Supplemental Information

Quantifying homologous proteins and proteoforms

Dmitry Malioutov, Tianchi Chen, Jacob Jaffe, Edoardo Airoldi, Bogdan Budnik & Nikolai Slavov

Correspondence should be addressed to: nslavov@alum.mit.edu

This PDF file includes:

Supplemental Description of Modeling and Inference

Supplemental Discussion

Supplemental References

Supplemental Description of Modeling and Inference

1 Model Description

Consider that we are interested in estimating the abundance of K homologous proteins or proteoforms based on the levels of M peptides quantified across N conditions. To this end, the abundance of the i^{th} peptide quantified in the j^{th} condition (x_{ij}) is modeled as a linear superposition of the abundances of all proteins containing the peptide, scaled by an unknown peptide-specific nuisance constant (z_i) that characterizes how well the peptide was cleaved, chromatographically separated and ionized in the experiment (Figure 1). Applying this generative model to all peptides mapping to a set of homologous proteoforms or proteins results in an algebraic system of equations 1:

$$\underbrace{\begin{pmatrix} x_{11} & \dots & x_{1N} \\ \vdots & \ddots & \vdots \\ x_{M1} & \dots & x_{MN} \end{pmatrix}}_{\text{Peptide levels (data)}} = \underbrace{\begin{pmatrix} z_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & z_M \end{pmatrix}}_{\text{Nuisance}} \underbrace{\begin{pmatrix} s_{11} & \dots & s_{1K} \\ \vdots & \ddots & \vdots \\ s_{M1} & \dots & s_{MK} \end{pmatrix}}_{\text{Design matrix}} \underbrace{\begin{pmatrix} p_{11} & \dots & p_{1N} \\ \vdots & \ddots & \vdots \\ p_{K1} & \dots & p_{KN} \end{pmatrix}}_{\text{Protein levels}} \qquad (1)$$

In system 1 above, $\mathbf{X} = \mathbf{ZSP}$, $\mathbf{X} > 0$ are the MS measured intensities of M peptide in N physiological conditions, $\mathbf{Z} = \text{diag}(\mathbf{z})$ is an unknown diagonal matrix with peptide-specific nuisances $\mathbf{z} > 0$, $\mathbf{S}$ is a known $\{0, 1, 2, 3, \dots\}$ design/signature matrix of integers reflecting the number of times the amino acid sequence of the i^{th} peptide occurs in the corresponding K proteins, and $\mathbf{P} \geq 0$ is an unknown matrix of positive protein concentrations.[1] HI*quant* has to recover the

[1] *Nomenclature note:* all vectors are denoted in ***bold*** and small letters; all matrices are ***bold*** and CAPITAL letters. All matrices inferred from the measured peptide levels are denoted with $\hat{}$, all matrices inferred from simulated data are denoted with $\tilde{}$;

decomposition of the measured data $\mathbf{X}$ in terms of the known $\mathbf{S}$ and the unknown protein levels $\mathbf{P}$ and nuisances $\mathbf{Z}$.

Note that at best we can hope to recover $\mathbf{Z}$ and $\mathbf{P}$ up-to one scalar scaling, since $\alpha\mathbf{Z}$ and $\frac{1}{\alpha}\mathbf{P}$ give equivalent solutions for $\mathbf{X}$ and $\mathbf{S}$ for any $\alpha \neq 0$. In practice this is not a problem since peptide intensities quantified by MS are scaled arbitrarily and thus we generally need a scaling factor for converting the arbitrary intensity scale into number of molecules per cell. Also note that in the absence of shared peptides, $\mathbf{S}$ is an identity matrix and the equations are uncoupled. Then we can not even recover $\mathbf{Z}$ up-to a scalar multiple since then we have $\mathbf{X} = \mathbf{ZP}$ and any diagonal rescaling of $\mathbf{Z}$ results in a feasible solution. This case corresponds to having no shared peptides (only unique ones) and emphasizes the value of shared peptides for HI*quant*.

Below we derive a framework for establishing when a HI*quant* problem has a unique solution and a set of solvers that formulate and solve such problems within convex optimization framework. Before the rigorous treatment of the general case, we work out the analytic solution of the simplest problem as means of developing intuition.

1.1 Simple example with an exact analytic solution

In this example, we have measured $M = 3$ peptides across $N = 2$ conditions. These M peptides have sequences found in $K = 2$ proteins/proteoforms. Thus the example can be described by the system 2:

$$\begin{pmatrix} z_1 & 0 & 0 \\ 0 & z_2 & 0 \\ 0 & 0 & z_3 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{pmatrix} = \begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ x_{31} & x_{32} \end{pmatrix} \tag{2}$$

System 2 can be written in terms of $MN = 6$ linear equations that have $M + KN = 7$ unknown variables. Since the number of unknowns exceeds the number of equations by 1, system 2 can be solved at best up to 1 constant scaling (degree of freedom). Since, we only look for solutions up to scaling by a constant, we set $p_{11} = 1$. Fixing p_{11} effectively removes one unknown and thus there are just enough equations to determine the solution uniquely. This is the smallest example for which there is a unique solution. The exact solution is (substituting in order from top to bottom):

$$\begin{aligned} p_{11} &= 1 \\ p_{12} &= \frac{x_{12}}{x_{11}} p_{11} \\ p_{21} &= \frac{p_{11} - \frac{x_{31}}{x_{32}} p_{12}}{\frac{x_{31}x_{22}}{x_{32}x_{21}} - 1} \\ p_{22} &= \frac{x_{22}}{x_{21}} p_{21} \end{aligned} \qquad \begin{aligned} z_1 &= \frac{x_{11}}{p_{11}} \\ z_2 &= \frac{x_{21}}{p_{21}} \\ z_3 &= \frac{x_{31}}{p_{11} + p_{21}}. \end{aligned}$$

In the special case when $x_{22}x_{31} \equiv x_{21}x_{32}$, the above exact solution is unique up two scalings and in all other cases, it is unique to a single constant scaling. As a way of choosing this constant, one might set $p_{11} = 1$, or can introduce additional equations, such as normalization equations for total protein abundances (e.g., $\sum_i p_{i1} = 1$). When the system satisfies the normalization equations, the relative proteoform abundances can be interpreted as probabilities.

2 Existence and uniqueness of solutions

Beyond the best simple example considered in the previous section, we explore the existence and uniqueness of solutions of system (1) for K proteoforms (and homologous proteins) with M peptides quantified across N conditions. Depending on the problem (i.e., $\mathbf{S}$ and $\mathbf{X}$), (1) may have an ensemble of solutions within a high-dimensional feasibility space or it may have a unique solution up to a single scaling (one dimensional feasibility space) as was the case for system (2). In this section, we first focus on determining the dimensionality of the feasibility space (uniqueness of the solution constrained by the data) and then on algorithms guaranteed to find the optimal solution in the least squares (ℓ_2) sense.

At first sight, the general problem defined by system 1 appears to be a complex matrix-factorization problem, possibly requiring nonlinear optimization to solve without guarantees for having a solution or finding an optimal solution if one exists. This first impression, fortunately, is incorrect. The key to solving the problem defined in section 1 is rearranging the matrices to obtain a standard linear form that then can be analyzed by linear algebra tools. We start by defining auxiliary vector $\boldsymbol{\lambda} = [\frac{1}{z_1}, ..., \frac{1}{z_M}]$, and matrix $\boldsymbol{\Lambda} = \text{diag}(\boldsymbol{\lambda})$. We multiply system 1 by $\boldsymbol{\Lambda} = \mathbf{Z}^{-1}$ on both sides and rearrange to system 3 in which we want to find $\boldsymbol{\Lambda}$ and $\mathbf{P}$:

$$\boldsymbol{\Lambda}\mathbf{X} = \mathbf{S}\mathbf{P} \tag{3}$$

If the nuisances of some peptides are very low, this inversion of $\mathbf{Z}$ may introduce numerical instabilities. If low nuisances reflect poor digestion and ionization of peptides, the problem might be addressed by using a different protiase or even compiling the peptides with the highest intensities from several digestions, each of which using a different protease. If the peptides with low nuisances and thus intensities are not essential for constraining the model or correspond to very lowly abundant proteoforms, they can be removed from the model.

Since both unknowns ($\boldsymbol{\Lambda}$ and $\mathbf{P}$) enter equation 3 linearly, we can can rearrange the equation to a standard linear system. To obtain a standard linear form, we transpose both sides and move everything to one side to get:

$$\mathbf{P}^T\mathbf{S}^T - \mathbf{X}^T\boldsymbol{\Lambda} = 0 \tag{4}$$

Now let us stack columns of $\mathbf{P}^T$, i.e. rows of $\mathbf{P}$ into a vector $\mathbf{p} = \text{vec}(\mathbf{P}^T)$. Then $\text{vec}(\mathbf{P}^T\mathbf{S}^T) = (\mathbf{S} \otimes \mathbf{I})\text{vec}(\mathbf{p})$, where $\otimes$ stands for the kronecker product. Similarly define a matrix $\text{blkdiag}(\mathbf{X}^T)$, which is a block-diagonal matrix with columns of $\mathbf{X}^T$, i.e. rows of $\mathbf{X}$ in diagonal blocks. This way we have $\mathbf{X}^T\boldsymbol{\Lambda} = \text{blkdiag}(\mathbf{X}^T)\boldsymbol{\lambda}$. Combining these rearrangements together results in:

$$\underbrace{\begin{bmatrix} (\mathbf{S} \otimes \mathbf{I}) & \Big| & -\,\text{blkdiag}(\mathbf{X}^T) \end{bmatrix}}_{\mathbf{A}} \begin{bmatrix} \text{vec}(\mathbf{p}) \\ \text{vec}(\boldsymbol{\lambda}) \end{bmatrix} = 0 \tag{5}$$

Now we can use the power of linear algebraic methods to analyze the solution space of our problem and identify the optimal unique solution of our problem if it exists. In particular, any vector in the nullspace of $\mathbf{A} = \big[(\mathbf{S} \otimes \mathbf{I})| \; - \text{blkdiag}(\mathbf{X}^T)\big]$ is a solution to system 3 and thus to the original problem defined by system 1. To give a more intuitive description of the key matrix $\mathbf{A} \in \mathbb{R}^{MN \times KN+M}$, below we display $\mathbf{A}$ explicitly for the simplest example, $N = 2$, $M = 2$ and $K = 2$:

$$\begin{pmatrix} z_1 & 0 \\ 0 & z_2 \end{pmatrix} \begin{pmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{pmatrix} \begin{pmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{pmatrix} = \begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{pmatrix}$$

The corresponding matrix $\mathbf{A}$ whose null space we need to find is:

$$\begin{aligned} A &= \left[\mathbf{S} \otimes \mathbf{I}_N \mid \; - blkdiag(\mathbf{X}^T_{1:N,1}, .., \mathbf{X}^T_{1:N,M})\right] \\ &= \begin{pmatrix} s_{11} & 0 & s_{12} & 0 & -X_{11} & 0 \\ 0 & s_{11} & 0 & s_{12} & -X_{12} & 0 \\ s_{21} & 0 & s_{22} & 0 & 0 & -X_{21} \\ 0 & s_{21} & 0 & s_{22} & 0 & -X_{22} \end{pmatrix} \end{aligned}$$

A unique solution corresponds to a one dimensional (1D) null space of $\mathbf{A}$. The null space can be 1D only when the number of columns ($KN + M$) is equal or smaller than the number of rows (MN), i.e., $KN + M \leq MN$. If the number of peptides (M) is larger than the number of proteoforms (K), as the number of quantified conditions (N) increases, the number of rows grows faster than the number of columns of $\mathbf{A}$; thus, for large enough N, $\mathbf{A}$ will be a tall matrix. This condition is easily calculated for any one problem and required for $\mathbf{A}$ to have a 1D space:

$$KN + M \geq MN + 1 \implies \quad M - 1 \geq N(M - K)$$

Proof 1: Solutions are under-defined at $N = 1$

In the special case of a single quantified condition, i.e., $N = 1$, the number of unknowns ($K + M$) is always larger than the number of equations (M). Thus, the the protein levels in system (1) cannot be estimated independently from the nuisances. This mathematical result corresponds well with the physical intuition: HI*quant* estimates protein levels only from isotopologue ratios, and a single condition without heavy standards does not provide any isotopologue ratios.

In general, the dimension of the null space indicates the number of degrees of freedom in the solution since any linear combination of the vectors in the null space is a solution. Before attempting to solve a practical problem, therefore, we evaluate the numbers of degrees of freedom constrained by the design matrix $\mathbf{S}$ and the data $\mathbf{X}$. If the matrix $\mathbf{A}$ has a single vector in its nullspace, then the problem has a unique solution up to a single scaling. This case is of greatest practical interest since it means that the matrix of protein levels ($\mathbf{P}$) is identifiable up to a single scaling. Many problems constrained by MS data generate $\mathbf{A}$ matrices with 1 dimensional null space and are thus identifiable.

3 Inference of proteoform stoichiometries

3.1 Identifying protein clusters

First, HI*quant* aims to find how many homologous protein clusters are defined by the measured protein levels in the input file. To this end, HI*quant* builds a network in which each node represents a protein, and each link between nodes represents an existing shared peptide between two proteins. Then, each connected component of this network corresponds to a homologous protein cluster. HI*quant* identifies all connected components and compiles their associated peptide level matrices $\mathbf{X}$ and design matrices $\mathbf{S}$.

The peptide levels ($\mathbf{X}$) and their mappings to proteins ($\mathbf{S}$) constrain a solution, the protein levels ($\mathbf{P}$). We want to infer this solution by convex optimization since convexity guarantees optimal solutions without local minima. Below is a list of convex solvers that we have developed to infer $\mathbf{P}$.

3.2 SVD solver

In the noiseless case, the solution is simply the null space of $\mathbf{A}$ in equation (5). This null space can be found by singular value decomposition (SVD) of A as the space defined by the singular vectors with zero singular values. However, noise in the data may corrupt the null space to the point that the null space of $\mathbf{A}$ is lost (all singular values are larger than zero). Thus, while SVD is optimal in the noiseless case, it may perform poorly in the presence of noise. To mitigate such complications, we devised algorithms that are more robust to noise. The subsections below summarize our approaches to solving equations (1) and (5) that can handle noisy data and provide estimates for the accuracy of the inferred protein levels.

3.3 Quadratic programming based solver

The robustness to noise can be improved by incorporating the knowledge that protein concentrations and nuisances are non negative numbers. HI*quant* incorporates this fact as a constraint by solving the convex optimization problem defined by equation (6) using quadratic programming (QP) via interior point methods. The solution $\hat{\mathbf{u}}$ corresponds to a vector in the positive quadrant that approximates the null space of $\mathbf{A}$ in the absence of noise.

$$\hat{\mathbf{u}} = \arg\min_{\mathbf{u}} \mathbf{u}^T(\mathbf{A}^T\mathbf{A})\mathbf{u} \qquad \text{subject to } \mathbf{u} > 0 \text{ and } \langle \mathbf{u}, \mathbf{1} \rangle = 1 \tag{6}$$

The first $K \times N$ elements of $\hat{\mathbf{u}}$ correspond to the elements of $\mathbf{P}$ (the protein levels). The last M elements of $\hat{\mathbf{u}}$ correspond to the nuisances.

3.4 Coordinate descent based solvers

An alternative method to solving system 1 is by defining the optimization problem,

$$(\hat{\mathbf{P}}, \hat{\mathbf{z}}) = \min_{\mathbf{z},\mathbf{P}} \|\mathbf{X} - \text{diag}(\mathbf{z})\mathbf{S}\mathbf{P}\|_F^2, \tag{7}$$

and solving it by coordinate descent, first holding $\mathbf{z}$ constant and optimizing over $\mathbf{P}$, and then holding $\mathbf{P}$ constant and optimizing over $\mathbf{z}$. Note that both sub-problems have simple closed form-solutions and furthermore, optimization over $\mathbf{z}$ is separable – so we can find a solution for each z_i independently, and combine them.

This approach is known as coordinate descent and may have disadvantages, including that it may require many iterations to converge. It is, however, robust to measurement noise in the data ($\mathbf{X}$) and practically performs very well both on simulated and real experimental data. While the formulation (7) is not convex (and thus not theoretically guaranteed to find the optimal solution), it can be made convex by the simple rearrangement in equation (3). This results in the convex coordinate descent problem:

$$(\hat{\mathbf{P}}, \hat{\boldsymbol{\lambda}}) = \min_{\boldsymbol{\lambda},\mathbf{P}} \|\text{diag}(\boldsymbol{\lambda})\mathbf{X} - \mathbf{SP}\|_F^2, \tag{8}$$

As with the quadratic solver, we require that nuisances and protein levels are non negative $\mathbf{z} > 0,\ \mathbf{P} > 0$. This requirement can be implemented by a projected coordinate descent, where after each iteration we also project on the positive orthant, and update the current estimates, e.g., $\mathbf{P}$ is updated by taking $\mathbf{P} = max(\mathbf{P}, \epsilon_t)$, with ϵ_t being a small positive constant.

3.5 Structured total least squares solver

The above methods do not take into account the noise structure of the matrix $\mathbf{A}$. Some of its elements are not contaminated by noise, i.e., $\mathbf{S} \otimes \mathbf{I}$, while the others are contaminated by noise that can be estimated, i.e., blkdiag$(\mathbf{X}^T)$. To incorporate this type of information into the solution, we developed a convex algorithm to solving total least square (TLS) problems (Malioutov and Slavov, 2014). This convex TLS solver is based on relaxing/approximating the rank of $\mathbf{A}$ with its nuclear norm (Malioutov and Slavov, 2014).

4 Confidence intervals and reliability

In the absence of noise, HI*quant* is guaranteed to infer the correct proteoform stoichiometries. However, MS data are noisy to varying degrees, and in general the reliability of the inference is fundamentally dependent on the reliability of the MS data. Thus, the utility of HI*quant* inferences depends critically on the ability to derive reliability estimates for the inferred protein levels, $\hat{\mathbf{P}}$. To derive such reliability estimates, we built a random forest classifier based on features that are informative for the accuracy and reliability of the inferred $\hat{\mathbf{P}}$. These features include:

1. The goodness of fit for the model as quantified by the fraction of variance in the data (X) that is explained by the model, i.e., the R^2 for the solution of the quadratic programming (QP) solver.

2. The fraction of negative elements in the smallest singular vector of $\mathbf{A}$, $\mathbf{v}_1$. In the absence of noise, all elements of $\mathbf{v}_1$ should be non negative as they reflect non negative physical quantities; as noise contamination increases, the elements of $\mathbf{v}_1$ can begin to turn negative. Thus, the fractions of negative elements of $\mathbf{v}_1$ reflects noise contamination of the null space. Since singular vectors are determined up to a scalar multiple, which can be negative, HI*quant* flips the signs of $\mathbf{v}_1$ if necessary, i.e., $\mathbf{v}_1 = \mathbf{median}(\mathbf{sign}(\mathbf{v}_1))\mathbf{v}_1$.

3. The norm of the relative peptide levels, $\mathbf{X}$, which quantifies the magnitude of peptide changes across conditions and should reflects signal to noise ratios.

4. The mean, minimum and maximum coefficient of variation (CV) of the inferred protein levels, $\hat{\boldsymbol{P}}$.

5. The mean correlation between the columns of $\mathbf{X}$ as a measure of linear independence / independence in the data.

6. The size of the last eigen spacing of $\mathbf{A}$, denoted here by sp. Consider the l rank ordered (largest to smallest) singular values of $\mathbf{A}$. We compute the last eigen spacing as $sp = \frac{\sigma_{l-1}-\sigma_l}{\sigma_{l-1}+\sigma_l}$. In the absence of noise, σ_l=0, and thus sp=1. In the other extreme, when the null space of $\mathbf{A}$ is heavily contaminated by noise and $\sigma_{l-1} \approx \sigma_l$, $sp \approx 0$. Thus the magnitude of the last eigen spacing sp is informative of the degree to which noise has distorted the null space of $\mathbf{A}$.

7. The cosine of the angle between the protein levels inferred by the quadratic programming solver (QP) and the singular value decomposition solver (SVD).

8. The cosine of the angle between the protein levels inferred by the quadratic programming solver and coordinate descent solver (CD).

To assess the reliability of the protein levels (denoted by $\hat{\boldsymbol{P}}$) inferred from measured pepetide levels, HI*quant* builds a random forest classifier, which estimated the relative error in the inferred protein levels. To build the classifier, HI*quant* simulates data that resemble the measured data as much as possible. To this end, HI*quant* uses $\hat{\boldsymbol{P}}$ to simulate data so that the simulated data reflect the best estimates of the variability found in the real data. Furthermore, HI*quant* adds noise to the simulated peptide levels that covers the noise range expected for relative quantification by MS, ranging from 5 % to 30 %. In particular, each $\hat{\mathbf{P}}$ is used to simulate multiple peptide level matrices and each peptide matrix is contaminated by different noise level. HI*quant* constructs at least 5 additional simulated peptide levels with different noise. Each simulated peptide level $\widetilde{\boldsymbol{X}}$ is calculated in the following way:

$$\widetilde{\boldsymbol{X}} = \hat{\boldsymbol{Z}}\mathbf{S}\hat{\boldsymbol{P}} + n\hat{\boldsymbol{Z}}\mathbf{S}\hat{\boldsymbol{P}}\mathcal{N}(0,1), \quad n \in \{0.05, 0.1, 0.15, 0.2, 0.25, 0.3\} \tag{9}$$

From each simulated noisy peptide level matrix $\widetilde{\boldsymbol{X}}$, HI*quant* infers protein levels $\widetilde{\boldsymbol{P}}$ (by CD, QP, SVD solvers) and the associated feature vector $\widetilde{\boldsymbol{f}}$ which contains all features described above. $\widetilde{\boldsymbol{P}}$ is scaled the same way $\hat{\boldsymbol{P}}$ (to a median of 1) so that $\widetilde{\boldsymbol{P}}$ has the same median as the $\hat{\boldsymbol{P}}$, because as explained above the protein levels can be inferred up to a scaler scaling. If the number of protein clusters in the data is small, HI*quant* adds more than 5 noise additions to

Then, HI*quant* computes the relative error between the protein levels ($\widetilde{\boldsymbol{P}}$) inferred from the simulated noisy data and the protein levels used to simulate the data, $\hat{\boldsymbol{P}}$. The relative error is defined as the fractional (percent) differences between the inferred protein / proteoform levels and the correct ones. To calculate this error metric for each protein cluster at each noise level, HI*quant* performs a element-wised operation to obtain the fractional differences. Thus, HI*quant* calculates the absolute value of the relative error for each protein cluster to be:

$$\boldsymbol{E} := \frac{abs(\widetilde{\boldsymbol{P}} - \hat{\boldsymbol{P}})}{\hat{\boldsymbol{P}}} \tag{10}$$

The relative error for each simulated protein cluster is estimated as the median relative error for all protein levels across all conditions in the protein cluster:

$$e = median(\boldsymbol{E}) \tag{11}$$

These scalar errors e (response variable) and the associated predictors ($\hat{\boldsymbol{f}}$) at each noise level and for all protein homologous cluster are grouped into the final relative error metric (response variable) $\boldsymbol{e}$ and predictors matrix $\widetilde{\boldsymbol{F}}$. Finally, HI*quant* trains the random forest classifier using $\log(\boldsymbol{e})$ and $\widetilde{\boldsymbol{F}}$ with 300 regression trees.

To estimate the reliability of each $\hat{\boldsymbol{P}}$, we apply the trained random forest classifier to its features $\hat{\boldsymbol{f}}$ and estimate the associated relative error. These estimated relative error for each cluster is reported in its header in the output datafile. The estimated relative errors across all clusters are displayed as a distribution on the output web page.

5 Evaluating the random forest classifier

We evaluated the reliability and performance of the random forest classifier by applying it to simulated data with known relative errors. HI*quant* uses $\hat{\boldsymbol{P}}$ inferred from measured peptide level and generates the simulated data with noise set N as described above. To validate the random forest classifier, HI*quant* trains the classifier using 70% of the simulated data and tests its performance on the remaining 30% of the data. The relative errors correlate strongly (spearman $\rho > 0.7$) to the corresponding errors predicted by the random forest classifier, suggesting that the features and the classification model support reliable estimates of inference reliability.

Supplemental Discussion

Inference and quantification of complex proteoforms can be very challenging, and in some cases the data cannot support full inference, either because relative quantification is not accurate enough or because not all proteoforms were included in the inference or because not enough peptides were quantified; the latter limitation can be alleviated by compiling peptides quantified from several digestions, each of which using a different protease. In such problematic cases, HI*quant* inference will either result in multiple solutions consistent with the data, i.e., feasibility space of solutions, or in poor fits to the data that will be assigned low reliability by the validation module described above.

Currently, HI*quant* builds a model comprised of all proteoforms having at least one quantified unique peptide. In some cases, e.g., a ribosomal protein with a few paralogs, the proteoforms with quantified peptides include all possible proteoforms. If major proteoforms do not have quantified unique peptides in the data, HI*quant* cannot fit the data, which is manifested either with an underdetermined system (i.e., $\mathbf{A}$ has high dimensional null space) or with poor fit of the model, i.e., low R^2 and high relative error determined by the reliability analysis. Such cases involving large number of combinatorially occurring PTMs can be simplified by computing the fractional site occupancy of each PTM separately rather than inferring the full length proteoform. Alternatively the abundance of all missing proteoforms can be lumped into one proteoform whose abundances equal the sum of abundances of proteoforms that could not be inferred individually.

References

Cox, J. and Mann, M. (2008). MaxQuant enables high peptide identification rates, individualized ppb-range mass accuracies and proteome-wide protein quantification. Nature biotechnology *26*, 1367–1372.

Creech, A. L., Taylor, J. E., Maier, V. K., Wu, X., Feeney, C. M., Udeshi, N. D., Peach, S. E., Boehm, J. S., Lee, J. T., Carr, S. A. et al. (2015). Building the Connectivity Map of epigenetics: Chromatin profiling by quantitative targeted mass spectrometry. Methods *72*, 57–64.

Malioutov, D. and Slavov, N. (2014). Convex Total Least Squares. Journal of Machine Learning Research *32*, 109 – 117.

Slavov, N., Budnik, B., Schwab, D., Airoldi, E. and van Oudenaarden, A. (2014). Constant Growth Rate Can Be Supported by Decreasing Energy Flux and Increasing Aerobic Glycolysis. Cell Reports *7*, 705 – 714.