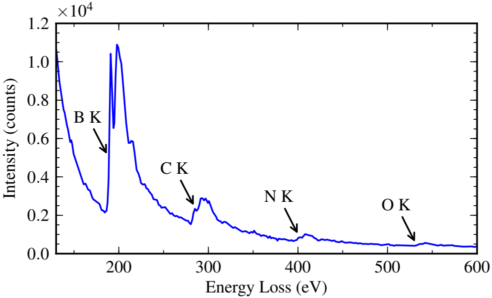
Introduction to EELS curve fitting

Francisco de la Peña

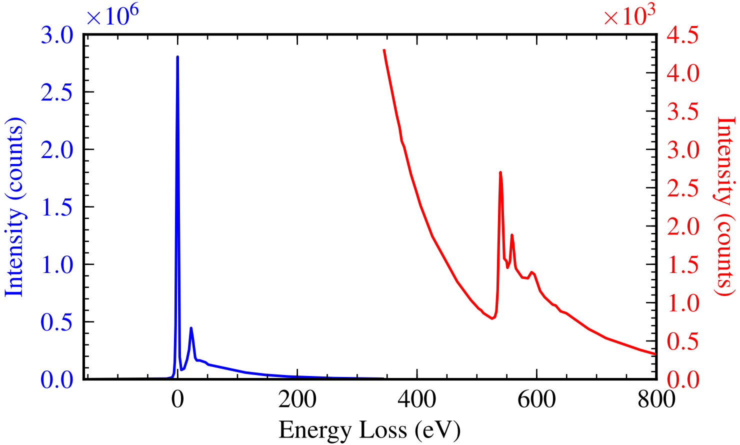
Université de Lille

HyperSpy Workshop 2021
ePSIC Diamond Light Source (Cloud)
20th of April 2021

EELS spectrum from BN NPs

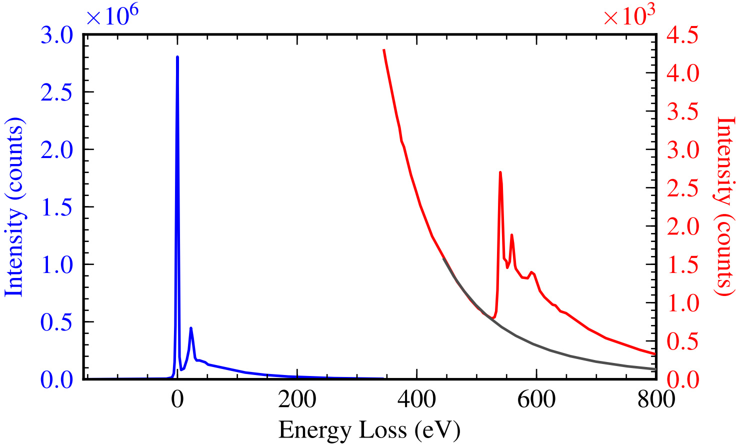


The “windows” method



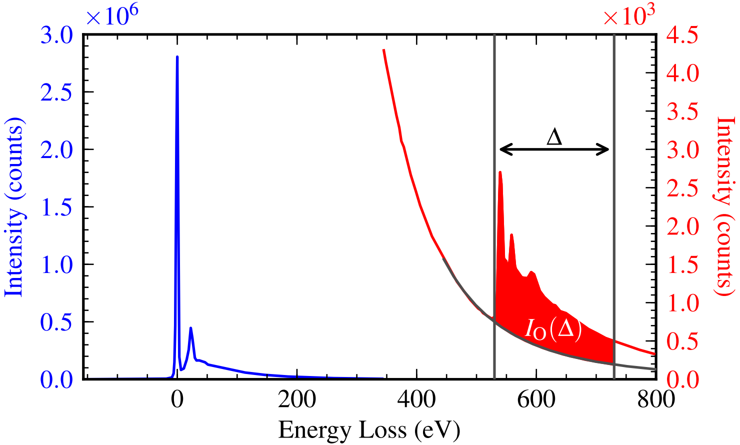
The “windows” method

$$N_O \approx \text{———}$$



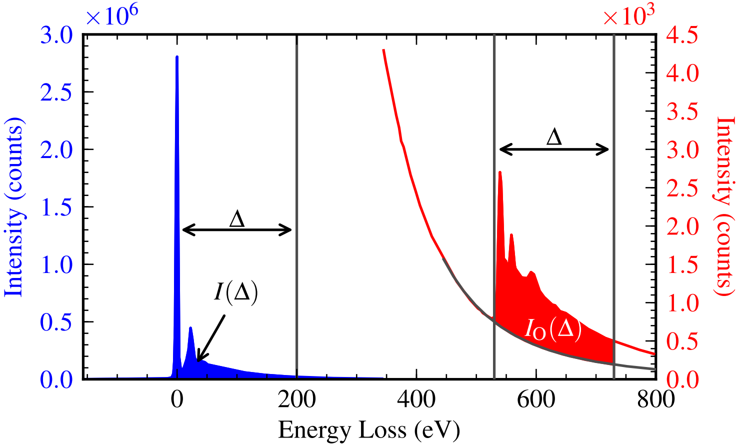
The "windows" method

$$N_{\mathrm{O}} \approx \frac{I_{\mathrm{O}}(\Delta,\beta)}{}$$



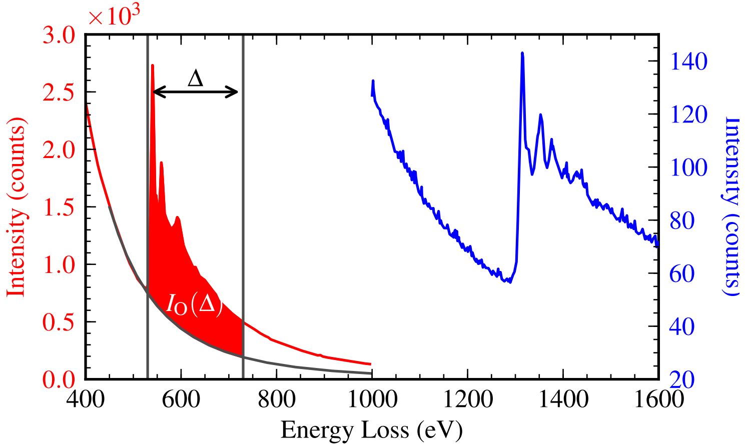
The "windows" method

$$N_O \approx \frac{I_O(\Delta,\beta)}{I(\Delta,\beta)}\sigma_O^{-1}(\Delta,\beta)$$



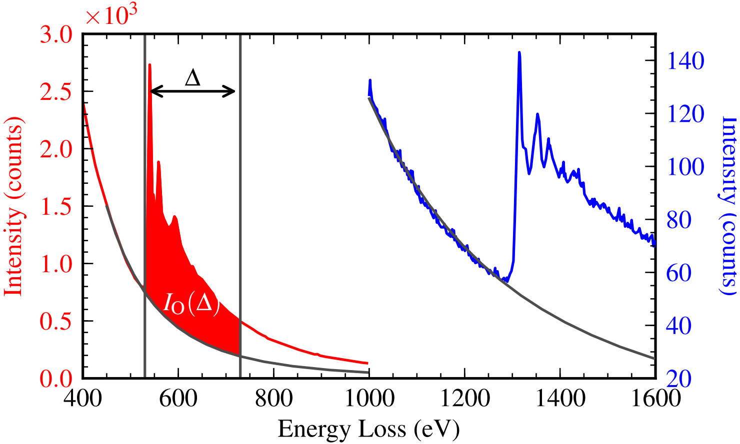
The "windows" method

$$N_{\mathrm{O}} \approx \frac{I_{\mathrm{O}}(\Delta,\beta)}{I(\Delta,\beta)}\sigma_{\mathrm{O}}^{-1}(\Delta,\beta) \qquad N_{\mathrm{Mg}} \approx \frac{I_{\mathrm{Mg}}(\Delta,\beta)}{I(\Delta,\beta)}\sigma_{\mathrm{Mg}}^{-1}(\Delta,\beta)$$



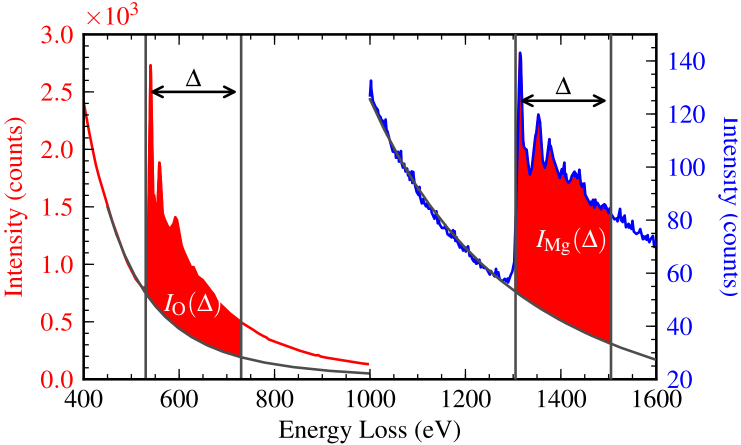
The "windows" method

$$N_{\mathrm{O}} \approx \frac{I_{\mathrm{O}}(\Delta,\beta)}{I(\Delta,\beta)}\sigma_{\mathrm{O}}^{-1}(\Delta,\beta) \qquad N_{\mathrm{Mg}} \approx \frac{I_{\mathrm{Mg}}(\Delta,\beta)}{I(\Delta,\beta)}\sigma_{\mathrm{Mg}}^{-1}(\Delta,\beta)$$



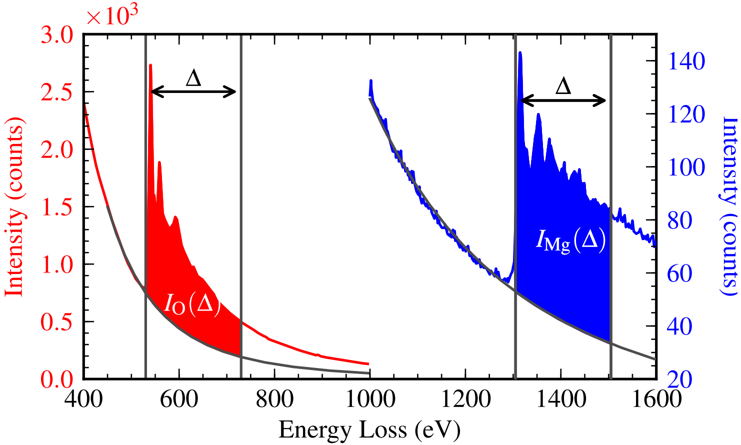
The "windows" method

$$N_{\mathrm{O}} \approx \frac{I_{\mathrm{O}}(\Delta,\beta)}{I(\Delta,\beta)}\sigma_{\mathrm{O}}^{-1}(\Delta,\beta) \qquad N_{\mathrm{Mg}} \approx \frac{I_{\mathrm{Mg}}(\Delta,\beta)}{I(\Delta,\beta)}\sigma_{\mathrm{Mg}}^{-1}(\Delta,\beta)$$

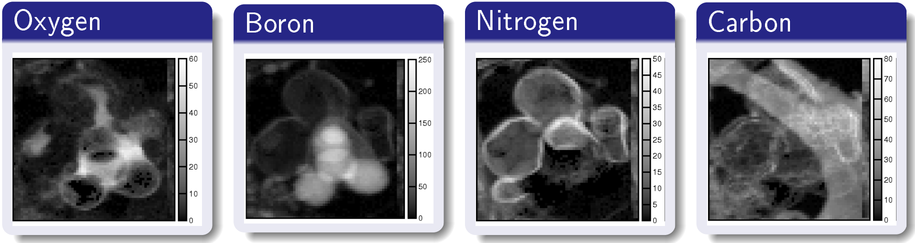


The "windows" method

$$\frac{N_{\mathrm{O}}}{N_{\mathrm{Mg}}} \approx \frac{I_{\mathrm{O}}(\Delta,\beta)}{I_{\mathrm{Mg}}(\Delta,\beta)} \frac{\sigma_{\mathrm{Mg}}(\Delta,\beta)}{\sigma_{\mathrm{O}}(\Delta,\beta)}$$



EELS elemental of BN nanoparticle



Arenal et al., Ultramicroscopy 2008

Some limitations of the “windows” method

- Overlapping edges

Some limitations of the "windows" method

- Overlapping edges
- It always returns a result (what feels good) but, how do we know that it is correct?

Some limitations of the "windows" method

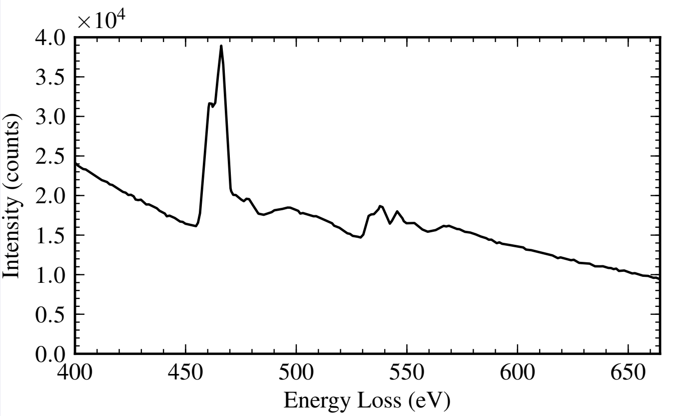
- Overlapping edges
- It always returns a result (what feels good) but, how do we know that it is correct?
- Only analyses a fraction of the available signal (non-optimal SNR)

Some limitations of the "windows" method

- Overlapping edges
- It always returns a result (what feels good) but, how do we know that it is correct?
- Only analyses a fraction of the available signal (non-optimal SNR)
- Useful information gets lost (fine structures changes, energy onset shifts...)

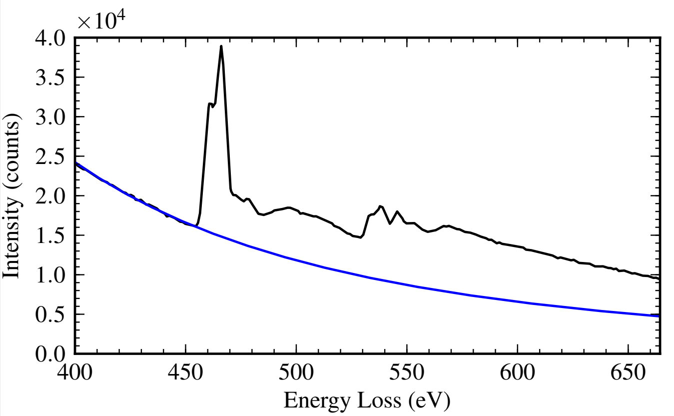
The curve fitting method: an example

$SrTiO_3$ Spectrum



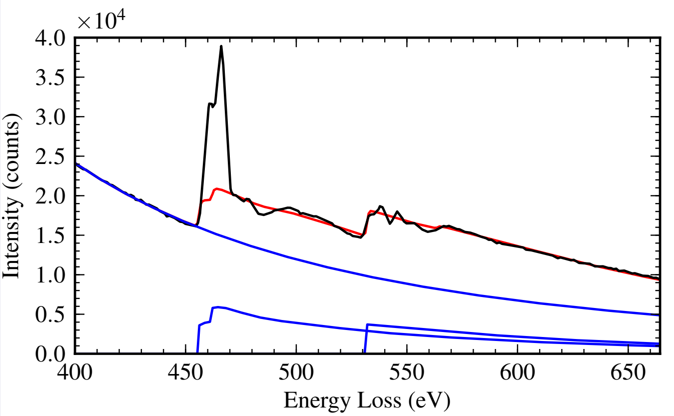
The curve fitting method: an example

$M(E) = AE^{-r}$



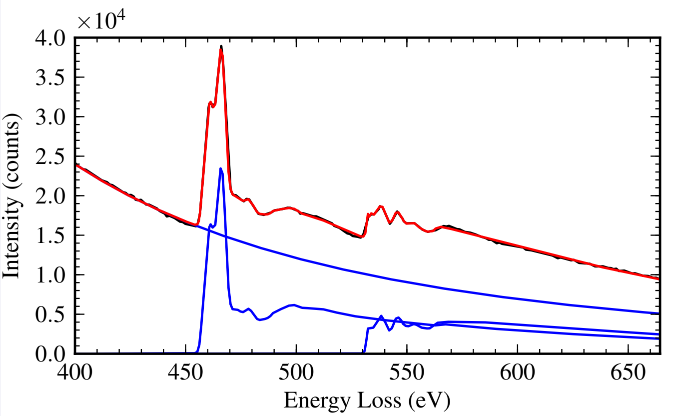
The curve fitting method: an example

$$M(E) = AE^{-r} + I_{\mathrm{Ti}}\sigma_{\mathrm{Ti}}(E) + I_{\mathrm{O}}\sigma_{\mathrm{O}}(E)) * L(E)$$



The curve fitting method: an example

$$M(E) = AE^{-r} + (N_{\mathrm{Ti}} f_{\mathrm{Ti}}(E)\sigma_{\mathrm{Ti}}(E) + N_{\mathrm{O}} f_{\mathrm{O}}(E)\sigma_{\mathrm{O}}(E)) * L(E)$$



Assumptions

- There is a *known* function, f, that relates the *independent variable* X and the *dependent variable* Y.
  $Y \approx f(X, \beta) + \epsilon(f(X, \beta))$

Assumptions

- There is a *known* function, f, that relates the *independent variable* X and the *dependent variable* Y.
  $Y \approx f(X, \beta) + \epsilon(f(X, \beta))$
- The number of unknown parameters, β is *equal or less* thant the number of different observations of the independent variable

Assumptions

- There is a *known* function, f, that relates the *independent variable* X and the *dependent variable* Y.
  $Y \approx f(X, \beta) + \epsilon(f(X, \beta))$
- The number of unknown parameters, β is *equal or less* thant the number of different observations of the independent variable
- The probability distribution of the statistical error (ϵ) is known

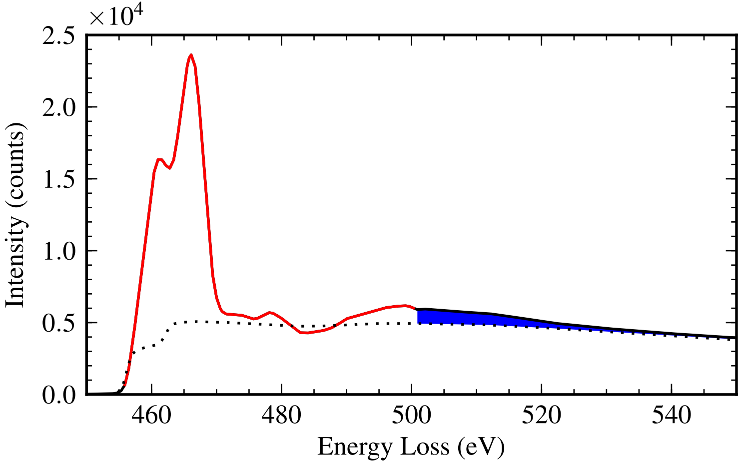
Components of the model

Parametric model of the high energy loss spectrum for elemental and bonding quantification:

$$M(E;\mathrm{parameters}) = AE^{-r} + \left(\sum_i N_i f_i\left(E\right) \int_0^{q(\beta)} \sigma_i\left(E,q\right) dq\right) * L(E)$$

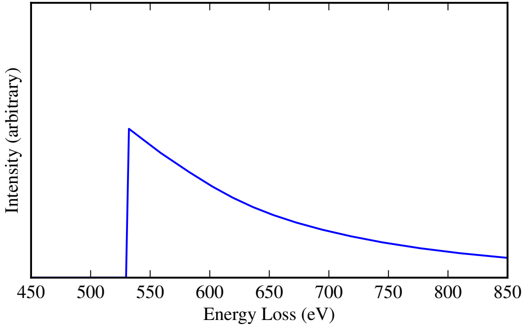
- AE^{-r}: background model
- σ_i^{FS}: cross section of each ionization edge, i
- N_i: $\mathrm{atoms}/\mathrm{nm}^2$
- $f_i(E)$: function that mimics the fine structure of each ionization edge, e.g. gaussian, fingerprints, splines...
- $L(E)$: experimental low loss spectrum.

Why adding the fine structure to the model?



Ionization edge fine structure

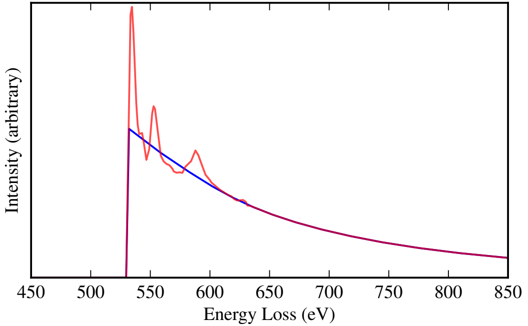
Oxygen K ionization edge from MgO



Ionization edge fine structure

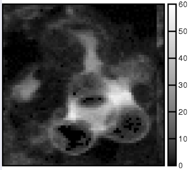
- In solids, the first $\sim 40\,\mathrm{eV}$ are strongly modified by the final density of states $\Rightarrow$ carries bonding information

Oxygen K ionization edge from MgO

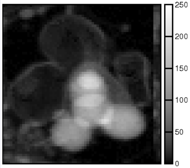


EELS elemental and bonding maps of BN nanoparticle

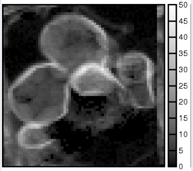
Oxygen



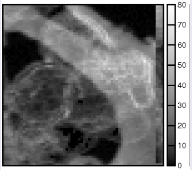
Boron



Nitrogen



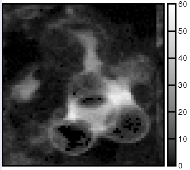
Carbon



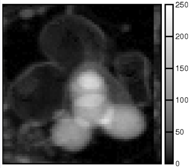
Arenal et al., Ultramicroscopy 2008

EELS elemental and bonding maps of BN nanoparticle

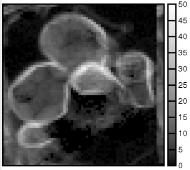
Oxygen



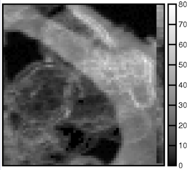
Boron



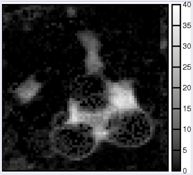
Nitrogen



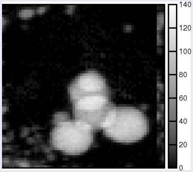
Carbon



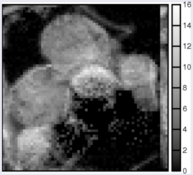
Boron oxide



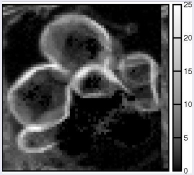
Boron pure



BN $\perp$



BN ||



Arenal et al., Ultramicroscopy 2008

Parameter value and error estimation

- Regression analysis, in addition to estimate the value of β, can estimate the error

Parameter value and error estimation

- Regression analysis, in addition to estimate the value of β, can estimate the error
- The most common estimation method is non-linear least squares *(NLLS)*, however this *assumes that the noise is Gaussian distributed*

Parameter value and error estimation

- Regression analysis, in addition to estimate the value of β, can estimate the error
- The most common estimation method is non-linear least squares *(NLLS)*, however this *assumes that the noise is Gaussian distributed*
- If the noise distribution is known (e.g. Poissonian) we can use:

Parameter value and error estimation

- Regression analysis, in addition to estimate the value of β, can estimate the error
- The most common estimation method is non-linear least squares *(NLLS)*, however this *assumes that the noise is Gaussian distributed*
- If the noise distribution is known (e.g. Poissonian) we can use:
    - Weighted non-linear least squares (WNNLS)

Parameter value and error estimation

- Regression analysis, in addition to estimate the value of β, can estimate the error
- The most common estimation method is non-linear least squares *(NLLS)*, however this *assumes that the noise is Gaussian distributed*
- If the noise distribution is known (e.g. Poissonian) we can use:
  - Weighted non-linear least squares (WNNLS)
  - Maximum likelihood estimation (ML)

To keep in mind

- The estimation of the parameters value and error *will be wrong* if the noise probability distribution is wrong

To keep in mind

- The estimation of the parameters value and error *will be wrong* if the noise probability distribution is wrong
- In EELS the noise is a mixture of *Poisson and Gaussian noise.*

To keep in mind

- The estimation of the parameters value and error *will be wrong* if the noise probability distribution is wrong
- In EELS the noise is a mixture of *Poisson and Gaussian noise.*
- WNNLS can approximate well Poissonian noise when the number of counts is high enough (almost always in EELS)

To keep in mind

- The estimation of the parameters value and error *will be wrong* if the noise probability distribution is wrong
- In EELS the noise is a mixture of *Poisson and Gaussian noise.*
- WNNLS can approximate well Poissonian noise when the number of counts is high enough (almost always in EELS)
- Non-linear parameter estimation is an iterative process that *is very sensitive to the starting parameters*

Key articles

- Steele, J., Titchmarsh, J., Chapman, J., and Paterson, J. (1985). A single-stage process for quantifying electron energy-loss spectra. Ultramicroscopy, 17(3):273–276.

Key articles

- Steele, J., Titchmarsh, J., Chapman, J., and Paterson, J. (1985). A single-stage process for quantifying electron energy-loss spectra. Ultramicroscopy, 17(3):273–276.
- Manoubi, T., Tencé, M., Walls, M. G., and Colliex, C. (1990). Curve fitting methods for quantitative analysis in electron energy loss spectroscopy. Microscopy Microanalysis Microstructures, 1(1):23.

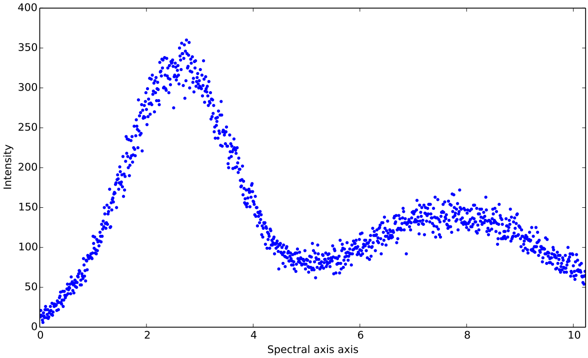
Key articles

- Steele, J., Titchmarsh, J., Chapman, J., and Paterson, J. (1985). A single-stage process for quantifying electron energy-loss spectra. Ultramicroscopy, 17(3):273–276.
- Manoubi, T., Tencé, M., Walls, M. G., and Colliex, C. (1990). Curve fitting methods for quantitative analysis in electron energy loss spectroscopy. Microscopy Microanalysis Microstructures, 1(1):23.
- Verbeeck, J. and Aert, S. V. (2004). Model based quantification of EELS spectra. Ultramicroscopy, 101(2-4):207–224.

Software

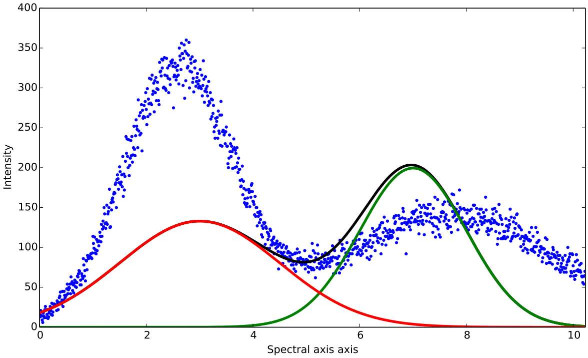
- EELSModel http://www.eelsmodel.ua.ac.be/ (open source)
- HyperSpy http://hyperspy.org (open source)
- Digital Micrograph

Non-linear optimisation routine



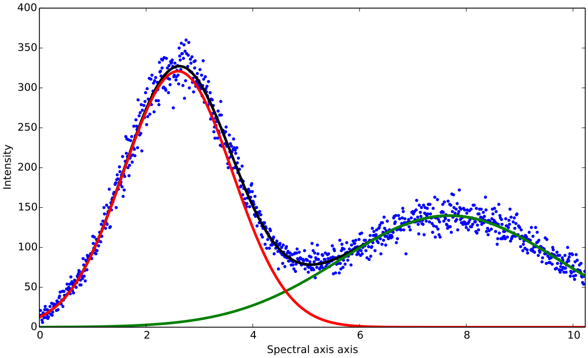
- Set stating parameters.
- Fit.

Non-linear optimisation routine



- Set stating parameters.
- Fit.

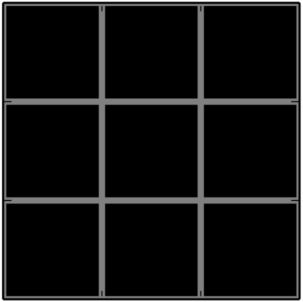
Non-linear optimisation routine



- Set stating parameters.
- Fit.

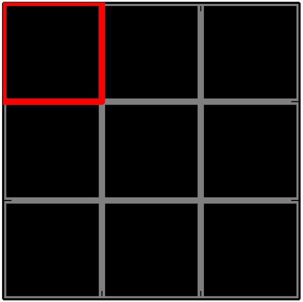
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



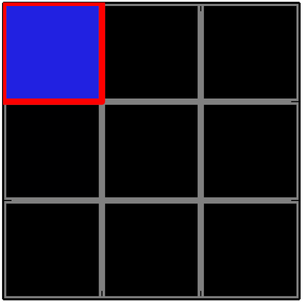
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



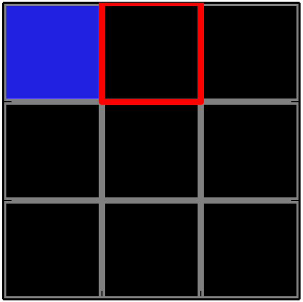
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



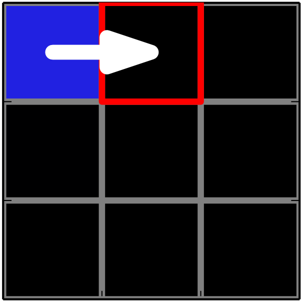
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



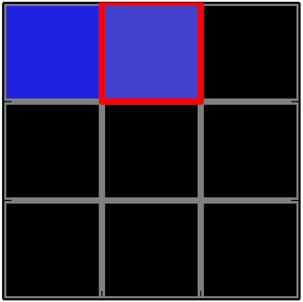
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



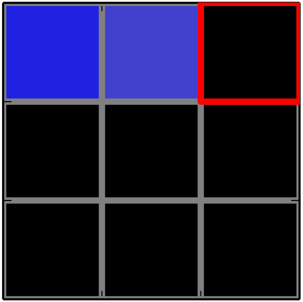
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



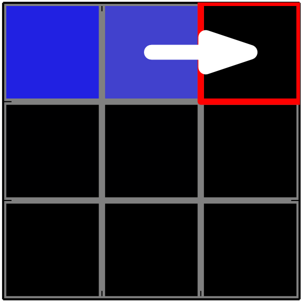
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



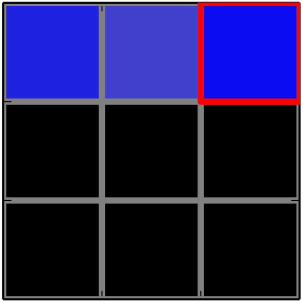
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



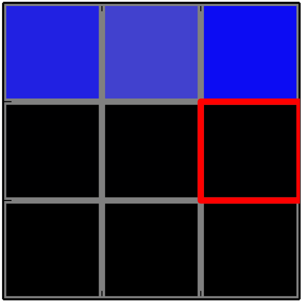
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



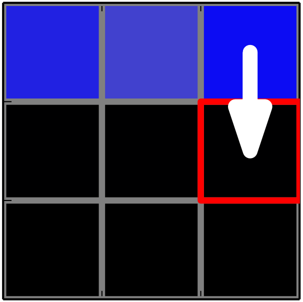
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



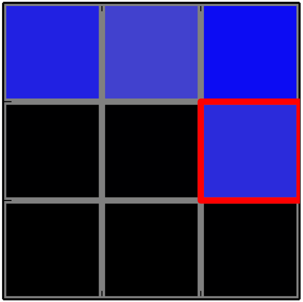
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



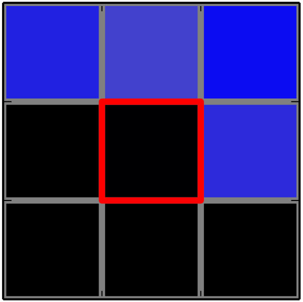
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



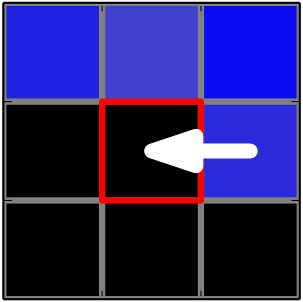
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



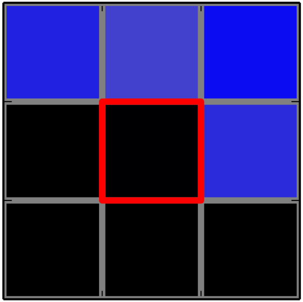
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



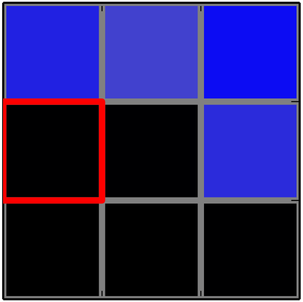
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



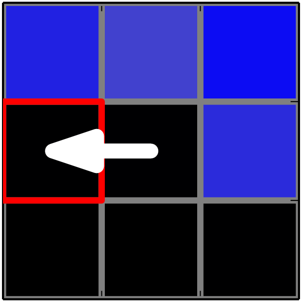
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



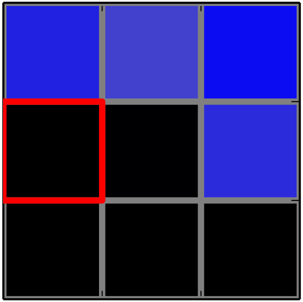
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



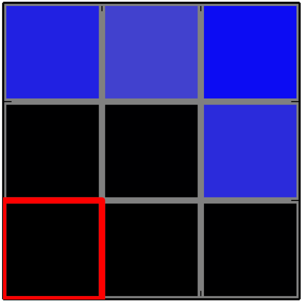
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



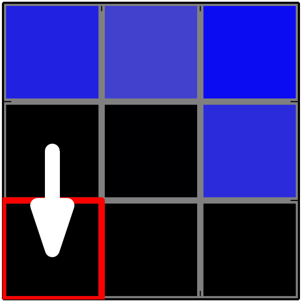
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



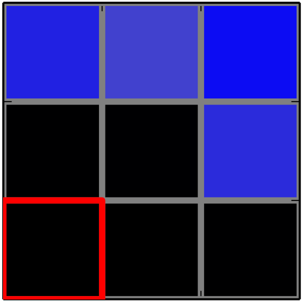
Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.



Fitting routine n-dimensions

- Set stating parameters.
- Fit.
- Move to next element.
- Copy parameter values from previous fit.