1 INSTALLATION

You should place the temDM MSA.gtk plugin into a PlugIns folder of DigitalMicrograph. There are typically several such folders, for instance C:ProgramData/Gatan/Plugins. If import of Velox XEDS spectrum-images is desired, you should also place there an appropriated HDF5-reading plugin. These open-source plugins were compiled by Tore Niermann, TU Berlin:

hdf5_GMS2X_amd64.dll for the 64-bit system,
hdf5_GMS2X_x86.dll for the 32-bit system.

The script find plugins folders.s included in the distribution package will help you to localize desired folders. Open find plugins folders.s in DigitalMicrograph and run it by pressing execute or by pressing ENTER with holding the CNTR key. Read the list of available plugins folders. The first folder in the list is most appropriated for placing the temDM plugins.

Some folders can be hidden in Windows. If you do not see all folders, make them visible in the Windows explorer:

Windows 7: Organize tab - Folders and search options
- View tab - click show hidden files, folders and drivers checkbox.

Window 10: View tab - click hidden items checkbox.

• Drop temDM MSA.gtk and hdf5_GMS2X_amd64.dll into the chosen Plugins folder.

• Restart DigitalMicrograph and find the MSA items in the Menu temDM : MSA.

To update the version, just overwrite the plugin of a previous version in the Plugins folder. All versions of temDM MSA have the same name to avoid confusion with loading ambiguous commands. If you have several versions, it is recommended to keep PlugIns in individual folders with meaningful names like temDM MSA basic version 1_XX.

2 BASIC PCA TREATMENT

This chapter will introduce you in how to perform Principal Component Analysis (PCA) of STEM EELS (Electron Energy-Loss Spectroscopy) or XEDS (X-ray Energy Dispersive Spectroscopy) spectrum-images. First of all you have to open the data cube where the results of your spectrum-imaging are stored. That might be an EELS data acquired by the Gatan software and stored in the standard format - with spatial dimensions along X and Y and the energy axis along Z (not directly visible). For the purpose of learning you might open the simulated EELS spectrum-image available in http://temdm.com/web/msa/. The simulated datacube EELScube.dm3 mimics the format and hidden information tags of real experimental data cubes. You can inspect how the spectrum is changing from pixel to pixel in EELScube.dm3.

• Put EELScube.dm3 in front and choose temDM : MSA : view spectrum in the DigitalMicrograph menu.

A green marker appears inside the image while a spectrum is displayed in the separate window. You notice that the spectrum is always very noisy. This
spectrum-image was generated noisy by purpose. You may drag the marker across the image - the spectrum will be live updated. By default, the marker points to the one-pixel area but you can enlarge the green rectangle and get the spectrum summed over the larger area. If the area is too large for live update, the spectrum will be updated as soon as you release the mouse button. A spectrum from larger areas look much nicer, is not it? Well, averaging removes the random noise. However, you probably need the spatial information, not just an spectrum averaged over a large area, thus you wish to know how spectra change from pixel to pixel. With the PCA denoising, you can obtain such a nice spectrum at EACH pixel. Just continue, you get it.

How to stop viewing spectrum?

- Close live-spectrum display without saving (you might keep alt-key pressed to omit DigitalMicrograph saving prompt)

Lets treat EELScube.dm3 and try to improve it.

All results of the statistical analysis will be kept in a special image EELScube MSA.dm3. Before you generate such an image, decide whether your Principal Component Analysis (PCA) will be weighted or not. The weighting treatment is needed to equalize the Poisson noise across the dataset. Weighting is strongly recommended. In particular, unweighted PCA will NOT pick up the relevant data variation in the example EELScube.dm3. Instead, the random variations of the low-energy background will be highlighted. Weighting really makes a good job!

You may choose to weight over spatial and energy dimensions or over energy only. Weighting for only spatial dimensions makes no sense for typical STEM spectrum-images.

To generate EELScube MSA.dm3:

- Open the proceed PCA tool by choosing temDM : MSA : proceed PCA,

- press make matrix button while having the original EELScube.dm3 in front. The spectrum averaged over all pixels appears. You are prompt to choose the energy region you are interested in. Move the selected region as you desire.

- press Treat selected. In case you click Cancel, the whole available energy range will be used. This guaranties you do not loose any useful information but also implies the higher memory consumption.

- If you feel something goes wrong, you can stop the conversion into EELScube MSA.dm3 at any moment by pressing stop.
image”. However the 3D data distribution is not lost! This is attached to the image in a matrix form suitable for further processing. Such image will be called MSA image through this guide. Do not be surprised that MSA images are quite large when stored. There is a lot of hidden information in it. All MSA results such as PCA loadings and scores will be kept in this image as tags. The key point is - you may play with different parameters of the PCA decomposition (like number of components, centering, clustering) and still use the same EELScube MSA.dm3. No need to start again with the original data cube. But you do have to generate a new MSA image if you wish to change the energy range or to evaluate weighted vs unweighted treatment.

The proceed PCA tool is your first temDM MSA tool. Later you will learn more tools. For your convenience, all temDM MSA tools “remember” their positions. Next time you open a tool, it appears exactly where you left it previously. If the position was accidentally set too crazy and you cannot find it when reopened, open the tool with holding the SHIFT key. The tool’s position will be reset.

Now you are ready to extract PCA components. Before doing that, it is a good idea to check few parameters. One is centering, which means that all your data variations will be counted relative the average spectrum. Centering is strongly recommended. What is also important is the number of extracted components. Keep in mind that a real system seldom consists of more then 5-7 independent variation trends. Thus, if you see some non-noise signal in your 15th PCA component, your treatment is most probably not optimal. However, the number of the extracted components must exceed the number of the expected variation trends because you should ensure that the noise level has been approached and there is no need in the further PCA decomposition.

It is recommended to start always with extracting 2 PCA components. That would give you a good idea about your dataset. Later you may add more PCA components without overwriting the first ones.

- press PCA using 2 (default) components while having EELScube MSA.dm3 in front. Wait until the extraction is finished and the results appear.
- Input more components, for example 5, and press PCA again. Decline overwriting the previously extracted components by clicking cancel. Confirm that you do wish to add more components. Extraction will start. You can interrupt the extraction procedure at any moment by pressing
Once the PCA decomposition was executed successfully, several forms of presenting the results show up.

One is a *scree plot*, i.e. the variance of data within each PCA component. The PCA components are sorted in such a way that the variance decreases with increasing the component index (this is actually the essence of PCA). Inspection of the scree plot might help you to find the number of meaningful components in the system. For instance, the *EELScube.dm3* data shows two PCA components with large variance while components of index 3-5 exhibit the lower variance that changes little from index to index. That is just a noise! It is clear that all forthcoming PCA components would not be very different because you have approached the noise level already. Thus, we conclude that this data cube exhibits two PCA components, i.e. two independent variation trends.

A scree plot is important but not the only hint to determine the number of non-noise components in a dataset. More tricks will be demonstrated below.

Another important PCA results are spectra of PCA components, or *loadings*. In this example, the only two first components show the meaningful spectra. PCA components 3-5 (hidden in the shown display) are simply noise. The average spectrum is also displayed in the same graph. All PCA components should be considered as the deviations from this average spectrum. Do not try to interpret what each component actually means at this stage of treatment. The interpretation aspects will be touched later.

This all will look differently if you choose to do uncentered PCA. Then, the first component is effectively the average spectrum, while the total number of components is increased by one. Such a style of treatment does not make much sense. If you disagree, just try it!

A very important characteristic is a *scatterplot*, i.e. the joint distribution of two selected component’s scores. After the extraction of the PCA components, the scatterplot of 1st vs 2nd score will be automatically displayed. Later you may inspect the scatterplots between any couple of components. The scatterplot of the two first components is typically most informative, that is why it is displayed automatically. In the present example you see that *EELScube.dm3* shows two independent trends of the data variation (two branches in the scatter figure).

All these three pictures are for your information only. After you have a look, just close them without saving. They are automatically stored in *EELScube MSA.dm3*. You can easily display them any time when necessary.

If you agree that *EELScube.dm3* consists of only two non-noise components, you can reconstruct the data cube using these two components only and get rid of noise.

- Choose the number of components for reconstruction. As a little exercise input 3, not 2 for *EELScube.dm3*. 

• Click **reconstruct**. You get the list of components ready to include in reconstruction. That is your last chance to reduce the number of components. Uncheck PCA 3 and press **OK**.

![Image of reconstruction tool](image)

Theoretically you can remove a PCA component in between the meaningful ones, for example remove PCA 2 but leave PCA 3. This however is NOT recommended to do unless you see a clear artefact in the intermediate component.

Finally the reconstructed data cube **EELScube REC(3).dm3** appears.

You might wish to compare it with the original data cube.

• Place **EELScube REC(3).dm3** front-most and **EELScube.dm3** second front-most; choose **temDM : MSA : twin view spectrum** in the DigitalMicrograph menu.

• To stop twin-view-spectrum close spectrum window without saving.

Green markers will appear in both the images while the extracted spectra will be displayed as an overlay in the separate window. You may drag or resize the marker at any image. The companion marker will move synchronously.

![Image of reconstructed data cube](image)

Do you see the difference between the original and the reconstructed data cubes? Well, this is the simulated data. Real examples are be not as straightforward... Nevertheless if you tune all steps carefully, the results might be astonishing as a plenty of unwanted noise is gone.

Now you probably wish to store the results of your PCA decomposition. Try to store **EELScube MSA.dm3**.

Ooops... Why this 2D image is so large? It is even a bit larger in size then the original spectrum-image. This is because the original 3D data is kept as a hidden matrix in **EELScube MSA.dm3**. But you can get rid of that.

• Click **compress** while having **EELScube MSA.dm3** in front.

A new 2D image - **EELScube COMP.dm3** - seemingly identical to **EELScube MSA.dm3** - appears. It consists of all the results of your analysis except of...
the original noisy data. Store it and find out that it is VERY COMPACT.

What is important - for 95% of further operations, EELScube COMP.dm3 is identical to EELScube MSA.dm3. You may do the further MSA (Multivariate Statistical Analysis) : rotation, clustering, endmembering and others. Enjoy the wide range of treatment possibilities with the very compactly compressed format! You do not need necessarily to store the large EELScube MSA.dm3 file. Just find the right number of PCA components, ensure that the reconstruction goes correctly and convert EELScube MSA.dm3 into EELScube COMP.dm3. Then, you may close EELScube MSA.dm3 without saving.

You can always get the reconstructed spectrum-image from your compressed image.

- Click extract while having EELScube COMP.dm3 in front.

This would take a while because the reconstruction process is to be repeated. Actually, extract is equivalent to pressing the reconstruct button with the only difference that you need not specify the list of components for reconstruction. The components list used last time before the compression will be utilized for building the denoised spectrum-image.

At this point, job is basically done. However, you are advised to explore more features of the temDM MSA package.

3 TREAT XEDS DATA

XEDS (X-ray Energy Dispersive Spectroscopy, sometimes called EDX or EDS) data cubes can be treated in the same way. You can convert the XEDS data collected by the Bruker software (ESPRIT) into a DigitalMicrograph image. As an example, use the Magical EDX example free down loadable at http://temdm.com/web/msa/.

- Store the h-4.bcf data from ESPRIT as h-4.raw file. If you do not have ESPRIT installed in your computer, use h-4.raw from the downloaded package.

- Open the import tool by choosing temDM : MSA : import in the DigitalMicrograph menu.

- Click import button in import Bruker ESPRIT box, browse and choose the desired h-4.raw file (the ripple file h-4.rpl file must also be in the same folder). Conversion take a while. You can stop it anytime by clicking stop.

Unfortunately the calibrations are not kept in the h-4.raw file, thus you should calibrate the converted data
cube manually by inputting the proper numbers in the fields “scale” (how many nanometers are in one pixel) and “dispersion” (how many keV are in one energy channel). The value “origin” is usually 0 for spatial calibration but must be set at some positive number for energy one. It varies from system to system but fixed for a given spectrometer and a given spectrometer dispersion. Just check which channel is assigned to ZERO in your ESPRIT. For the MagiCal EDX example, the calibration should be
scale: 0.67 nm
dispersion: 0.01
origin: 48.

- Click calibrate while having the converted spectrum-image front most.

You can also import XEDS spectrum-images stored by FEI Velox (version > 1.2). As an example, use the superalloy 1510 example free downloadable at http://temdm.com/web/msa/.

- Click import button in import FEI Velox box, browse and choose the superalloy 1510.emd file.

- The range of available frames to integrate in the spectrum-image appears. If you are sure all frames are appropriated for integration, just click OK.

This spectrum-image will be already properly calibrated. If it is not (might happen because of misinterpretation of attached metadata), perform calibration manually.

XEDS data are typically quite sparse. This makes severe problems in PCA treatment. Therefore, it is strongly recommended to pre-treat the XEDS dataset before the PCA processing. You can, for instance, bin it.

- Open the filtering tool by choosing temDM: MSA : filtering in the DigitalMicrograph menu.

- Choose the binning factors 2 on both X and Y.

- Click run button.

This would reduce the sparseness in the data and greatly reduce its spatial dimensions although the number of energy channels will be unchanged (for the moment, there is no energy binning in the temDM MSA package).

In case your PC has no problem to handle large spectrum-images, you might smooth the very noisy data by application of the “gaussian” filter. This filter makes the weighted averaging according the Gaussian function kernel. The parameter is the standard deviation sigma (expressed in pixels) in the Gaussian distribution. Both spatial and energy filtering is available. Currently, the allowed range is 0.6 < sigma < 2.6. Just to try it:

- Having the binned h-4_2x2.dm3 spectrum-image in front and leaving the default sigma=1 in Gaussian filtering SI' box, press spatial or energy to proceed filtering.

The Gaussian filtering is rather long procedure in the basic version, thus reserve some time for it.

Finally you might wish to check how your average image (image averaged over all energy channels) or average spectrum (spectrum averaged over all probes) looks like.

- Click spectrum or image.

Then, you may denoise the resulted h-4_2x2_xGss cube like it was described in the previous section.

### 4 View Results

- Open the view results tool by clicking temDM: MSA : view results.
With this tool you can any time retrieve the stored information from EELScube MSA/COMP.dm3. For instance you can

- type the two component’s indexes in the input fields and plot the corresponding scatterplot by clicking plot X-Y.

The sequence of the indexes plays a role! The index which stays in the left-hand field will be your horizontal axis while the right-hand index will be the vertical one.

You can now investigate the data distribution in the factor space with the desired deepness. Actually, you will quickly discover that all scatterplots with the PCA components higher than 3 are essentially same - they form a round distribution of the uncorrelated noise nature. This is another way to determine the number of the non-noise PCA components in your dataset.

Then you get the maps showing how each PCA component is distributed across the object. The scores can be delivered as a number of 2D images or may be compacted into a data cube (chose your preference throw the spanner button). In the latter case, you can scroll among different PCA components using the standard DigitalMicrograph tool - Slice. Again you see that the scores of the PCA components with the index 3 and greater show nothing but noise.

Explore now the three buttons at the bottom of the view results tool.

- Clicking variance will plot the standard scree plot that you have seen already after finishing your PCA decomposition.
- Clicking kurtosis will calculate and plot the kurtosis for each extracted component.

Kurtosis is a measure of non-Gaussianity in the data distribution. It may be negative or positive for an arbitrary distribution but it must be exactly zero for an ideally Gaussian-distributed data.

You might notice that the components with the index 3 and greater show almost zero kurtosis. A Gaussian-distributed component is very likely just a noise. This is an alternative way to determine the number of non-
Finally you can inspect how the PCA decomposition was proceeded.

- Click iterations.

You have now access to the number of iterations required for finding each component. You can also see how the variance and precision (difference between two subsequent iterations) were progressing during the iterations. This inspection help to identify the cases when the PCA algorithm did not converge for any reasons.

5 Rotation of PCA Results

You probably noticed that the obtained PCA loadings are not well interpretable. This is a typical feature of PCA - you can describe successfully the important data variation with few loadings but you cannot say what each loading actually means. You might try to improve interpretability by rotation of the PCA loadings in the factor space. Three rotation methods are incorporated in the temDM MSA package:

Independent Component Analysis (ICA)

Maximal simplicity (varimax)

Free rotation

How to do that?

- Open the MSA rotation tool by clicking temDM : MSA : rotate results.

The rotation operation implies visualization of a scatterplot for the selected couple of components. This noise components. Caution: a very large Kurtosis is also not good - this is a fingerprint of heavy outliers in the component.
is needed for interaction with a user. Before starting the rotation, define the indexes of the most representative components. For the example EELScube MSA.dm3 (you can use also the compact EELScube COMP.dm3 version), leave the default 1-2 scatterplot.

Lets start with ICA. ICA attempts to find the directions along which the data distribution deviates most drastically from the Gaussian distribution. A criterion to be maximized is the kurtosis which is zero for the Gaussian distribution and positive or negative for the non-Gaussian ones.

- Click ICA. You get the list of PCA components to be rotated. You may uncheck any of the components, then they will be fixed, i.e. not rotated. Do not do that for the moment. Press OK.

The rotation matrix appears which shows how the new components are composed from the old ones. You might notice that components 1-2 are rotated moderately while components 3-5 very significantly (and randomly) altered. This is because components 3-5 are pure noise, thus the algorithm is not able to find any optimal combination of them. The only rotation of components 1, 2 is meaningful.

Finally you see the kurtosis of the new components and their loadings. The kurtosis of two first components is pretty large (although negative for component 1 and positive for component 2) while it is close to zero for components 3-5. That is because the latter are noise.

You also see the 1-2 scatterplot, which indicates that the major variation trend is almost along with the loading of ICA 1 component. This is the best job ICA can deliver for this set. In fact, ICA results do not differ very much from the PCA ones for this example. In other datasets, this might be very different.

Lets explore now the varimax method. This method attempts to achieve the best simplicity of data - that means the majority of pixels must have the large coordinate along one axis while zero or small coordinates along the other axes. In other words, the data points should be as close (in average) to the coordinate axes as possible.

- Click varimax. You get the list of PCA components to be rotated. You may uncheck any of the components, then they will be fixed, i.e. not rotated. Do not do that for the moment. Press OK.

The new components loadings and scatterplot appears.
You don’t see much changes? The PCA components are rotated by only 3 degree and there is no any particular simplicity in the results? That is because the rotation center was fixed. With such a rotation center (which is an average spectrum) you cannot get the better simplicity. Try another option - varimax with redefinition of the center.

- Click varimax(C). The scatterplot appears with the red spot denoting the rotation center. Drag the spot to the desired place in the 1-2 component plane. Press New center. As before you get the list of PCA components to be rotated. Change nothing, press OK.

The results are now much more reasonable. The majority of the data pixels are close to 1 or 2 axis and loadings are better understandable. Of course, the extracted loadings correspond to the expected latent factors only approximately. This is because two trends are not perpendicular to each other in the factor space while the rotation is kept orthogonal in the temDM MSA package. More accurately interpretable loadings can be extracted by finding EndMembers as will be described in the corresponding section.

Generally, the center of the factor space seldom lies on the PCA component axis. In this situation, the classical varimax rotation is not very successful unless you redefine the center. For the moment temDM MSA requires manual adjustment of the rotation center.

Such recentering may be performed prior the ICA analysis too (button ICA(C)). This however would not deliver much better results in this particular example.

Finally you might practice in choosing the components to rotate. Do the varimax rotation as above but uncheck the components 3-5 in the list. You get the same varimax results as before but much faster because the senseless rotation of the noise PCA components is omitted.

The trick with redefinition of the center can be naturally extended to the rotation fully defined by a user. This is called free rotation in the temDM MSA package.

- Click free. The scatterplot appears with the red arrow in the middle. The live spectrum profile shows the loading corresponding to the current direction of the arrow. Move around the red arrow on the scatterplot while watching the resulted loading spectrum. Orient the arrow such a way that its beginning denotes the rotation center while its end points to the desired direction for the best interpretable loadings. Press Rotate.

You get components 1 and 2 rotated as you specified. Of course, the rotation was proceed in the 1-2 plane only, while the rest components are unchanged. If you wish to rotate another axes, change the components indexes at the bottom of the MSA rotation tool. Free rotation affects only a couple of components at once. This is different from ICA and varimax that can rotate all available PCA components in one action.

This style of rotation is of coarse subjective, no quantitative criterion stays behind. However the real systems are often so complicated that any established ro-
tation criteria fail. Still human brains have a chance to pick up the hidden structure in the data distribution. A flexible look-inside sometimes overperforms any fixed algorithm!

- Click free. Move around the red arrow and learn what kind of spectrum appears in the live profile. After you have investigated all directions, press Cancel. The rotation will be NOT executed.

The free rotation may be proceed iteratively using the different couples of components. The results of the previous free rotation are not lost but suggested to you as the initial point for the next rotation operation. This is different from ICA and varimax which always start from the original PCA components.

If you feel your multiple free rotations has came to the configuration you don’t understand anymore, you always can return to the original orientation of the PCA components.

- Just click erase. The next free rotation will start from the original PCA components.

6 CLUSTERING

The example EELScube MSA.dm3 shows the two trends of variation, more or less independent of each other. This is a typical feature of STEM datasets; the real systems might show even more complicated structures. One way to deal with the complexity of datasets is to break it on several clusters in the factor space.

- Open the clustering tool by clicking temDM : MSA : clustering.

- Having EELScube MSA.dm3 in front, generate the scatterplot of the representative PCA components by clicking plot X-Y. The scatterplot of the two first PCA components is usually most useful.
• Click new. The ellipse appears in the scatter plot.

• Orient the ellipse in the way it covers the points to be included in the cluster. You can drag the main axis of the ellipse and tune its proportion by stretching/expand the minor axis. Click store when ready. You may also exit the process without storing a cluster by clicking cancel.

• Click again new cluster and manage the second cluster. Click store. If some pixels from the old cluster appear to lay within the new one, they will change their affiliation, i.e. will be attached to the new cluster.

do not need to store it. They are stored automatically in EELScube MSA.dm3. You may check it:

• press show all. The binary images showing the geometry of each cluster in the real space appear. Additionally the average spectrum for each cluster will be calculated and shown.

Clusters are shown by points of different colors in a scatterplot. The default colors are - 1: red, 2: green, 3: blue, 4: pink, 5: yellow, 6: light-blue, 7: brown. For your convenience the pallet is always shown on the tool. All clusters with the index higher than 7 will be orange because the colors in a pallet are limited.

One point should be stressed: Although you design your clusters in a 2D-plot, the actual structure is multi-dimensional. Thus, it is recommended to view it from several scatterplots.

• type 3 in the first field of the scatter components. Press plot X-Y. A new scatter plot appears with the same clusters but seen from the other perspective.

You might notice that every time you create a new cluster, a certain binary image appears that tells you how your cluster looks in the real space. As usual, you

You may continue to design your clusters at any scatterplot you want. The changed clusters will be automatically repainted in the other scatterplots. Such
multi-dimensional manipulation sounds challenging but after you get used to scatterplots, you will really have a fun exploring the n-dimensional space!

- You can anytime delete the last designed cluster. Then all its pixels will be marked as not belonging to any cluster and repainted in black.

- You can also correct any of existing clusters. In that case you will be prompt to choose the color of the corrected cluster, then the ellipse for choosing the new area will appear. Just drag it to the desired position and press store.

**Important:** the old pixels are not excluded from the corrected cluster but are ADDED to those located in the new ellipse. This way you can design clusters of a very complicated geometry.

You have now two clusters stored in **EELScube MSA.dm3**. You should proceed PCA again, this time in each cluster individually. To do that

- Open the proceed PCA tool by choosing **temDM : MSA : proceed PCA**.

- Input 2 in number components (it is sufficient). Click **PCA**. The package recognizes that data are clustered and ask you whether you wish to treat clustered.

- Choose treat all clusters and press OK. Clusters will be treated non-stop one after the other.

Do you see the difference? Each cluster is now essentially a one-componential object. It shows a simple scatterplot and a very transparent loading. As previously, close all displayed loadings, variances and scatterplots without saving. Do not create mess on the DigitalMicrograph desk! All these results are already stored in **EELScube MSA.dm3**.

So, clustering might be used to break the data onto the elementary pieces of simply interpretable structures. Often, users are interested in one particular part of an investigated area. In that case you might design the
only one cluster and then treat cluster 1 while the rest of the data will be ignored.

Finally, you can make reconstruction using the clustered data.

- Click Reconstruct. A list of available clusters will appear.

- Choose the desired clusters and the desired number of components in each cluster. One component in each cluster is sufficient for EELScube. Press OK.

The reconstructed cube combines the pixels from all chosen clusters. The pixels which do not belong to any chosen clusters will appear dark.

7 VIEW ROTATED AND CLUSTERED RESULTS

The view result tool can be used to display loadings, scores and scatterplots for your rotated or clustered data.

- Open the view results tool by clicking temDM : MSA : view results.

- Choose cluster index. Index 0 implies NO clustering.

- Choose the rotation method in the drop down menu.

- Use the appropriate buttons to view the desired stuff.

You can also check the variance or kurtosis of the obtained rotated components. Get iteration will show you the available information about how your rotated components were obtained. This is content-dependent. The free rotation has no iterations at all. The iteration in varimax is done for all components at once.

We have included extra methods in our extended treatment - clustering and rotation (and more methods like alternating least-square (ALS) fit will be further explored). This means we are now not within Principal Component Analysis (PCA) anymore but rather stepped into the more general area - Multivariate Statistical Analysis. (MSA).

8 ENDMEMBERS

You see that the MSA treatment can produce a lot of results to view and analyze. Maybe too much...

Review of such output implies strong abstract thinking: the loadings are differential, i.e. counted from some average spectra; the scores may be both positive and negative, et cet. et cet.

Can we come to some simpler, intuitively clear description of the results? Yes.

Try to recast it in terms of EndMembers. In such description you deal with few well interpretable spectra and few maps showing the contribution of these spectra across the object. Use again the example EELScube MSA where you have extracted at least 5 PCA components.

- Open the End Members tool by clicking temDM : MSA : end members.

- Click tune new while having EELScube MSA.dm3 or EELScube COMP.dm3 in front.

A couple of new windows will appear. Find the 1-2 scatterplot. The red and green markers are located in the center of the scatter plot. They denote the positions of two EndMember spectra in the factor space while two other windows display the actual spectra. Try to drag the EndMember markers within the scatterplot and see how the spectra will change.
For your convenience, you may move the displayed spectra somewhere to the side of the DigitalMicrograph working space, zoom the intensity scale and the range of the displayed energy channels using the usual tricks of DigitalMicrograph (scale is shifted by dragging mouse with left button down while the CNTR key should be hold to zoom the scale).

From the very name of the method you might guess that the reasonable EndMember positions are usually found at the periphery of your scatterplot. How to arrive at the right positions?

By playing around you might discover some kind of a “saddle point” where the spectrum looks particularly simple. The deviation from that point results in the unphysical profile or in complication of the spectrum. In the given example, place the red marker such as the peak at 530eV disappears.
Similarly, move the green marker a bit up-down and find the position where the peak at 1310eV is NOT observed at the corresponding spectrum.

Have you noticed that the significant portion of the scatterplot remains unattended? This is because this object requires at least 3 EndMembers for the adequate description.

- Click add. The third (blue) EndMember marker will appear with the corresponding spectrum.

As previously, try to find an optimal “saddle point” for its position. In this case it would be the condition of the complete disappearance of the peak at 1530eV.

You might note that during the tune of EndMembers, a special image - “PCA fractions” - appears. It shows the content of each EndMember in terms of the original PCA components. Do not close it until you finally tuned everything - this mixture matrix is important for the tuning process: when it is opened, tune is active.

Up to now you explored the only one scatterplot. It is OK for this particular example but usually insufficient for real-life objects. You must look at your dataset from the perspective of SEVERAL scatterplots.

- Click plot X-Y. The fields at the right denote the desired components to be plotted. A new scatterplot will appear while your tuned EndMembers are already there.

Now you can have fun dragging the EndMember marker at different scatterplots. They will move synchronously in both displays. Not bad exercise for the n-dimensional thinking, is not it?
The 3-2 scatterplot is not very useful for the given example. The theoretically reasonable EndMembers should all sit at the line coinciding with the 2nd PCA component axis while the contribution along the 3rd one must be zero.

This scatterplot is however useful to learn another option of the temDM MSA program - Alternating Least Square (ALS) fit.

Even with this simplified example, it is not possible to check all and any scatterplots. If you followed the present guide, you have already extracted at least 5 PCA components - that is a 5-dimensional space! In real systems, 5-10 dimensions are quite typical.

To handle it, temDM MSA package employs the combination of the manual movement of EndMembers in the most important dimensions with the automatic ALS fit along the other ones.

Just to play:

- displace one of the EndMember in the 3-2 scatterplot outside the axis of the 2nd component.
- Click fit ALS. You see that the EndMember markers order them self in one line. Click several times fit ALS (ALS is an iterative method) and you notice that this line will approach the axis of the 2nd component.

Finally, plot the maps showing the fractions of all EndMember spectra in the dataset.

- Click show maps. Three maps each corresponding the specific EndMember will appear. All the EndMember spectra will be displayed overlaid in the separate window.
You probably guess that the 1st EndMember spectrum corresponds to pure Al, the 2nd - to AlO and the 3rd to MgO EELS spectra. The maps show nicely the distribution of these compounds. You might combine them into one colored map by using the standard DigitalMicrograph tool - Color Mix.

9 SETTINGS

Some tools have a spanner icon giving you access to the processing and displaying parameters. Here is a short description of tunable parameters. Note that several tool box may have access to the same parameter.

**filtering tool:**
- **default binning:** Default binning value.
- **default spatial sigma:** Default standard deviation \( \sigma \) in Gaussian smearing among pixels.
- **default energy sigma:** Default standard deviation \( \sigma \) in Gaussian smearing among energy channels.
- **clip after spatial filtering:** Filtering is not perfect at the border of spectrum image because there are less neighbors to average. You may choose to clip the border pixel by checking this box.
- **clip after energy filtering:** The same for energy filtering - you may clip the border energy channels

**import tool:**
- **show HDF5:** The tag structure of the Velox HDF5 format may be shown. This helps to retrieve the standard names for data pieces if they differ from the default ones.
- **default scale:** Default scale for spatial calibration.
- **default spatial origin:** Default origin pixel in spatial calibration.
- **default scale unit:** Default units for spatial calibration.
- **default dispersion:** Default dispersion for energy calibration.
- **default energy origin:** Default origin channel in energy calibration.

**proceed PCA tool:**
- **PCA convergence:** Convergence of the NIPALS algorithm of PCA can be estimated either from the angular difference between two subsequently found loading vectors (loading spectra are vectors in the n-dimensional energy space) or from the difference in their eigenvalues.
- **PCA max deviation:** Criterion for stopping iterations in the NIPALS algorithm - maximal allowed difference between two subsequent iterations. Here: 0.02 rad (if angular convergence is chosen) or 0.02% (in the case of value convergence).
- **PCA max iterations:** Maximal number of iterations in
the NIPALS algorithm of PCA. If it is reached, the procedure will be stopped even if the stopping criterion is still not satisfied. The warning message “Precision ?” will be issued. This usually happens when exploring the pure noise components.

**view results tool:**

- **sparsity in X-Y plot:** Sparsity parameter defines a mesh to fill in a X-Y scatterplot. In a too sparse plot, tiny black pixels might be hardly visible. A too dense plot looks rough. Here the number of the empty (white) pixels will be 10 times the number of the occupied ones.
- **max displayed loadings:** Maximal number of displayed loads (too many slices might make the display hard to handle).
- **scores cubed:** Scores can be displayed as many individual images or as one datacube.

**rotation tool:**

- **ICA:max deviation:** ICA convergence criterion: maximal angular difference between two subsequently iterated loadings, here: 0.02 rad.
- **ICA:max iterations** Maximal number of iterations in the fast ICA algorithm.
- **ICA:contrast function:** fast ICA algorithm requires a certain “contrast” function for iterations. temDM MSA can use cubical polynomial (p3), hyperbolic-tangential (tanh) or exponential (exp) contrast functions.

**end members tool:**

- **sparsity in X-Y plot:** Sparsity parameter defines a mesh to fill in a X-Y scatterplot. In a too sparse plot, tiny black pixels might be hardly visible. A too dense plot looks rough. Here the number of the empty (white) pixels will be 10 times the number of the occupied ones.
- **member pointer size:** The size of the marker pointing the position of loading in the X-Y scatterplot.
- **lambda:** Regularization parameter in Alternate Least Square (ALS) fit.
- **maps cubed:** Maps can be displayed as many individual images or as one datacube.

**clustering tool:**

- **sparsity in X-Y plot:** Sparsity parameter defines a mesh
to fill in a X-Y scatterplot. In a too sparse plot, tiny black pixels might be hardly visible. A too dense plot looks rough. Here the number of the empty (white) pixels will be 10 times the number of the occupied ones.

**ellipse points:** Ellipse for capturing clusters may be plotted with different number of points (here 10 points). The larger number of points makes the shape smoother but works slower.

**ellipse gap:** To move easier the ellipse its axes extend few pixels outside the ellipse itself (here 3 pixels outside).

**mask cubed:** Cluster masks can be displayed as many individual images or as one datacube

### 10 Scripting Support

You might design your own automatic treatment flow if you know the basics of DigitalMicrograph scripting. The package temDM MSA extends the library of DigitalMicrograph scripting language by adding the specific MSA functions. The file temDM MSA scripting.chm included in the distribution package consists of description of the available commands for MSA.

There are a couple of scripting examples in the distribution package. Note that the temDM MSA commands consist of “MSA_112” in their name. The rest commands are the part of the standard DigitalMicrograph scripting interface.

The following example simulates an “EFTEM series” from a STEM datacube.

```plaintext
//This script simulates an "EFTEM series" from a STEM datacube

//Energy limits for EFTEM series
number E_Start_eV = 1230
number E_End_eV = 1380
number E_Step_eV = 50
number E_Window_eV = 40

//get DataCube
image DataCube := getFrontImage()

//read its dimensions
number width,height,depth
DataCube.get3Dsize(width,height,depth)

//read energy calibration
number dispersion = DataCube.
    ImageGetDimensionScale(2)
number origin = DataCube.
    ImageGetDimensionOrigin(2)
result("\ndispersion_"+dispersion +"_origin_"+Origin)

//iterate through energy and integrate within an energy window
for (number E_eV = E_Start_eV; E_eV <= E_End_eV; E_eV += E_Step_eV)
{
    //convert from eV to channels
    number Start = MSA112_eVtoCh(E_eV-E_Window_eV/2,origin,
        dispersion)
    number End = MSA112_eVtoCh(E_eV+E_Window_eV/2,origin,
        dispersion)
    result("\n"+E_eV)
    //select a section from the DataCube
    image CubeSection = DataCube
        [0,0,Start,width,height,End]
    //render along energy axia
    image Eftem = MSA112_Render3DtoFront(
        CubeSection)
    //set name ans display
    Eftem.setName("EFTEM_"+E_eV+"eV")
    Eftem.showimage()
}
```

The next script investigates the dependence of the eigenvalues on the number of pixels in the dataset.

```plaintext
//This function generates a random mask
image randomCluster(number N, number fract)
{
    image Cluster:= realImage("cluster",4,1,N)
    //mask image of total N pixels
    number range = 1/(1-fract)
    Cluster = random()*range
    Cluster = tert(cluster>1,1,0)
    //fract*N of pixels are 1, (1-fract)*N pixels are 0
    //choice is random
    return cluster
}
```

//parameters for convergence
number deviaSpec = 0.02
number maxItt = 50
//get an MSA image where the dataMatrix is kept as an attachment
image MSAimage := getFrontImage()
image DataMatrix = MSA112_GetMSAMatrix(MSAimage)

//get the number of pixels
number m = DataMatrix.
ImageGetDimensionSize(1)

//define range and steps for fractioning the dataMatrix
number fractIni = 0.1
number fractFin = 1
number fractStep = 0.1

//fractioning
for (number fract = fractIni; fract <= fractFin; fract = fract + fractStep)
{
    //make a random mask to select a certain fraction from DataMatrix
    image Cluster := integerImage("cluster",2,0,1,m)
    Cluster = randomCluster(m,fract)
    number actualFract = mean(Cluster)
    //actual fraction can slightly differ from the nominal one
    result("\nfraction\n" + actualFract)

    //apply the mask to DataMatrix
    image DataMatrixSmaller = MSA112_TakeCluster(dataMatrix,1,cluster,1)
    //1: do centering; 0: do not
    //mask with index 1 is chosen

    //declare values and images
    number EigenValue // eigenvalue of a component
    image Pvec,Tvec // loading and score of a component
    number Nitt   // number of iterations required for convergence

    //extract 1st PCA component
    Nitt = MSA112_FindNIPALS(DataMatrixSmaller,Pvec,Tvec,EigenValue,0,deviaSpec,maxItt)
    EigenValue /= (m*actualFract)
    //now the eigenvalue is the variance within a component
    result("\neigenValue1\n" + EigenValue)

    //reduce DataMatrix by the 1st PCA component
    DataMatrixSmaller = MSA112_ReduceDataMatrix(DataMatrixSmaller,Pvec,Tvec)

    //extract 2nd PCA component
    Nitt = MSA112_FindNIPALS(DataMatrixSmaller,Pvec,Tvec,EigenValue,0,deviaSpec,maxItt)
    EigenValue /= (m*actualFract)
    //now the eigenvalue is the variance within a component
    result("\neigenValue2\n" + EigenValue)
}

11 Troubleshoot

An error with the message “ambiguous function” is generated.

Most probably you have loaded tem DM MSA.gtk twice. Localize all Plugins folders with script find plugins folders.s and check them. Only one tem DM MSA.gtk plugin must be there.

You cannot load a temDM MSA tool from the menu. There is a failure message or the tool frame simply does not appear.

Last time, the tool has finish functioning incorrectly. Open the tool again with holding the “shift” key. The tool will be refreshed.

You played too much with the temDM MSA settings. The program now fails or runs too long. You have no idea what have you changed

Choose New Script in the DigitalMicrograph menu. A new text window appears. Type there MSA112_delete() and press Execute. The default setting will be restored.
REFERENCES

