- Protein Quantification

Let $G_i = \{P_1, P_2, \cdots, P_k\}$ be the proteins in the proteome, $i, k \in \mathbb{N}^+$.

Define $\mathcal{P} = \{p_1, p_2, \cdots, p_n\}$ as the identified peptides. $n \in \mathbb{N}^+$

Let S_i be the quantitative intensity for each peptide p_i.

Note that p_i is generated from G_i.

★ Classic Method of Weighted Mean

$S = \sum_{i=1}^{n} w_i S_i$ with $\sum w_i = 1$ where $i \in \mathbb{N}^+$ and w_i is the weight of the peptide.

↓

But it doesn't consider the uncertainty of observation !!!

↓

Common sense: Variance represents uncertainty

↓

Volatility

$$Var(X) = E[(X-\mu)^2]$$

↓

$Var(X) \uparrow \rightarrow$ Unreliable $\uparrow$

We can't get the true variance from Mass Spectrometry ὃ▵ὃ

because MS is not an ideal, infinite, repeatable measurement.

>< You said you'd rely on "repetitions" to estimate the variance?

Nope!!! The estimation $= \sum$ (Var of noise + Var of biological var + instability of instrument)

|

QuantUMS = Quantification Using Minimum Uncertainty in MS

[On the premise of peptide-protein mapping, using all available signals for the optimal protein quantification and make this quantity the most reliable.]

Which one is reliable? Which one is not reliable?

⇩

Uncertainty → Variance Modeling

Logic:

① Background: Do protein quantification for a precursor p.

Let $s_1, \dots, s_N$ be the signal for either MS1 or MS2 fragmentation ion, $N \in \mathbb{N}^+$.

Let $x_i \in \mathbb{R}^+$ be the observed intensity and let $C_i \in [0,1]$ be the score.

x_i is $\log_2$ ratio.

Let δ represent Variance.

Goal: Construct the optimal estimation $\hat{x}$ to express the true value of precursor.

$$\hat{x} = \frac{\sum_{i=1}^{N} w_i x_i}{\sum_{i=1}^{N} w_i} \quad \text{where} \quad w_i \propto \frac{1}{\delta_i^2} \qquad (1)$$

How to build models for δ?

② Log-scale linear model:

$$\log(\delta_i^2) = \theta_1 T_s^{(i)} + \theta_2 T_c^{(i)} + \theta_3 T_{s.c}^{(i)}$$

where $T_s^{(i)} = S_i^{-1}$, $T_c^{(i)} = 1 - \sqrt{C_i}$, $T_{s.c}^{(i)} = \sqrt{T_s^{(i)} \cdot T_c^{(i)}}$, $\theta_k = (\theta_k')^2$, $k = 1,2,3$

Then $\delta_i^2 = e^{\theta_1 T_s^{(i)} + \theta_2 T_c^{(i)} + \theta_3 T_{s.c}^{(i)}}$

Thus $w_i = (\delta_i^2)^{-1} = e^{-(\theta_1 T_s^{(i)} + \theta_2 T_c^{(i)} + \theta_3 T_{s.c}^{(i)})} = f_i(\theta_1, \theta_2, \theta_3)$ (2)

③ Bias correction for both x_i and $\hat{x}_i$

$$B_i(\alpha_1, \alpha_2, \alpha_3) = \alpha_1 \sqrt{T_s^{(i)}} + \alpha_2 \sqrt{T_c^{(i)}} + \alpha_3 \sqrt{T_{s.c}^{(i)}}$$

$$x_i' = x_i + B_i(\alpha_1, \alpha_2, \alpha_3)$$

④ Fusion

plug (2) into (1),

1. Direct Fusion Method

$$\hat{x}_i(\theta_1, \theta_2, \theta_3) = \frac{\sum_{i=1}^N w_i x_i}{\sum_{i=1}^N w_i}$$

$$= \frac{\sum_{i=1}^N f_i(\theta_1, \theta_2, \theta_3) \cdot x_i}{\sum_{i=1}^N f_i(\theta_1, \theta_2, \theta_3)}$$

$$= \frac{\sum_{i=1}^N e^{-(\theta_1 T_s^{(i)} + \theta_2 T_c^{(i)} + \theta_3 T_{s.c}^{(i)})} \cdot x_i}{\sum_{i=1}^N e^{-(\theta_1 T_s^{(i)} + \theta_2 T_c^{(i)} + \theta_3 T_{s.c}^{(i)})}} \qquad (3)$$

2. Iterative propagation across acquisition

For each fragmentation m, from run i refer to run j

$$x_{m,j,i} = q_i + s_{m,j} - s_{m,i}$$

Weighted average for i: $$x_{m,j} = \frac{\sum_i w_{m,j,i} \cdot x_{m,j,i}}{\sum_i w_{m,j,i}}$$

Weighted average for m: $$\hat{x}_j = \frac{\sum_m w_m \cdot x_{m,j}}{\sum_m w_m}$$

⑤

1. Loss function classic

$$\ell(\theta_1, \theta_2, \theta_3, \alpha_1, \alpha_2, \alpha_3) = \sum_p \sum_{i=1}^{N} f_i(\theta_1, \theta_2, \theta_3) \cdot [x_i' - \hat{x}_i(\theta_1, \theta_2, \theta_3, \alpha_1, \alpha_2, \alpha_3)]^2$$

$$\ell(\theta, \alpha) = \sum_p \sum_{i=1}^{N} f_i(\theta) \cdot [x_i + B_i(\alpha) - \hat{x}_i(\theta, \alpha)]^2$$

↓

Weighted Least squares

In conclusion, if x_i and $\hat{x}_i$ have large difference, then w_i is small

Essentially, Self-supervised regression model + Weighted average model

2. Loss function - QuantUMS

- Precision loss

$$\ell_{prec} = \frac{1}{N} \sum_{m,i} (x_{m,i} - s_{m_{sel},i})^2$$

where $x_{m,i}$: The estimate value of feature m at aquisition i

$S_{m_{sel},i}$: the signal of best m at run i

[According to score C. QuantUMS chose the highest average score]

- Accuracy loss:

deviation ↑ relative abundance ↑

$$l_{acc} = \frac{\sum_{m,i} (x_{m,i} - S_{m_{sel},i})(S_{m_{sel},i} - \mu) w_i}{\sum_{m,i} w_i^2 \cdot Var},$$

where $S_i \sim N(\mu, Var)$,

$w_i' = e^{-S_{m_{sel},i}}$

- Class problem : Signal with low abundance are weak and easy to be ignore.

↓

Without any correction, weak signals has been "averaged out".

↓

"Ratio compression"

Ex: low abunce ($S_{m_{sel},i} < \mu$)

$x_{m,i} - S_{m_{sel},i}$ <0 ↙ ↘ $S_{m_{sel},i} - \mu$ <0

×

$\downarrow$

>0

High abunce $\rightarrow$ product >0

$\downarrow$

$W_i \uparrow$ when signal is weaker

$\downarrow$

- Optimize L_{acc} :

Low signal will not be undervalued

High overvalued

$\rightarrow$ formula ≈ 0

In conclusion,

$\mathcal{L}(\theta, \alpha) = l_{prec} + \lambda l_{acc}$, λ is a constant

$\nabla \mathcal{L} = \nabla_\theta l_{prec} + \lambda \nabla_\theta l_{acc}$

$\nabla \mathcal{L} = \nabla_\alpha l_{prec} + \lambda \nabla l_{acc}$

⑥ Backpropagation, update θ, a

Goal: minimize $\mathcal{L}(\theta, \alpha)$

$$\frac{\partial \mathcal{L}}{\partial \theta_k} \qquad \frac{\partial \mathcal{L}}{\partial \alpha_k}$$

$k=1,2,3$

- Gradient descent:

$$\theta_k^{(t+1)} = \theta_k^{(t)} - \eta \cdot \frac{\partial \mathcal{L}}{\partial \theta_k}$$

$$\alpha_k^{(t+1)} = \alpha_k^{(t)} - \eta \cdot \frac{\partial \mathcal{L}}{\partial \alpha_k}$$, η is the learning rate

↓ I know I need to follow the current gradient,

but how far I should go? (How large is the η ?)

↓

Automatically adjust η !!!

↓

- Armijo condition:

$$\mathcal{L}(x - \eta \nabla \mathcal{L}) \leq \mathcal{L}(x) - c \cdot \eta \cdot \|\nabla \mathcal{L}\|^2$$

where $\mathcal{L}(x)$: loss function

x : θ or a

d : -1

η : learning rate

η: learning rate

$\nabla f(x)^T$: $\frac{\partial \mathcal{L}}{\partial \theta_k}$ or $\frac{\partial \mathcal{L}}{\partial \alpha_k}$

C : Constant — Control the tolerance level of declining

Euclideam Norm:

$$\|\nabla f(\theta, \alpha)\|^2 = \sum_i \left(\frac{\partial f}{\partial \theta_i}\right)^2 + \sum_i \left(\frac{\partial f}{\partial \alpha_i}\right)^2$$

↓ Why we use Armijo condition?

$\because$ $\mathcal{L}(\theta, \alpha)$ is non-linear and has nested structure

$\therefore$ It's very easy that big η cause the increase of loss.

$\therefore$ We need to calculate η.

Thus, input: x_i, θ, α

↓

linear model: δ^2, w, bias

fusion: $\hat{x}$

Loss func: $\mathcal{L}(\theta, \alpha)$

↓

backpropagation: $\frac{\partial \mathcal{L}}{\partial \theta_k}$, $\frac{\partial \mathcal{L}}{\partial \alpha_k}$

Armijo: η

↓

update θ, α

↓

Until $L(\theta, \alpha)$ is minimized and converged.

Explaination for Fig 1.

	Human	E.coli
A	1	50
B	1	33
C	1	20

A/C	Human	E.coli
FC Fold change	1	2.5
$\log_2 FC$	0	1.32