# Applications of unsupervised learning to astronomical datasets

Dalya Baron, TAU

SOMACHINE 2021

April 2021

# What is unsupervised learning?

A general term that incorporates a large set of statistical tools, used to perform data exploration, such as clustering analysis, dimensionality reduction, visualization, and outlier detection.

# What is the difference between supervised and unsupervised learning?

# What is the difference between supervised and unsupervised learning?

Figure of merit

# What is the difference between supervised and unsupervised learning?

Figure of merit

## Supervised Learning:

**Input:** a list of objects with measured features and labels.

The algorithm is optimizing a score (=cost function) that depends on the input labels and the predicted ones.

**Prior knowledge is required!**

## Unsupervised Learning:

**Input:** a list of objects with measured features.

The algorithm detects clusters, outliers, or reduces the dimensions of the dataset.

**Prior knowledge is not(?) required!**

# Types of unsupervised learning

- **Clustering:** K-means, hierarchical clustering, Gaussian mixture models, etc.
- **Dimensionality reduction:**
  (I) PCA, ICA, and NMF.
  (II) tSNE, UMAP, and the Sequencer.
  (III) Self organizing maps and autoencoders.
- **Outlier detection:**
  (I) Isolation Forests.
  (II) Using supervised learning (+ one class SVM).
  (III) Using unsupervised learning.

For more information see Baron (2019): https://arxiv.org/abs/1904.07248

# Why unsupervised learning?

Such algorithms can be used to explore complex datasets, with minimal assumptions and knowledge.

They are of particular importance to scientific research, since they can be used to extract new knowledge from existing datasets, and can facilitate new discoveries.

# Why unsupervised learning?

## (1) Clustering:

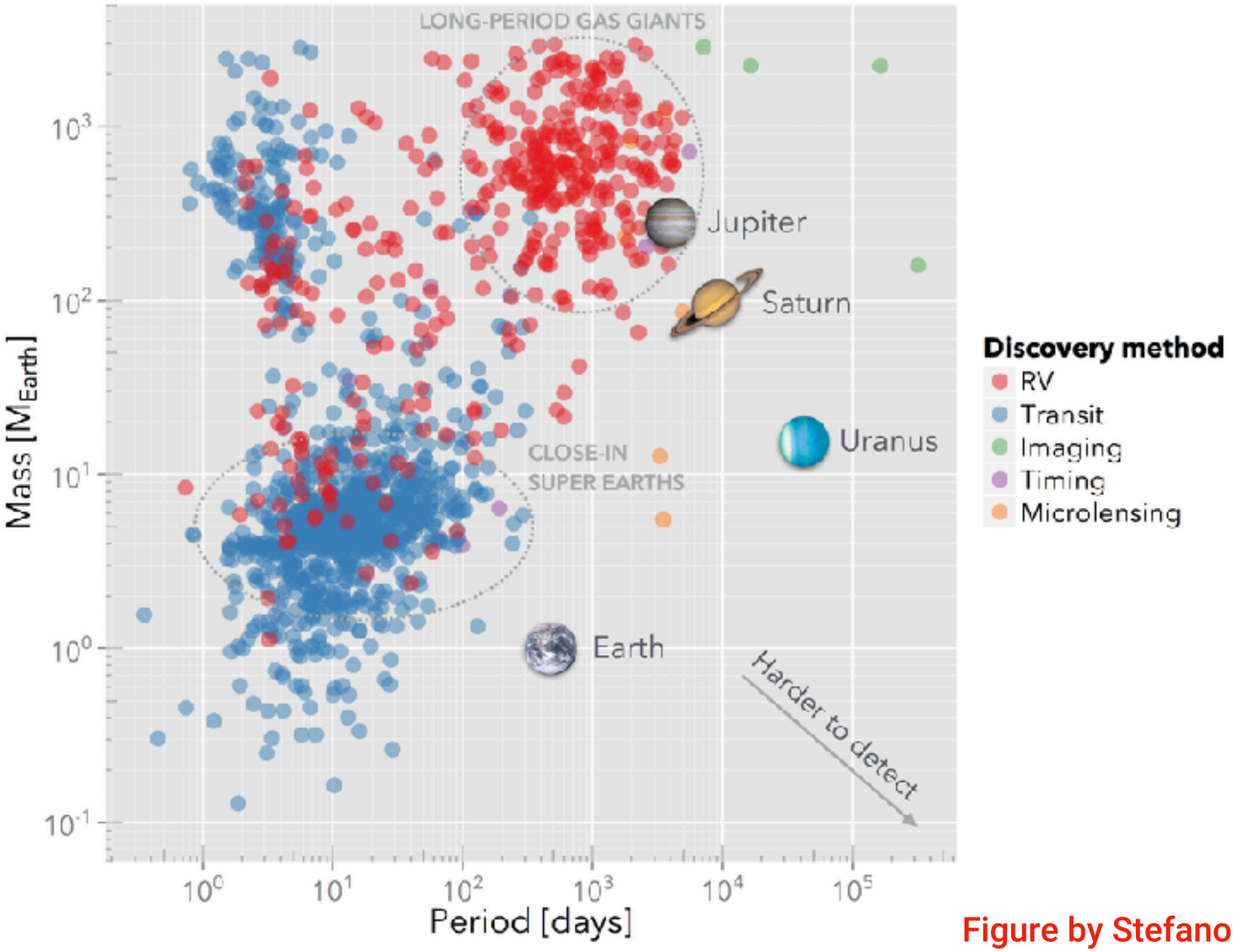
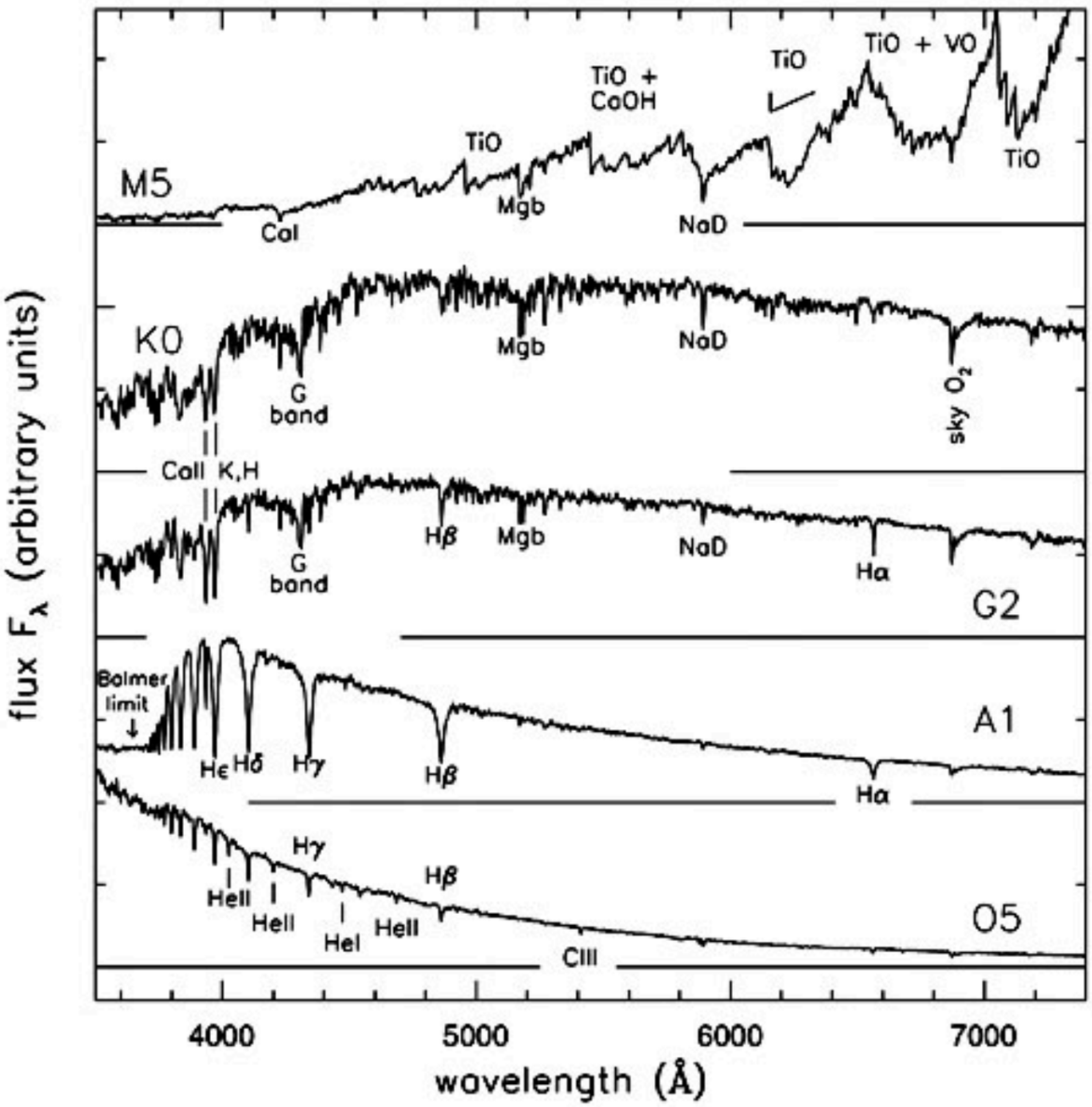
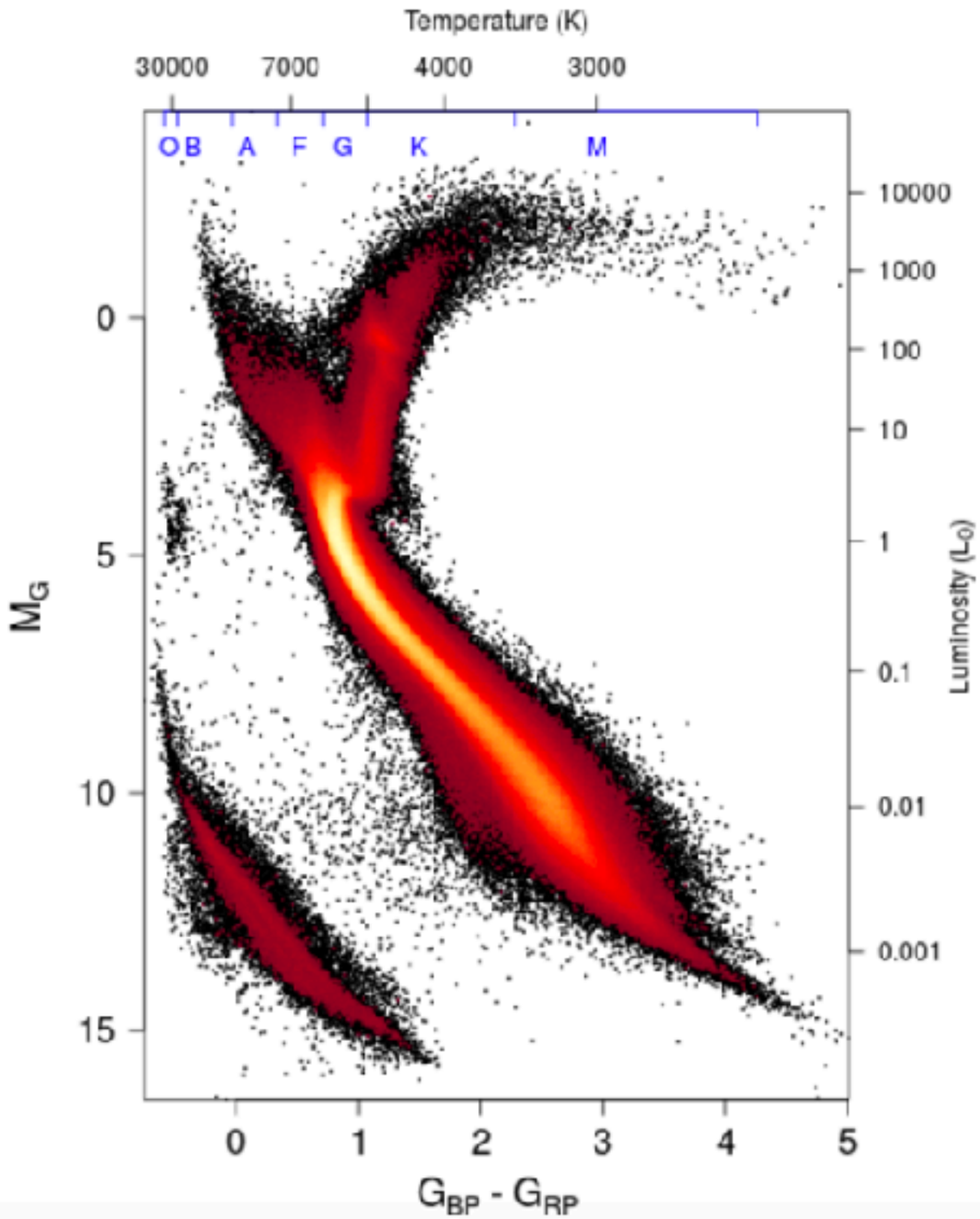



Figure by Stefano Meschiari

# Why unsupervised learning?

## (2) Dimensionality reduction:

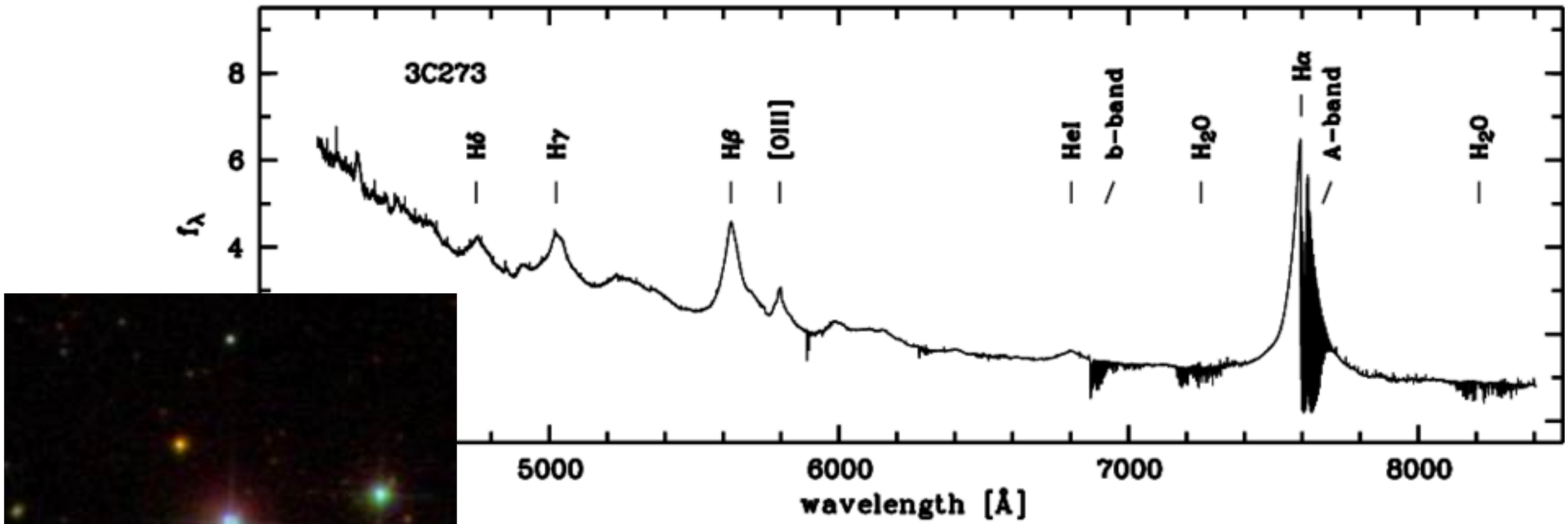



From: Gaia Collaboration et al. 2018

# Why unsupervised learning?

## (3) Outlier detection:





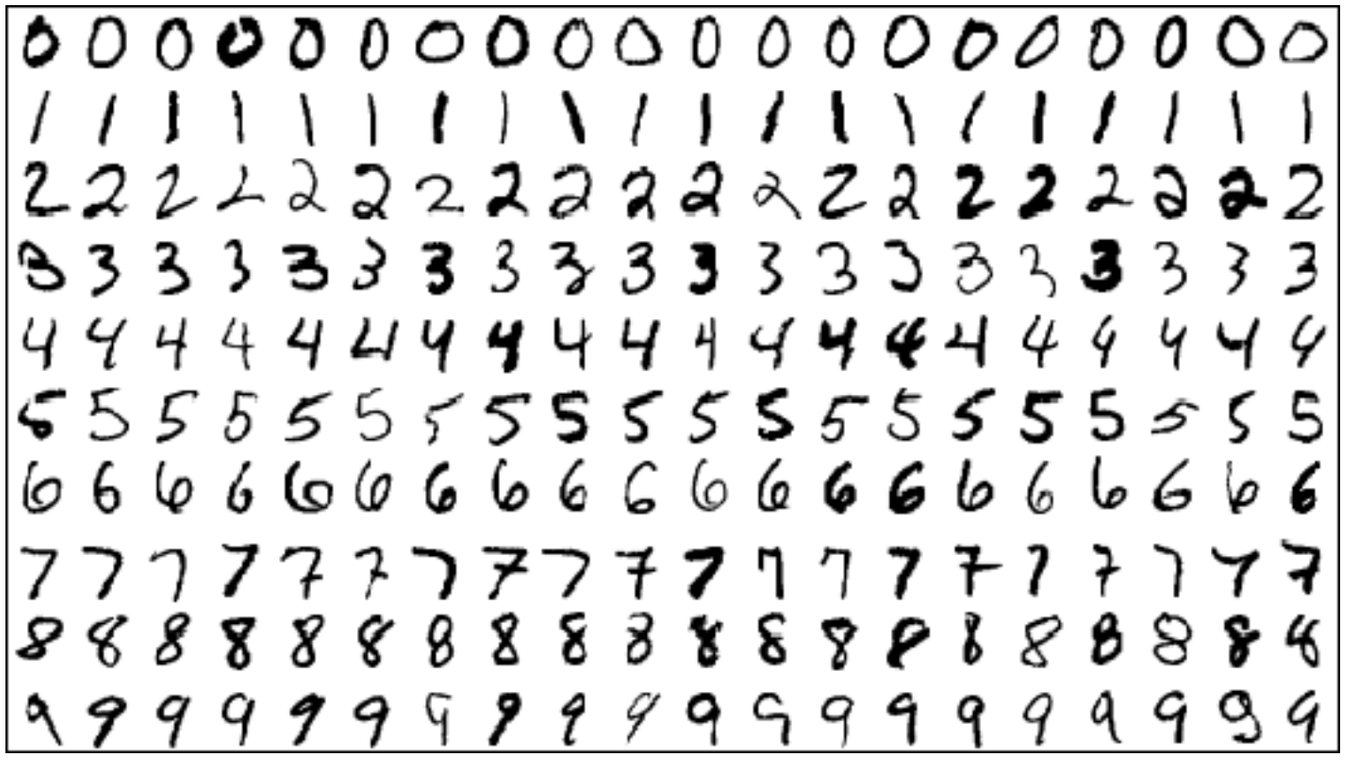

From: SDSS & Dietrich+ 1999

# Dimensionality Reduction

# tSNE and UMAP

**Dimensionality reduction algorithms used for embedding of high-dimensional data into a low dimensional space (typically 2D or 3D).**

**MNIST dataset:** 28x28 features per image

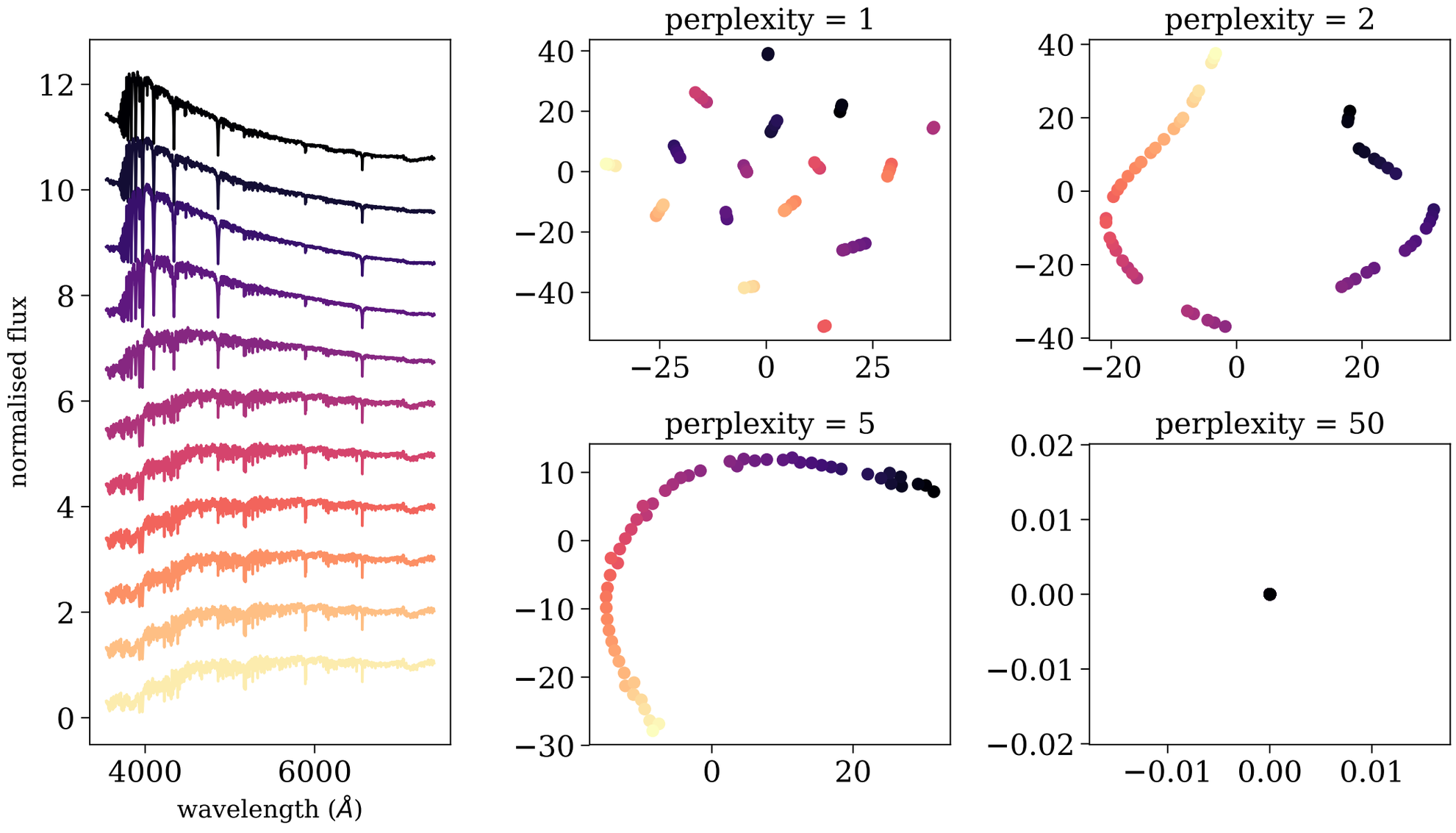



**tSNE:** L.J.P. van der Maaten and G.E. Hinton (2008), https://lvdmaaten.github.io/tsne/

**UMAP:** Leland McInnes, John Healy, and James Melville (2018), https://umap-learn.readthedocs.io/en/latest/

# tSNE and UMAP

**The resulting embedding depends on several choices (e.g., the distance metric) and on several hyper-parameters.**



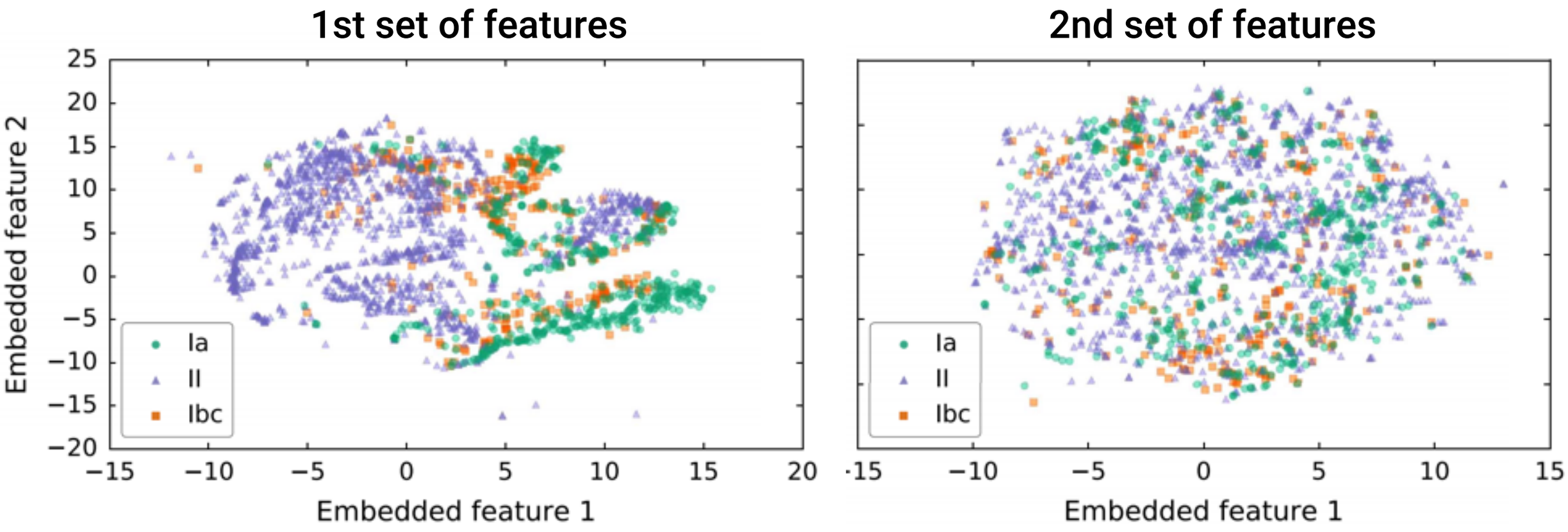

**tSNE:** L.J.P. van der Maaten and G.E. Hinton (2008), https://lvdmaaten.github.io/tsne/

**UMAP:** Leland McInnes, John Healy, and James Melville (2018), https://umap-learn.readthedocs.io/en/latest/

# tSNE: example with supernovae classification

**Photometric classification of supernovae with machine learning:**

One of the challenges in supernova classification tasks is the selection of a set of features. Dimensionality reduction algorithms, such as tSNE, can be used to explore different sets of features in order to assess their information content.

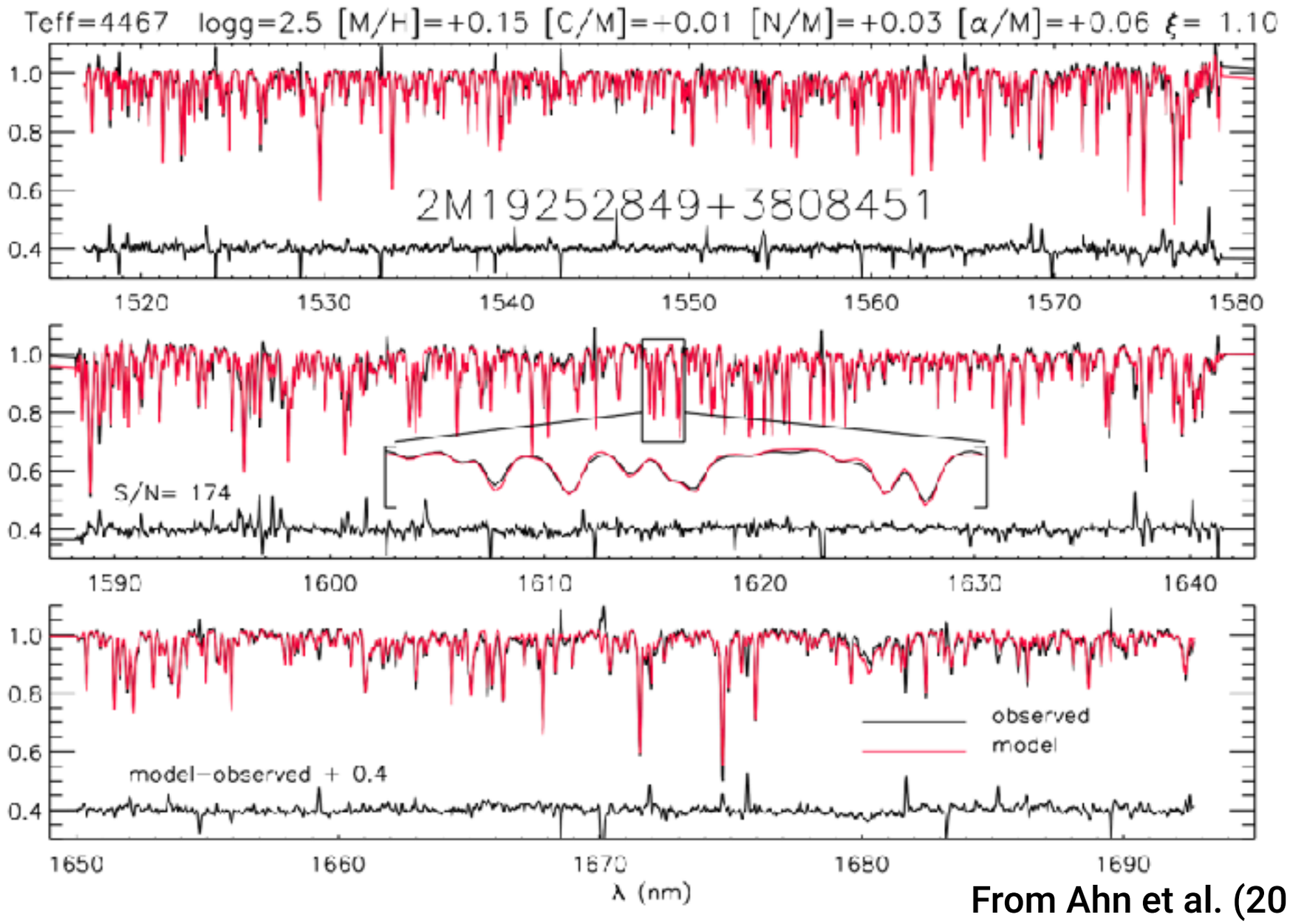



From: Lochner et al. (2016)

# tSNE: example with APOGEE stars

**APOGEE: high resolution infrared spectra of more than 100,000 stars in the Milky Way.**

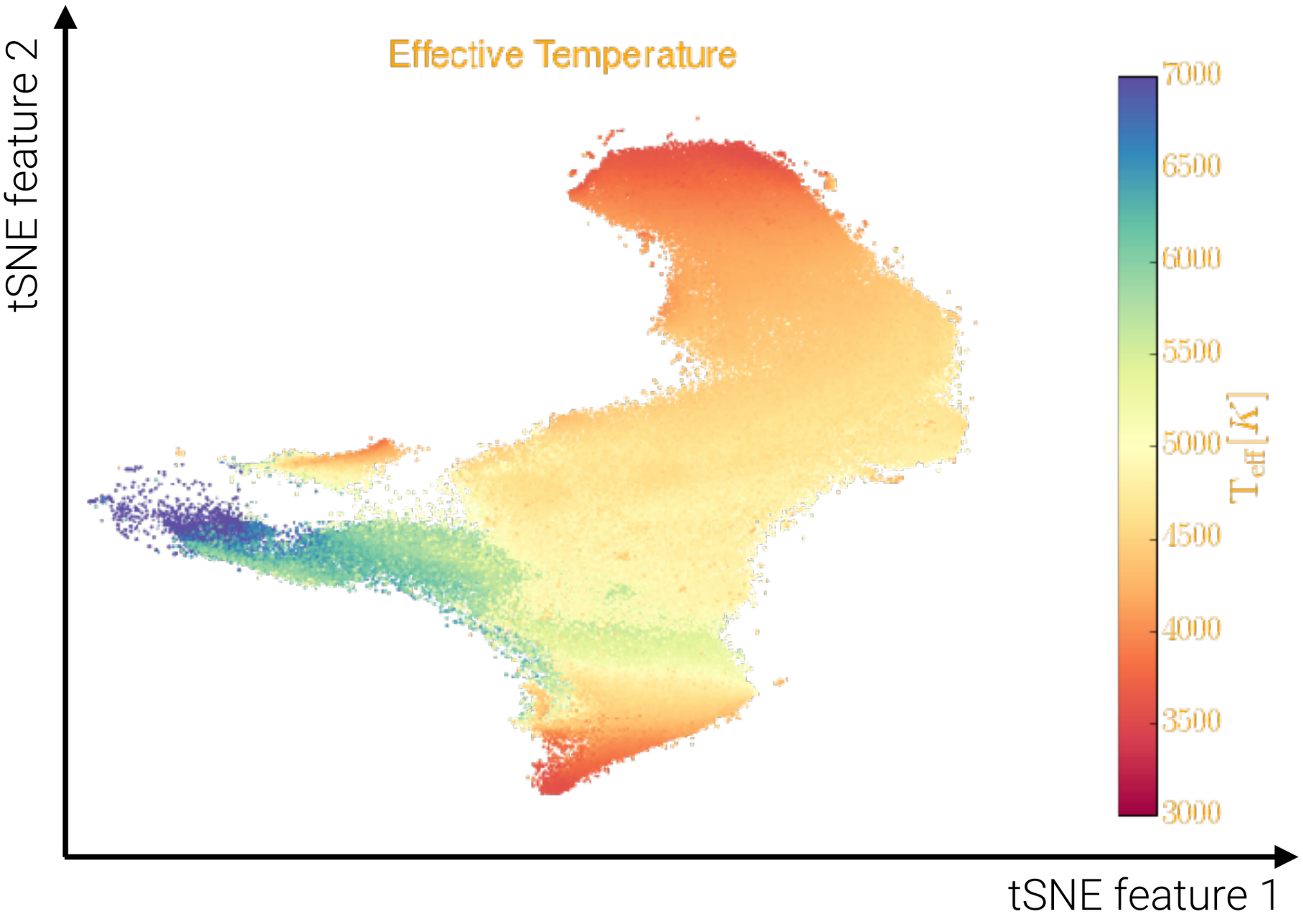
The survey provides the processed infrared spectra, and catalogs of radial velocities, stellar parameters, and abundances derived from these spectra.




From Ahn et al. (2014)

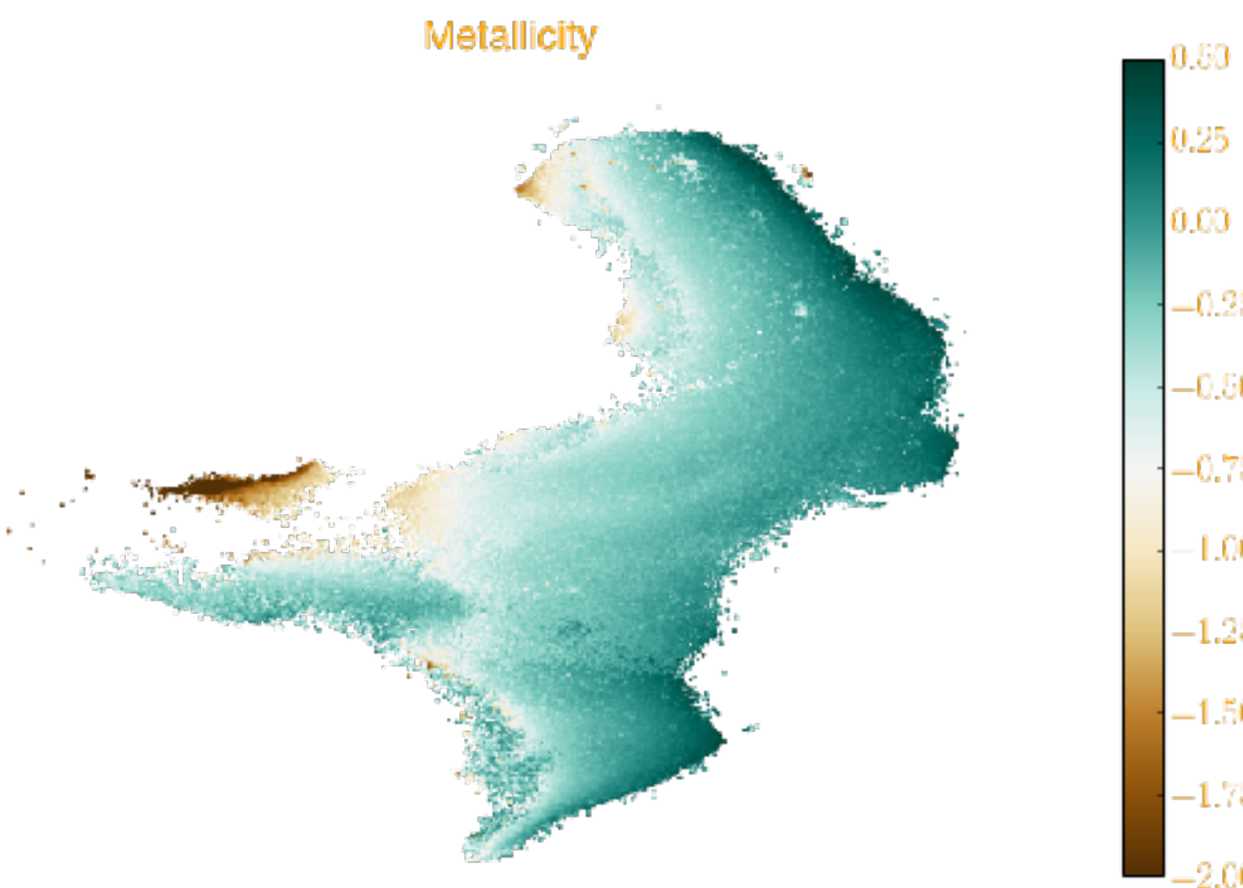
# tSNE: example with APOGEE stars

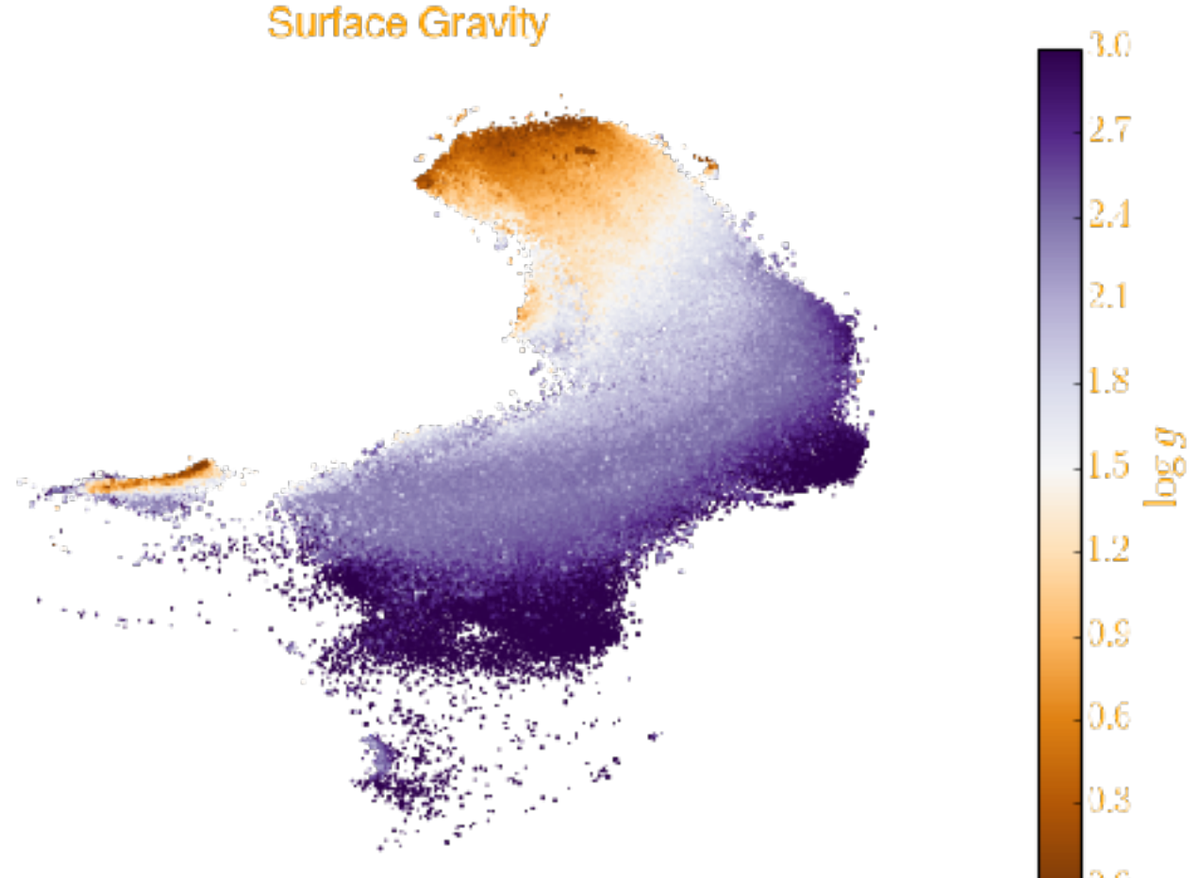
**Dimensionality reduction of the APOGEE dataset:** we assigned distances between the objects in the sample using unsupervised Random Forest (see Baron & Poznanski 2017), and applied tSNE for dimensionality reduction. The resulting embedding was then colored according to derived parameters from the public catalog (see Reis et al. 2018).



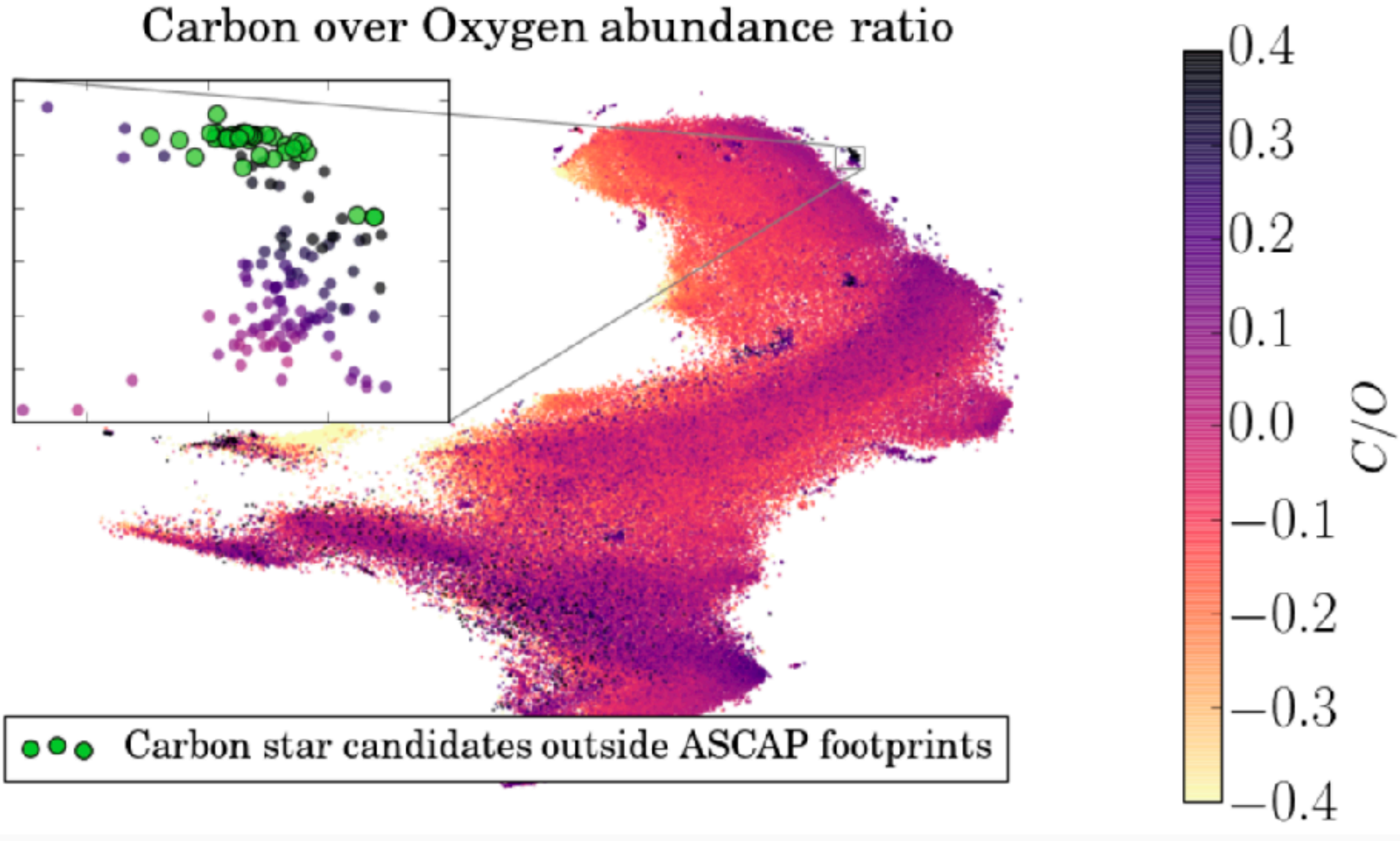
# tSNE: example with APOGEE stars

**Dimensionality reduction of the APOGEE dataset:** we assigned distances between the objects in the sample using unsupervised Random Forest (see Baron & Poznanski 2017), and applied tSNE for dimensionality reduction. The resulting embedding was then colored according to derived parameters from the public catalog (see Reis et al. 2018).



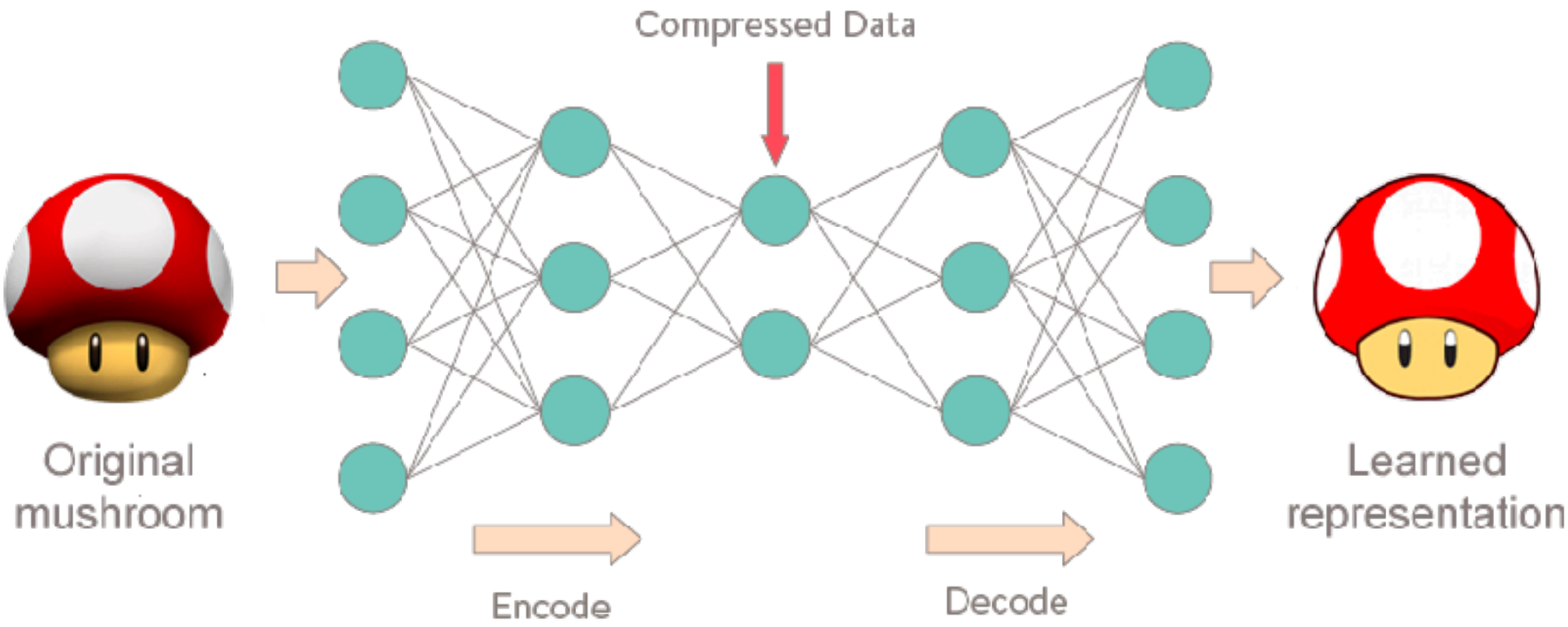



# tSNE: example with APOGEE stars



# Autoencoders

A neural network used to learn an efficient low-dimensional representation of the input dataset, and can be used for compression, dimensionality reduction, and visualization.




**loss function =** $($  $-$  $)^2$

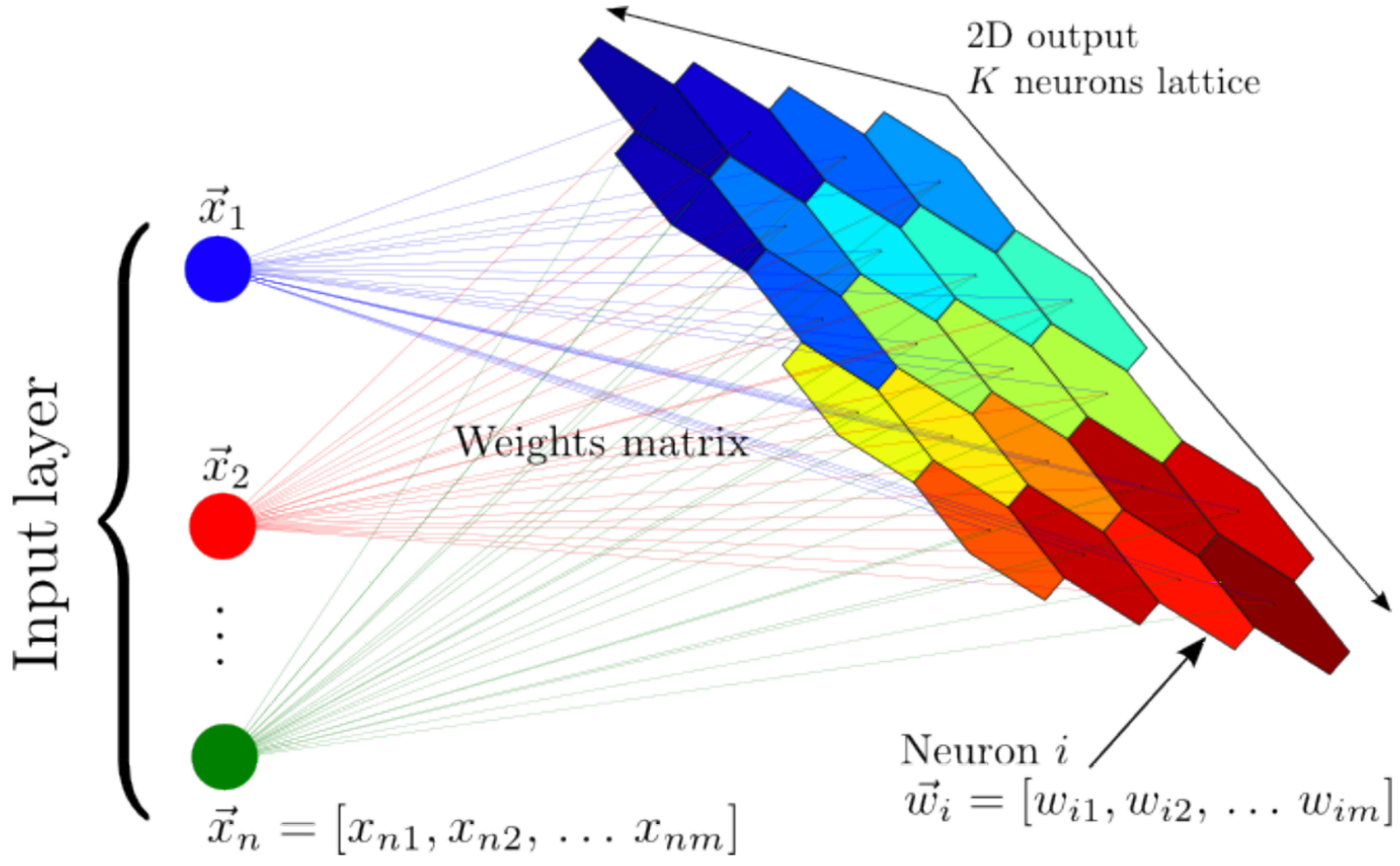
Original mushroom

Learned representation

**Examples in astronomy include:** Gianniotis et al. (2015); Yang & Li (2015); Gianniotis et al. (2016); Ma et al. (2018b); Schawinski et al. (2018); Ralph et al. (2019); Sedaghat et al. (2021).

# Self-Organizing Maps

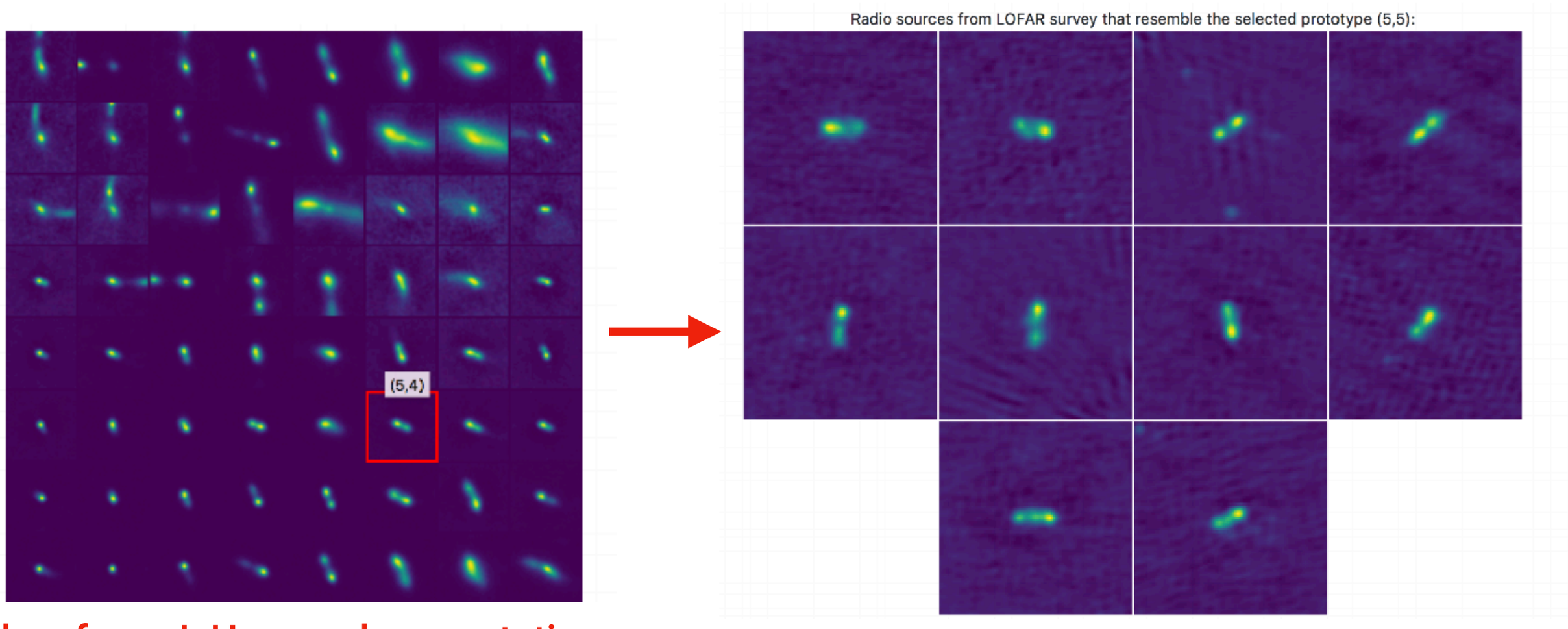
**An unsupervised neural network used to produce a low dimensional representation of the input dataset using a set of prototypes. The prototypes are built during training to match as closely as possible the input data.**




**Examples in astronomy include:** Fustes et al. (2013); Carrasco Kind & Brunner (2014); Armstrong et al. (2016); Polsterer et al. (2016); Armstrong et al. (2017); Meusinger et al. (2017); Rahmani et al. (2018); Galvin et al. (2019); Ralph et al. (2019).

# Self-Organizing Maps

SOM prototypes allow a fast and efficient exploration of large datasets. The distance from the prototypes can be used to retrieve similar objects and to search for outliers.

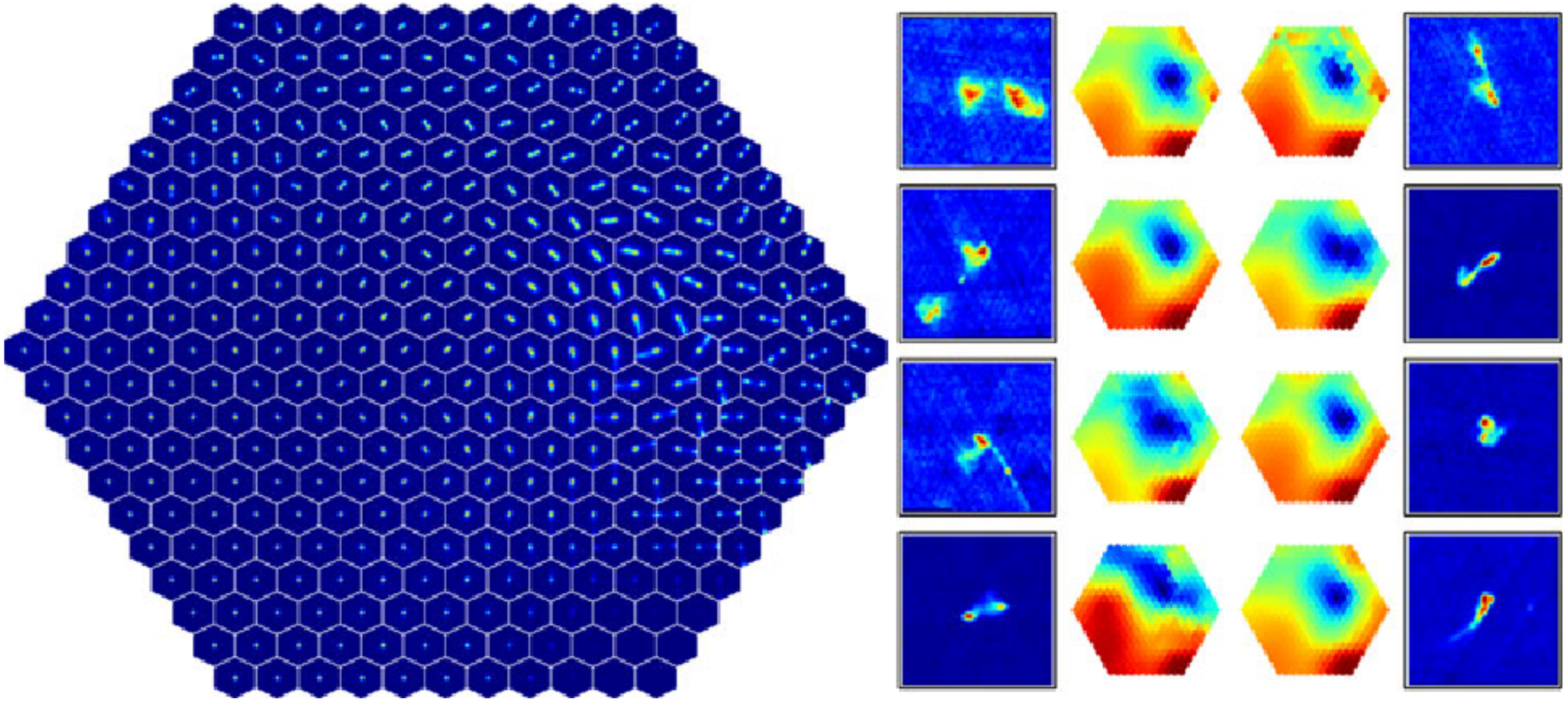



Taken from J. Harwood presentation

**Examples in astronomy include:** Fustes et al. (2013); Carrasco Kind & Brunner (2014); Armstrong et al. (2016); Polsterer et al. (2016); Armstrong et al. (2017); Meusinger et al. (2017); Rahmani et al. (2018); Galvin et al. (2019); Ralph et al. (2019).

# Self-Organizing Maps: PINK

The low dimensional representation and the resulting prototypes depend on internal choices (e.g., distance assignment). Thus, they are not invariant under rotations and flips.

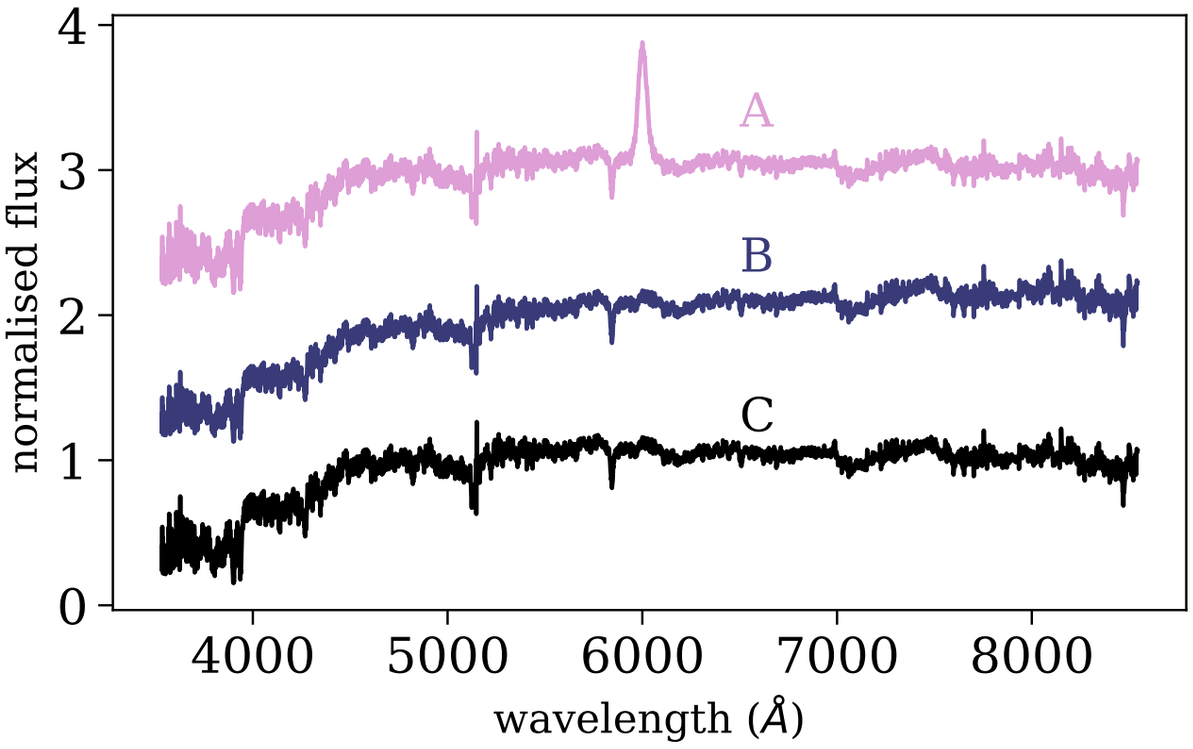


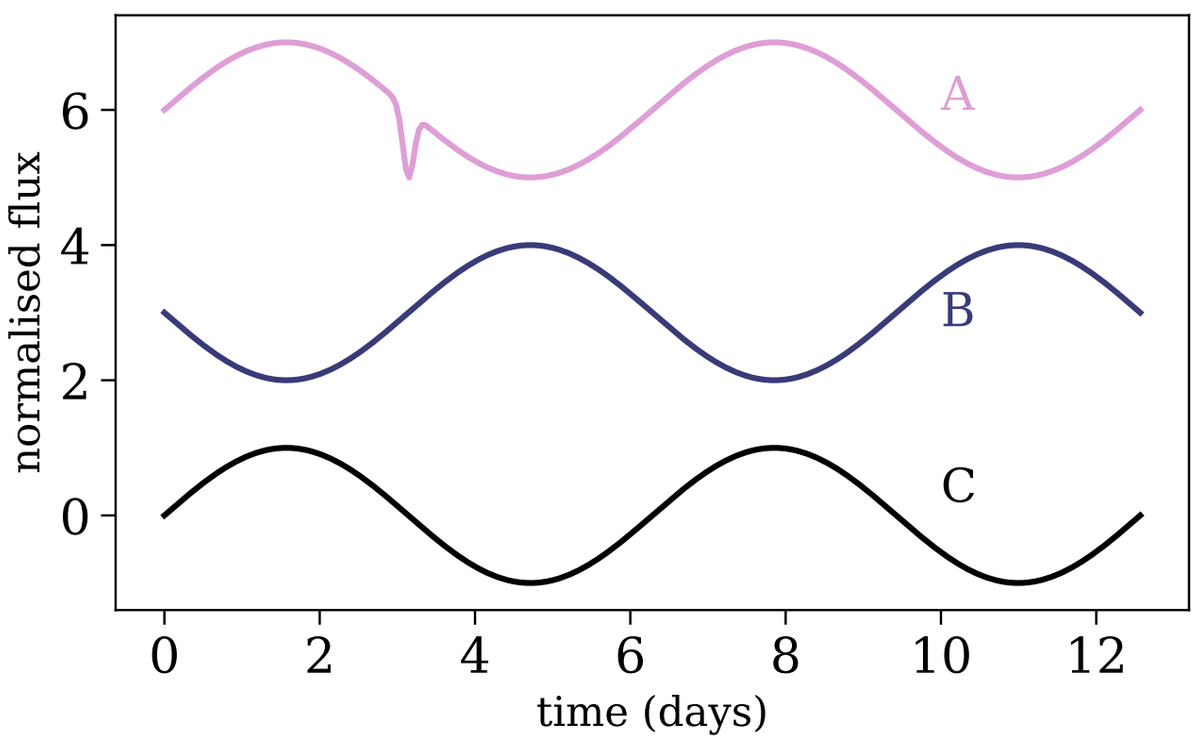
**Examples in astronomy include:** Polsterer et al. (2016); Galvin et al. (2019); Mostert et al. (2021)

# Current Challenges

(*) The resulting dimensionality reduction depends on the algorithm’s hyper-parameters. How do we choose the “correct" hyper-parameter values?

(*) Most of the algorithms measure distances between the objects in the sample. Which distance metric is appropriate for the dataset at hand?

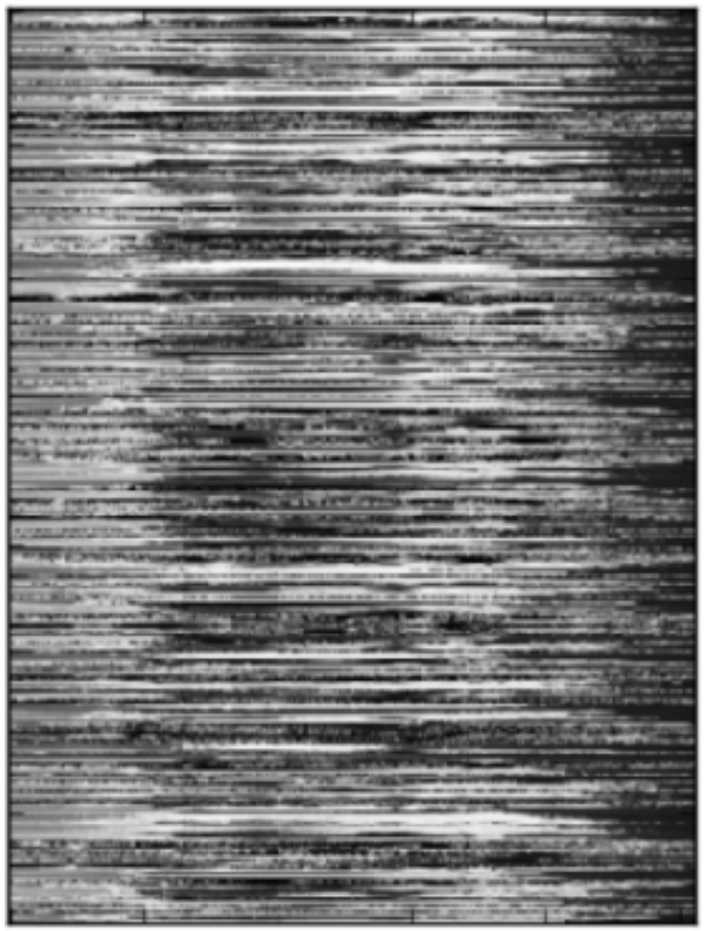


# The Sequencer

**Baron & Ménard (2020); Arxiv: https://arxiv.org/abs/2006.13948**

Input



**The algorithm reorders the data according to a detected sequence:**

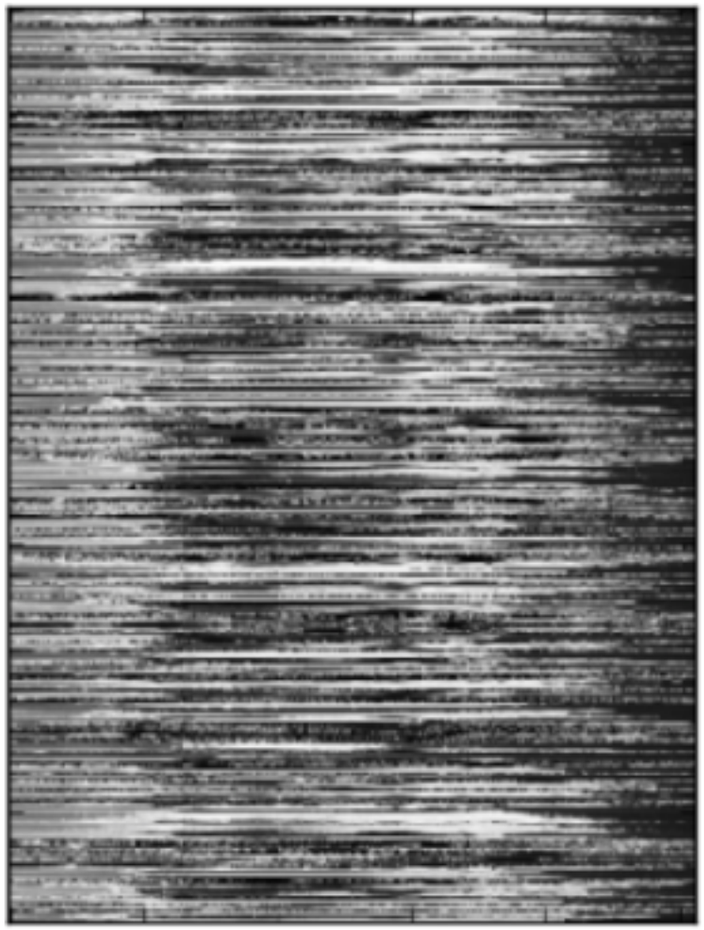
- Based on pure statistics - no training, no randomness, result is always the same.
- Provides a **score**.
- Algorithm hyper-parameters and distance assignments are optimized using the score.

  **So, result does not depend on hyper-parameters or the distance metric.**

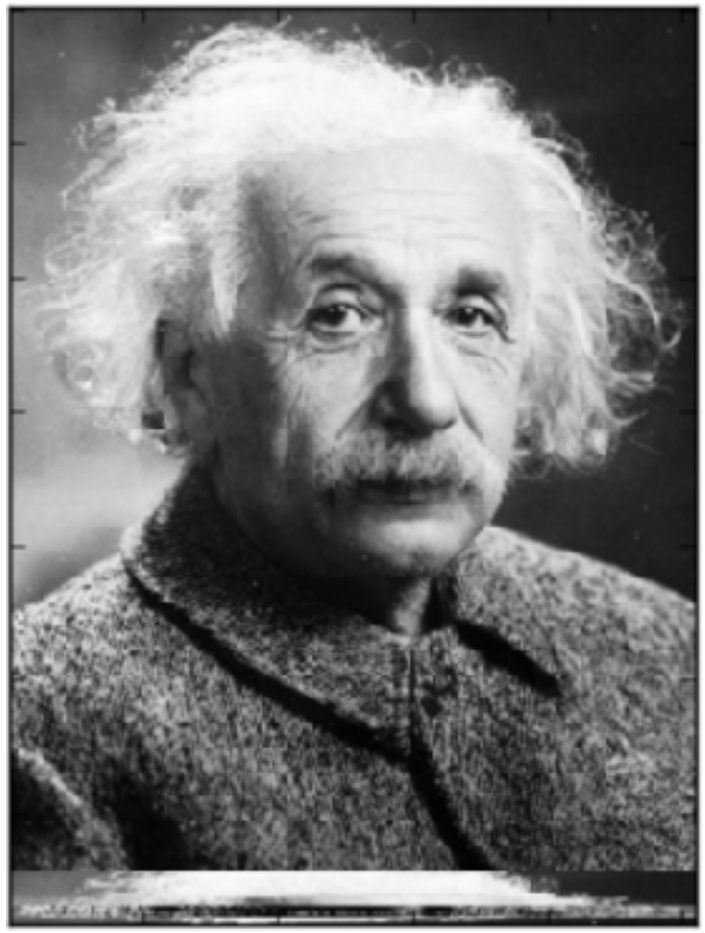
# The Sequencer

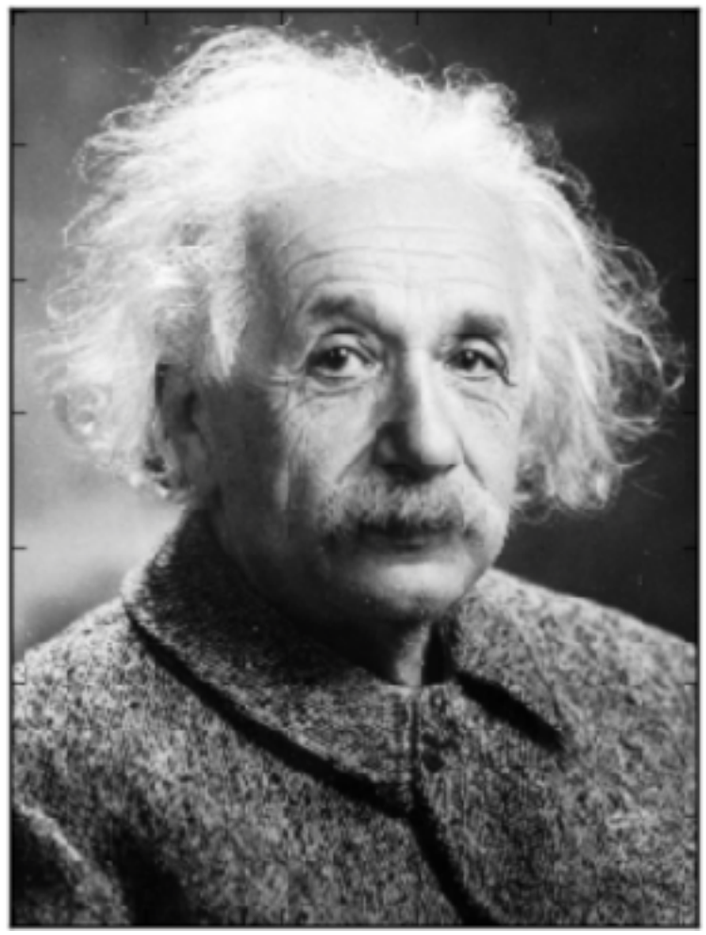
**Baron & Ménard (2020); Arxiv: https://arxiv.org/abs/2006.13948**

Input



Output

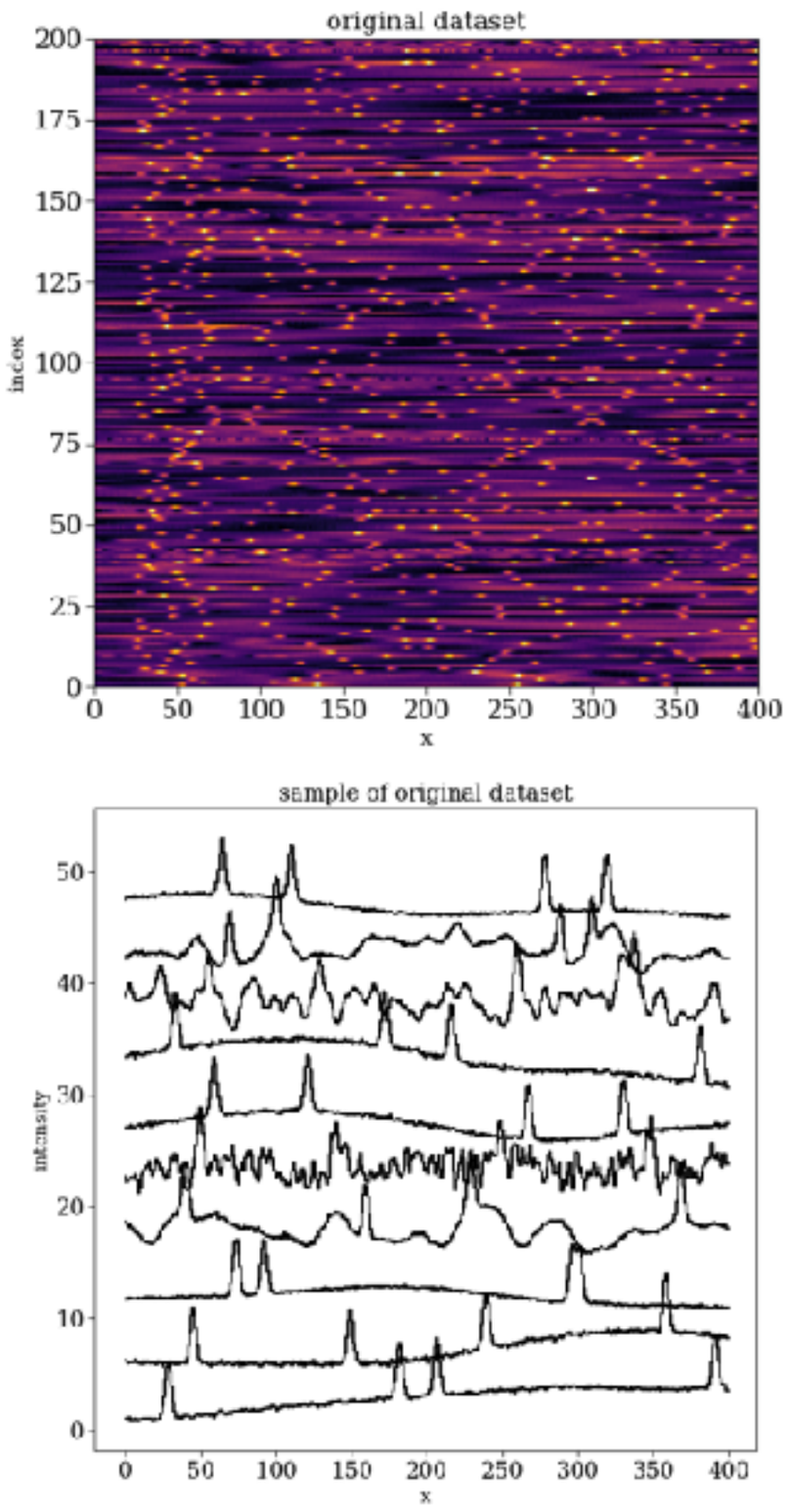


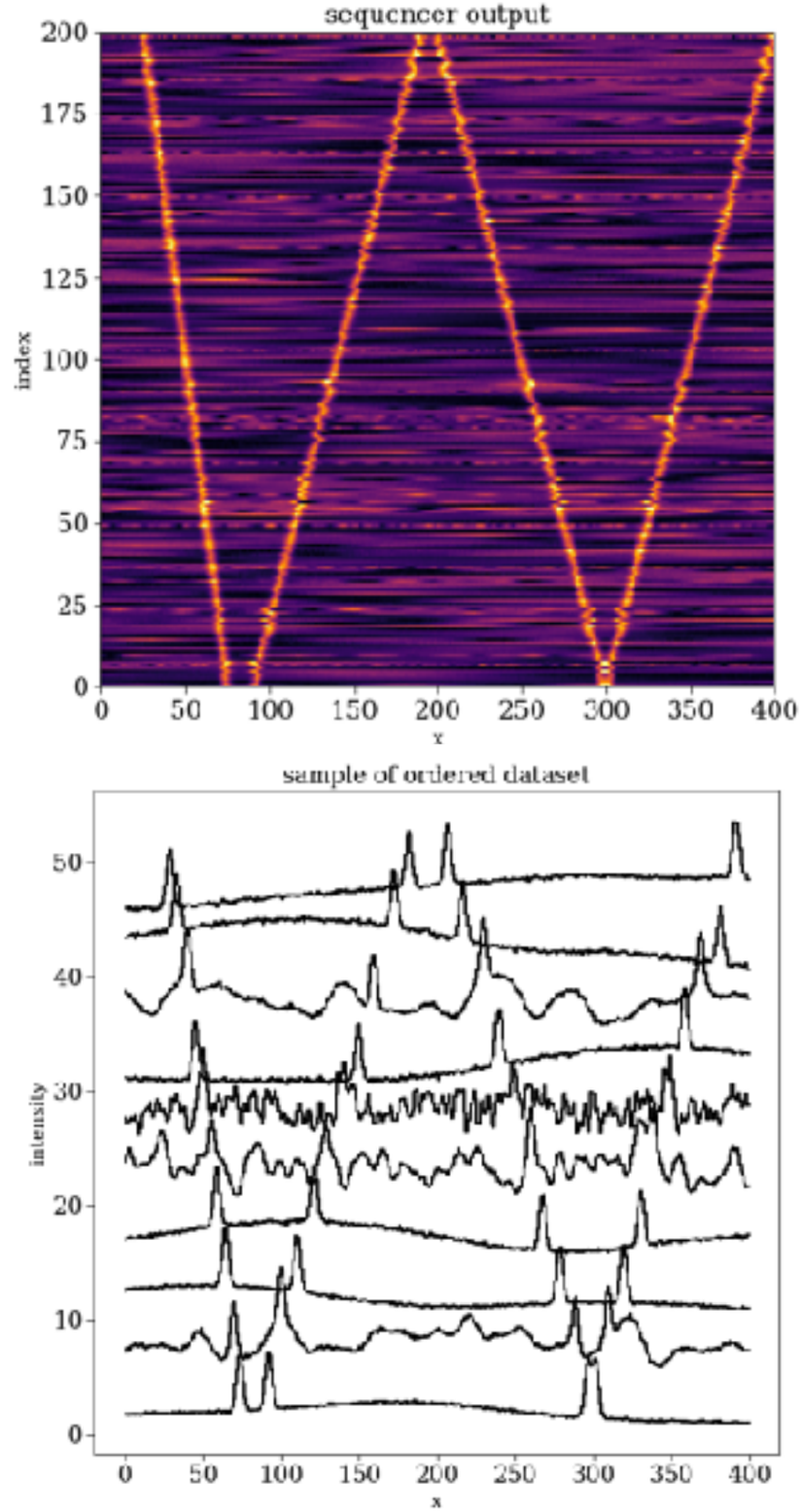


**The algorithm reorders the data according to a detected sequence:**

- Based on pure statistics - no training, no randomness, result is always the same.
- Provides a **score**.
- Algorithm hyper-parameters and distance assignments are optimized using the score.
  **So, result does not depend on hyper-parameters or the distance metric.**

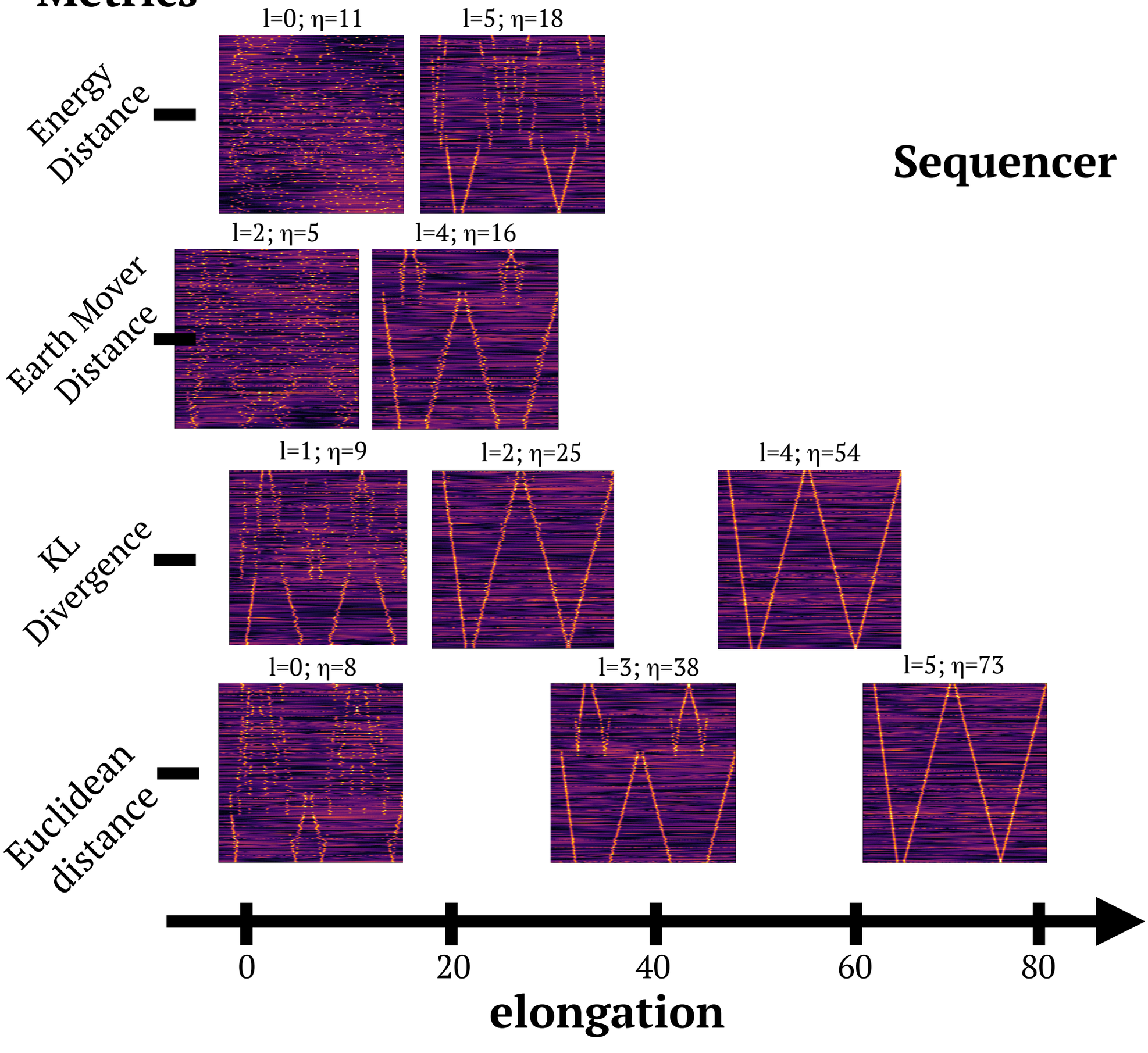
# Why is a figure of merit important?



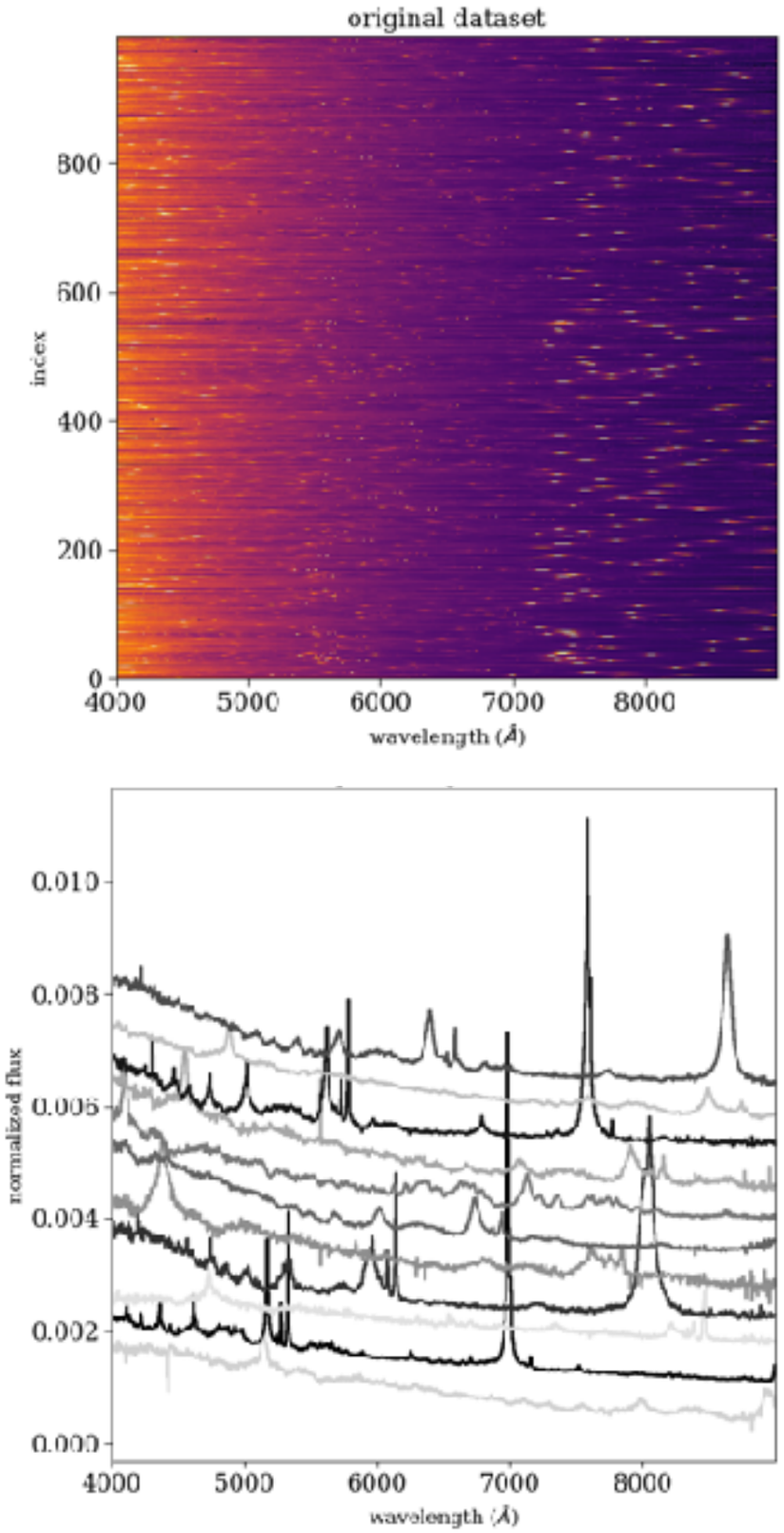



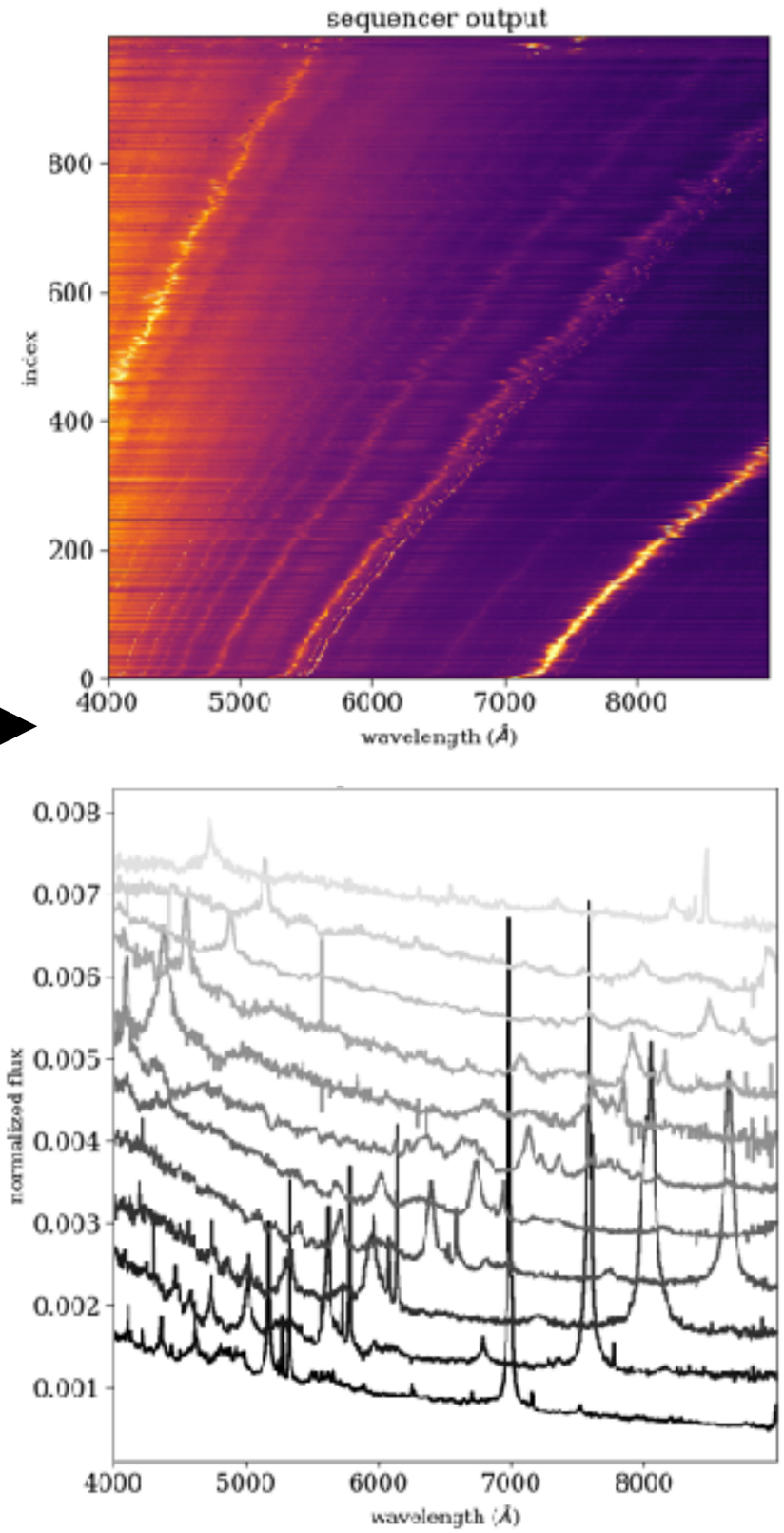
# Metrics

# Sequencer



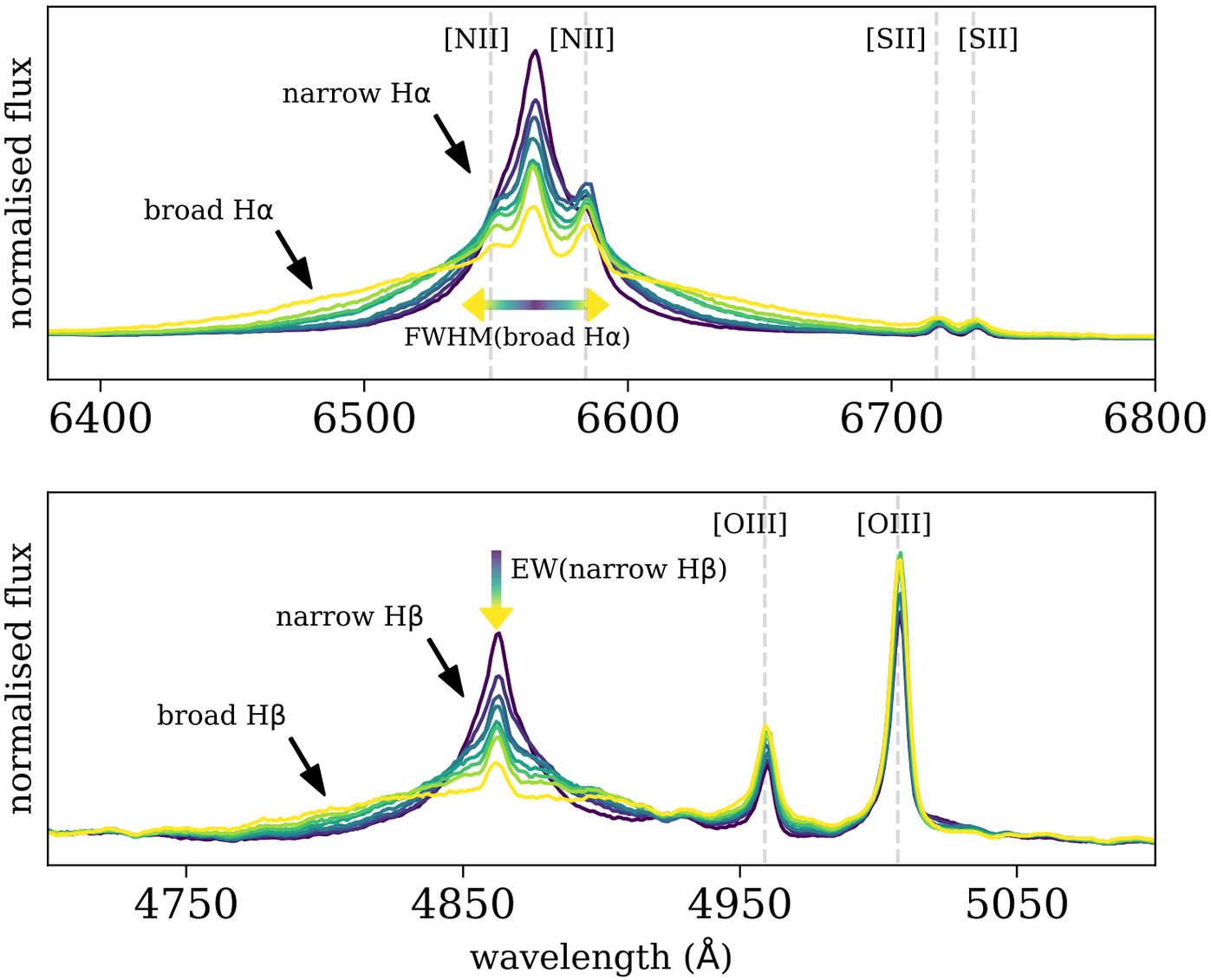
# The quasar redshift sequence case

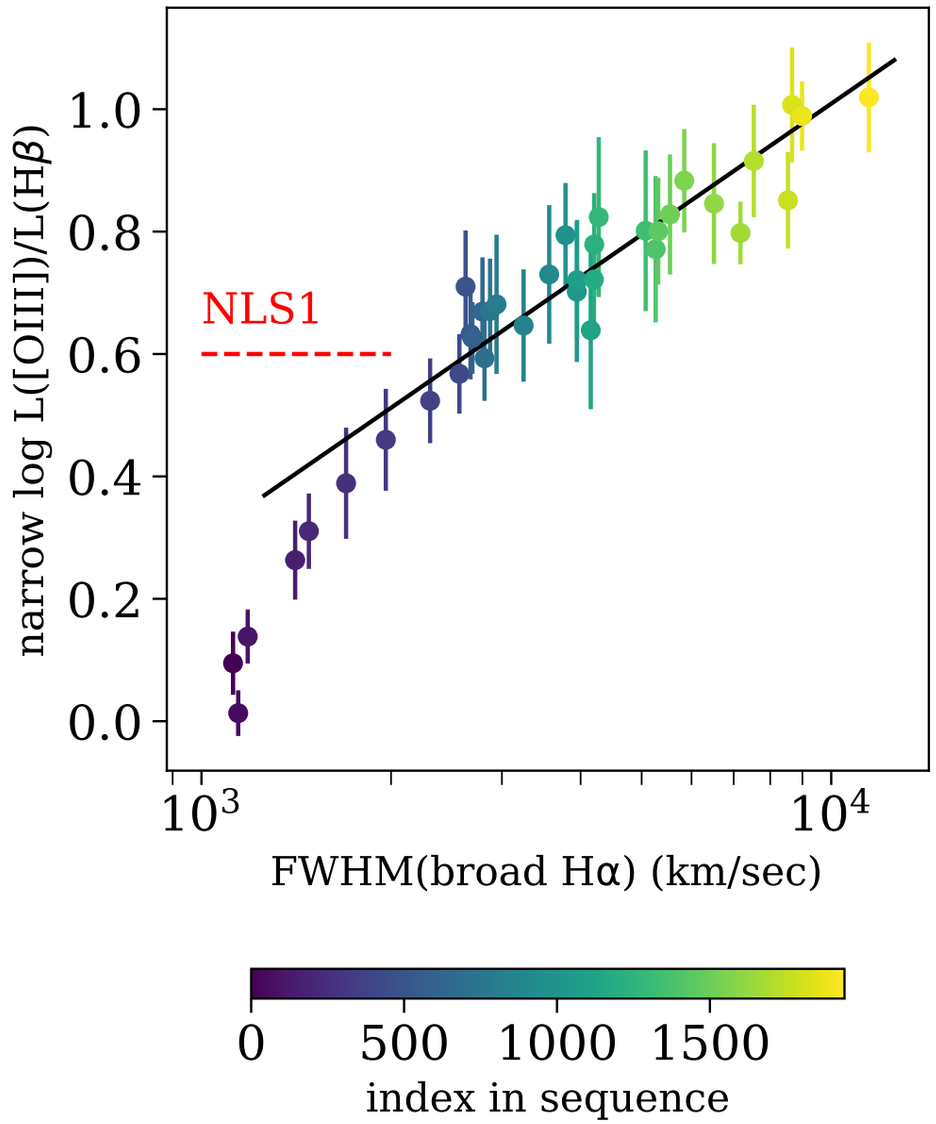


# The Sequencer: a new correlation discovered in Active Galactic Nuclei.

Baron & Ménard (2019)



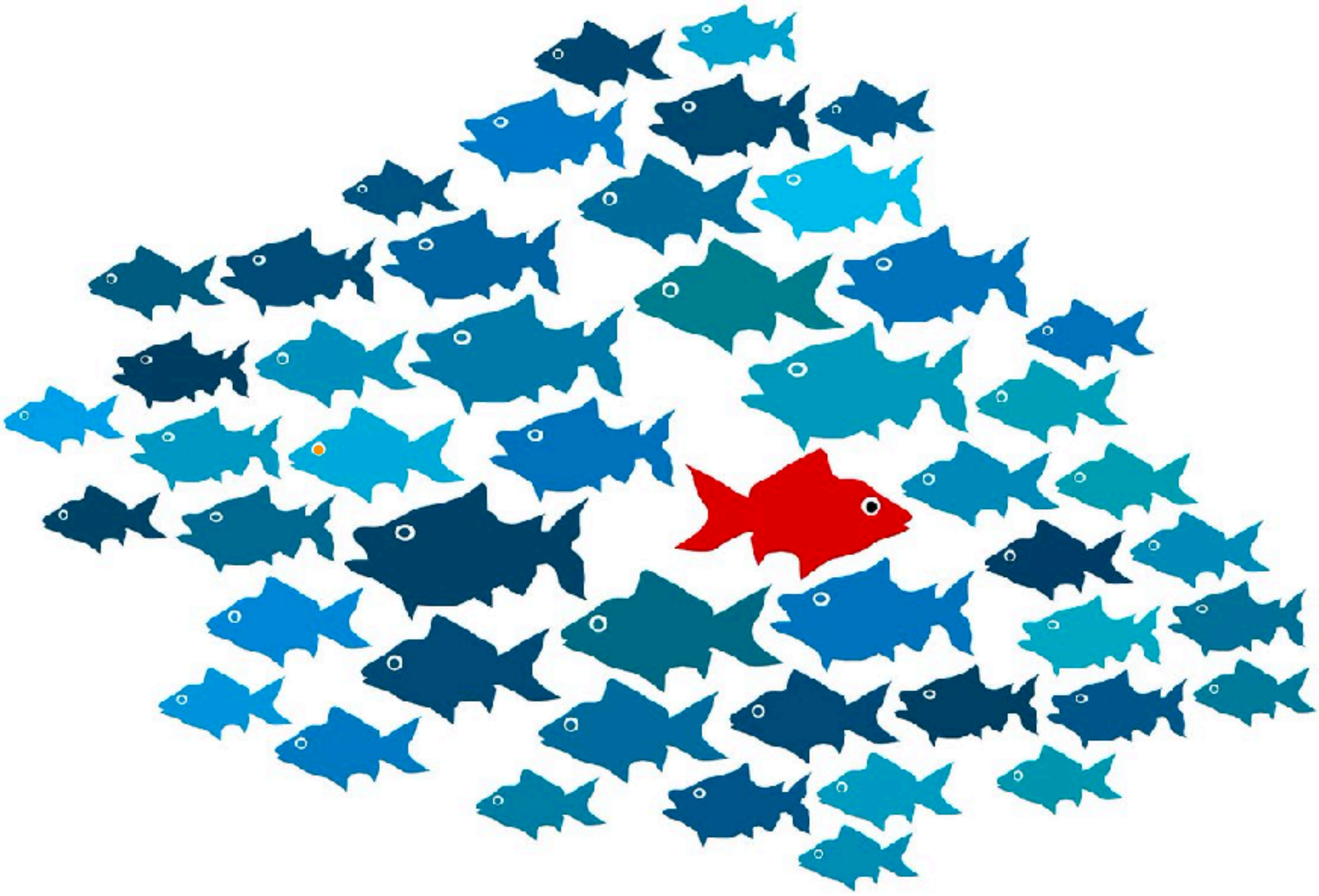



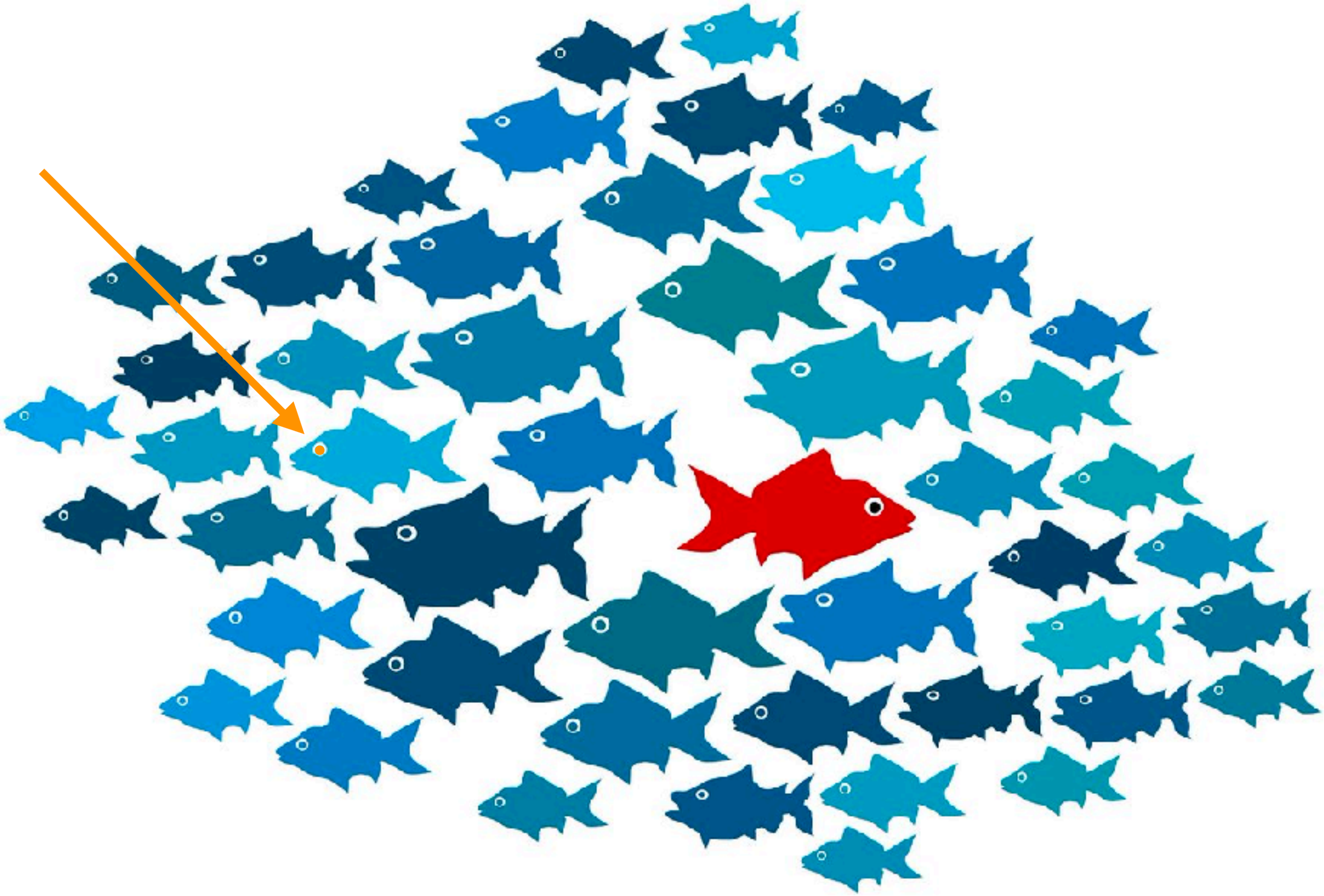
# Outlier Detection

# What are outliers?

- **“Bad” object:** problem with the instrument, faulty observation, pipeline error, etc.
- **Misclassified object:** a star in a dataset of galaxies.
- **Tail of a distribution:** the most-luminous galaxy in the sample.
- **Unknown unknowns:** completely new objects we did not know we should be looking for.

**In astronomy:** processes which happen on shorter time scales.





# How can we find outliers?

(1) **Serendipitously:** an expert going through their data and finding unexpected objects. -> **Usually not applicable for large datasets**.

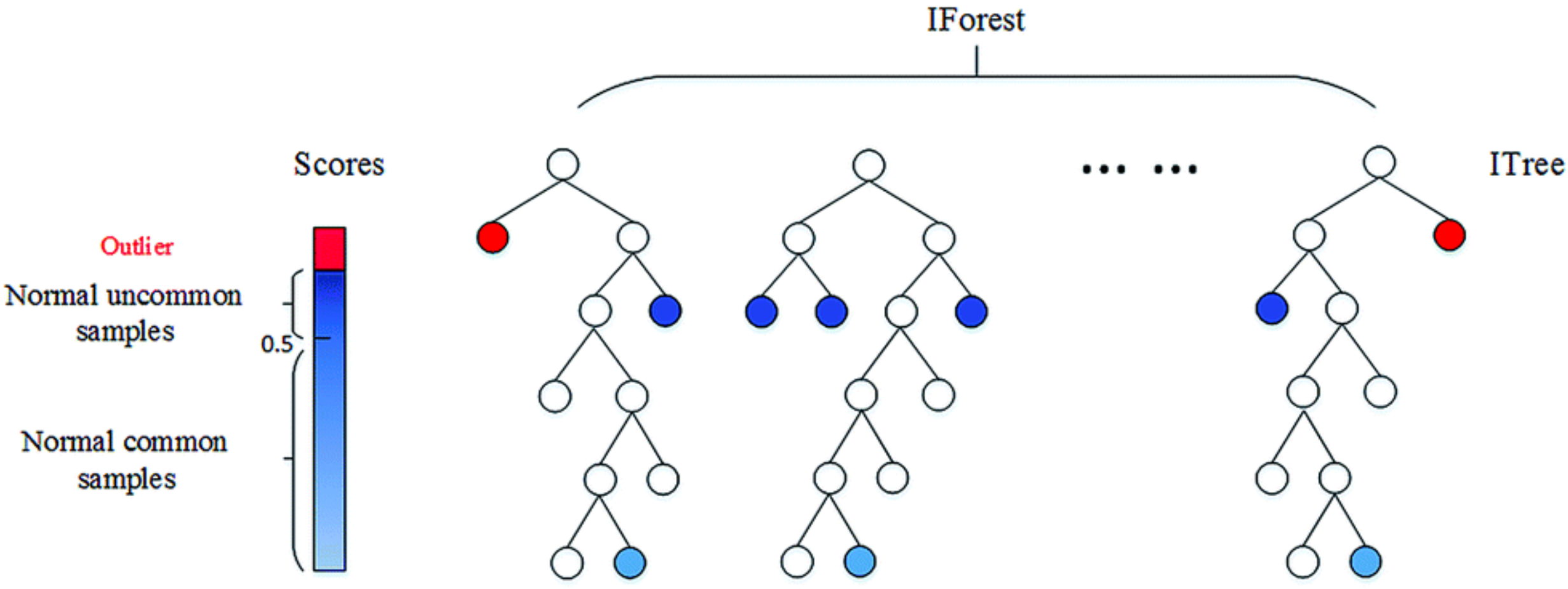
# How can we find outliers with ML?

**(1) Using supervised learning:** objects which have low probability to belong to any of the known classes will be considered outliers (or one-class SVM). -> **Usually find the outliers that “shout the loudest”**.

# How can we find outliers with ML?

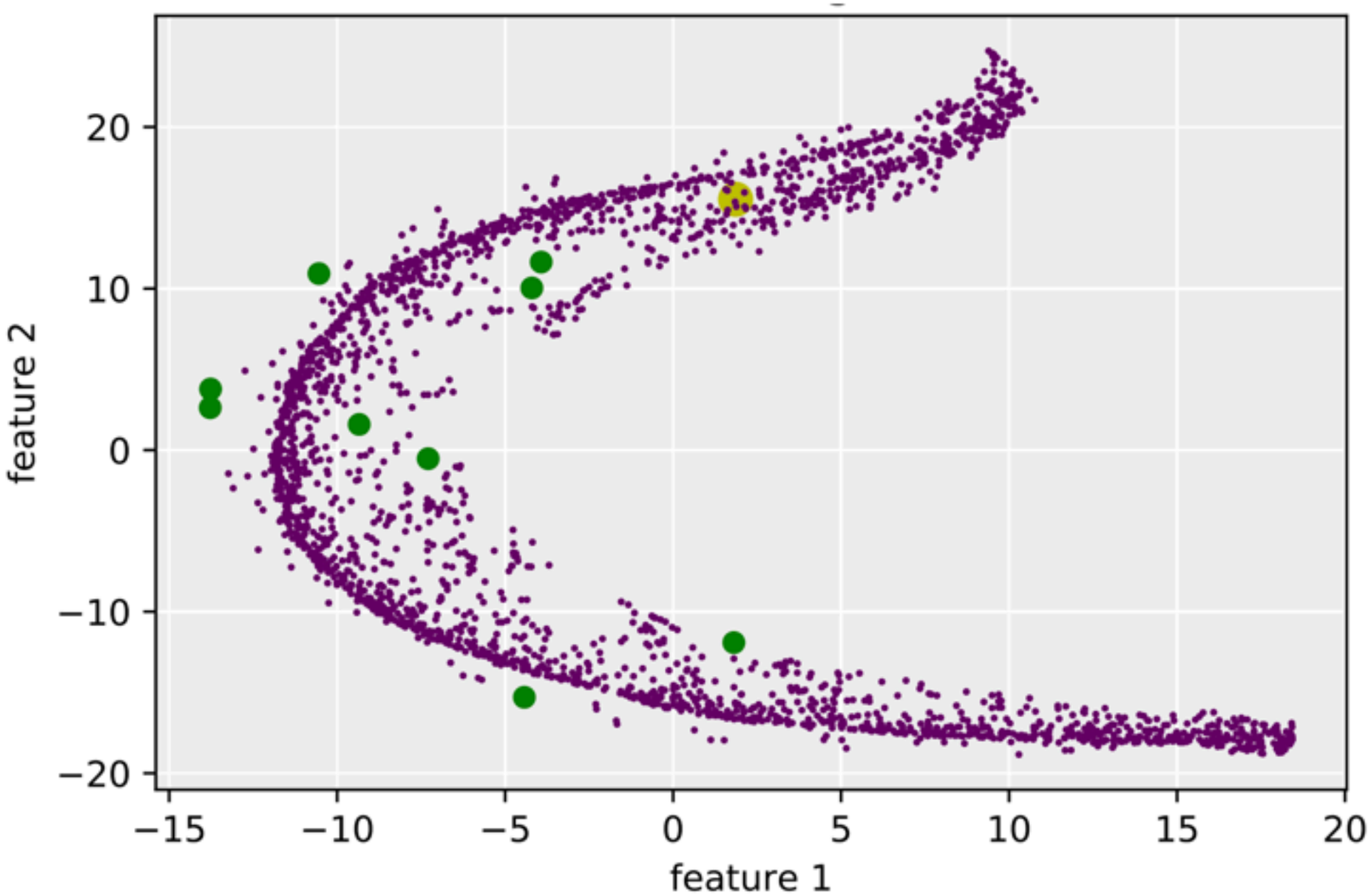
**(2) Using unsupervised learning:** Isolation Forests, unsupervised distance assignment, and using dimensionality reduction algorithms (tSNE, UMAP, autoencoders). -> **Can be used to find more subtle outliers, but strongly depends on domain-specific decisions.**

# Outlier detection with Isolation Forests:




**See recent examples by:** Pruzhinskaya (2019), Doorenbos et al. (2020), Martínez-Galarza et al. (2020), and Webb et al. (2020).

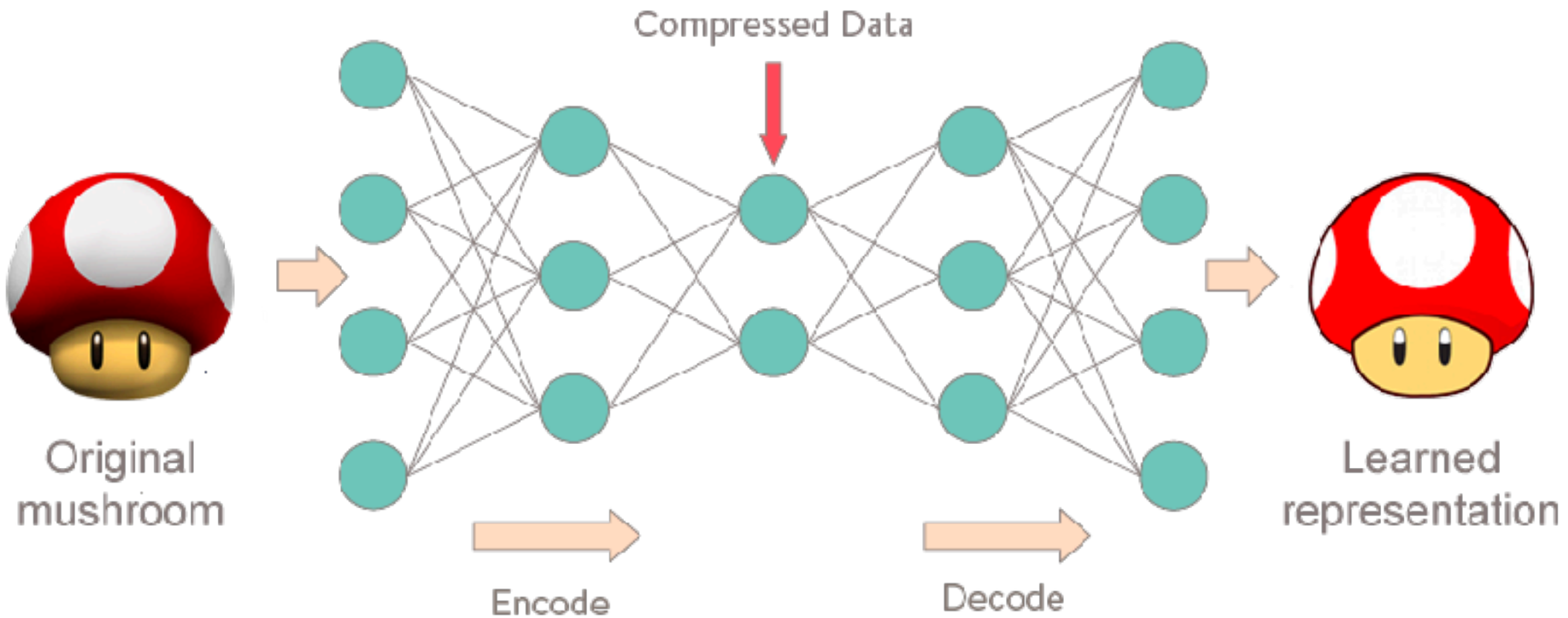
# Outlier detection with tSNE/UMAP:




**See examples by:** Reis et al. (2018), Giles & Walkowicz (2019), Martínez-Galarza et al. (2020).

# Outlier detection with autoencoders or GANs:

**Define outliers as (i) objects with a poor reconstruction, or (ii) outliers in the latent space.**




**See recent examples by:** Ichinohe et al. (2019), D'Addona et al. (2020), Formsma & Saifollahi et al. (2020), Portillo et al. (2020), Storey-Fisher et al. (2020).

# Outlier detection with unsupervised RF:

Random Forest can be used as an unsupervised learning algorithm, to estimate distances between the objects in the sample (Baron & Poznanski 2017). The RF is trained on the entire dataset, and is particularly sensitive to correlations between different features. The objects with the largest distances from the rest are defined as outliers.


**See examples by:** Baron & Poznanski (2017), Reis et al. (2018), Martínez-Galarza et al. (2020), D'Addona et al. (2020).

# Summary and a few tips

- Unsupervised learning algorithms can be used to explore complex datasets, and can thus facilitate new discoveries.

- Stay as close as possible to the original dataset: derived features might be problematic.

- Not truly unsupervised: the results of these algorithms depends on several choices, such as distance metric and hyper-parameters:

  - Domain-specific choices, such as an appropriate distance metric, are expected to give better results.

  - Unsupervised learning is a first step in the long process of making discoveries. This process requires scientists.