A new take on missing value imputation for bottom-up label-free LC-MS/MS proteomics

Lucas Etourneau[1,2], Laura Fancello[1], Samuel Wieczorek[1], Nelle Varoquaux[2], and Thomas Burger[1]

[1]Univ. Grenoble Alpes, CNRS, CEA, Inserm, BGE UA13, ProFI FR2048, 38000, Grenoble, France
[2]TIMC, Univ. Grenoble Alpes, CNRS, Grenoble INP, 38000, Grenoble, France

November 9, 2023

Abstract

Label-free bottom-up proteomics using mass spectrometry and liquid chromatography has long established as one of the most popular high-throughput analysis workflow for proteome characterization. However, it produces data hindered by complex and heterogeneous missing values, which imputation has long remained problematic. To cope with this, we introduce Pirat, an algorithm that harnesses this challenge following an unprecedented approach. Notably, it models the instrument limit by estimating a global censoring mechanism from the data available. Moreover, it leverages the correlations between enzymatic cleavage products (*i.e.*, peptides or precursor ions), while offering a natural way to integrate complementary transcriptomic information, when available. Our benchmarking on several datasets covering a variety of experimental designs (number of samples, acquisition mode, missingness patterns, etc.) and using a variety of metrics (differential analysis ground truth or imputation errors) shows that Pirat outperforms all pre-existing imputation methods. These results pinpoint the potential of Pirat as an advanced tool for imputation in proteomic data analysis, and more generally underscore the worthiness of improving imputation by explicitly modeling the correlation structures either grounded to the analytical pipeline or to the molecular biology central dogma governing multiple omic approaches.

Keywords: (1) Mass Spectrometry Based Proteomics; (2) Missing Value Imputation; (3) Missing Not At Random; (4) Multi-Omic Imputation; (5) Penalized Likelihood Estimation

1 Introduction

Bottom-up label-free LC-MS/MS (tandem mass spectrometry coupled with liquid chromatography) stands out as one of the most widely used method to characterize the proteome of biological samples. Analysing the massive amount of MS signals generated by such assays to produce an abundance matrix (a data table storing the abundance of each protein measured in each replicate sample, where samples are listed row-wise and proteins column-wise) requires robust, mathematically well-grounded and fast-evolving software, which development raises computational and theoretical challenges. Notably two difficulties are specific to the bottom-up label-free workflow: First, protein abundances are not directly measured, as sample preparation involves proteolysis (*i.e.*, the cleavage of each protein into several *peptides* using an enzyme like trypsin) and as MS measurements require the peptides to be ionized beforehand (they are then referred to as *precursor ions*, or simply *precursors*). Thus, inferring the identity and quantity of proteins requires two aggregation steps: from precursors to peptides, and from peptides to proteins. Second, raw LC-MS/MS data are hampered by the presence of a large number of missing values (MVs) in the abundance matrices: the overall MVs rate can reach 50% [1] (whether that is at the level of precursor, peptide, or protein abundances); and at least 50% of peptides usually have at least one MV [2].

The origin of MVs is complex and multi-faceted: since [3], it has been customary to separate them into two types. The first one gathers all the MVs resulting from the various workflow imperfections (such as peptide ionization issues, enzymatic miscleavages, too complex samples, false peptide identifications, *etc.*) and which may affect proteins broadly randomly, regardless of their abundance. The second category corresponds to the censorship mechanism resulting from the instrumental limit: precursors with an abundance below this limit yield MVs. A major issue with this censorship relates to its stochastic and dynamic nature, as the MS range changes along the acquisition process [4]. As a downside of the dramatic proportion and complex nature of MVs, analysing only fully observed biomolecules (be them precursors, peptides or proteins) is usually not considered, as it would discard too much biologically relevant information.

While it is theoretically safer to conduct the analysis keeping MVs as such [5] (*e.g.*, it is possible to identify differentially expressed proteins using models that cope with MVs [6, 7, 8, 9, 10]), many downstream investigation techniques require in practice to impute the MVs first (*i.e.*, to estimate the missing abundances), despite the risk of introducing biases in the original data. This is why, finding methods to accurately impute the manifold of MVs has long been a key challenge of computational proteomic research.

Based on a decade old and vast literature, some consensus about proteomic MV imputation has emerged. First, imputing missing protein abundances has been demonstrated to be suboptimal [1] as a result of the effect of the peptide-to-protein aggregation operator on missing values. However, whether imputation should be performed at precursor- or peptide-level has not been sorted out yet. Second, it is insightful to distinguish MVs according to the following well-acknowledged statistical categories [5]: Missing Completely At Random (MCAR), where the probability that a value is missing does not depend on any observed or missing value in the data; Missing At Random (MAR), where the same probability may only depend on observed values; and Missing Not At Random (MNAR) in any other case. Accordingly, in bottom-up label free proteomics, MVs resulting from the lower instrumental limit are classically assumed to follow an MNAR mechanism [3, 11, 1], while other MVs, in absence of sufficiently refined MAR-based description, are modelled as MCARs. Third, on the MVs classified as MCARs and on those classified as MNARs, the algorithms providing the best imputations are not the same [1]. This is notably why their co-existence has motivated the development of meta-imputation tools, *i.e.*, algorithms taking as input one or several imputation methods(s), and delivering as output a more refined imputation result (*e.g.*, [12, 13, 14, 15, 16, 17]); as well as of diagnosis tools capable of proposing an imputation strategy tailored to the data specificities [18, 19]. Regardless of their practical interest, leveraging those approaches is only possible if classical imputation methods are available in the first place, which explains why we hereafter restrict to them.

Numerous as exhaustive as possible reviews propose comparisons between a host of imputation algorithms (see for example [1, 3, 11, 20, 2, 21], as well as Section 4.3). Although those works generally concur on the least-accurate approaches, they do not on the most accurate ones, which depend on the dataset or on the experimental design. Moreover, while MCAR/MAR-devoted methods have established their robustness in many scientific domains where MVs have hardly specific behaviors, the MNAR-devoted methods used in proteomics are often simple and univariate, thus poorly informative [10] and leading to larger imputation errors. A notable exception is msImpute [22], which very recent publication has unveiled promising results. Contrarily to most anterior methods, msImpute simultaneously tackle MCARs and MNARs, as it estimates the type of each MV, and interpolates MAR and MNAR imputation distributions accordingly. Yet, some parameters (as the barycenter weights or the MNAR assumption) are fixed in a predetermined manner, which may hinder the generalization capabilities of the approach. Pushing the logic a step further, few works [6, 23] propose to bypass any MCAR/MNAR distinction, as the relevant model should directly estimate the probability of missing depending on the intensity, thus in a full MNAR setting. Among them PEMM [6], has been an important source of inspiration for this work. It proposes a penalized likelihood model involving a random left-censoring mechanism of the mean and covariance matrix of the protein-level data, which estimation yields natural parametric imputations. Unfortunately, this essentially theoretical proposal has shown many practical limitations, which explains its scarce use on real proteomics data.

More broadly, proteomic MV imputation aims at filling gaps in otherwise difficult to interpret biological data, as does multi-omic data integration: This field has elaborated on the natural assumption that a single omic modality contains partial-only information which can be compensated for if multiple omics technologies can be applied to same or related samples. To do so, it is often proposed to estimate a common latent distribution modelling the underlying biological phenomena (classically using either matrix factorization [24, 25, 26, 27] or deep learning [28, 29]). Then, the latent model is instrumental for a variety of tasks, such as pathway identification [26, 24, 27], sample extrapolation [26, 29, 28], gene-set analysis [27], and possibly imputation [30], even though none of the available methods was specifically developed (and tested) to fulfil this task in the challenging context of bottom-up label-free LC-MS/MS data. We have however remarked a recent (still unpublished) attempt by Gupta *et al.* [31] to train graph neural network in a gene-specific manner to impute protein-level MVs from such data, following a logic broadly akin to the one presented here.

Our most important contribution is to propose a concrete route to improve the imputation of LC-MS/MS-based label-free bottom-up proteomic data by leveraging a well-established analytical and biochemical truism: precursors or peptides originating from the same protein should exhibit correlations that are insightful to guess the trend of a protein across several samples. In addition, as insightful correlations can also be exploited between different omics modalities, we have generalized the approach to incorporate quantitative transcriptomic data, hereby opening the path to multi-omic based imputation of proteomic data. Doing so is however not sufficient to tackle the MNAR issue. We have thus implemented these concepts using an imputation algorithm which estimates a single model for both censored and random MVs. Although the model roots on PEMM [6], its estimation is achieved using a completely different strategy, more stable, scalable, and without parameter-tuning or functional approximations, as to fit the concrete constraints of proteomic data analysts. The resulting software, referred to as Pirat (available on GitHub `https://github.com/prostarproteomics/Pirat`), signifi-

cantly outperforms all other imputation methods on a variety of tasks, datasets, and situations. All our results are reproducible with the code available on GitHub `https://github.com/TrEE-TIMC/Pirat_experiments`.

2 Results

2.1 Pirat: a novel imputation method for proteomics data

Pirat (standing for *Precursor or Peptide Imputation under Random Truncation*) software works as follows: Firstly, depending on the input data, it creates either Peptide Groups or Precursor Groups. Both are based on protein belonging (details in Section 4.1.2) and are abbreviated as PGs, whereas peptides or precursors in a same PG are qualified as *siblings*. For sake of readability, we hereafter refer to peptides only, precursors being mentioned only if they require specific processing. The resulting biochemically informed dependence graph can optionally be enriched with transcriptomic data: for any PG, one then appends the sample-wise abundance vector(s) of transcript(s) (details in Section 4.1.10).

Secondly, Pirat estimates over all the PGs a global missingness mechanism:

$$P(M_{i,j} = 1 | X_{i,j}, \Gamma) = \min\left(1, \exp(-\gamma_0 - \gamma_1 X_{i,j})\right) \ , \tag{1}$$

where $\Gamma = \{\gamma_0, \gamma_1\}$ is referred to as the *missingness parameter*, and where $M_{i,j}$ (respectively, $X_{i,j}$) denotes the missingness response (respectively, abundance value) of the i-th sample and the j-th peptide. The missingness parameter is estimated using a kernel regression (details in Section 4.1.4) as to fit the missingness pattern of each dataset.

Thirdly, Pirat estimates the same model as PEMM [6] (derived from Rubin's selection model [5]), yet using an original approach. Briefly, it aims at maximizing the penalized log-likelihood of observed data and missingness response, which amounts to maximizing the following quantity:

$$\log P(X_{\text{obs}} | \mu, \Sigma) + \log \mathbb{E}_{X_{\text{mis}} | X_{\text{obs}}, \mu, \Gamma} \left[P(M_{\text{mis}} | X_{\text{mis}}, \Gamma) \right] + Q_{K,\lambda}(\Sigma) \ ,$$

where μ and Σ are the mean and covariance matrix of a multivariate Gaussian distribution, Q is a penalty on Σ and subscripts "obs" and "mis" respectively refer to indices of the missing and observed parts of the dataset (see Section 4.1.1 for the detailed mathematical notations). In Pirat, this model is fitted on each PG independently, with respect to μ and Σ, considering Γ is known. Likewise, penalty hyperparameters K and λ are automatically set for each PG in the empirical Bayes framework (details in Section 4.1.6). To avoid the approximations of the missingness mechanism (defined in Equation 3) that hamper PEMM's results, we propose a tractable and differentiable lower bound of the above quantity. Therefore, the lower bound can be maximized with respect to μ and Σ using a quasi-Newton optimization procedure (L-BFGS [32]). Finally, to decrease computation costs and memory usage, as well as to enforce Σ positive definiteness, Pirat leverages an alternative parameterization of Σ based on its log-Cholesky factorization.

Fourthly, once the model is fitted, we propose to impute the MVs by their conditional mean with respect to observed values and their missingness response. The conditional mean being devoid of closed form, it is computed using standard Monte-Carlo integration [33].

2.2 Pirat outperforms state-of-the-art on differential analysis task

We first evaluate our method on several biomarker selection tasks where significantly differentially expressed proteins are sought. We rely on three benchmark datasets for which differentially expressed proteins are known as their relative abundances are controlled. They essentially involve spiked standard proteins in a complex yet constant biological background. These datasets have been produced with different MS acquisition modes (either data dependent or independent, a.k.a. DDA or DIA), at precursor- or peptide-level, and using different standard mixtures (see Section 4.2).

The feature selection is performed using the p-values resulting from the significance testing of the peptides' differential abundance, after imputing with our method (Pirat) as well as with 15 different state-of-the-art or popular methods (described in Section 4.3). Although receiving operating characteristic (ROC) curves are classically used to assess feature selection (they can be found in Sup. Mat. Figure 5), we have relied on precision-recall (PR) curves (Figure 1) instead. They provide broadly similar rankings, but the PR curves display the False Discovery Proportion (FDP), which directly relates to the FDR one often controls in proteomic experiments [34]. More precisely, FDR being classically controlled at 1% or 5%, we present on Figure 1 partial PR curves focused on the high precision (low FDP) region to better assess selection performances in this setting. Global area under the entire PR curve (AUCPR) are also given in Table 1.

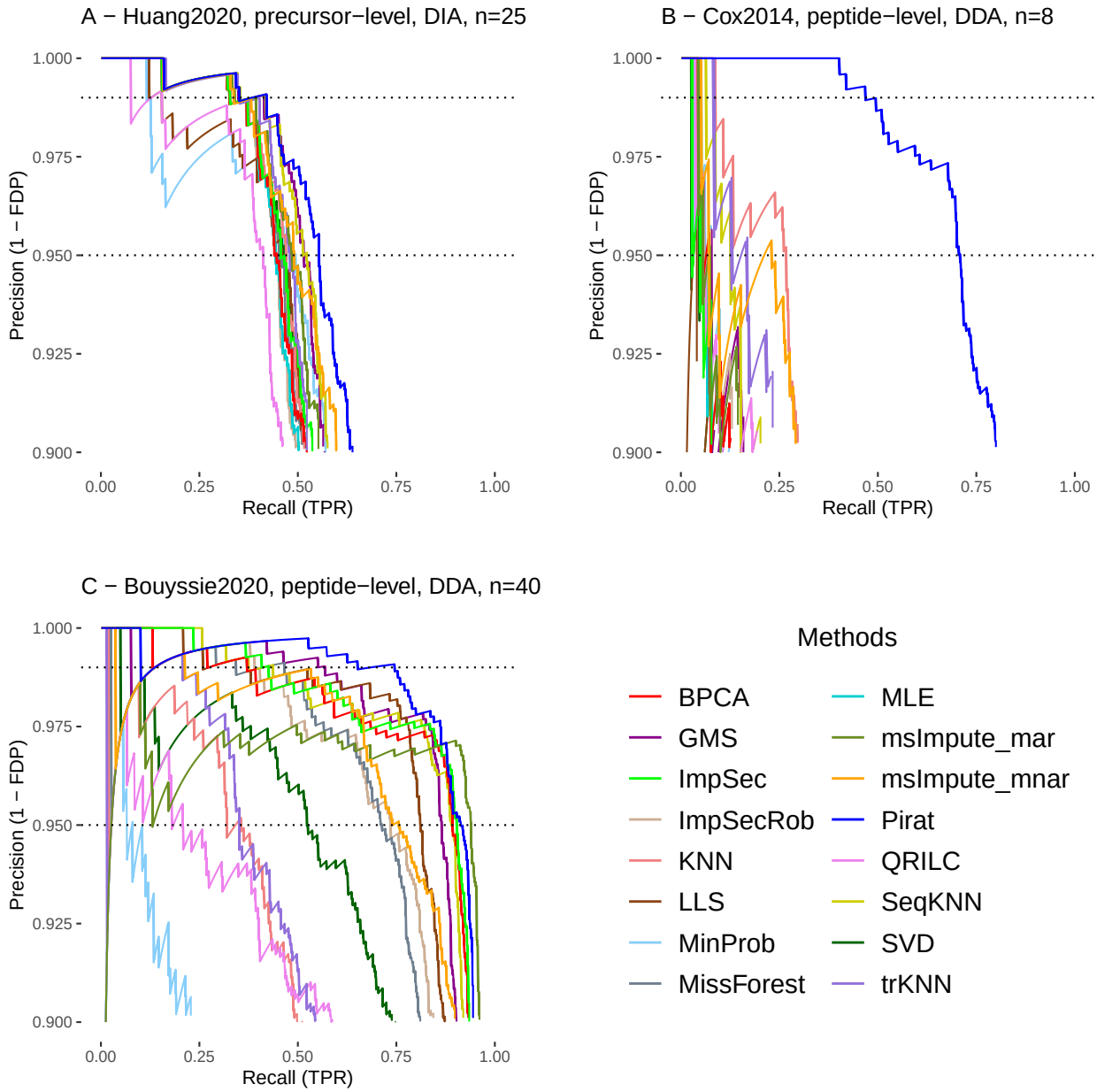


Figure 1: Differential abundance PR curves between 90% and 100% precision comparing Pirat to 15 imputation procedures on three benchmark datasets (A - Bouyssié2020, B - Cox2014, C - Huang2020), for which the name of the study, the imputation level (peptide or precursor), the type of acquisition (DIA or DDA), and the total number of replicates (n) are indicated in the subplot titles. The 1% and 5% FDP level are shown by the dotted lines.

On the three benchmark datasets, Pirat achieves the highest AUCPR (by a margin of 1.25% to 11.6% with respect to the second-best methods), thus indicating it best preserves the differential abundances without introducing false positives. Precisely, Pirat provides the best PR trade-off on Huang2020 and Bouyssie2020 in the low FDP region, and almost entirely dominates the other methods on Cox2014. Beyond Pirat performances, these experiments are insightful with many respects: Several highly popular methods like MissForest, KNN, SVD, or MLE have poor performances with respect to other less popular yet old methods like TrKNN or ImpSec. Moreover, both msImpute methods (one of the most recent methods, published in April 2023) are very efficient, yet not as much as indicated in their seminal paper [22]. The reason is twofold: First, although well ranked in terms of global AUCPR, which concurs with the published results, they are outperformed by many other methods in the low FDP region (for example, SeqKNN exhibits a better PR tradeoff at low FDP than both on Figure 1 A and C.) Second, they cannot impute missing values if less than 4 quantitative values are observed, so that the performances drop on datasets with fewer samples, like Cox2014 (more detailed explanations in Section 4.3). Conversely, Pirat clearly outperforms other methods on this dataset, which shows it can safely be applied in context of scarce samples. Most importantly, the lack of stability of other methods with respect to Pirat is worthwhile: Across the 3 datasets, the top scoring methods vary, first because the differences being sometimes

	AUCPR % (rank)		
Dataset	A - Huang2020	B - Cox2014	C - Bouyssié2020
BPCA	66.06 (10)	29.96	94.57 (5)
GMS	66.30 (9)	31.25	92.57 (7)
ImpSec	67.20 (7)	31.20	95.08 (2)
ImpSecRob	64.11	29.74	90.47 (9)
KNN	59.33	51.94 (6)	79.94
LLS	62.76	31.48	92.19 (8)
MinProb	72.56 (4)	37.62 (7)	63.96
MissForest	63.81	3.60	88.97
MLE	63.38	34.56 (9)	23.72
msImpute_mar	73.47 (3)	57.37 (5)	94.85 (3)
msImpute_mnar	75.62 (2)	75.68 (2)	94.77 (4)
Pirat	**76.89 (1)**	**87.25 (1)**	**97.22 (1)**
QRILC	68.08 (6)	62.41 (4)	84.66
SeqKNN	66.60 (8)	34.91 (8)	93.81 (6)
SVD	64.54	31.83 (10)	89.66 (10)
trKNN	68.35 (5)	70.49 (3)	81.68

Table 1: Global area under the precision-recall curves (and ranking into brackets) comparing Pirat with 15 imputation algorithms on a differential analysis task using three benchmark datasets (Bouyssié2020, Cox2014, Huang2020).

marginal, the ranking is sensitive to random fluctuations; but also because of the changes in the MCAR/MNAR proportion. On Bouyssie2020, the MAR/MCAR oriented methods, such as msImpute_mar, ImpSeq, BPCA, and SeqKNN perform best, whereas on Cox2014 and Huang2020 some MNAR oriented methods (like trKNN, msImpute_mnar, MinProb, and QRILC) display the best performances. Overall, only Pirat remains in the top 3 for all datasets, and it does so with the first rank. Such performances are achievable only because Pirat automatically estimates the missingness mechanism underlying each dataset (see Section 4.1.4). In a context where the MCAR/MNAR proportion is often unknown, it is of the utmost importance to have imputation algorithms which are resilient to the nature of missing values, so that a single method can be applied regardless of the data.

2.3 Pirat's performances are more stable with respect to MNAR ratio

Although Pirat displays the highest performances in an end-to-end task like biomarker discovery, a detailed analysis of its strengths and weaknesses is of interest to better outline its application scope. To do so, we hereafter refine the evaluations of Pirat by means of mask-and-impute experiments (*i.e.*, missing values are artificially added in real datasets, so that it is possible to measure the difference between the masked values and their imputed counterparts). To provide a baseline to the evaluations, we compare to the best ranked methods according to the previous experiments. However, we were not able to include msImpute_mnar, msImpute_mar, and trKNN despite their relatively good performances, because of their restriction on the number of observed values (see Section 4.3). Therefore, the subsequent evaluations compare Pirat with ImpSec (ranked 2 on the MAR dataset Bouyssie2020), SeqKNN (3 times in the top 10), QRILC, MinProb, and BPCA (1 time in the Top 5 and in the Top 10). Like Pirat, they can handle peptides with at least two observed values, while constituting reference methods for the MCAR (ImpSec, SeqKNN, and BPCA) and MNAR (QRILC and MinProb) scenarios.

In this experiment, we rely on datasets from two other LC-MS/MS proteomic analyses, detailed in Section 4.2. The first one (Capizzi2022 [35]) is composed of 10 samples processed at the peptide level (16% of MVs). The second one (Vilallongue2022 [36]) contains 8 samples processed at the precursor level (14% of MVs). For both datasets, we only consider the subset of peptides or precursors with no missing value, to which we introduce artificial MVs at a rate equal to that of the original dataset. Finally, we filter out peptides or precursors with one or zero remaining observed values. These artificial MVs are a mix of MCARs and MNARs, where the former ones are generated according to a Probit mechanism [37], as described in Section 4.5. Then, we impute the missing values using the algorithms to benchmark and compute their associated Root Mean Square Error and Mean Absolute Error (RMSE and MAE, see Section 4.5). We repeat this operation for different MNAR proportions (0%, 25%, 50%, 75%, and 100%) and seeds, leading to the curves of Figure 2.

We do not display QRILC and MinProb results in Figure 2 as their MAE and RMSE is 3 to 4 times higher than that of other methods (full plots available in Sup. Mat. Figure 6). For those methods, the difference of performances between mask-and-impute and differential analysis is striking although easily explainable: these MNAR-devoted methods are highly biased towards low abundances and work in a univariate setting (*i.e.*, apart

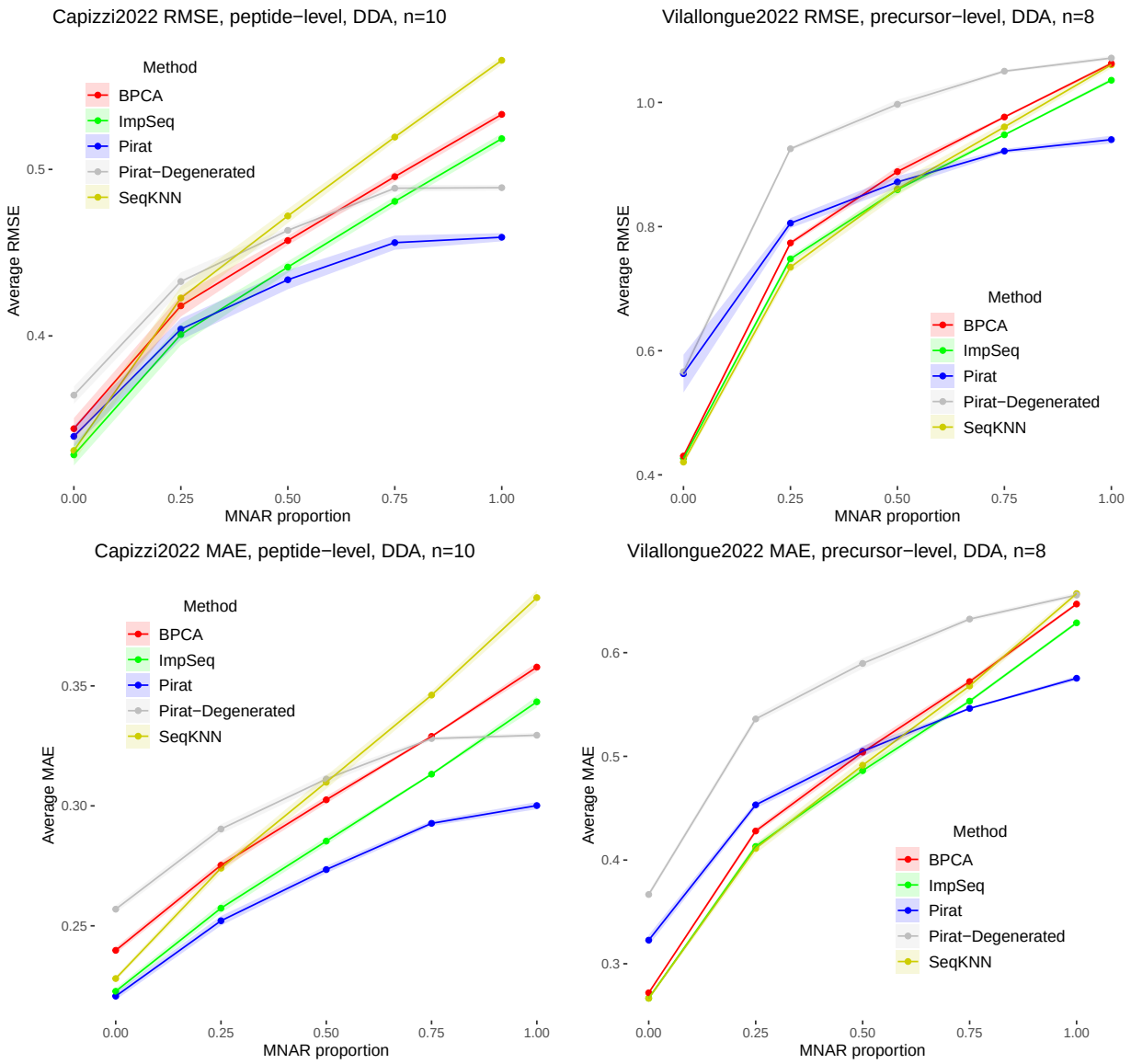


Figure 2: Average RMSE (top) and MAE (bottom) of best imputation methods (according to the previous results) in function of the proportion of MNAR values on Capizzi2022 (left) and Vilallongue2022 (right). The imputation level (peptide or precursor), the type of acquisition (DIA or DDA), and the total number of replicates (n) are indicated in the subplot titles. The errors averaged over 5 different seeds and margins correspond to standard deviations.

from the knowledge of a ill-defined instrumental limit, no information is used), thus resulting in poor quantitative predictions. Yet, such erroneous predictions may not hinder differential analysis when the low abundance bias is biologically relevant: for example, when a protein is not expressed in a phenotype, imputing its peptides with excessively low values is sometimes harmless.

All the other reference methods tested (PBCA, ImpSeq, and SeqKNN) have similar trends in terms of RMSE and MAE: As MCAR-devoted methods, the associated RMSE and MAE increase with the MNAR proportion (as opposed to QRILC and MinProb, see Sup. Mat. Figure 6). Although these three methods are broadly equally performing, ImpSec seems slightly but consistently more accurate, regardless of the MNAR ratio. As for Pirat, even if it is outperformed on Vilallongue2022 in scenarios with a majority of MCAR values, its performances are closer to the top MCAR methods than to MNAR ones on MCAR values. As for the MNARs, it clearly outperforms any other methods. Finally, over the entire range of MNAR/MCAR ratio, Pirat has more stable and more accurate MAE and RMSE trends.

Although a no-MNAR (or full-MCAR) scenario is not realistic in LC-MS/MS analyses, comparing the associated errors (MAE or RMSE) sheds an interesting light on Pirat's behaviour. When there are few to no MNAR values, the left-censoring model of Pirat loses its interest. In this case, the model boils down to a multivariate abundance one: the estimation can only leverage the correlations between sibling peptides. Conversely, BPCA, SeqKNN, and ImpSeq can leverage the entire feature-wise or sample-wise dependency structure. Hence, the

performances on MCARs seem directly related to the magnitude of the within-PG correlations (in the worst case of completely uncorrelated peptides, Pirat would amount to a univariate mean imputation, with likely poor performances). To verify this, we have developed a graphical tool (see Sup. Mat. A.4), which helps assessing the within-PG correlations. Using it on Capizzi2022 and Vilallongue2022 is insightful to understand the difference of performances: With good within-PG correlations, it is not surprising that Pirat performs excellently on Capizzi2022, even with small MNAR ratios. Conversely, with mediocre correlations, Vilallongue2022 is naturally more challenging for Pirat. Pushing the logic further, we evaluated Pirat on a dataset with within-PG correlations hardly larger than between-PG ones (see Sup. Mat. A.4). Without surprise, on this dataset, Pirat is competitive only in the 100% MNAR scenario. Estimating the precise MNAR ratio of real datasets has always been challenging [14, 1]. Even though the amount of MNARs is generally assumed to be significant, we recommend Pirat users to visually assess the within-PG correlations using the tool of Sup. Mat. A.4 as to check whether it will exploit a sufficient amount of common information to correctly impute the few MARs of the dataset. In case both the correlations and the MNAR ratio are anticipated to be low, it might be wiser to prefer a classical algorithm of the state-of-the-art, like for instance ImpSeq. This should notably be the case for peptidomics, top-down, or metabolomics experiments where enzymatic digestion is not used, *de facto* leading to the absence of PGs and thus of within-PG correlations. However, in classical bottom-up proteomics experiments, visualizing no differences between within-PG and random correlations (as in Sup. Mat. A.4) should raise awareness about the quality of the data. Indeed, low within-PG correlations implies that the siblings peptides have possibly different quantitative trends across samples, so that the subsequent protein roll-up will be hazardous, regardless of the imputation quality. Before imputation, it seems important in those cases to question the data acquisition than to opt for an algorithm that does not account for within-PG correlations. Conversely, for datasets where high within-PG correlations are verified using our graphical tool or where classical amounts of MNAR values are anticipated, Pirat will provide unmatched imputation quality.

Finally, Pirat's requirement for solid within-PGs correlations unveils a possible weakness: Weakly-covered proteins (*e.g.*, proteins for which too few peptides are available, which occur in all proteomic experiments) may not be correctly imputed, as no correlation can be leveraged. To assess the performances of Pirat in those situations, we have also considered a degenerated version of Pirat (referred to as Pirat-Degenerated on Figure 2), which processes peptides in a univariate way (*i.e.*, as if all PGs were of size 1). Then, Pirat essentially boils down to an imputation by a shifted mean (where the shift depends on the estimated γ_1 parameter, see Section 4.1.3). As expected, the imputation performances decrease significantly compared to Pirat: Although the trend is similar to that of Pirat thanks to the censorship model, the broadly constant gap between them confirms the importance of leveraging correlations to boost the imputation performances, while highlighting the risk of lower-quality imputation for weakly-covered proteins.

2.4 Improving peptide imputation for proteins with low coverage

We now evaluate various options to best process the peptides of weakly covered proteins. Precisely, we focus on PGs of size 1 (*i.e.*, proteins for which only a single MS evidence is available), thus the most difficult situation for Pirat. We refer to these PGs as *singleton PGs*. As borderline cases, there is no unique satisfactory way to process them: we therefore propose three possible approaches, among which the experimental context (rather than the purely numerical performances) should help choosing. The first one is to adopt a quantitative version of the two-peptide rule often used at peptide identification [38], which would lead to filter out singleton PGs with missing values. The second one is to leverage sample-wise correlations for imputation, with a similar logic as other algorithms (*e.g.*, like ImpSeq or BPCA). The third one is to rely on complementary transcriptomic analyses, opening the path to a multi-omic view on imputation.

To evaluate these approaches, we rely on two datasets involving both label-free proteomics and transcriptomics: Ropers2021 [39] and Habowski2020 [40]. Ropers *et al.* studied the effect of a synthetic growth switch on *E. coli* by performing paired proteomic and transcriptomic analyses of a wild-type and mutated strain at respectively 2 and 4 timepoints, thus resulting in 6 conditions (3 replicates each). In the mutated strain, protein and mRNA expressions are expected to decrease over time at different rate due to the differences of half-life of mRNA and proteins. As such, the samples have similar compositions leading to a relatively low MV rate at precursor-level (about 15%,) and we expect the transcriptomic-proteomic correlations to be exploitable thanks to the paired design.

In contrast, Habowski2020 results from unpaired transcriptomic and proteomic analyses of six types of mouse colon cells (3 replicates each for the proteomic experiments and 2 to 5 for the transcriptomic ones). Each condition being a different cell type, the sample composition in proteins varies dramatically between the different conditions. This results in an important amount of precursor-level MVs (which reaches 50%) with values missing on an entire condition (a.k.a. MEC [41]). Note that, compared to Ropers2021, this second dataset is crucial to (1) measure the possible negative impact of uncorrelated transcriptomic data in a multi-omic imputation approach; (2) assess the importance of within-PG correlations to correctly impute poorly observed peptides, containing mostly MNARs (notably MECs). See section 4.2 for more details on both datasets.

To evaluate the performance of these three approaches, we rely on a mask-and-impute experiment similar to the previous one, with the additional goal to perturbate the true distribution of MVs the least amount possible. For this reason, we use the entire set of precursors (instead of using precursors with no MVs) and we add only a small amount of MVs (1% of the total number of true MVs in each dataset). This results in a few values that can be used to assess the performance of the approaches. We thus consider only the two extreme scenarios: 0% and 100% MNARs (respectively named MCAR and MNAR scenarios in the following experiments) and repeat this process ten times to assess the overall distribution of errors in each scenario.

2.4.1 Quantitative two-peptide rule

Even though it is technically possible, one should wonder whether it is reasonable to impute missing values for singleton PGs. First, protein abundances for those PGs will directly result from imputation. Consequently, the quality of the imputation will weight a lot in downstream analysis (*e.g.*, differential analysis for biomarker discovery). Second, we cannot exploit the peptide-wise covariances for imputation, and thus suspect deterioration of the imputation for singleton PGs. To confirm this, we compare the median imputation error on singleton PGs with the error of PGs of size greater than two in the MNAR and MCAR settings. Results from this experiment (Figure 3, Sup. Mat. Figure 9) show that in most cases MVs from singleton PGs are indeed harder to impute than for all other PGs. Akin to the two-peptide rule in protein identification, a simple but effective strategy is to filter out singleton PGs that require missing value imputation to avoid errors in the imputation step to excessively affect downstream analysis. We call this strategy the *quantitative two-peptide rule* or simply the *two-peptide rule*.

Looking more closely at Figure 3, the results of Habowski2020 in the MNAR setting are intriguing: we do not observe that the pseudo-MNAR values in this dataset are harder to impute on singleton PGs than on other PGs. A plausible explanation is that the pseudo-MVs of non-singleton peptides (which we compute the error on) in this setting are located on peptides containing mostly MVs. To confirm this, we show in Sup. Mat. Figure 10 (A, B, C, D) the histograms of the number of MVs contained in non-singleton-peptides carrying pseudo-MVs. We see that histogram A, representing Habowski2020 MNARs, has a clearly different trend from other scenarios: It is the only situation where pseudo-MVs are, for the vast majority, carried by very scarce peptides containing mostly MECs. Hence, correlations estimated by Pirat are uninformative as they are based on very few values coming from other conditions, which concurs with the drop of performances over non-singleton PGs with respect to singleton PGs in the MNAR setting.

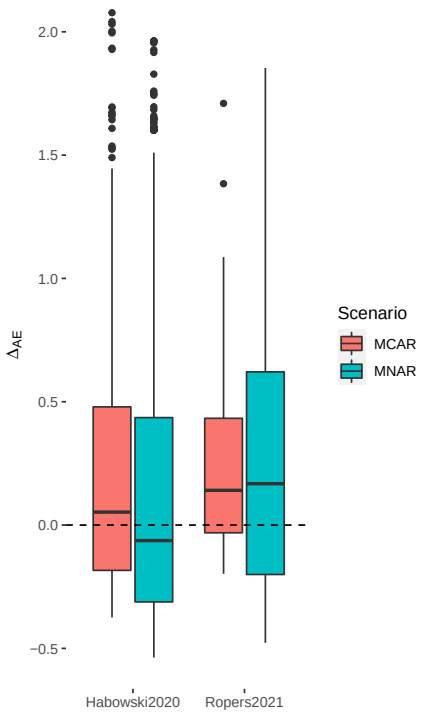


Figure 3: Boxplots of the distributions of the differences (denoted as Δ_{AE}) between (*i*) the absolute errors in singleton PGs and (*ii*) the median of the absolute errors in all other PGs, after Pirat imputation, for a given dataset (Habowski202 or Ropers2021) and MNAR / MCAR scenario. Hence, for a given dataset and MNAR setting, the part of the boxplot that is above zero corresponds to absolute errors that are greater than the median of absolute errors for non-singleton PGs.

Despite its unquestionable statistical cautiousness, the two-peptide rule approach is sometimes unsatisfactory. Beyond the case of Habowski2020 illustrated above, consider the rather frequent scenario of a confidently identified protein-specific peptide. In such cases, observing distinct observation/missingness patterns hints towards a promising putative biomarker, and consequently, the proposed quantitative version of the two-peptide rule is too stringent. To cope with this, we propose an alternative approach that leverages sample-wise correlations.

2.4.2 Sample-wise correlations

Pirat is based on the assumption the samples are independently distributed, see Section 4.1.3. Yet, other algorithms like ImpSec [42] assume differently and leverage dependencies between samples. This suggests the following *ad-hoc* strategy: First, impute all non-singleton PGs using Pirat; second, extract sample-wise covariance matrix of the completed data; third, use this matrix to impute the remaining MVs in singleton PGs (more details in Section 4.1.9). We refer to this extension as Pirat-S (sample-wise correlation).

Figure 4 shows, among others, the absolute errors of Pirat and Pirat-S. Clearly, Pirat-S improves upon Pirat in the MCAR setting for both datasets. This is not surprising: Pirat-S is conceptually close to ImpSeq, which has demonstrated excellent performances on MCARs (see Section 2.3). The improvement is more nuanced in the MNAR setting. Specifically, it seems to deteriorate the performances on Habowski2020, for the same reason as exhibited with the quantitative two-peptide rule, but on singleton peptides: As pseudo-MNAR MVs are mostly carried by empty peptides (see histograms E, F, G and H of Sup. Mat. Figure 10), too few values are left to derive informative correlations, notably when they do not originate from the same sample (as with MECs). Although Pirat-S seems to be a safe choice in all other cases, medium-only imputation performances on MNARs with MEC (or similar) patterns reveal a limit of this approach. Indeed, such patterns often result from proteins either absent or too close to the quantitation limit to be consistently measured, so that improving their imputation would *de facto* amount to extending the quantitation range of MS-proteomics. This is why, we propose a third alternative leveraging transcript/protein correlations.

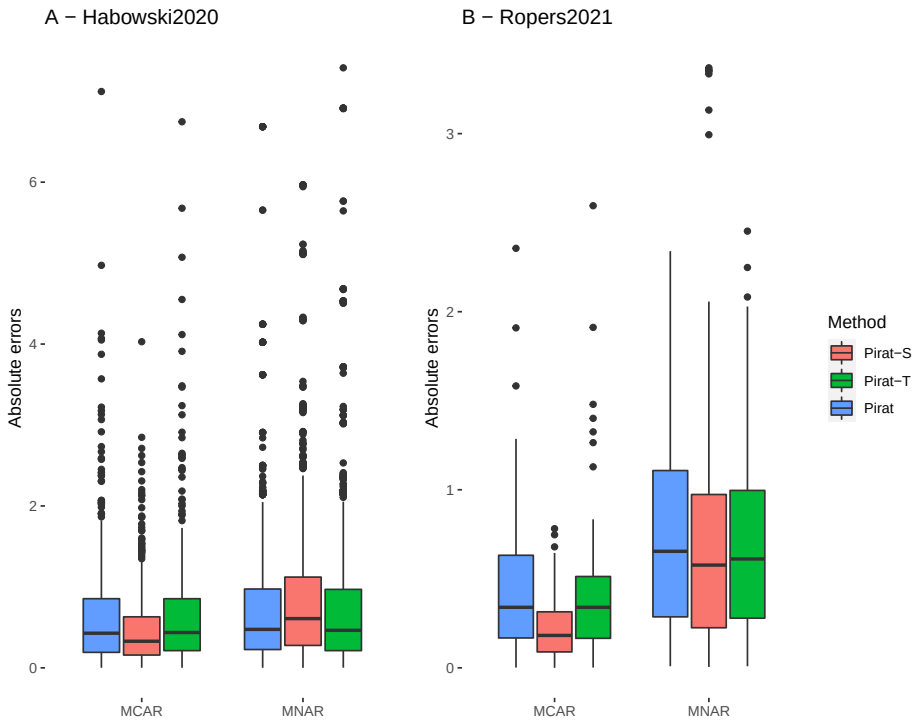


Figure 4: Boxplot of absolute errors for singleton PGs in A - Habowski202 and B - Ropers2021 datasets of Pirat, Pirat-S and Pirat-T.

2.4.3 Integrating transcriptomic information

The transcriptome-informed version of Pirat is termed Pirat-T and works as follows: For a given singleton PG, it simply concatenates the transcript abundance vector(s) to the peptide or precursor ones (more details are given in Section 4.1.10). Intuitively, the performance increment is expected to depend on the correlations between peptides and transcripts abundances. This is verified on Figure 4: the distribution error is shifted downwards on Ropers2021, where proteomic and transcriptomic samples are paired. On the contrary, the

differences seem marginal on Habowski2020, where samples are not paired, thus making the transcriptomic patterns hardly exploitable on the proteomic data. Importantly, although transcriptomics is not helpful in this case, it does not deteriorate the performances either. This indicates that even in case a researcher wonders about the transcriptomic data quality or about their correlations with proteomics, it is not riskier to incorporate them for downstream imputation: Pirat-T robustly integrates transcriptomic data for imputation, without over interpreting mRNA variations. When comparing Pirat-S and Pirat-T, it appears the latter do not outperform the former, indicating that in general, sample-wise correlations can be more robustly leveraged than transcriptomic-proteomic ones (see [43] for a discussion related to this topic). Although disappointing from a multi-omics integration perspective, this conclusion suffers one noteworthy exception: Habowski2020 on the MNAR setting, and despite the lack of paired design. Following the same logic as before, we conclude that Pirat-T is instrumental for datasets with MECs, as frequently observed when comparing different tissues. As a matter of fact, this confirms the intuition that complementary omic studies can be effective sources of information to explore more safely the proteome beyond the quantitation limit of mass spectrometry.

3 Conclusion

In this work, we show that Pirat outperforms previously published imputation methods across various experimental settings. On datasets endowed with differential abundance ground truth labels but with different missingness patterns and replicate numbers, Pirat consistently achieves the best PR trade-off (notably at low false discovery proportion), whereas other competitive methods display less stable and less accurate performances. Likewise, on mask-and-impute experiments, Pirat exhibits lower imputation errors (whatever the metrics, be it RMSE or MAE) for all scenarios with a majority of MNARs. As for the other hardly realistic scenarios with no to few missing values resulting from the instrument censorship, the performances are more variable, yet close to the best ones in the worst cases.

The only weakness our experiments have uncovered relates to Pirat's sensivity to the correlation structure, which has led us to provide the users' community with a diagnosis tool. More specifically, Pirat performances vary with the correlation level amongst sibling peptides. Even though "discordant" peptides among siblings are likely to be observed as a result of unknown post-translational modifications or of poorly quantified peptides [44], they should be too scarce to affect the correlation distribution. Therefore, too low correlations should raise the experimenter's awareness about the biochemical consistency of the assay. The above case suffers one exception, peptides without siblings (*i.e.*, in singleton PGs), for which the difficulty to impute using Pirat does not relate to the experiment quality. To process them, we propose three alternatives to the original approach: (*i*) the quantitative two-peptide rule, *i.e.*, discarding proteins with a single not fully quantified peptide (*ii*) If transcriptomic data is available, using Pirat-T can be insightful when correlations between peptide and transcript levels are anticipated (notably with paired assays), while it will at least not impair imputation otherwise. (*iii*) If transcriptomic is not available or not relevant for the proteomic study, sample-wise correlations can be leveraged using Pirat-S. However, this requires similar sample compositions to avoid power loss. With this regard, we draw attention to the marginal differences of global performances between Pirat, its two-peptide rule extension, Pirat-T, and Pirat-S: Once averaged on an entire dataset, the imputation error on singleton PGs alone does not weight as much as the experimental design to drive the imputation strategy. We therefore encourage Pirat's users to question their need with respect to proteins with low coverage before opting which strategy to use. Lastly, our experiments have reported Pirat's excellent performances on peptide- and precursor-level datasets, thus ensuring its safe use in both cases. These results unfortunately do not allow us to formulate an educated opinion about which level is most appropriate to impute. Closing this debate would require testing the two approaches with a metric authorizing their comparison.

Considering its performances, we believe future research in proteomics imputation will advantageously leverage Pirat's paradigms. Let us recall them: (*i*) the estimation of a global missingness mechanism inferred from the data; (*ii*) an explicit modeling of the biochemical dependencies that are known to result from the analytical pipeline; (*iii*) the possibility to include other omic measurements that are expected to correlate to the proteomic ones. Pushing further the developments in either of these three trends may require a different mathematical backbone than that of Pirat (*i.e.*, not necessarily involving the penalized likelihood model of Equation 4), however, multiple paths are promising: regarding the missingness mechanism, a natural parameter estimator reads as the maximum likelihood estimate (MLE) over all the PGs. However, it requires replacing the optimization procedure described in Section 4.1.7 by a joint MLE, which memory and computation cost would be prohibitive. Fortunately, as Pirat processes each PG independently, it is highly parallelizable if acceleration is needed. Also, other missingness patterns like Probit or Logit [23] can fit the data, but before investigating them, some guarantees are necessary about their log-likelihood (or any lower bound, see Section 4.1.7) being tractable and differentiable. From a biolgical standpoint, databases of known protein-protein interactions (PPI) are a source of supplementary correlation patterns, and incorporating them can be instrumental, notably for singleton PGs (as to find them pseudo-siblings capable of guiding the imputation). Likewise, known biologically or analytically

relevant interactions can also be encoded in the prior of the covariance matrix Σ. For example, the correlations between protein-specific sibling peptides are expected to be larger than those involving either shared peptides, transcripts or PPI, so that it would make sense to adapt accordingly the scaling matrix prior described in Section 4.1.6. Finally, Pirat proposes to incorporate transcriptomic data, but other high-throughput technologies provide access to gene-wise or PG-wise quantitative information, which integration into Pirat follows the same logic (broadly speaking, one just appends the quantitative vectors of the various omics into a single matrix). Therefore, Pirat naturally opens the path to more elaborated integrative multi-omic methods to futher tackle the challenge of proteomic data imputation.

4 Method

4.1 Pirat algorithm

4.1.1 Notations

Before diving into the description of the Pirat algorithm, let us introduce some notations.

- X: a complete matrix of peptide $\log_2$ abundances, with rows referring to samples and columns to peptides, of size $n \times p$.
- $X_{i,j}$: the abundance value in X of i-th row (sample i) and j-th column (peptide j); when i (respectively, j) is replaced by ".", one refers to the to j-th column of X (respectively, the i-th column of X): $X_{.,j}$ thus depicts the peptide vector (respectively, $X_{i,.}$ depicts the sample vector).
- M: an indicator matrix of the same size as X, reflecting whether an abundance value is missing or not. The same indexing notations as of X applies to M ($M_{i,j}$, $M_{.,j}$, $M_{i,.}$).
- $M_{i,j}$: missingness indicator of abundance value of j-th peptide in i-th sample (1 if missing, 0 otherwise).
- $X_{\text{obs}}, M_{\text{obs}}$ (resp. $X_{\text{mis}}, M_{\text{mis}}$): all values of X and M corresponding to observed (resp. missing) abundance values.
- x, m, *etc.*: We use uppercase letters (*e.g.*, $X_{i,j}$ and $M_{i,j}$) when considering random variables and lowercase ones (*e.g.*, $x_{i,j}$ and $m_{i,j}$) when considering their realisations.
- $\{i, \text{obs}\}$ (resp. $\{i, \text{mis}\}$): for a given sample i, set of (i, j) pairs such that $x_{i,j}$ is observed (resp. missing). For clarity, brackets are removed when using this notation as subscript.
- μ: the vector of the theoretical mean abundance of the p peptides.
- Σ: the covariance matrix of the peptide's abundances, of size $p \times p$. The same indexing notation as for matrices X and M ($\Sigma_{j,j'}$ indicates the covariance between peptides j and j').
- Γ: the set of parameters defining the missingness mechanism (see definition in Section 4.1.3).
- I denotes the identity matrix.

4.1.2 Creating peptide groups

We define peptide groups or precursor groups (PGs) according to protein belonging. It is usually assumed that sibling peptides and precursors will quantitatively behave similarly across treatment groups/biological conditions, even though the presence of isoforms, post-translational modifications, or miscleavage can impact these correlations [44]. Peptides from nested proteins form a unique PG and we duplicate peptides that are shared between PGs. We impute MVs in each PG separately and the multiple imputations obtained for shared peptides are averaged. For sake of readability, peptides or precursors are hereafter simply referred to as peptides, as PGs are processed similarly in both cases.

4.1.3 Model's assumptions

Our model relies on the following four main assumptions regarding the whole peptide dataset (here, X and M refer to all the peptides of all the PGs of the dataset):

1. The $X_{i,.}$ are i.i.d. and normally distributed with parameters μ and Σ.

2. The missingness mechanism is self-masked for all peptides [45], (*i.e.*, the probability of an abundance value being missing, knowing the entire sample's abundances, only depends on the abundance itself):

$$P(M_{i,j} = 1|X_{i,.}) = P(M_{i,j} = 1|X_{i,j}) \quad \forall i, j \tag{2}$$

3. The missingness responses for each peptide of a sample are independent conditionally to their abundance [45]:

4. The missingness mechanism can be written as [6] :

$$P(M_{i,j} = 1|x_{i,j}, \Gamma) = \min(1, \exp(-\gamma_0 - \gamma_1 x_{i,j})), \text{ where } \Gamma = \{\gamma_0, \gamma_1\} \text{ with } \gamma_1 \geq 0; \tag{3}$$

   We assume here that there is a minimum detection threshold $-\gamma_0/\gamma_1$ below which no peptide can be quantified, and that the probability for a peptide to be missing exponentially decays with log-intensity.

These assumptions ensure identifiability of μ, Σ, and Γ parameters in univariate and multivariate cases, as demonstrated in [45, 37].

4.1.4 Estimation of the missingness parameters

The set of parameters Γ over the whole peptide dataset is estimated as follows:

1. Sort the peptides by their observable mean, and denote by α_i the mean of the i-th peptide i in the ordered vector α;

2. For $i = \lceil \frac{k+1}{2} \rceil, \ldots, p + 1 - \lfloor \frac{k+1}{2} \rfloor$ (where $\lceil . \rceil$ and $\lfloor . \rfloor$ denotes the upper and lower rounding), compute the rolling average of the missingness percentage y_i over the ordered peptides, with a window of size k; which leads to $p - k + 1$ points (α_i, y_i);

3. Fit a linear model on the points (α_i, y_i) by ordinary least squares, and set the parameters of the missingness mechanisms 3 as the coefficients obtained.

At $k = 10$, we observe a clear and smooth decreasing trend $\log(y)$ with respect to α on two real datasets (Habowski2020 and Ropers2021) and a satisfying R^2 (representing goodness of fit, see Sup. Mat. Figure 11). This value is thus used in all experiments. The parameters γ_0 and γ_1 are hereafter considered fixed and known.

4.1.5 Penalized likelihood model

Considering the missingness parameters Γ known, we now propose to maximize the joint log-likelihood $\mathcal{L}$ of the observed values and of the missingness response (following Rubin's selection model [5]) with respect to the Gaussian parameters μ and Σ, iteratively for each PG:

$$(\hat{\mu}, \hat{\Sigma}) = \underset{\mu, \Sigma}{\operatorname{argmax}} \quad \mathcal{L}(\mu, \Sigma|X_{\text{obs}}, M, \Gamma) + Q_{K,\lambda}(\Sigma)\,, \tag{4}$$

where X and M hereafter refer to the abundance matrix and missingness response of a single PG, and where $Q_{K,\lambda}(\Sigma)$ is a penalty term on Σ with hyperparameters K and λ (see Section 4.1.6), as proposed in [6]. Hence, maximizing this penalized log-likelihood amounts to maximizing the posterior distribution of parameters μ, Σ, and Γ. Considering previous assumptions, the log-likelihood of the selection model decomposes as:

$$\begin{aligned}
\mathcal{L}(\mu, \Sigma|X_{\text{obs}}, M, \Gamma) &= \log P(X_{\text{obs}}|\mu, \Sigma) + \log P(M_{\text{obs}}|X_{\text{obs}}, \Gamma) + \log \mathbb{E}_{X_{\text{mis}}|X_{\text{obs}},\mu,\Gamma}\left[P(M_{\text{mis}}|X_{\text{mis}}, \Gamma)\right] && (5)\\
&= \log P(X_{\text{obs}}|\mu, \Sigma) + \log P(M_{\text{obs}}|X_{\text{obs}}, \Gamma) && (6)\\
&\quad + \sum_{i}^{n} \log \mathbb{E}_{X_{i,\text{mis}}|X_{i,\text{obs}},\mu,\Sigma}\left[\prod_{j \in \{i,\text{mis}\}} P(M_{i,j} = 1|X_{i,j}, \Gamma)\right].
\end{aligned}$$

4.1.6 Penalty over Σ

We estimate Σ and μ for each PG independently by optimizing Equation 4. Therefore, the penalty hyperparameters λ and K must be tuned automatically (as many times as PGs). To do so, we rely on a Bayesian interpretation of the penalty term $Q_{K,\lambda}(\Sigma)$: it can be viewed as an inverse Wishart prior of the covariance matrix Σ. This penalty can be rewritten as:

$$\begin{aligned}
Q_{K,\lambda}(\Sigma) &= \frac{1}{2} 2K \log(|\Sigma|) + \frac{1}{2} tr(2\lambda I \Sigma^{-1}) && (7)\\
&= \log[p_{\mathcal{W}^{-1}(2K-p-1, 2\lambda I)}(\Sigma)] + C\,,
\end{aligned}$$

where $p_{\mathcal{W}^{-1}(2K-p-1,2\lambda I)}$ denotes the density of an inverse Wishart distribution with parameters $(2K-p-1, 2\lambda I)$, and where C is constant with respect to Σ. Hence, K is related to the number of degrees of freedom in Σ estimate, and λ is related to the scaling matrix $2\lambda I$.

This leads us to empirically estimate the hyperparameters of each PG from the set of all fully observed peptides by relying on the empirical Bayes framework [46] where we leverage that, in the univariate case, the inverse Wishart distribution boils down the inverse-gamma one. Specifically, we compute their empirical variance and estimate the parameters of an inverse-gamma distribution (denoted α and β) through an MLE. Finally, by setting $K = \alpha + p$ and $\lambda = \beta$, the marginal prior distribution of each diagonal element of Σ (the variance of each peptide) is an inverse-gamma with parameters (α, β). Hence, the peptide's variances have the same prior distribution among all PGs (this property is instrumental as before looking at any data, one would not expect the variance of a peptide to depend on the PG it belongs to). Without relaxing this property, we increase α by a factor 2 to better constrain the estimation of the covariance matrix (following [6], which showed in various scenarios that increasing K improves the estimation of Σ).

4.1.7 Deriving a lower-bound of the penalized log-likelihood to estimate μ and Σ

A main issue regarding the estimation of μ and Σ by maximizing Equation 5, is that the expectation from the log-likelihood is not analytically tractable for any missingness mechanism. To compute it, Chen *et. al* [6] approximated Equation 3 as:

$$P(m_{i,j} = 1|x_{i,j}, \Gamma) = \exp(-\gamma_0 - \gamma_1 x_{i,j}) \tag{8}$$

and set any observed value below $-\gamma_0/\gamma_1$ as missing. We claim that this approximation makes the log-likelihood unbounded, and hinders convergence in some cases, according to the following results. Indeed, using this approximation, we have (see [6], Supp. Mat. Appendix D) $\forall i \in \{1, ..., n\}$:

$$\mathbb{E}_{X_{i,\text{mis}}|X_{i,\text{obs}},\mu,\Sigma}[P(M_{i,\text{mis}}|X_{i,\text{mis}}, \Gamma)] = \exp(-\gamma_0|\{i, \text{mis}\}| - \gamma_1 \mathbf{1}^T \tilde{\mu}_i + 1/2 \cdot \gamma_1^2 \mathbf{1}^T \tilde{\Sigma}_i \mathbf{1}), \tag{9}$$

where $\tilde{\mu}_i, \tilde{\Sigma}_i$ are the parameters of the normal distribution $X_{i,\text{mis}}|X_{i,\text{obs}}$. The above quantity diverges towards $+\infty$ if any coefficient of $\tilde{\Sigma}_i$ tends towards $+\infty$, which is not consistent with the definition of a probability distribution.

However, using Jensen's inequality, and the original missingness mechanism from Equation 3, we can derive a tractable and differentiable lower bound of the likelihood that can be optimized. Concretely, to estimate μ and Σ for a given PG, we propose to maximize the following lower bound, $\forall i \in \{1, ..., n\}$:

$$\begin{aligned}\log \mathbb{E}_{X_{i,\text{mis}}|X_{i,\text{obs}},\mu,\Sigma}[P(M_{i,\text{mis}}|X_{i,\text{mis}}, \Gamma)] &\geq \mathbb{E}_{X_{i,\text{mis}}|X_{i,\text{obs}},\mu,\Sigma}\Big[\sum_{j\in\{i,\text{mis}\}} \log P(m_{i,j} = 1|x_{i,j}, \Gamma)\Big] \\ &\geq \sum_{j\in\{i,\text{mis}\}} \mathbb{E}_{X_{i,j}|X_{i,\text{obs}},\mu,\Sigma}\big[\log P(m_{i,j} = 1|x_{i,j}, \Gamma)\big]\,. \end{aligned} \tag{10}$$

Let $W_i \sim X_{i,\text{mis}}|X_{i,\text{obs}}, \mu, \Sigma \sim \mathcal{N}(\tilde{\mu}_i, \tilde{\Sigma}_i)$, let w_i be the realisation of W_i, and let n_i^{mis} be the number of missing values in the i-th sample. Using the parametric model from Equation 3, $\forall j \in \{i, \text{mis}\}$:

$$\begin{aligned}\mathbb{E}_{X_{i,j}|X_{i,\text{obs}},\mu,\Sigma}\big[\log P(m_{i,j} = 1|x_{i,j}, \Gamma)\big] &= \int_{-\frac{\gamma_0}{\gamma_1}}^{+\infty} (-\gamma_0 - \gamma_1 w_{i,j})\phi(w_{i,j}|\tilde{\mu}_{i,j}, \tilde{\sigma}_{i,j})\,\mathrm{d}w_{i,j} \\ &= -\gamma_0\left(1 - \Phi\left(-\frac{\gamma_0}{\gamma_1}|\tilde{\mu}_{i,j}, \tilde{\sigma}_{i,j}\right)\right) - \gamma_1 \int_{-\frac{\gamma_0}{\gamma_1}}^{+\infty} w_{i,j}\phi(w_{i,j}|\tilde{\mu}_{i,j}, \tilde{\sigma}_{i,j})\,\mathrm{d}w_{i,j}\,,\end{aligned} \tag{11}$$

where $\phi(\cdot|\mu, \sigma)$ (resp. $\Phi(\cdot|\mu, \sigma)$) denotes the (resp. cumulative) distribution function of a normal distribution with parameters μ, σ. Then, using properties of truncated normal distribution, we have:

$$\begin{aligned}\mathbb{E}_{X_{i,j}|X_{i,\text{obs}},\mu,\Sigma}\big[\log P(m_{i,j} = 1|x_{i,j}, \Gamma)\big] &= -\left(1 - \Phi\left(-\frac{\gamma_0}{\gamma_1}\Big|\tilde{\mu}_{i,j}, \tilde{\sigma}_{i,j}\right)\right)\left(\gamma_0 + \gamma_1 \mathbb{E}\left[W_{i,j}\Big|W_{i,j} \geq -\frac{\gamma_0}{\gamma_1}\right]\right) \\ &= -\left(1 - \Phi\left(-\frac{\gamma_0}{\gamma_1}\Big|\tilde{\mu}_{i,j}, \tilde{\sigma}_{i,j}\right)\right)(\gamma_0 + \gamma_1\tilde{\mu}_{i,j}) - \gamma_1\tilde{\sigma}_{i,j}^2\phi\left(-\frac{\gamma_0}{\gamma_1}\Big|\tilde{\mu}_{i,j}, \tilde{\sigma}_{i,j}\right)\,.\end{aligned} \tag{12}$$

The expression of Equation 13 is differentiable with respect to parameters $\tilde{\mu}_{i,j}, \tilde{\sigma}_{i,j}$. These two are the parameters of the conditional distribution of $X_{i,j}$ with respect to $X_{i,\text{obs}}$. They are obtained by the linear combination of μ and the Schur complement of the covariance matrix of observed values in i-th sample. Schur complement is differentiable with respect to Σ so that $\tilde{\mu}_{i,j}$ and $\tilde{\sigma}_{i,j}$ are differentiable with respect to Σ and μ. Hence, we can

use any automatic differentiation tool, combined with optimization algorithm, to maximize the following lower bound of Equation 4:

$$\tilde{\mathcal{L}}(\mu, \Sigma, |X_{\text{obs}}, M, \Gamma) = \log P(X_{\text{obs}}|\mu, \Sigma) + \log P(M_{\text{obs}}|X_{\text{obs}}, \Gamma) + \sum_{i}^{n} \sum_{j \in \{i,\text{mis}\}} \mathbb{E}_{X_{i,j}|X_{i,\text{obs}},\mu,\Sigma}\big[\log P(m_{i,j} = 1|X_{i,j}, \Gamma)\big] + Q_{K,\lambda}\,. \tag{13}$$

Specifically, we use L-BFGS (a quasi-newton method) combined with Armijo backtracking line-search, implemented with Pytorch [47].

To ensure Σ positive-definiteness during the optimization process, we re-parametrize Σ by its log-Cholesky factorization [48]. Finally, to avoid ill-conditioned matrix that could cause numerical instabilities, we apply Tikhonov regularization method and subsequently add a fixed value $\epsilon_\Sigma = 10^{-4}$ to the diagonal of Σ. Doing so does not impact on the optimum solution, as the minimum variance observed in all the datasets processed so far has been $\geq 10^{-3}$.

4.1.8 Imputation

Once all parameters μ, Σ, and Γ are estimated, we impute missing values by their conditional mean. To do so, we use the following result [6]:

$$\mathbb{E}_{X_{i,\text{mis}}|X_{i,\text{obs}},M_{i,.},\mu,\Sigma,\Gamma}[X_{i,\text{mis}}] = \frac{\mathbb{E}_{X_{i,\text{mis}}|X_{i,\text{obs}},\mu,\Sigma}[P(M_{i,\text{mis}}|X_{i,\text{mis}}, \Gamma)X_{i,\text{mis}}]}{\mathbb{E}_{X_{i,\text{mis}}|X_{i,\text{obs}},\mu,\Sigma}[P(M_{i,\text{mis}}|X_{i,\text{mis}}, \Gamma)]} \tag{14}$$

The distribution on which the left expectation is computed is not accessible, however, the distributions of the right expectations are Gaussian. Hence, we use Monte-Carlo integration to estimate them and compute the conditional mean of $X_{i,\text{mis}}$. Note that, using the same approach, the order 2 moment of $X_{i,\text{mis}}$ can be computed by simply squaring it in the above equation, which gives access the variance of $X_{i,\text{mis}}$.

4.1.9 Sample wise correlation to impute PGs singleton

We propose in Pirat-S to leverage sample-wise correlations to impute peptides of singleton PGs. To do so, we first estimate missingness parameters Γ and hyperparameters K and λ on the whole dataset and impute only PGs of size > 1 with the classical version of Pirat, described above. Second, we compute the empirical sample mean and sample-wise covariance matrix of the transposed imputed part of the dataset. Finally, we impute the remaining MVs by their conditional mean with respect to observed values, only using the observed values and the empirical sample mean and covariance matrix.

4.1.10 Transcriptomic integration

Pirat-T extends Pirat by enabling the integration of transcriptomic quantitative information, when available, as to guide peptide imputation. To do so, it requires a dataset of $\log_2$ mRNA expressions of samples from the same phenotype(s) as the proteomic ones. However, increments in imputation performances may require paired transcriptomic and proteomic samples. It also requires a correspondence table between transcripts and PGs, for instance, based on their original gene(s). In this work, transcriptomics is essentially used to improve singleton PG imputation (see Section 2.4.3). However, Pirat's methodology is more versatile and the packaged code makes it possible to tune which PGs are imputed using complementary transcriptomic information, depending on their size.

Concretely, Pirat-T works as follows: for each PG, it integrates all the transcripts (i) for which non-zero values are measured in at least two conditions, and (ii) which are associated to at least one of the proteins of the PG. In the case proteomic and transcriptomic samples are paired, the log-count vectors of the transcripts are simply appended to the PGs. Otherwise, a condition-wise mean log-count vector is used instead. *In fine*, each mRNA log-count vector is processed like an additional (fully observed) peptide, and thus contributes to imputation depending on its correlation with the sibling peptide(s).

4.2 Datasets

To conduct our experiments, we rely on the following publicly available datasets:

- **Cox2014** [49]: a mixture of 48 recombinant human proteins that are available as an equimolar mixture (UPS1) or mixed at defined ratios spanning 6 orders of magnitude (UPS2) with respect to UPS1 (10, 1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}). UPS1 and UPS2 are separately digested and resulting peptides are spiked into a *E. coli* lysate, resulting in two distinct experimental conditions. Each condition is analysed four

times in data dependent acquisition (DDA, *i.e.*, a single precursor is iteratively selected for fragmentation and identification by the mass spectrometer according to its response level in the first MS acquisition), resulting in four technical replicates per conditions. We use the file `peptide.txt` available at repository PXD000279, containing peptide-level abundances.

- **Bouyssie2020** [50]: a mixture of UPS1 proteins spiked at different concentrations in the same yeast lysate. The dataset contains 10 conditions with 4 technical replicates each (each condition corresponding to a different UPS1 quantity, namely, 0.01-0.05-0.1-0.250-0.5-1-5-10-25-50 fmol, for 1μg of yeast lysate). Hence, unlike Cox2014, within each condition, all the UPS proteins are spiked in with identical concentrations. Replicates are analysed in DDA mode. We use peptide intensities from `allsamples_sum.xlsx` file available at repository PXD009815.

- **Huang2020** [51]: the UPS2 mixture spiked in tissue lysates from 25 mouse cerebellum samples. Five conditions (with 5 technical replicates each) are generated from spiked UPS2 proteins in known and different concentrations (namely 0.75-0.83-1.07-2.04-7.54 amol/μl) into mouse cerebellum samples. The LC-MS/MS runs are acquired by the data independent method (DIA, *i.e*, a set of precursors in a given mass to charge range are iteratively selected for fragmentation and identification, in order to cover the whole MS acquisition, regardless of the measured intensities). We use the published Spectronaut results in `Spike-in-biol-var-OT-SN-Report.txt` available at repository PXD016647, and imputed MS1 quantification at precursor level.

- **Capizzi2022** [35]: a study on Huntington's disease effect on axonal growth in mouse. Two conditions with 5 biological replicates each are compared using LC-MS/MS in a data dependent acquisition mode (DDA), representing wild-type vs genetically modified model. Details of sample preparation can be found in the original article. The accession number for this dataset is PXD023885. Raw data is processed in the same manner as described in the original paper to obtain peptide-level quantification table.

- **Vilallongue2022** [36]: a study on the influence of injury on visual targets in mouse. Five different tissue types are analysed by LC-MS/MS in DDA acquisition mode in independent quantification tables, with each time a control and an injured condition (4 biological replicates each). To test our method, we choose the two tissues that show the best and worse within-PG peptide correlations among the five (see A.4): respectively, the suprachiasmatic nucleus tissue (SCN), and the superior colliculus (SC). The details of sample preparation can be found in the original article. The accession number for this dataset is PXD029325. Raw data is processed in the same manner as described in original paper to obtain precursor-level quantification table.

- **Habowski2020** [40]: a study on the differentiation of mouse colon epithelial cells. Stem cells and 5 differentiated cells are analysed at proteomic and transcriptomic levels in an unpaired manner. Three biological replicates are used for each cell type in proteomic analyses. The transcriptome dataset is produced by Illumina paired-end stranded sequencing. The proteome dataset is produced by LC-MS/MS in DDA acquisition mode.

  - **Transcriptomic data processing**
    We download the raw sequencing data from GEO (GSE143915). Preprocessing and quality control are performed using the Trimmomatic [52] and FastQC tools respectively. Trimmed reads are aligned to the mouse GRCm39 genome assembly (filename given in Sup. Mat. Section A.2) by the STAR mapping software (version 2.7.8a) [53] provided with the Ensembl gene model given in Sup. Mat. Section A.2 for mapping splices. Read counts are generated using HTSeq (version 0.9.1; option -s no) [54]. DESeq2 version 1.22.1 [55] is used to generate normalized counts.
  - **Proteomic data processing**
    We download the raw spectra from ProteomeXchange (accession ID: PXD019351). Peptides and proteins are identified using Mascot (version 2.7.0.1, Matrix Science) searching the Ensembl protein database for *Mus musculus* GRCm39 (filename given in Sup. Mat. Section A.2) appended with an in-house classical contaminant database. Mascot search is performed with the following parameters: trypsin/P as enzyme and two missed cleavages allowed; precursor and fragment mass error tolerances at 10 ppm and 20 ppm, respectively. The following peptide modifications are allowed during the search: Acetyl (Protein N-term, variable) and Oxidation (M, variable). The Proline software [50] is used to validate identifications: conservation of rank 1 peptides, peptide length ≥ 7 amino acids, false discovery rate (FDR) of peptide-spectrum-match identifications $< 1\%$ as calculated by Benjamini-Hochberg procedure and minimum of 1 specific peptide per identified protein group. Peptide ion intensities are calculated from extracted mass spectrum intensities of all peptides and normalized using variance stabilizing transformation with Prostar [56]. A more detailed description of the parameters used in the quantification step are provided in Supplementary Table 1.

- **Ropers2021** [39]: a study on growth-arrested and wild type *E. coli* cells carrying a plasmid for glycerol production at various time points. A wild type and a modified strain are analysed in a paired manner at proteomic and transcriptomic levels. The modified strain is analysed at four different time points, resulting in four different conditions, and two others from the wild-type on the two first time points. Three biological replicates are made for each condition. The transcriptome dataset is produced by stranded sequencing on the Ion S5 using the Ion 540 chip. The proteome dataset is produced using LS-MS/MS in DDA acquisition model.

  - **Transcriptomic data processing**: Same procedure than for Habowski2020 is applied. GEO accession number is GSE168336. Trimmed reads are aligned to the reference *E. coli* K12 substrain MG1655 genome (Genbank assembly accession: GCA_000005845) provided with the Ensembl gene model given in Sup. Mat. Section A.2 for mapping splices.
  - **Proteomic data processing**:
    The same procedure than for Habowski2020 is applied, except for the following: Raw spectra are downloaded from ProteomeXchange (accession ID: PXD024231). Peptides and proteins are identified by searching the same Ensembl protein database used in transcriptomic analysis (for this Ensembl genome assembly of *E. coli* the protein identifier is identical to the transcript identifier), appended with an in-house classical contaminant database, the plasmid pCL1920 protein sequences and the protein sequences of the GPD1 and GPP2 yeast genes which were cloned into it (Uniprot accessions: Q00055 and P40106). Carbamidomethyl (C, fixed) peptide modification is additionaly allowed during search. Peptides with length ≥ 6 amino acids are conserved at validation of identifications. A more detailed description of the parameters used in the quantification step are provided in Supplementary Table 2.

On all datasets, we remove peptides or precursors having strictly less than 2 observations. Finally, we apply $\log_2$ transformation both on peptide or precursor intensities and transcript normalized counts (when available).

4.3 List of competitive methods

To benchmark Pirat, we use 15 state-of-the-art imputation methods, many of which have already been pinpointed by NAguideR evaluation software [18]: Bayesian Principal Component Analysis (BPCA) [57] and SVD [58] parametrized with a number of components equal to the number of compared conditions in each dataset; GMS [59] with 3 fold cross-validation for the tuning of its inner parameter (termed λ); K-nearest neighbors (KNN) [58] SeqKNN [60], trKNN [61] with the number of neighbors k set to 10; ImpSeq [42] and its outlier oriented version ImpSeqRob [62], Local Least Square (LLS) [63], MinProb [1], MissForest [64], MLE [55], msImpute in MAR and MNAR versions [22], and QRILC [1] with default parameters. MissForest and LLS were tested with an input format where peptides are in columns and samples in rows, contrarily to all other methods. Amongst those methods, the following ones are specifically devoted to low abundance censored values: MinProb, msImpute_mnar, QRILC, TrKNN.

While most methods require only 2 observed values (sometimes less) per peptide, GMS and trKNN require at least 3 of them, and both msImpute versions, 4 of them. This is an issue for two reasons: First, because, on controlled datasets, the biological variability is so small that not imputing those peptides before testing for differential abundance does not hamper the results, hereby artificially boosting the feature selection performances with respect to real-life cases. Second, because observing only two values is common for peptides nearby the detection limit, so that a concrete assessment on the MNAR imputation error is not possible with these algorithms. This is why, regardless of their performances on the biomarker selection task (see Section 2.2), GMS, trKNN, msImpute_MNAR, and msImpute_MAR could not be considered in the mask-and-impute experiments.

Likewise, three remaining state-of-the-art methods could not be included in the benchmark for various reasons: (*i*) Penalized Expectation Maximization for Missing Values (PEMM) [6], which had originally been validated on a simulated dataset with 100 proteins and could not scale up to an entire real-life peptide/precursor dataset as a result of its prohibitive computational cost; (*ii*) The PIMMS methods [65] which are based on deep-learning approaches, and which consequently require large cohorts (more than 50 samples) to be fully efficient; (*iii*) ProJect [66], a general purpose omics imputation method which code has not yet been commented, structured and packaged at the time of our evaluations, so that we could not make it work on the benchmark data. Finally, few elder methods with previously demonstrated poor performances (*e.g.*, zero imputation, mean imputation, accelerated failure time [67] etc.) where not included in the benchmark, for sake of clear enough plots.

Finally, in our experiments, we did not consider any meta imputation algorithm (post-processor, like GSimp [12, 16], ensemble imputation like IMP4P [14], chained imputation like MICE [68], etc.), as their performances directly relates to the imputation algorithm they rely on, and as most of them can be extended to incorporate new input algorithms like Pirat.

4.4 Differential abundance validation

On datasets with known ground-truth about differentially expressed proteins [49, 50, 51], we compare our methods with those described in Section 4.3 using the following procedure: *(i)* Impute missing peptide (or precursor) abundance values. *(ii)* Test the all-mean equality of each peptide (or precursor) using the one-way ANOVA omnibus test (from R package *stats* [69]) and retain the resulting p-values. If a peptide cannot be imputed by a given algorithm (because of too few observed values), the test is performed nonetheless if the number of observed values allows for it (see [69] for details), otherwise we set its p-value to one. *(iii)* Display the precision-recall curves for the precision range [90%, 100%] (*i.e.*, low FDP setting) for each method, and compute the global area under the curve.

4.5 Mask-and-impute experiments

To precisely evaluate imputation errors, it has become customary to add pseudo-missing values with different MCAR/MNAR proportions, and to compute Root Mean Square Error (RMSE) or Mean Absolute Error (MAE) between imputed and ground-truth values, which read as:

$$\mathrm{RMSE}(X, X^{\mathrm{imp}}) = \sqrt{\frac{1}{n} \sum_{(i,j) \in \mathcal{I}_{\mathrm{pseudo}}} (X_{i,j} - X^{\mathrm{imp}}_{i,j})^2}\,, \tag{15}$$

$$\mathrm{MAE}(X, X^{\mathrm{imp}}) = \frac{1}{n} \sum_{(i,j) \in \mathcal{I}_{\mathrm{pseudo}}} |X_{i,j} - X^{\mathrm{imp}}_{i,j}| \tag{16}$$

where $\mathcal{I}_{\mathrm{pseudo}}$ is the set of indices of pseudo-missing values and where X^{imp} is the imputed matrix. We generate pseudo MNAR values using a Probit left-censoring mechanism, *i.e.* the probability for a value x to be missing is 1 - $\Phi(x|\nu, \tau)$, where the ν and τ are mean and standard deviation from the Gaussian cumulative distribution function Φ [37]. The overall missing value rate α and the MNAR/MCAR proportion β are controlled by applying the following procedure (adapted from [1]):

1. Set $\tau = \sigma/2$ where σ is the overall standard deviation of the dataset.
2. For each values of ν' on a linear scale between $\mu - 3\sigma$ and μ (the overall mean of the dataset), compute the expected overall rates q of missing values when applying Probit left-censoring wih parameters ν' and τ, and retain these values. In practice we compute 100 points in total.
3. Interpolate points (ν', q) obtained.
4. Compute ν associated to $\beta\alpha$ (the desired overall MNAR rate) with interpolated curve. If $\alpha\beta$ is not in the range of interpolated curve, choose larger scale for ν' at step 2.
5. Add MNAR values using Probit left-censoring with parameters ν and τ.
6. Add MCAR values with probability $\frac{\alpha(1-\beta)}{1-\alpha\beta}$.

This method ensures that:

- The overall MNAR rate is $\beta\alpha$ (by construction of interpolated curve of (ν', q)).
- The overall MV is rate is α, as

$$\begin{aligned} P(x \text{ is missing}) &= P((x \text{ is MCAR }) \cup (x \text{ is MNAR})) \\ &= P(x \text{ is MCAR}) + P(x \text{ is MNAR}) - P(x \text{ is MCAR } \cap x \text{ is MNAR}) \\ &= \frac{\alpha(1-\beta)}{1-\alpha\beta} + \beta\alpha - \frac{\beta\alpha^2(1-\beta)}{1-\alpha\beta} = \alpha\,, \end{aligned} \tag{17}$$

  where x is an observed abundance value.

Finally, we compare all the methods in our mask-and-impute experiments on the very same artificially masked datasets.

References

[1] C. Lazar, L. Gatto, M. Ferro, *et al.* 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, 2016. ISSN 15353907. doi:10.1021/acs.jproteome.5b00981.

[2] M. Liu and A. Dongre. Proper imputation of missing values in proteomics datasets for differential expression analysis. *Briefings in Bioinformatics*, 22(3):1–17, 2021. ISSN 1467-5463. doi:10.1093/bib/bbaa112.

[3] Y. V. Karpievitch, A. R. Dabney, and R. D. Smith. Normalization and missing value imputation for label-free LC-MS analysis. *BMC Bioinformatics 2012 13:16*, 13(16):1–9, 2012. ISSN 1471-2105. doi:10.1186/1471-2105-13-S16-S5.

[4] V. Vidova and Z. Spacil. A review on mass spectrometry-based quantitative proteomics: Targeted and data independent acquisition. *Analytica Chimica Acta*, 964:7–23, 2017. ISSN 0003-2670. doi:10.1016/J.ACA.2017.01.059.

[5] R. J. Little and D. B. Rubin. *Statistical analysis with missing data*, volume 793. John Wiley & Sons, 2019. doi:10.1002/9781119482260.

[6] L. S. Chen, R. L. Prentice, and P. Wang. A penalized EM algorithm incorporating missing data mechanism for Gaussian parameter estimation. *Biometrics*, 70(2):312–322, 2014. ISSN 1541-0420. doi:10.1111/BIOM.12149.

[7] S. Y. Ryu, W. J. Qian, D. G. Camp, *et al.* Detecting differential protein expression in large-scale population proteomics. *Bioinformatics*, 30(19):2741–2746, 2014. ISSN 1367-4803. doi:10.1093/BIOINFORMATICS/BTU341.

[8] J. J. O'brien, H. P. Gunawardena, J. A. Paulo, *et al.* The effects of nonignorable missing data on label-free mass spectrometry proteomics experiments. *The annals of applied statistics*, 12(4):2075, 2018. ISSN 19417330. doi:10.1214/18-AOAS1144.

[9] L. J. E. Goeminne, A. Sticker, L. Martens, *et al.* MSqRob Takes the Missing Hurdle: Uniting Intensity- and Count-Based Proteomics. *Analytical Chemistry*, 92(9):6278–6287, 2020. ISSN 0003-2700. doi:10.1021/acs.analchem.9b04375.

[10] M. Chion and A. Leroy. A bayesian framework for multivariate differential analysis accounting for missing data, 2023. doi:10.48550/arXiv.2307.08975.

[11] B. J. M. Webb-Robertson, H. K. Wiberg, M. M. Matzke, *et al.* Review, evaluation, and discussion of the challenges of missing value imputation for mass spectrometry-based label-free global proteomics. *Journal of Proteome Research*, 14(5):1993–2001, 2015. ISSN 15353907. doi:10.1021/pr501138h.

[12] R. Wei, J. Wang, E. Jia, *et al.* GSimp: A Gibbs sampler based left-censored missing value imputation approach for metabolomics studies. *PLoS Computational Biology*, 14(1):e1005973, 2018. ISSN 15537358. doi:10.1371/journal.pcbi.1005973.

[13] W. Ma, S. Kim, S. Chowdhury, *et al.* DreamAI: algorithm for the imputation of proteomics data. *bioRxiv*, page 2020.07.21.214205, 2020. doi:10.1101/2020.07.21.214205.

[14] Q. G. Gianetto, S. Wieczorek, Y. Couté, and T. Burger. A peptide-level multiple imputation strategy accounting for the different natures of missing values in proteomics data. *bioRxiv*, page 2020.05.29.122770, 2020. doi:10.1101/2020.05.29.122770.

[15] M. L. Gardner and M. A. Freitas. Multiple Imputation Approaches Applied to the Missing Value Problem in Bottom-Up Proteomics. *International Journal of Molecular Sciences*, 22(17):9650, 2021. ISSN 1422-0067. doi:10.3390/ijms22179650.

[16] J. Wang, X. Gong, M. Hu, and L. Zhao. Improved GSimp: A Flexible Missing Value Imputation Method to Support Regulatory Bioequivalence Assessment. *Annals of Biomedical Engineering 2022*, pages 1–11, 2022. ISSN 1573-9686. doi:10.1007/S10439-022-03070-4.

[17] M. Chion, C. Carapito, and F. Bertrand. Accounting for multiple imputation-induced variability for differential analysis in mass spectrometry-based label-free quantitative proteomics. *PLOS Computational Biology*, 18(8):e1010420, 2022. ISSN 1553-7358. doi:10.1371/journal.pcbi.1010420.

[18] S. Wang, W. Li, L. Hu, *et al.* NAguideR: Performing and prioritizing missing value imputations for consistent bottom-up proteomic analyses. *Nucleic Acids Research*, 48(14):e83, 2020. ISSN 13624962. doi:10.1093/nar/gkaa498.

[19] W. Kong, H. W. H. Hui, H. Peng, and W. W. B. Goh. Dealing with missing values in proteomics data. *PROTEOMICS*, page 2200092, 2022. ISSN 1615-9853. doi:10.1002/PMIC.202200092.

[20] L. Jin, Y. Bi, C. Hu, *et al.* A comparative study of evaluating missing value imputation methods in label-free proteomics. *Scientific Reports*, 11(1):1760, 2021. ISSN 20452322. doi:10.1038/s41598-021-81279-4.

[21] M. Shen, Y. T. Chang, C. T. Wu, *et al.* Comparative assessment and novel strategy on methods for imputing proteomics data. *Scientific Reports 2022 12:1*, 12(1):1–11, 2022. ISSN 2045-2322. doi:10.1038/s41598-022-04938-0.

[22] S. Hediyeh-zadeh, A. I. Webb, and M. J. Davis. MsImpute: Estimation of missing peptide intensity data in label-free quantitative mass spectrometry. *Molecular & Cellular Proteomics*, page 100558, 2023. ISSN 1535-9476. doi:10.1016/J.MCPRO.2023.100558.

[23] M. Li and G. K. Smyth. Neither random nor censored: estimating intensity-dependent probabilities for missing values in label-free proteomics. *Bioinformatics*, 39(5), 2023. ISSN 1367-4811. doi:10.1093/bioinformatics/btad200.

[24] R. Argelaguet, B. Velten, D. Arnol, *et al.* Multi-Omics Factor Analysis—a framework for unsupervised integration of multi-omics data sets. *Molecular Systems Biology*, 14(6):e8124, 2018. ISSN 1744-4292. doi:10.15252/msb.20178124.

[25] E. Leppäaho, S. Kaski, and M. E. Khan. GFA: Exploratory Analysis of Multiple Data Sources with Group Factor Analysis Muhammad Ammad-ud-din. *Journal of Machine Learning Research*, 18:1–5, 2017.

[26] F. Rohart, B. Gautier, A. Singh, and K.-A. L. Cao. mixOmics: An R package for 'omics feature selection and multiple data integration. *PLOS Computational Biology*, 13(11):e1005752, 2017. ISSN 1553-7358. doi:10.1371/JOURNAL.PCBI.1005752.

[27] C. Meng, A. Basunia, B. Peters, *et al.* MOGSA: Integrative single sample gene-set analysis of multiple omics data. *Molecular and Cellular Proteomics*, 18(8):S153–S168, 2019. ISSN 15359484. doi:10.1074/mcp.TIR118.001251.

[28] Z. Zhou, C. Ye, J. Wang, and N. R. Zhang. Surface protein imputation from single cell transcriptomes by deep neural networks. *Nature Communications*, 11(1):1–10, 2020. ISSN 20411723. doi:10.1038/s41467-020-14391-0.

[29] M. P. Barzine, K. Freivalds, J. C. Wright, *et al.* Using Deep Learning to Extrapolate Protein Expression Measurements. *Proteomics*, 20(21-22), 2020. ISSN 16159861. doi:10.1002/pmic.202000009.

[30] J. E. Flores, D. M. Claborne, Z. D. Weller, *et al.* Missing data in multi-omics integration: Recent advances through artificial intelligence. *Frontiers in Artificial Intelligence*, 6, 2023. ISSN 26248212. doi:10.3389/frai.2023.1098308.

[31] S. Gupta, C. N. Schlaffner, S. Ahmed, *et al.* Imputing protein abundance by modeling molecular relationships using gnns. `https://www.iscb.org/ismbeccb2023-programme/posters`, 2023. Accessed: 2023-10-05.

[32] D. C. Liu and J. Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1-3):503–528, 1989. ISSN 00255610. doi:10.1007/BF01589116/METRICS.

[33] C. P. Robert and G. Casella. *Monte Carlo Integration*, pages 71–138. Springer New York, New York, NY, 1999. ISBN 978-1-4757-3071-5. doi:10.1007/978-1-4757-3071-5_3.

[34] T. Burger. Gentle Introduction to the Statistical Foundations of False Discovery Rate in Quantitative Proteomics. *Journal of Proteome Research*, 17(1):12–22, 2018. ISSN 15353907. doi:10.1021/ACS.JPROTEOME.7B00170.

[35] M. Capizzi, R. Carpentier, E. Denarier, *et al.* Developmental defects in Huntington's disease show that axonal growth and microtubule reorganization require NUMA1. *Neuron*, 110(1):36–50.e5, 2022. ISSN 0896-6273. doi:10.1016/J.NEURON.2021.10.033.

[36] N. Vilallongue, J. Schaeffer, A. M. Hesse, *et al.* Guidance landscapes unveiled by quantitative proteomics to control reinnervation in adult visual system. *Nature Communications*, 13(1):1–20, 2022. ISSN 20411723. doi:10.1038/s41467-022-33799-4.

[37] W. Miao, P. Ding, and Z. Geng. Identifiability of Normal and Normal Mixture Models with Nonignorable Missing Data. *Journal of the American Statistical Association*, 111(516):1673–1683, 2016. ISSN 1537274X. doi:10.1080/01621459.2015.1105808.

[38] M. Munteanu, R. Pais, V. Peta, *et al.* Long-term prognostic value of the FibroTest in patients with non-alcoholic fatty liver disease, compared to chronic hepatitis C, B, and alcoholic liver disease. *Alimentary Pharmacology & Therapeutics*, 48(10):1117–1127, 2018. ISSN 1365-2036. doi:10.1111/APT.14990.

[39] D. Ropers, Y. Couté, L. Faure, *et al.* Multiomics Study of Bacterial Growth Arrest in a Synthetic Biology Application. *ACS Synthetic Biology*, 10(11):2910–2926, 2021. ISSN 21615063. doi:10.1021/ACSSYNBIO.1C00115.

[40] A. N. Habowski, J. L. Flesher, J. M. Bates, *et al.* Transcriptomic and proteomic signatures of stemness and differentiation in the colon crypt. *Communications Biology 2020 3:1*, 3(1):1–17, 2020. ISSN 2399-3642. doi:10.1038/s42003-020-01181-z.

[41] S. Wieczorek, F. Combes, H. Borges, and T. Burger. Protein-level statistical analysis of quantitative label-free proteomics data with ProStaR. *Methods in Molecular Biology*, 1959:225–246, 2019. ISSN 10643745. doi:10.1007/978-1-4939-9164-8_15/COVER.

[42] S. Verboven, K. V. Branden, and P. Goos. Sequential imputation for missing values. *Computational Biology and Chemistry*, 31(5-6):320–327, 2007. ISSN 1476-9271. doi:10.1016/J.COMPBIOLCHEM.2007.07.001.

[43] N. Fortelny, C. M. Overall, P. Pavlidis, and G. V. C. Freue. Can we predict protein from mRNA levels? *Nature 2017 547:7664*, 547(7664):E19–E20, 2017. ISSN 1476-4687. doi:10.1038/nature22293.

[44] M. Dermit and J. G. Meyer. Peptide Correlation Analysis (PeCorA) Reveals Differential Proteoform Regulation. *Journal of Proteome Research*, 2021. ISSN 15353907. doi:10.1021/acs.jproteome.0c00602.

[45] A. Sportisse, C. Boyer, and J. Josse. Estimation and Imputation in Probabilistic Principal Component Analysis with Missing Not At Random Data. *Advances in Neural Information Processing Systems*, 33:7067–7077, 2020.

[46] B. Efron and C. Morris. Limiting the Risk of Bayes and Empirical Bayes Estimators—Part II: The Empirical Bayes Case. *Journal of the American Statistical Association*, 67(337):130–139, 1972. ISSN 0162-1459. doi:10.1080/01621459.1972.10481215.

[47] Z. Shi, B. Wen, Q. Gao, and B. Zhang. Feature Selection Methods for Protein Biomarker Discovery from Proteomics or Multi-omics Data. *Molecular & Cellular Proteomics*, page 100083, 2021. ISSN 15359476. doi:10.1016/j.mcpro.2021.100083.

[48] J. C. Pinheiro and D. M. Bates. Unconstrained parametrizations for variance-covariance matrices. *Statistics and Computing 1996*, 6(3):289–296, 1996. ISSN 1573-1375. doi:10.1007/BF00140873.

[49] J. Cox, M. Y. Hein, C. A. Luber, *et al.* Accurate Proteome-wide Label-free Quantification by Delayed Normalization and Maximal Peptide Ratio Extraction, Termed MaxLFQ. *Molecular & Cellular Proteomics : MCP*, 13(9):2513, 2014. ISSN 15359484. doi:10.1074/MCP.M113.031591.

[50] D. Bouyssié, A. M. Hesse, E. Mouton-Barbosa, *et al.* Proline: an efficient and user-friendly software suite for large-scale proteomics. *Bioinformatics*, 36(10):3148–3155, 2020. ISSN 1367-4803. doi:10.1093/BIOINFORMATICS/BTAA118.

[51] T. Huang, R. Bruderer, J. Muntel, *et al.* Combining Precursor and Fragment Information for Improved Detection of Differential Abundance in Data Independent Acquisition. *Molecular & Cellular Proteomics*, 19(2):421–430, 2020. ISSN 1535-9476. doi:10.1074/MCP.RA119.001705.

[52] A. M. Bolger, M. Lohse, and B. Usadel. Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics*, 30(15):2114–2120, 2014. ISSN 1367-4803. doi:10.1093/BIOINFORMATICS/BTU170.

[53] A. Dobin, C. A. Davis, F. Schlesinger, *et al.* STAR: ultrafast universal RNA-seq aligner. *Bioinformatics*, 29(1):15–21, 2013. ISSN 1367-4803. doi:10.1093/BIOINFORMATICS/BTS635.

[54] S. Anders, P. T. Pyl, and W. Huber. HTSeq—a Python framework to work with high-throughput sequencing data. *Bioinformatics*, 31(2):166–169, 2015. ISSN 1367-4803. doi:10.1093/BIOINFORMATICS/BTU638.

[55] M. I. Love, W. Huber, and S. Anders. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology*, 15(12):1–21, 2014. ISSN 1474760X. doi:10.1186/S13059-014-0550-8/FIGURES/9.

[56] S. Wieczorek, F. Combes, C. Lazar, *et al.* DAPAR & ProStaR: software to perform statistical analyses in quantitative discovery proteomics. *Bioinformatics*, 33(1):135–136, 2017. ISSN 1367-4803. doi:10.1093/BIOINFORMATICS/BTW580.

[57] S. Oba, M. A. Sato, I. Takemasa, *et al.* A Bayesian missing value estimation method for gene expression profile data. *Bioinformatics*, 19(16):2088–2096, 2003. ISSN 1367-4803. doi:10.1093/BIOINFORMATICS/BTG287.

[58] O. Troyanskaya, M. Cantor, G. Sherlock, *et al.* Missing value estimation methods for DNA microarrays. *Bioinformatics*, 17(6):520–525, 2001. ISSN 1367-4803. doi:10.1093/BIOINFORMATICS/17.6.520.

[59] Q. Li, K. Fisher, W. Meng, *et al.* GMSimpute: a generalized two-step Lasso approach to impute missing values in label-free mass spectrum analysis. *Bioinformatics*, 36(1):257–263, 2020. ISSN 1367-4803. doi:10.1093/BIOINFORMATICS/BTZ488.

[60] K. Y. Kim, B. J. Kim, and G. S. Yi. Reuse of imputed data in microarray analysis increases imputation efficiency. *BMC Bioinformatics*, 5(1):1–9, 2004. ISSN 14712105. doi:10.1186/1471-2105-5-160/FIGURES/3.

[61] J. S. Shah, S. N. Rai, A. P. DeFilippis, *et al.* Distribution based nearest neighbor imputation for truncated high dimensional data with applications to pre-clinical and clinical metabolomics studies. *BMC Bioinformatics*, 18(1):1–13, 2017. ISSN 14712105. doi:10.1186/S12859-017-1547-6/TABLES/4.

[62] K. V. Branden and S. Verboven. Robust data imputation. *Computational Biology and Chemistry*, 33(1):7–13, 2009. ISSN 1476-9271. doi:10.1016/J.COMPBIOLCHEM.2008.07.019.

[63] H. Kim, G. H. Golub, and H. Park. Missing value estimation for DNA microarray gene expression data: local least squares imputation. *Bioinformatics*, 21(2):187–198, 2005. ISSN 1367-4803. doi:10.1093/BIOINFORMATICS/BTH499.

[64] D. J. Stekhoven and P. Bühlmann. Missforest-Non-parametric missing value imputation for mixed-type data. *Bioinformatics*, 28(1):112–118, 2012. ISSN 13674803. doi:10.1093/BIOINFORMATICS/BTR597.

[65] H. Webel, L. Niu, A. B. Nielsen, *et al.* Imputation of label-free quantitative mass spectrometry-based proteomics data using self supervised deep learning. *bioRxiv*, page 2023.01.12.523792, 2023. doi:10.1101/2023.01.12.523792.

[66] W. Kong, B. J. H. Wong, H. W. H. Hui, *et al.* ProJect: a powerful mixed-model missing value imputation method. *Briefings in Bioinformatics*, 24(4), 2023. ISSN 14774054. doi:10.1093/BIB/BBAD233.

[67] S. L. Taylor, G. S. Leiserowitz, and K. Kim. Accounting for Undetected Compounds in Statistical Analyses of Mass Spectrometry 'Omic Studies. *Statistical applications in genetics and molecular biology*, 12(6):703, 2013. doi:10.1515/SAGMB-2013-0021.

[68] S. van Buuren and K. Groothuis-Oudshoorn. mice: Multivariate imputation by chained equations in r. *Journal of Statistical Software*, 45(3):1–67, 2011. doi:10.18637/jss.v045.i03.

[69] R Core Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2013. ISBN 3-900051-07-0.

A Supplementary Materials

A.1 ROC curves

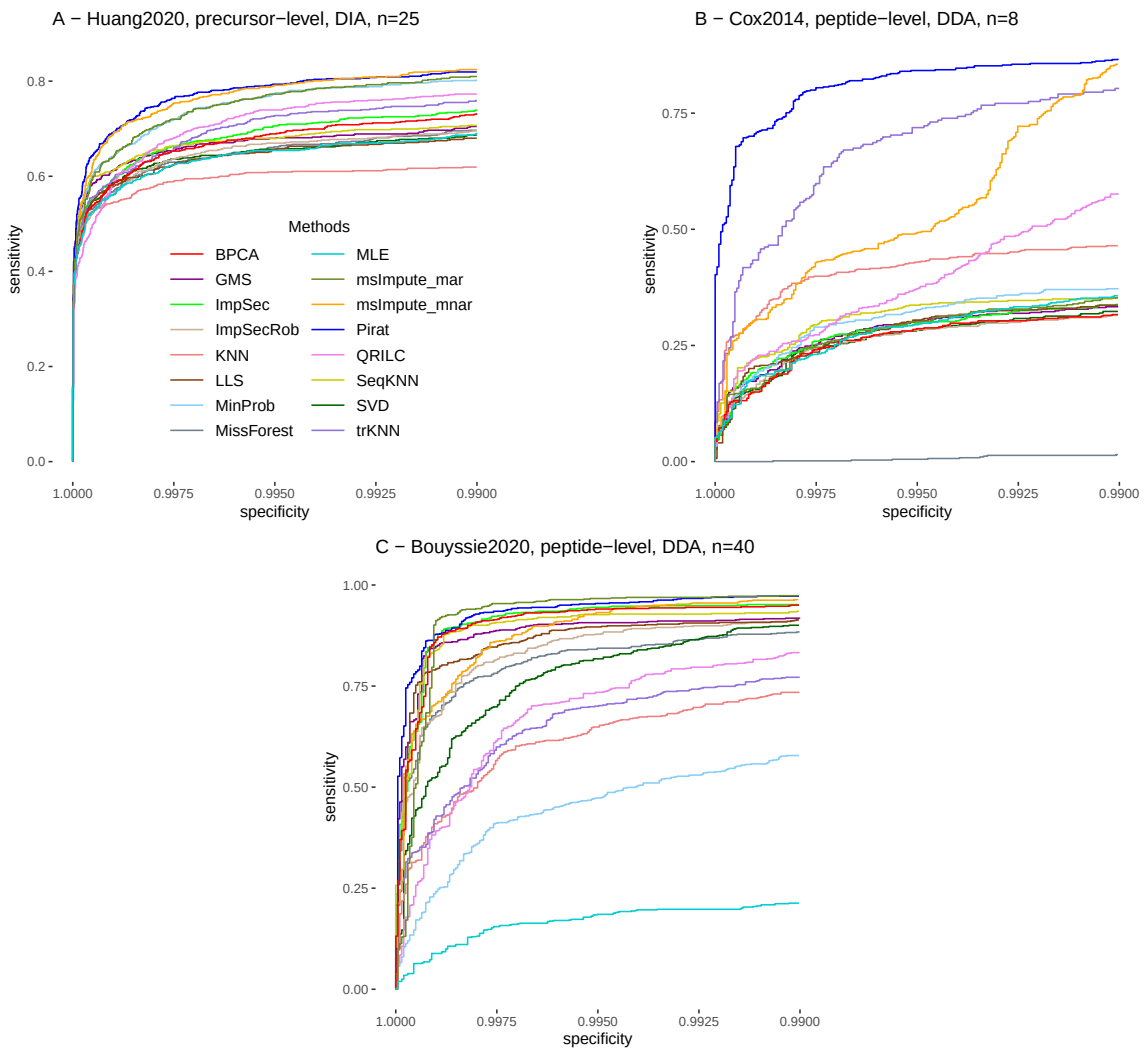


Figure 5: ROC curves from p-values associated to the differential analysis validation on the three datasets Huang2020, Cox2014 and Bouyssié2020. We show the curves between 100% and 99% specificity to better differentiate the methods when a stringent selection threshold is applied, which is often the case when FDR is controlled.

A.2 Ensembl Gene models, genome assembly, protein databases

This section refers to the filenames of Ensembl gene model, genome assembly and protein databases used to build proteomic and transcriptomic datasets of Ropers2021 and Habowski2020 (see Section 4.2).

A.2.1 Ropers2021

The Ensemble genome assembly used for read alignment in transcriptomic analysis is the following:
`GCA_000005845.2.fasta`

The Ensemble gene model used for mapping slices in transcriptomic analysis and for peptide identification in proteomic alignment is the following :
`Escherichia_coli_str_k_12_substr_mg1655_gca_000005845.ASM584v2.51.gtf`

A.2.2 Habowski2020

The Ensembl genome assembly used for read alignment in transcriptomic analysis is the following:
`Mus_musculus.GRCm39.dna.primary_assembly.fa`

The Ensemble gene model used for mapping slices in transcriptomic analysis is the following:
`Mus_musculus.GRCm39.104.gtf`

The Ensemble protein database used for peptide identification in proteomic analysis is the following:
`Mus_musculus.GRCm39.pep.all.fa`

A.3 Mean MAE and RMSE for QRILC and MinProb

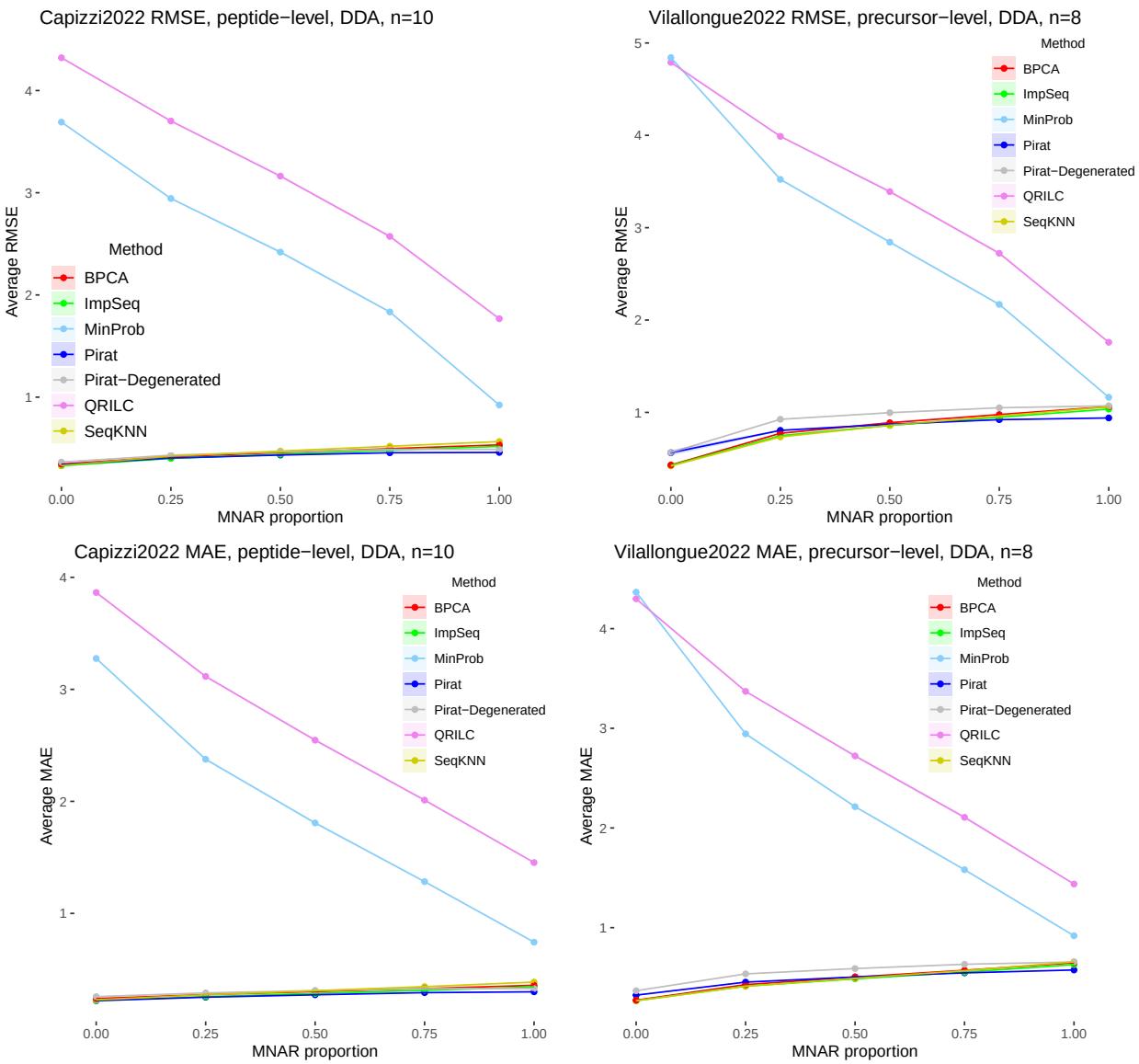


Figure 6: Average RMSE (top) and MAE (bottom) of MinProb, QRILC, BPCA, ImpSeq, Pirat, Pirat-Degenerated and SeqKNN in function of the proportion of MNAR values on Capizzi2022 (left) and Villalongue2022 (right). The imputation level (peptide or precursor), the type of acquisition (DIA or DDA), and the total number of replicates (n) are also indicated. The average RMSE and MAE was computed over 5 different seeds, and margins correspond to standard deviation.

A.4 Correlations of peptides in Capizzi2022 and Vilallongue2022

We show on Figure 7 the empirical densities of correlations between peptides taken randomly in the dataset, and between siblings, for Capizzi2022 and Vilallongue2022 (the empirical densities are smooth with a gaussian kernel to enhance interpretation). These can be easily plotted with the `plot_pep_correlations` function available in Pirat package. The more the within-PG correlation distribution is right-shifted with respect to that of random correlations, the more within-PGs correlations can be leveraged to impute MVs, and the better Pirat's performances. We observe on Figure 7 that the two distributions differ more in Capizzi2022 than in Vilallongue2022, resulting in improved RMSE/MAE performances, especially in MCAR setting (see Section 2.3).

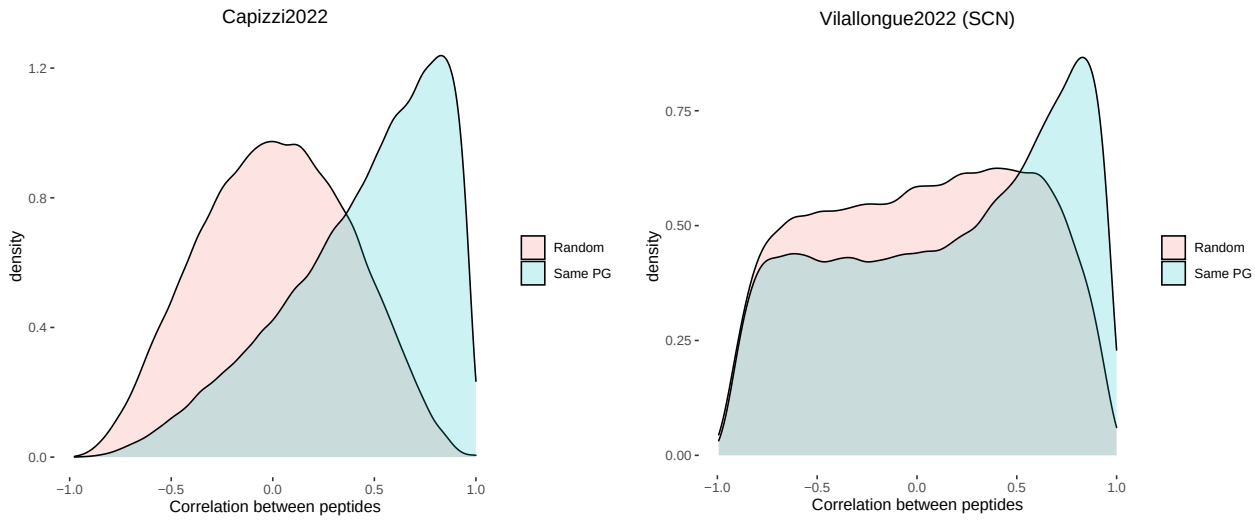


Figure 7: Empirical densities of correlations between peptides chosen randomly and between sibling peptides, for Capizzi2022 and Vilallongue2022 (SCN tissue).

We also give a counter-example of a dataset for which within-PG peptide correlations are very low (the data also come for Vilallongue2022 study, but with a different tissue, referred to as SC) compared to the random peptide correlations (see Figure 8). In this setting, Pirat hardly competes with other methods in terms of RMSE and MAE, and only outperforms them in 100% MNAR setting.

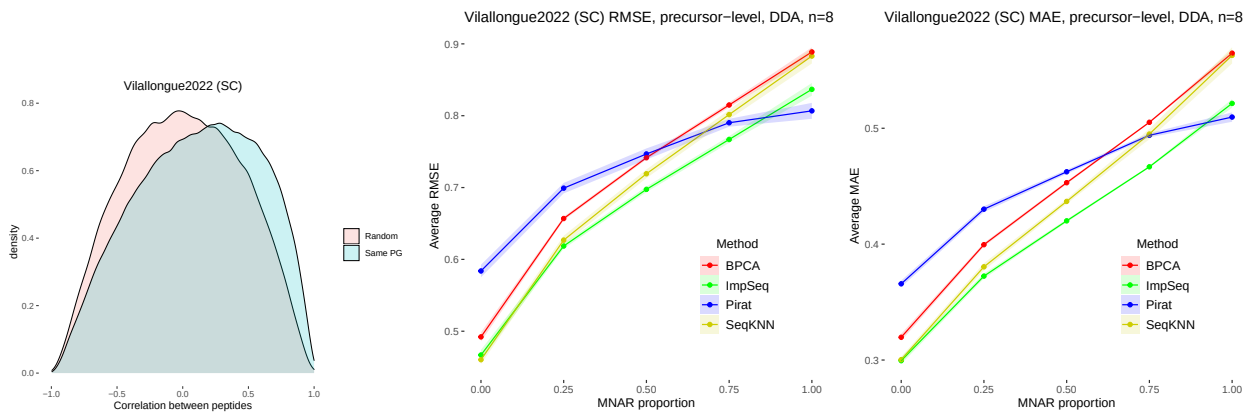


Figure 8: Empirical densities of correlations between peptides chosen randomly and between sibling peptides, for Vilallongue2022 (SC tissue), and associated RMSE and MAE vs MNAR proportion curves for different imputation methods on this dataset, according the experimental setting from Section 2.3.

A.5 Absolute errors for PGs of size one and others on Habowski2020 and Ropers2021

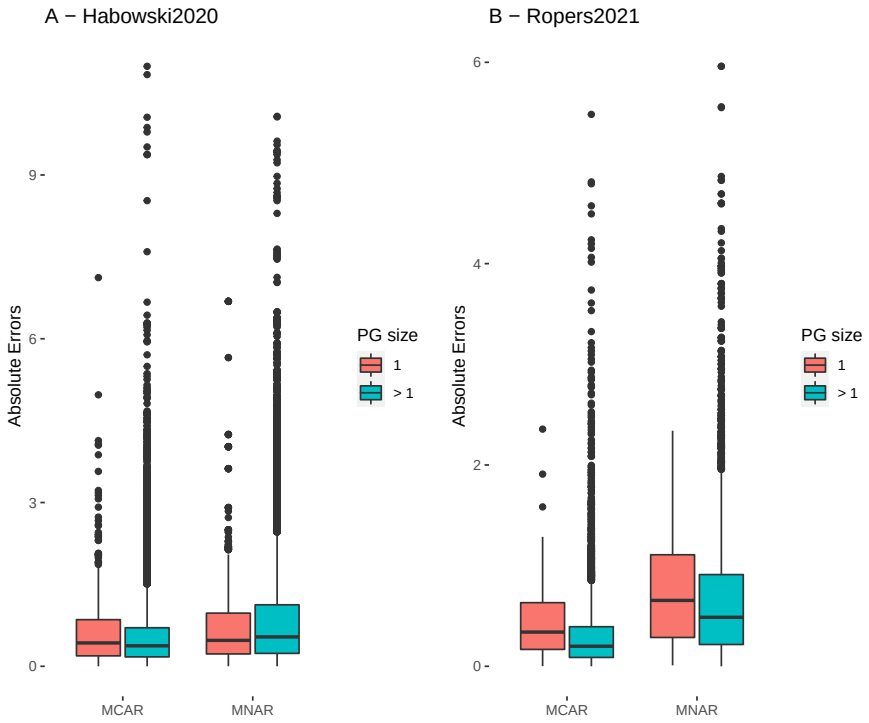


Figure 9: Boxplot of absolute errors of Pirat on PG of size equal to one and PGs of size superior to 1 on A - Habowki2020 and B - Ropers2021. Note that in the mask-and-impute experiment, much more pseudo-MVs comes from PGs of size superior to one than in singleton PGs (sometimes 50 times more), which explains the large difference between the number of outliers between for different PG size..

A.6 MV distribution in Habowski2020 and Ropers2021 experiments

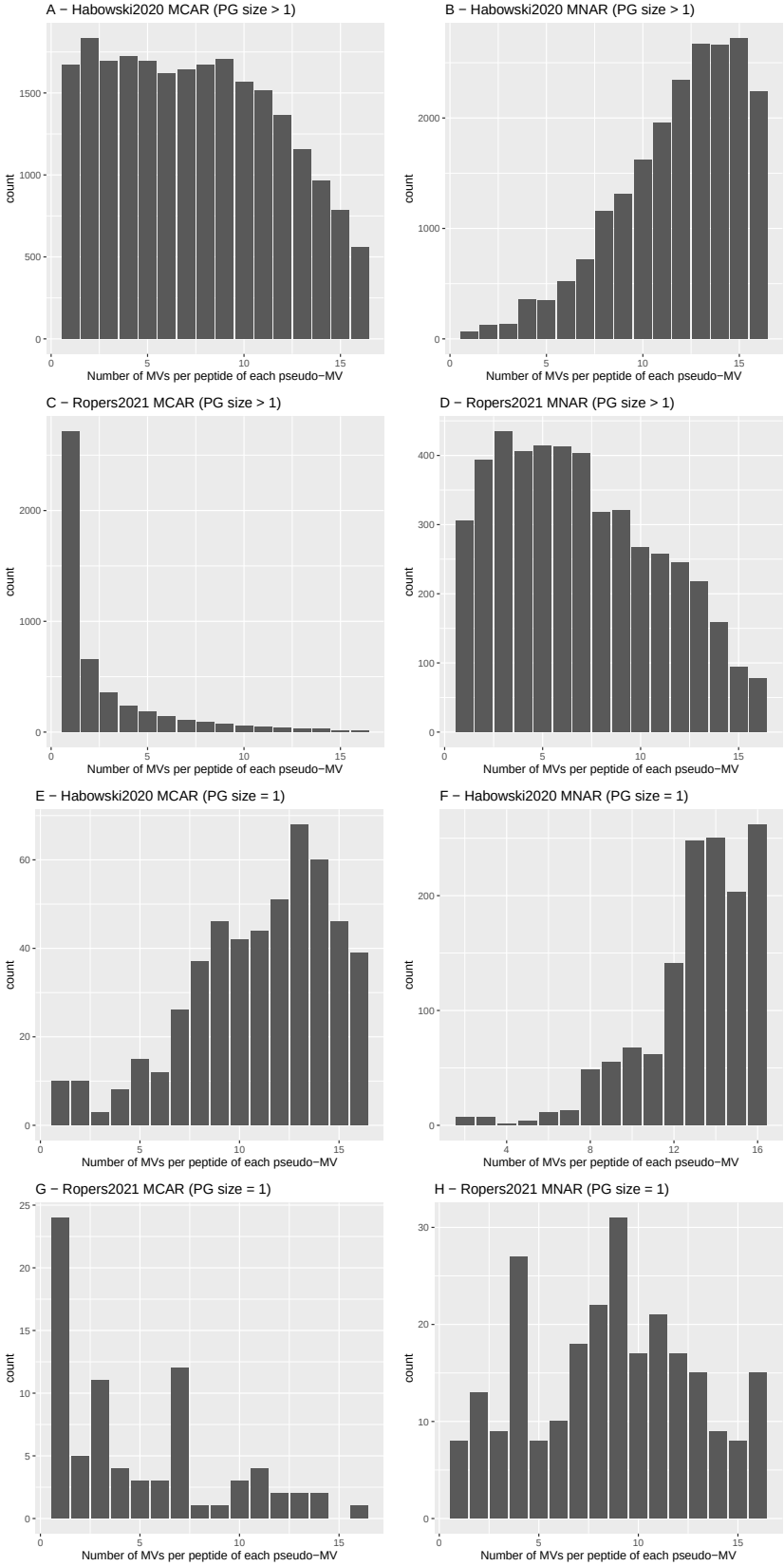


Figure 10: Histograms counting, for each pseudo-MV, the number of MVs (real or pseudo) in the same peptide, for Habowski2020 (A, B, E, F) and Ropers2021 (C, D, G, H), in MCAR (A, C, E, G) and MNAR (B, D, F, H) setting, and over 10 different seeds. We separate histograms for peptides contained in singleton PG (A, B, C, D) and non-singleton PGs (E, F, G, H). Note that the number of samples equals 18 in both datasets.

A.7 Fitting of missingness mechanism

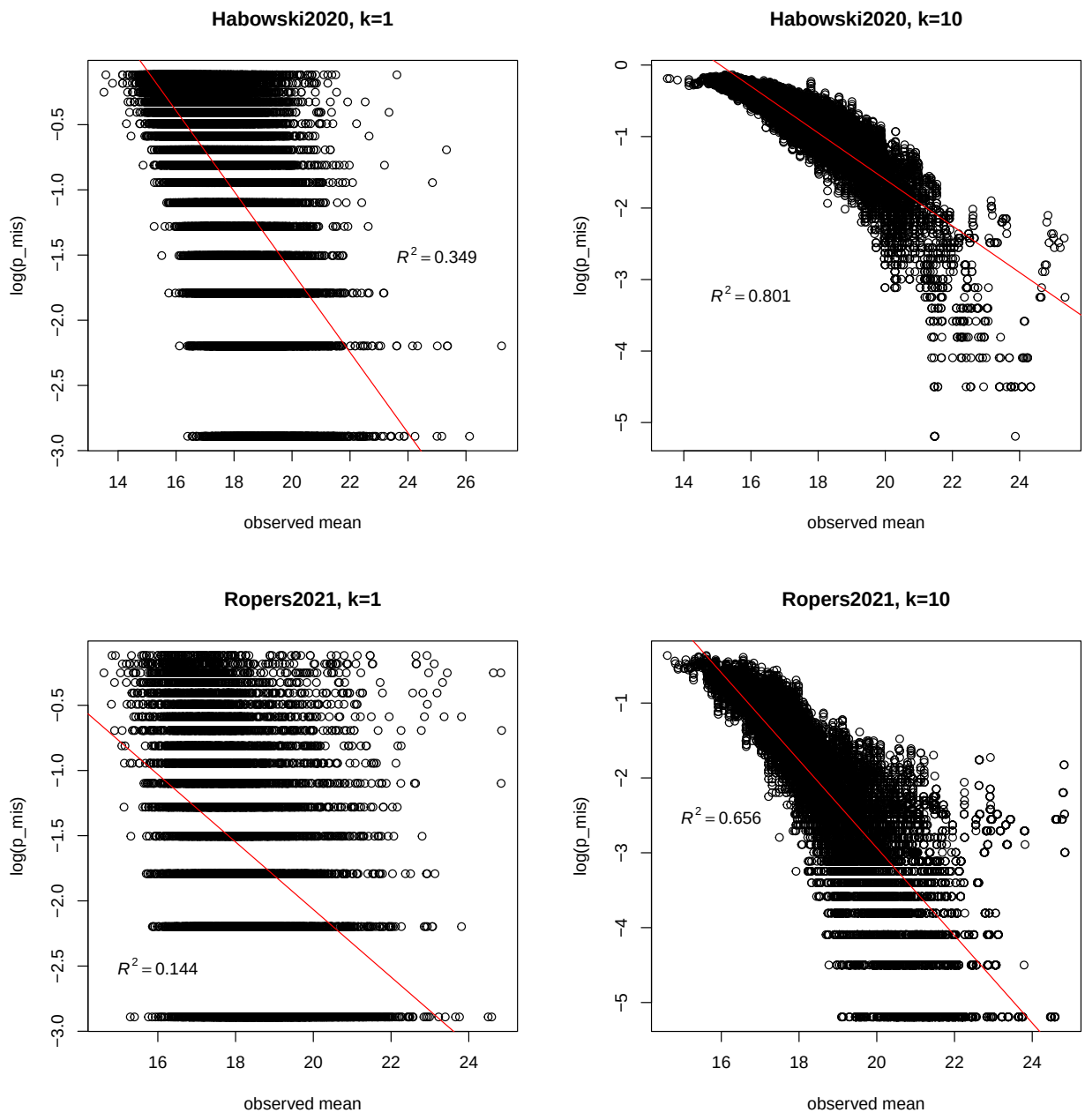
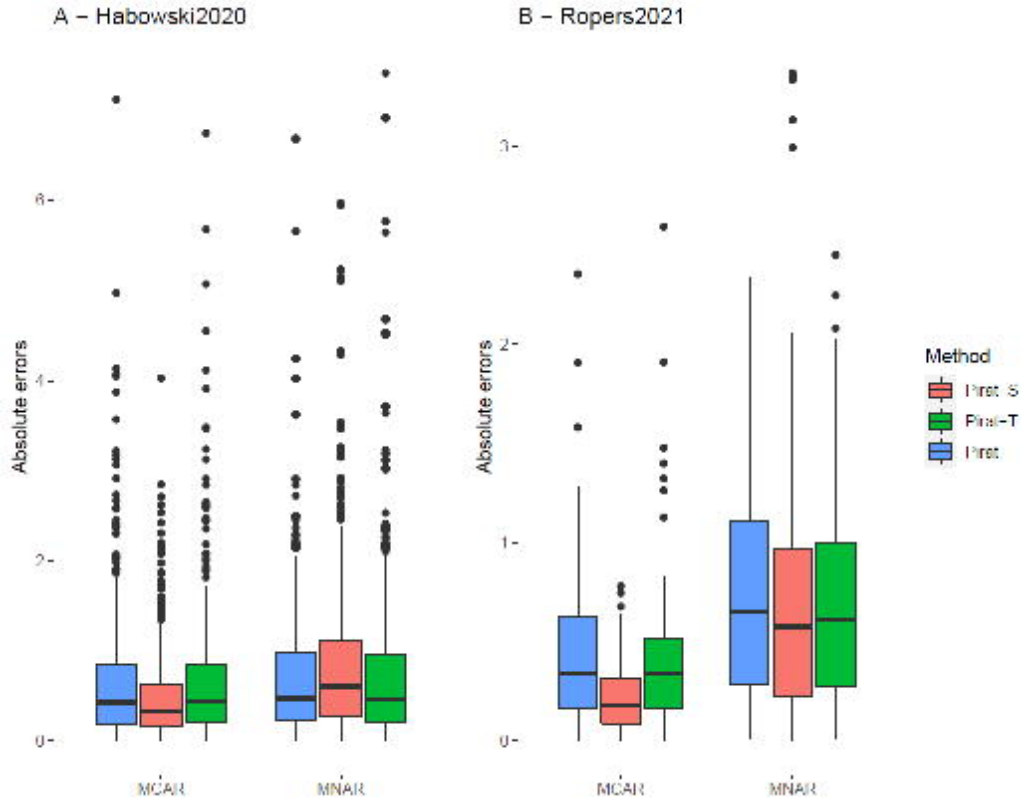
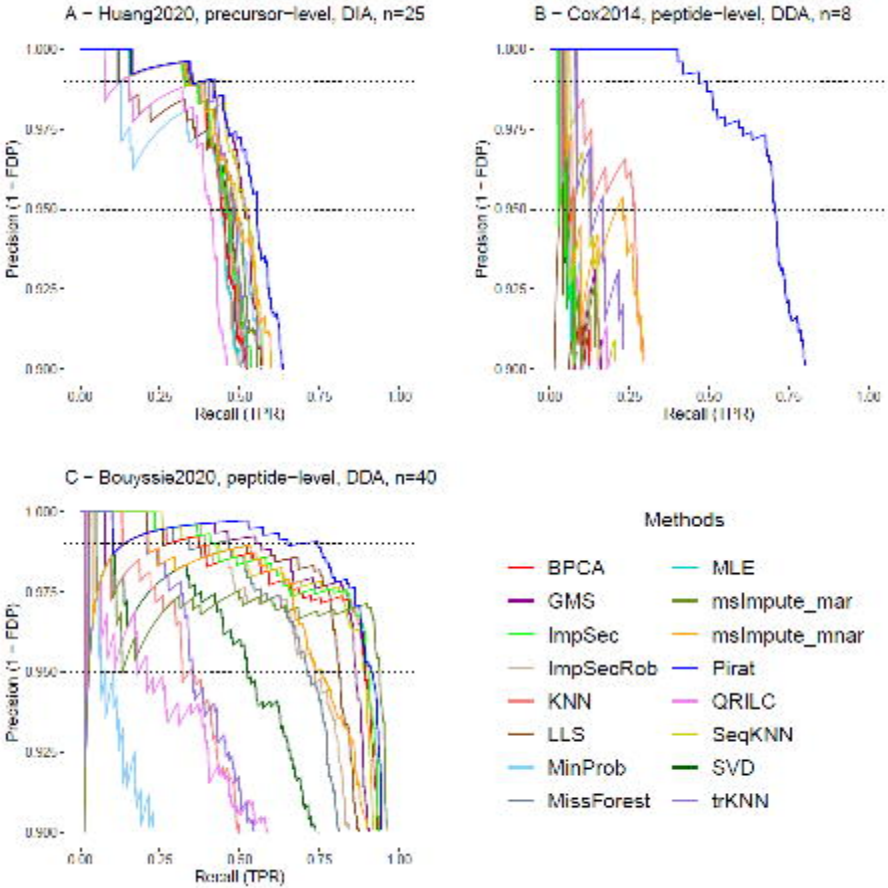
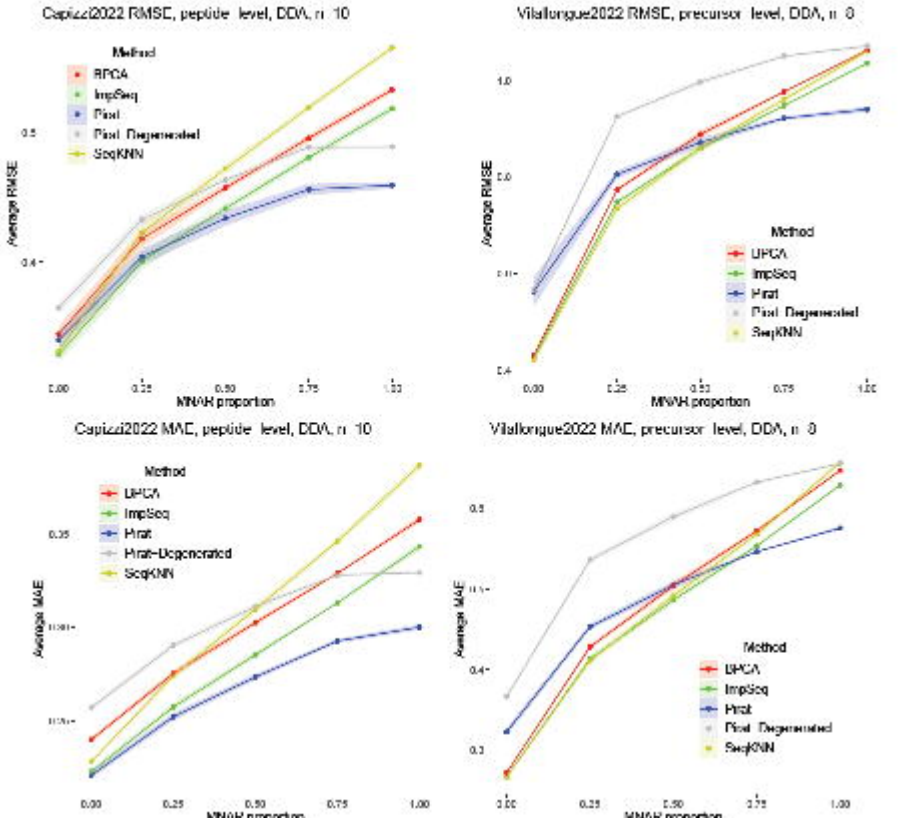
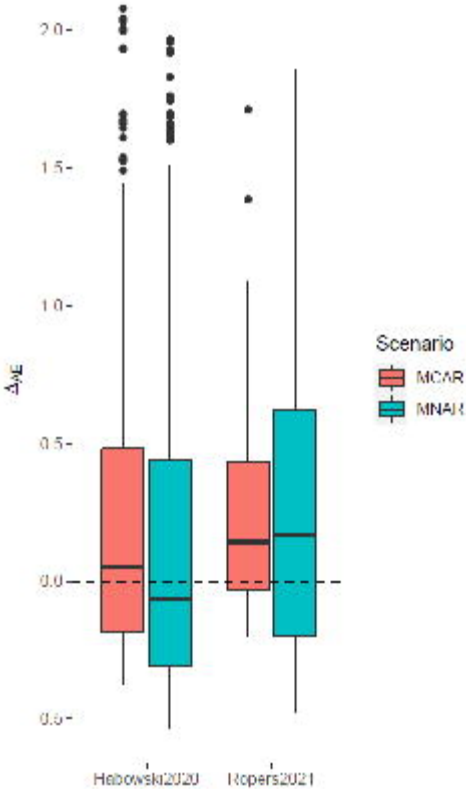


Figure 11: Regression of the log-probability of missing onto mean observed abundance following method described in Section 4.1.4, for Habowski2020 and Ropers2021, for $k = 1$ and $k = 10$. Residuals sum of squares (R^2) of the linear regression are also displayed.


A – Habowski2020
B – Ropers2021
Absolute errors
0
2
4
6
1
3
MCAR
MNAR
Method
Pinet-S
Pinet-T
Pinet


A – Huang2020, precursor–level, DIA, n=25
B – Cox2014, peptide–level, DDA, n=8
C – Bouyssie2020, peptide–level, DDA, n=40
Precision (1 – FDP)
Recall (TPR)
1.000
0.975
0.950
0.925
0.900
0.00
0.25
0.50
0.75
1.00
Methods
BPCA
GMS
ImpSec
ImpSecRob
KNN
LLS
MinProb
MissForest
MLE
msImpute_mar
msImpute_mnar
Pirat
QRILC
SeqKNN
SVD
trKNN


Capizzi2022 RMSE, peptide level, DDA, n 10
Method
RPCA
ImpSeq
Prat
Prat Degenerated
SeqKNN
Average RMSE
MNAR proportion
Vilallongue2022 RMSE, precursor level, DDA, n 0
Method
RPCA
ImpSeq
Prat
Prat Degenerated
SeqKNN
Average RMSE
MNAR proportion
Capizzi2022 MAE, peptide level, DDA, n 10
Method
RPCA
ImpSeq
Prat
Prat-Degenerated
SeqKNN
Average MAE
MNAR proportion
Vilallongue2022 MAE, precursor level, DDA, n 0
Method
RPCA
ImpSeq
Prat
Prat Degenerated
SeqKNN
Average MAE
MNAR proportion


2.0
1.5
1.0
0.5
0.0
-0.5
Δ_{ME}
Scenario
MCAR
MNAR
Habowski2020
Ropers2021