# EDS Classification for MATLAB

Austin M. Weber [1,2*]

1 Byrd Polar and Climate Research Center, Columbus, Ohio **2** School of Earth Sciences, The Ohio State University, Columbus, Ohio ***** These authors contributed equally.

Summary

`EDS Classification for MATLAB` provides a suite of useful functions for working with scanning electron microcope (SEM) and energy dispersive x-ray spectrometry (EDS) data in MATLAB. The primary objective of the software repository is to provide users with functions to quickly and accurately determine the mineral compositions of single particles from their EDS data. This includes a novel machine learning classifier and three additional sorting algorithms that have been transcribed from the peer-reviewed literature. The repository also provides functions for importing and visualizing x-ray energy spectra, as well as functions for extracting and working with hidden metadata in SEM images.

Statement of need

The mineral classification of single particles (i.e., dust) has major implications for atmospheric and glaciological research. For instance, there is a great need for building mineral dust datasets that can be used to improve the predictive ability of climate models (Gieré, 2010). Mineral dust observations are also severely limited in many glaciated mountain regions, meaning that the dust's radiative impact on glacier melt rates cannot be easily determined (Gilardoni et al., 2022).

One of the most common methods for reconstructing the mineral dust history of a location or region has been the use of a scanning electron microscope (SEM) with an energy dispersive x-ray spectrometry (EDS) detector to analyze single particles extracted from ice cores (Donarummo Jr et al., 2003; Nagatsuka et al., 2021; Wu et al., 2016). However, the process of classifying individual particles based on their EDS data is highly laborious and prone to error due to the complexity of certain minerals' elemental signatures. The adoption of numerous classification techniques is typically required to improve the certainty of the classifications before finalizing the results. Many have used the dichotomous key in Severin (2004) to compare EDS spectra to that of mineral standards, but this protocol requires access to a physical or digital copy of the book and cannot be automated. Others have used published sorting algorithms like the one by Donarummo Jr et al. (2003), but these algorithms are limited in the number of minerals that they can identify and, to the author's knowledge, each of these algorithms is only available in print and not as a functional computer program.

To satisfy the needs of earth and atmospheric scientists seeking to rapidly classify mineral particles from EDS data with a high degree of certainty, the `EDS Classification for MATLAB` software repository has been established. This library of functions provides users with the ability to work with SEM-EDS data in the MATLAB programming language without difficulty.

Methods

Mineral Classification

The function library introduces a novel machine learning-based mineral classification model (the "Weber algorithm") that has been trained with EDS data collected on 18 mineral standards and exhibits an overall accuracy exceeding 99% (see **Supplemental Information**). The library also includes functions for three sorting algorithms that have been painstakingly transcribed from the literature into MATLAB code. This includes the aforementioned Donarummo algorithm (Donarummo Jr et al., 2003) which can be used to classify 16 unique aluminosilicate minerals based on EDS net intensity data, the Panta algorithm (Panta et al., 2023) which can be used to classify 17 unique minerals and 6 mineral groups based on EDS atom percent data, and the Kandler algorithm (Kandler et al., 2011) which can be used to categorized EDS atom percent data into broad mineral groups, classes, and refractive indexes.

The four mineral classification algorithms are available as individual functions (`weber_classification`, `donarummo_classifiation`, `panta_classifiation`, and `kandler_classification`) as well as packaged together in a single function (`eds_classification`). Each function has clearly written documentation that can be viewed using the MATLAB `help` or `doc` commands. The general syntax, however, is very basic:

```
minerals = eds_classifiation(data,Algorithm="Name")
```

where `data` is a table of EDS data and `"Name"` specifies the desired algorithm; i.e., `"Weber"` (default), `"Donarummo"`, `"Kandler"`, or `"Panta"`. The output `minerals` is a categorical vector containing the mineral assignments for each row in the input table.

Visualizing X-ray Energy Data

For individuals who want to compare the x-ray spectra of their single particle measurements to the standard spectra in Severin (2004), it will be necessary to visualize each energy spectrum along with labels for the major characteristic x-rays. Most EDS software save x-ray energy data in the EMSA (`.msa`) file format which is compiled differently depending on the software and therefore makes it difficult to simply import the data using a base MATLAB function such as `readtable`. While methods for visualizing EDS spectra are available for the Julia programming language (Ritchie, 2022) and as a desktop Java application (see **DTSA-II**), there are currently no simple alternatives to perform these tasks in MATLAB.

The `EDS Classification for MATLAB` software repository introduces straightforward functionality for importing `.msa` files into MATLAB and visualizing the EDS spectrum contained in a file. Just use the `xray_plot` function to view the spectrum and the `xray_peak_label` function to label the characteristic x-rays with the most likely element.

```
plt = xray_plot('energy_spectrum.msa');
xray_peak_label(plt)
```

Users may also import x-ray energy data directly into the MATLAB workspace along with the corresponding metadata using `read_msa`.

```
[data,metadata] = read_msa('filename.msa');
```

Extract metadata from SEM images

Images collected with SEM software are usually saved as 16-bit TIFs and contain valuable metadata regarding the image itself as well as the operating conditions of the instrument at the time the image was taken. But while MATLAB users with access to the **Image Processing Toolbox** can typically view image metadata with the `imfinfo` function, the metadata stored in an SEM image are not so easily accessed. This is because the image metadata become compressed into a single character vector that is nested within several structure objects. Many

80 string operations or regular expressions are therefore needed to access the relevant metadata.
81 To combat this, a function called `get_sem_metadata` has been written to automatically extract
82 the useful information from a SEM image.

```
md = get_sem_metadata('filename.tif');
```

Acknowledgements

Copyrights

References

97 Donarummo Jr, J., Ram, M., & Stoermer, E. F. (2003). Possible deposit of soil dust from
98 the 1930's US dust bowl identified in Greenland ice. *Geophysical Research Letters*, *30*(6).
99 https://doi.org/10.1029/2002GL016641

100 Gieré, R. (Ed.). (2010). Atmospheric Particles. *Elements*, *6*(4). https://www.
101 elementsmagazine.org/atmospheric-particles/

102 Gilardoni, S., Di Mauro, B., & Bonasoni, P. (2022). Black carbon, organic carbon, and mineral
103 dust in South American tropical glaciers: A review. *Global and Planetary Change*, *213*,
104 103837. https://doi.org/10.1016/j.gloplacha.2022.103837

105 Kandler, K., Lieke, K., Benker, N., Emmel, C., Küpper, M., Müller-Ebert, D., Ebert, M.,
106 Scheuvens, D., Schladitz, A., Schütz, L., & others. (2011). Electron microscopy of particles
107 collected at Praia, Cape Verde, during the Saharan Mineral Dust Experiment: Particle
108 chemistry, shape, mixing state and complex refractive index. *Tellus B: Chemical and
109 Physical Meteorology*, *63*(4), 475–496. https://doi.org/10.1111/j.1600-0889.2011.00550.x

110 Nagatsuka, N., Goto-Azuma, K., Tsushima, A., Fujita, K., Matoba, S., Onuma, Y., Dallmayr,
111 R., Kadota, M., Hirabayashi, M., Ogata, J., & others. (2021). Variations in mineralogy of
112 dust in an ice core obtained from northwestern Greenland over the past 100 years. *Climate
113 of the Past*, *17*(3), 1341–1362. https://doi.org/10.5194/cp-17-1341-2021

114 Panta, A., Kandler, K., Alastuey, A., González-Flórez, C., González-Romero, A., Klose, M.,
115 Querol, X., Reche, C., Yus-Díez, J., & Pérez García-Pando, C. (2023). Insights into the
116 single particle composition, size, mixing state and aspect ratio of freshly emitted mineral dust
117 from field measurements in the Moroccan Sahara using electron microscopy. *Atmospheric
118 Chemistry and Physics*, *23*(6), 3861–3885. https://doi.org/10.5194/acp-23-3861-2023

119 Ritchie, N. W. M. (2022). Reproducible Spectrum and Hyperspectrum Data Analysis Us-
120 ing NeXL. *Microscopy and Microanalysis*, *28*(2), 478–495. https://doi.org/10.1017/
121 S143192762200023X

Severin, K. P. (2004). *Energy Dispersive Spectrometry of Common Rock Forming Minerals*. Kluwer Academic Publishers. https://doi.org/10.1007/978-1-4020-2841-0

Wu, G., Zhang, X., Zhang, C., & Xu, T. (2016). Mineralogical and morphological properties of individual dust particles in ice cores from the Tibetan Plateau. *Journal of Glaciology*, *62*(231), 46–53. https://doi.org/10.1017/jog.2016.8