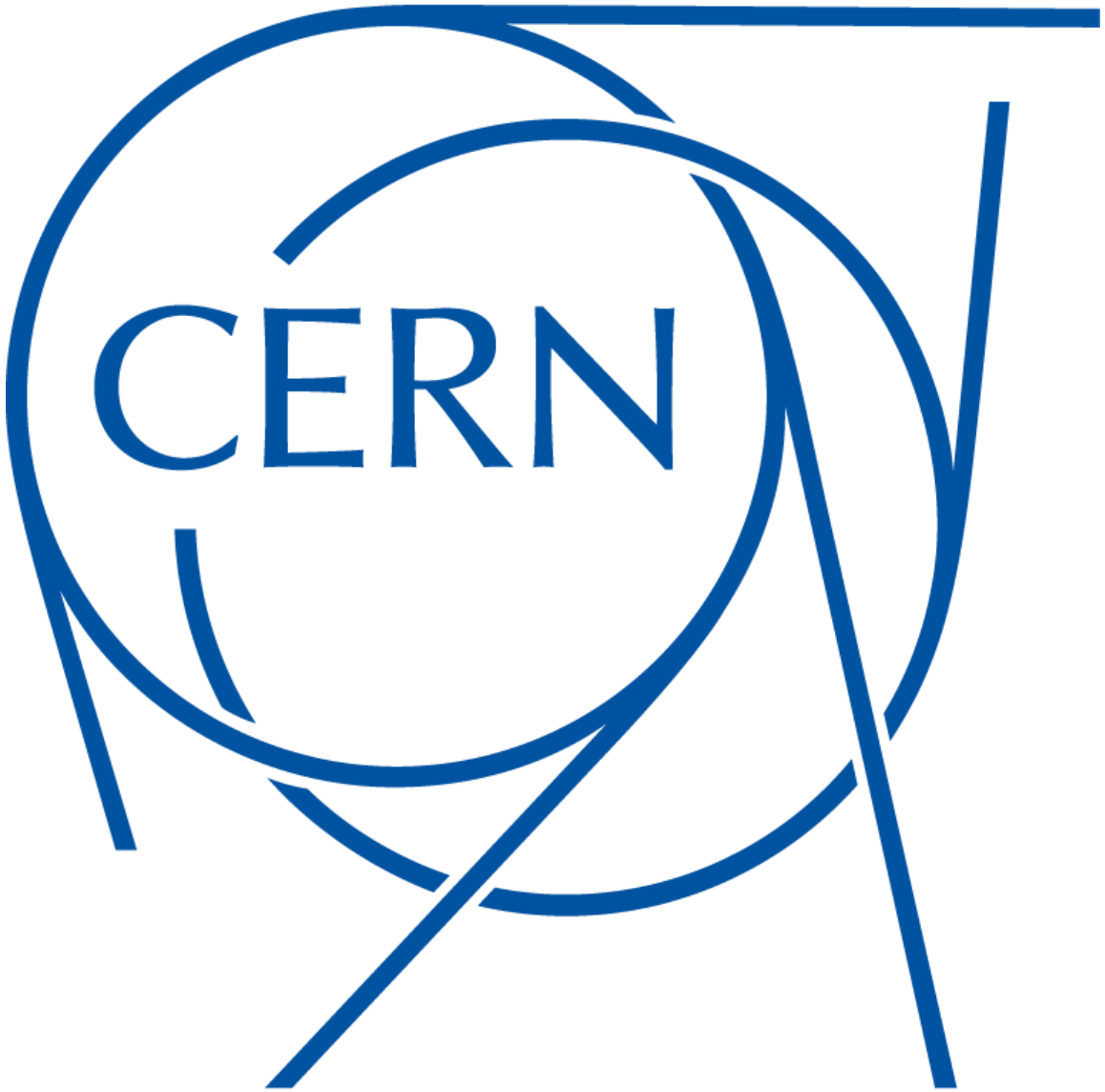
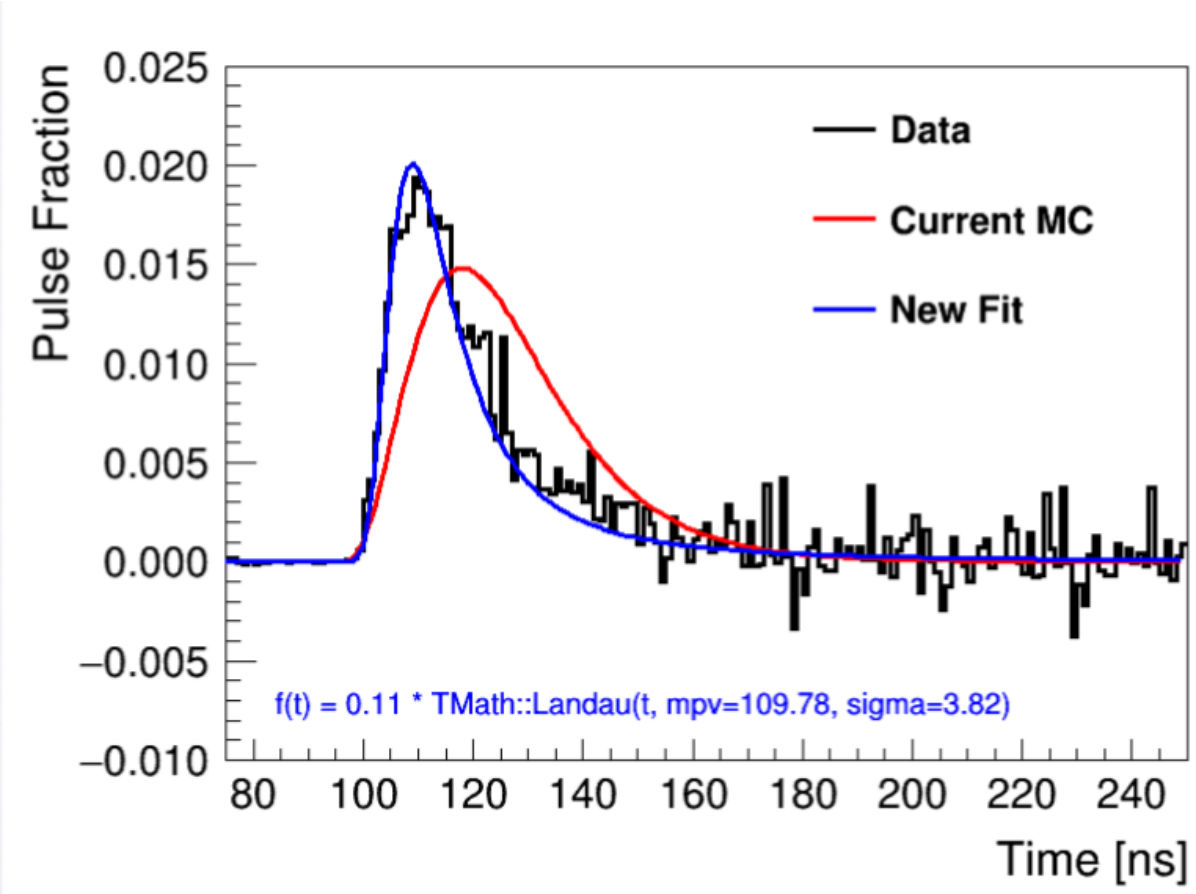
Non-negative least squares

Marco Barbone



Regression problem



Min Chi-Square

$$X^2 = (\vec{y} - P\vec{x})^{-1} Cov^{-1} (\vec{y} - P\vec{x})$$

Sample heights

Amplitude

Expected sample heights

Min Chi-Square

$$X^2 = \left(\vec{S} - A \times \vec{F}\right)^{-1} Cov^{-1} \left(\vec{S} - A \times \vec{F}\right)$$

Sample heights

Amplitude

Expected sample heights

There are positivity constrains

Input -> Output

Inputs:

1. Real valued matrix *Z*
2. Known terms vector *b*

Output:

1. Solution vector *x*

NNLS Initialization (1/4)

A. Initialization

A1. $P = \varnothing$

A2. $R = \{1, 2, \ldots, M\}$

A3. $\mathbf{d} = \mathbf{0}$

A4. $\mathbf{w} = \mathbf{Z}^T(\mathbf{x} - \mathbf{Z}\mathbf{d})$

NNLS Main loop (2/4)

B. Main loop

B1. Proceed if $\mathrm{R} \neq \varnothing \wedge [\max_{n \in \mathrm{R}}(w_n) > \text{tolerance}]$

B2. $m = \underset{n \in \mathrm{R}}{\operatorname{argmax}}(w_n)$

B3. Include the index m in P and remove it from R

B4. $\mathbf{s}^{\mathrm{P}} = [(\mathbf{Z}^{\mathrm{P}})^{\mathrm{T}}\mathbf{Z}^{\mathrm{P}}]^{-1}(\mathbf{Z}^{\mathrm{P}})^{\mathrm{T}}\mathbf{x}$

NNLS Inner loop (3/4)

C. Inner loop

C1. Proceed if $\min(\mathbf{s}^{\mathrm{P}}) \leq 0$

C2. $\alpha = -\min_{n \in \mathrm{P}} [d_n / (d_n - s_n)]$

C3. $\mathbf{d} := \mathbf{d} + \alpha(\mathbf{s} - \mathbf{d})$

C4. Update R and P

C5. $\mathbf{s}^{\mathrm{P}} = [(\mathbf{Z}^{\mathrm{P}})^{\mathrm{T}} \mathbf{Z}^{\mathrm{P}}]^{-1} (\mathbf{Z}^{\mathrm{P}})^{\mathrm{T}} \mathbf{x}$

C6. $\mathbf{s}^{\mathrm{R}} = \mathbf{0}$

NNLS loop end (4/4)

end C

B5. $\mathbf{d}=\mathbf{s}$

B6. $\mathbf{w}=\mathbf{Z}^{\mathrm{T}}(\mathbf{x}-\mathbf{Z}\mathbf{d})$

end B

Bottleneck

$$s^p = [(z^p)^T(z^p)]^{-1}(z^p)x$$

1. Numerical problems -> Decompositions
2. Low performance -> FNNLS

Numerical issues

QR decomposition solve this problems and also improve the performances

Performances improvement: FNNLS

$$\mathbf{w} = (\mathbf{Z}^{\mathrm{T}}\mathbf{x}) - (\mathbf{Z}^{\mathrm{T}}\mathbf{Z})\mathbf{d}$$

$$\mathbf{s}^{\mathrm{P}} = [(\mathbf{Z}^{\mathrm{T}}\mathbf{Z})^{\mathrm{P}}]^{-1}(\mathbf{Z}^{\mathrm{T}}\mathbf{x})^{\mathrm{P}}$$

Check list

- Understand the problem
- CPU version running
- Test passing
- GPU version running
- Test passing

Todo list

- Get LDLT and LLT working
- Plug the algorithm into the codebase
- Profiling
- Optimization
- Send to nvidia for further improvements

Thanks!