

Machine Learning for Multi-dimensional Spectroscopy Analysis

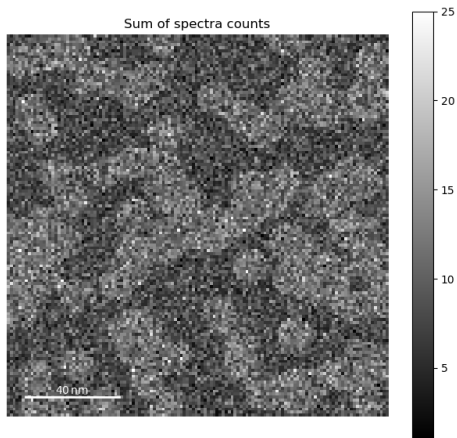


HyperSpy Workshop 2021
ePSIC Diamond Light Source (Cloud)
20th of April 2021

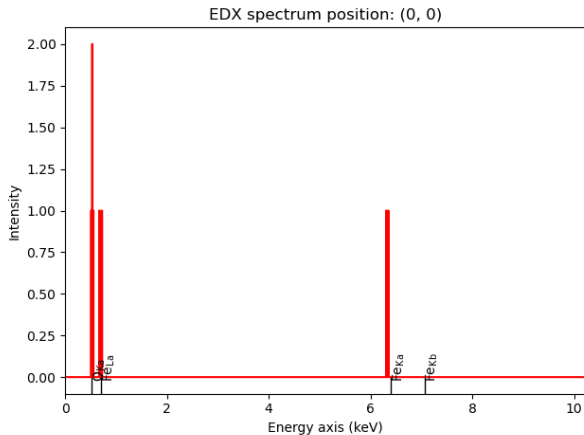
- 1 Composition segmentation
- 2 Low-rank approximation and blind source separation

- 1 Composition segmentation
- 2 Low-rank approximation and blind source separation

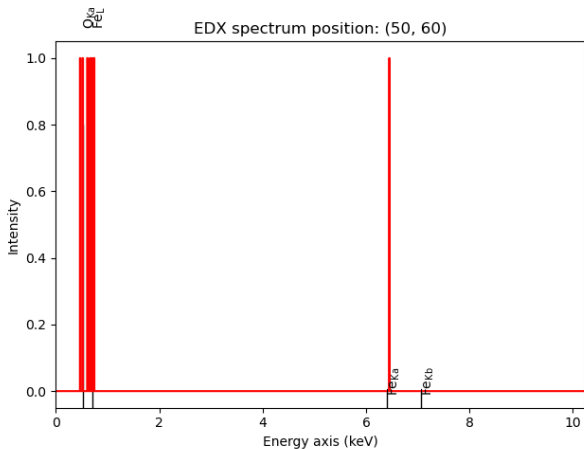
Homogeneous sample?



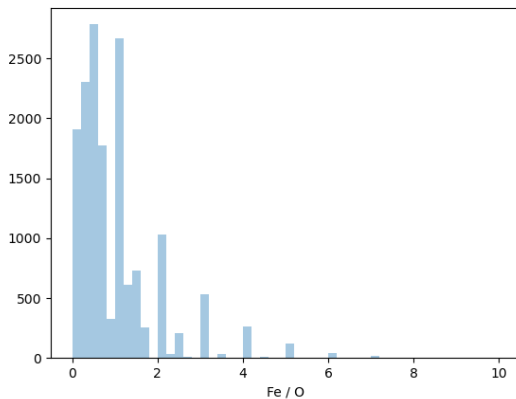
EDX spectrum from sample



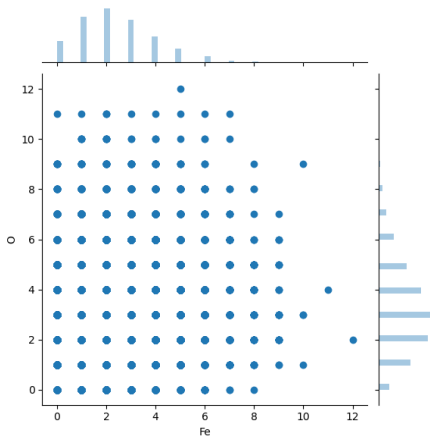
EDX spectrum from sample



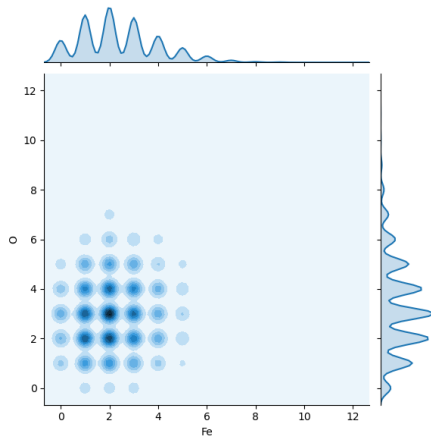
Fe / O ratio



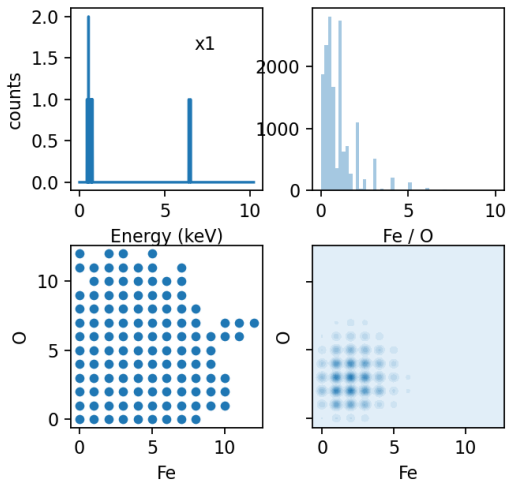
Scatter plot



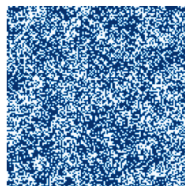
KDE plot



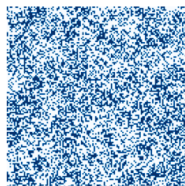
Classification by KMEANS clustering – the effect of SNR



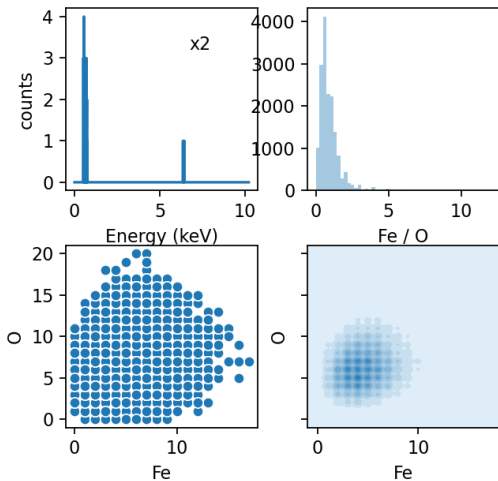
Fe / O = 0.507



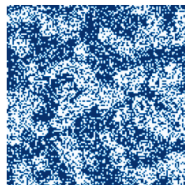
Fe / O = 1.161



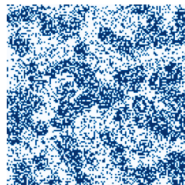
Classification by KMEANS clustering – the effect of SNR



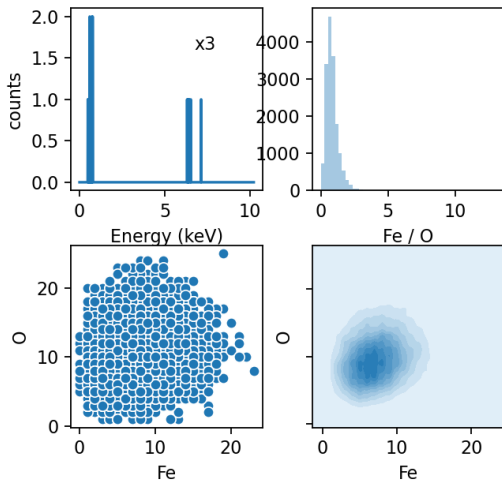
Fe / O = 0.846



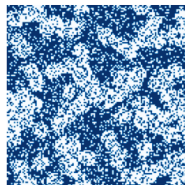
Fe / O = 0.682



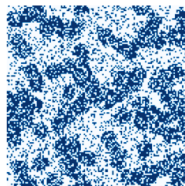
Classification by KMEANS clustering – the effect of SNR



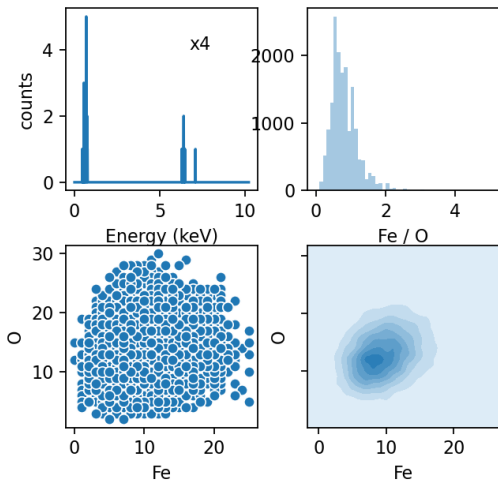
Fe / O = 0.732



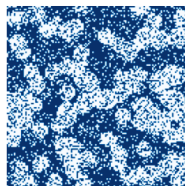
Fe / O = 0.778



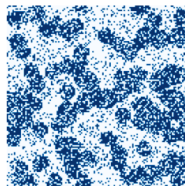
Classification by KMEANS clustering – the effect of SNR



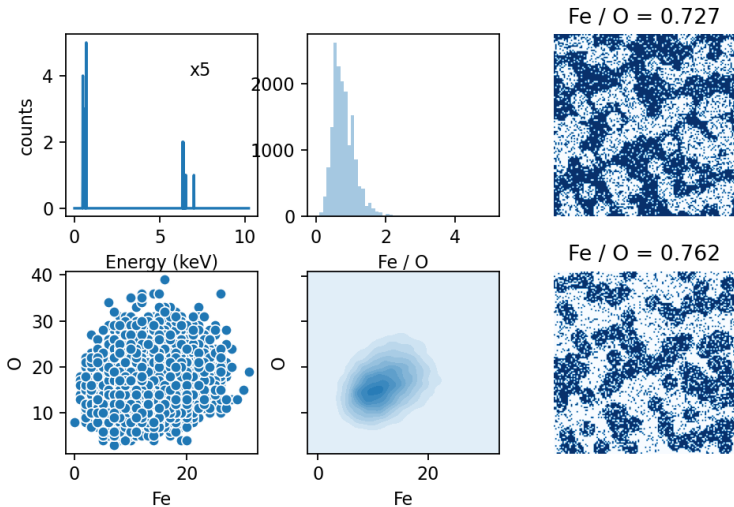
Fe / O = 0.748



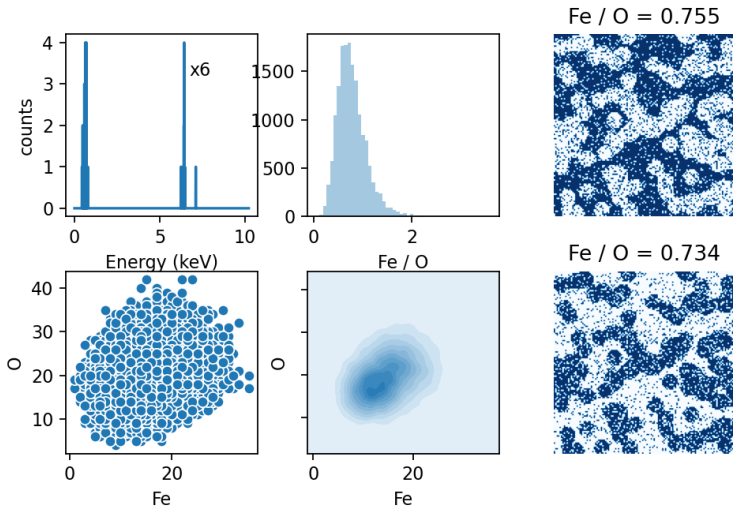
Fe / O = 0.744



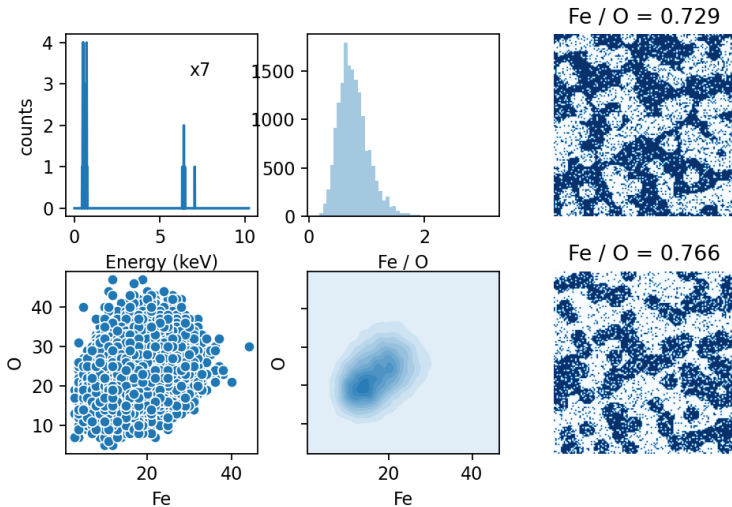
Classification by KMEANS clustering – the effect of SNR



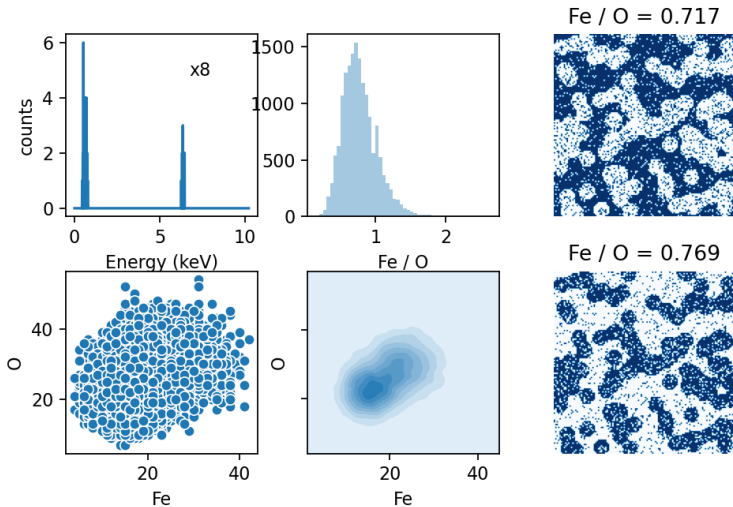
Classification by KMEANS clustering – the effect of SNR



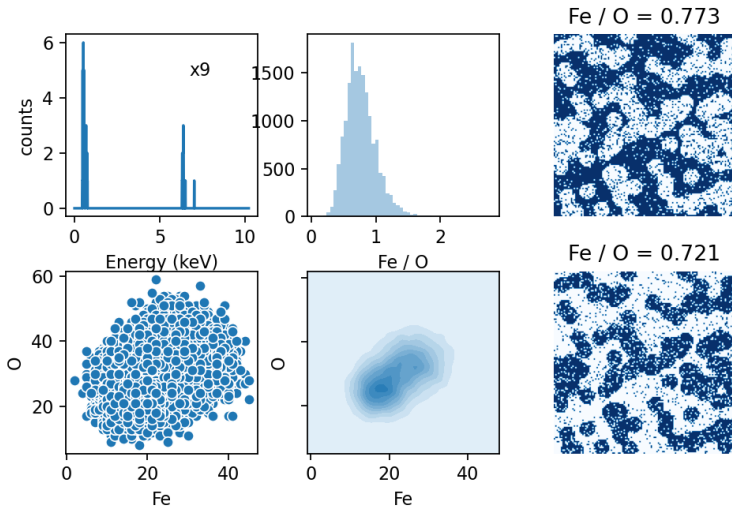
Classification by KMEANS clustering – the effect of SNR



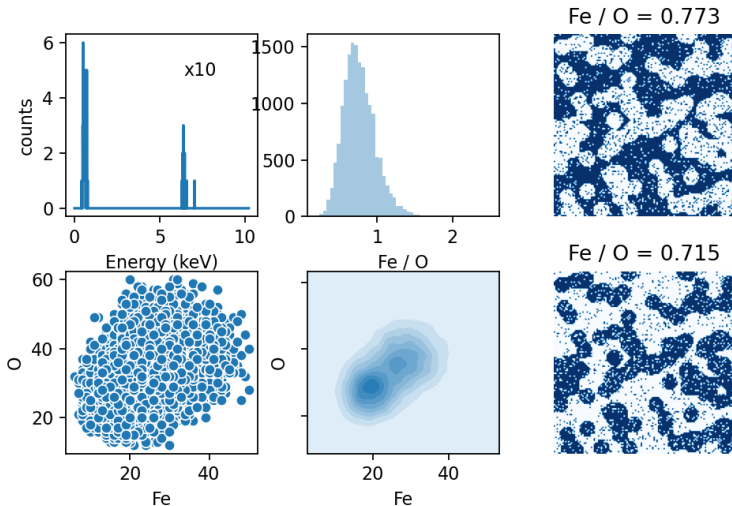
Classification by KMEANS clustering – the effect of SNR



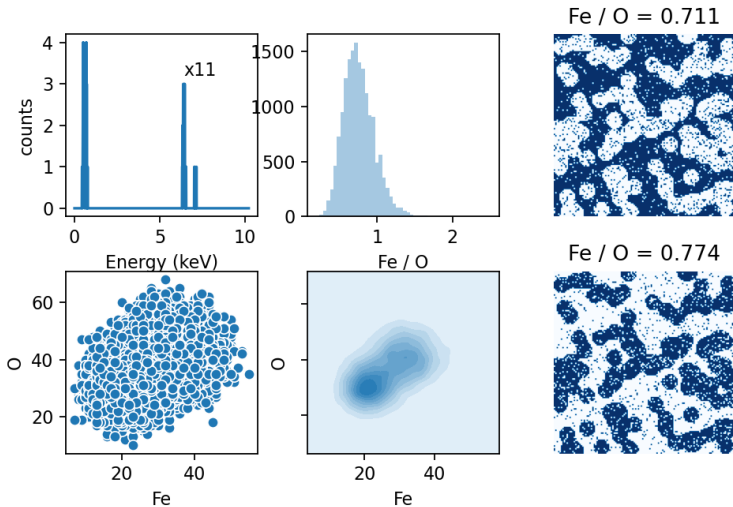
Classification by KMEANS clustering – the effect of SNR



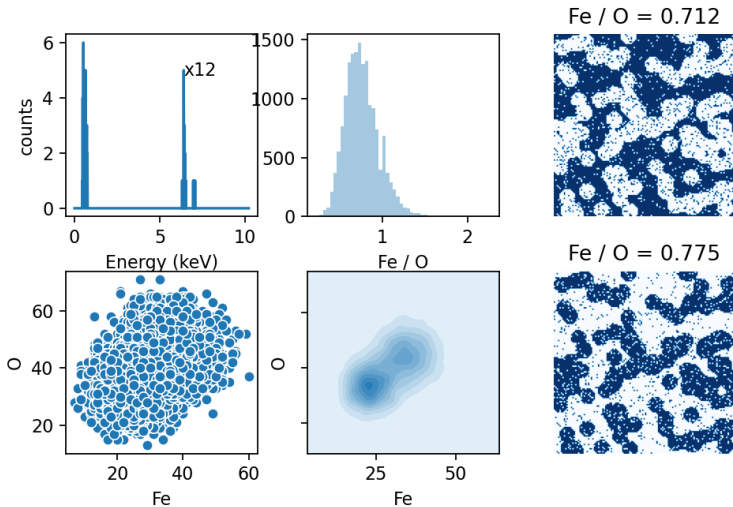
Classification by KMEANS clustering – the effect of SNR



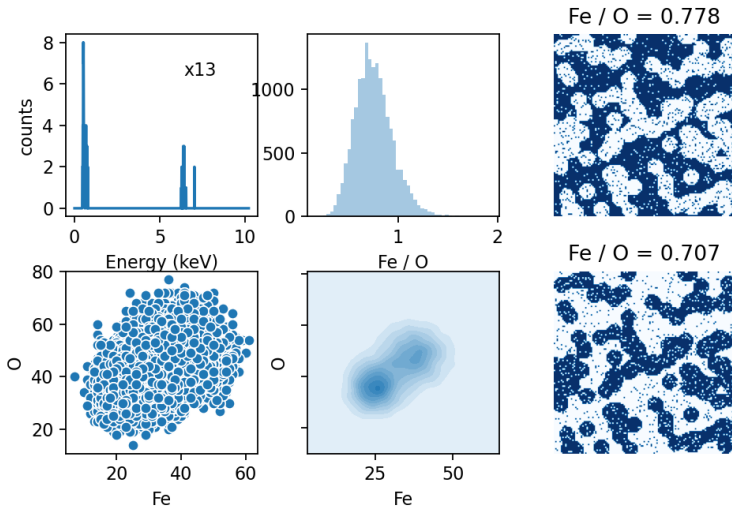
Classification by KMEANS clustering – the effect of SNR



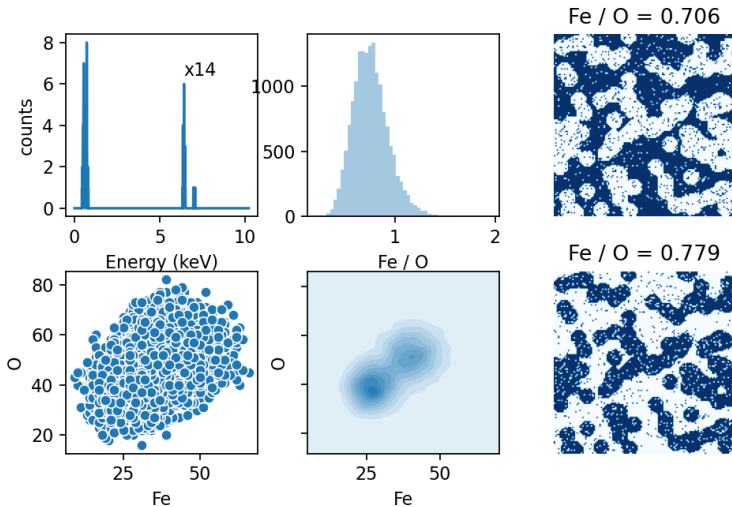
Classification by KMEANS clustering – the effect of SNR



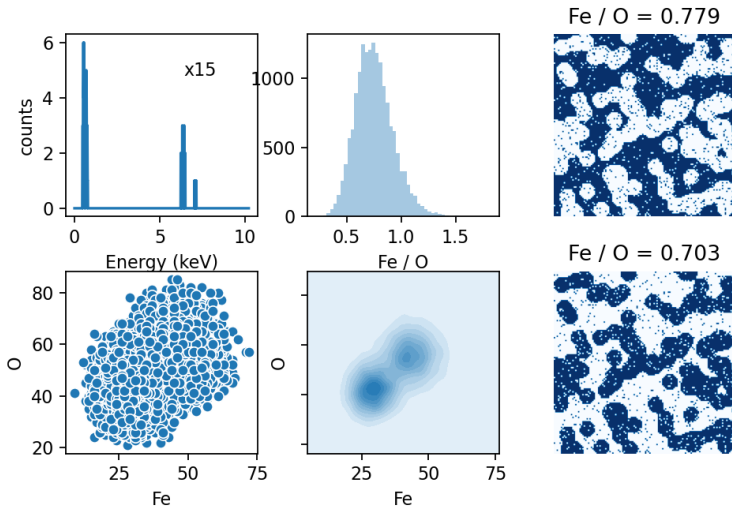
Classification by KMEANS clustering – the effect of SNR



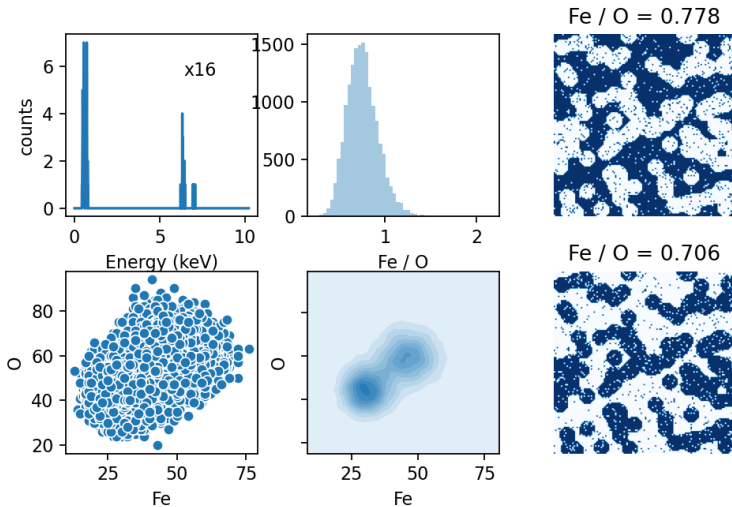
Classification by KMEANS clustering – the effect of SNR



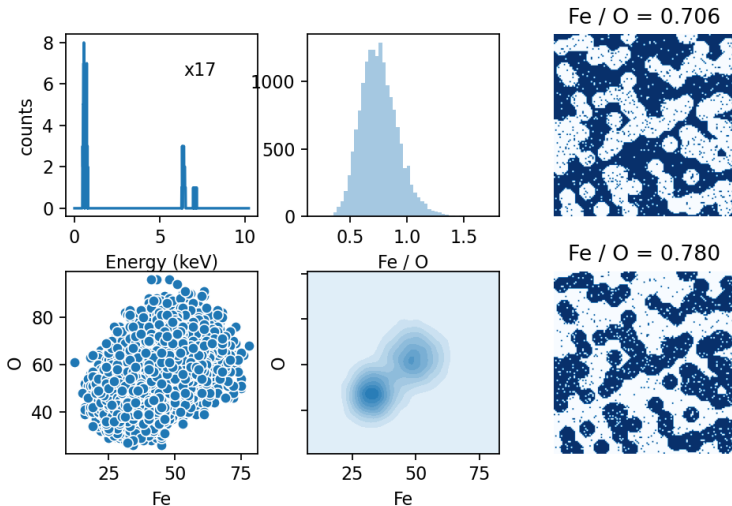
Classification by KMEANS clustering – the effect of SNR



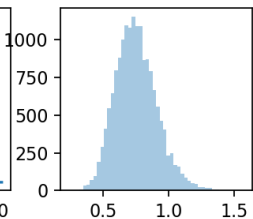
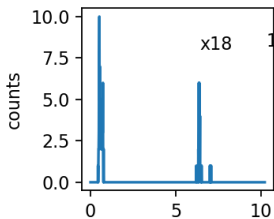
Classification by KMEANS clustering – the effect of SNR



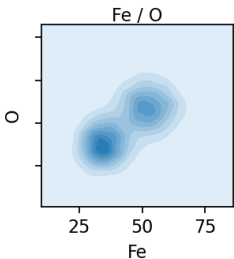
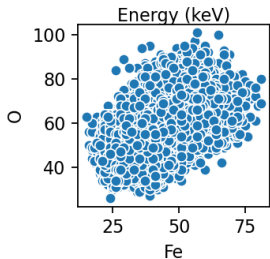
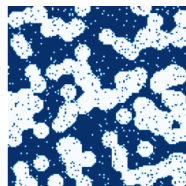
Classification by KMEANS clustering – the effect of SNR



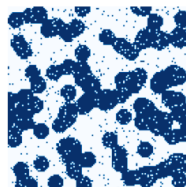
Classification by KMEANS clustering – the effect of SNR



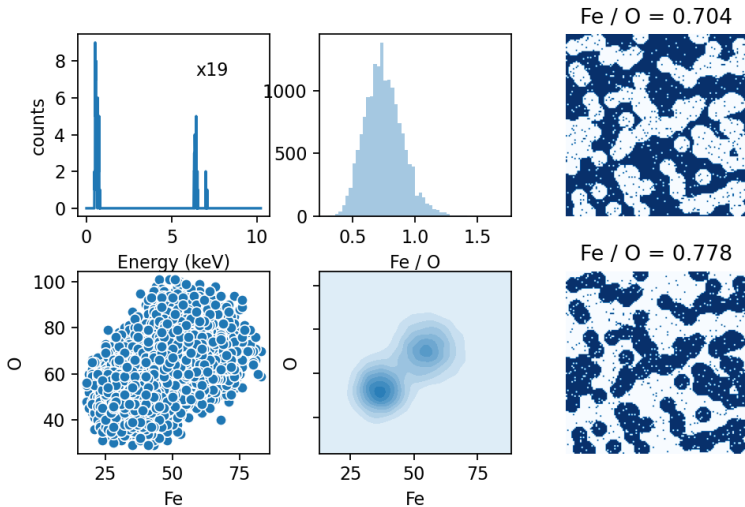
Fe / O = 0.779



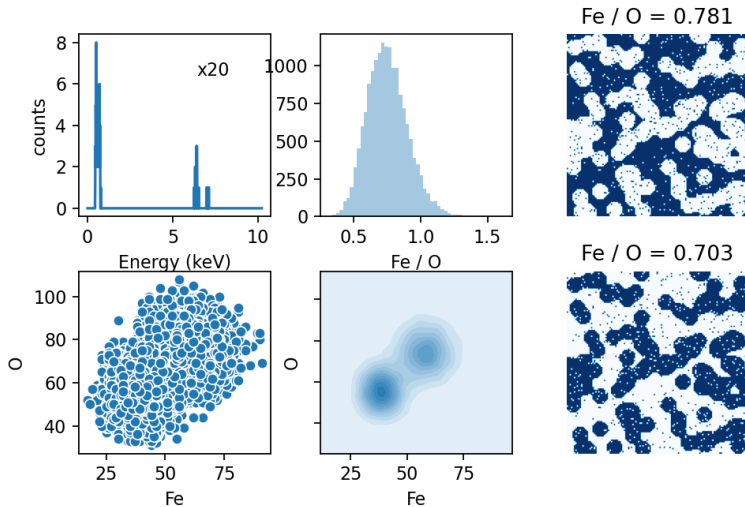
Fe / O = 0.703



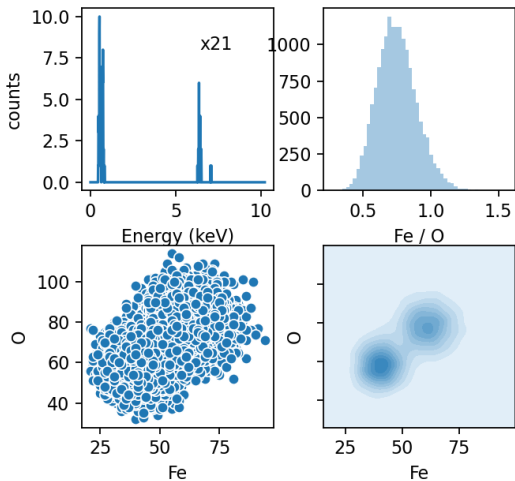
Classification by KMEANS clustering – the effect of SNR



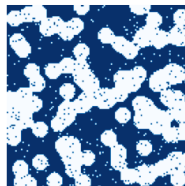
Classification by KMEANS clustering – the effect of SNR



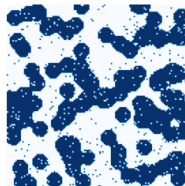
Classification by KMEANS clustering – the effect of SNR



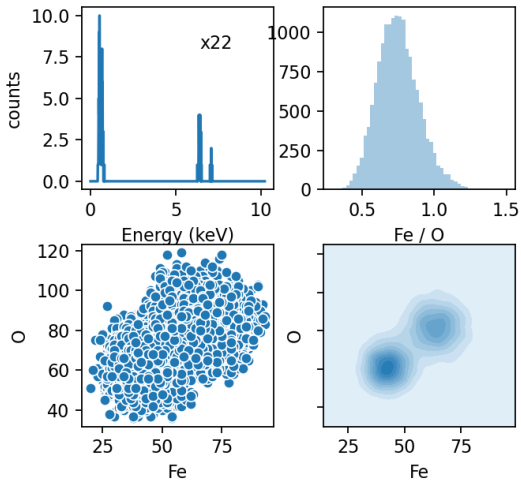
Fe / O = 0.783



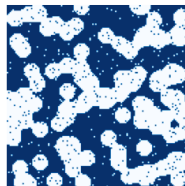
Fe / O = 0.702



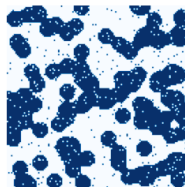
Classification by KMEANS clustering – the effect of SNR



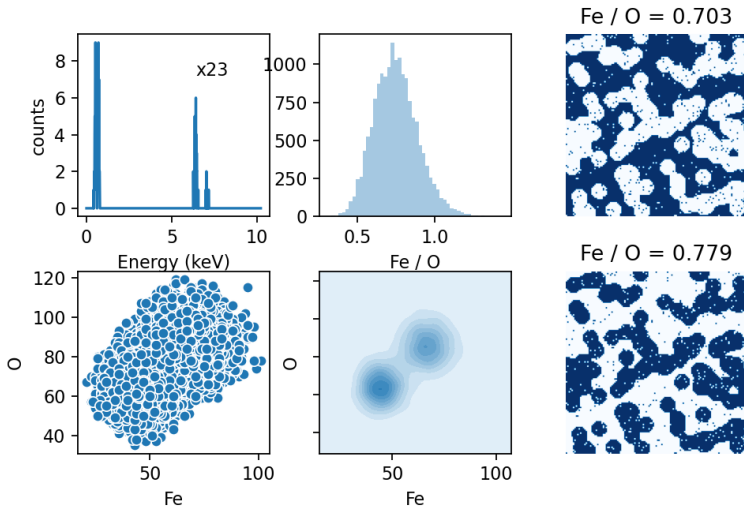
Fe / O = 0.782



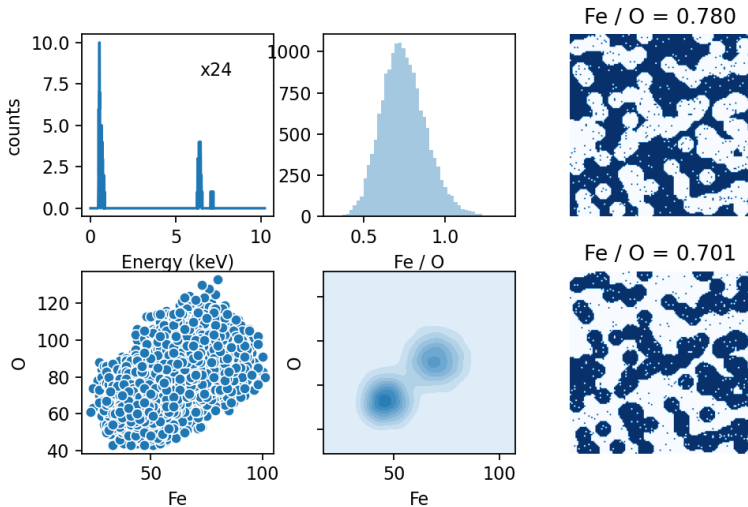
Fe / O = 0.703



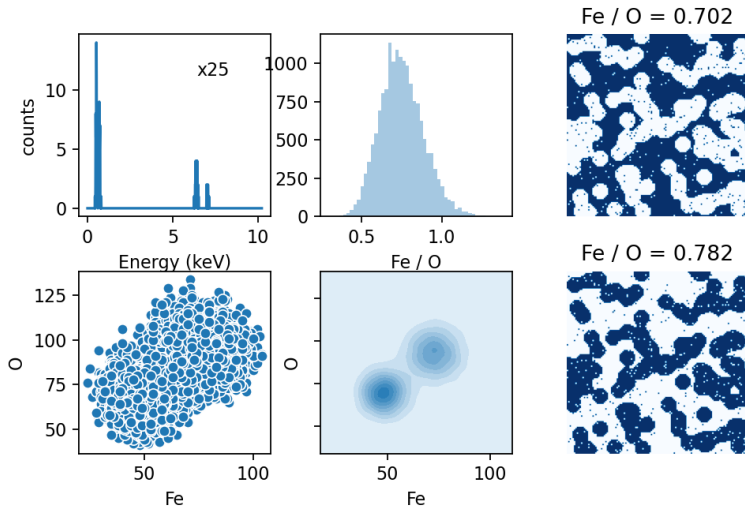
Classification by KMEANS clustering – the effect of SNR



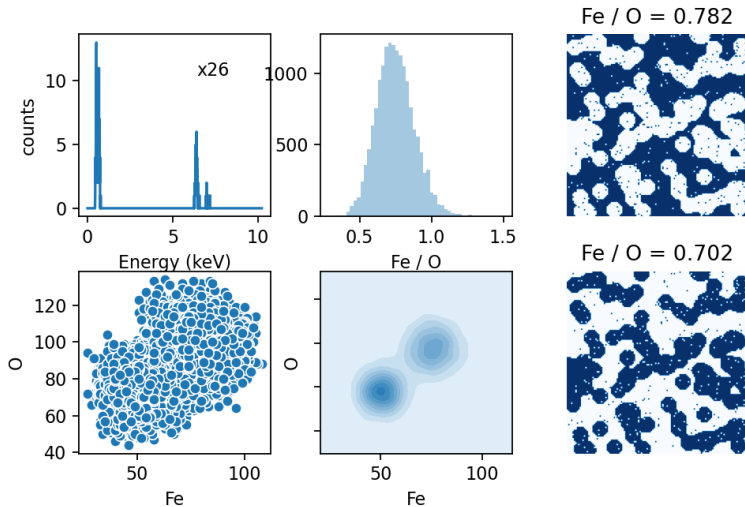
Classification by KMEANS clustering – the effect of SNR



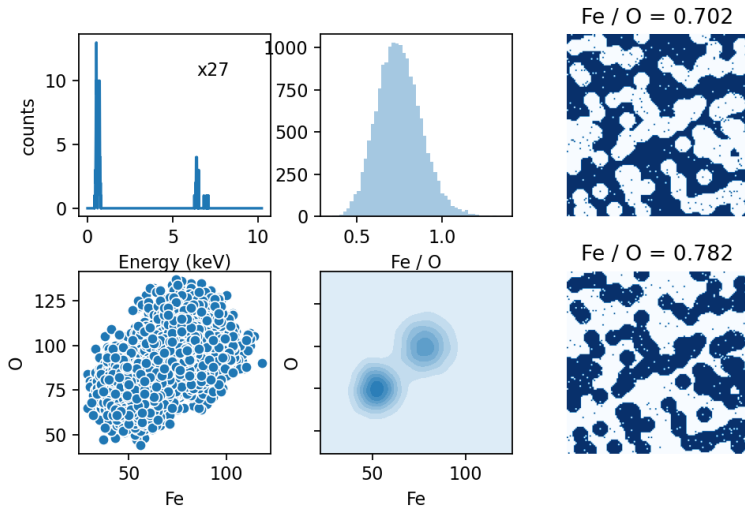
Classification by KMEANS clustering – the effect of SNR



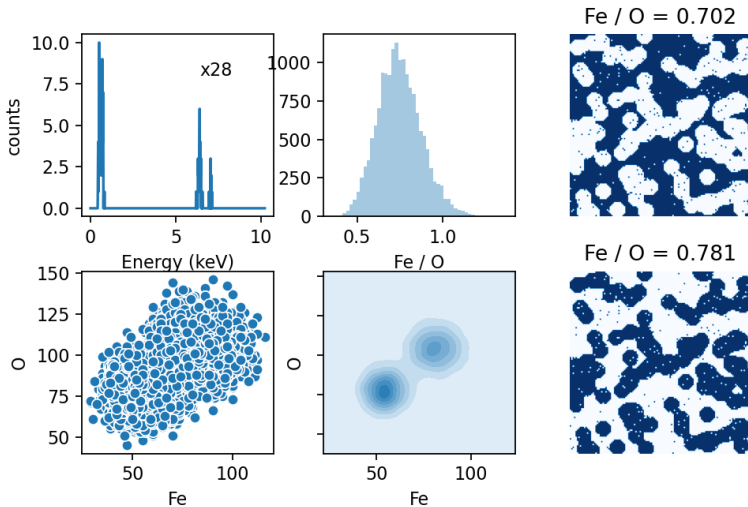
Classification by KMEANS clustering – the effect of SNR



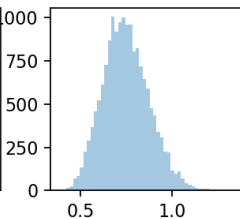
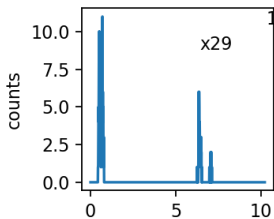
Classification by KMEANS clustering – the effect of SNR



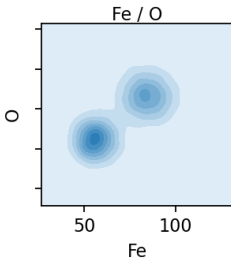
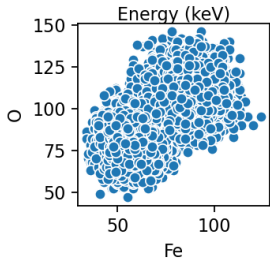
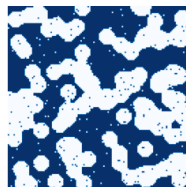
Classification by KMEANS clustering – the effect of SNR



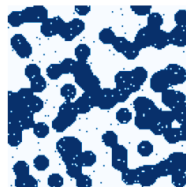
Classification by KMEANS clustering – the effect of SNR



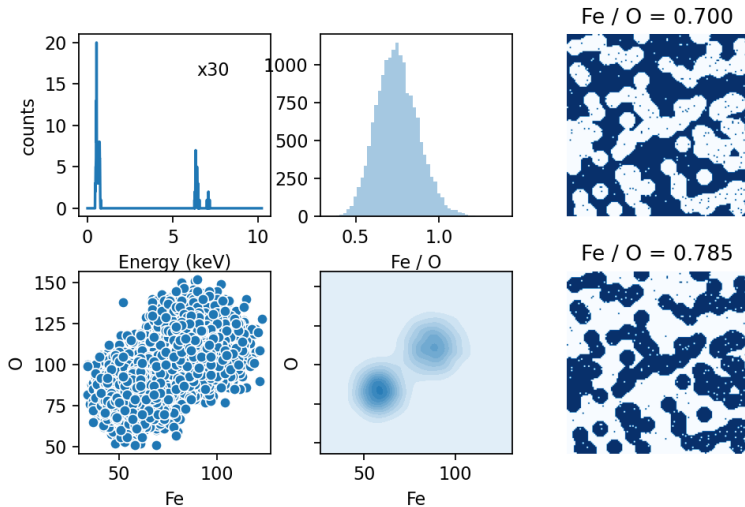
Fe / O = 0.783



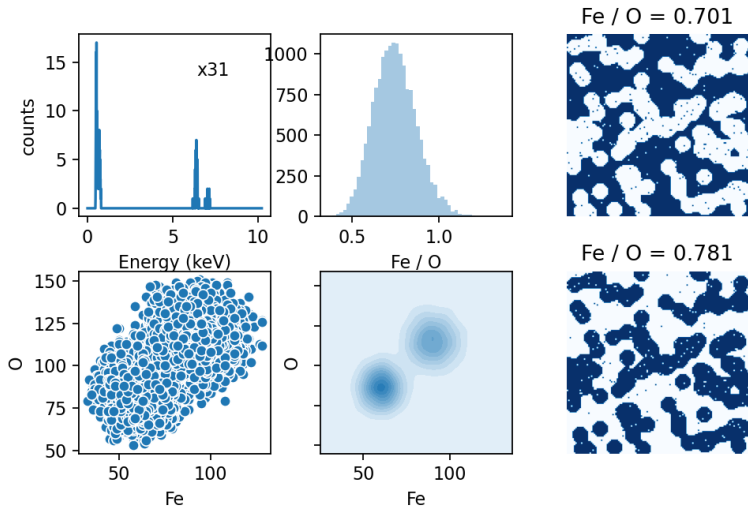
Fe / O = 0.702



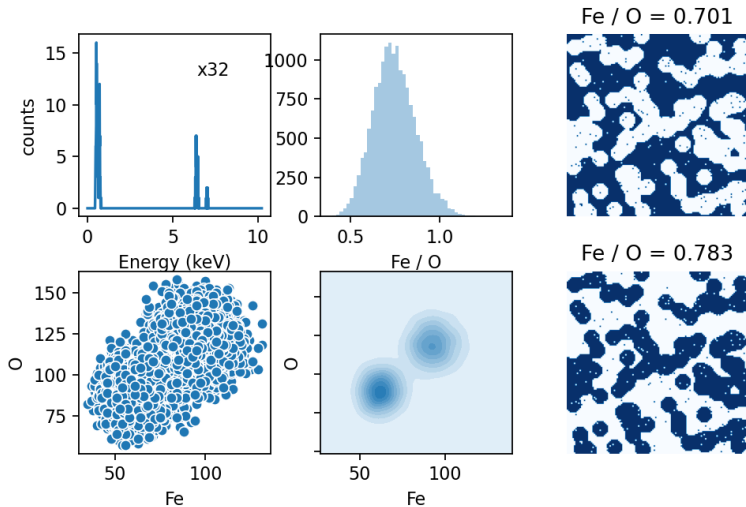
Classification by KMEANS clustering – the effect of SNR



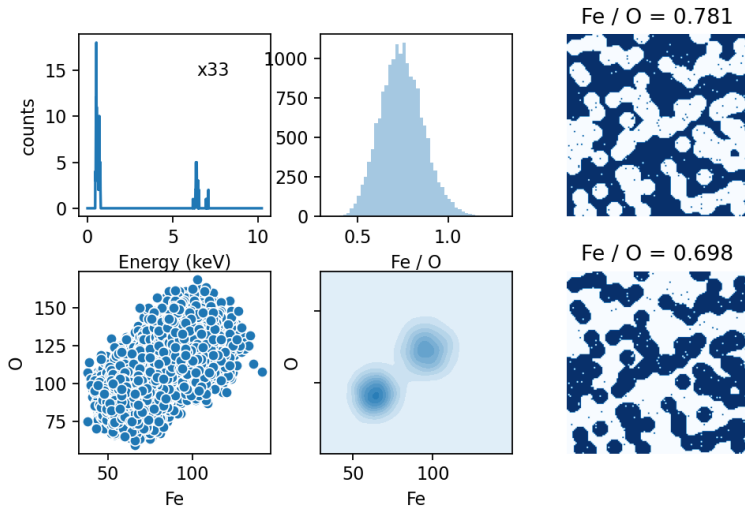
Classification by KMEANS clustering – the effect of SNR



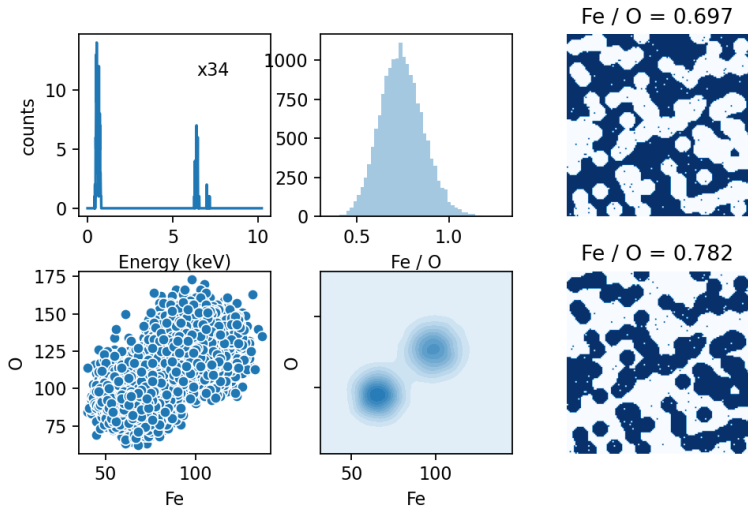
Classification by KMEANS clustering – the effect of SNR



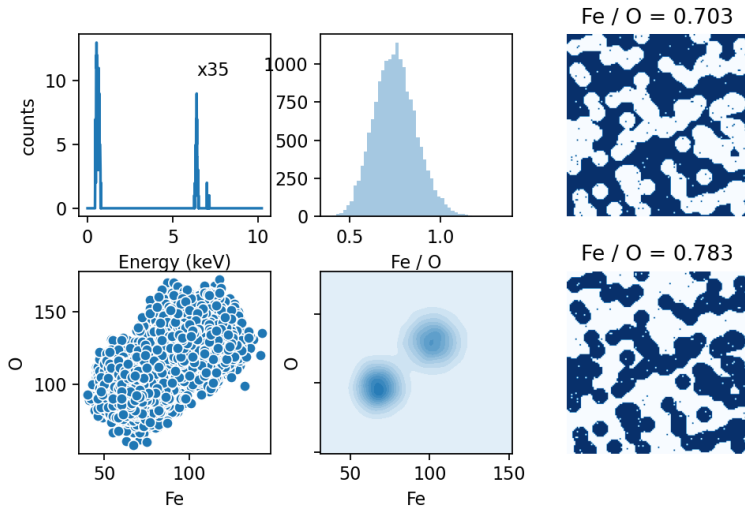
Classification by KMEANS clustering – the effect of SNR



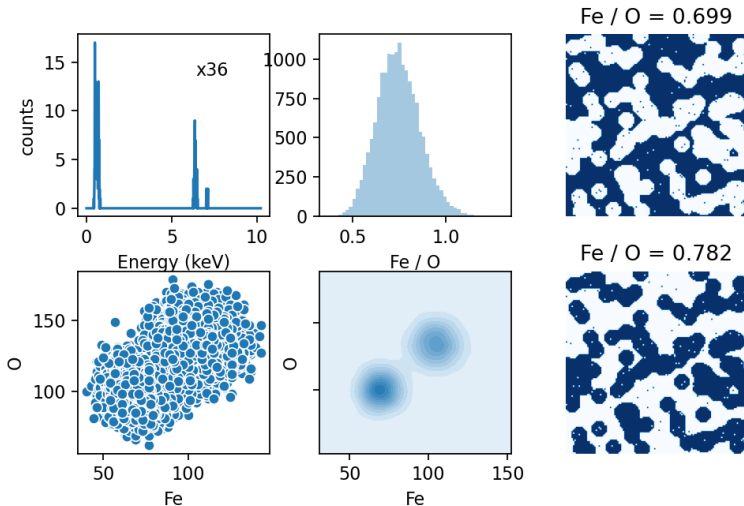
Classification by KMEANS clustering – the effect of SNR



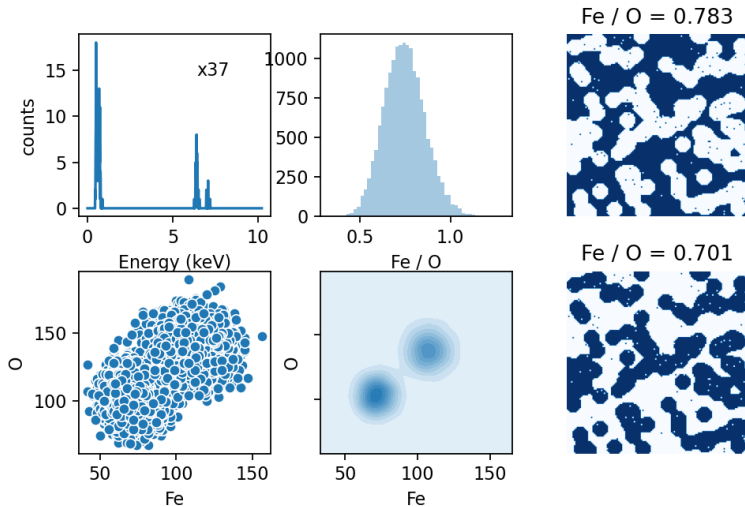
Classification by KMEANS clustering – the effect of SNR



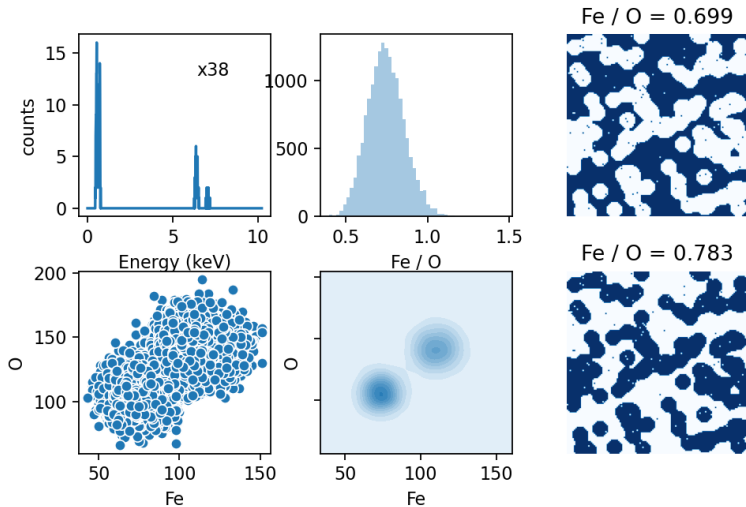
Classification by KMEANS clustering – the effect of SNR



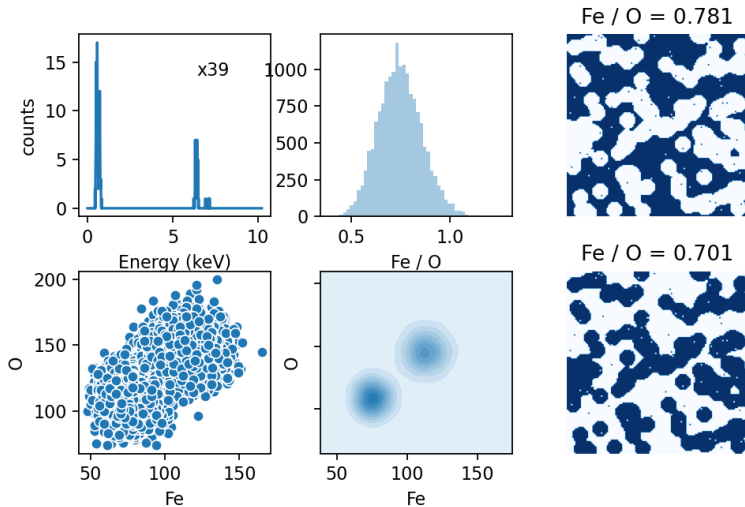
Classification by KMEANS clustering – the effect of SNR



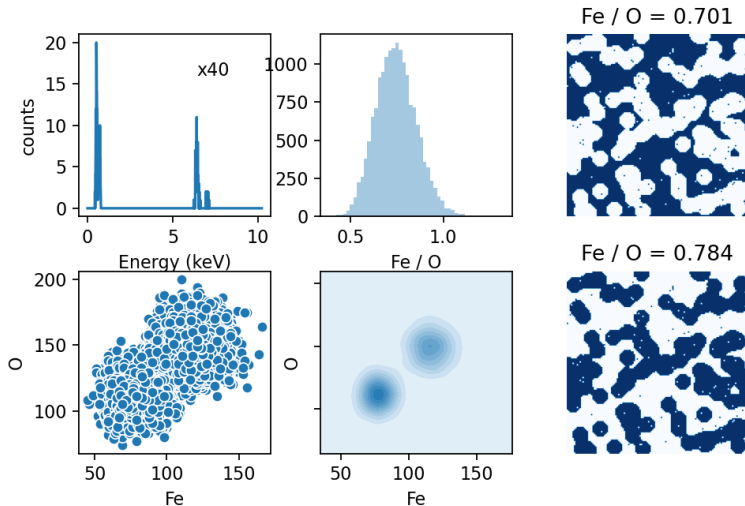
Classification by KMEANS clustering – the effect of SNR



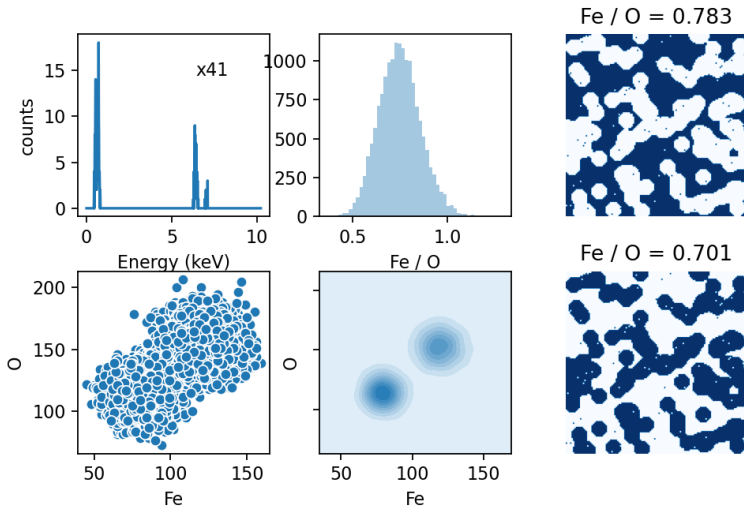
Classification by KMEANS clustering – the effect of SNR



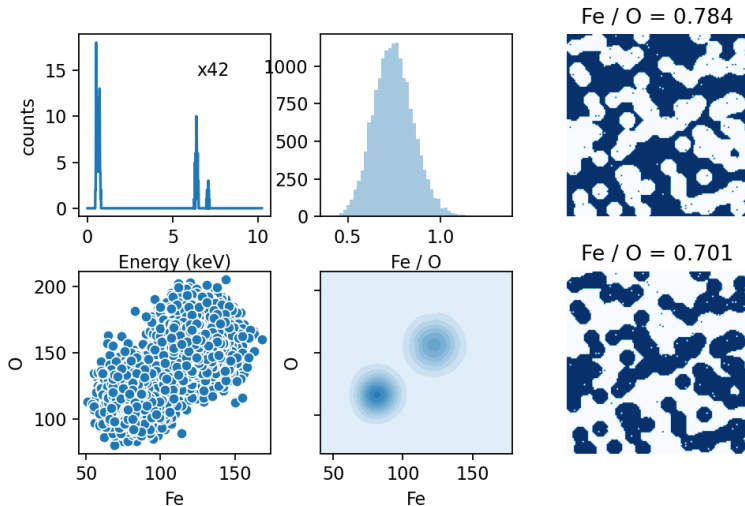
Classification by KMEANS clustering – the effect of SNR



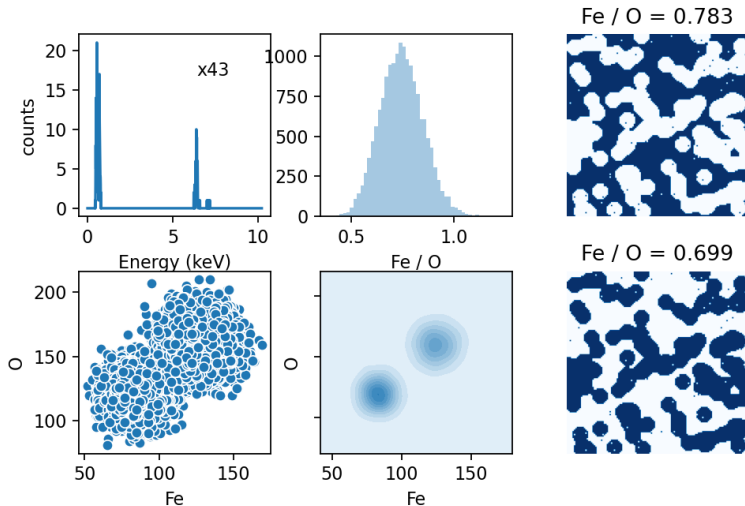
Classification by KMEANS clustering – the effect of SNR



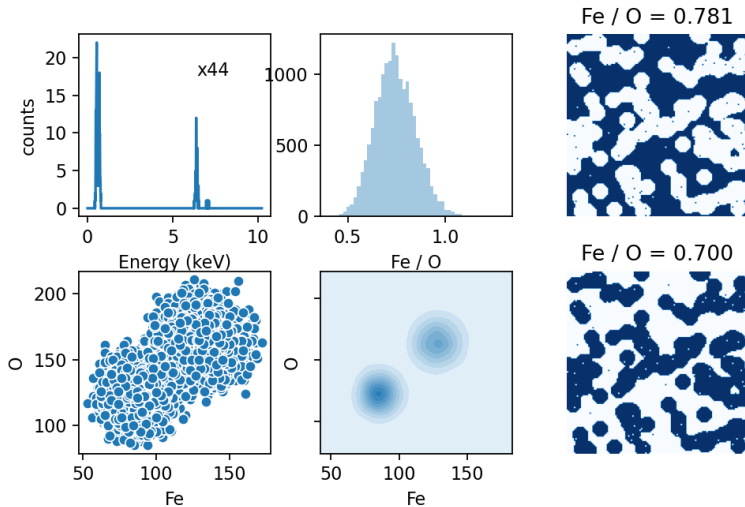
Classification by KMEANS clustering – the effect of SNR



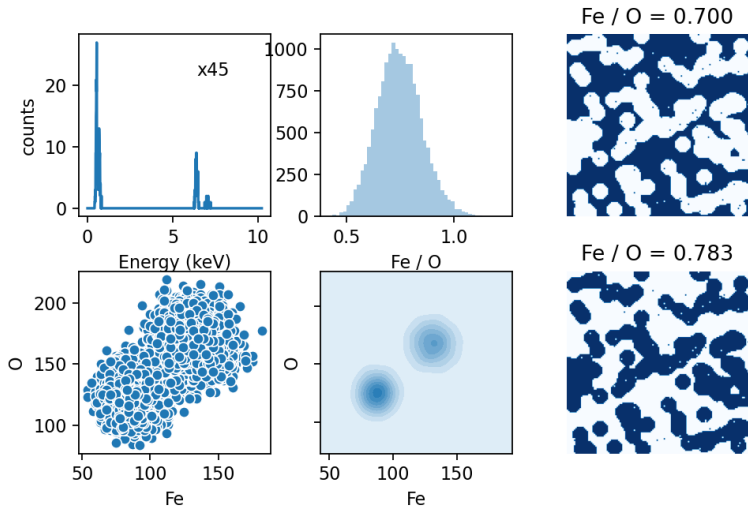
Classification by KMEANS clustering – the effect of SNR



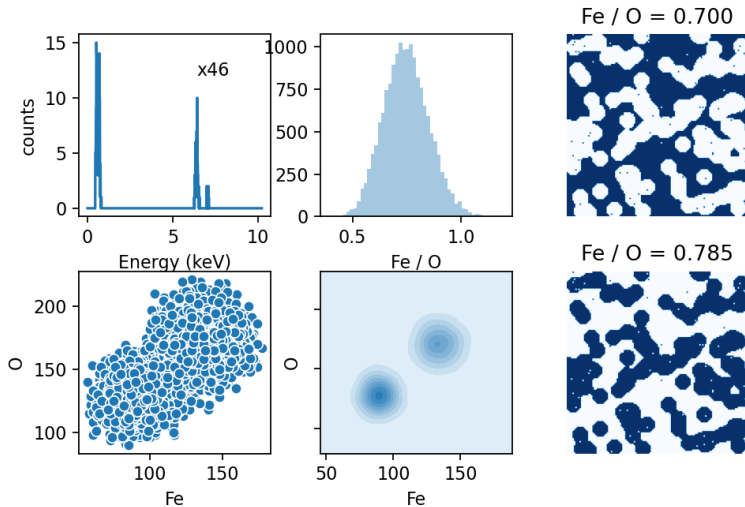
Classification by KMEANS clustering – the effect of SNR



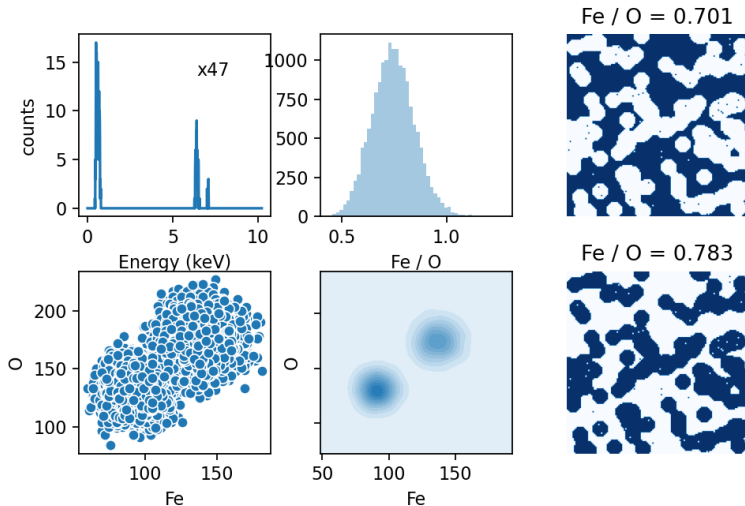
Classification by KMEANS clustering – the effect of SNR



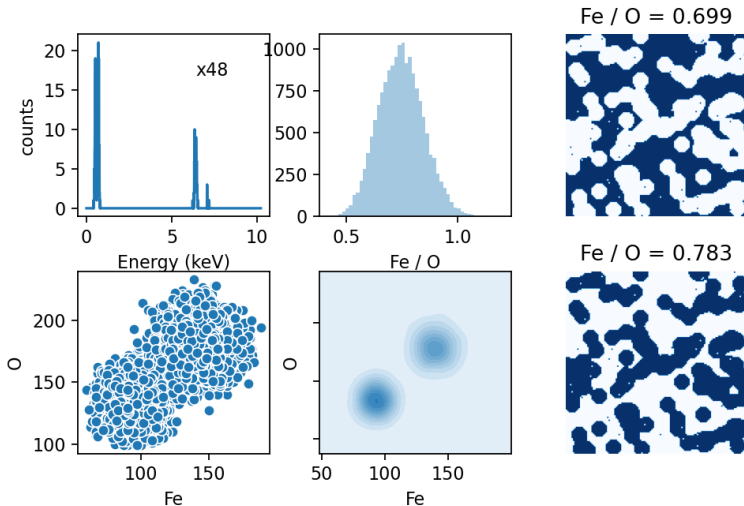
Classification by KMEANS clustering – the effect of SNR



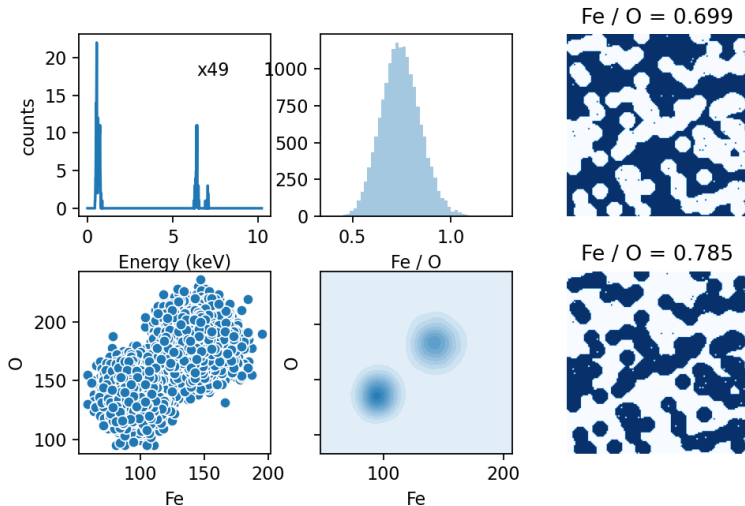
Classification by KMEANS clustering – the effect of SNR



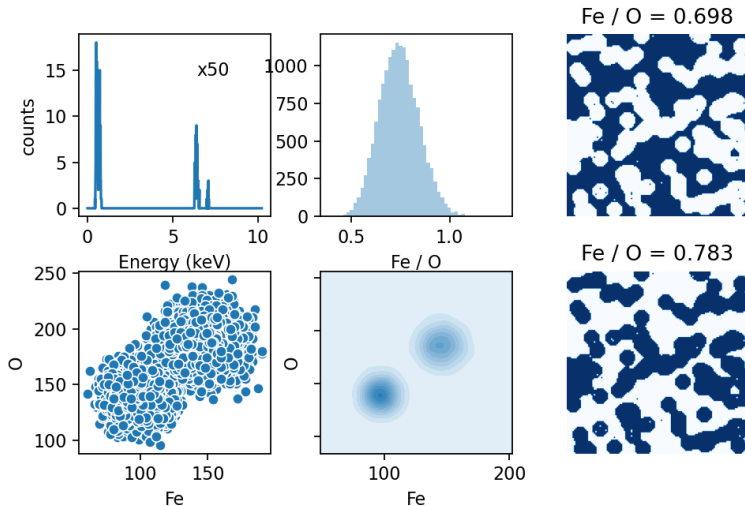
Classification by KMEANS clustering – the effect of SNR



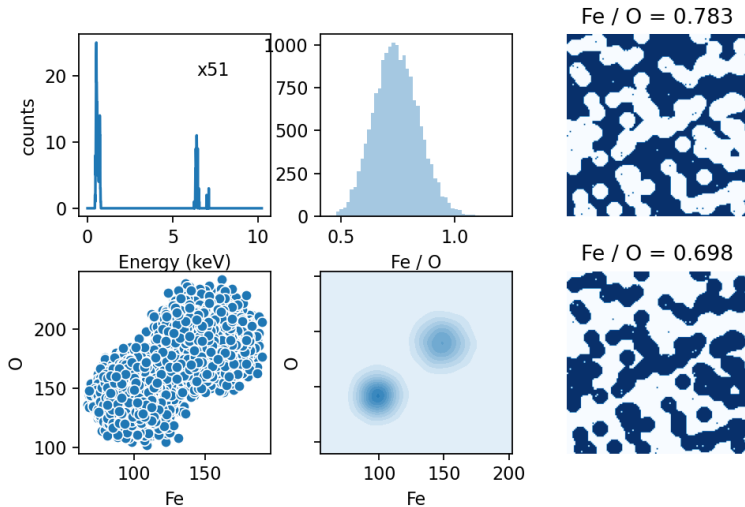
Classification by KMEANS clustering – the effect of SNR



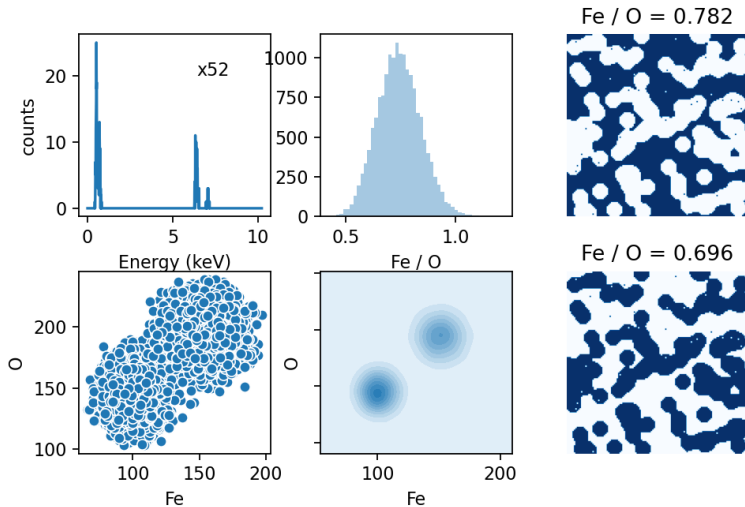
Classification by KMEANS clustering – the effect of SNR



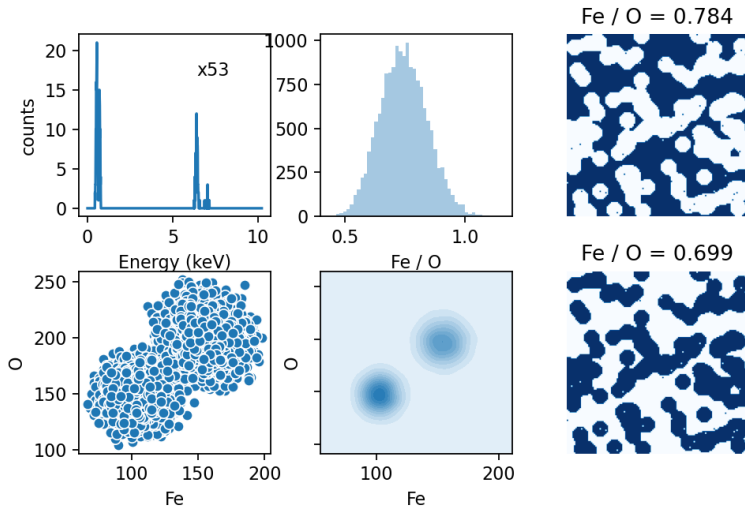
Classification by KMEANS clustering – the effect of SNR



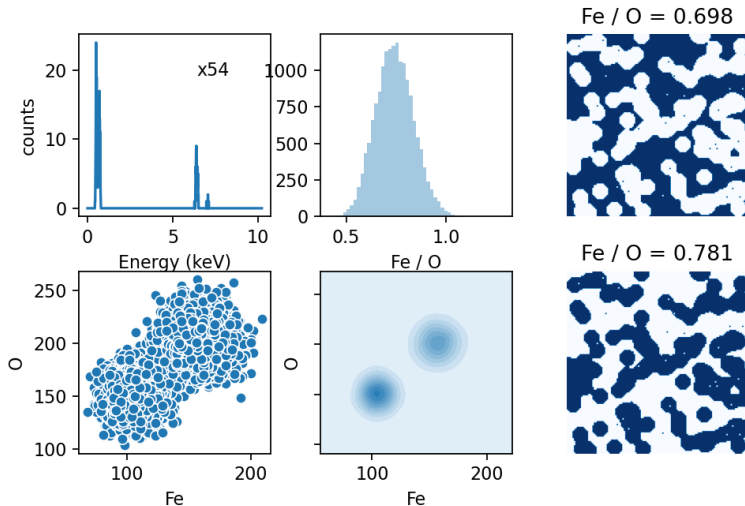
Classification by KMEANS clustering – the effect of SNR



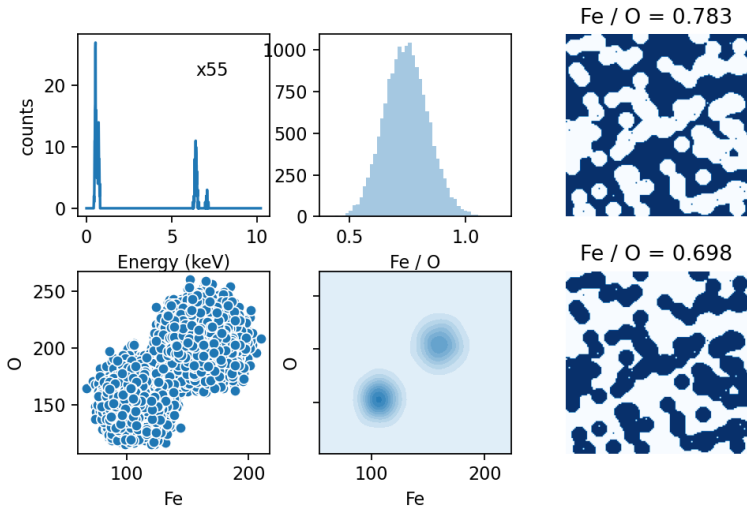
Classification by KMEANS clustering – the effect of SNR



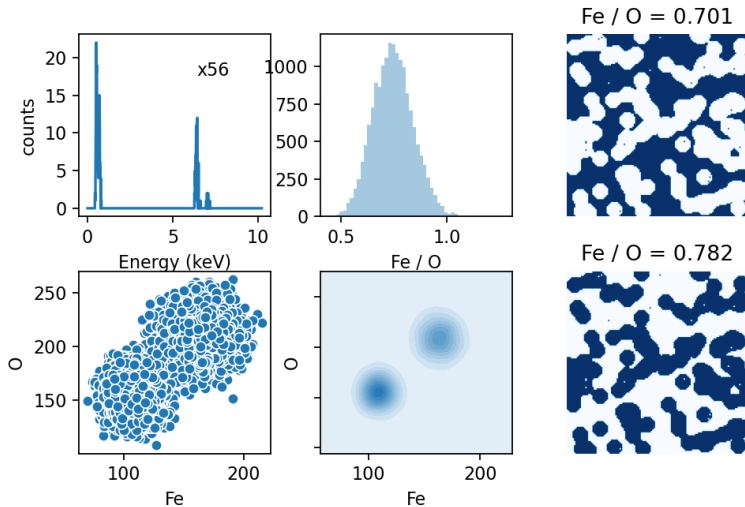
Classification by KMEANS clustering – the effect of SNR



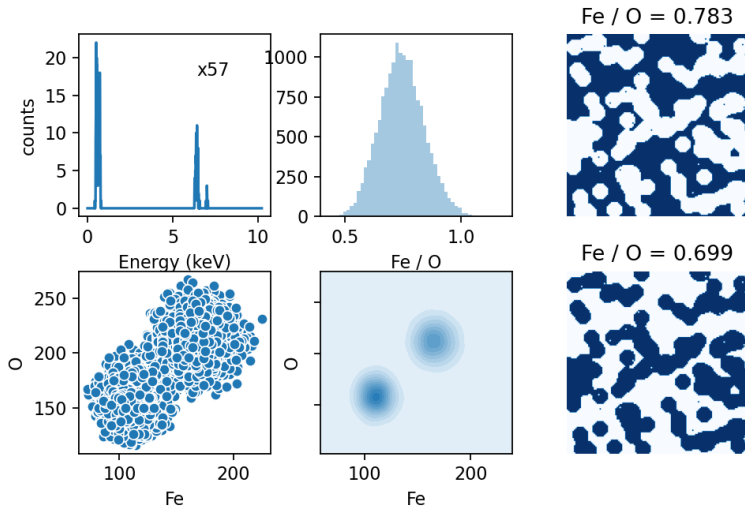
Classification by KMEANS clustering – the effect of SNR



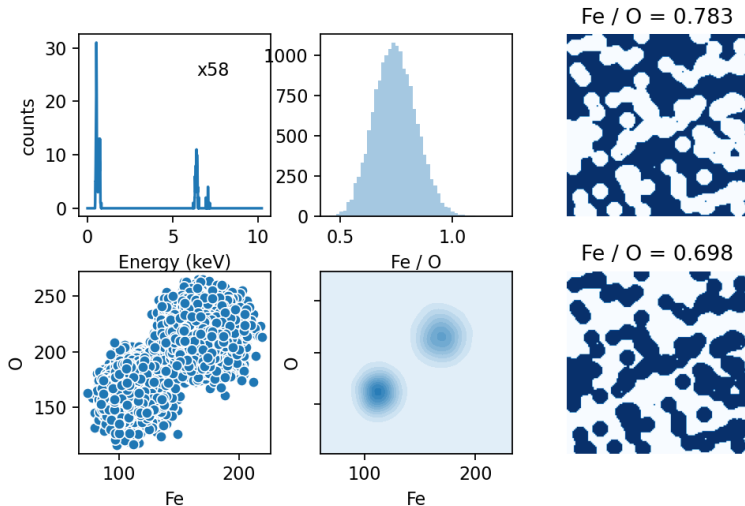
Classification by KMEANS clustering – the effect of SNR



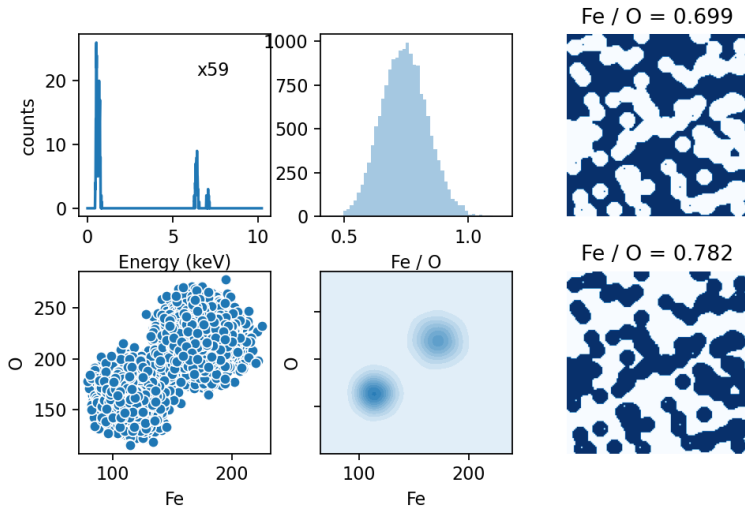
Classification by KMEANS clustering – the effect of SNR



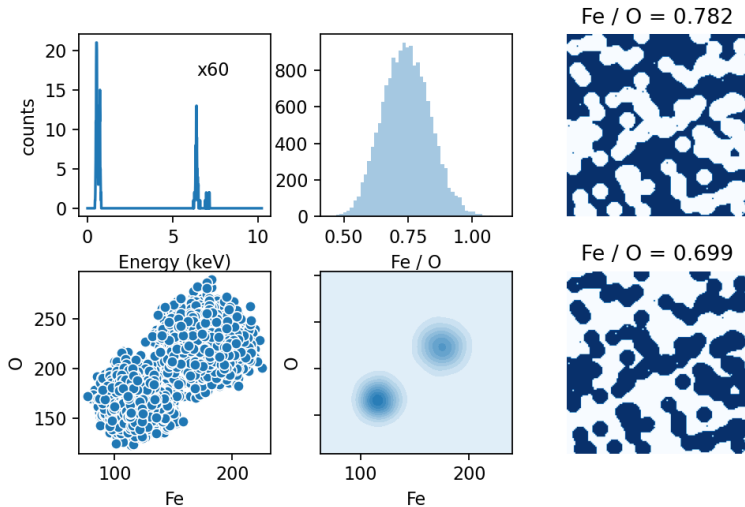
Classification by KMEANS clustering – the effect of SNR



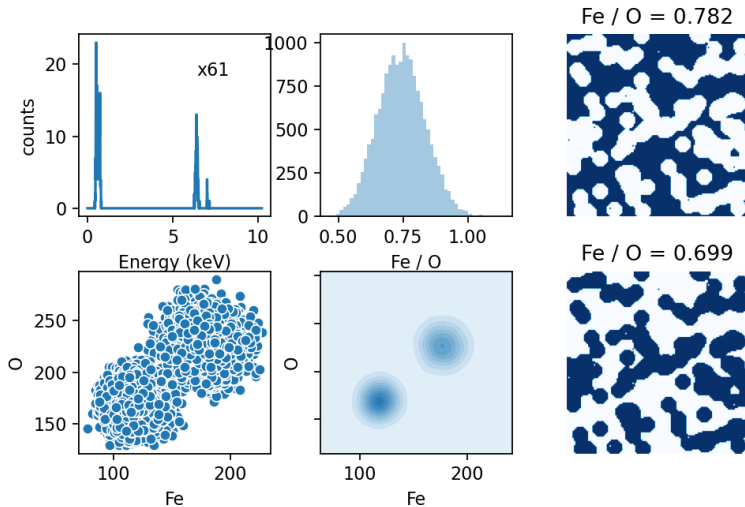
Classification by KMEANS clustering – the effect of SNR



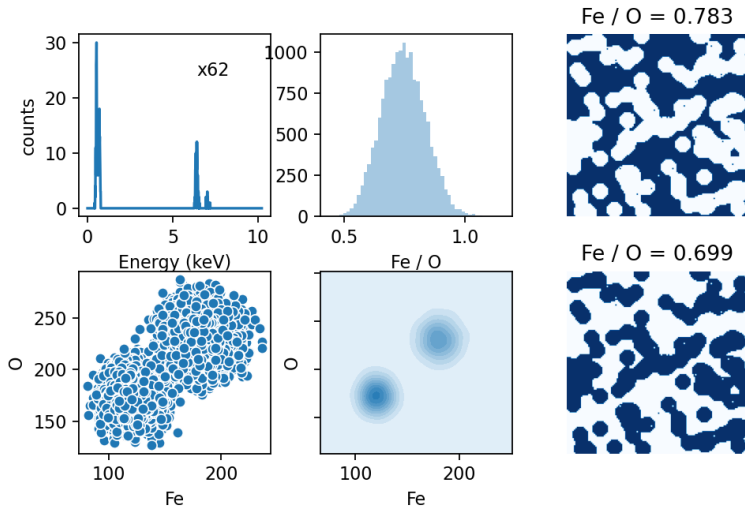
Classification by KMEANS clustering – the effect of SNR



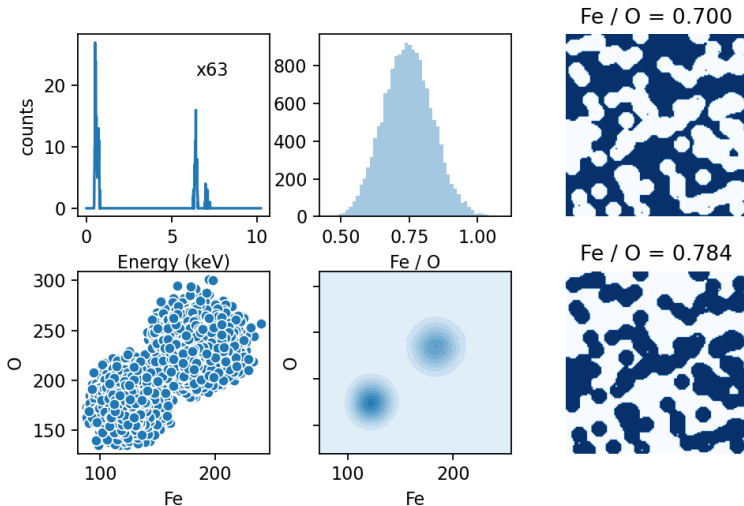
Classification by KMEANS clustering – the effect of SNR



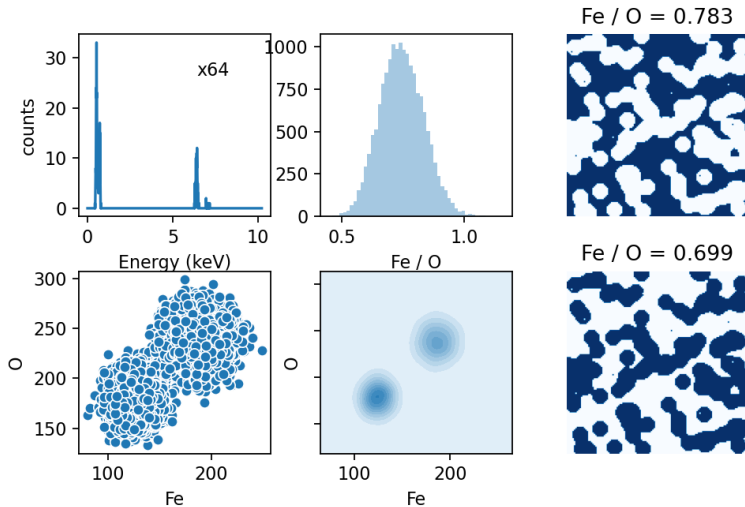
Classification by KMEANS clustering – the effect of SNR



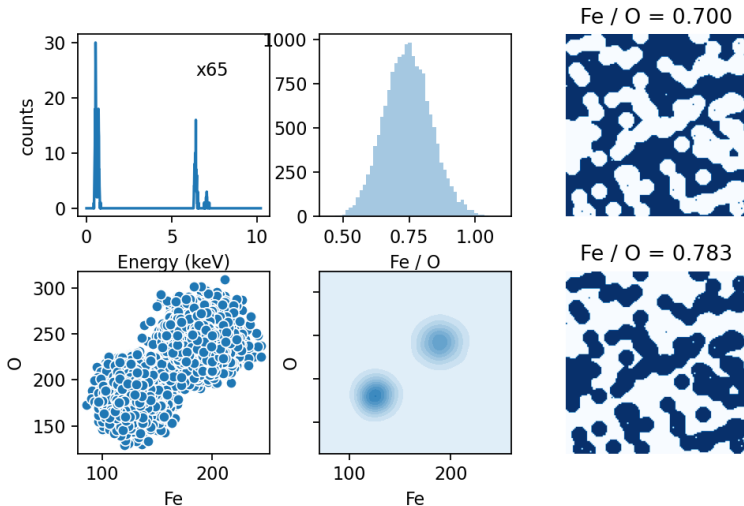
Classification by KMEANS clustering – the effect of SNR



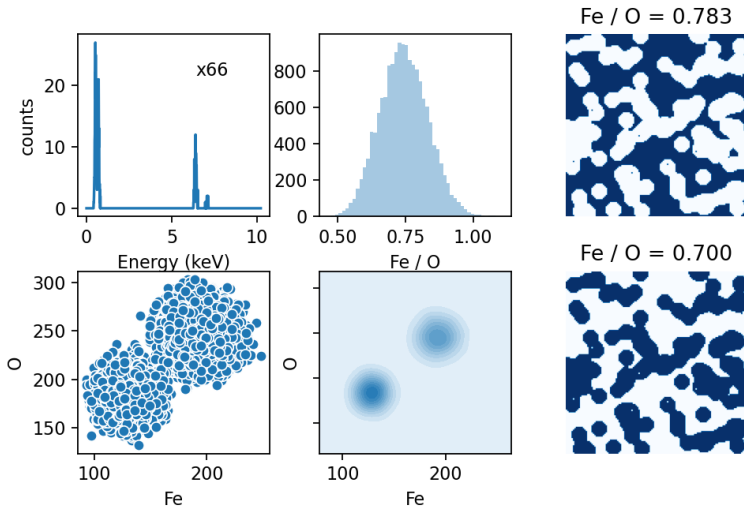
Classification by KMEANS clustering – the effect of SNR



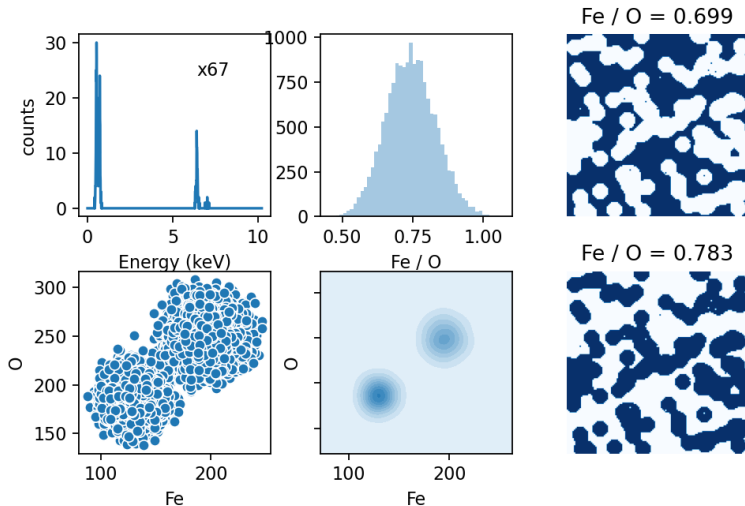
Classification by KMEANS clustering – the effect of SNR



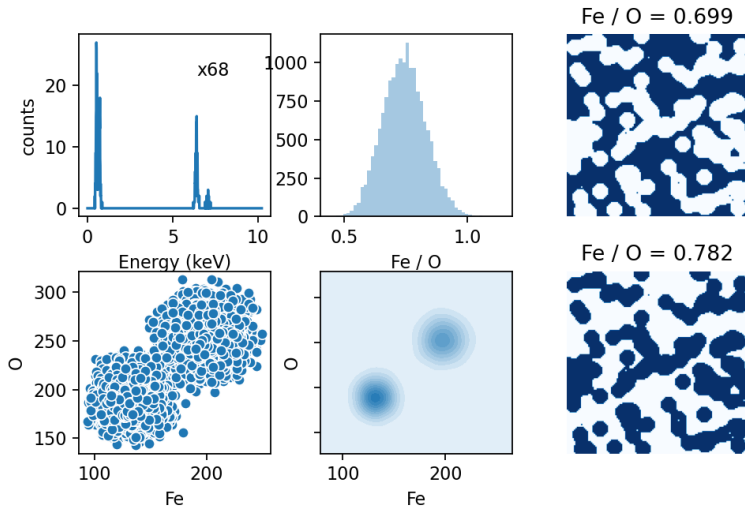
Classification by KMEANS clustering – the effect of SNR



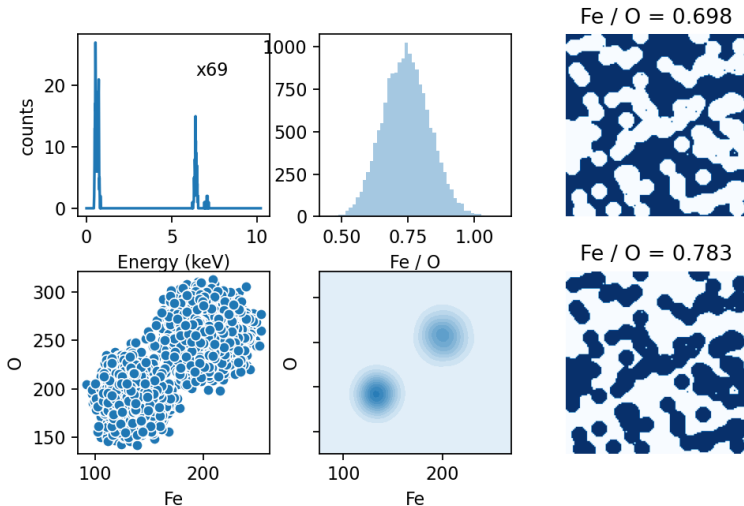
Classification by KMEANS clustering – the effect of SNR



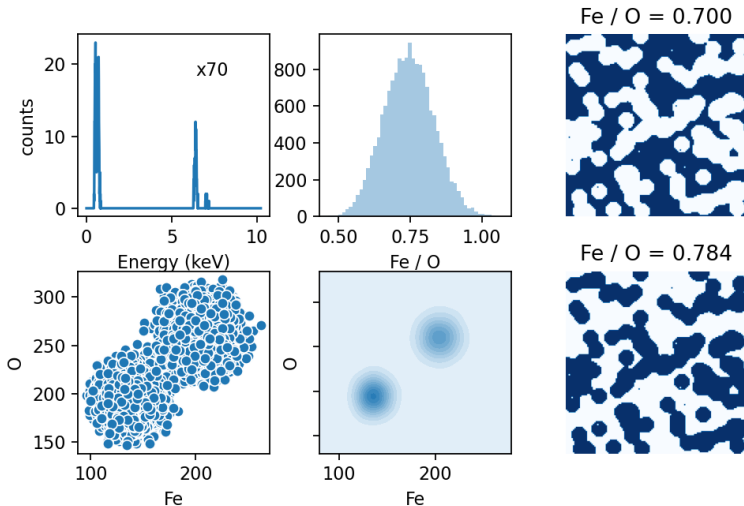
Classification by KMEANS clustering – the effect of SNR



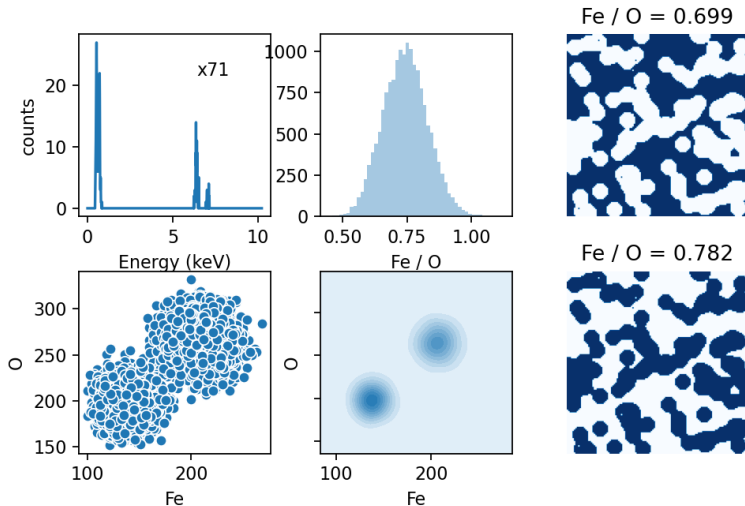
Classification by KMEANS clustering – the effect of SNR



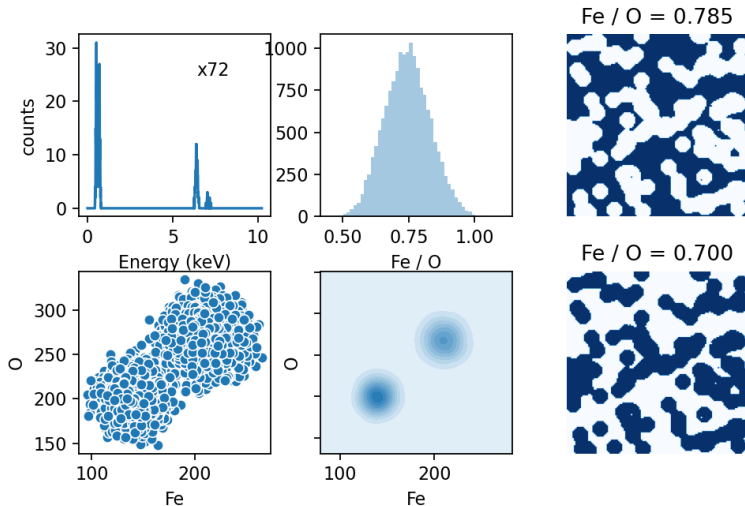
Classification by KMEANS clustering – the effect of SNR



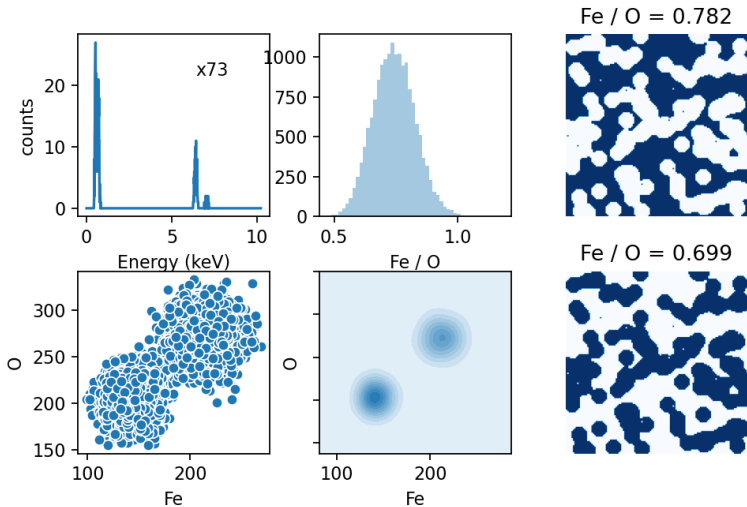
Classification by KMEANS clustering – the effect of SNR



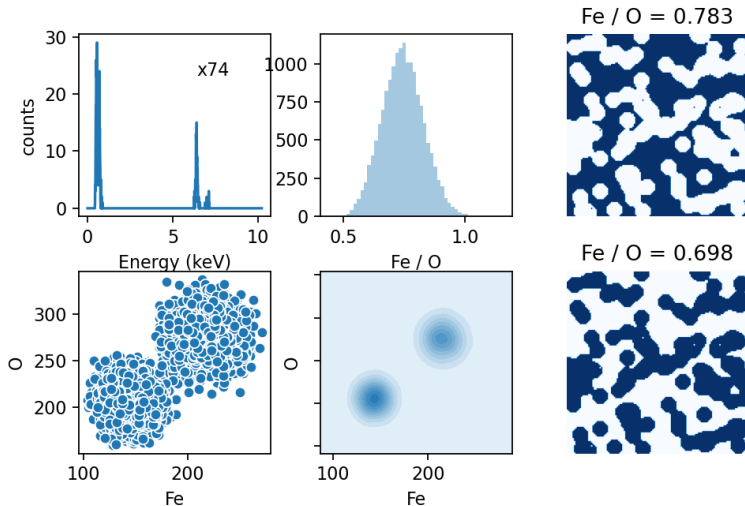
Classification by KMEANS clustering – the effect of SNR



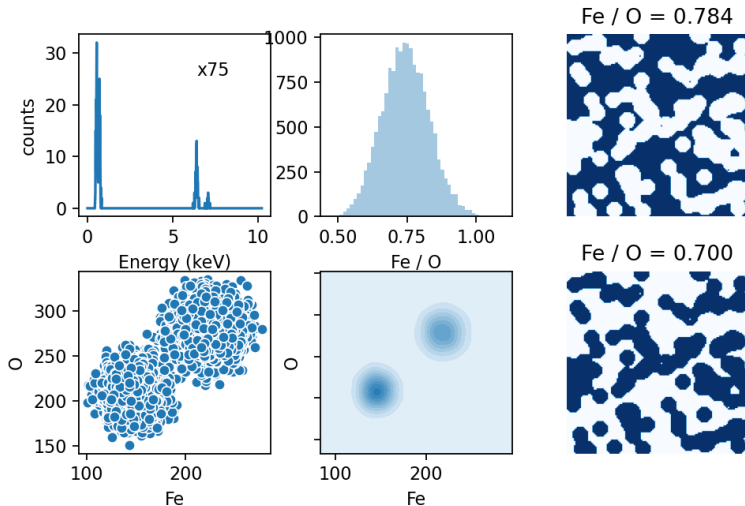
Classification by KMEANS clustering – the effect of SNR



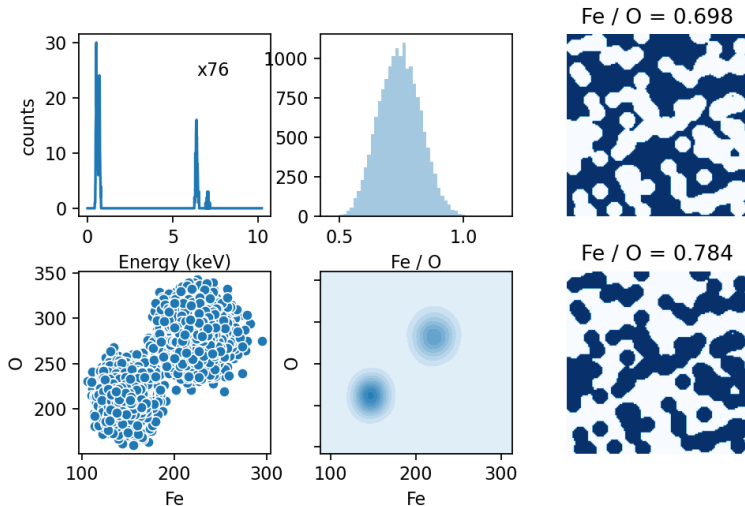
Classification by KMEANS clustering – the effect of SNR



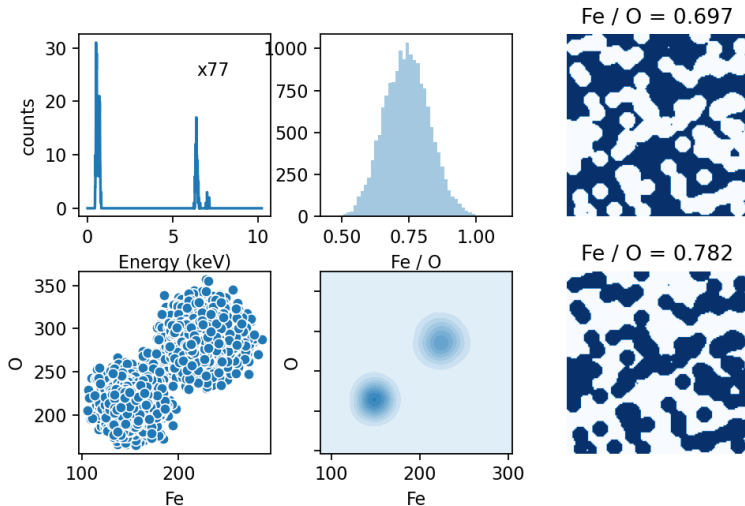
Classification by KMEANS clustering – the effect of SNR



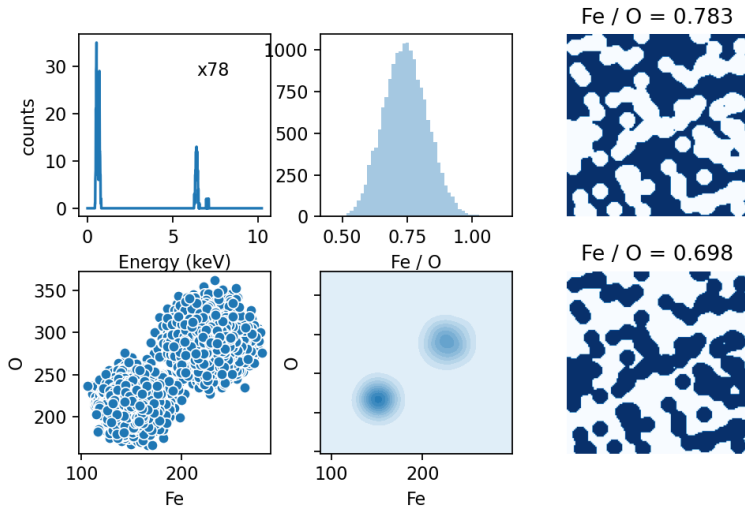
Classification by KMEANS clustering – the effect of SNR



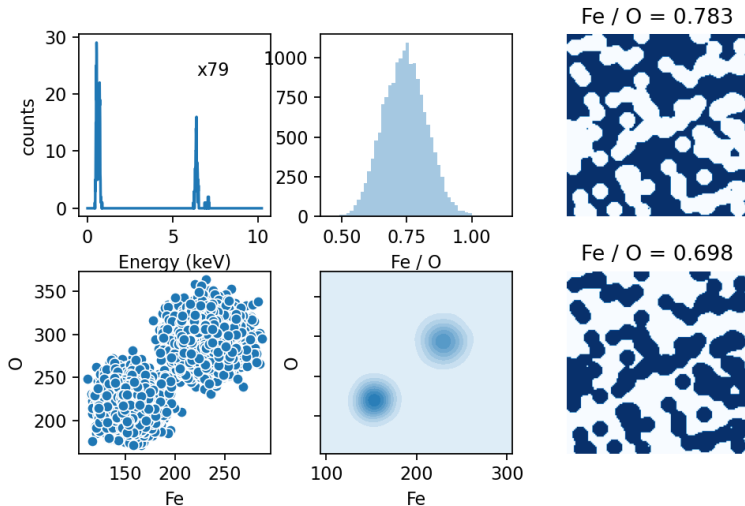
Classification by KMEANS clustering – the effect of SNR



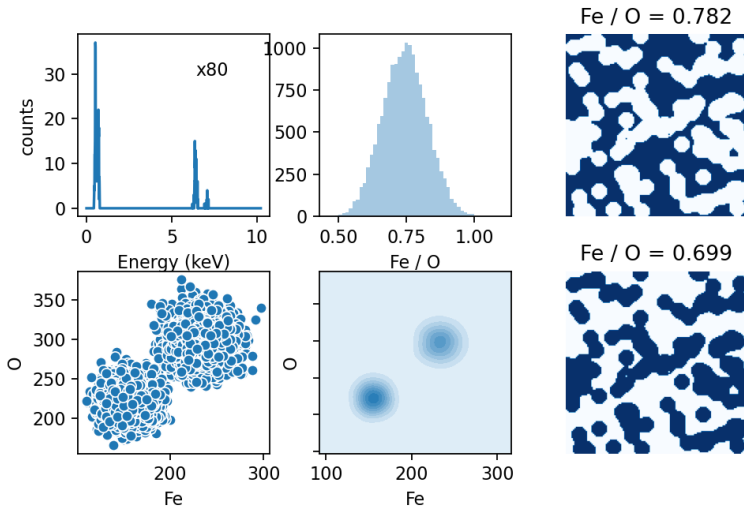
Classification by KMEANS clustering – the effect of SNR



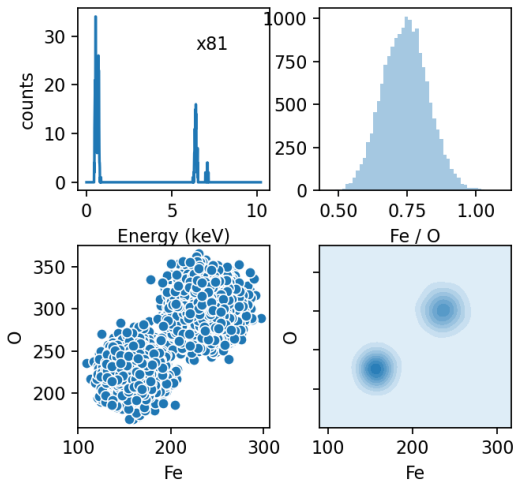
Classification by KMEANS clustering – the effect of SNR



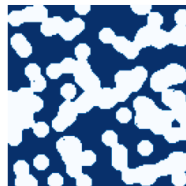
Classification by KMEANS clustering – the effect of SNR



Classification by KMEANS clustering – the effect of SNR



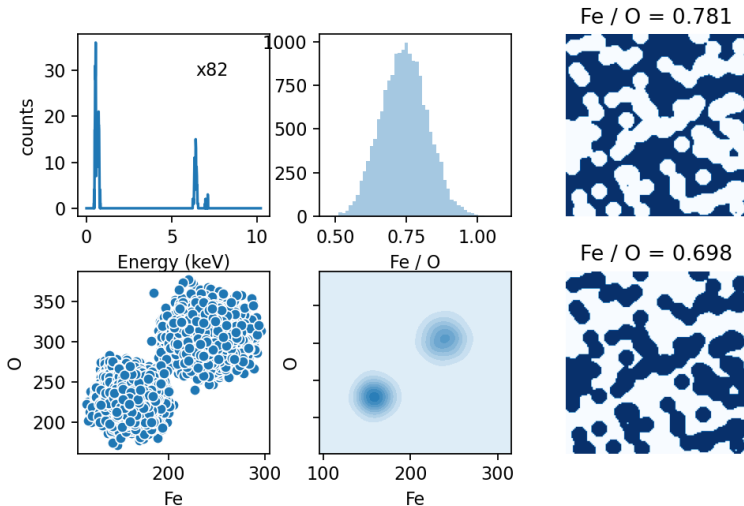
Fe / O = 0.699



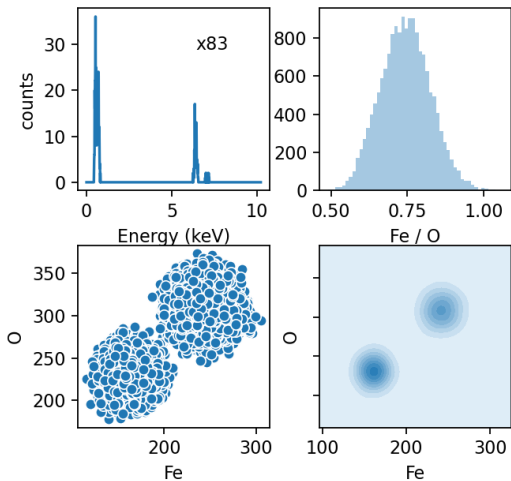
Fe / O = 0.783



Classification by KMEANS clustering – the effect of SNR



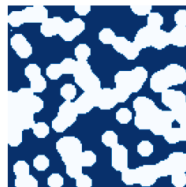
Classification by KMEANS clustering – the effect of SNR



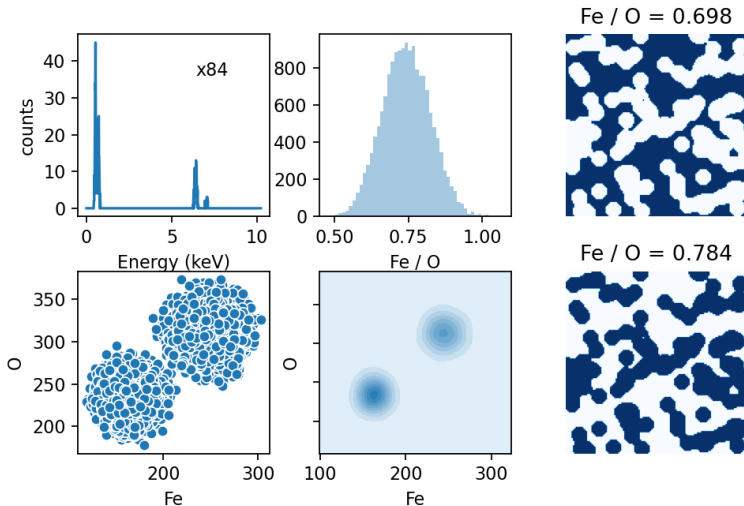
Fe / O = 0.699



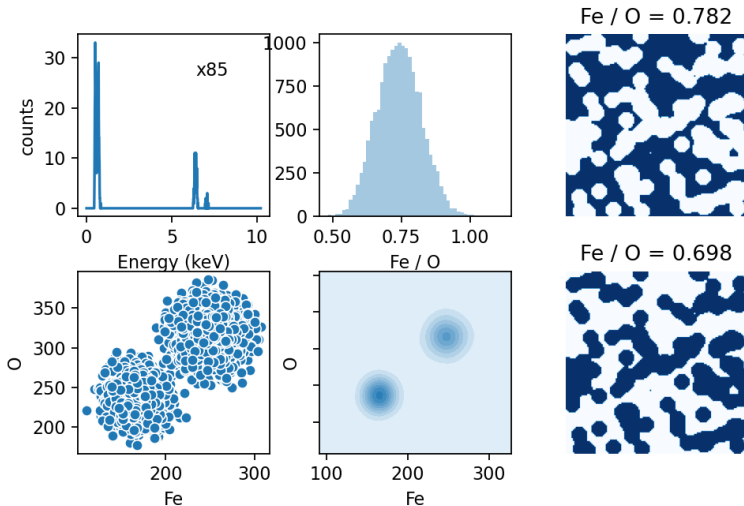
Fe / O = 0.784



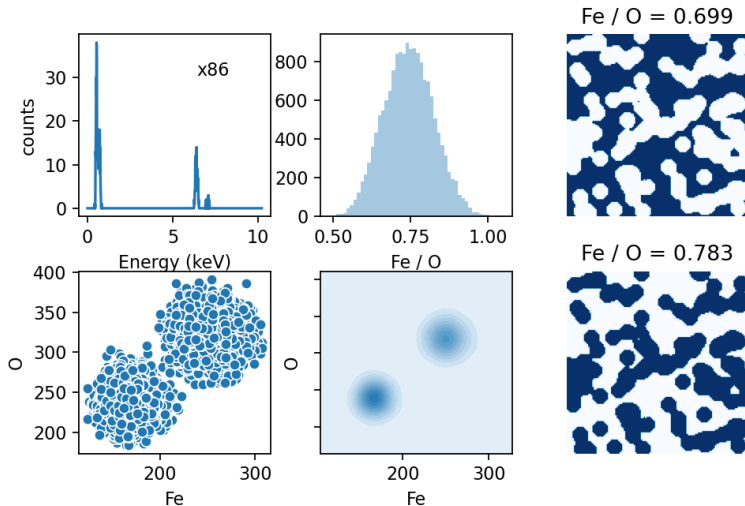
Classification by KMEANS clustering – the effect of SNR



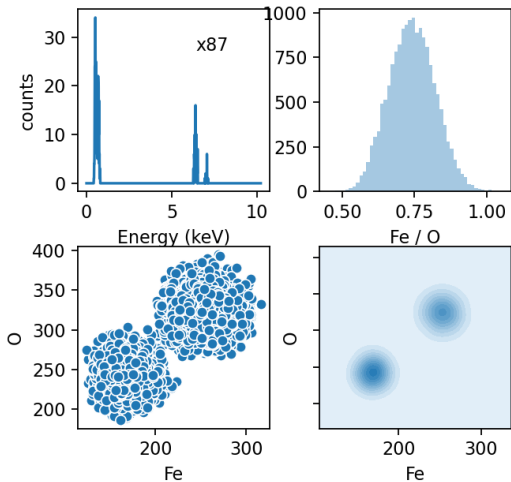
Classification by KMEANS clustering – the effect of SNR



Classification by KMEANS clustering – the effect of SNR



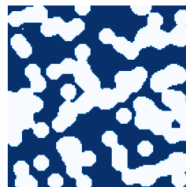
Classification by KMEANS clustering – the effect of SNR



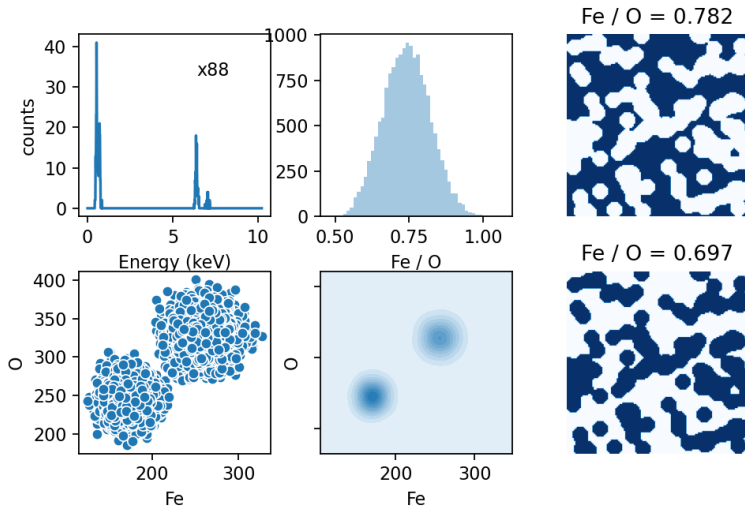
Fe / O = 0.781



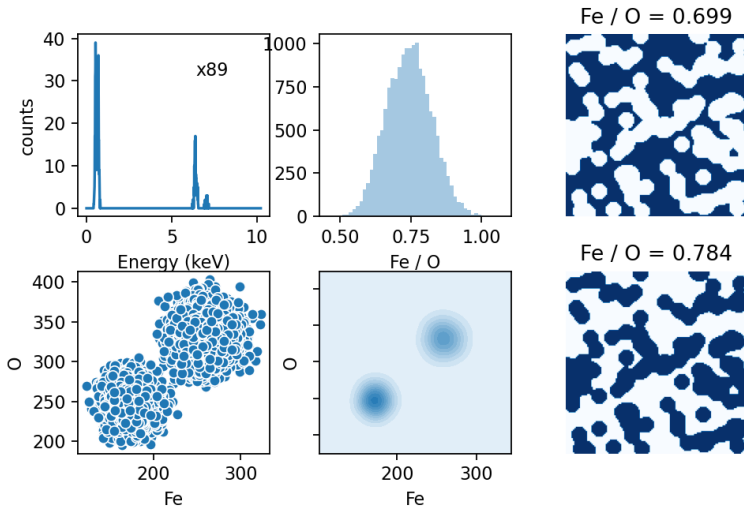
Fe / O = 0.698



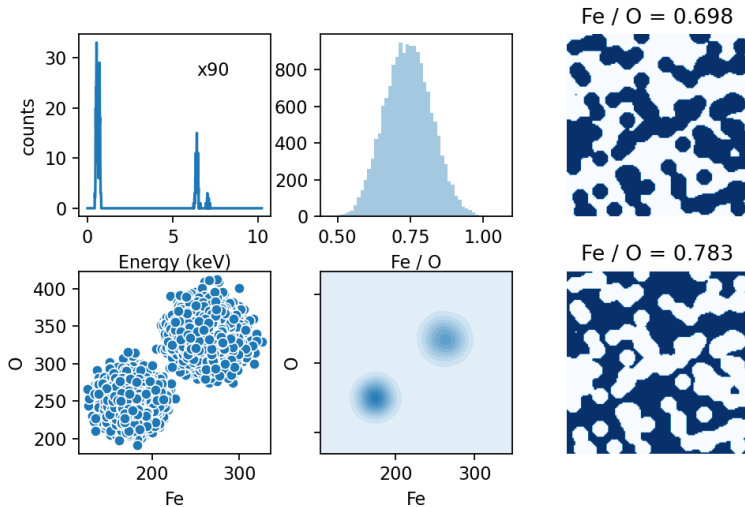
Classification by KMEANS clustering – the effect of SNR



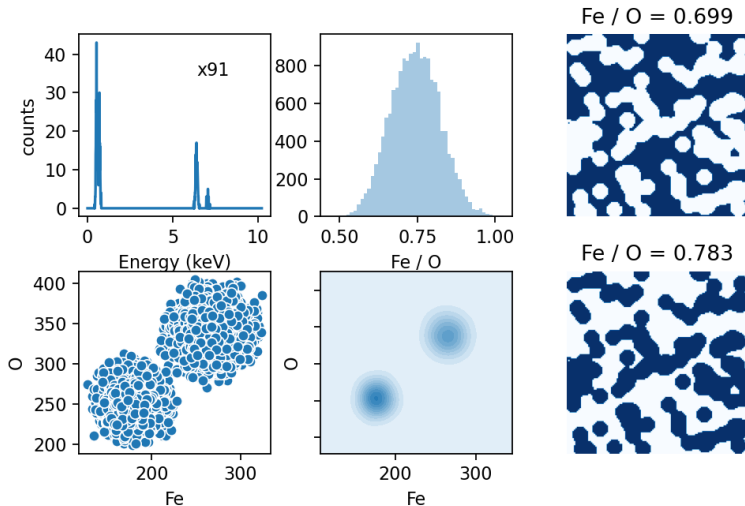
Classification by KMEANS clustering – the effect of SNR



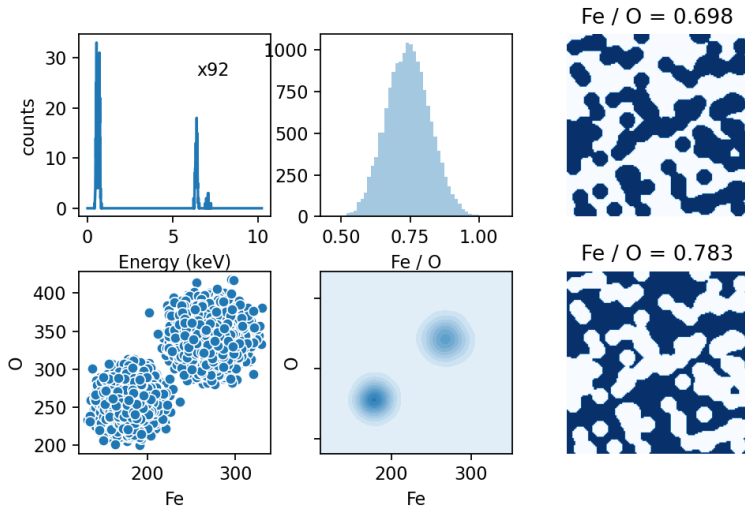
Classification by KMEANS clustering – the effect of SNR



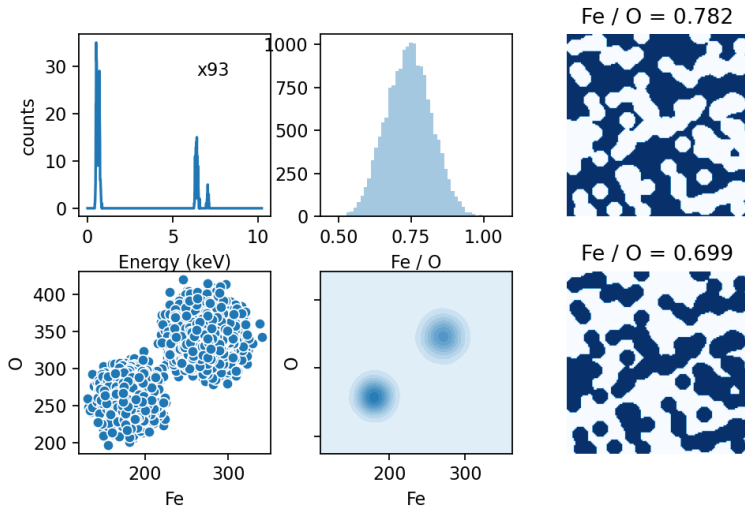
Classification by KMEANS clustering – the effect of SNR



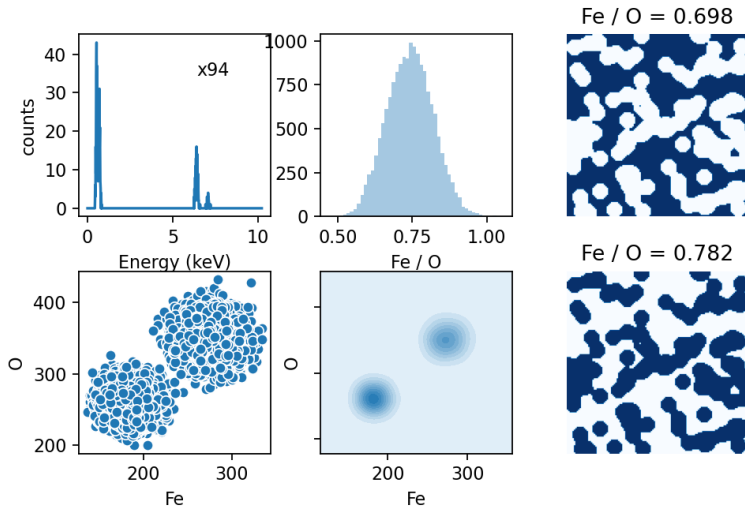
Classification by KMEANS clustering – the effect of SNR



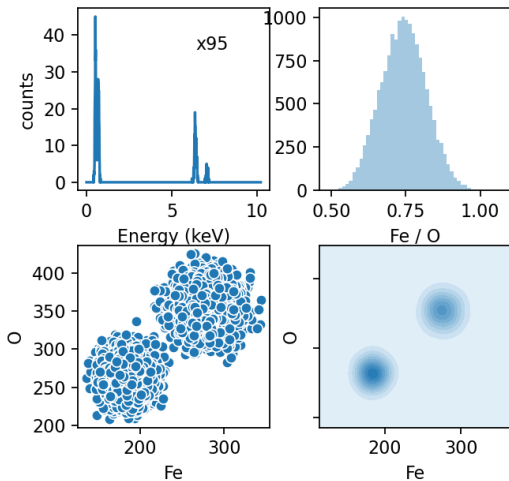
Classification by KMEANS clustering – the effect of SNR



Classification by KMEANS clustering – the effect of SNR



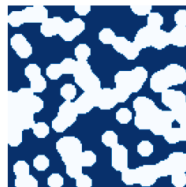
Classification by KMEANS clustering – the effect of SNR



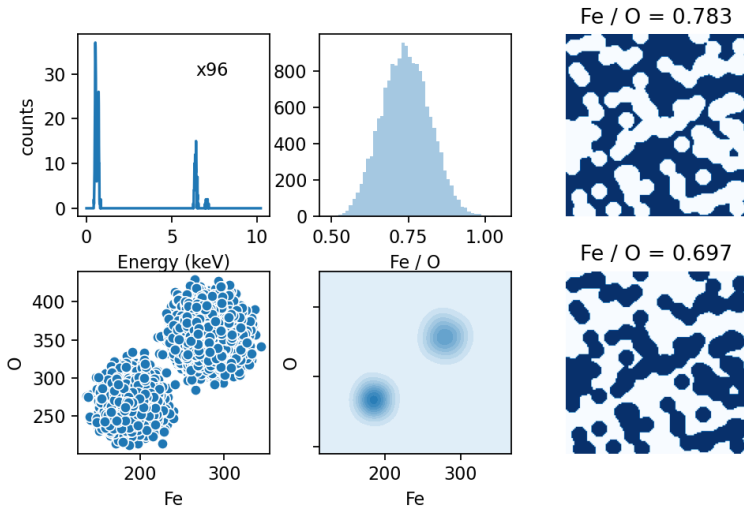
Fe / O = 0.699



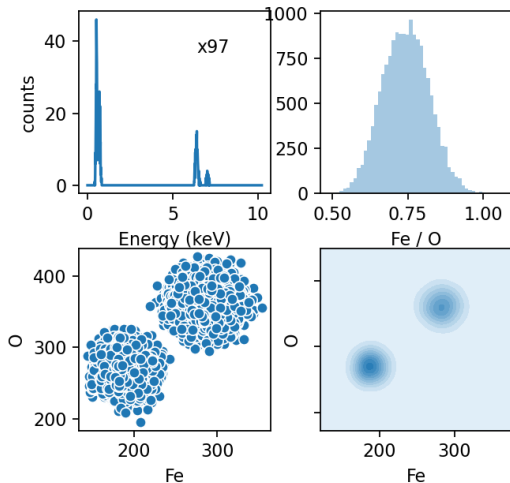
Fe / O = 0.783



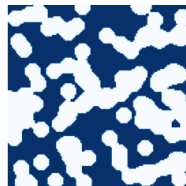
Classification by KMEANS clustering – the effect of SNR



Classification by KMEANS clustering – the effect of SNR



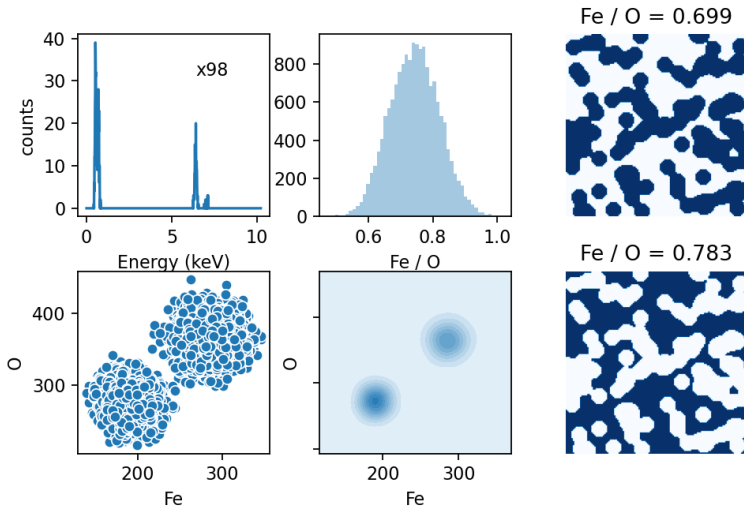
Fe / O = 0.784



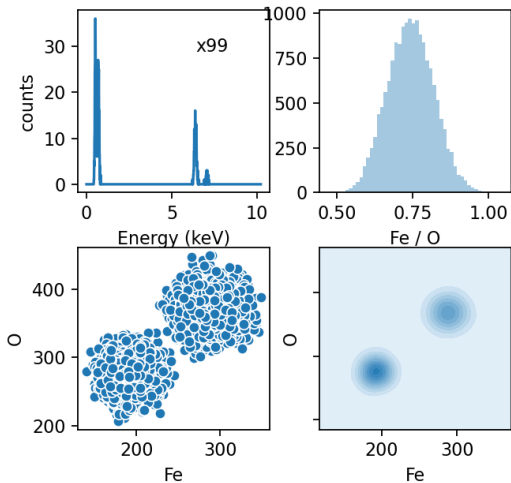
Fe / O = 0.698



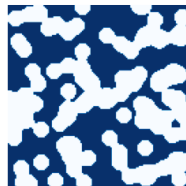
Classification by KMEANS clustering – the effect of SNR



Classification by KMEANS clustering – the effect of SNR



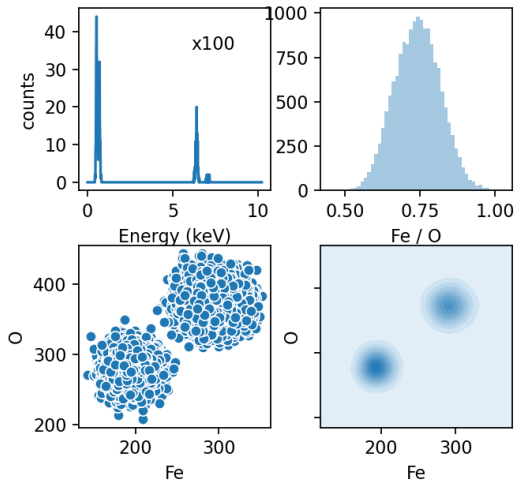
Fe / O = 0.783



Fe / O = 0.700



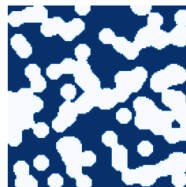
Classification by KMEANS clustering – the effect of SNR



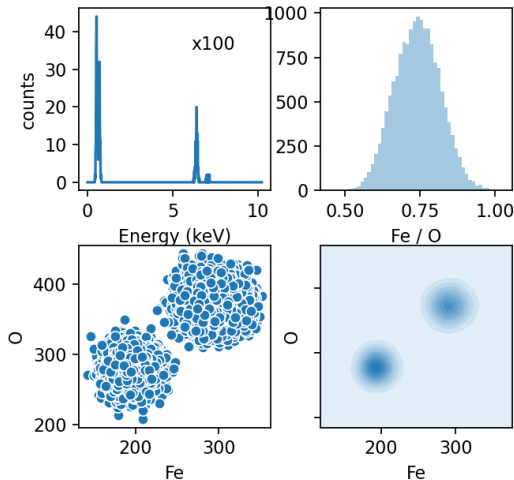
Fe / O = 0.698



Fe / O = 0.782



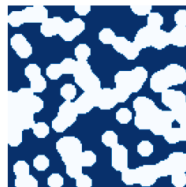
Classification by KMEANS clustering – the effect of SNR



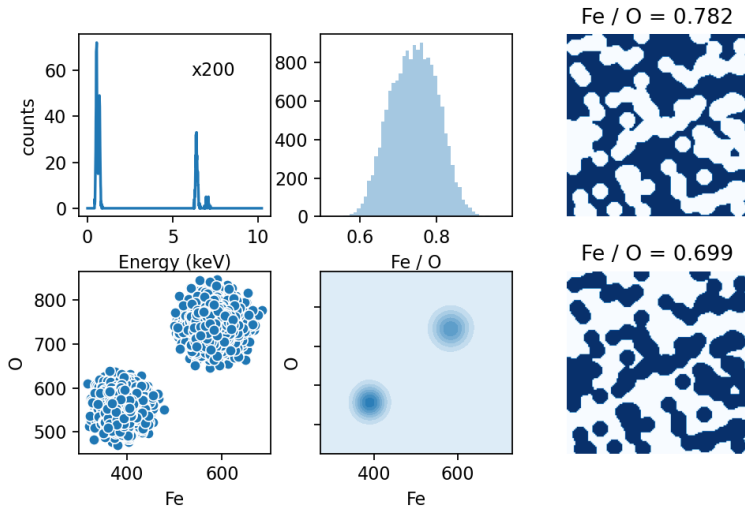
Fe / O = 0.698



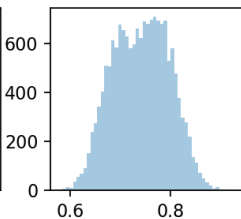
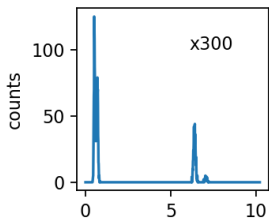
Fe / O = 0.782



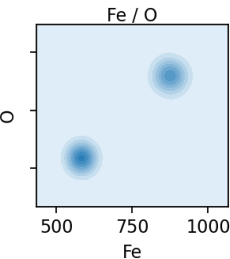
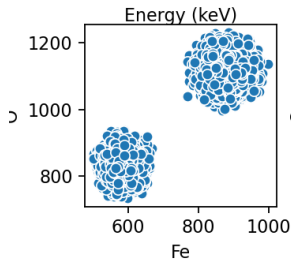
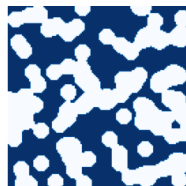
Classification by KMEANS clustering – the effect of SNR



Classification by KMEANS clustering – the effect of SNR



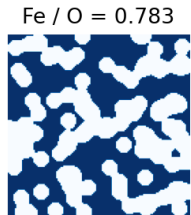
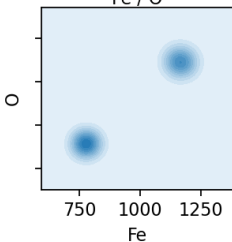
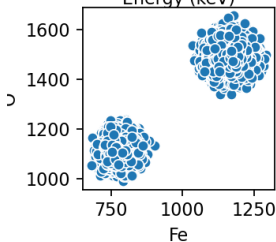
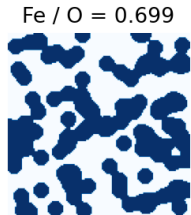
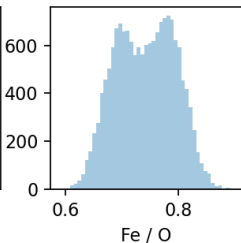
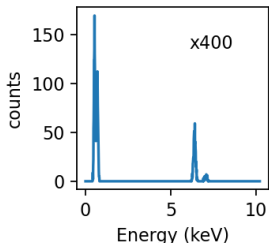
Fe / O = 0.783



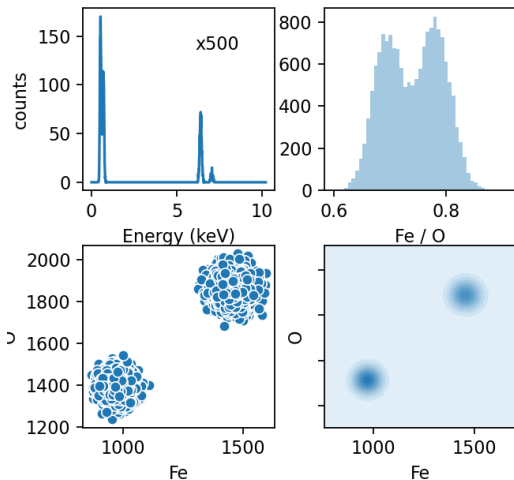
Fe / O = 0.698



Classification by KMEANS clustering – the effect of SNR



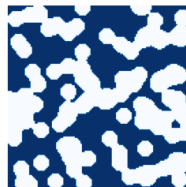
Classification by KMEANS clustering – the effect of SNR



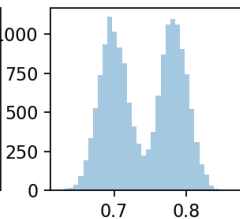
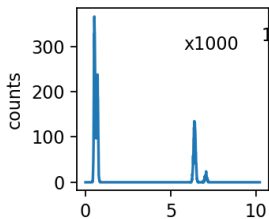
Fe / O = 0.698



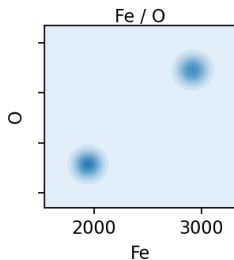
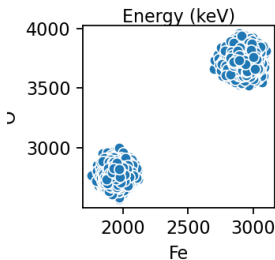
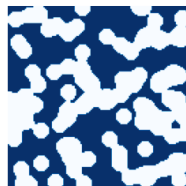
Fe / O = 0.783



Classification by KMEANS clustering – the effect of SNR



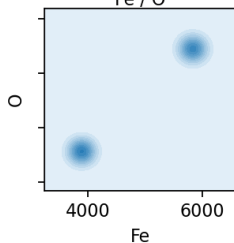
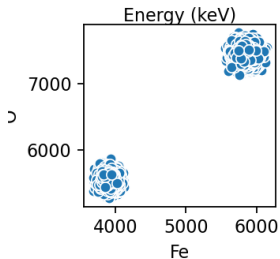
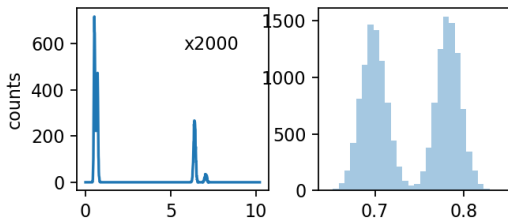
Fe / O = 0.783



Fe / O = 0.698



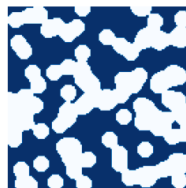
Classification by KMEANS clustering – the effect of SNR



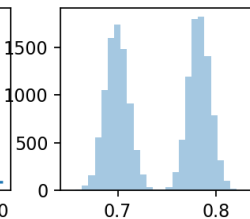
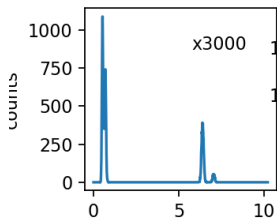
Fe / O = 0.698



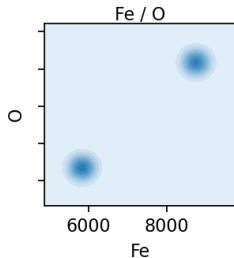
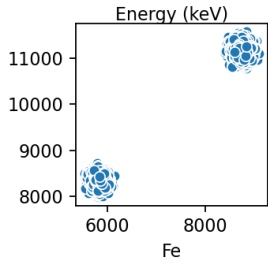
Fe / O = 0.783



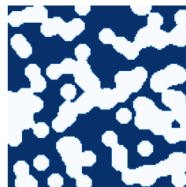
Classification by KMEANS clustering – the effect of SNR



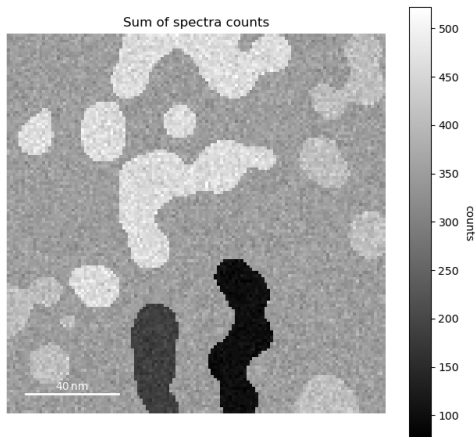
Fe / O = 0.699

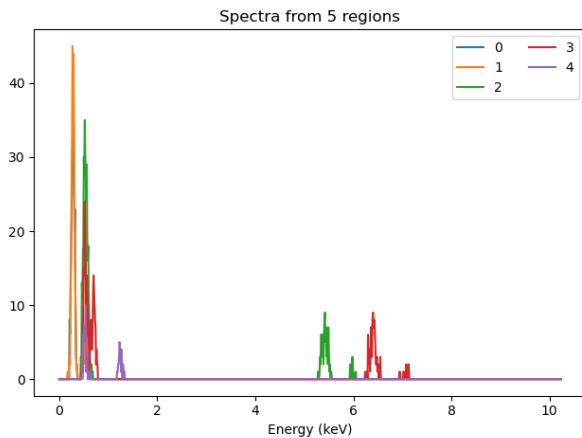


Fe / O = 0.783



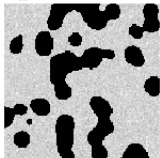
Flat sample with more compounds





Elemental maps

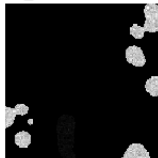
C_Ka at 0.28 keV



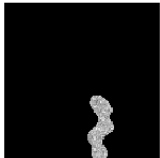
Cr_Ka at 5.41 keV



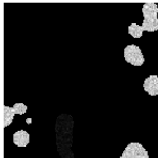
Fe_Ka at 6.40 keV



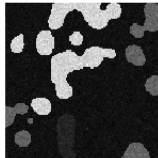
Mg_Ka at 1.25 keV



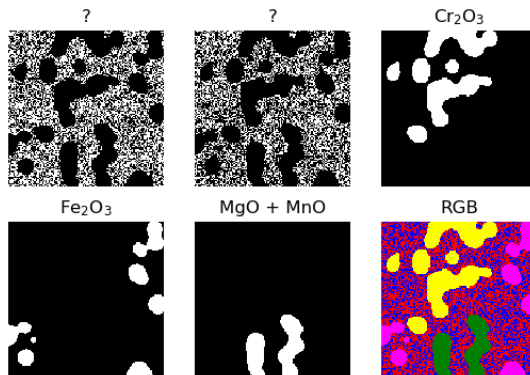
Mn_Kb at 6.49 keV



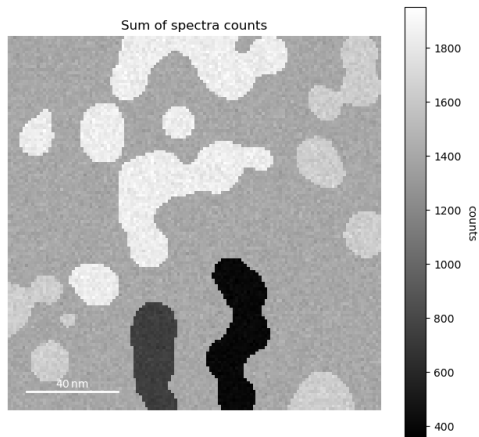
O_Ka at 0.52 keV

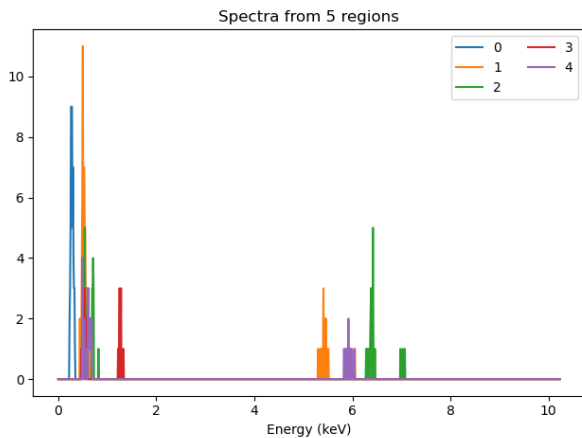


KMeans clusters



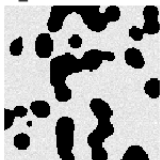
Same sample, better SNR





Elemental maps

C_Ka at 0.28 keV



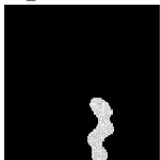
Cr_Ka at 5.41 keV



Fe_Ka at 6.40 keV



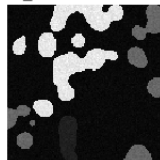
Mg_Ka at 1.25 keV



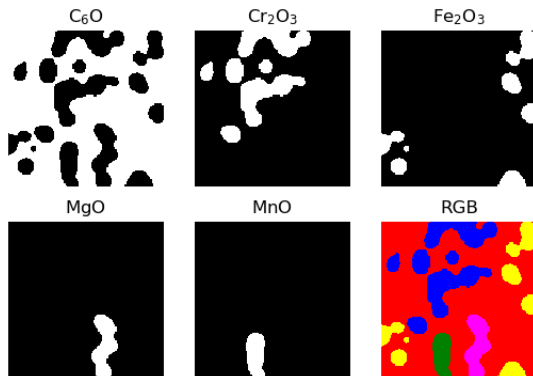
Mn_Kb at 6.49 keV



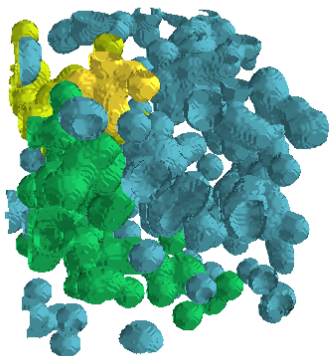
O_Ka at 0.52 keV



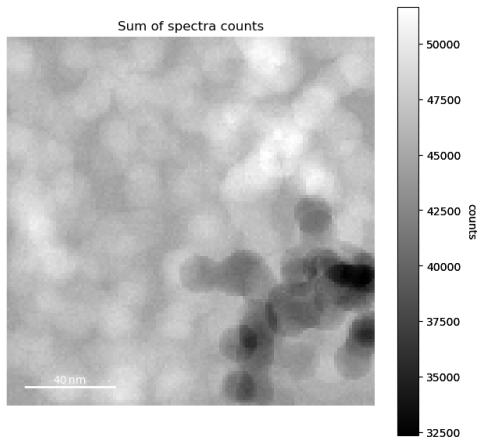
KMeans clusters



Non-uniform 3D sample

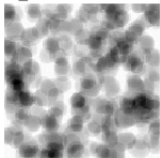


Non-uniform 3D sample projection

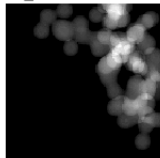


Elemental maps

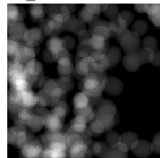
C_Ka at 0.28 keV



Cr_Ka at 5.41 keV



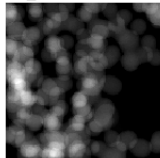
Fe_Ka at 6.40 keV



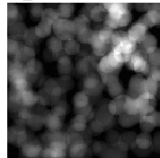
Mg_Ka at 1.25 keV



Mn_Kb at 6.49 keV



O_Ka at 0.52 keV



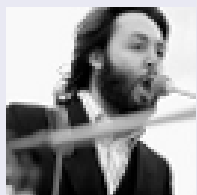
- 1 Composition segmentation
- 2 Low-rank approximation and blind source separation

The problem: linear mixing

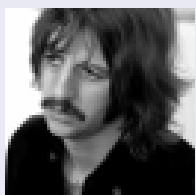
Linearity assumption

$$[a_{i,j}]_{10000 \times (64 \times 64)} = [u_{i,j}]_{(10000) \times 4} \times [v_{i,j}]_{4 \times (64 \times 64)}$$

Paul



Ringo



George



John



The problem: linear mixing

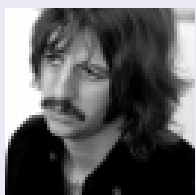
Linearity assumption

$$[a_{i,j}]_{10000 \times (64 \times 64)} = [u_{i,j}]_{(10000) \times 4} \times [v_{i,j}]_{4 \times (64 \times 64)}$$

Paul



Ringo



George



John



Mix 1



Mix 2



Mix 3



Mix 4



The solution: blind source separation

$$[a_{i,j}]_{I \times I} \times S = \tilde{S}$$

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix} \times \begin{pmatrix} \text{Image 1} \\ \text{Image 2} \\ \text{Image 3} \\ \text{Image 4} \end{pmatrix} = \begin{pmatrix} \text{Image 1} \\ \text{Image 2} \\ \text{Image 3} \\ \text{Image 4} \end{pmatrix}$$

The diagram illustrates the blind source separation equation. On the left, a 4x4 matrix of coefficients a_{ij} is shown. This matrix is multiplied by a column vector of four grayscale images of a man playing a guitar. The result is another column vector of four grayscale images, which are identical to the original images in the vector, representing the reconstructed sources.

The solution: blind source separation

$$[a_{i,j}]_{I \times I} \times S = \tilde{S}$$

The diagram shows the equation $[a_{i,j}]_{I \times I} \times S = \tilde{S}$. The matrix $[a_{i,j}]_{I \times I}$ is represented as a 4x4 grid of elements $a_{11}, a_{12}, a_{13}, a_{14}, a_{21}, a_{22}, a_{23}, a_{24}, a_{31}, a_{32}, a_{33}, a_{34}, a_{41}, a_{42}, a_{43}, a_{44}$. A large red question mark is superimposed over the middle elements of this matrix, signifying that the source matrix S is unknown. The matrix S is shown as a vertical column of four grayscale images of a person. The matrix \tilde{S} is shown as a vertical column of four grayscale images of a person, which are the mixture of the sources in S . The equation is written as $[a_{i,j}]_{I \times I}^{-1} \times \left(\begin{array}{c} \text{img}_1 \\ \text{img}_2 \\ \text{img}_3 \\ \text{img}_4 \end{array} \right) = \left(\begin{array}{c} \text{img}_1 \\ \text{img}_2 \\ \text{img}_3 \\ \text{img}_4 \end{array} \right)$.

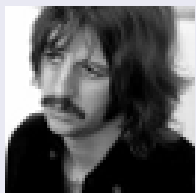
The problem: noisy multidimensional linear mixing

$$[a_{i,j}]_{10000 \times (64 \times 64)} = [u_{i,j}]_{(10000) \times 4} \times [v_{i,j}]_{4 \times (64 \times 64)}$$

Paul



Ringo



George



John



Mix 1



Mix 2



Mix 3



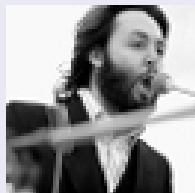
Mix 4



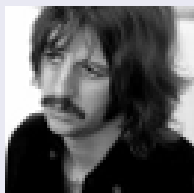
The problem: noisy multidimensional linear mixing

$$[a_{i,j}]_{10000 \times (64 \times 64)} = [u_{i,j}]_{(10000) \times 4} \times [v_{i,j}]_{4 \times (64 \times 64)} + \text{noise}$$

Paul



Ringo



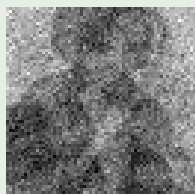
George



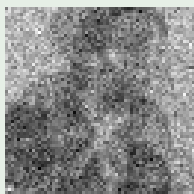
John



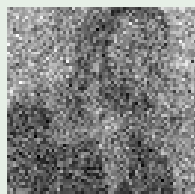
Mix 1



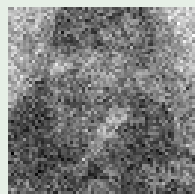
Mix 2



Mix 3



Mix 4



The solution: singular value decomposition (SVD)

Theorem

Any matrix $A \in \mathbb{R}^{m \times n}$ can be factorised into a singular value decomposition (SVD),

$$A = USV^T \quad (1)$$

where $U \in \mathbb{R}^{m \times m}$ and $V \in \mathbb{R}^{n \times n}$ are orthogonal matrices, and $S \in \mathbb{R}^{m \times n}$ is diagonal with $r = \text{rank}(A)$ leading positive entries. The p diagonal entries of S are denoted σ_i for $i = 1, \dots, p$ where $p = \min\{m, n\}$ and are called the singular values of A . They satisfy the property $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p$.

Low-rank approximation

Dimensionality reduction / low rank approximation

Theorem

Eckart-Young-Mirsky theorem

If $k < r = \text{rank}(A)$ and $A_k = \sum_{i=1}^k \sigma_i u_i v_i^T$, then

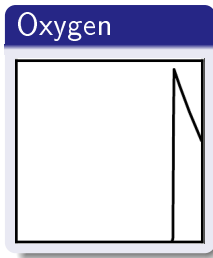
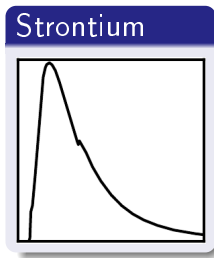
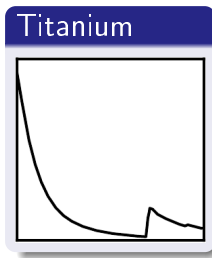
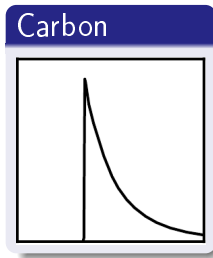
$$\min_{\text{rank}(B)=k} \|A - B\|_2 = \|A - A_k\|_2 = \sqrt{\sum_{i=k+1}^p \sigma_i^2}$$

input image: 1



EELS BSS with The Beatles

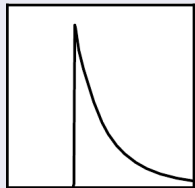
$$[d_{i,j}] \quad \times 1024 = \quad \times 4 \times [s_{i,j}] 4 \times 1024$$



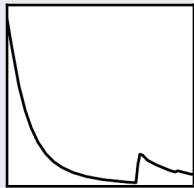
EELS BSS with The Beatles

$$[d_{i,j}]_{(134 \times 134) \times 1024} = [p_{i,j}]_{(134 \times 134) \times 4} \times [s_{i,j}]_{4 \times 1024}$$

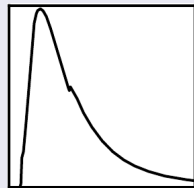
Carbon



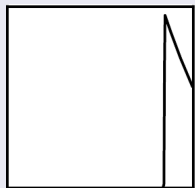
Titanium



Strontium



Oxygen



Carbon



Titanium



Strontium



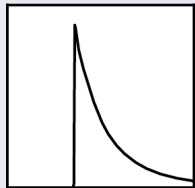
Oxygen



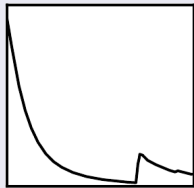
EELS BSS with The Beatles

$$[d_{i,j}]_{(134 \times 134) \times 1024} = [p_{i,j}]_{(134 \times 134) \times 4} \times [s_{i,j}]_{4 \times 1024} + \text{Poisson noise}$$

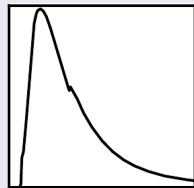
Carbon



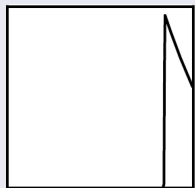
Titanium



Strontium



Oxygen



Carbon



Titanium



Strontium



Oxygen



- Singular value decomposition
 - is very useful for
 - Data denoising with no information loss
 - Rank estimation
 - Dimensionality reduction

- Singular value decomposition
 - is very useful for
 - Data denoising with no information loss
 - Rank estimation
 - Dimensionality reduction
 - The SNR improves with the number of trials in the dataset

- Singular value decomposition
 - is very useful for
 - Data denoising with no information loss
 - Rank estimation
 - Dimensionality reduction
 - The SNR improves with the number of trials in the dataset
- BSS
 - Separates sources from a mixture
 - The accuracy increases with SNR

