Exploration of hyperspectral datasets with unsupervised learning techniques

Beatrice Baschetti[1,2]

[1] *Department of Geosciences, University of Padova, Padova, Italy;*

[2] *INAF-IAPS, Rome, Italy.*

Geology & Planetary Mapping Winter School,
22-26 January 2024

GMAP Geological Mapping

europlanet 2024 Research Infrastructure

Machine Learning systems

The system learns with…

- **Labelled** data ⟶ **Supervised Learning**
- **Unlabelled** data ⟶ **Unsupervised Learning**
- **Partially labelled** data ⟶ **Semisupervised Learning**
- Data in a **dynamic** environment, performing actions with rewards and penalties, in order to get the most reward over time ⟶ **Reinforcement Learning**

Unsupervised learning, why?

In many real-world cases data is unlabelled!

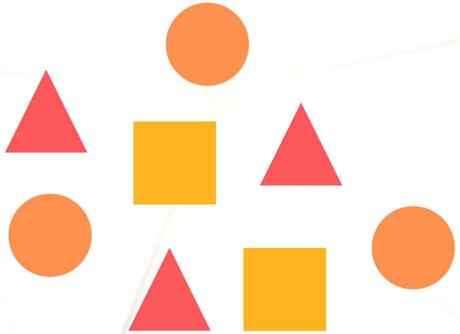
For **example**:

- Unlabelled images (wildlife photographs, surveillance cameras, satellite images, medical images…)
- Unlabelled text data (text from websites, books, articles, social media…)
- Financial data (transaction records, stock prices…)
- Dataset from a new instrument or sensors

(And collecting and labelling data is also a very hard job…)

Unsupervised learning, why?

On unlabelled data you can...



- Discover patterns
- Understand your data
- Detect outliers
- Select relevant features

Branches of unsupervised learning

- Clustering;
- Anomaly detection/novelty detection;
- Visualization and dimensionality reduction;
- Association.

Unsupervised learning - clustering

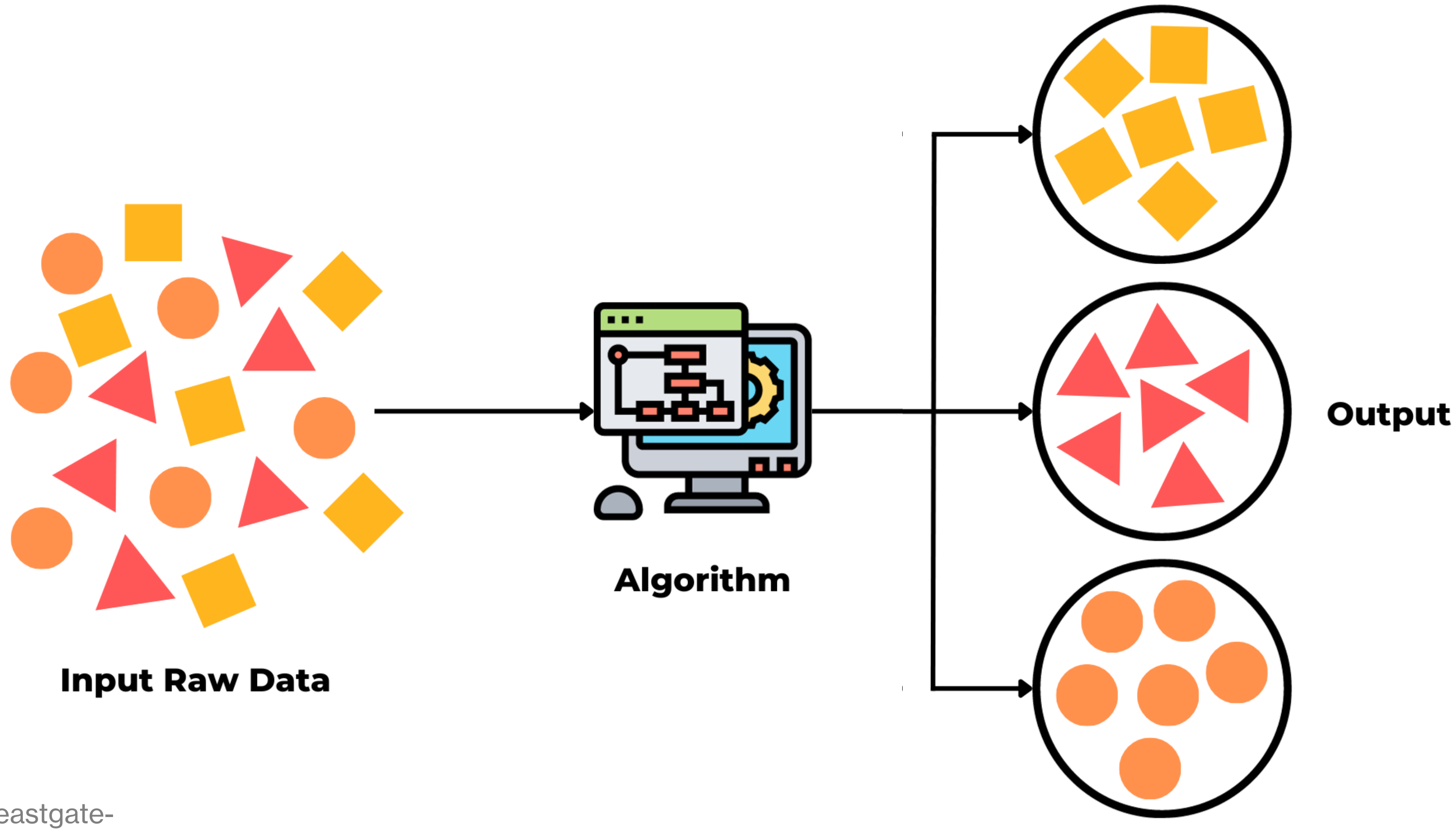


Image credits: https://eastgate-software.com/what-is-unsupervised-learning/

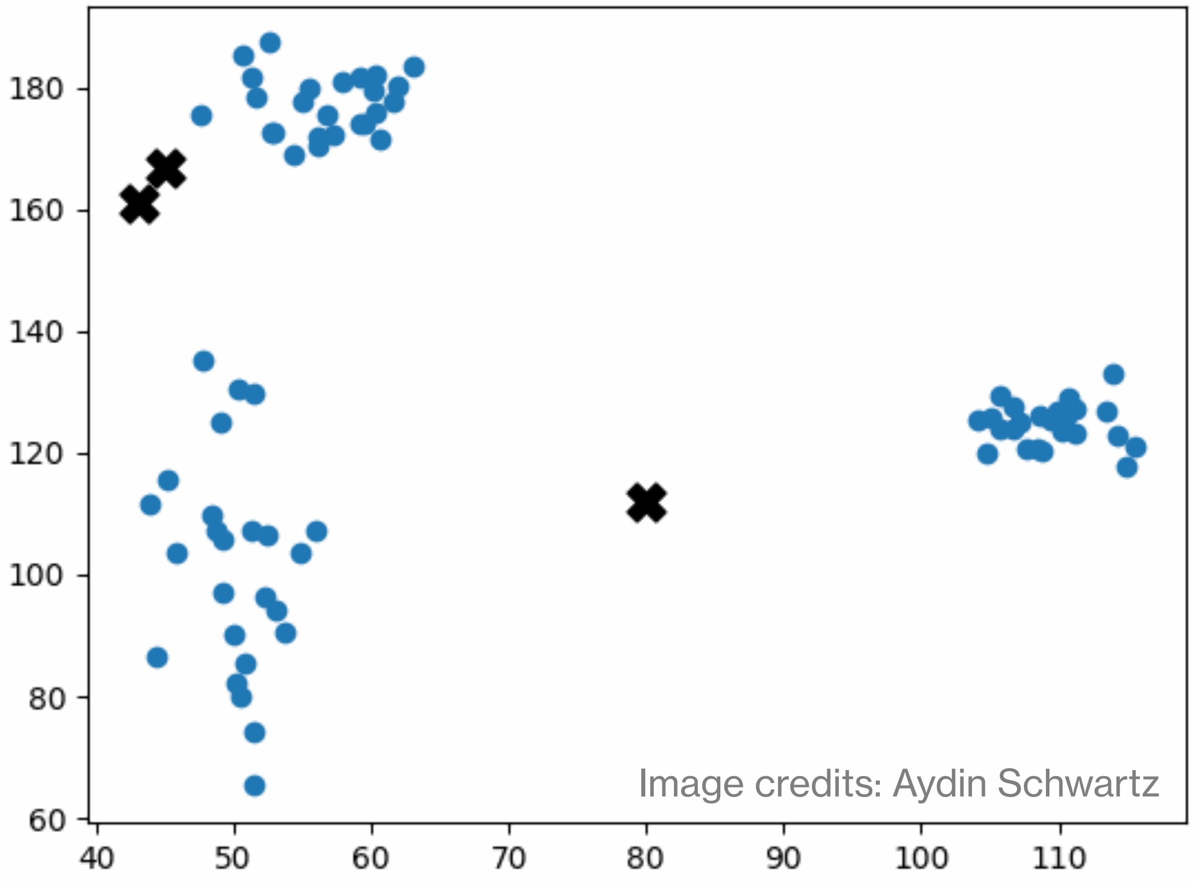
Unsupervised learning - clustering

Popular clustering algorithms

- K-Means
- Hierarchical clustering
- DBSCAN (Density-Based Spatial Clustering of Applications with Noise)
- Gaussian Mixture Models

Clustering algorithms – centroid based

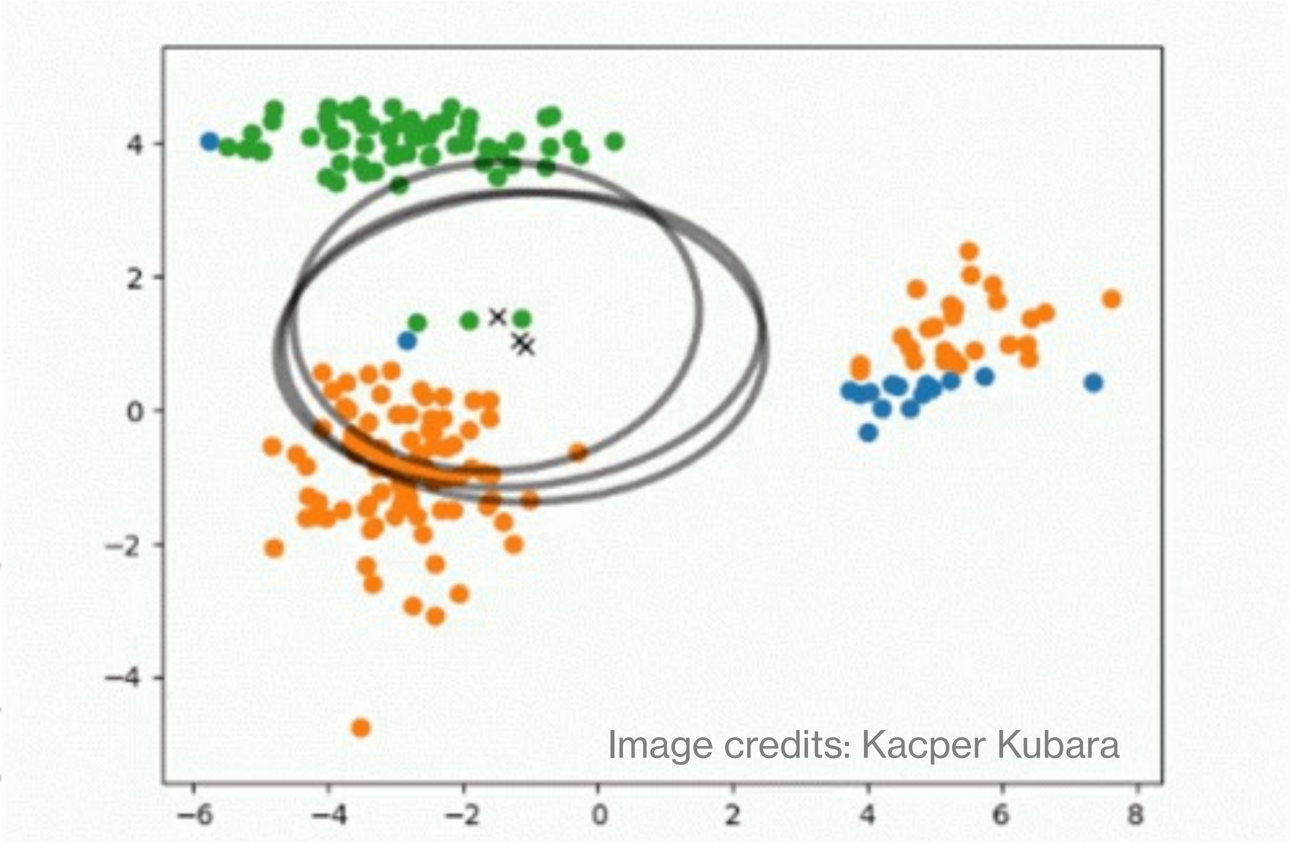
K-Means



- Very quick and efficient clustering
- Suitable for large datasets
- Numbers of clusters is a required input
- Not suitable for clusters with nonconvex shape or very different sizes
- Sensitive to noise and outliers

Clustering algorithms - probabilistic

Gaussian Mixture Models



- Can be quick and efficient clustering with some constraints
- Suitable for moderate-size datasets
- Numbers of clusters is a required input
- Can capture more complex cluster shapes than k-Means
- Sensitive to noise and outliers

Dimensionality reduction, always a good idea

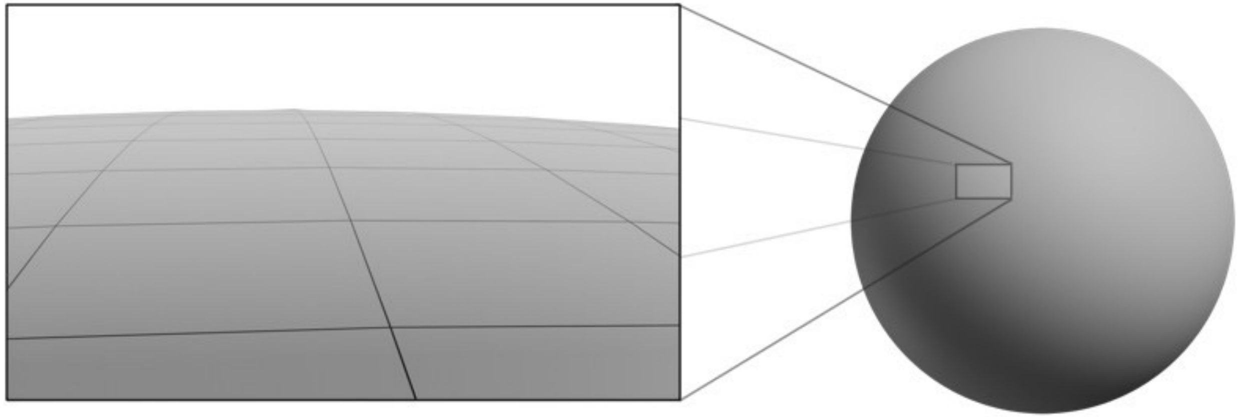
- Speeds up the clustering process, reducing its computational complexity
- Preserves most relevant information
- Improves quality of clustering
- Helps with interpretation and visualization

Dimensionality reduction

Possible approaches

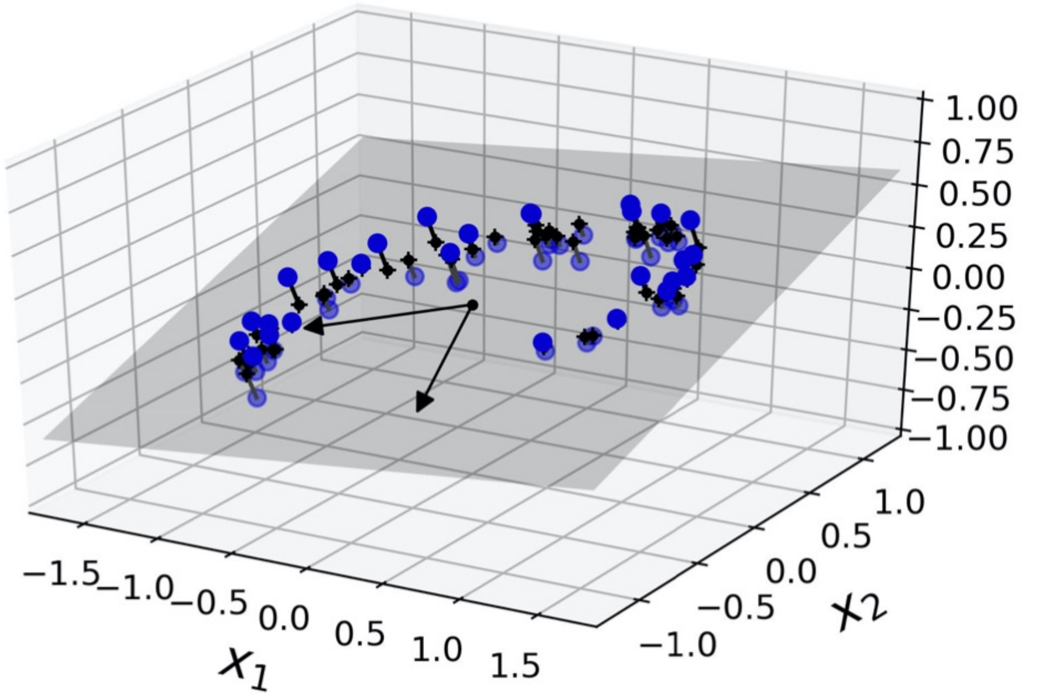
- **Projection**: assumes training instances lie close to a lower-dimensional subspace
- **Manifold learning**: assumes training instances lie close to a lower-dimensional *manifold**

*Manifolds are spaces that when observed locally, look like simple, familiar shapes such as planes or curves, but when considered in their entirety, might have complex and non-linear structures.



Dimensionality reduction

Projection



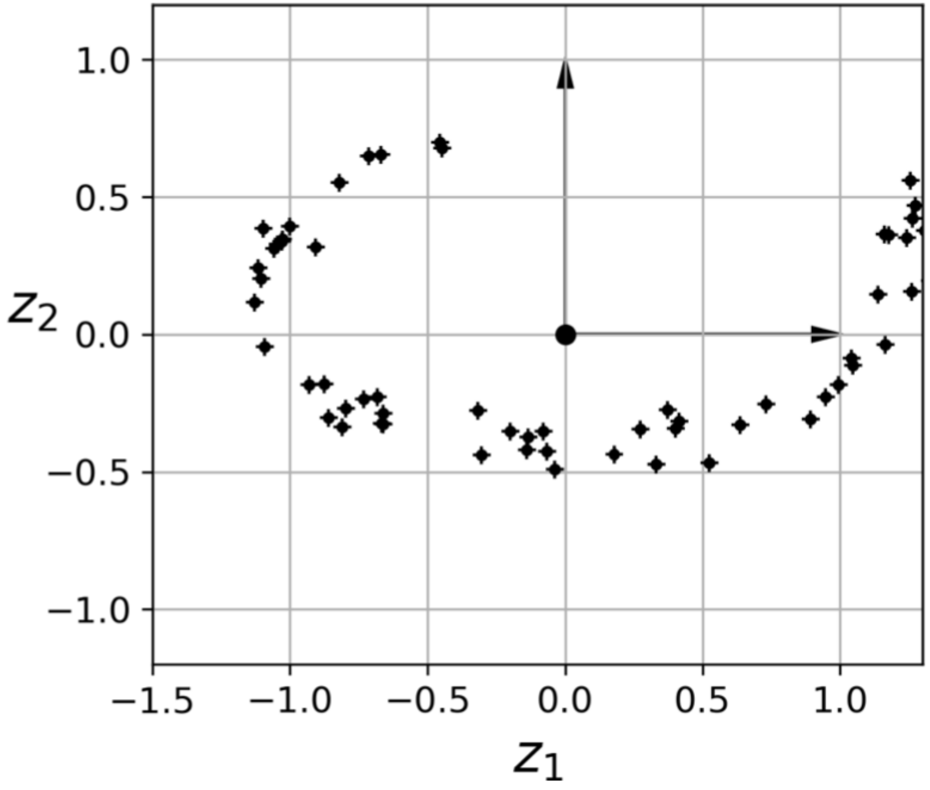
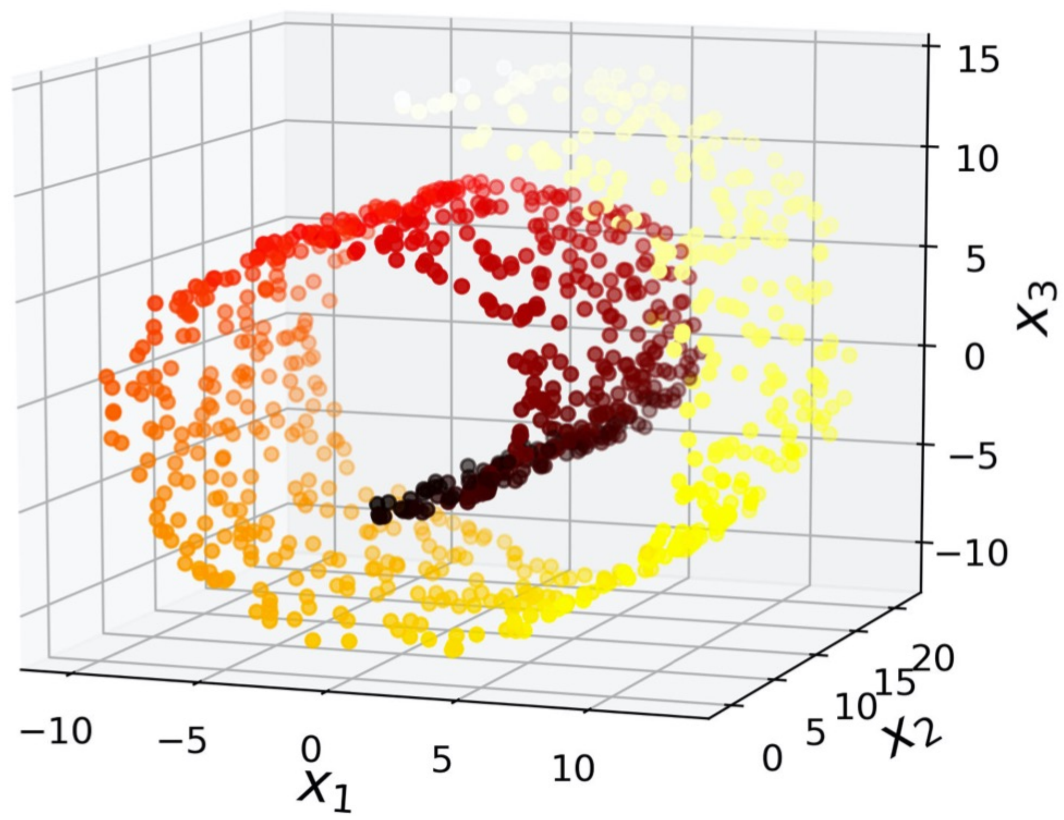


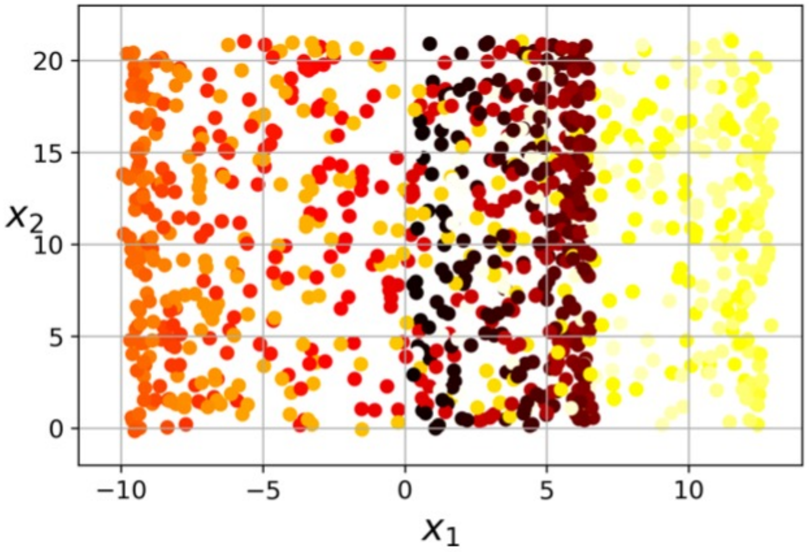
Image credits: Aurélien Géron

Dimensionality reduction



→

Projection



Manifold learning

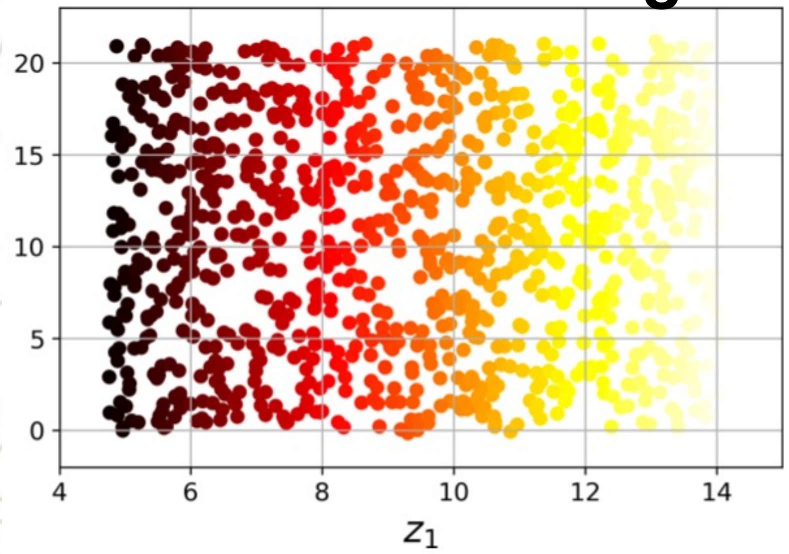


Image credits: Aurélien Géron

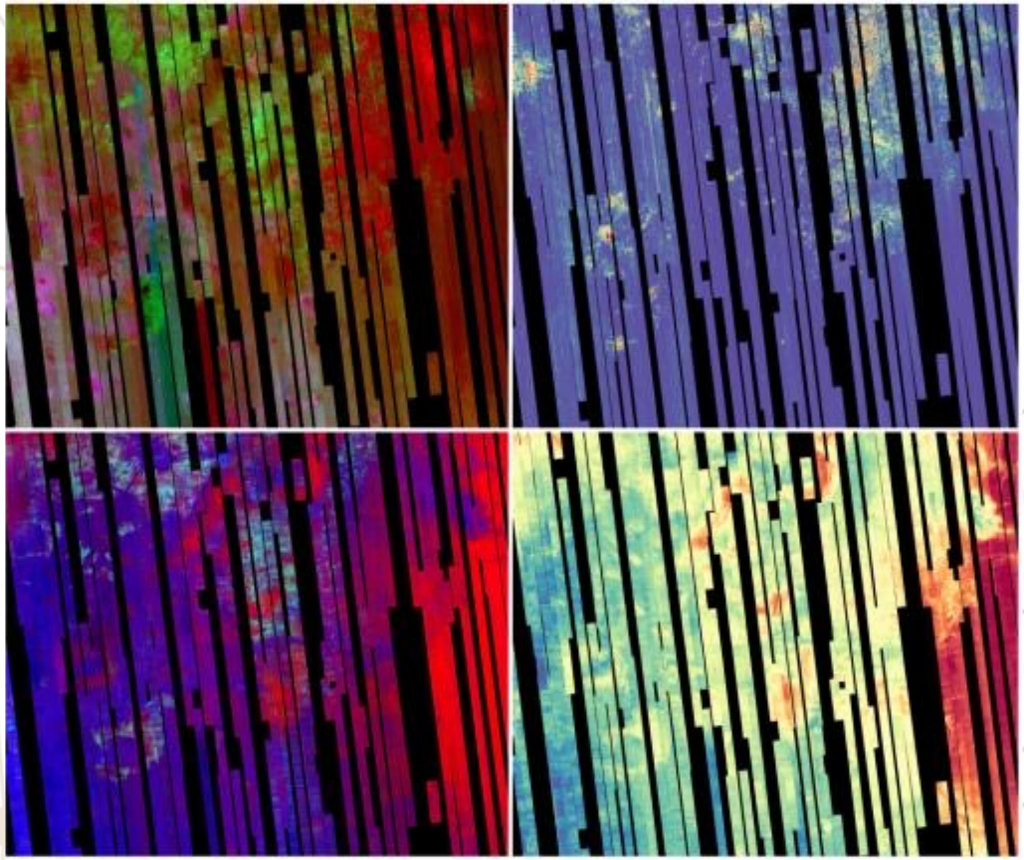
Dimensionality reduction

Possible approaches

- **Projection:** Principal Component Analysis – PCA (most used)
- **Manifold learning**: Locally Linear Embedding (LLE), t-distributed Stochastic Neighbour Embedding (t-SNE), Uniform Manifold Approximation and Projection (UMAP)

Hyperspectral datasets

- Large datasets
- Have a lot of features
- Non-linear and complex structures
- Usually noisy and have artefacts



Credits: NASA/JPL-Caltech/JHU-APL

Hyperspectral datasets

→ Choose clustering algorithms that can handle big datasets.

→ Use dimensionality reduction (in particular, try manifold learning).

→ Filter your data (for artefacts, noise and other irrelevant features).

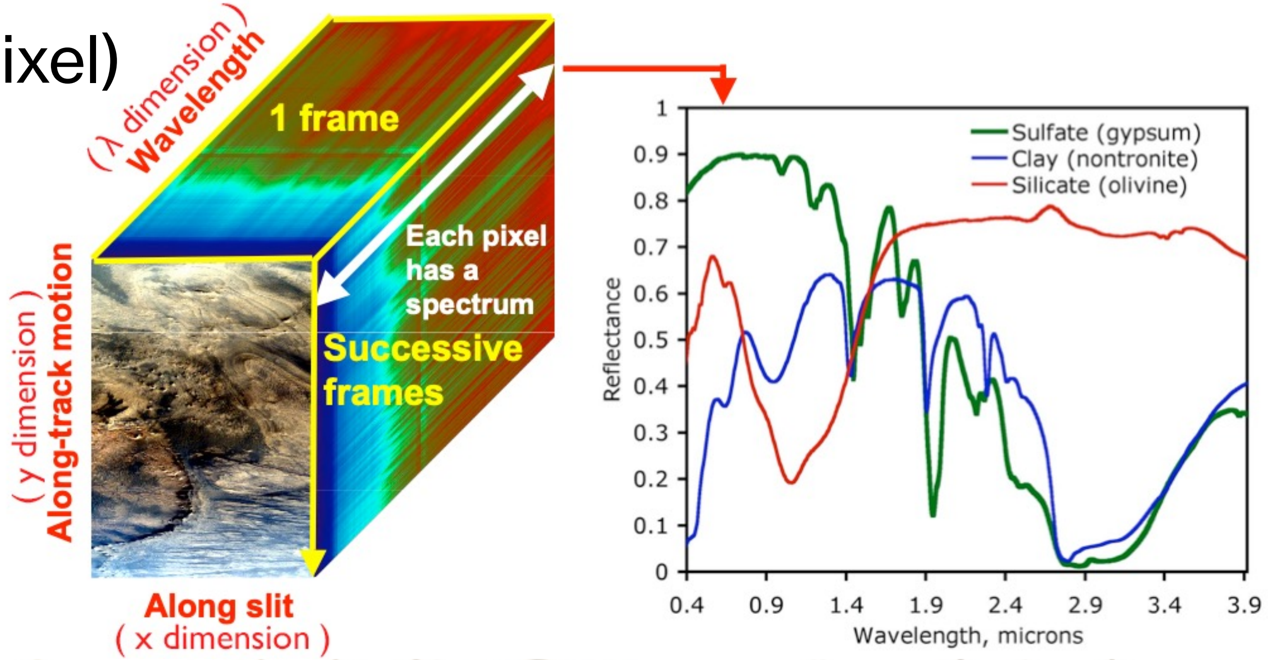
CRISM data 

CRISM: Compact Reconnaissance Imaging Spectrometer for Mars

- Hyperspectral visible-infrared spectrometer
  - 0.4-4 microns range
  - 544 spectral channels

  high spectral resolution
  - high spatial resolution (up to 18 m/pixel)

~1 GB

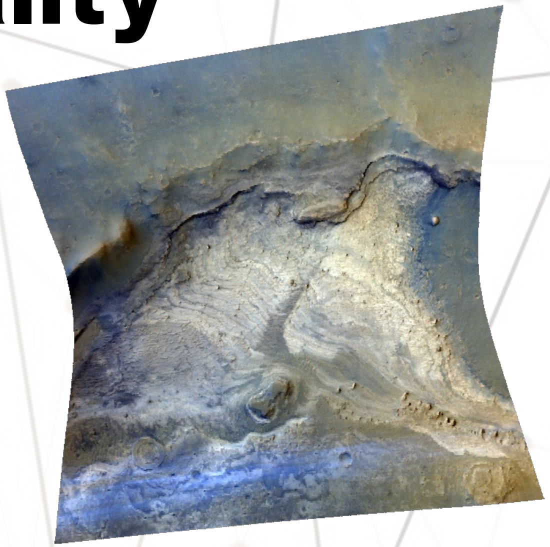


Clustering CRISM data: filtering and flattening

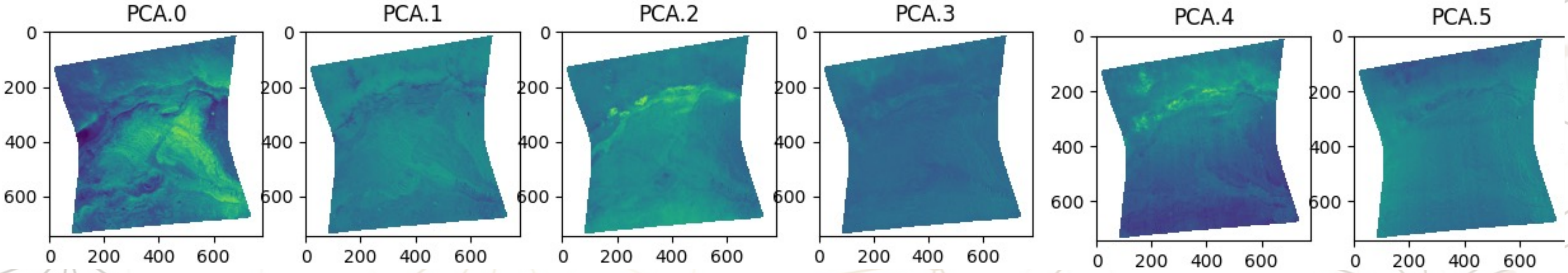
- Remove invalid pixels from your data (borders of the image)
- Choose a wavelength range based on your needs: e.g., 1.0-2.6 microns for primary and secondary minerals…
- Filter the data for artefacts: e.g.,1.645-1.704 um (known filter boundary artefact), 1.948-2.060 um (CO_2 artefact)
- Flatten the spatial dimension of your data $\underset{\text{3D}}{(n, m, \lambda)} \longrightarrow \underset{\text{2D}}{(n \times m, \lambda)}$

$\longrightarrow$ Example of shape of data: (581064, 211)

Clustering CRISM data: dimensionality reduction

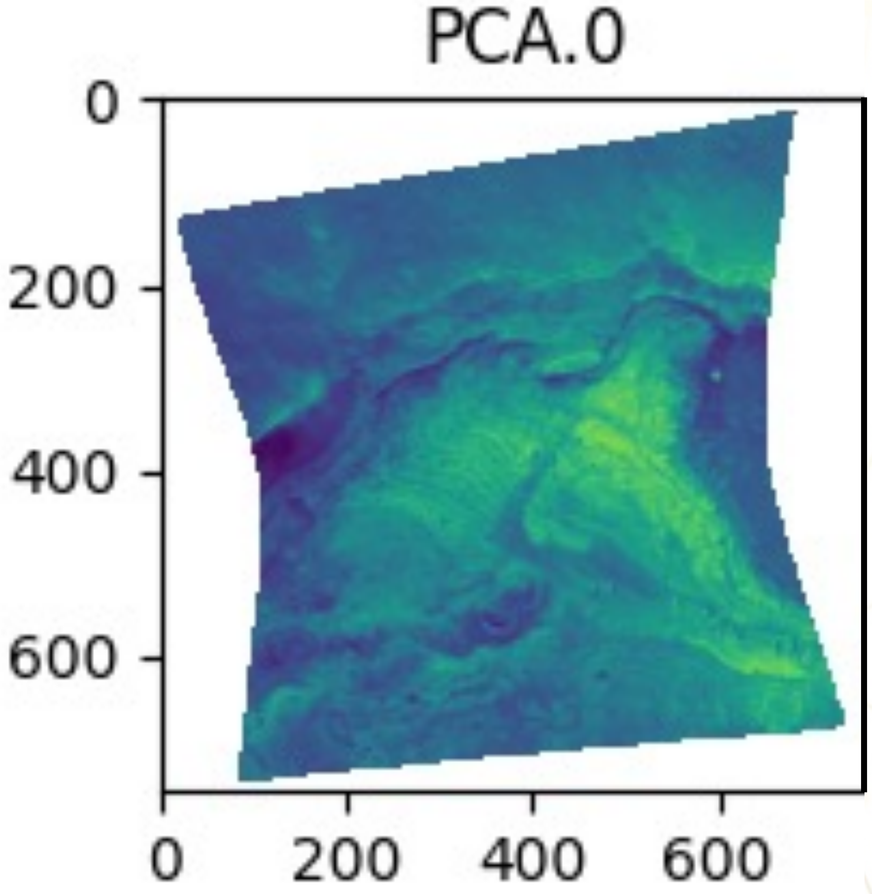


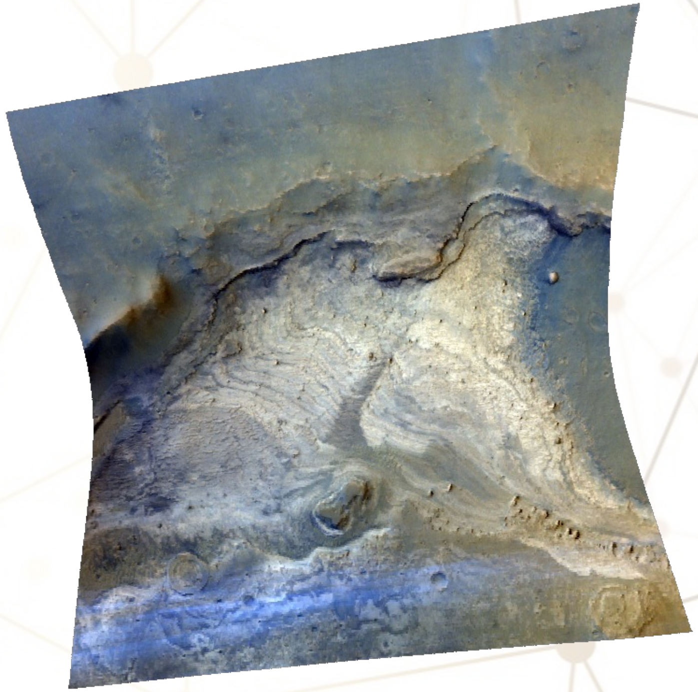
Principal Component Analysis, first 6 components:



From (581064, 211) → to (581064, 6) array

Clustering CRISM data: dimensionality reduction





First component represents the most variance in your data…

…It looks like here it mostly reflects the average reflectance level of the image.

Is this something we want to include in our cluster analysis?

(581064, 6-1) -> (581064, 5) array

Clustering CRISM data: dimensionality reduction

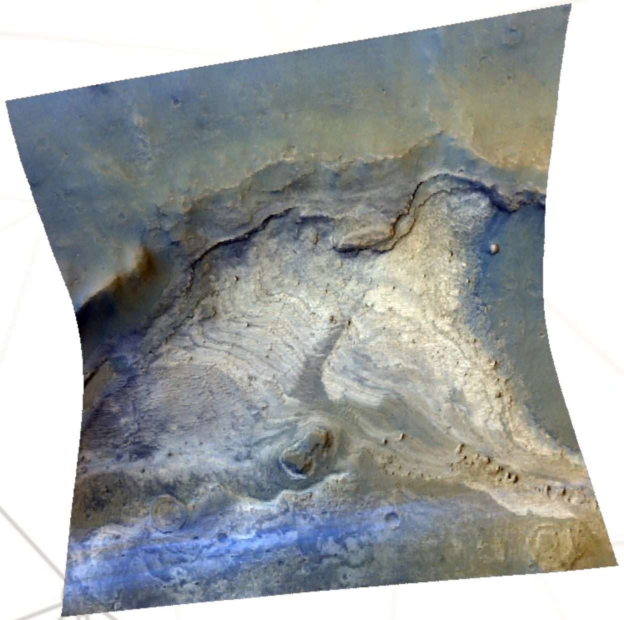
PCA + Manifold learning (e.g., UMAP)

(581064, 5) -> (581064, 2) array!

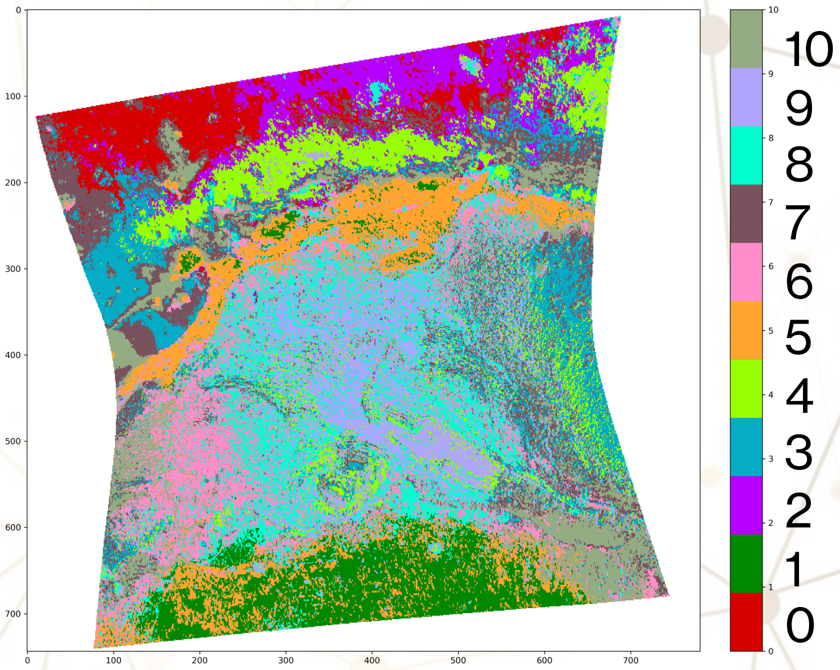
https://umap-learn.readthedocs.io/en/latest/
https://pair-code.github.io/understanding-umap/

Clustering CRISM data: applying clustering algorithm

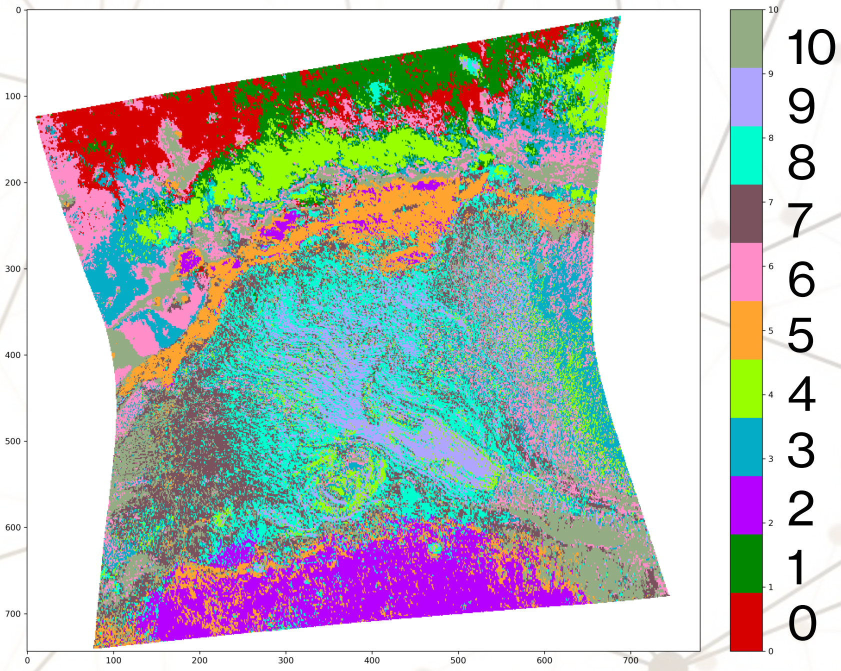
K-Means and Gaussian Mixture Model for 11 clusters



K-Means

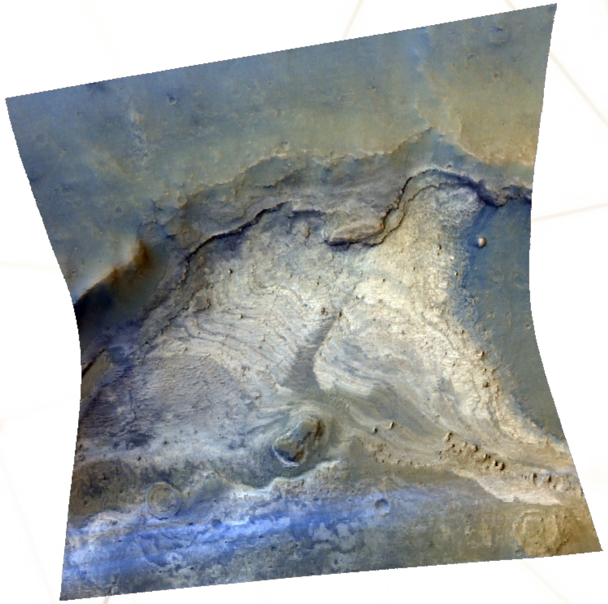


Gaussian Mixture Model (GMM)

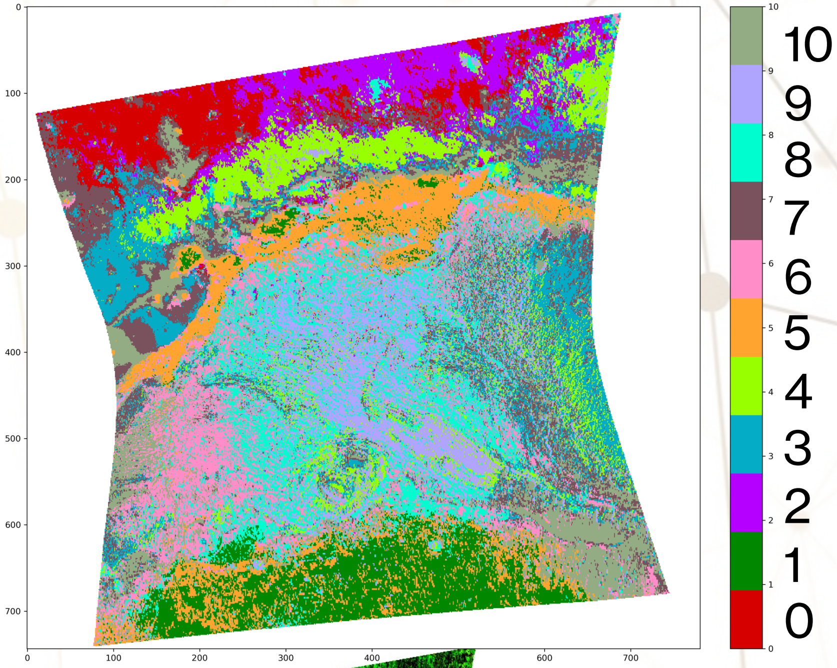


https://scikit-learn.org/stable/modules/clustering.html#clustering

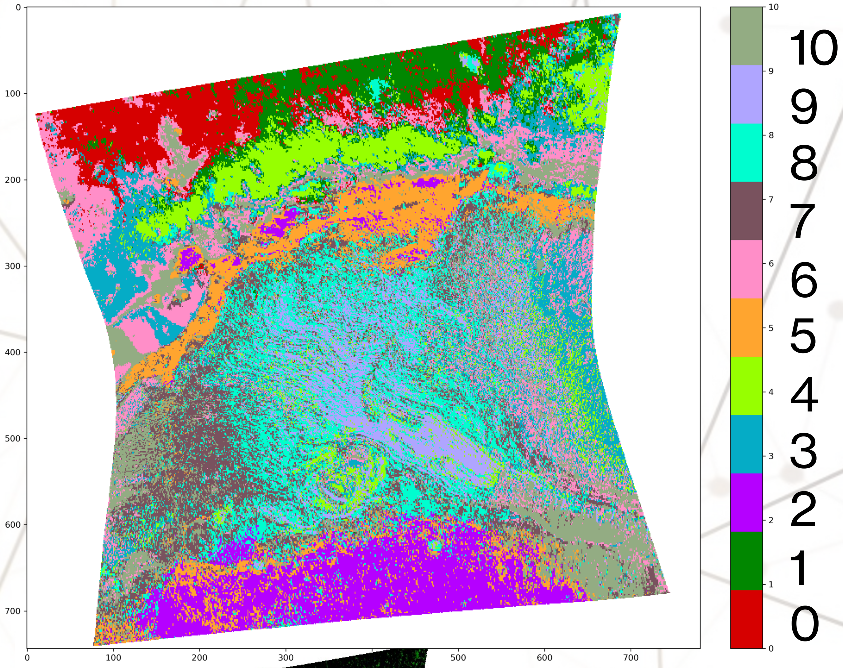
Clustering CRISM data: applying clustering algorithm

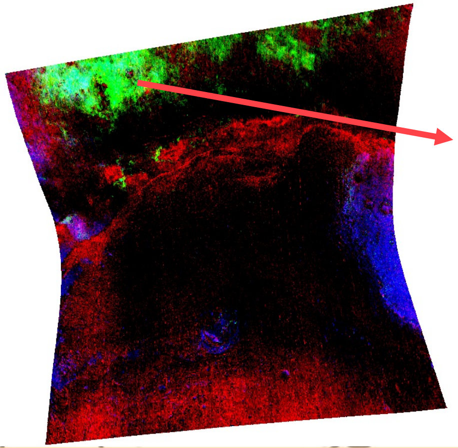


K-Means

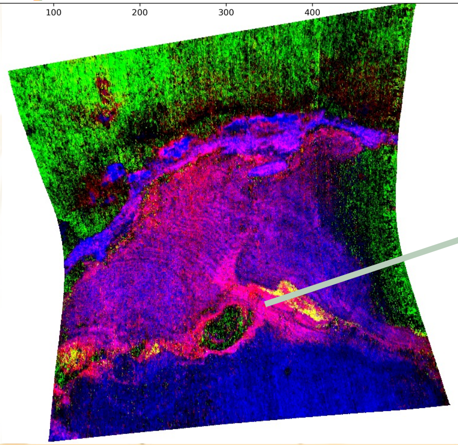


Gaussian Mixture Model (GMM)

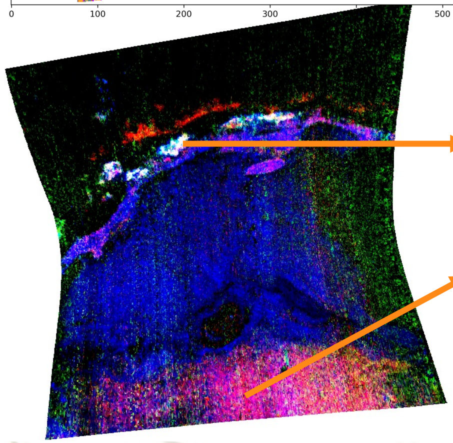




Pyroxenes

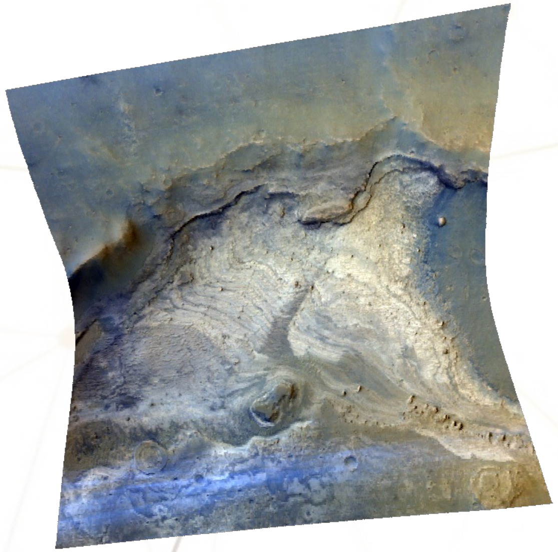


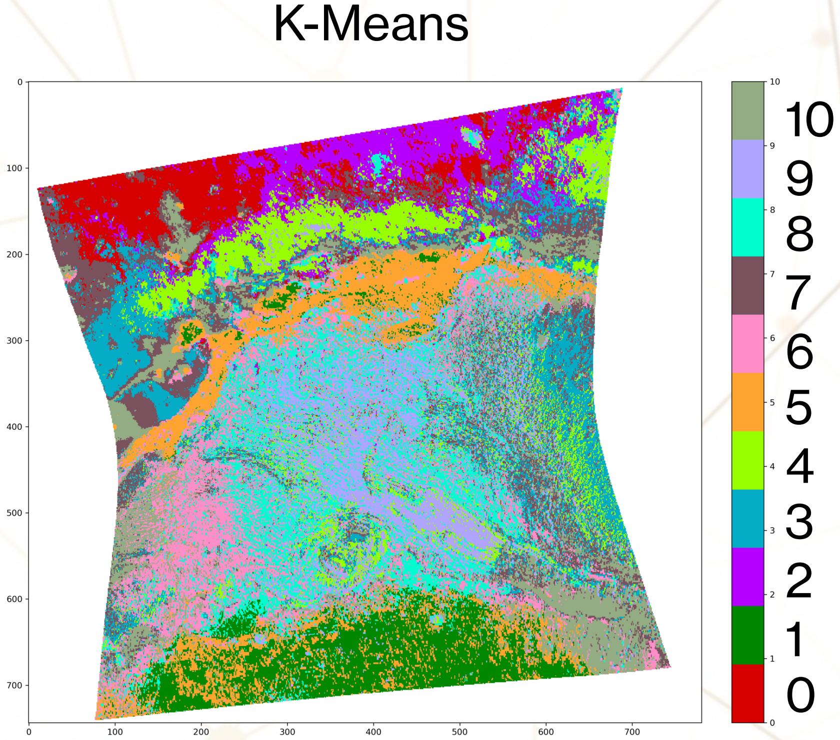
Sulfates (poly/mono)

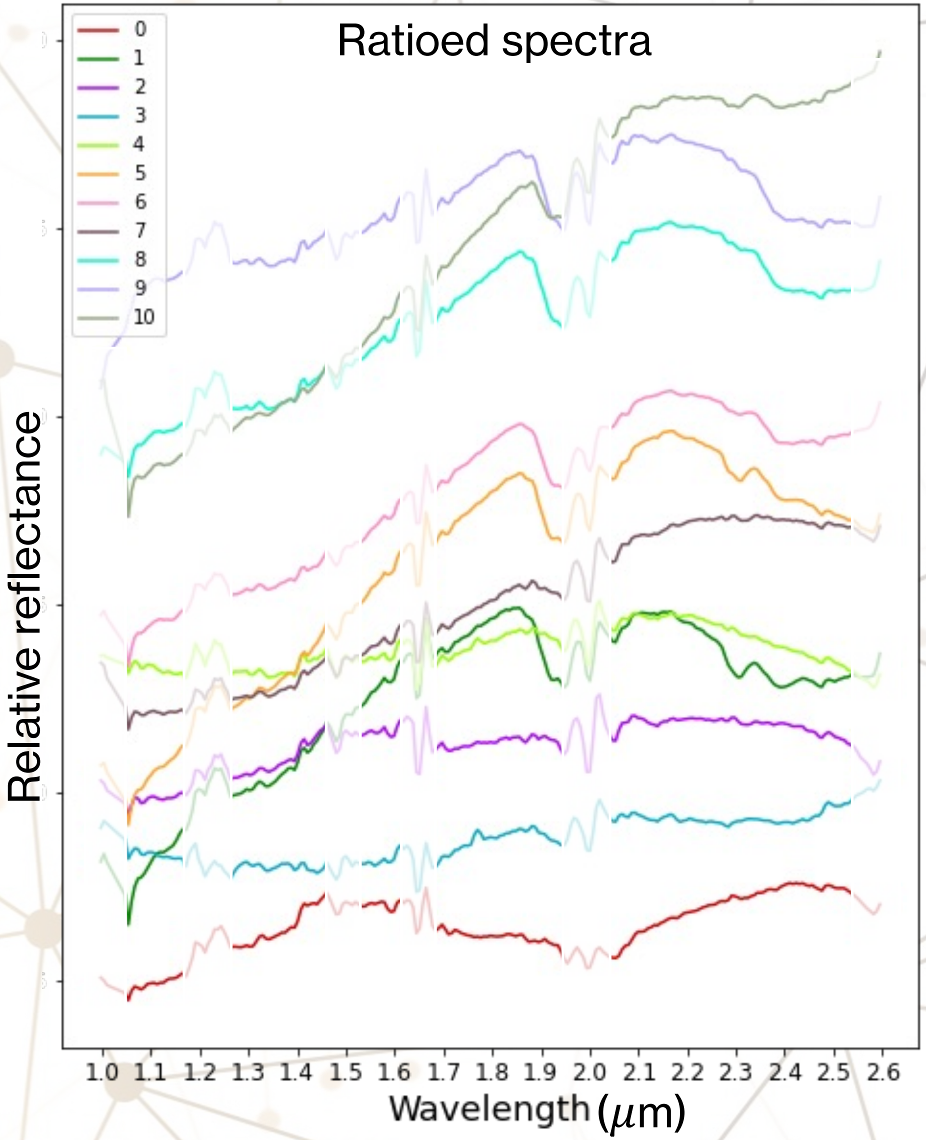


Fe/Mg phyllosilicates

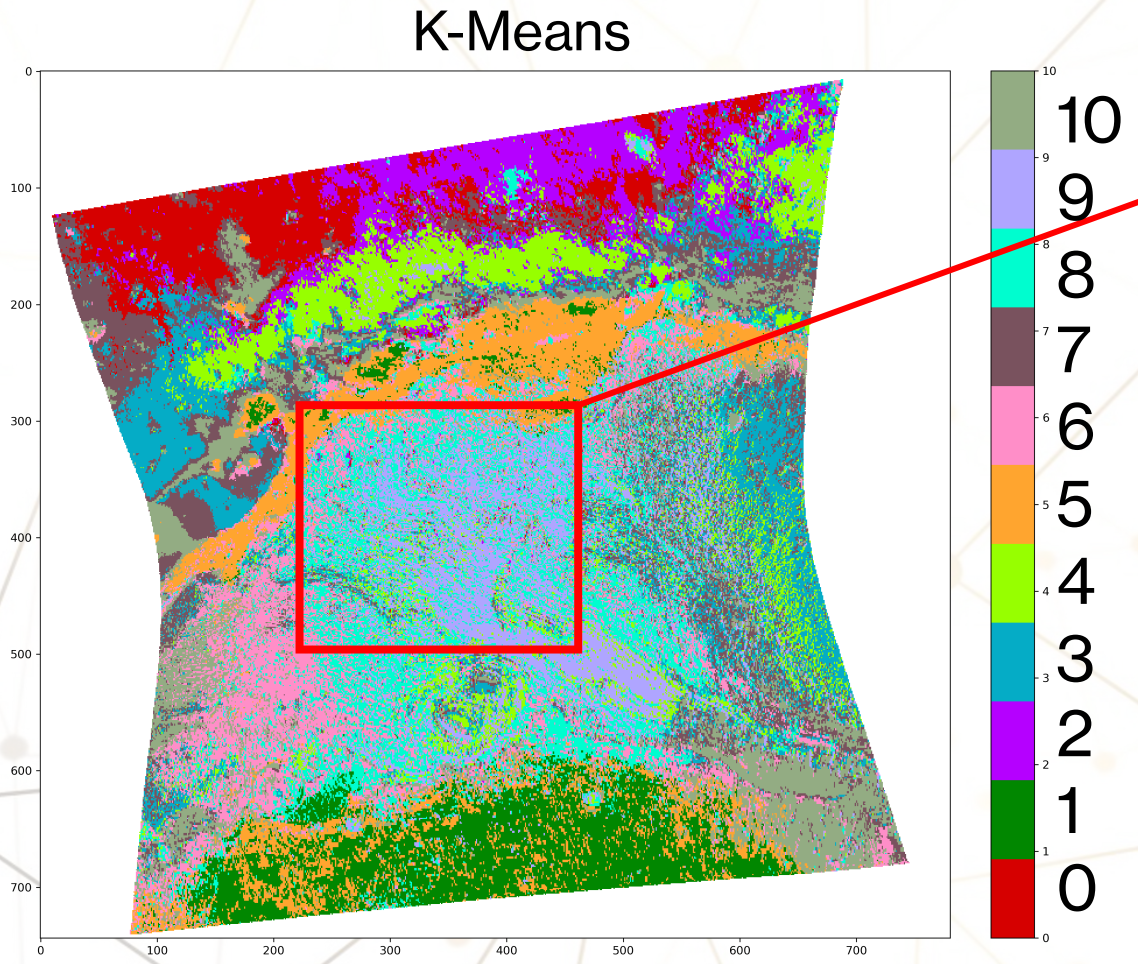
Clustering CRISM data: applying clustering algorithm



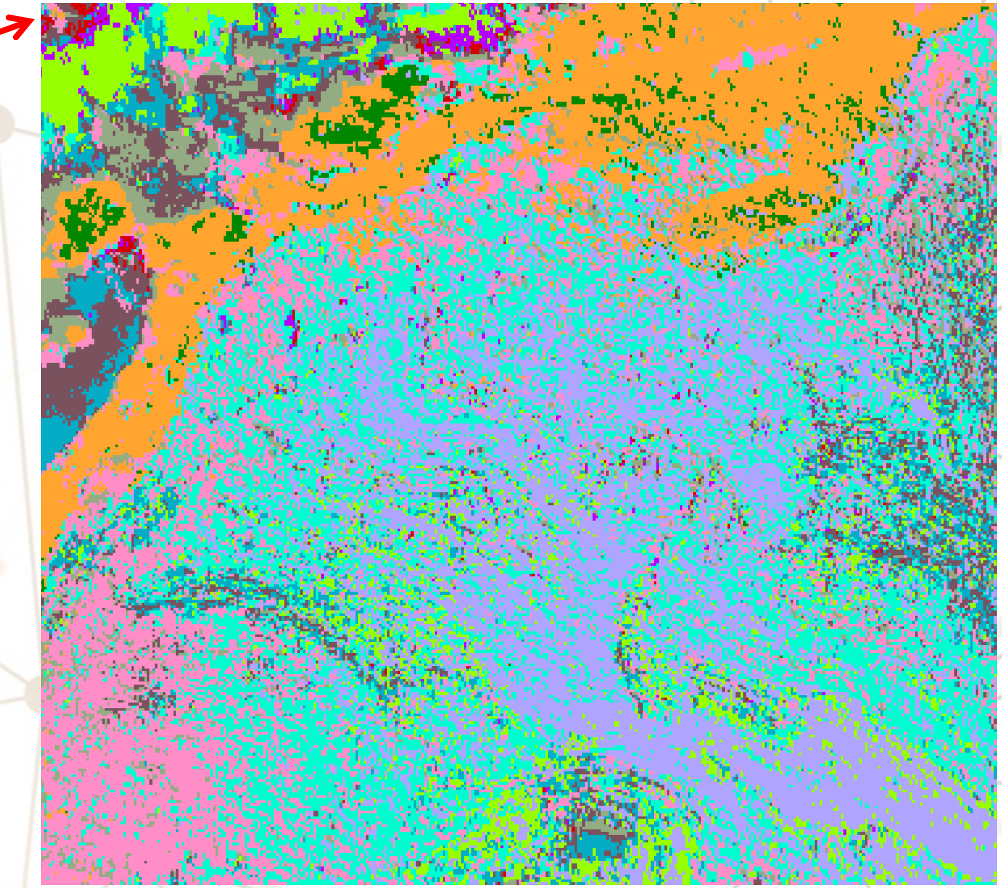




Clustering CRISM data: applying clustering algorithm



Layers of different composition!



Clustering CRISM data: finding right number of clusters

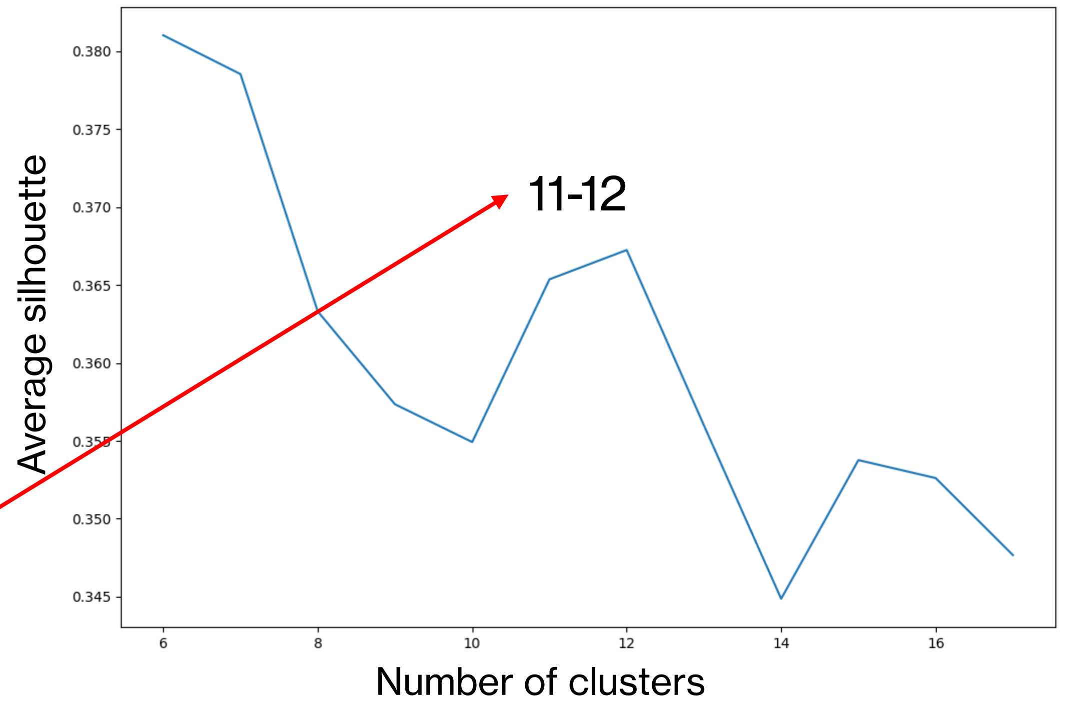
Silhouette criterium: measures how close each point in one cluster is to points in neighbouring clusters

Silhouette has a range of [-1, 1]

- Coefficients near +1 indicate good clustering
- Coefficients near -1 indicate incorrect clustering

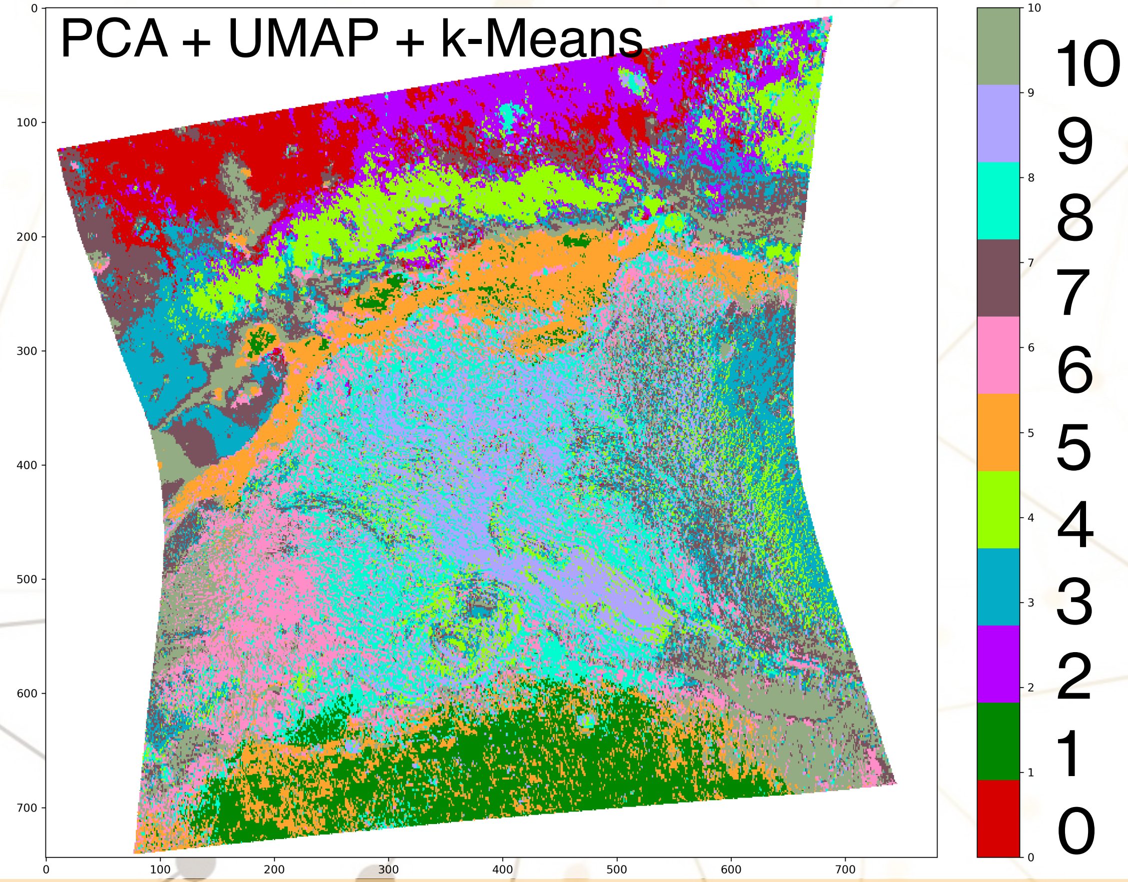
In this case, values around 11 should be optimal

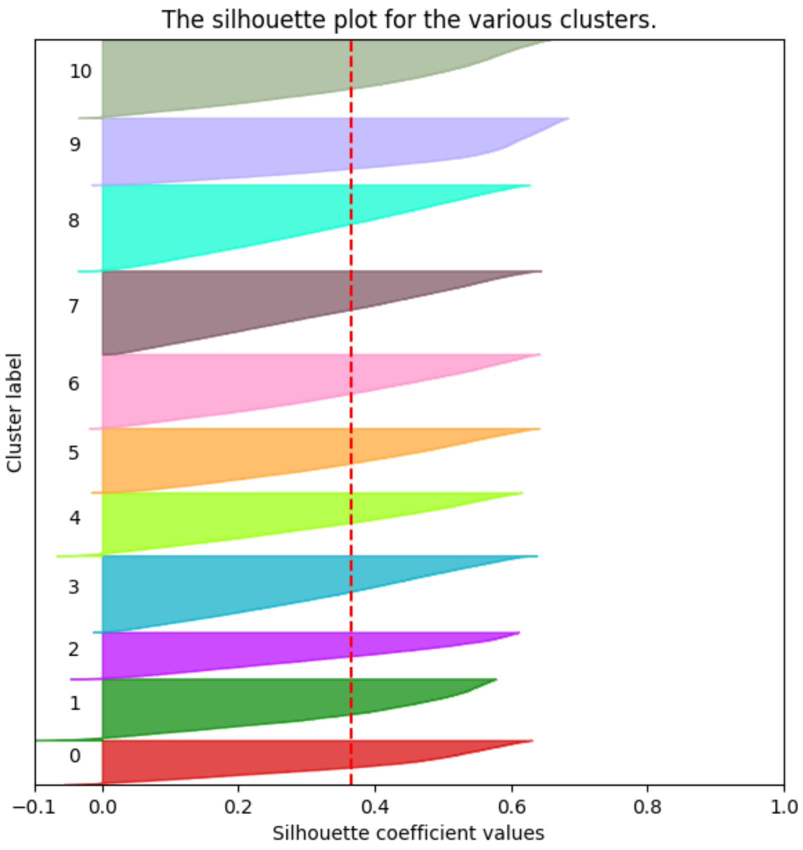
PCA + UMAP + k-means



Clustering CRISM data: evaluate quality of clusters

Silhouette criterium: example of good clustering

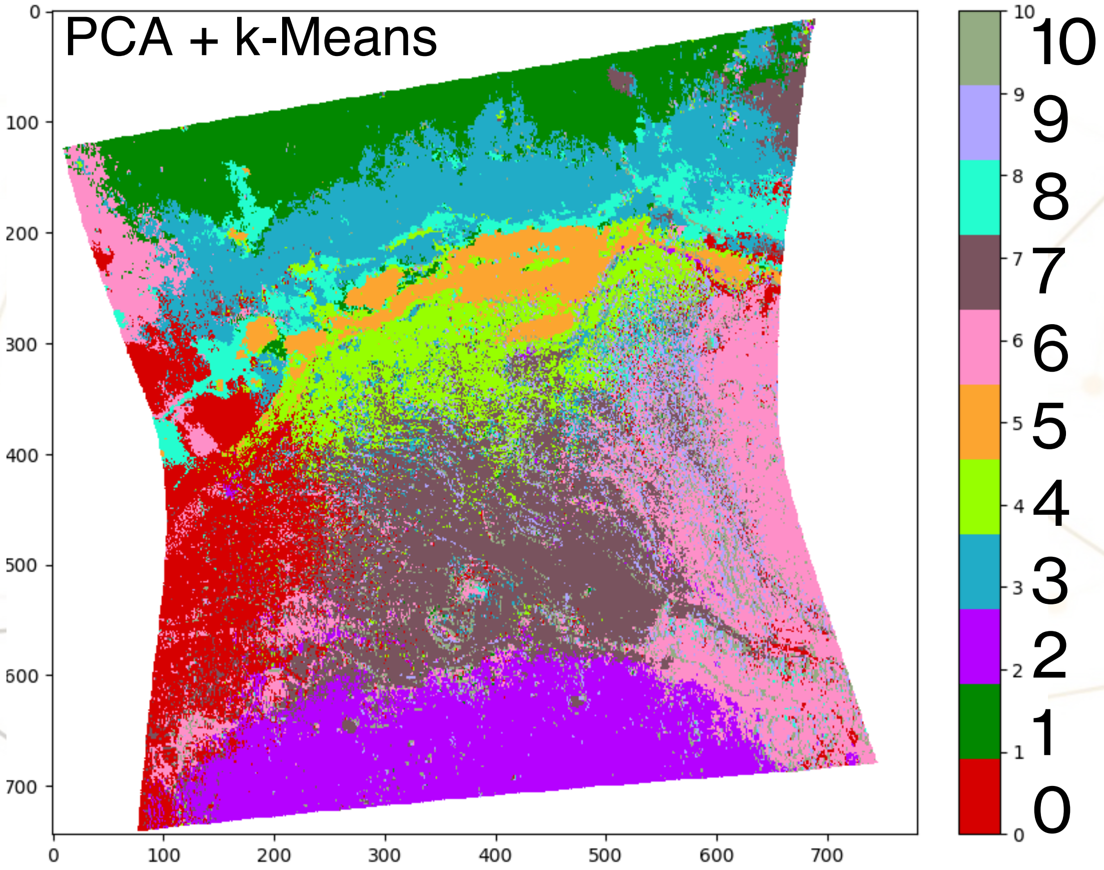


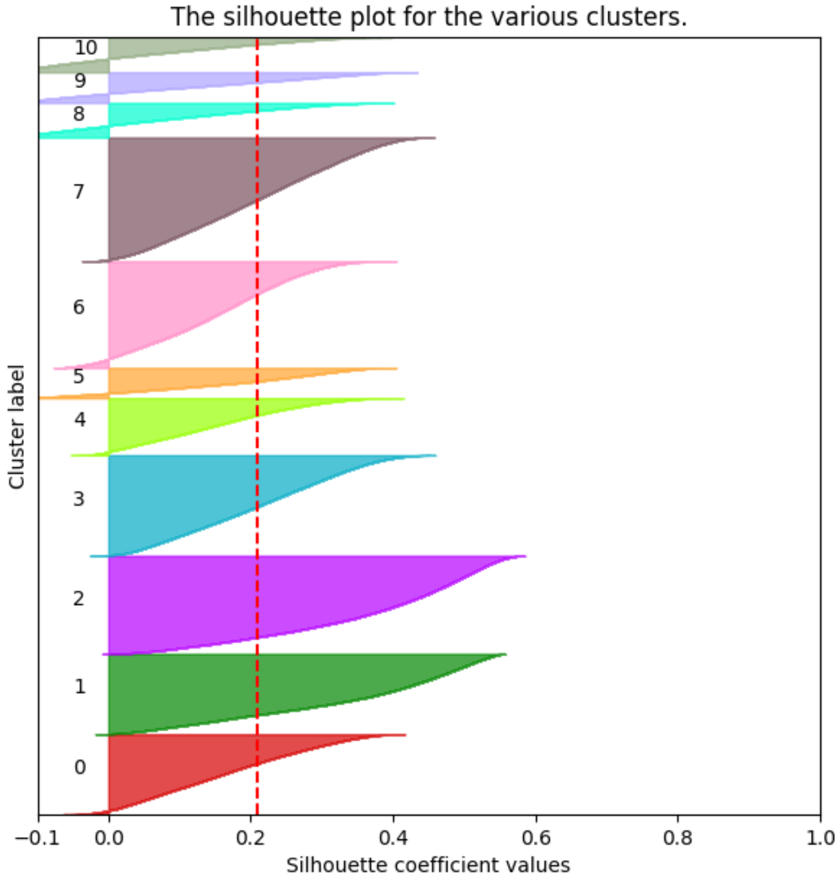


https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html

Clustering CRISM data: evaluate quality of clusters

Silhouette criterium: example of sub-optimal clustering





Clustering CRISM data: comparing models

Algorithm	Silhouette score
PCA + k-Means	0.211
PCA + GMM	0.184
PCA* + k-Means	0.209
PCA* + GMM	0.195
PCA* + UMAP + k-Means	0.365
PCA* + UMAP + GMM	0.357

Summary and conclusions

- Unsupervised learning is a powerful tool to explore unlabelled data, find patterns, detect outliers, reduce the dimensionality...
- k-Means and Gaussian Mixture Models can be applied to hyperspectral datasets (as on Mars, with CRISM data) with good results.
- Pre-processing and some interpretation of PCA components is important and can improve the performance of your algorithm.
- Silhouette score will help you find the most appropriate number of clusters, evaluate clustering quality and compare different models.
- Overall, clustering spectral data can be a very helpful complement to more traditional spectral analysis techniques.

Code availability and further readings

Code

https://github.com/beatricebs/CRISM-python-unsupervised-clustering

Books on ML

Jiawei Han, Micheline Kamber, Jian Pei, “Data Mining, concepts and techniques (Third Edition)” – 2012, Morgan Kaufmann Publishers (Elsevier)

Aurélien Géron, “Hands-on machine learning with Scikit-learn, Keras and Tensor Flow (Second Edition)” – 2019, O’Reilly Media

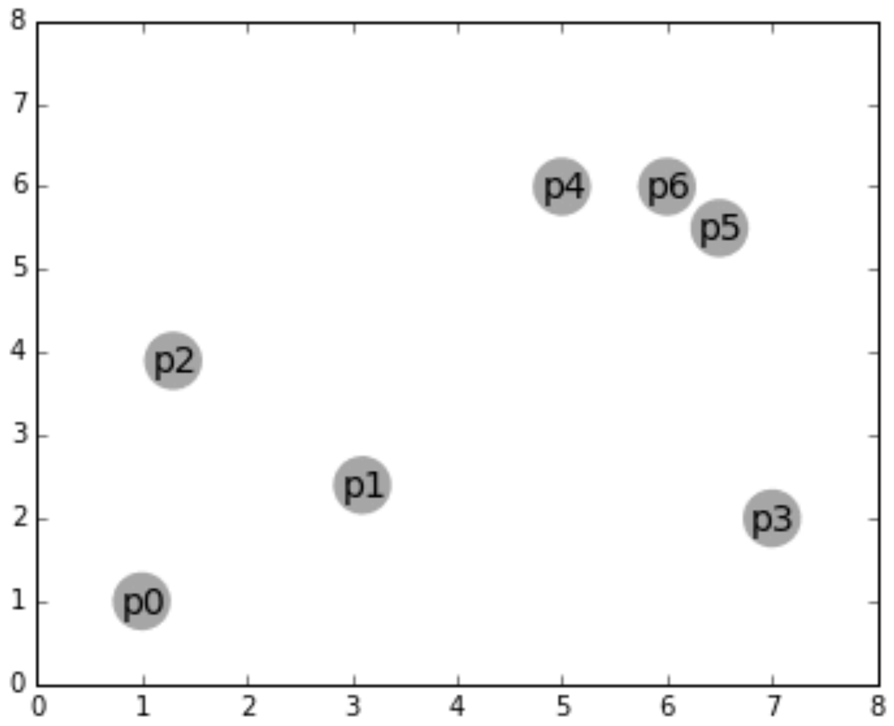
Andreas C. Muller and Sarah Guido, “Introduction to Machine Learning with Python” – 2017, O’Reilly Media

Thank you!

Contact: beatrice.baschetti@inaf.it

Clustering algorithms - hierarchical

Hierarchical clustering



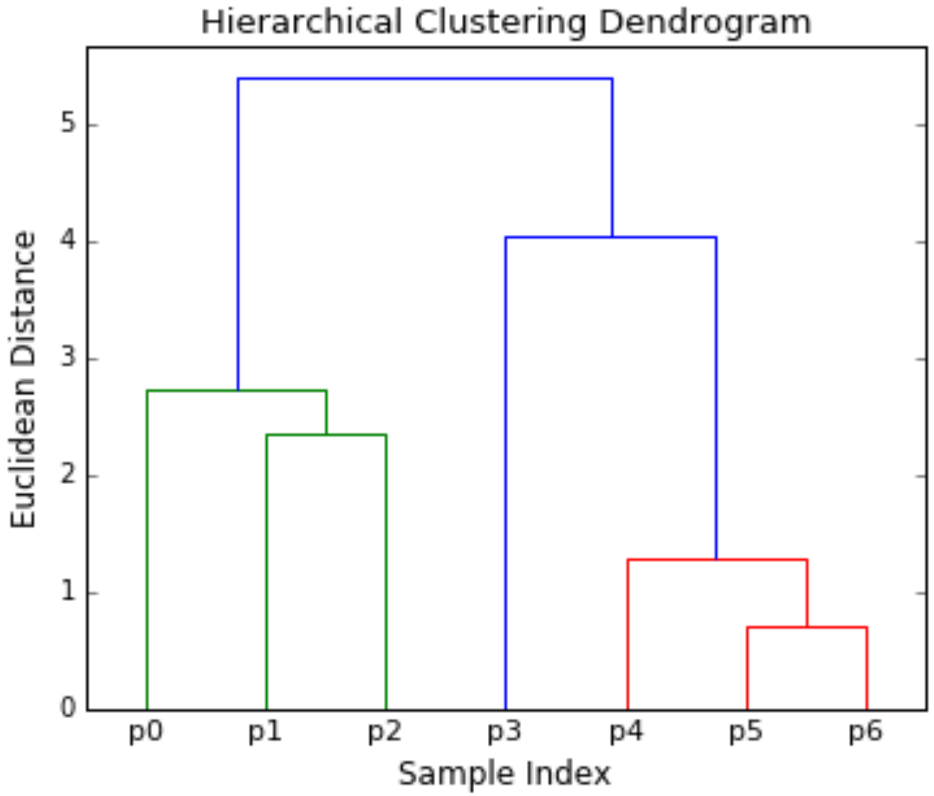


Image credits: David Sheehan

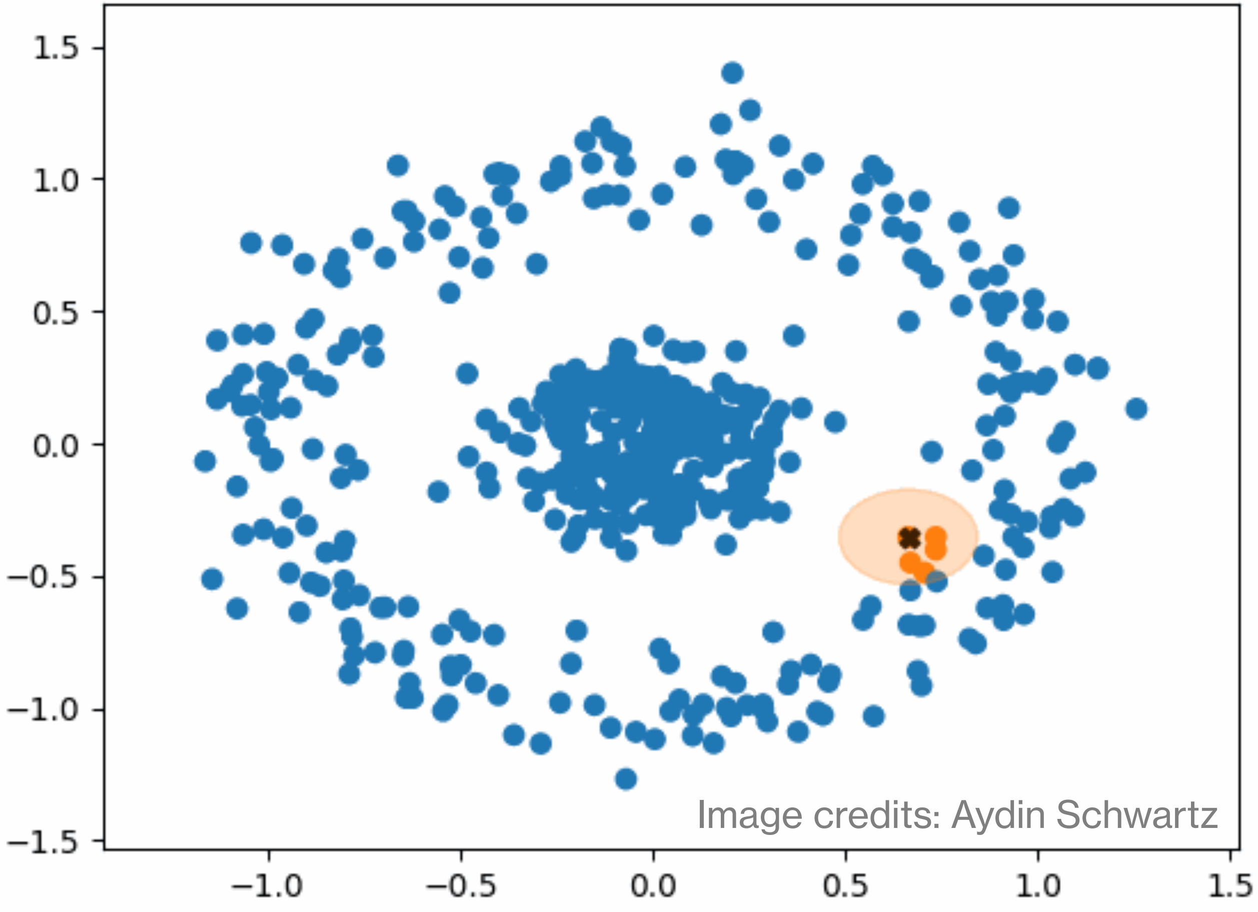
Clustering algorithms - hierarchical

Hierarchical clustering

- Intuitive and easily interpretable ☺
- No need to specify number of clusters ☺
- Not suitable for large datasets ☹
- Merge/split decisions highly influence quality of clusters ☹

Clustering algorithms – density based

DBSCAN



- Not really suitable for large datasets
- No need to specify number of clusters
- Effective in finding arbitrary-shaped clusters
- Robust to noise and outliers