF	15	mouse	DIA
MSV000084000	yeast-lumos	Orbitrap Fusion Lumos	3, 6 (GPF)	yeast	DIA

Table 3: **Summary of datasets used in the study.** GPF: Gas-phase fractionation DIA.

4.4 Entrapment database generation using random sequences

The shuffled entrapment databases were generated differently for precursor-level and peptide-level FDR control evaluation than they were for protein-level analysis. In both cases, the original target protein sequences for human (UP000005640, 20597 proteins), yeast (UP000002311, 6060 proteins) and mouse (UP000000589, 21701 proteins) were downloaded from UniProt (02/2024).

For precursor/peptide-level analysis, the original target proteins were first *in silico* digested into peptides using trypsin (without proline suppression) with one missed cleavage allowed. The original target peptides database consisted of those with lengths between 7 and 35 amino acids. Then for each original target peptide, we attempted to generate a paired random entrapment peptide as follows. Specifically, the original peptide was shuffled while keeping the C-terminal amino acid fixed and then searched against all original target peptides as well as the previously generated random peptides to ensure it is distinct from all of those. If it was not, we retried to generate such a distinct shuffled peptide up to an additional 20 times. If all those attempts failed we removed the corresponding peptide from the original target database. To generate r matching random entrapment peptides we repeated this shuffling process up to $20 + r$ times to try and obtain r distinct entrapment peptides for each original target peptide. A peptide for which we failed to generate r distinct shuffles after $20 + r$ attempts was removed from the original target database. Supplementary Algorithm 1 summarizes this procedure in pseudocode.

In the protein-level FDR evaluation, for each original target protein we generated a paired random entrapment protein as follows. We first *in silico* digested the original protein into peptides using trypsin (without proline suppression). In this step, all the peptides, irrespective of length and mass constraints, were retained and no missed cleavages were considered. Then, for each of these peptides we tried to generate a distinct randomly shuffled entrapment peptide (again, while fixing the C-terminal) as above. This was tried up to 20 additional times for each original peptide and if all failed to generate a distinct peptide then (in contrast to the peptide-level analysis) the entrapment peptide associated with the target peptide was identical to the target. Finally, each original peptide in the considered protein was swapped with its randomly generated one. Note that a peptide that appears in multiple proteins or multiple times within a protein was consistently swapped with its uniquely associated paired entrapment peptide in all the proteins it appears. To create an r-fold entrapment protein database, we attempted to associate with each digested original target peptide r distinct randomly shuffled entrapment peptides as above. If we failed to do so in $20 + r$ attempted shuffles then we randomly sampled with replacement $r - n$ additional entrapment peptides from the $n > 0$ distinct shuffles that we managed to generate. If there were no distinct shuffles at all ($n = 0$) then the selected r entrapment peptides were all identical to the original digested peptides. We then used these r entrapment peptides to define r associated entrapment proteins as described above for $r = 1$.

4.5 Entrapment database generation using sequences from foreign species

To generate protein-level entrapment databases using proteins from foreign species, a set of proteins from the selected foreign species were randomly selected as entrapment proteins to achieve the desired ratio of r

entrapment-to-original target proteins (we used $r \geq 1$).

To generate peptide-level entrapment databases, both original target proteins and the proteins from the foreign species were *in silico* digested into peptides using trypsin (without proline suppression) with one missed cleavage allowed. Again, we only considered digested target and entrapment peptides with length between 7 and 35 amino acids. Any foreign species peptides that matched an original target peptide were removed. Finally, we randomly selected as many of the remaining foreign peptides as needed to achieve the desired ratio of r entrapment-to-(remaining)-original target peptides (we used $r \geq 1$). Supplementary Algorithm 2 summarizes this procedure in pseudocode.

Two sets of foreign species were used in this study. The first set of foreign species consisted of *Arabidopsis thaliana* and *Saccharomyces cerevisiae*. The protein sequences from these two species were downloaded from UniProt (02/2024). Specifically, 27448 proteins from *Arabidopsis thaliana* (UP000006548) and 6060 proteins from *Saccharomyces cerevisiae* (UP000002311) were used. The second set of foreign species consisted of *Macaca mulatta*, *Callithrix jacchus*, *Papio anubis* and *Mus musculus*. The protein sequences from these species were downloaded from UniProt (primates: 04/2024, mouse: 02/2024). Specifically, 21590 proteins from *Papio anubis* (UP000028761), 21893 proteins from *Macaca mulatta* (UP000006718), 22027 proteins from *Callithrix jacchus* (UP000008225) and 21701 proteins from *Mus musculus* (UP000000589) were used.

4.6 FDR control evaluation procedures

4.6.1 Using the ISB18 controlled experiment data (with Tide)

We used DDA spectra from an 18-protein mixture, called the ISB18, acquired from a controlled experiment [21]. We implemented our entrapment methods using two different setups: (a) randomly shuffled entrapment sequences and (b) randomly shuffled entrapment sequences in the presence of foreign target sequences from the castor proteome. We used the 9 ms2 spectrum files and the castor proteome from [27] and directly downloaded the in-sample protein database `20130710-ISB18-extended.fasta` from `https://regis-web.systemsbiology.net/PublicDatasets/database`.

Here we used the following variant of the shuffling entrapment protocol described in Section 4.4. For approach (a) we created 100 randomly shuffled databases by digesting the 18-protein database using the tide-index command (default settings) from a recent version of Crux (v4.1.6809338) [19] and randomly shuffling the resulting original target peptides $\mathcal{T}$ while fixing both the C-terminal and the N-terminal amino acids in place. We implemented a narrow search of the combined spectrum files against each of the 100 combined target+shuffled entrapment databases using tide-search (using the automatic fragment and precursor tolerance selection). Next, each entrapment method was used by considering 3 of the narrow search files, designating 1 of the randomly shuffled databases as $\mathcal{E}_{\mathcal{T}}$ and the remaining 2 randomly shuffled databases as the decoys for $\mathcal{T} \cup \mathcal{E}_{\mathcal{T}}$. A small number of randomly shuffled peptides were problematic because they appeared in two different sets: the decoy set of $\mathcal{T}$, $\mathcal{E}_{\mathcal{T}}$, or the decoy set of $\mathcal{E}_{\mathcal{T}}$, and some low complex target peptides were unable to produce 3 distinct random shuffles. Hence, any target peptide and their corresponding shuffled peptides that contained such a problematic peptide were removed from the search files. Next, we joined the 3 search files and implemented peptide-level analysis using the PSM-and-peptide protocol with XCorr scores. Finally, we estimated the FDP using each of the entrapment methods. Because we have 100 randomly shuffled databases, we repeated the above analysis 100 times using a different choice of the 3 narrow search files, ensuring that each narrow search file is considered exactly 3 times in total. We then estimated the FDR by taking the average of our 100 FDP estimates. The 95% coverage bands that account for the decoy and entrapment sequence variability were computed for each FDR threshold as $\pm 1.96 \cdot \sigma_n/\sqrt{n}$, where σ_n is the standard deviation of the $n = 100$ estimated FDPs at that threshold.

For approach (b) we prepared a new "original target" peptide database, $\mathcal{T}$, by combining the target sequences digested from the ISB18 protein mixture and the foreign sequences digested from the castor proteome. We then followed the same steps in (a) to obtain 100 FDP estimations that were averaged to obtain an FDR estimate using the paired and sample entrapment methods. We also obtained 100 "direct FDP" estimates, which rely on the number of discovered shuffled entrapment sequences and castor peptides to estimate the number of false discoveries. Specifically, the direct estimate is the ratio of the number of castor and shuffled entrapment discoveries over the total number of (ISB18+castor+shuffled entrapment) discoveries. These FDP estimates were also averaged to obtain a "direct FDR" estimate.

4.6.2 Tide

To evaluate the FDR control in Tide (within Crux v4.1.6809338) [7] in a more typical setting, the HEK293 DDA MS/MS data was analyzed using Tide. Specifically, a peptide-level paired entrapment database was first generated using the method described in Section 4.4. Tide-index from the Crux toolkit (https://crux.ms/) was then used to randomly shuffle both the original target peptides and entrapment peptides in the peptide-level entrapment database to obtain decoys (while keeping the C-terminal amino acid fixed) and build an index later used for tide-search. Carbamidomethylation of cysteine was set as a fixed modification and no variable modification was used in this step. Next, the HEK293 MS/MS data was searched against the combined database generated in the previous step using tide-search from the Crux toolkit with the following parameters: tailor-calibration, enabled; Sp scoring, enabled; precursor ion mass tolerance, 20 ppm; fragment tolerance, 0.02 m/z; fragment offset, 0. All other Tide parameters were set as default. FDR control of the Tide search result at the peptide level was performed using the single-decoy Percolator-RESET (version 0.0.6) [9] with specifying the Tailor score [44] as the primary score (all other parameters were set as default).

4.6.3 Sage

To evaluate the peptide-level FDR control in Sage (version 0.14.6) [25], we generated the same target+shuffled entrapment database as described above for Tide (Section 4.6.2). In addition, to demonstrate the evolutionary distance problem with foreign entrapment, we also used Sage with foreign peptide entrapments as described in Section 4.5. In both cases the HEK293 dataset was searched against the target+entrapment database using Sage with the following parameters: fixed modification, carbamidomethyl (C); no variable modifications; precursor ion mass tolerance, 20 ppm; fragment ion tolerance, 20 ppm; enzyme digestion was disabled; peptide length range, 7–35; isotope error range was set to "[0,0]". All other parameters were set as default. Sage's built-in FDR control procedure was used. The "peptide_q" from Sage's output was used as the peptide q-value for downstream analysis.

4.6.4 MS-GF+

To evaluate the FDR control in MS-GF+ (version 2023.01.12) [20], a peptide-level entrapment database was used that contained sample peptides, paired entrapment peptides and their paired decoy peptides. The peptide-level entrapment database was generated using the method described in Section 4.4. Specifically, in generating the peptide database, three different random peptides were generated for each target peptide, one decoy peptide was taken as paired entrapment peptide for the target while the other two decoy peptides were taken as decoy peptides. The HEK293 dataset was searched against the entrapment database using the following parameters: fixed modification, Carbamidomethyl (C); no variable modifications; precursor ion mass tolerance, 20 ppm; range of allowed isotope peak errors, "0,0"; peptide length range, 7 - 35; instrument ID, 3 (Q-Exactive); fragmentation method, 3 (HCD); protocol ID, 5 (Standard); n-terminal methionine cleavage was disabled. No enzyme digestion was applied. The parameter "-tda" was set to 0 to allow using the decoy peptides contained in the peptide database. All other parameters were set as default. The "PepQValue" from MS-GF+'s output was used as the peptide q-value for downstream analysis.

4.6.5 FragPipe

To evaluate peptide-level FDR control of the FragPipe pipeline (version 21.1) [22, 46, 48] on DDA data, the HEK293 dataset was searched against the same peptide entrapment database used in the MS-GF+ peptide-level FDR control analysis. The "Default" workflow setting in FragPipe was used with several parameters changed as follows. The enzyme digestion was configured to disable *in-silico* digestion. The setting of "Clip N-term M" was disabled. No variable modification was used. The isotope error was set to zero. The calibration and optimization setting was disabled. The "pin" files generated by MSBooster [46] were combined and used for Percolator (version 3.6.4) [16] analysis. The peptide-level result from Percolator was then used for downstream analysis.

4.6.6 MaxQuant

To evaluate protein-level FDR control of MaxQuant (version 2.6.5.0) [5] on DDA data, the raw files of the HEK293 dataset were searched against two protein-level entrapment databases separately using the following parameters: fixed modification, carbamidomethyl (C); variable modifications, oxidation (M); the default enzyme "Trypsin/P" was used with a maximum of one missed cleavage site allowed; the setting of "Include contaminants" was disabled; protein-level FDR threshold was set as 0.1. All other parameters were set to their default values. The first entrapment database was generated using the random shuffling method described in Section 4.4 with $r = 1$. The second entrapment database was generated using the method described in Section 4.5, in which the proteins from *Arabidopsis thaliana* were used as entrapment proteins with $r = 1$. The protein files "proteinGroups.txt" generated by MaxQuant were used for downstream analysis. The "Q-value" from the files was used as protein q-value for downstream analysis.

4.6.7 DIA-NN

For precursor-level FDR control evaluation of DIA-NN (version 1.8.1) [6], a peptide-level entrapment database was used that contained the original target and their paired entrapment peptides for each DIA dataset. Each peptide-level entrapment database was generated using the method described in Section 4.4. DIA-NN analysis was performed using the following parameters: fixed modification, carbamidomethyl (C); no variable modifications; enzyme digestion was disabled; peptide length range, 7–35; precursor charge range, 2–4. The setting of "N-term M excision" was disabled. The precursor FDR was set to 10%. All other parameters were set to their default values. For single run DIA data, the "Q.Value" from the main report was used as precursor q-value for downstream analysis. For datasets with multiple runs, the "Lib.Q.Value" from the main report was used as precursor q-value for downstream analysis.

For protein-level FDR control evaluation, we ran DIA-NN in its library-free mode using an entrapment database as described in Section 4.4. The enzyme and peptide length settings were the same as peptide-level entrapment database generation in the precursor-level FDR control evaluation. The precursor FDR threshold was set to 1%. All other parameters were set as the same with the precursor-level analysis. For single run DIA data, the "PG.Q.Value" from the main report was used as protein q-value for downstream analysis. For datasets with multiple runs, the "Lib.PG.Q.Value" from the main report was used as protein q-value for downstream analysis. If a protein group included ≥ 2 proteins and at least one of them was from the original target database, it was taken as an original target protein group.

4.6.8 EncyclopeDIA

We evaluated both the peptide-level and the protein-level FDR control of the gas phase fractionated (GPF) chromatogram library analysis workflow with EncyclopeDIA (version 2.12.30) [39, 40].

We first used Oktoberfest (version 0.6.2) with Prosit models (fragment ion intensity prediction model: Prosit_2020_intensity_HCD, retention time model: Prosit_2019_irt) [12, 37] to generate two *in silico* spectral libraries for each DIA dataset that were later used for EncyclopeDIA analysis. In the spectral library generation step, carbamidomethyl of cysteine was considered as a fixed modification, and no variable modifications were considered. Precursor charges 2 to 4 were considered. The normalized collision energy (NCE) parameter was set to 27. The first spectral library was used for peptide-level FDR control analysis in which the input to Oktoberfest for library generation was a peptide database in csv format containing the target peptides and their paired entrapment peptides as described in Section 4.4. The second spectral library was used for protein-level FDR control analysis in which the input to Oktoberfest for library generation was a protein database containing the target proteins and their paired entrapment proteins as described in Section 4.4. For the second library generation for each DIA dataset, trypsin (without proline suppression) with one missed cleavage allowed was used and only peptides with lengths between 7 and 35 amino acids were considered in Oktoberfest.

Next, for each FDR control evaluation analysis (peptide level or protein level), we generated a new spectral library by searching a set of GPF library DIA runs against the corresponding *in silico* spectral library using EncyclopeDIA.

Finally, we searched quant DIA runs against the GPF-derived spectral library in each FDR control evaluation analysis. In the analysis, the V2 scoring of EncyclopeDIA was enabled except for the TripleTOF

5600 dataset (PXD028735, human-tripletof). For protein-level FDR evaluation, the protein FDR threshold in the quant DIA analysis was set to 10% and the peptide FDR threshold was set to 1%. If a protein group included ≥ 2 proteins and at least one of them was from the original target database, it was taken as an original target protein group. For peptide-level FDR evaluation, the protein FDR threshold in the quant DIA analysis was set to 1% and the peptide FDR threshold was set to 10%. The EncyclopeDIA analysis was run through the nf-skyline-dia-ms workflow (https://nf-skyline-dia-ms.readthedocs.io).

4.6.9 Spectronaut

We evaluated both the precursor-level and the protein-level FDR control of the library-free analysis workflow (directDIA) in Spectronaut (version 18.7.240325.55695).

For evaluating the protein-level FDR control we constructed a protein database containing the original target proteins and their paired entrapment proteins as described in Section 4.4. We used Spectronaut with the following settings: enzyme specificity, trypsin (without proline suppression); maximum missed cleavages, 1; peptide length range, 7 - 35; toggle N-terminal M, disabled. All other parameters were set as default. We used the "PG.Qvalue" from Spectronaut output as the protein q-value for downstream analysis. If a protein group included ≥ 2 proteins and at least one of them was from the original target database, it was taken as an original target protein group.

For evaluating the precursor-level FDR control we constructed a peptide database containing the target peptides and their paired entrapment peptides as described in Section 4.4. Because Spectronaut only estimates precursor-level FDR for each run, this analysis was done using only one MS run from each DIA dataset. No variable modification was set and no enzyme digestion was applied. The peptide length range was set from 7 to 35; toggle N-terminal M was disabled. In addition, the precursor PEP cutoff, protein q-value cutoff (experiment and run), protein PEP cutoff were set to 0.99. All other Spectronaut parameters were set as default. We used the "EG.Qvalue" from Spectronaut output was used as the precursor q-value for downstream analysis.

Code availability We implemented the entrapment database generation methods as well as the different FDP estimation methods in a Java tool called FDRBench. The source code is available with an Apache license at `https://github.com/Noble-Lab/FDRBench`.

Acknowledgments This work was supported by National Science Foundation award 2245300 and National Institutes of Health award R24 GM141156.

Conflict of interest The MacCoss Lab at the University of Washington receives funding from Agilent, Bruker, Sciex, Shimadzu, Thermo Fisher Scientific, and Waters to support the development of Skyline, a quantitative analysis software tool. MJM is a paid consultant for Thermo Fisher Scientific.