Biostatistics (2018), **0**, 0, *pp.* 1–38
doi:10.1093/biostatistics/revised-suppmat-lrt

SUPPLEMENTARY MATERIAL TO PEPA test: fast and powerful differential analysis from relative quantitative proteomics data using shared peptides

LAURENT JACOB*[,1], FLORENCE COMBES[2], THOMAS BURGER[2]

[1] *Université de Lyon; Université Lyon 1; CNRS; UMR 5558, Laboratoire de Biométrie et Biologie Évolutive, Lyon, France*

[2] *BIG-BGE (Université Grenoble Alpes, CNRS, CEA, INSERM), 38000 Grenoble, France*

laurent.jacob@univ-lyon1.fr

Summary

Section A describes the state of the art for differential abundant protein detection. Section B describes additional results on a dataset of mouse livers, illustrating the influence of shared peptides on differential analysis. Section C discusses our linear approximation of the log-normal model. Section D introduces a fast heuristic for our testing procedure. Sections E and F contain proofs of Propositions 1 and 2 respectively. Section G provides details about the simulation protocol and spike-in data used in the main manuscript. Finally, we show plots obtained using different hyperparameters than the ones used in the main manuscript in Section H.

Key words: Supplementary material to "More powerful differential analysis of relative quantitative proteomics", by Jacob *and others*.

*To whom correspondence should be addressed.

A. State-of-the-art

Leaving aside the various preprocessing steps that are necessary to account for *e.g.* batch effects or missing values (see for instance Wieczorek *and others* (2017)), methods for differential analysis of proteomic datasets can be divided in two main families: *peptide-based* and *aggregation-based* methods, also referred to as summarization-based in Goeminne *and others* (2015). In the latter ones, peptide-level information is first aggregated at the protein level and proteins are then tested for differential abundance using these summaries. Peptide-based models on the other hand do not rely on an aggregation step and build a test statistic using peptide intensities as a sampling unit.

A.1 *Aggregation-based models*

A.1.1 *Overview* Although both families allow to rank proteins according to their significance, aggregation-based methods are much more widely used on proteomics platforms than peptide-based models, as protein level abundance values make more sense to many practitioners and are easier to interpret.

The most commonly used aggregation methods avoid the issue of shared peptides by only considering protein-specific ones: either all of them, so as to involve as much information as possible in the process, or only the most abundant ones (Silva *and others*, 2006), which are best identified and least error-prone. Alternatively, all peptides can be retained for each protein whether they are shared or not. The intensities of the retained peptides are then summed or averaged, sometimes after being weighted by protein-level information such as the coverage of the protein by MS-observable peptides (Schwanhäusser *and others*, 2011). We compare these approaches in Section A.1.2 and show that summing or averaging all protein-specific peptides provides the best results.

More sophisticated aggregation methods have been proposed to account as precisely as possi-

ble for shared peptides. Dost *and others* (2012) split their intensities between their parent proteins by recasting the problem into a resource allocation framework, for which efficient optimization techniques are available. To the best of our knowledge (as the precise algorithm is not published and the code no longer available), it provides best results when an MS-detectability coefficient for each peptide is specified. In practice, such coefficients are scarcely known, limiting the applicability of the method to routinely analyzed and well-characterized proteomes. SCAMPI (Gerster *and others*, 2014) relies on a linear model that accounts for peptide-protein relationship like our method. However, it was designed to quantify protein abundances in single sample experiments by means of isotope labelling, and it does not generalize to joint estimation across several samples or hypothesis testing, as precisely illustrated in Section A.1.3.

After the aggregation step, a test for differential abundance is performed at the protein level. The most widely used procedure is the Student t-test, as well as its regularized versions SAM (Tusher *and others*, 2001) and Limma (Smyth, 2005). We present an empirical study of these different test statistics in the context of protein differential abundance analysis in Section A.1.4.

A.1.2 *Preliminary comparisons: aggregation step* Aggregation methods can be differenciated according to two criteria: first, the involved operator (sum or mean of peptide intensities, possibly followed by normalization according to each protein properties, see for instance Silva *and others* (2006)) and second, the set of retained peptides. The first one is an important question in absolute quantitative proteomics but has little influence in relative quantification: any difference between these operators equally applies to both compared conditions, so that at protein level, the significance of the differential abundance is not affected.

With regard to the second point, it appears (see Figure 1) that using all peptides indistinctly (*i.e.* both specific and shared peptides are considered as if they were protein-specific) leads to less accurate differential analysis than only relying on specific ones, as shared peptides generally

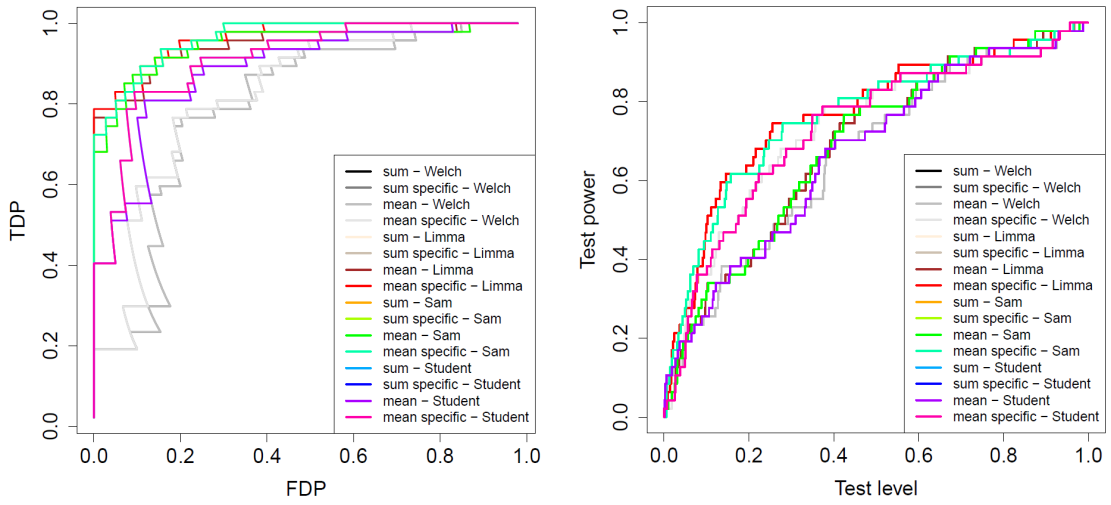

Fig. 1. Performance plots of the various aggregation methods on a real (left) and simulated (right) dataset. Curves related to sum-based aggregation are systematically hidden by curves related to mean-based aggregation, for they lead to exactly equal results. Whatever the statistical test, using all peptides rather than only the protein-specific ones leads to a decrement of the performances.

leads to protein abundance overestimation. Similarly, using only the most abundant peptides is less efficient as it leads to a loss of information. This last point is more thoroughly discussed in the experimental section of the main manuscript.

A.1.3 *Preliminary comparisons: SCAMPI* SCAMPI (Gerster *and others*, 2014) was initially proposed for absolute quantification experiments with isotope labelling, to estimate the ratio of the labelled protein over the original one for each protein within a given sample. The latter can then be inferred on the basis of the known concentration of the former, that has been artificially introduced in the sample. It is thus tempting to apply SCAMPI statistical framework to relative quantification. To do so, SCAMPI authors suggest "*running SCAMPI on each replicate/condition separately*". However, our experiments showed this procedure is not accurate. To explain this, we have considered a dummy dataset where the same sample was replicated 6 times, so as to mimic a perfect experiment with no instrumental variability. We have then applied SCAMPI to each replicate separately as suggested. We expected equal protein abundance estimates, but obtained

rather different results as illustrated on Figure 2: although within-sample differences of abundance seem to be respected (as for isotope/original protein abundance ratios), between-sample protein abundances are not.

A.1.4 *Preliminary comparisons: testing procedures* As with aggregation, several methods are reported in the proteomics literature to select putative differentially abundant proteins. The oldest one is based on selecting the proteins with the greatest fold-change (Ting *and others*, 2009), that is the absolute value of the difference (between the conditions) of mean log-transformed abundances. Although practically efficient, this method is nowadays hardly used, as it does not allow to control the false discovery rate associated to the set of selected proteins. Statistical tests are now classically considered, and followed by multiple test corrections. While specific tests are required for spectral count data (such as for instance the Beta-Binomial test Pham *and others* (2010)), extracted ion chromatogram data fit well the assumptions underlying the t-test. Several variations are classically considered:

- The original Student t-test and its Welch generalization to conditions with different numbers of replicates;
- SAM (Tusher *and others*, 2001), where the variance estimate is regularized by a fudge factor;
- Limma (Smyth, 2005), where the variance estimates are shrinked across proteins.

We report the results of an experiment comparing these procedures in Figure 1.

Student outperforms Welch The Welch t-test is theoretically of interest to process datasets with missing values. In label-free proteomics experiments however, there are often too few replicates per condition to deal with missing values, and imputation must be conducted first (Lazar *and others*, 2016), so that the interest of Welch t-test is disputable. In our

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]
0	24.81108	24.81108	24.81108	24.81108	24.81108	24.81108
1	24.47067	24.47067	24.47067	24.47067	24.47067	24.47067
2	24.37186	24.37186	24.37186	24.37186	24.37186	24.37186
3	19.97561	19.97561	19.97561	19.97561	19.97561	19.97561
4	24.11815	24.11815	24.11815	24.11815	24.11815	24.11815
5	24.49320	24.49320	24.49320	24.49320	24.49320	24.49320

	result.1	result.2	result.3	result.4	result.5	result.6
[1,]	8.957604	0.3636305	8.380070	8.188959	0.63055545	0.7002375
[2,]	9.507808	0.8342064	8.934568	8.738239	1.11035233	1.1746180
[3,]	8.186841	-0.1993891	7.605187	7.440010	0.05581622	0.1323902
[4,]	10.688386	1.5948444	10.102379	9.874593	1.88450746	1.9391035
[5,]	10.983744	1.7877183	10.396137	10.157562	2.08111735	2.1332778
[6,]	9.971229	1.1698344	9.394426	9.193354	1.45175959	1.5119261

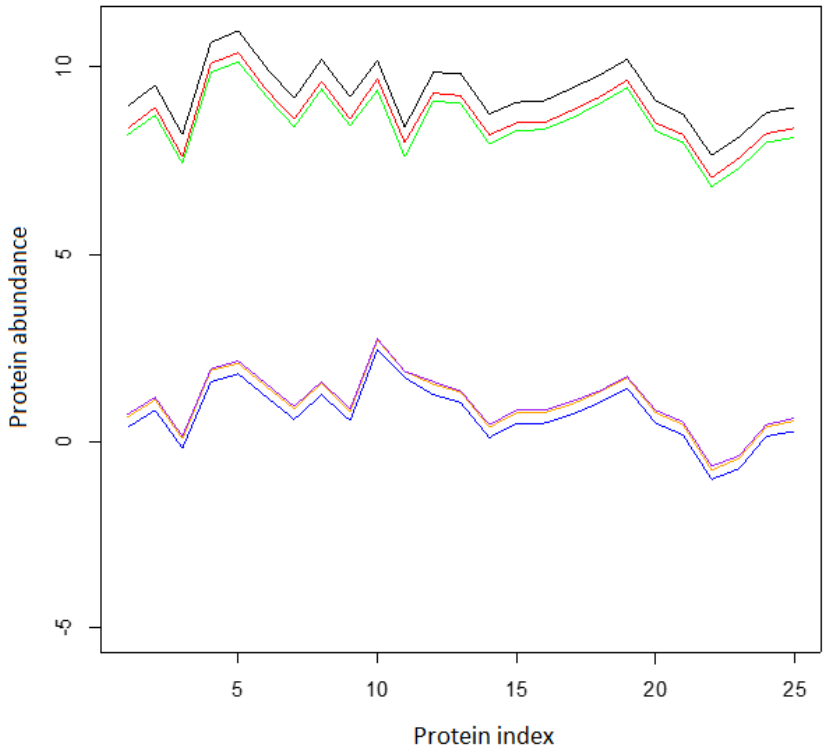

Fig. 2. Illustration of SCAMPI results on relative quantification experiments: Top, the six first lines of a fictive dataset with exactly equal replicates; Middle, the corresponding results as provided by the iterative application of SCAMPI on the six replicates; Bottom, the abundance display of the first 25 proteins of each sample (each colored line correspond to a specific sample). Contrarily to what is expected, there are significant differences between the supposedly equal samples.

experiments with equal number of replicates within each condition and after imputation we observe that Student's t-test systematically outperforms Welch's. This is probably caused by the small number of samples which makes separate variance estimation per group more difficult.

Regularization helps As expected from the proteomics literature, Limma and SAM perform better than Student's. However, Student *t*-test remains useful to represent baseline performances.

Limma and SAM lead to similar performances As both our method and the SAM test are based on the same regularization principle, we use SAM in our comparisons to represent the state-of-the-art performances. We note however that limma accepts more complex hierarchical designs while the native implementation of SAM cannot represent technical batches.

SAM is classically used in proteomics by using the fudge factor to mimic a threshold on the fold change and picking the value that leads to the best detection performance on a dataset, as discussed in Giai Gianetto *and others* (2016). Such use is invalid as it amounts to overfitting and leads to over-optimistic results. Within this work, we only consider the automatic tuning of the fudge factor that is described in the original SAM publication Tusher *and others* (2001).

A.2 *Peptide-based models*

Despite the more general usage of aggregation-based methods, it has long been proposed to directly work at the peptide level, using a regression model on all the peptide intensities that corresponds to a same protein. Goeminne *and others* (2015) suggest that these approaches yield better performances than aggregation-based methods. To the best of our knowledge, the first wide-spread implementation of such a method is MSstats (Choi *and others*, 2014), on the basis of

preliminary works from the same group (Clough *and others*, 2009, 2012). More recently, Goeminne *and others* (2016) proposed another implementation including regularized estimators and a robust loss function.

Most peptide-based models discard shared peptides and apply the regression only to the protein-specific peptides. A few of them however attempt to account for shared peptides, yet none of them seem amenable to statistical inference in our context where hundreds of proteins are involved. Bukhman *and others* (2008) exploit a model akin to the one we use in this paper but include a factor representing the peptide-specific relationship between the measured peptide intensity and its actual abundance. This factor causes the negative log-likelihood to be non-convex in the set of parameters making its minimization non-trivial and possibly expensive – the authors restrict themselves to peptides shared by no more than two proteins. No algorithm or code is available for this method, to the best of our knowledge. More recently Blein-Nicolas *and others* (2012) have proposed AllP, which is also based on a similar model as our work but uses a log-normal model. The use of this distribution corresponds to a common assumption on observed peptide distribution (Podwojski *and others*, 2010). Unfortunately, maximizing the corresponding likelihood is more computationally demanding than that of the normal distribution we use, as no closed form maximizer is available. As reported in its original article, a synthetic datasets with 100 proteins requires 3 days of computation and for such a dataset the algorithm does not converge in 18% of the cases. As a result, it was impossible for us to apply AllP on the much larger (real or simulated) datasets that are considered in this work.

B. Additional experiments on a mouse liver dataset

Here we illustrate on a real biological problem the importance of shared peptides in proteomics. As with most real life dataset, no ground truth is available for this dataset, making its use for experimental validation impossible. We only use it to illustrate the amount of shared peptides

and their impact on differential abundance analysis.

B.1 *Dataset*

We consider a proteomic dataset obtained from the LC-MS/MS analysis of mouse liver samples. Indeed, the liver contains numerous protein isoforms and is especially adapted to illustrate the issue of shared peptides. Liver samples from mutant knock-out mice (n=12) were collected and compared to age-matched wild-type mice (n=10) at 36 and 44 weeks. All mice have the same genetic background (strain C57BL/6), except for the studied mutation, which is not detailed for confidentiality considerations (scientific exploitation of these data is still pending and submitted to the consortium rules which drives its intellectual property).

A biochemical protocol based on tissue lysis and ultracentrifugation was used to prepare mouse liver samples and separate soluble proteins from membrane proteins. The dataset considered here corresponds to the LC-MS/MS analysis of the soluble protein fraction only. Protein digestion was performed using the MED-FASP protocol (Wiśniewski, 2016) on a 10 kDa ultrafiltration device. Peptide digests were desalted using Macrospin C18 columns (Harvard apparatus) before LC-MS/MS analysis.

Peptide digests were analyzed using an Ultimate 3000 nanoLC-chromatography system (Dionex, Voisins le Bretonneux, France) coupled to a Q-Exactive_Quadrupole-Orbitrap Mass Spectrometer, (ThermoFisher Scientific). Peptides were separated on a 75 m x 250 mm C18 column (ReproSil-pur 1.9m, Cluzeau) using a 240 minutes gradient at a flow rate of 300 nl/min. The mass spectrometer was operated in data-dependent acquisition mode.

Raw files were processed using the MaxQuant software (Cox and Mann, 2008). Spectra were searched against the SwissProt database (Mus Musculus taxonomy). Trypsin was chosen as the enzyme and two missed cleavages were allowed. Peptides modifications allowed during the search were: carbamidomethylation (C, fixed), acetyl (Protein Nterm, variable) and oxidation (M, vari-

able). Minimum peptide length was set to seven amino acids. Minimum number of peptides, razor + unique peptides and unique peptides were all set to 1. Maximum false discovery rates (FDR) - calculated by employing a reverse database strategy - were set to 0.01 at peptide and protein levels. Sequences identified in the reverse database and trypsin were discarded.

B.2 *Results and discussion*

B.2.1 *Proportion of shared peptides* The resulting dataset contains 25315 peptides, pointing on 11747 different protein sequences in the identification database. Among them, 10694 do not have any specific peptide, that is 91% of the protein sequences. The remaining 9% typically have few specific peptides as illustrated on Figure 3 where the histogram rapidly decreases. Moreover, even proteins with specific peptides generally also have many shared peptides, as illustrated on Figure 4. This illustrates how widespread shared peptides are in real proteomics data.

In practice however, proteomics experts do not directly work on these data: given the proportion of proteins with no specific peptides, it would be impossible to derive reliable biological conclusions. They classically apply some preprocessing beforehand:

- First, they group proteins together so as to define equivalence classes: these so-called protein groups are then used as surrogate for proteins. The rationale behind is both practical and biological: it is often not possible to discriminate two rather similar sequences on the basis of the only peptides that are observable with MS, so that it makes sense to work on equivalence classes. Beside, on a more biological viewpoint, two homologous sequences often have similar biological functions so that replacing all the protein variants by a single protein group makes sense. However, on protein grouping has been performed it becomes impossible to discriminate between protein groups defined for practical reasons and those which are biologically consistent.

- Second, they rely on the quantitative signal to mitigate the risk of having badly identified

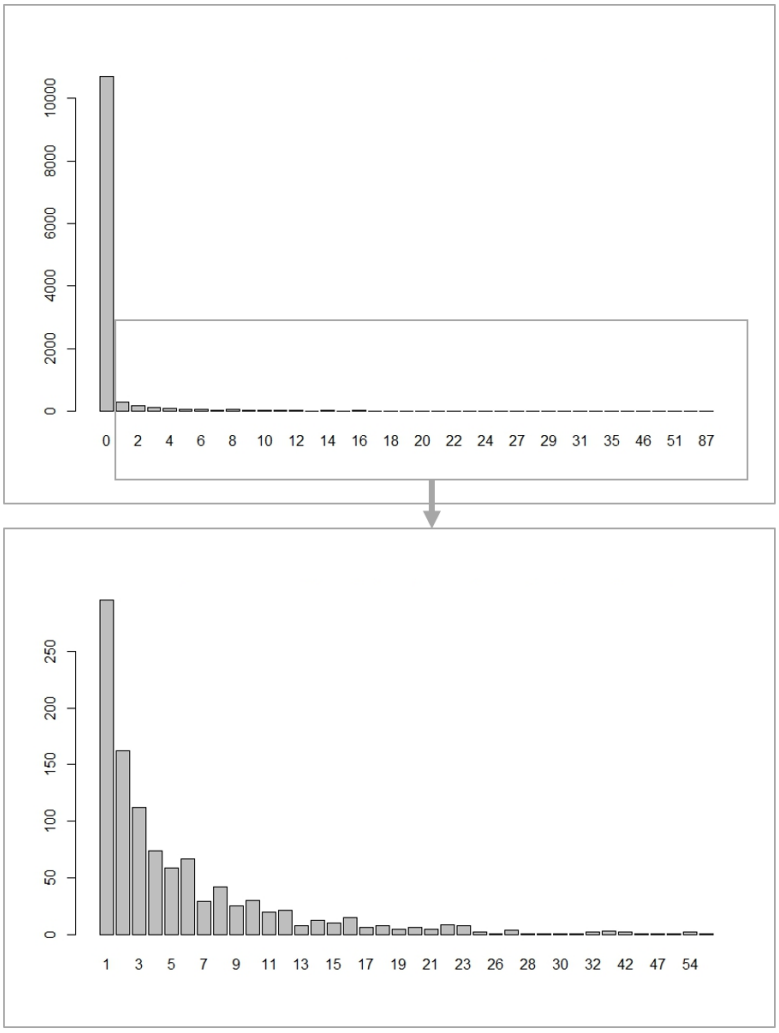

Fig. 3. Histogram of the number of specific peptides per protein. Top: full histogram. Bottom: histogram across proteins with at least one specific peptide.

peptides pointing on wrong protein sequences. Concretely, if before any imputation step, there are too many missing values for a given peptide, it is common practice to filter it out, as well as the protein-related information it carries.

These two steps considerably reduce the amount of shared peptide as a peptide shared between two sequences that are grouped together is no longer considered shared anymore, but at some cost. First, the dataset contains fewer protein groups than protein sequences, and fewer peptides

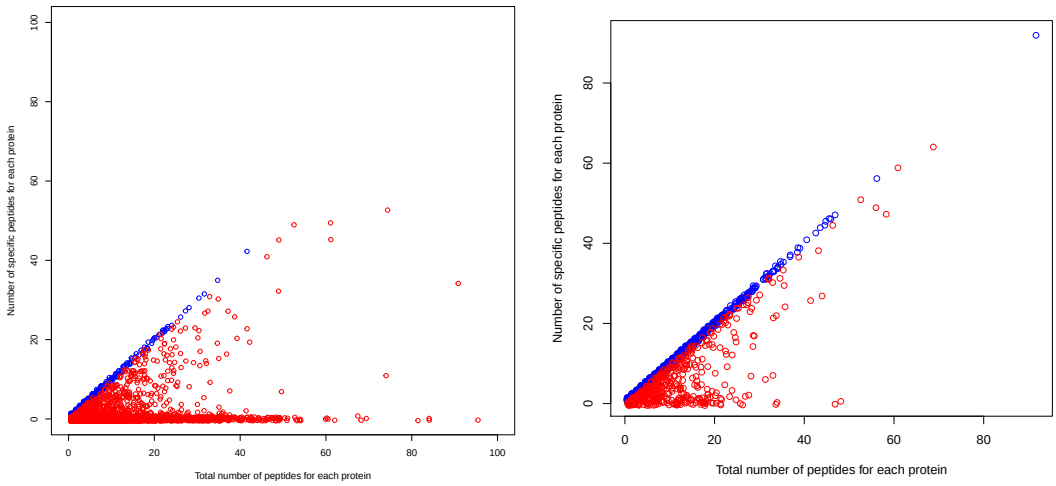

Fig. 4. Number of specific peptides as a function of the total number of peptides for all proteins in the dataset. Left: before protein grouping. Right: after protein grouping. A noise uniformly sampled from $[-0.5, 0.5] \times [-0.5, 0.5]$ was added to each point to separate multiple proteins with identical coordinates.

after filtering than before, so that the proteomic landscape is somehow truncated, or at least coarse-grained. Second, these two steps are subjective and practitioner-dependent by nature, making them less reproducible and reliable.

As an illustration, we have applied these two processing steps on the dataset, following the same subjective choices classically done by the proteomics experts in our lab. Protein grouping was performed with MaxQuant according to the above description. Then, through ProStaR software (Wieczorek *and others*, 2017), missing values were dealt with as following: First, peptides with less than 7 observed values (*i.e.*, up to 3 or 5 missing values for wild-type and mutant groups, respectively) were removed; second, the quantitative values were normalized by aligning the median abundance of each sample; finally, remaining missing values were imputed with a maximum likelihood algorithm. We obtained a dataset containing 17504 peptides and 2431 protein groups. Among them, only 88 have no specific peptides, so that one has moved from 91% of protein sequences with no specific peptides down to 3.6% of protein groups with no specific peptides. However, one must keep in mind that to reduce the effective impact of the shared

peptide problem, one has shifted from 25315 to 17504 peptides, and from 10694 to 2431 protein level entities (either sequences of groups): This coarse-graining amounts to reducing by 31% the number of peptides and to dividing the number of protein-level entities by 4.4. Even though the loss in terms of protein identities may be tampered by adequate preprocessing, it nonetheless remains concrete, which justifies the investigations presented in this article.

B.2.2 *Effect of shared peptides on differential analysis* The rationale of our approach is that differential analysis should be more accurate when exploiting shared peptides than when discarding them, as shared peptides increase the sample size. We confirmed this improvement in the experimental section of the main manuscript over small standardized datasets where the ground truth was known and on simulated data. The mouse liver dataset we discuss here has no such ground truth information. However, it is possible to conduct differential analyses with and without shared peptides, and to discuss how the results change. To do so, we have applied the test proposed in this article on the entire dataset, and on the dataset restricted to protein-specific peptides. We have then ranked the 2431 protein groups according to their p-values, and for each test, we have compared these two ordering (p-values equal to 1 have been given to the 88 proteins that are lost because of their absence of specific peptides).

As it clearly appears on Figure 5, a proportion of the protein groups are equivalently ranked by PEPA (either its ML or MAP version), regardless of the involvement of shared peptides – these groups are the dots along the diagonal line of each graph. However, it also appears that a proportion of protein groups have completely different ranking depending on whether the shared peptides are involved or not. Notably, this set of differently ranked protein groups goes beyond the 88 protein groups that are lost by absence of specific peptides – these lost protein groups are depicted by the horizontal chunk on the top of each graph, indicating that they have been downgraded to the bottom of the ranking because of p-values equal to 1. Numerous other protein

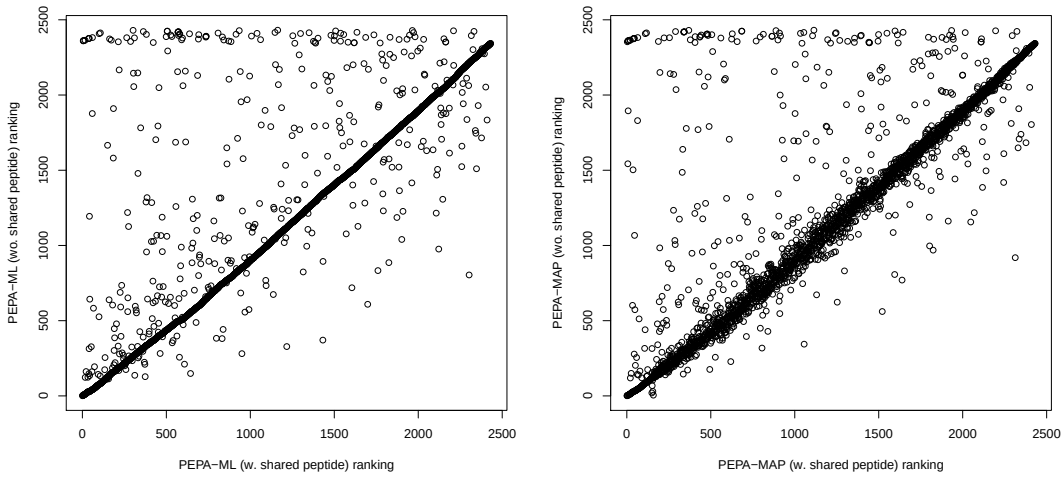

Fig. 5. Rank of the pvalues for the 2431 group-proteins, using or not using shared peptides. Left: using the maximum likelihood version of PEPA. Right: using the maximum a posteriori version of PEPA.

groups have a significantly different rank, indicating that accounting for shared peptides affects protein ranking. Figure 6 offers another view of the same phenomenon, by showing the proportion of proteins which are both among the N lowest p-values obtained with the procedure using shared peptides and among the N lowest p-values obtained without using these shared peptides, as a function of N. For most useful values of N, a large proportion of the selected proteins differs between the two procedures.

C. Approximation of the log-sum by a sum for the mean intensity of peptide log-abundances

As discussed in the main manuscript, the observed intensities $\tilde{y}_k$ from an MS/MS experiment are typically modeled as samples from a log-normal distribution (equation (1) in the manuscript). We approximate the mean $\ln\left(\sum_{j=1}^{p} x_{kj}\theta_j\right) + \alpha_k$ of the normal distribution of the log-intensity $\ln \tilde{y}_k$ of peptide k by a linear term $\sum_{j=1}^{p} x_{kj}\theta_j + \alpha_k$ (equation (3) in the manuscript) – the θ_j and α_k in the two representations are not expected to be the same. Since the likelihood ratio statistic that

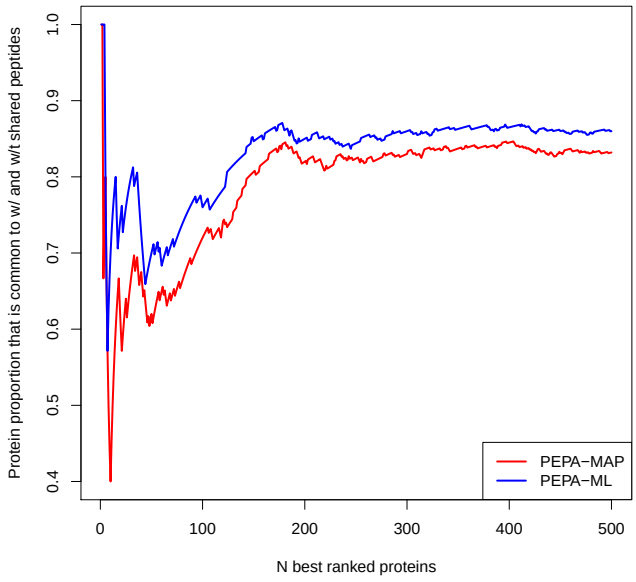

Fig. 6. Proportion of common protein between the N ones with lowest p-values obtained by the PEPA test using shared peptides or not, as a function of N.

we use only requires the squared residuals of the estimated model, this approximation allows us to compute the statistic by solving a problem of the form $\min_\beta \|Y - \mathbf{X}\beta\|^2$ which is amenable to the acceleration introduced in Proposition 1, rather than the more difficult $\min_\beta \|Y - \ln(\mathbf{X}\beta)\|^2$.

We restrict ourselves to a single sample for this discussion to make the notation lighter, extensions to several samples are straightforward. The first order Taylor expansion of $\ln \sum_{j=1}^{p} x_{kj}\theta_j = \ln x_k^\top \theta$ around the average $q^{-1}\sum_{j=1}^{p}\sum_{k=1}^{q} x_{kj}\theta_j \triangleq \bar{x}^\top\theta$ is $\ln \bar{x}^\top\theta + (x_k^\top\theta - \bar{x}^\top\theta)(\bar{x}^\top\theta)^{-1}$ so:

$$\begin{aligned}\ln x_k^\top\theta + \alpha_k &= \ln \bar{x}^\top\theta + \left(x_k^\top\theta - \bar{x}^\top\theta\right)\left(\bar{x}^\top\theta\right)^{-1} + \alpha_k + o\left(\left|\frac{x_k^\top\theta - \bar{x}^\top\theta}{\bar{x}^\top\theta}\right|\right)\\ &= x_k^\top\tilde{\theta} + \tilde{\alpha}_k + o\left(\left|\frac{x_k^\top\theta - \bar{x}^\top\theta}{\bar{x}^\top\theta}\right|\right),\end{aligned}$$

where $\tilde{\theta} = \theta(\bar{x}^\top\theta)^{-1}$ and $\tilde{\alpha}_k = \alpha_k + \bar{x}^\top\theta - 1$.

The perturbation over the residuals ε_{ik} obtained when doing a linear regression of the $\{y_k^i\}_{k=1,\ldots,q}^{i=1,\ldots,n}$ against $(X\ I_q)$ instead of the correct log-linear model $\ln \sum_{j=1}^{p} x_{kj}\theta_j + \alpha_k$ are therefore in $o\left(\left|\frac{x_k^\top\theta - \bar{x}^\top\theta}{\bar{x}^\top\theta}\right|\right)$.

Consequently, the effect of the approximation on sum of squared residuals used in the likelihood ratio statistic depends on the magnitude of the $\left|\frac{x_k^\top\theta-\bar{x}^\top\theta}{\bar{x}^\top\theta}\right|$ compared to σ.

We assessed the impact of the approximation error on our simulated data. To separate the effect of the linear approximation from the approximation caused by using a binary X instead of a continuous one, we compared $X\theta$ to its linear approximation on a binarized matrix for various proportions of shared peptides. Figure 7 shows the linear approximation as a function of $x_k^\top\theta$ and the corresponding relative approximation errors. The approximation was generally better when fewer peptides were shared as connected components were smaller and the approximation was done around the mean of only a few peptide abundances. The errors were generally small before σ, suggesting that they had a limited impact on the test statistic: the upper quartile of $\sigma^{-1}|\ln(x_k^\top\theta) - (\ln\bar{x}^\top\theta + (x_k^\top\theta - \bar{x}^\top\theta)(\bar{x}^\top\theta)^{-1})|$ across peptides was 0.12 for 50% of shared peptides, 0.06 for 10% and 0.02 for 5%.

More generally on real data, the quality of the approximation depends on the empirical variance of peptide abundances $x_k^\top\theta$ within each connected component: we expect it to be good as long as the ratio of the variations of peptide abundances around their average over this average are small before the gaussian residuals.

A more precise analysis should take into account the fact that the test statistic may be unaffected if the effect of the approximation on the residuals is large but similar under $\mathbf{H}_0$ and $\mathbf{H}_1$. For example when testing differential abundance for a protein j, a peptide log-abundance may being poorly approximated by the linear model without affecting the test statistic as long as its estimated log-abundance is similar unaffected by the constraint that $\theta_j = \theta'_j$.

D. PEPA-GLOBAL, A FAST HEURISTIC FOR THE PEPA TEST

As reported in Table 1 of the main manuscript, the computational bottleneck of PEPA is the computation of the connected components of the peptide-protein graph. Even if a connected

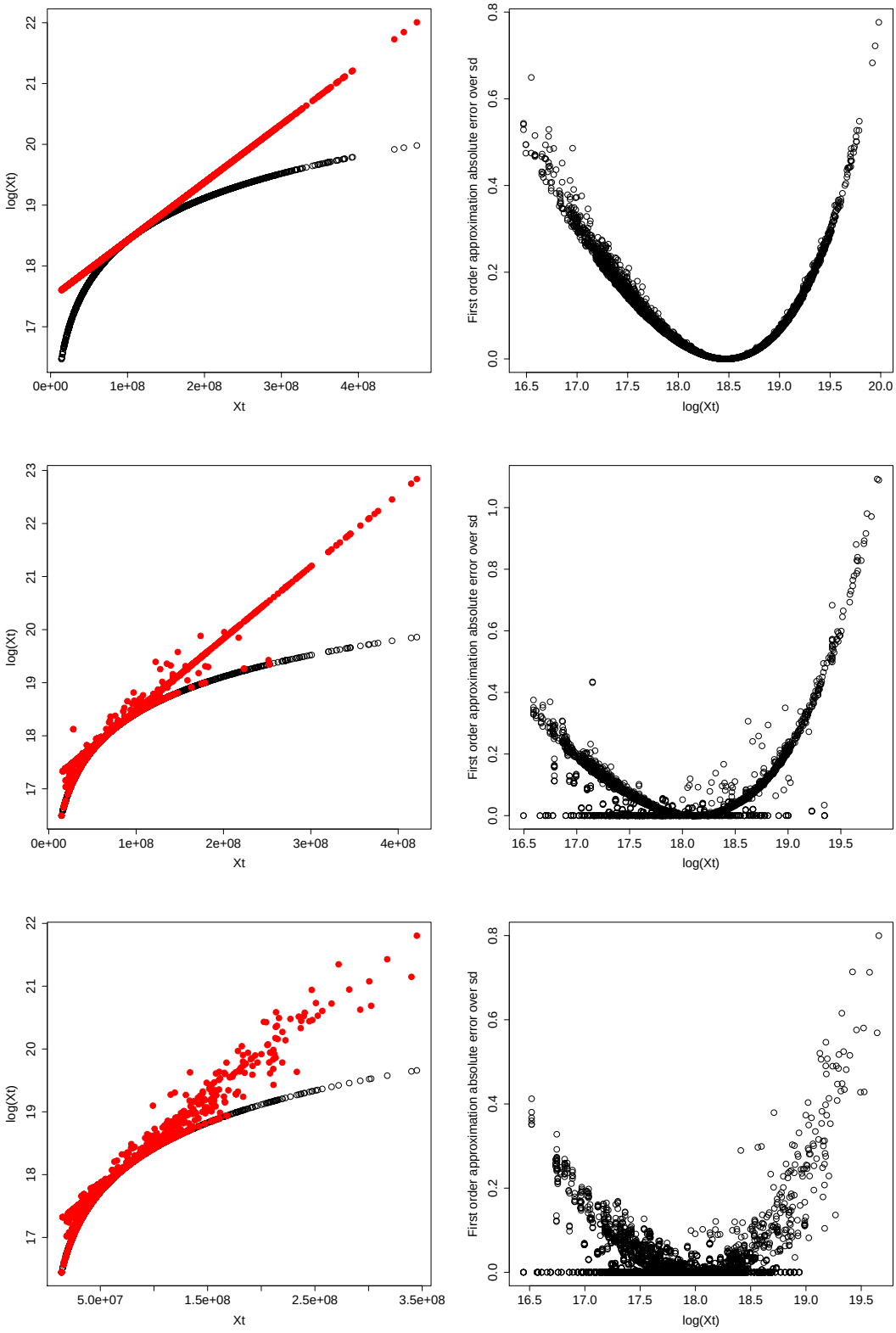

Fig. 7. First order approximation of $\ln(X\theta)$ (left) and corresponding absolute error relative to σ (right). Data were simulated with 50% (top), 10% (middle) and 5% (bottom) of shared peptides.

component involves a large number of peptides and proteins, computation of the test statistic itself using Proposition 1 only accounts for less than 10% of the total runtime. Computing the connected components on a real dataset such as the one used in Section B of this Supplementary Material takes between 2 and 3 hours on a standard workstation, which can be a deterrent for benchmarking phases.

Including peptides in the likelihood ratio test for a protein j whose observation does not affect the estimation of θ_j implicitly amounts to regularizing the variance estimate. Including all peptides in the dataset for example leads to a likelihood ratio statistic

$$\lambda_{\text{PEPA-global}} = (nq_j)^{-1} \left(\ln \left((nq_j)^{-1} \left\| y^j - \mathbf{X}_0^j \hat{\beta}_k^j \right\|^2 + c \right) - \ln \left((nq_j)^{-1} \left\| y^j - \mathbf{X}_1^j \hat{\beta}_k^j \right\|^2 + c \right) \right),$$

where $c \triangleq (nq_j)^{-1} \left\| y^{-j} - \mathbf{X}^{-j} \hat{\beta}_k^{-j} \right\|^2$, q_j is the number of peptides in the connected component of protein j. The j (resp. $-j$) superscript denotes the restriction of variables y, $\mathbf{X}$ and β to the set of peptides and proteins in the connected component of protein j (resp. complement of this set). Note that by construction $\mathbf{X}_0^{-j} = \mathbf{X}_1^{-j} \triangleq \mathbf{X}^{-j}$.

Skipping the computation of connected components and computing our likelihood ratio statistic across all peptides for each protein therefore yields a MAP version of the test, with a fudge factor c proportional to the sum of squared residuals for peptides outside the connected component of the tested protein. We refer to this procedure as PEPA-global. Interestingly, c is larger for connected components involving few peptides (small q_j, larger number of peptides outside the components leading to typically larger $\|y^{-j} - \mathbf{X}^{-j} \hat{\beta}_k^{-j}\|^2$). Since the sample size for tests involving proteins within these components are smaller, a larger amount of regularization is often beneficial. Accordingly, precision-recall curves for PEPA-global on the datasets studied in the main manuscript are qualitatively similar to those obtained for PEPA-ML and PEPA-MAP, slightly better on simulated data (Supplementary Figure 8) and slightly worse on Exp1_R25_pept (Supplementary Figure 9).

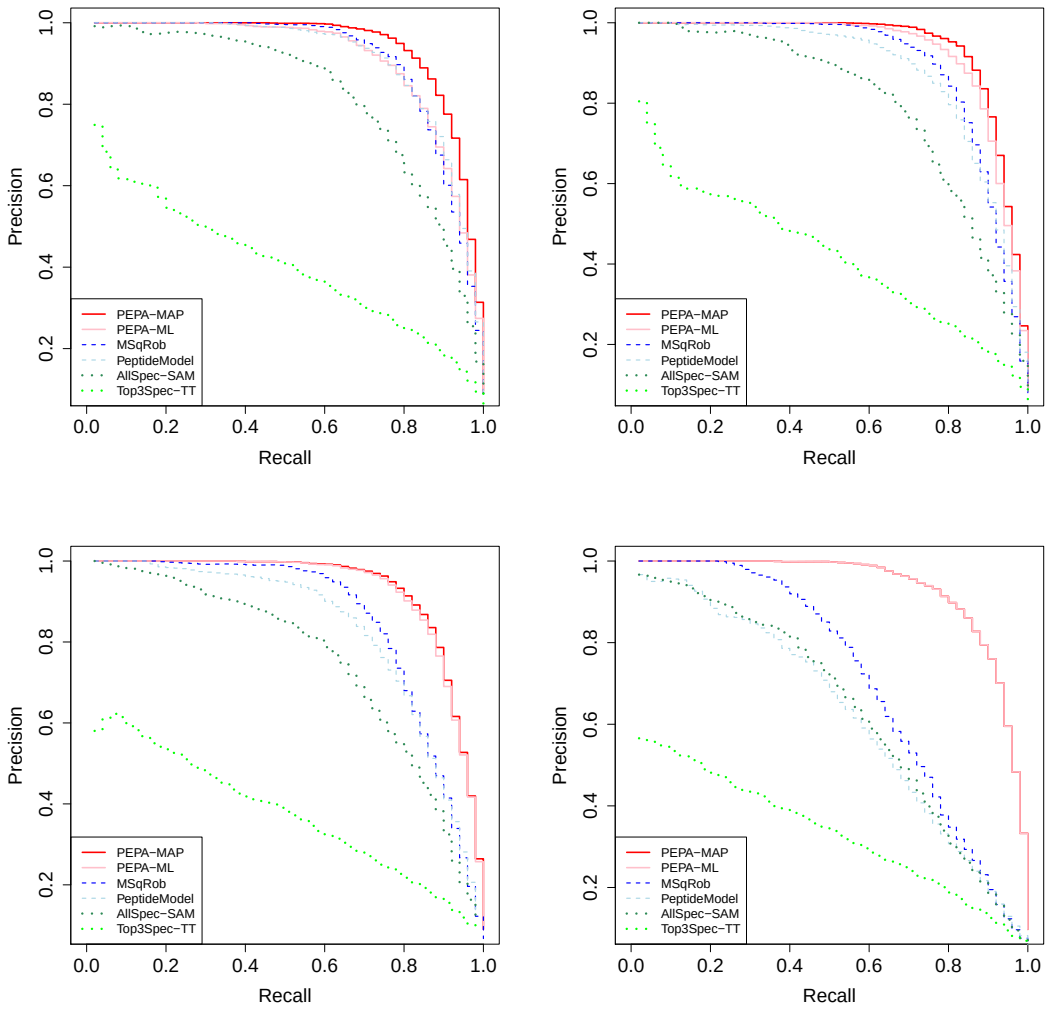

Fig. 8. PR curve of PEPA -ML, -MAP and -global on simulated data with 0% (upper left), 5% (upper right), 20% (lower left) and 50% (lower right) of shared peptides.

PEPA-global therefore provides a similar precision-recall performance as PEPA-ML and PEPA-MAP while being 10 to 30 times faster (see Table 1 of the main manuscript). However, its fudge parameter c is not explicitly controled and depends on the total number of peptides in the dataset, which is not desirable. More importantly, computing p-values for PEPA-global using Proposition 2 would require the knowledge of c which in turn requires computing the connected components.

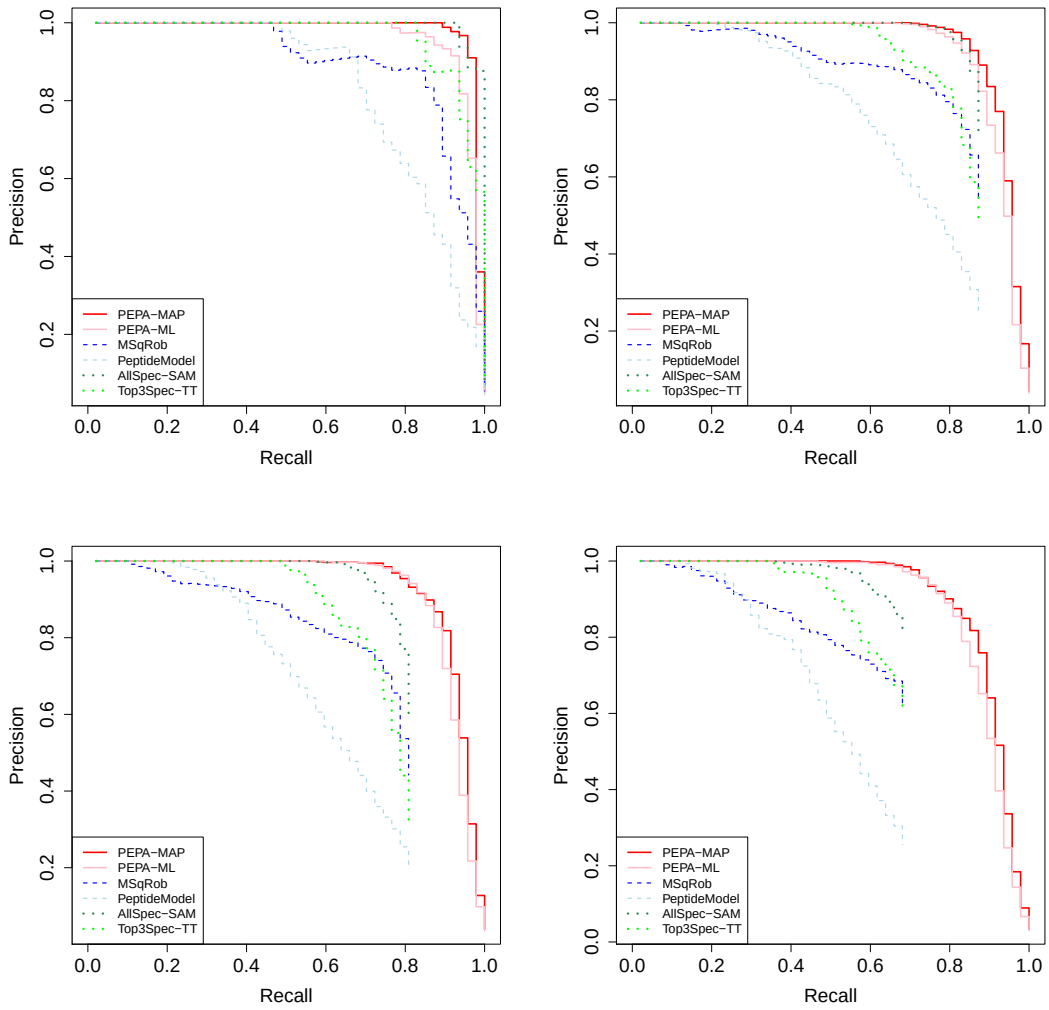

Fig. 9. PR curve of PEPA -ML, -MAP and -global on Exp1_R25_pept data with 0 (upper left), 120 (upper right), 200 (lower left) and 280 (lower right) artificially added shared peptides

We still provide PEPA-global as an option in the `pepa.test` function of our `DAPAR` Bioconductor package with two main applications in mind. First, practitioners may want to quickly compare testing procedures on pilote studies using positive and negative controls, *i.e.*, known differentially abundant and non-differentially abundant proteins. Such a comparison only requires to rank proteins and assess whether positives (resp. negative) are in the top (resp. bottom) of the ranking.

Second, on datasets with enough biological samples, p-values could be computed for PEPA-global using resampling techniques.

E. Proof of Proposition 1

Proof. The residuals can be re-writen using the Pythagorean theorem: $\min_\beta \|y - \mathbf{X}_k\beta\|^2 = \|y - \mathbf{X}_k(\mathbf{X}_k^\top\mathbf{X}_k)^\dagger\mathbf{X}_k^\top y\|^2 = \|y\|^2 - \|\mathbf{X}_k(\mathbf{X}_k^\top\mathbf{X}_k)^\dagger\mathbf{X}_k^\top y\|^2 = \|y\|^2 - \|(\mathbf{X}_k^\top\mathbf{X}_k)^{\frac{\dagger}{2}}\mathbf{X}_k^\top y\|^2$, where $(\mathbf{X}_k^\top\mathbf{X}_k)^{\frac{\dagger}{2}}$ denotes the square root of the pseudo-inverse of $(\mathbf{X}_k^\top\mathbf{X}_k)$.

For $k \in \{0, 1\}$, $\mathbf{X}_k^\top\mathbf{X}_k$ admits the Cholesky decomposition

$$\mathbf{X}_k^\top\mathbf{X}_k = L_k L_k^\top \tag{E.1}$$

with

$$L_0 = n^{\frac{1}{2}} \begin{pmatrix} X^\top \\ I_q \end{pmatrix}$$
$$L_1 = n^{\frac{1}{2}} \begin{pmatrix} 0 & X_{-j}^\top \\ a & n^{-1}n_1 x_j^\top \\ -a & n^{-1}n_2 x_j^\top \\ 0 & I_q \end{pmatrix},$$

where $a = n^{-1}(n_1 n_2)^{\frac{1}{2}}\|x_j\|$. We also notice that

$$\mathbf{X}_k^\top y = L_k h_k, \tag{E.2}$$

with

$$h_0 = n^{\frac{1}{2}}\bar{y}$$
$$h_1 = n^{\frac{1}{2}} \begin{pmatrix} n^{-1}(n_1 n_2)^{\frac{1}{2}}\|x_j\|^{-1}x_j^\top\left(\bar{y}^{(1)} - \bar{y}^{(2)}\right) \\ \bar{y} \end{pmatrix}.$$

Combining (E.1) and (E.2), we obtain $\|(\mathbf{X}_k^\top\mathbf{X}_k)^{\frac{\dagger}{2}}\mathbf{X}_k^\top y\|^2 = \|(L_k L_k^\top)^{\frac{\dagger}{2}} L_k h_k\|^2$. Since L_0 has full column rank q and L_1 has full column rank $q+1$, $\|(L_k L_k^\top)^{\frac{\dagger}{2}} L_k h_k\|^2 = \|U_k V_k^\top h_k\|^2 = \|h_k\|^2$, where $L_k = U_k\Lambda_k V_k^\top$ is the singular value decomposition of L_k. □

F. PROOF OF PROPOSITION 2

Proof. We first derive the null distribution of $\lambda(\hat{\sigma}_0^2 + s, \hat{\sigma}_1^2 + s)$ when

$$y_i | x_i, \beta, \sigma^2 \overset{iid}{\sim} \mathcal{N}(x_i^\top \beta, \sigma^2), i = 1, \dots, n, \tag{F.3}$$

and $(\hat{\beta}_0, \hat{\sigma}_0^2)$ and $(\hat{\beta}_1, \hat{\sigma}_1^2)$ are the maximum likelihood estimators of (β, σ^2) under $\mathbf{H}_0 : \beta_j = 0$ and $\mathbf{H}_1 : \beta_j \neq 0$ respectively.

We follow the line of proof of Wilk's theorem in (van der Vaart, 2007, Section 16.2). Classical asymptotic normality of the maximum likihood estimator provides that

$$\sqrt{n}\hat{\beta}_j \to \mathcal{N}\left(0, \left(\mathcal{I}^{-1}\right)_{j,j}\right),$$

where

$$\mathcal{I} = \begin{pmatrix} \frac{X^\top X}{\sigma^2} & 0 \\ 0 & -\partial^2_{\sigma^2} \ln f \end{pmatrix} \in \mathbb{R}^{(p+1)\times(p+1)}$$

is the Fisher information matrix with respect to parameters (β, σ^2) for the normal distribution f described in (F.3), and $X \in \mathbb{R}^{n \times p}$ is the matrix whose rows are the x_i.

We note that

$$\lambda(\hat{\sigma}_0^2 + s, \hat{\sigma}_1^2 + s) = n\left(\ln\left(\hat{\sigma}_1^2 + s\right) - \ln\left(\hat{\sigma}_0^2 + s\right)\right)$$

can also be written as $-2(g(\hat{\beta}_1, \hat{\sigma}_1^2, s) - g(\hat{\beta}_0, \hat{\sigma}_0^2, s))$ where

$$g(\beta, \sigma^2, s) = -\frac{n}{2}\left(\ln\left(\sigma^2 + s\right) + \left(n^{-1}\|y - X\beta\|^2 + s\right)\left(\sigma^2 + s\right)^{-1}\right).$$

Even though $g(\beta, \sigma^2, s)$ is not the log-likelihood of the y_i, its gradient $\nabla_{(\beta,\sigma^2)} g$ is cancelled at the maximum likelihood estimator $(\hat{\beta}_1, \hat{\sigma}_1^2)$. Using a second order Taylor approximation of $g(\hat{\beta}_1, \hat{\sigma}_1^2, s)$ around $g(\hat{\beta}_0, \hat{\sigma}_0^2, s)$, one can therefore show that

$$\lambda(\hat{\sigma}_0^2 + s, \hat{\sigma}_1^2 + s) \to \sqrt{n}\hat{\beta}_j \left(\left(\mathcal{I}_g^{-1}\right)_{j,j}\right)^{-1} \hat{\beta}_j \sqrt{n},$$

where $\mathcal{I}_g$ is minus the expectation of the Hessian matrix of g:

$$\mathcal{I}_g = \begin{pmatrix} \frac{X^\top X}{\sigma^2 + s} & 0 \\ 0 & -\partial^2_{\sigma^2} g \end{pmatrix}.$$

It follows that $\lambda \to z^2$, where $z \sim \mathcal{N}\left(0, \tilde{\sigma}^2\right)$ and

$$\tilde{\sigma}^2 = \left(\left(\mathcal{I}_g^{-1}\right)_{j,j}\right)^{-1} \left(\mathcal{I}^{-1}\right)_{j,j} = \frac{\sigma^2}{\sigma^2 + s},$$

and therefore $\frac{\sigma^2+s}{\sigma^2}\lambda(\hat{\sigma}_0^2 + s, \hat{\sigma}_1^2 + s) \to \chi_1^2$.

The proposition follows using a change of variable:

$$\tilde{\beta} = \left(\beta_1, \ldots, \beta_{j-1}, \frac{\beta_j - \beta'_j}{2}, \frac{\beta_j + \beta'_j}{2}, \beta_{j+1}, \ldots, \beta_p\right) \in \mathbb{R}^{p+1}$$
$$\tilde{x} = (x_1, \ldots, x_{j-1}, s, 1, x_{j+1}, \ldots, x_p) \in \mathbb{R}^{p+1},$$

where s is 1 for the first n_1 x_i and -1 for the remaining n_2.

Alternatively, it was brought to our attention that this result could be derived as a special case of Theorem 1 in Harchaoui *and others* (2008), which describes the distribution under the null hypothesis of homogeneity for a statistic built over samples in arbitrary spaces. Their statistic is a kernel version of the Hotelling T^2 statistic, boiling down to a squared t-statistic in the unidimensional case which corresponds to the likelihood ratio statistic for our model. In the univariate case, their general distribution also boils down to a χ_1^2 distribution. □

G. Datasets

Here we provide more details about the datasets used in the experiments presented in our manuscript.

G.1 *Simulation*

We first generate a $q \times p$ peptide-protein membership matrix X. To do so, we assign to each protein a number of peptides drawn from a $\mathcal{P}(q/p)$ Poisson distribution. We make sure that each protein is assigned at least one peptide, and that all peptides are assigned to one protein. We then randomly select a subset of peptides to be shared across proteins (we show results for

proportions 0%, 5%, 20% and 50%). For each of them, a Poisson draw with parameter 0.5 is used to determine the number of additional protein the peptide should connect to.

As explained in the introduction, the relationship between the measured peptide intensity and its actual abundance is peptide specific, so that the observed intensity of two protein-specific peptides from a same protein can be rather different (up to 3 orders of magnitude, according to Silva *and others* (2005)). To account for this fact, we multiply each positive entry of X by a draw from a $0.001 + \beta(1.1, 3)$ distribution. In real cases, only the binary X membership matrix is known thanks to the protein sequence database: accordingly, the modified matrix is only used to simulate the intensities but only its binary counterpart is involved in our test.

We draw a $\theta^{(i)} \in \mathbb{R}_+^p$ abundance vector for each of the $n = n_1 + n_2$ samples, in contrast to the model underlying our testing procedure where all samples under the same condition share the same θ parameter. More precisely, we draw one mean θ parameter for each condition from a log-normal distribution, then add independent normal perturbations around this mean for each sample (thresholding at 0 to make sure each $\theta^{(i)} \in \mathbb{R}_+^p$). The mean and variance of protein-level log-normal distribution are chosen to fit classically observed datasets – that is peptide-level distributions that follow a Gaussian model centered on 23.5 and with a standard deviation of 2 on the log2 scale. The individual normal perturbations on the average abundance θ_j of each protein j have mean 0 and standard deviation $0.005\theta_j$. Under $\mathbf{H_1}$, the first 50 proteins are considered to be differentially abundant. The ratio between conditions is sampled from a normal distribution with mean $R = 15$ and of standard deviation $0.05R$.

Finally the log-abundance y_i^k of peptide k in sample i is sampled from a normal distribution with mean $\log_2\left(X_k^\top \theta^{(i)}\right)$ and variance $(CV \times X_k^\top \theta^{(i)})$ where X_k is the k-th row of X and where CV is a coefficient of variation that was set to 0.1 in our experiments.

The distribution from which we sample y_i^k is therefore the log-normal model (2.1) of the main manuscript, with a peptide-specific variance and a sample-specific (rather than condition-specific)

protein abundance, which is a bit more realistic. Importantly, the expectation of y_i^k is not a linear combination of the parameter $\theta^{(i)}$, making our model (2.3) severely miss-specified.

G.2 *Spike-in data*

We used samples prepared according to the protocol of Ramus *and others* (2016). First, a lysate of yeast is split into $n = 2m$, so as to form an equal background for n samples. Then, a volume V of a mixture containing a series of precisely known human proteins is spiked in the first m samples so as to form the first biological condition. The other m samples receive a $R \times V$ volume of the same mixture so as to form the second biological condition with an abundance ratio R. As yeast and human proteins are different, any identified protein within each sample can uniquely be associated to its parent organism (*i.e.* yeast or human). During the relative quantification step, one should find that all and only the human proteins are differentially abundant.

We use the two datasets described in (Giai Gianetto *and others*, 2016), and which are available in the DAPARdata R package (Wieczorek *and others*, 2017). These two datasets contain 6 samples ($m = 3$) that have been prepared using the equimolar human protein mixture Sigma UPS1, including 48 human proteins. Their differential abundance ratios R are 2 and 2.5 for the first and second datasets, that are respectively referred to as Exp1_R2_pept and Exp1_R25_pept. Both datasets were preprocessed with the ProStaR software (Wieczorek *and others*, 2017), so as to make them compliant with the constraints of the reported experiments. First, all the peptides that corresponded to contaminant proteins or reversed proteins (*i.e.* peptides that were mistakenly identified as part of inexistent proteins) were removed, as well as those with more than one missing values out of three, in any of the two conditions. Second, the peptide intensities were normalized to account for replicate variability (within-condition median centering for Exp1_R25_pept, and global median centering for Exp1_R2_pept). Finally, the remaining missing values were imputed with an algorithm based on maximum likelihood estimation.

A limitation of this dataset is that it contains few shared peptides contrarily to real human samples, for a combination of reasons. First, the 48 human proteins available in the Sigma UPS1 equimolar mixture do not share peptides that are easily identified by mass spectrometry. Second, yeast is a simple organism that does not have a lot of homology sequences in its genome, so that there are very few shared peptides among its proteins. Finally, human and yeast proteins do not share many peptides, because they are rather different organisms. To cope with this issue, we artificially add shared peptides in these two real datasets, by merging pairs of peptides. Merging pairs of yeast peptides had no influence, as the resulting peptides (from non-differentially abundant yeast proteins) were even less prone to create error by introducing confusion between differentially and non-differentially abundant proteins. It was not possible to merge pairs of human peptides, because of their too limited number in the original datasets. We therefore resort to merging pairs of peptides where one is differentially abundant (human) and the other is not (yeast). To concretely implement such a merging procedure, a yeast peptide and a human one are randomly chosen, their identifications are merged, their abundances are summed, and their list of parent proteins are concatenated. This procedure can be repeated as many times as required to reach the desired amount of shared peptides. In the *Results* Section of the manuscript, we report experiments with respectively 0, 120, 200 and 280 artificial shared peptides. These numbers are to be compared to the total numbers of human peptides in the datasets that are 211 (respectively 290) in Exp1_R2_pept (respectively Exp1_R25_pept).

H. Additional plots

This Section presents plots produced in the same setting as those in the main manuscript, but using different parameters. Figure 10 provides PR curves for simulations with alternative proportions of shared peptides. Figure 11 and 12 provide PR curves for alternative numbers of shared peptides on the Exp1_R2_pept and Exp1_R25_pept datasets respectively. Figures 13 and 13 show

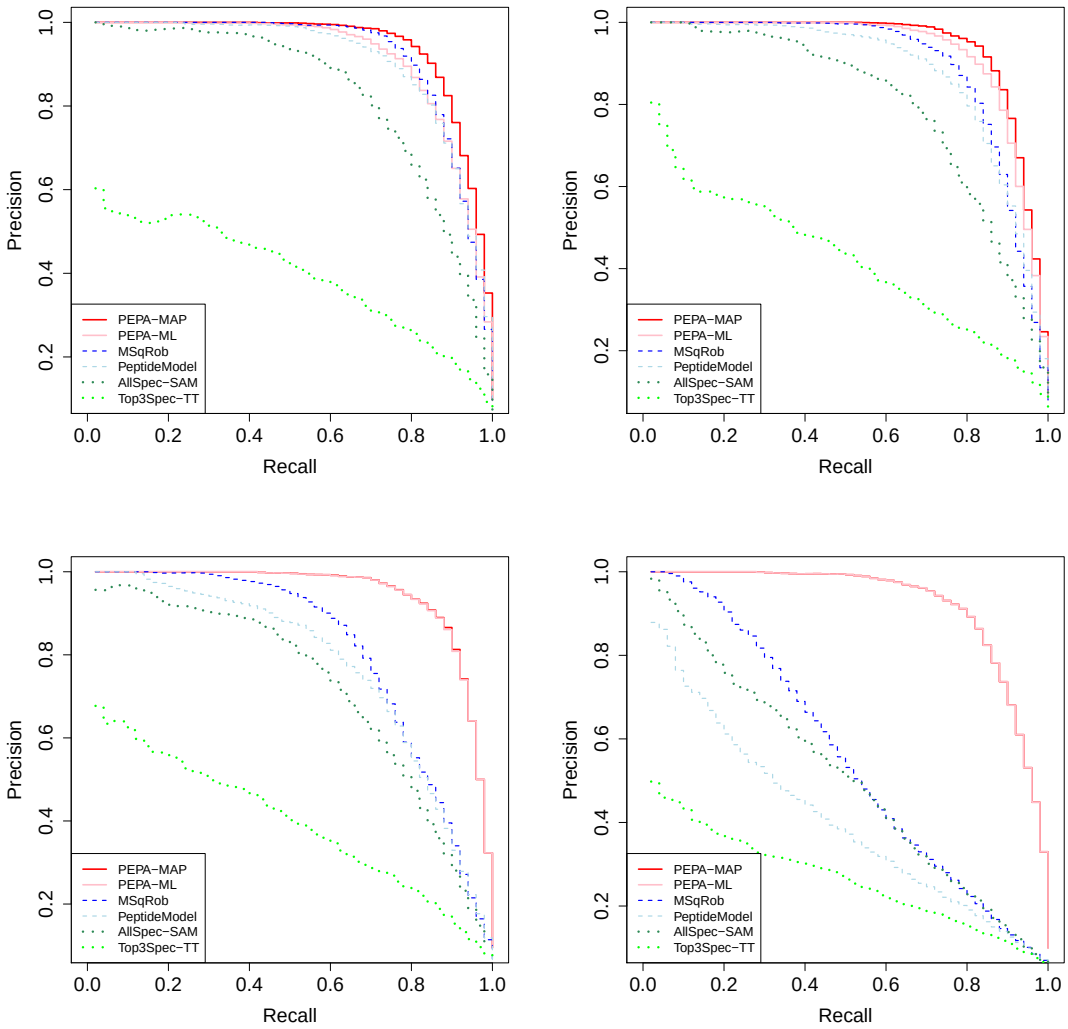

Fig. 10. PR curve on simulated data with 1% (upper left), 10% (upper right), 33% (lower left) and 67% (lower right) of shared peptides.

calibration plots one simulated data with different proportions of shared peptides. Figures 15, 16, 17 and 18 are calibration plots on Exp1_R2_pept and Exp1_R25_pept with different numbers of shared peptides.

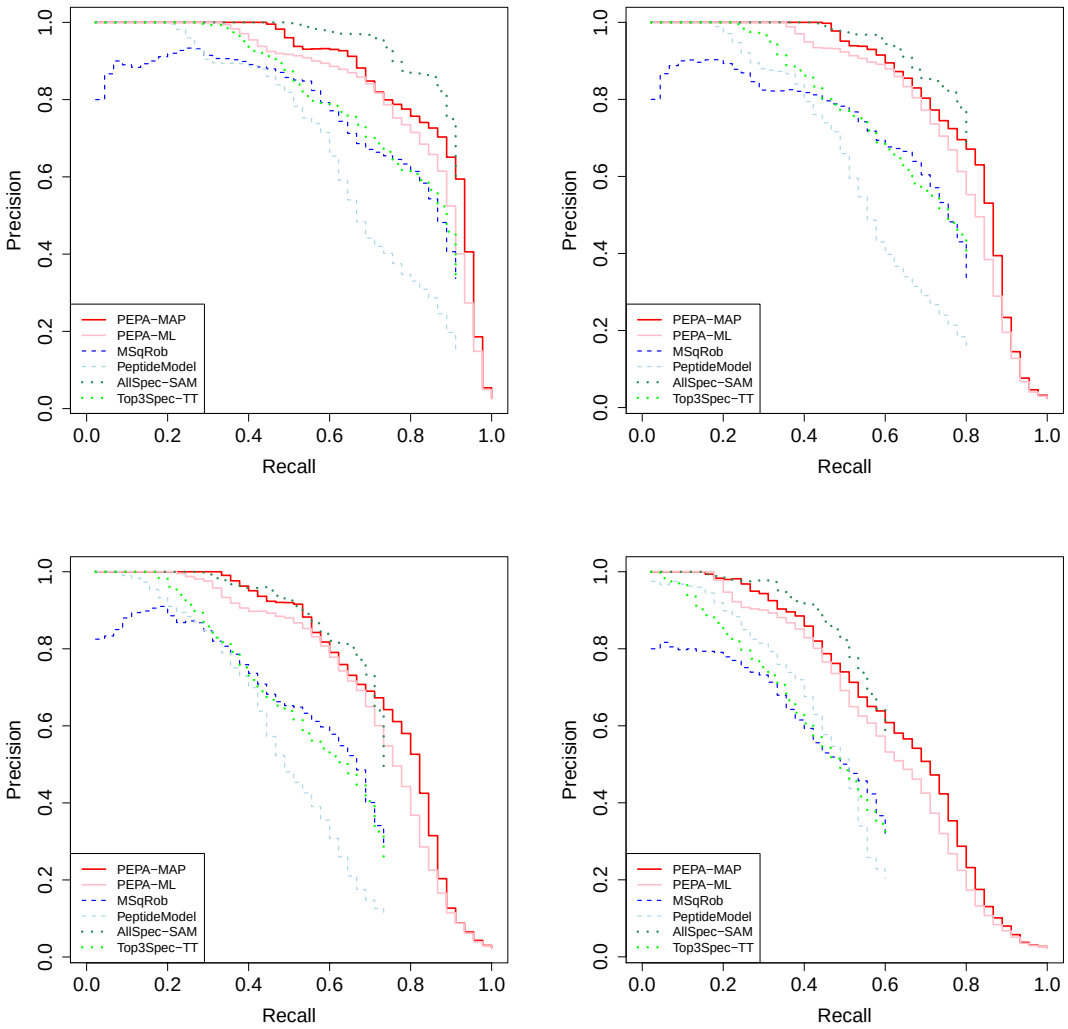

Fig. 11. PR curve on Exp1_R2_pept data with 40 (upper left), 80 (upper right), 160 (lower left) and 240 (lower right) artificially added shared peptides.

References

Blein-Nicolas, Mélisande, Xu, Hao, de Vienne, Dominique, Giraud, Christophe, Huet, Sylvie and Zivy, Michel. (2012). Including shared peptides for estimating protein abundances: A significant improvement for quantitative proteomics. *Proteomics* **12**(18), 2797–2801.

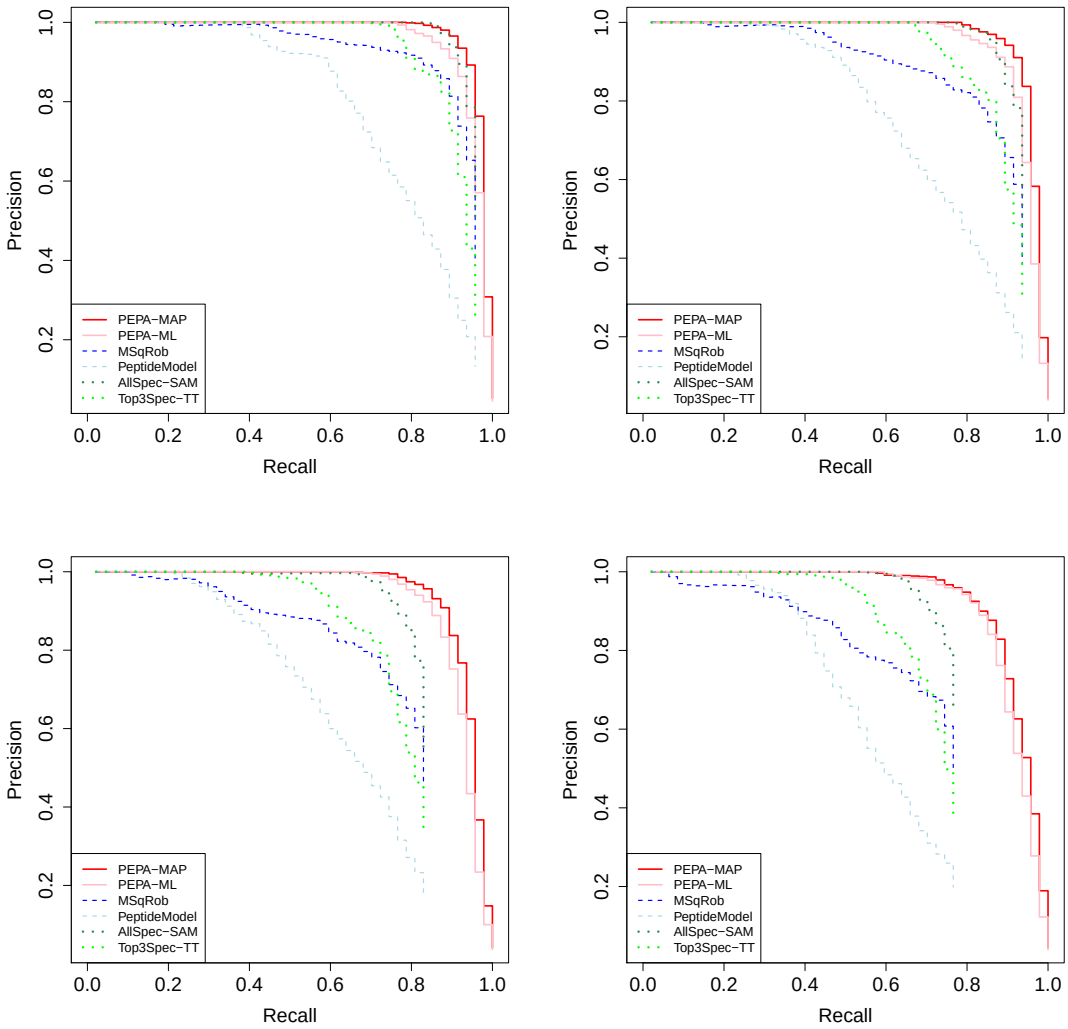

Fig. 12. PR curve on Exp1_R25_pept data with 40 (upper left), 80 (upper right), 160 (lower left) and 240 (lower right) artificially added shared peptides.

Bukhman, Yury V, Dharsee, Moyez, Ewing, Rob, Chu, Peter, Topaloglou, Thodoros, Le Bihan, Thierry, Goh, Theo, Duewel, Henry, Stewart, Ian I, Wisniewski, Jacek R *and others*. (2008). Design and analysis of quantitative differential proteomics investigations using lc-ms technology. *Journal of Bioinformatics and Computational Biology* **6**(01), 107–123.

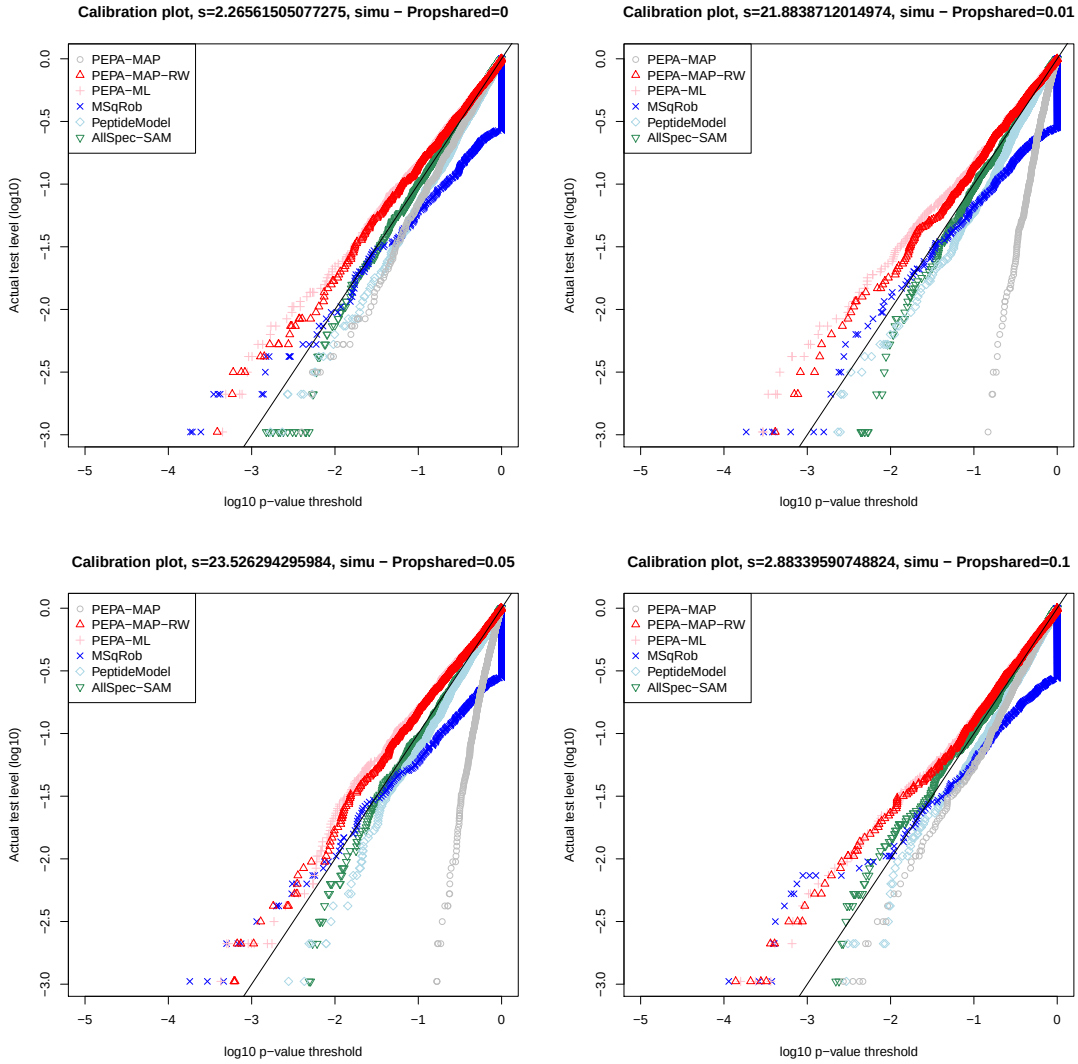

Fig. 13. Calibration plots for simulated data (1/2).

Choi, Meena, Chang, Ching-Yun, Clough, Timothy, Broudy, Daniel, Killeen, Trevor, MacLean, Brendan and Vitek, Olga. (2014). Msstats: an r package for statistical analysis of quantitative mass spectrometry-based proteomic experiments. *Bioinformatics* **30**(17), 2524–2526.

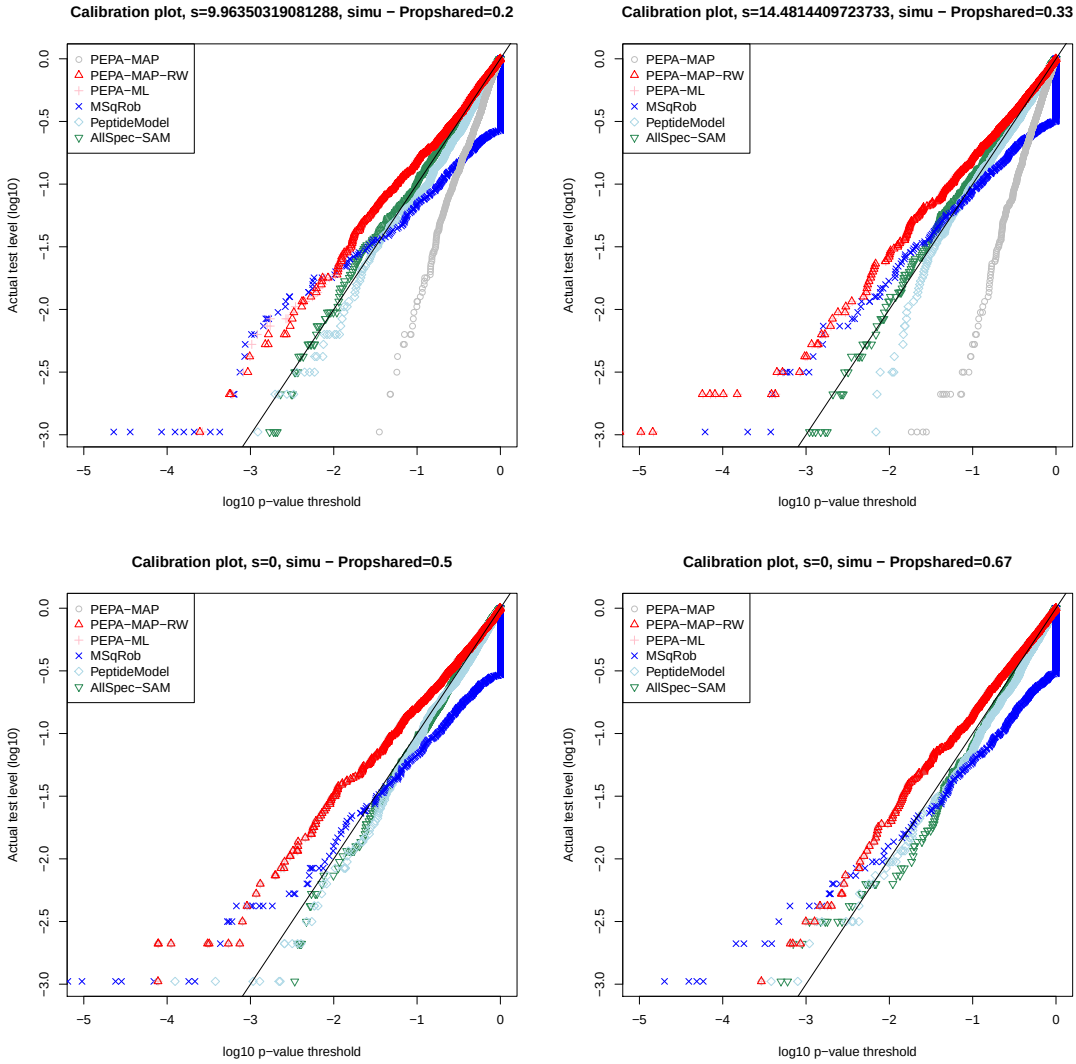

Fig. 14. Calibration plots for simulated data (2/2).

Clough, Timothy, Key, Melissa, Ott, Ilka, Ragg, Susanne, Schadow, Gunther and Vitek, Olga. (2009). Protein quantification in label-free lc-ms experiments. *Journal of proteome research* **8**(11), 5275–5284.

Clough, Timothy, Thaminy, Safia, Ragg, Susanne, Aebersold, Ruedi and Vitek,

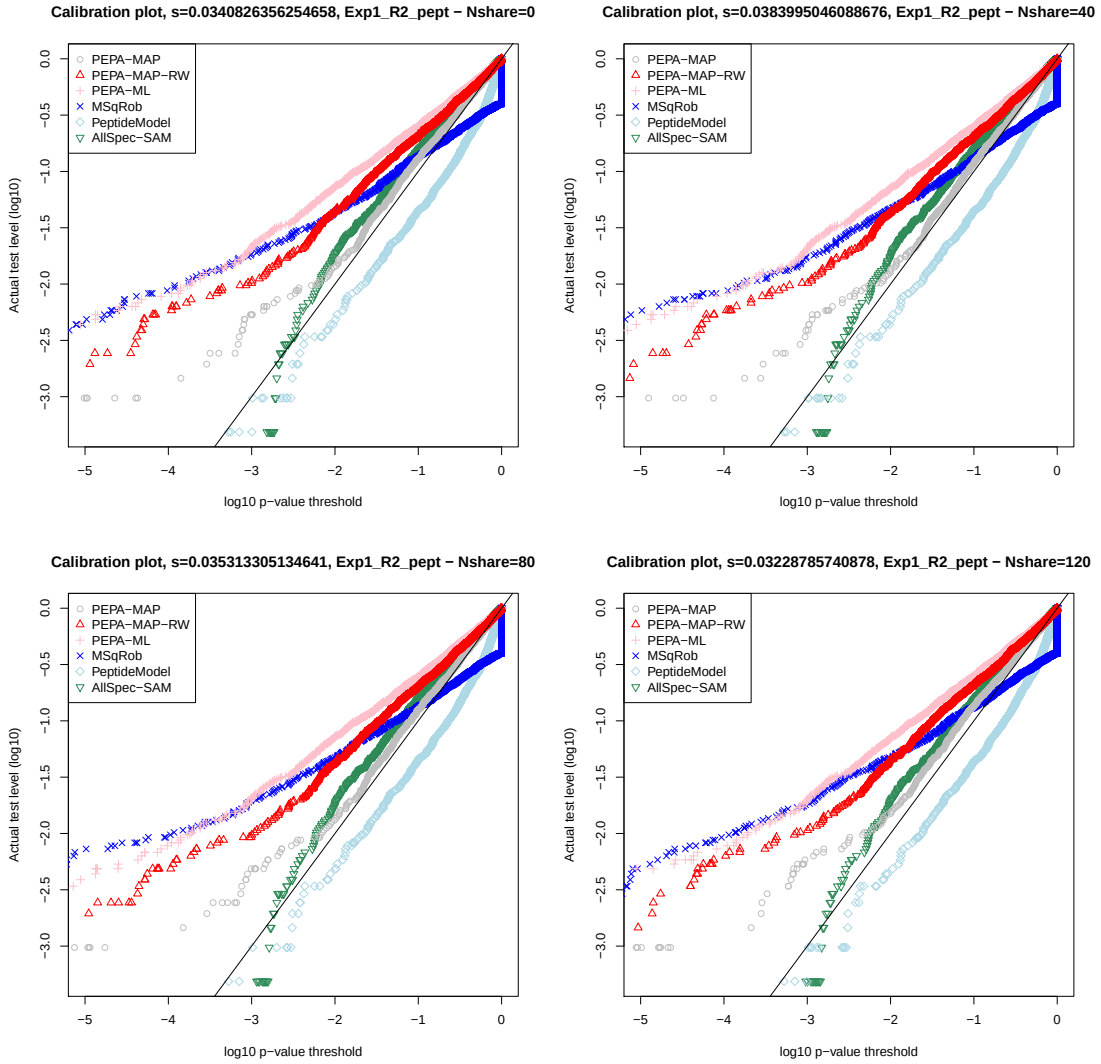

Fig. 15. Calibration plots for Exp1_R2_pept data (1/2).

Olga. (2012). Statistical protein quantification and significance analysis in label-free lc-ms experiments with complex designs. *BMC bioinformatics* **13**(16), S6.

Cox, Jürgen and Mann, Matthias. (2008). Maxquant enables high peptide identification rates, individualized ppb-range mass accuracies and proteome-wide protein quantification. *Na-*

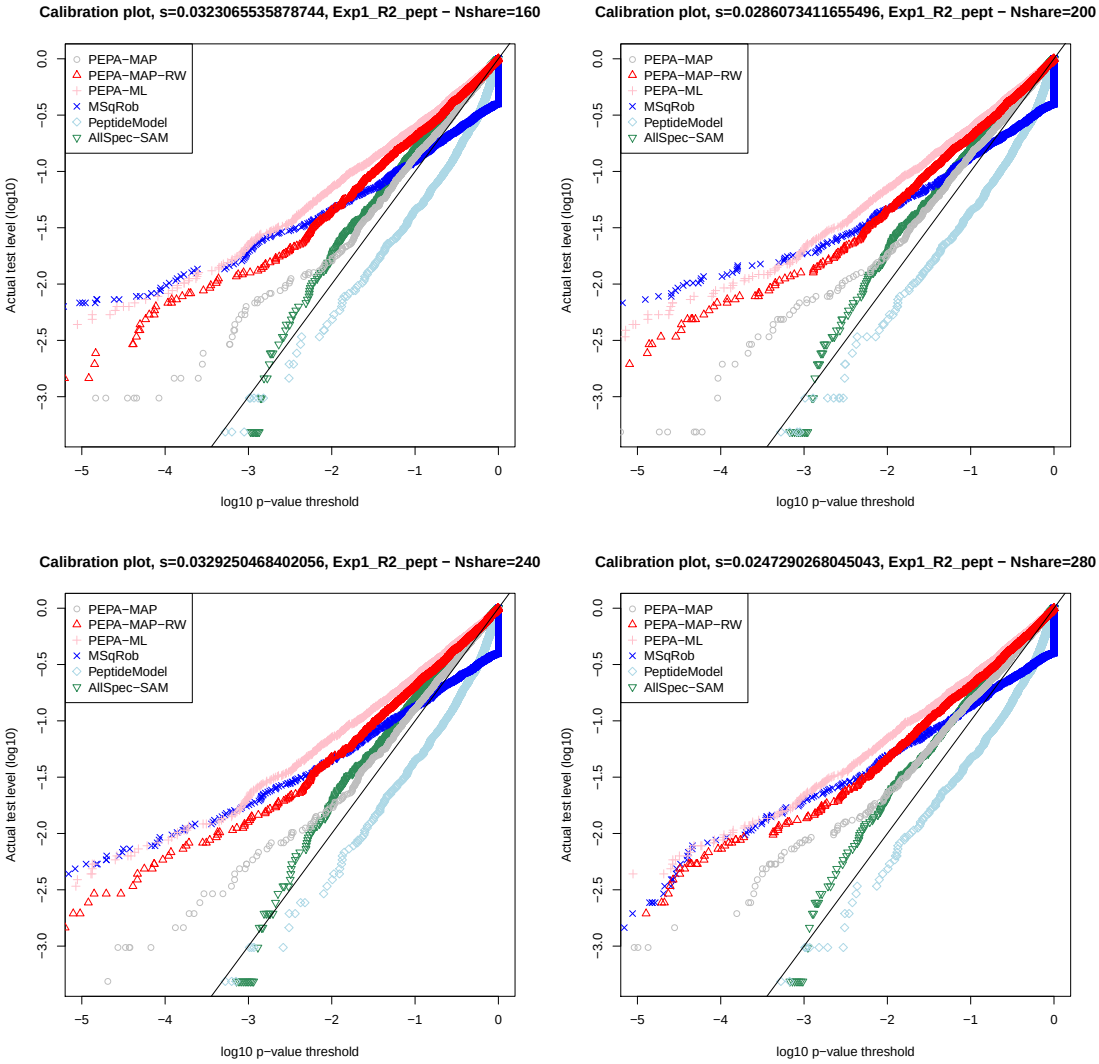

Fig. 16. Calibration plots for Exp1_R2_pept data (2/2).

ture biotechnology **26**(12), 1367–1372.

Dost, Banu, Bandeira, Nuno, Li, Xiangqian, Shen, Zhouxin, Briggs, Steven P and Bafna, Vineet. (2012). Accurate mass spectrometry based protein quantification via shared peptides. *Journal of Computational Biology* **19**(4), 337–348.

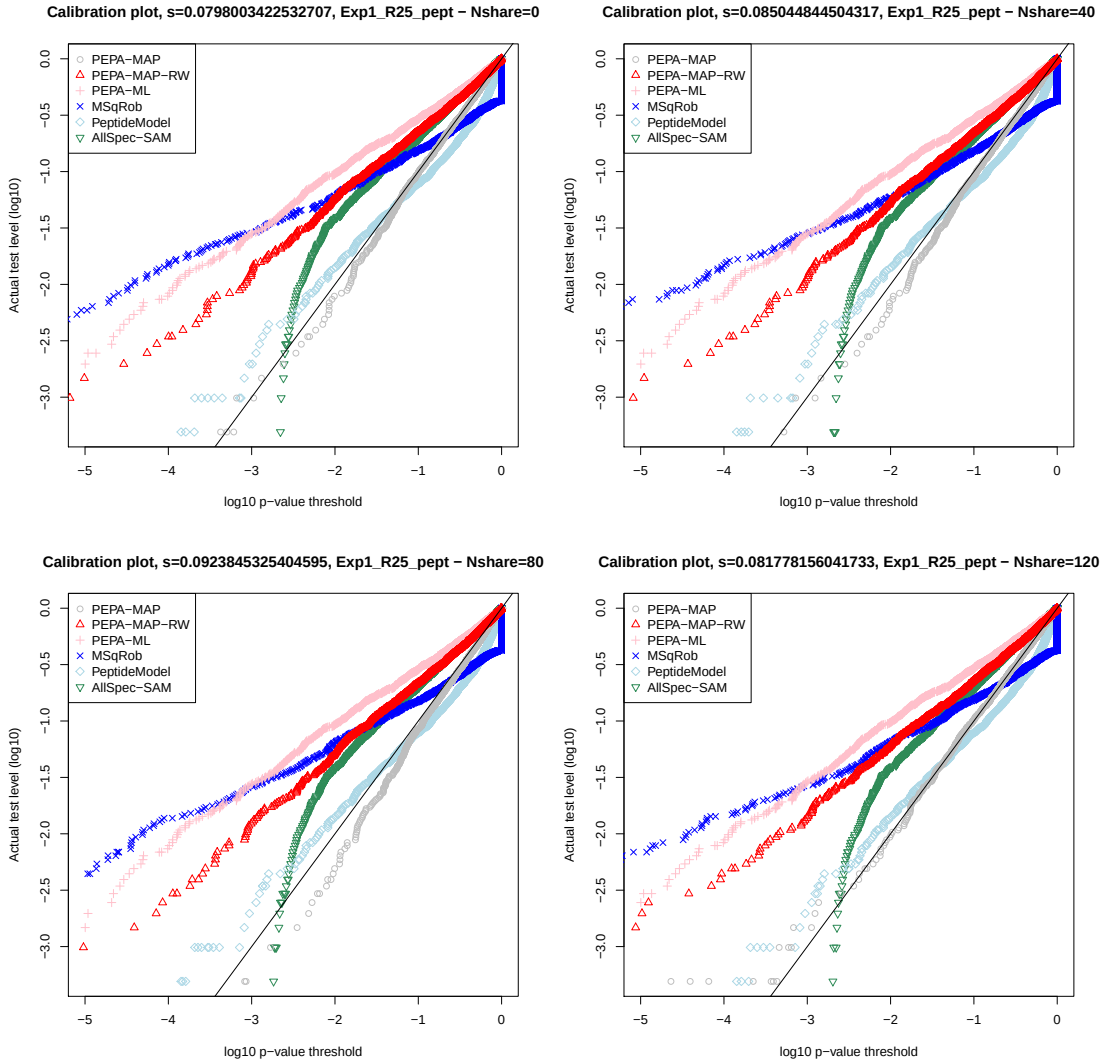

Fig. 17. Calibration plots for Exp1_R25_pept data (1/2).

Gerster, Sarah, Kwon, Taejoon, Ludwig, Christina, Matondo, Mariette, Vogel, Christine, Marcotte, Edward M, Aebersold, Ruedi and Bühlmann, Peter. (2014). Statistical approach to protein quantification. *Molecular & cellular proteomics* **13**(2), 666–677.

Giai Gianetto, Quentin, Combes, Florence, Ramus, Claire, Bruley, Christophe,

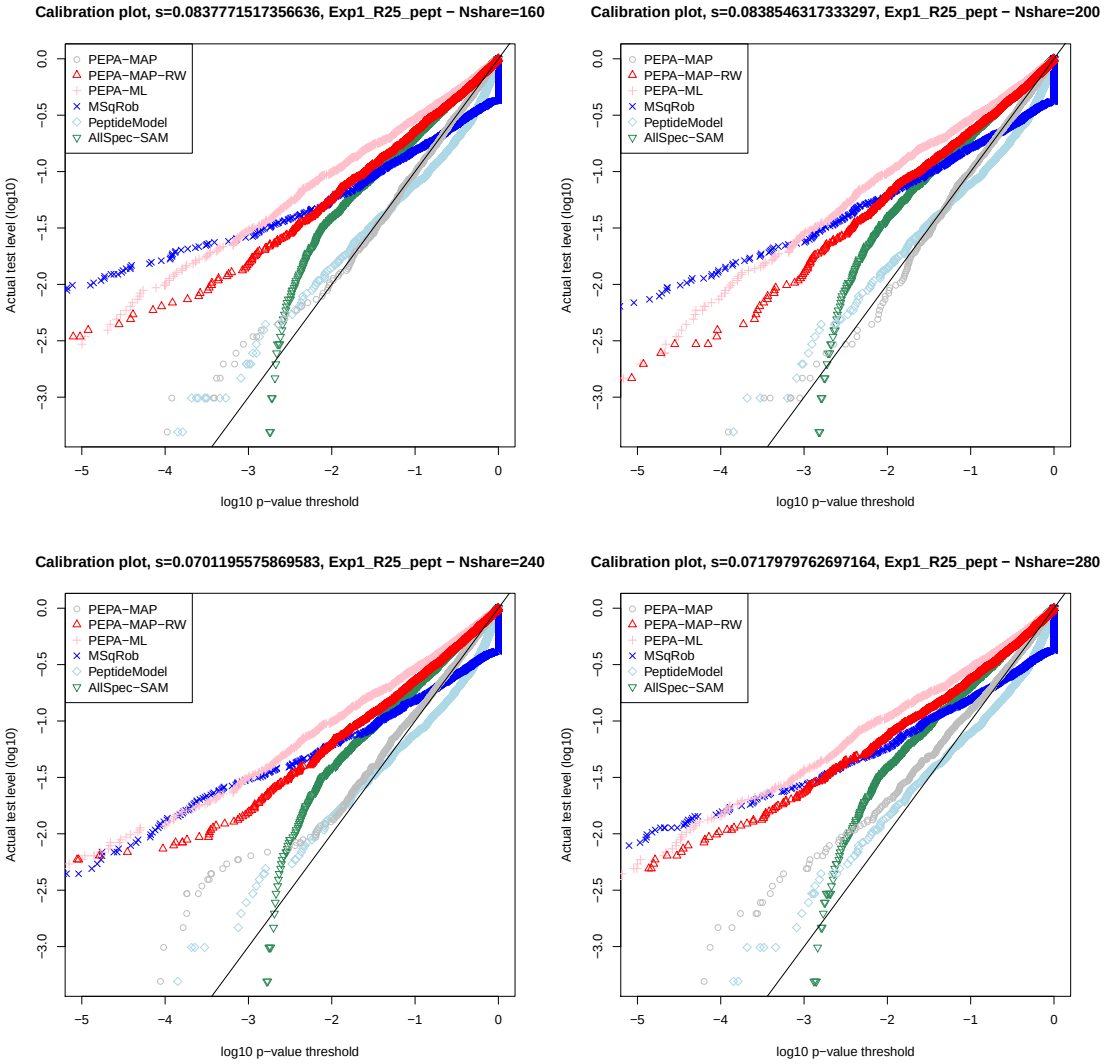

Fig. 18. Calibration plots for Exp1_R25_pept data (2/2).

Couté, Yohann and Burger, Thomas. (2016*a*). Calibration plot for proteomics: A graphical tool to visually check the assumptions underlying fdr control in quantitative experiments. *Proteomics* **16**(1), 29–32.

Giai Gianetto, Quentin, Couté, Yohann, Bruley, Christophe and Burger, Thomas.

(2016*b*). Uses and misuses of the fudge factor in quantitative discovery proteomics. *Proteomics* **16**(14), 1955–1960.

Goeminne, Ludger JE, Argentini, Andrea, Martens, Lennart and Clement, Lieven. (2015). Summarization vs peptide-based models in label-free quantitative proteomics: performance, pitfalls, and data analysis guidelines. *Journal of proteome research* **14**(6), 2457–2465.

Goeminne, Ludger JE, Gevaert, Kris and Clement, Lieven. (2016). Peptide-level robust ridge regression improves estimation, sensitivity, and specificity in data-dependent quantitative label-free shotgun proteomics. *Molecular & Cellular Proteomics* **15**(2), 657–668.

Harchaoui, Zaid, Bach, Francis and Moulines, Eric. (2008, April). Testing for Homogeneity with Kernel Fisher Discriminant Analysis. working paper or preprint.

Lazar, Cosmin, Gatto, Laurent, Ferro, Myriam, Bruley, Christophe and Burger, Thomas. (2016). Accounting for the multiple natures of missing values in label-free quantitative proteomics data sets to compare imputation strategies. *Journal of proteome research* **15**(4), 1116–1125.

Pham, Thang V, Piersma, Sander R, Warmoes, Marc and Jimenez, Connie R. (2010). On the beta-binomial model for analysis of spectral count data in label-free tandem mass spectrometry-based proteomics. *Bioinformatics* **26**(3), 363–369.

Podwojski, Katharina, Eisenacher, Martin, Kohl, Michael, Turewicz, Michael, Meyer, Helmut E, Rahnenführer, Jörg and Stephan, Christian. (2010). Peek a peak: a glance at statistics for quantitative label-free proteomics. *Expert review of proteomics* **7**(2), 249–261.

Ramus, Claire, Hovasse, Agnès, Marcellin, Marlène, Hesse, Anne-Marie, Mouton-Barbosa, Emmanuelle, Bouyssié, David, Vaca, Sebastian, Carapito, Christine,

Chaoui, Karima, Bruley, Christophe *and others*. (2016). Benchmarking quantitative label-free lc–ms data processing workflows using a complex spiked proteomic standard dataset. *Journal of Proteomics* **132**, 51–62.

Schwanhäusser, Björn, Busse, Dorothea, Li, Na, Dittmar, Gunnar, Schuchhardt, Johannes, Wolf, Jana, Chen, Wei and Selbach, Matthias. (2011). Global quantification of mammalian gene expression control. *Nature* **473**(7347), 337–342.

Silva, Jeffrey C, Denny, Richard, Dorschel, Craig A, Gorenstein, Marc, Kass, Ignatius J, Li, Guo-Zhong, McKenna, Therese, Nold, Michael J, Richardson, Keith, Young, Phillip *and others*. (2005). Quantitative proteomic analysis by accurate mass retention time pairs. *Analytical chemistry* **77**(7), 2187–2200.

Silva, Jeffrey C, Gorenstein, Marc V, Li, Guo-Zhong, Vissers, Johannes PC and Geromanos, Scott J. (2006). Absolute quantification of proteins by lcmse a virtue of parallel ms acquisition. *Molecular & Cellular Proteomics* **5**(1), 144–156.

Smyth, Gordon K. (2005). Limma: linear models for microarray data. In: *Bioinformatics and computational biology solutions using R and Bioconductor*. Springer, pp. 397–420.

Ting, Lily, Cowley, Mark J, Hoon, Seah Lay, Guilhaus, Michael, Raftery, Mark J and Cavicchioli, Ricardo. (2009). Normalization and statistical analysis of quantitative proteomics data generated by metabolic labeling. *Molecular & Cellular Proteomics* **8**(10), 2227–2242.

Tusher, Virginia Goss, Tibshirani, Robert and Chu, Gilbert. (2001). Significance analysis of microarrays applied to the ionizing radiation response. *Proceedings of the National Academy of Sciences* **98**(9), 5116–5121.

van der Vaart, A. W. (2007). *Asymptotic Statistics*. Cambridge.

Wieczorek, Samuel, Combes, Florence, Lazar, Cosmin, Giai Gianetto, Quentin, Gatto, Laurent, Dorffer, Alexia, Hesse, Anne-Marie, Cout, Yohann, Ferro, Myriam, Bruley, Christophe *and others*. (2017). Dapar & prostar: software to perform statistical analyses in quantitative discovery proteomics. *Bioinformatics* **33**(1), 135–136.

Wiśniewski, Jacek R. (2016). Quantitative evaluation of filter aided sample preparation (fasp) and multienzyme digestion fasp protocols. *Analytical chemistry* **88**(10), 5438–5443.

[*Received August 1, 2010; revised October 1, 2010; accepted for publication November 1, 2010*]