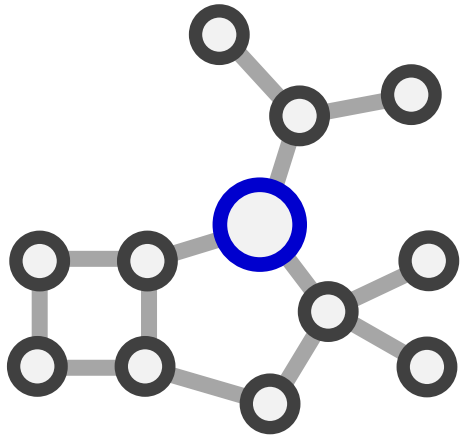
Unsupervised Learning

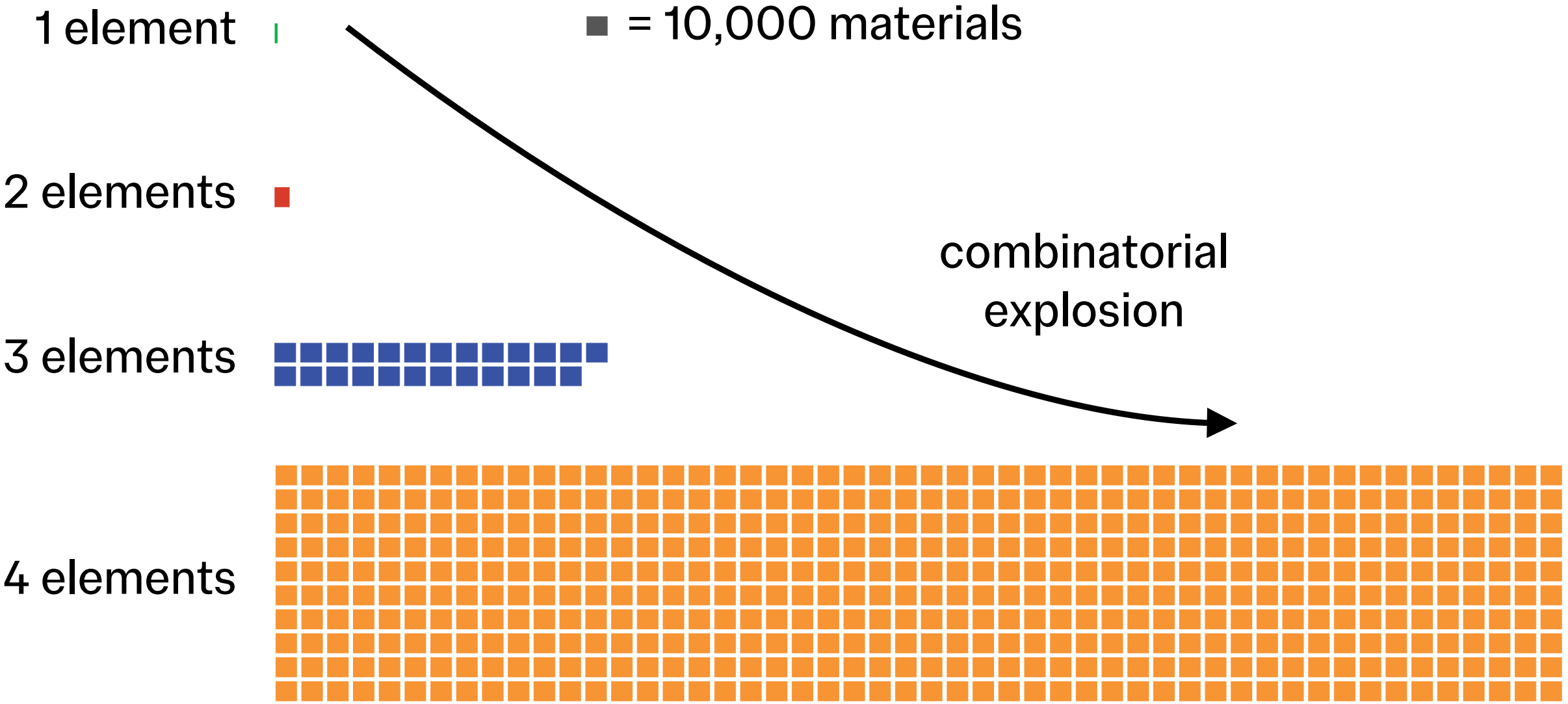
Chemistry and Materials Machine Learning School

Alex Ganose
Department of Chemistry
Imperial College London
a.ganose@imperial.ac.uk



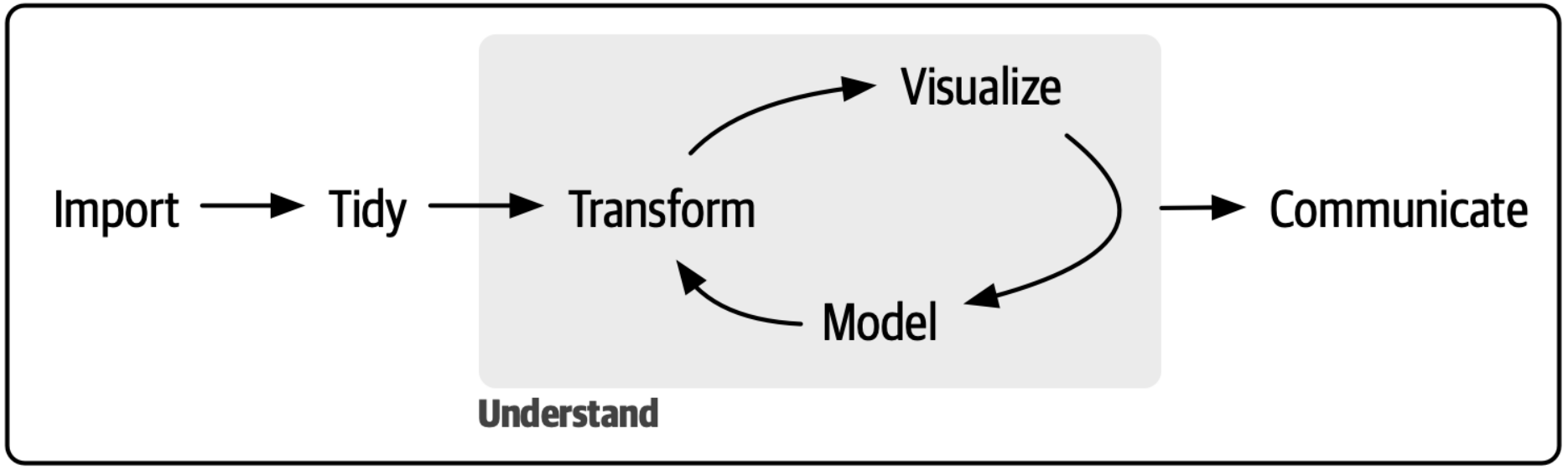
Group website: virtualatoms.org

Chemical space is enormous



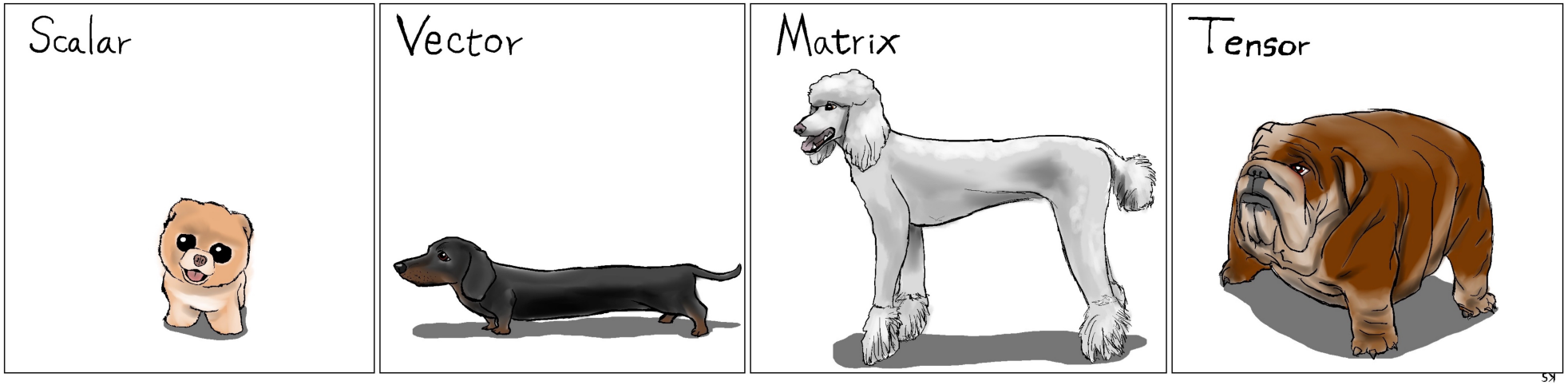
What is data science?

Interdisciplinary field using statistics, scientific computing, scientific methods, processing, visualisation, and algorithms to extract or extrapolate knowledge from potentially noisy, structured or unstructured data



Numerical data

Algorithms operate on multi-dimensional arrays of numerical data



x

1

x_i

$$\begin{bmatrix} 7 & 8 & 3 \end{bmatrix}$$

x_{ij}

$$\begin{bmatrix} 7 & 2 & 3 \\ 4 & 8 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

x_{ijk}

$$\begin{bmatrix} [1\ 7] & \cdots & [6\ 4] \\ \vdots & \ddots & \vdots \\ [5\ 6] & \cdots & [2\ 8] \end{bmatrix}$$

Image from https://karlstratos.com; note in Pytorch everything is a called a “tensor”

Function approximation

Much of machine learning is concerned with function approximation

Output

Reaction rate

Yield

Solubility

Band gap

etc...

$$y = f(x)$$

Input

Structure

Composition

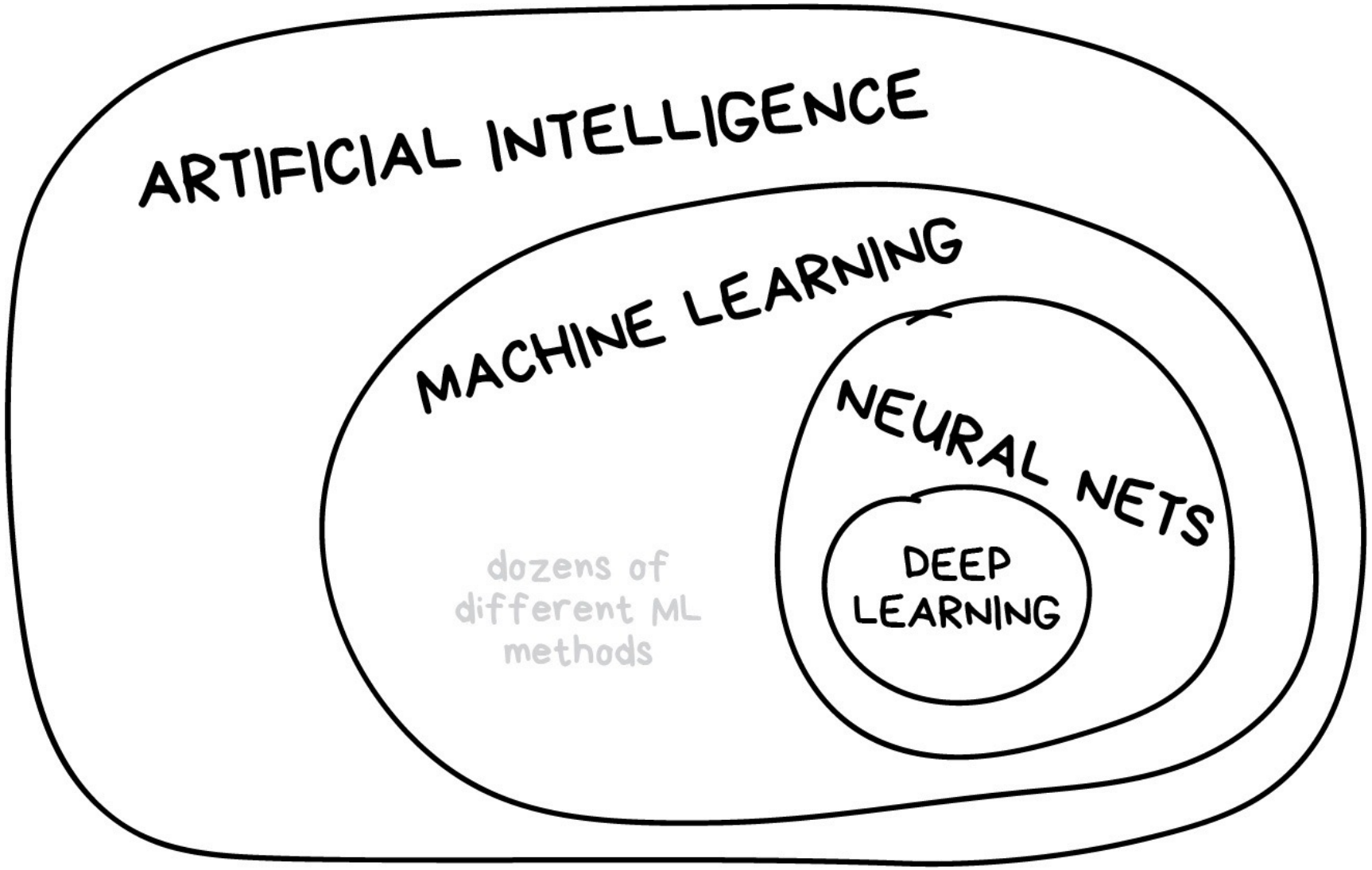
Catalyst

Temperature

etc...

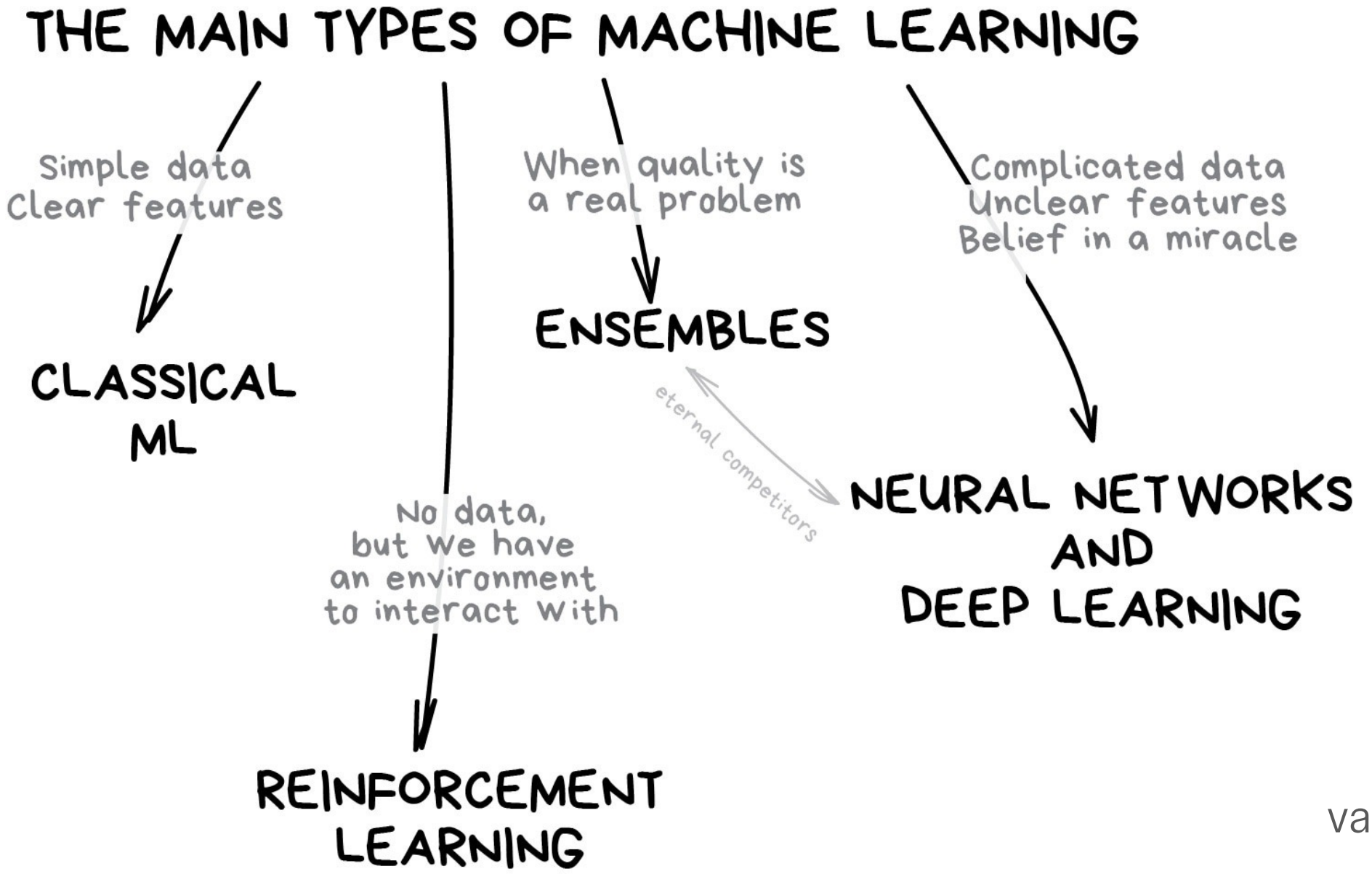
Field of artificial intelligence (AI)

Computational techniques that mimic human intelligence



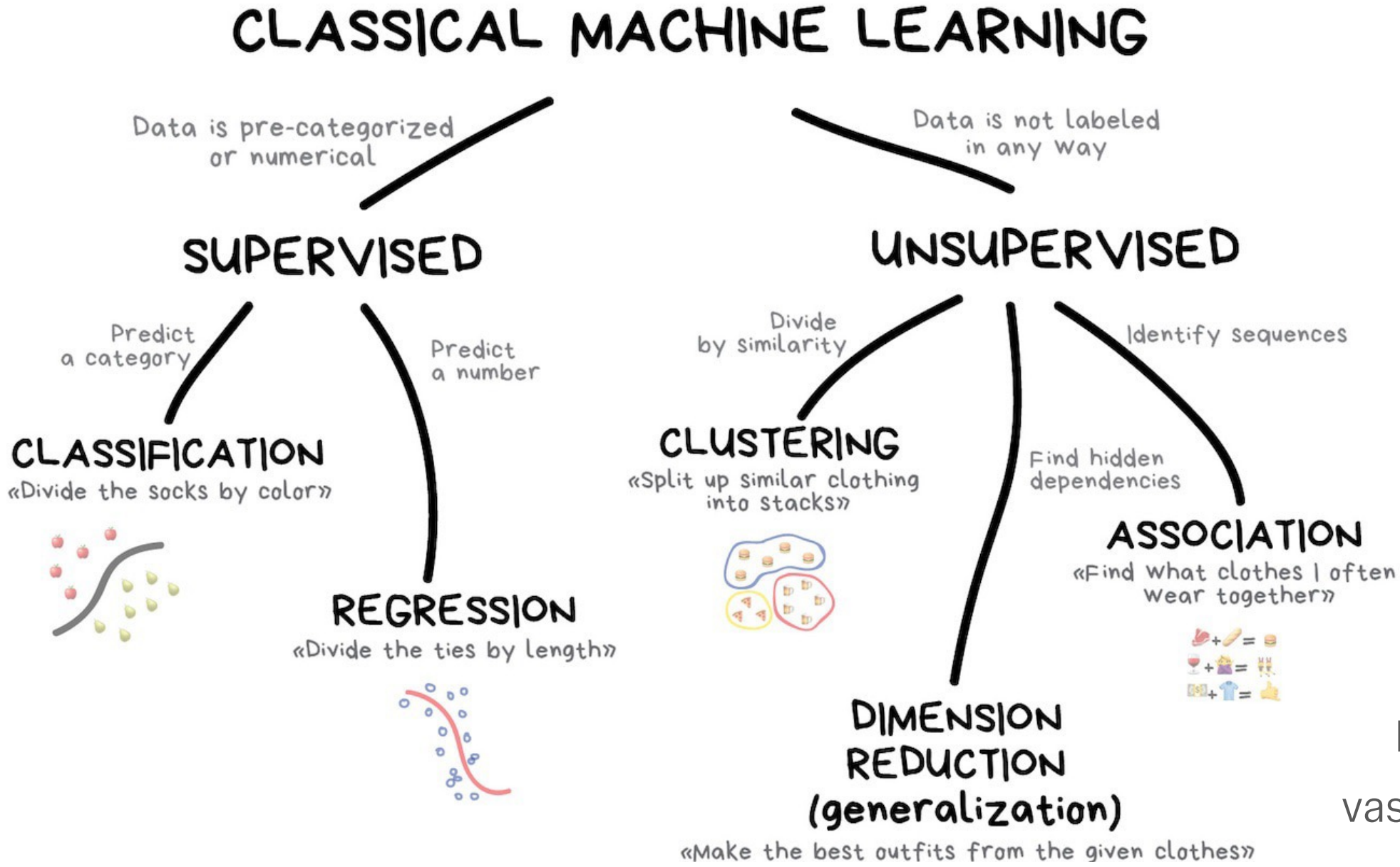
From vas3k.com

Machine learning



From vas3k.com

Classical machine learning



From vas3k.com

Unsupervised learning

Unsupervised learning is a key part of exploratory data analysis that helps to uncover intrinsic or latent descriptions of data

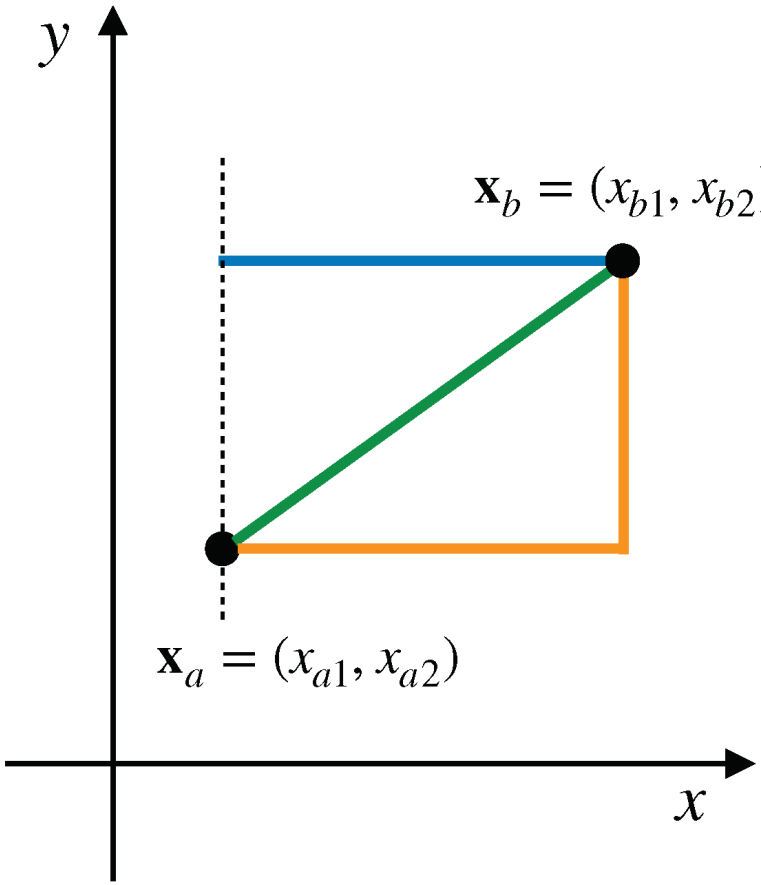
Computability: to find lower dimensional representation of our data to make it tractable to work with

Visualisation: to reveal hidden trends and relationships. E.g. identifying reaction coordinates or underly energy landscapes

Feature extraction: unsupervised learning can derive features for supervised learning – many unsupervised methods have closely aligned supervised equivalents

Distance in high dimensions

Minkowski distances: $d(\mathbf{x}_a, \mathbf{x}_b) = \left(\sum_{i=1}^{N} |x_{a,i} - x_{b,i}|^p\right)^{1/p}$



$p = 2$ Euclidean distance

$$\|\mathbf{x}_a - \mathbf{x}_b\|_2 = (|x_{a1} - x_{b1}|^2 + |x_{a2} - x_{b2}|^2)^{\frac{1}{2}}$$

$p = 1$ Manhattan distance

$$\|\mathbf{x}_a - \mathbf{x}_b\|_M = |x_{a1} - x_{b1}| + |x_{a2} - x_{b2}|$$

$p = \infty$ Chebyshev distance

$$\|\mathbf{x}_a - \mathbf{x}_b\|_\infty = \max\{|x_{a1} - x_{b1}|, |x_{a2} - x_{b2}|\}$$

Image from C. Fu and J. Yang, *Algorithms* 14, 54 (2021)

Distance in high dimensions

Distinction between distance measures

Euclidean: straight line between points. Use when data is dense and continuous. Features have similar scales

Manhattan: distance following gridlines. Use when data has different scales or grid-like structure

Chebyshev: maximum separation in one dimension. Use to emphasise largest difference; highlight outliers in feature space

Overview of Lecture 3

Unsupervised learning

A. Curse of dimensionality

B. Dimensionality reduction

C. Clustering

Overview of Lecture 3

Unsupervised learning

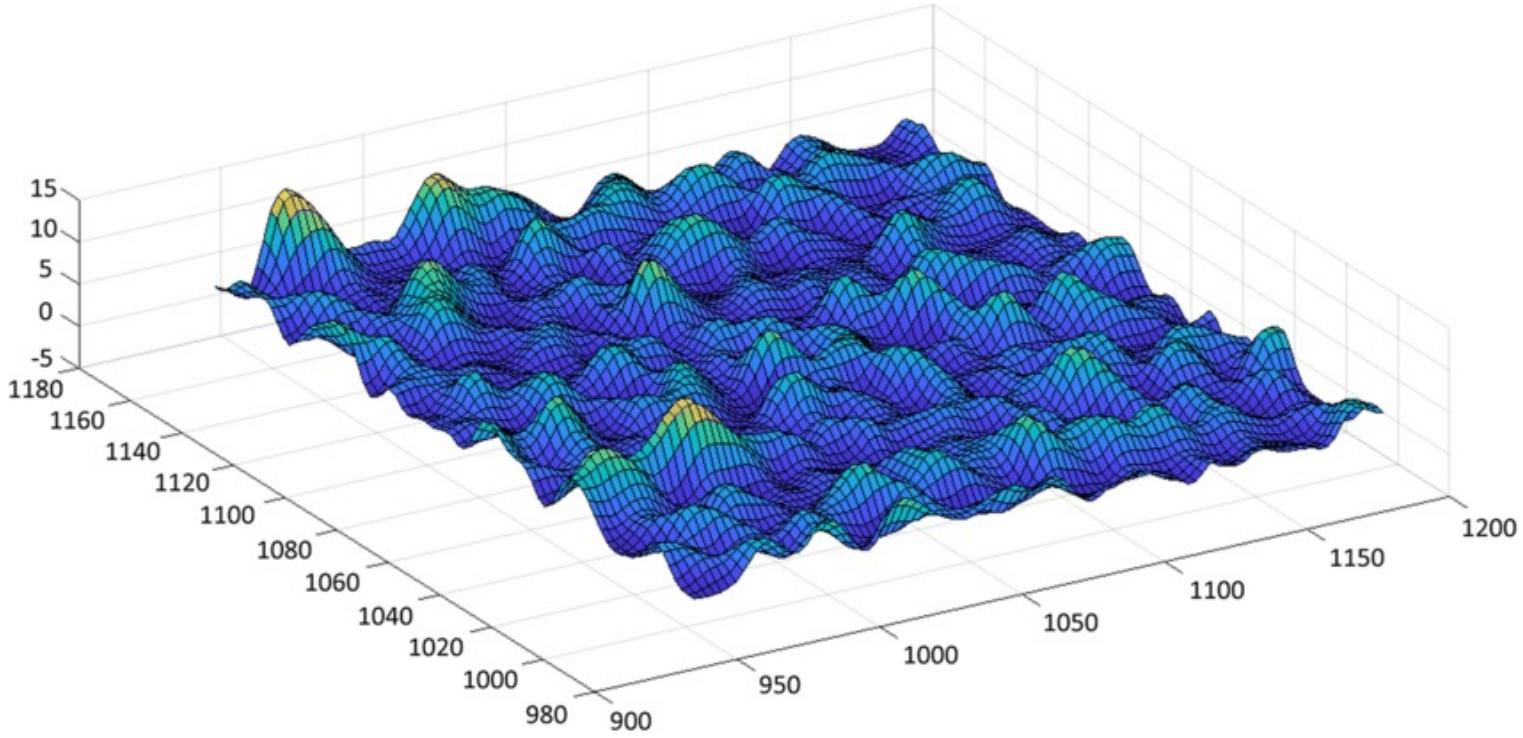
A. Curse of dimensionality

B. Dimensionality reduction

C. Clustering

Motivation

High-dimensional data refers to samples with many features that obscure the underlying landscape



High dimensional data can have many local minima, interaction terms, and effects taking place, with the number of parameters exceeding to the number of samples

Navigating high dimensional data

Example is transition state sampling in chemistry - “throwing ropes over mountain passes in the dark”

3N dimensions where N is the number of atoms

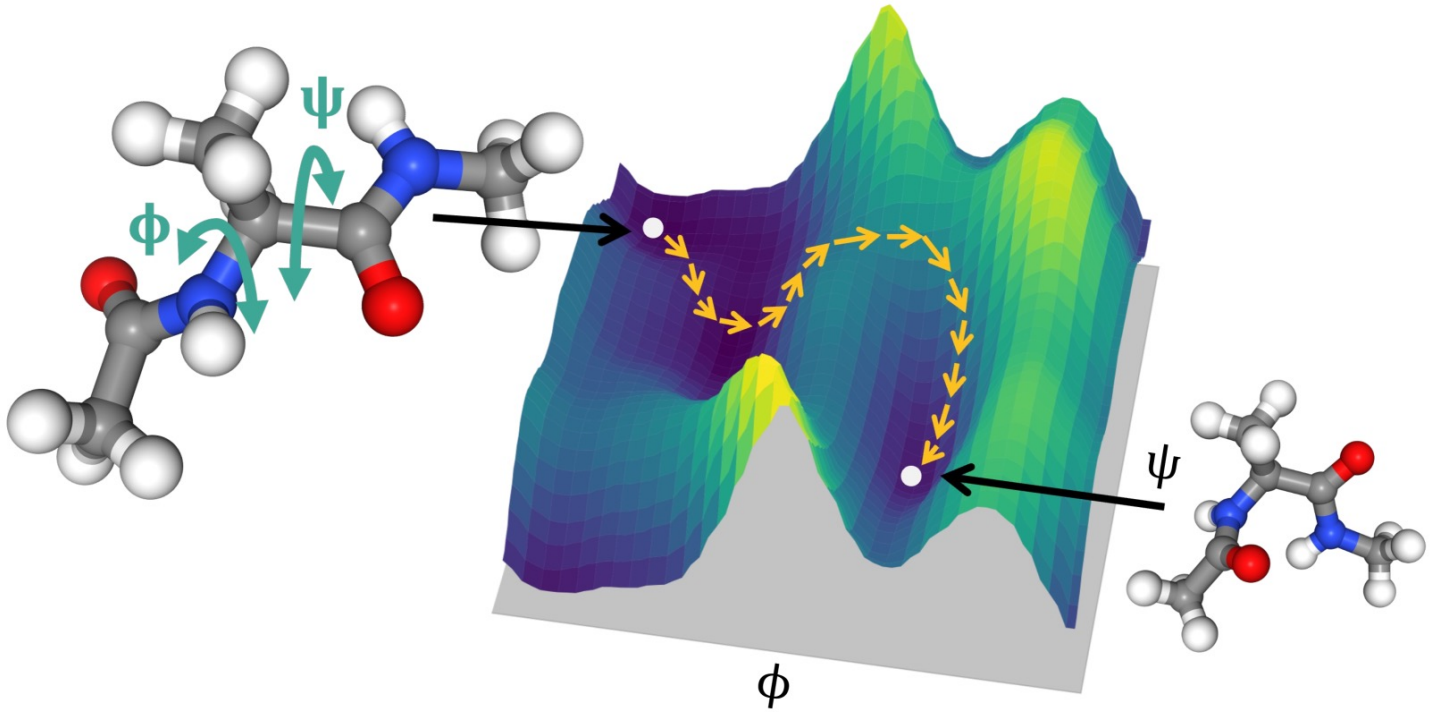
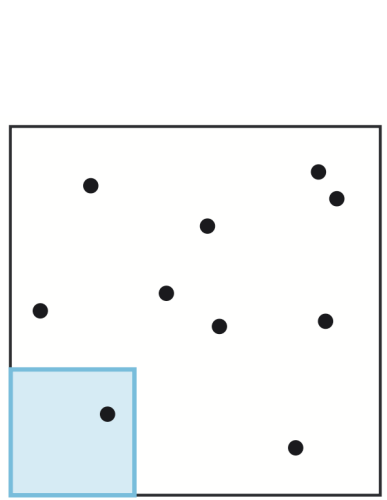


Image from: K. Seong et al. arXiv:2405.19961v2 (2024)

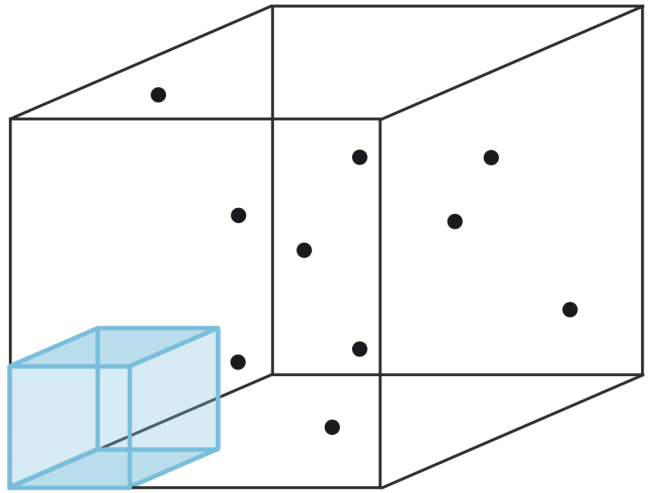
Curse of dimensionality

As the dimensionality increases, the volume of the space increases so fast that the available data becomes sparse

Num dimensions	Avg distance
2	0.52
3	0.66
8	1.13
100	4.08
1,000	12.9
1,000,000	408.25



2D



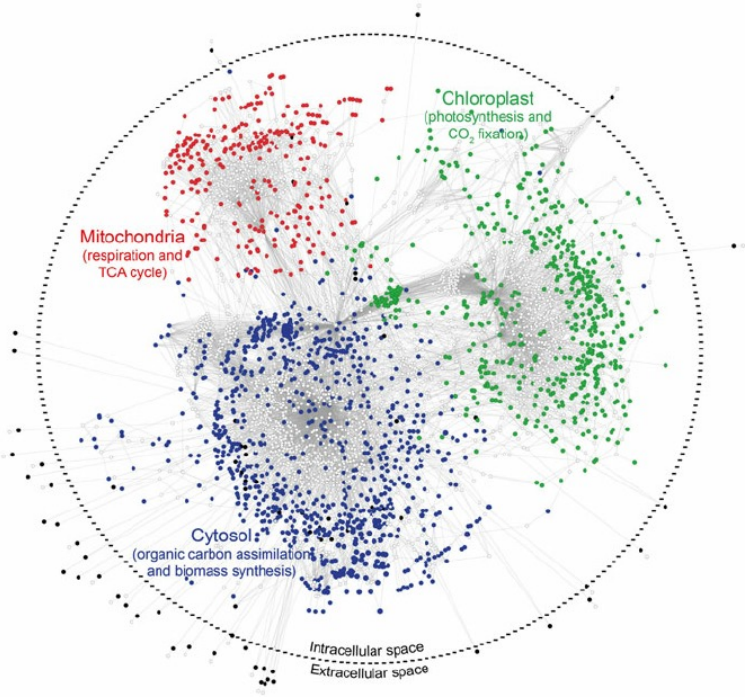
3D

See: https://mathworld.wolfram.com/HypercubeLinePicking.html

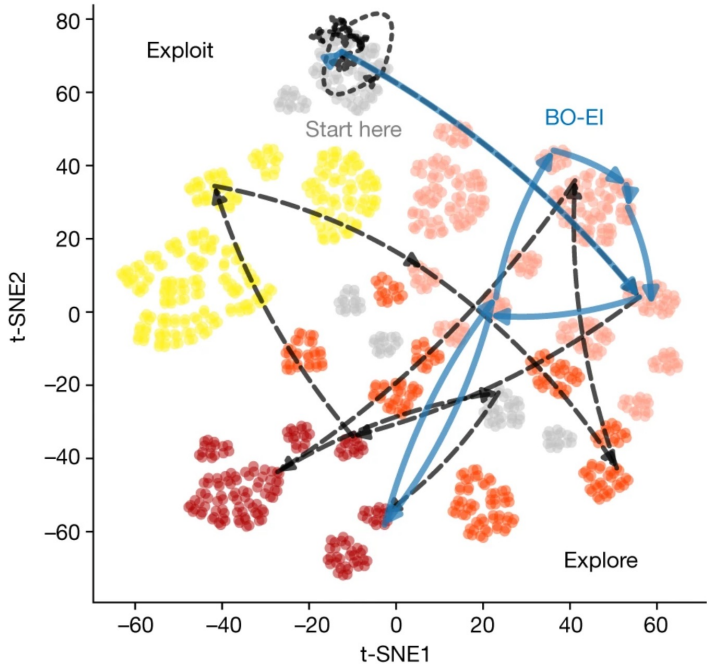
Chemical problems

Many problems in chemistry are high dimensional

Protein folding



Genomics



Reaction optimisation

Many others including microscopy data (images/volumetric), chemical combinatorics, time series (reaction kinetics), many-body wavefunctions, etc...

Overview of Lecture 3

Unsupervised learning

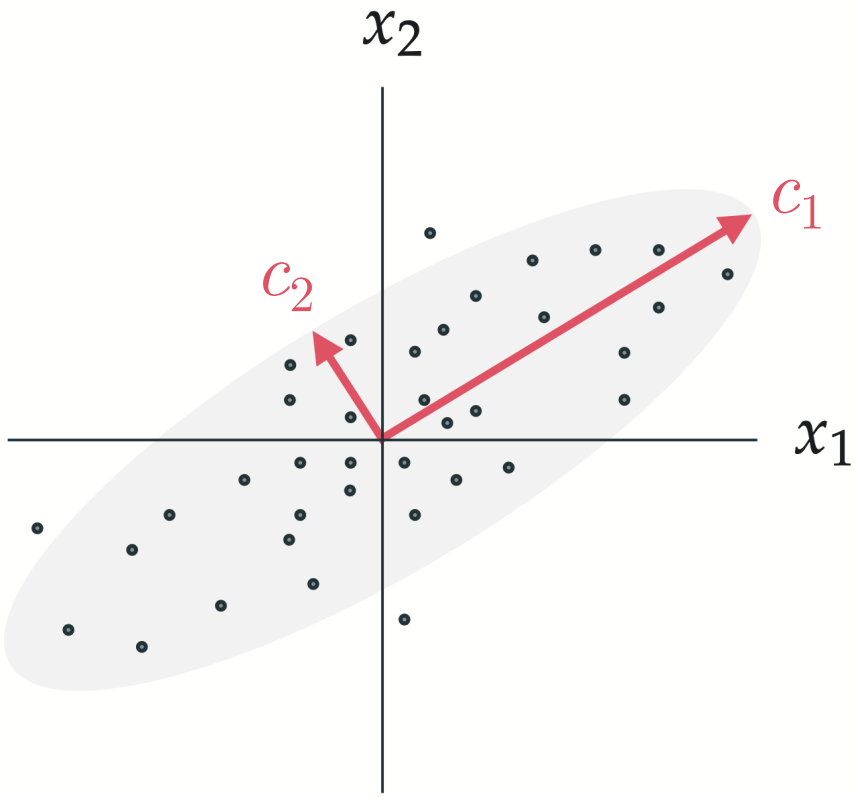
A. Curse of dimensionality

B. Dimensionality reduction

C. Clustering

Dimensionality reduction

A family of methods for identifying directions along which the data varies the most highly



Consider a 2D dataset

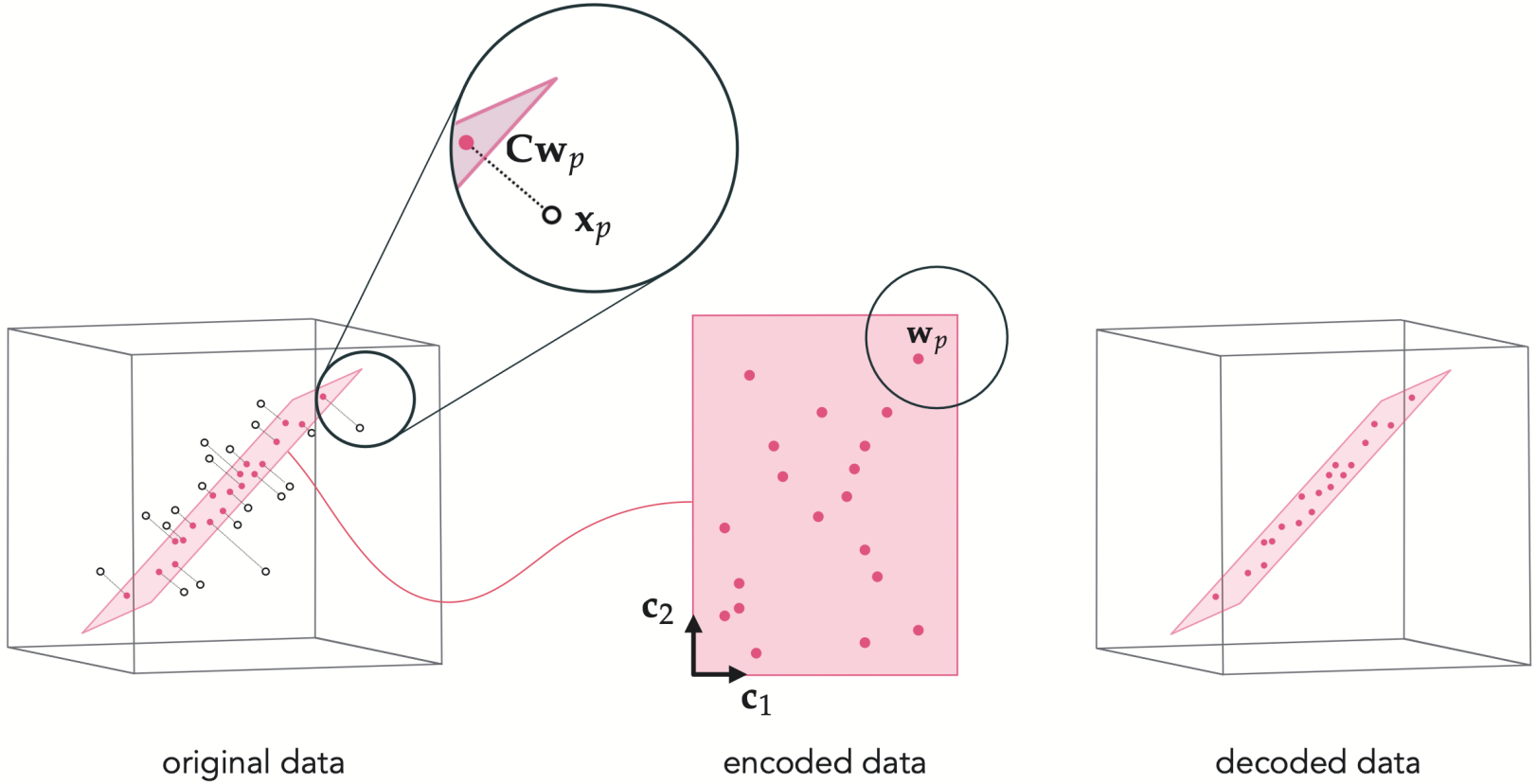
Most of the variation is along c_1

A smaller amount of variation is along c_2

Image from: https://github.com/neonwatty/machine_learning_refined

Principle component analysis

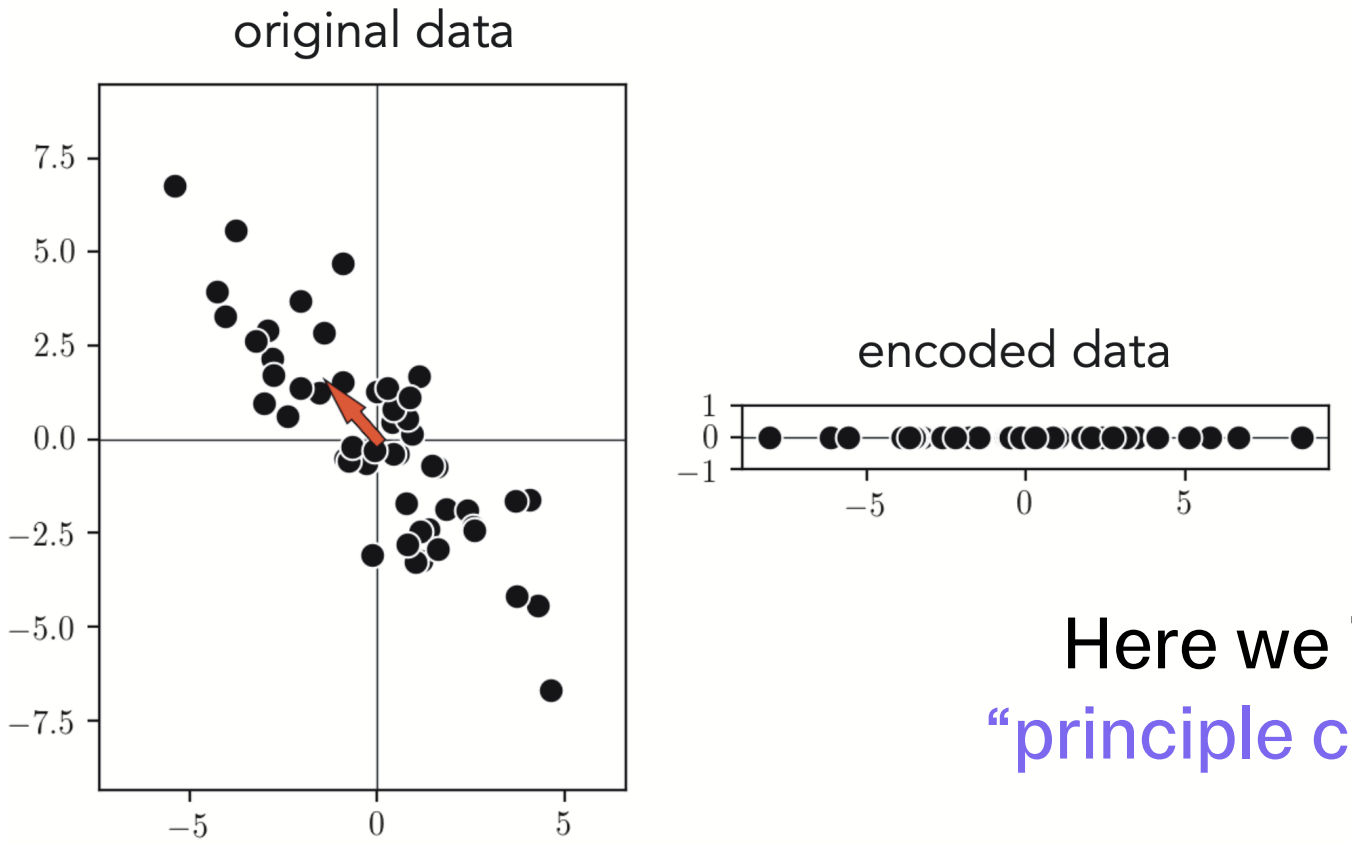
PCA is a method to identify linear combinations of the original features that explain as much variation in the data as possible



A form of **projection** - pick the hyperplane that lies closest to the data and project onto it

PCA example

Reduction of a two-dimensional dataset into one dimension



Here we identify 1
"principle component"

Warning: Dimensionality reduction results in information loss but this can be acceptable given the other benefits of dimensionality reduction

PCA in practice

Main steps to run PCA

1. **Preprocessing**: mean and centre the data, $\mathbf{X}$ (PCA requires the data to pass through the origin)

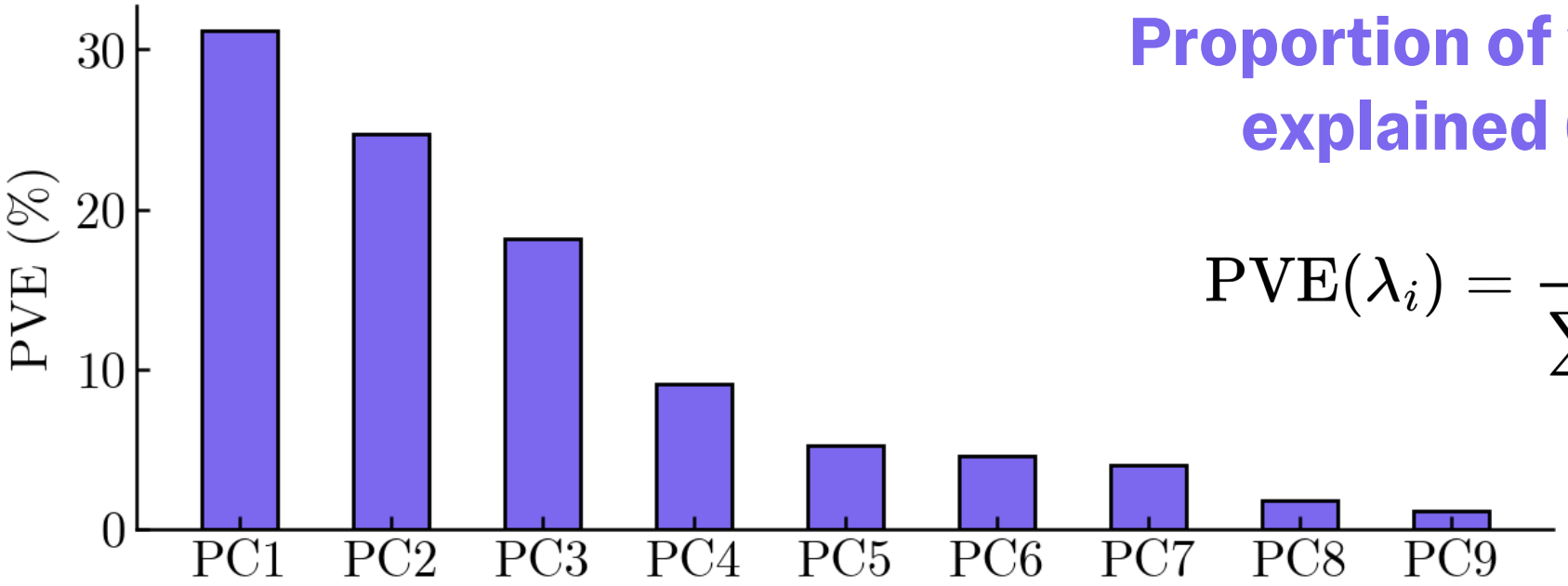
2. **Covariance**: compute the covariance matrix $\mathbf{\Sigma} = \mathbf{X}\mathbf{X}^{\mathrm{T}}/M$

3. **Eigen-decomposition**: calculate the eigenvalues and eigenvectors of $\mathbf{\Sigma} = \mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}}$

4. **Variance:** the eigenvector with the largest eigenvalue (λ_1) is the 1st "principle component" (k^{th} largest eigenvalue is the k^{th} PC)

Explaining variance

Each principle component explains some proportion of the total variance in the data



Proportion of variance explained (PVE)

$$\text{PVE}(\lambda_i) = \frac{\lambda_i}{\sum_{i=1}^{N} \lambda_i}$$

Only keep data projections onto principle components with large eigenvalues – you might lose some information but if the **eigenvalues are small**, you don't lose much

How many principle components should we use?

There's no simple answer to this question

Cross-validation: not available for this problem – CV allows us to estimate test error but in the unsupervised case there is no label

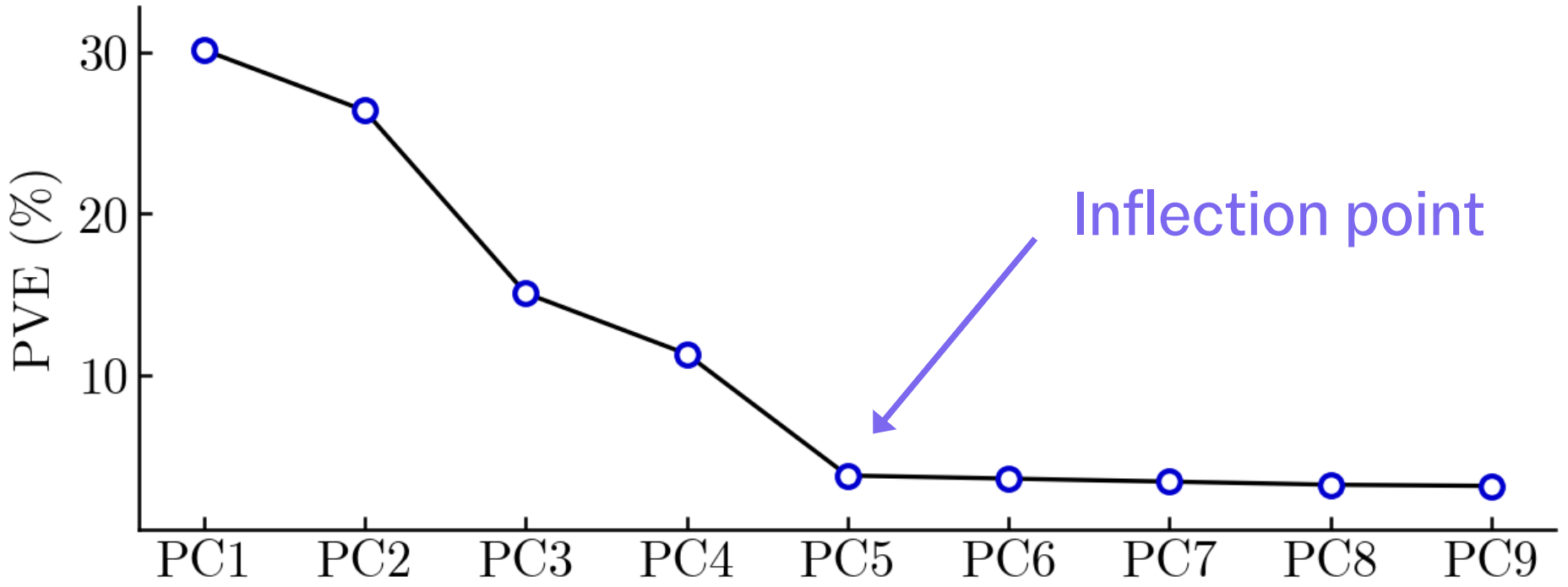
Feature reduction: the exception is when you're using PCA in supervised learning, here you can run CV and check the loss

Scree plots: an option but still arbitrary

Ultimately, no right answer – PCA is a tool for data exploration

Deciding the number of PCs – scree plots

Rule of thumb: stop at the elbow in the scree plot (k=5 here)

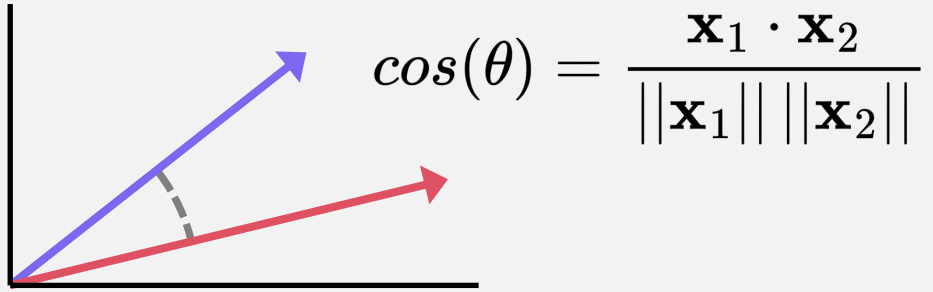


R. B. Cattell, *Multivariate Behavioral Research* 1, 2 (1966)

Example – clustering the periodic table

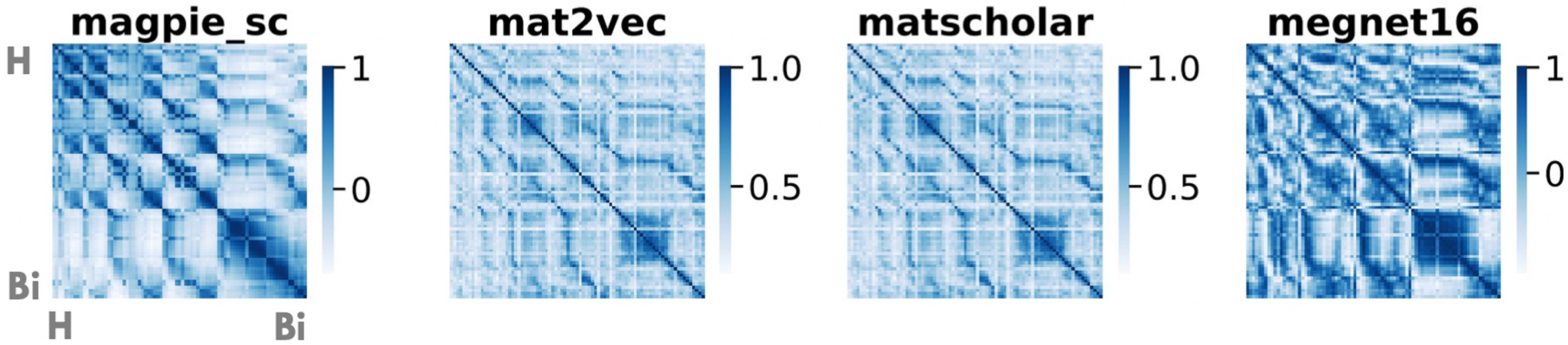
ML derived features for the elements – quantify distribution with distance, similarity, or correlation (e.g. Pearson)

Name	Dimension	Origin
Magpie[12]	22	Element properties
MatScholar[13]	200	Literature word embedding
Mat2Vec[14]	200	Literature word embedding
MEGnet[15]	16	Crystal graph neutral network
Oliynyk[16]	44	Element properties
Random_200	200	Random numbers
SkipAtom[17]	200	Structure graph pooling



$$cos(\theta) = \frac{\mathbf{x}_1 \cdot \mathbf{x}_2}{||\mathbf{x}_1||\,||\mathbf{x}_2||}$$

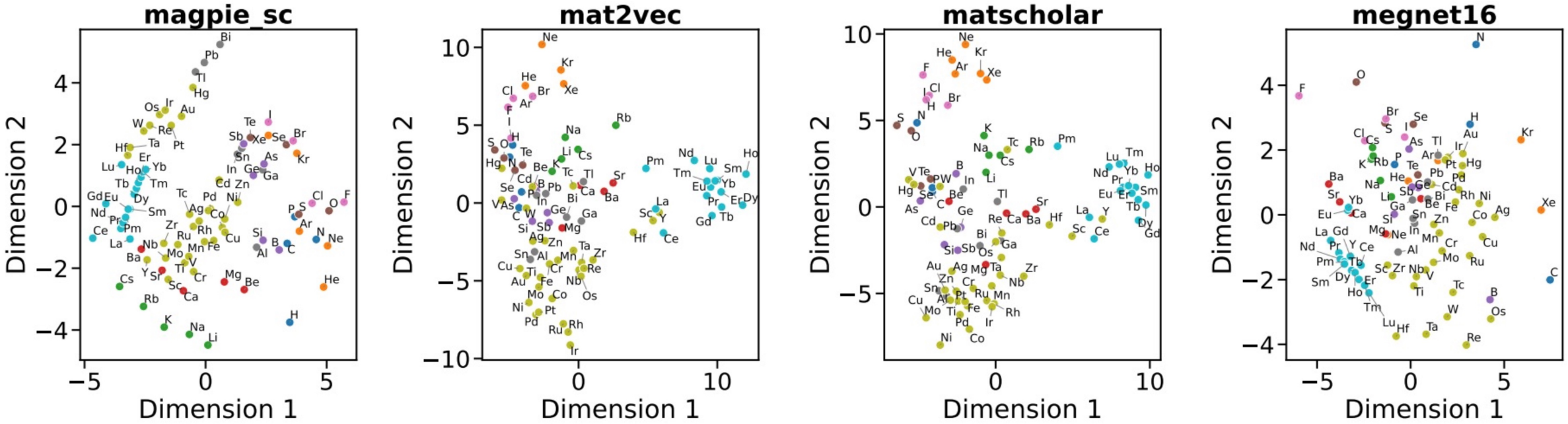
cosine similarity



A. Onwuli et al, *Digital Discovery* 2 1558 (2023)

Learned chemical similarity

Dimensionality reduction confirms a natural clustering of elements

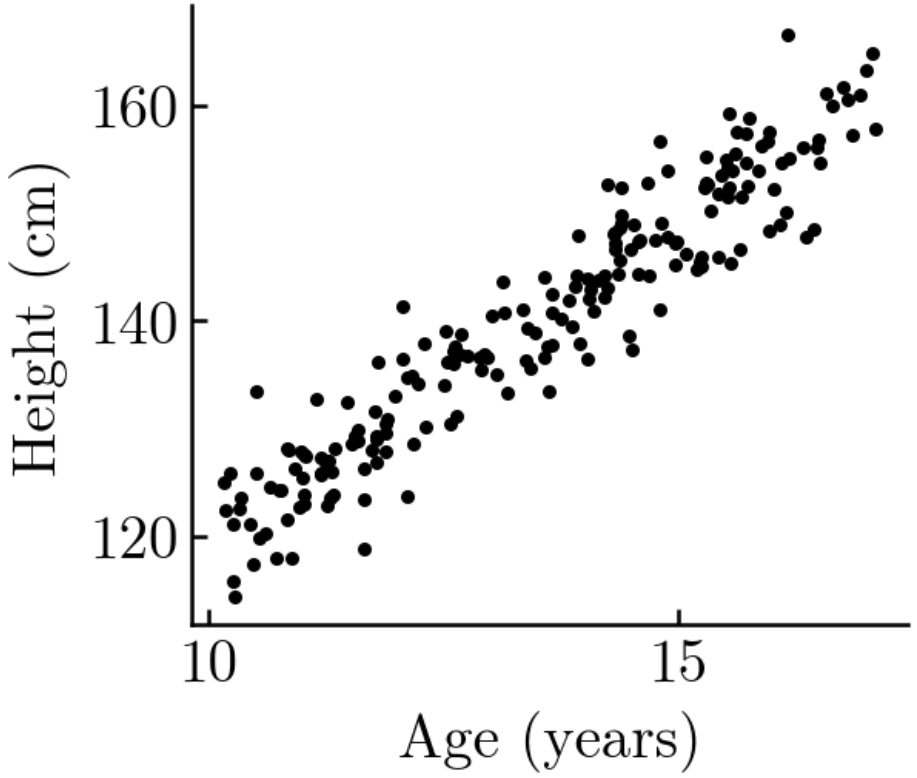


Principal component analysis with 2 dimensions

A. Onwuli et al, *Digital Discovery* 2 1558 (2023)

Principal components regression

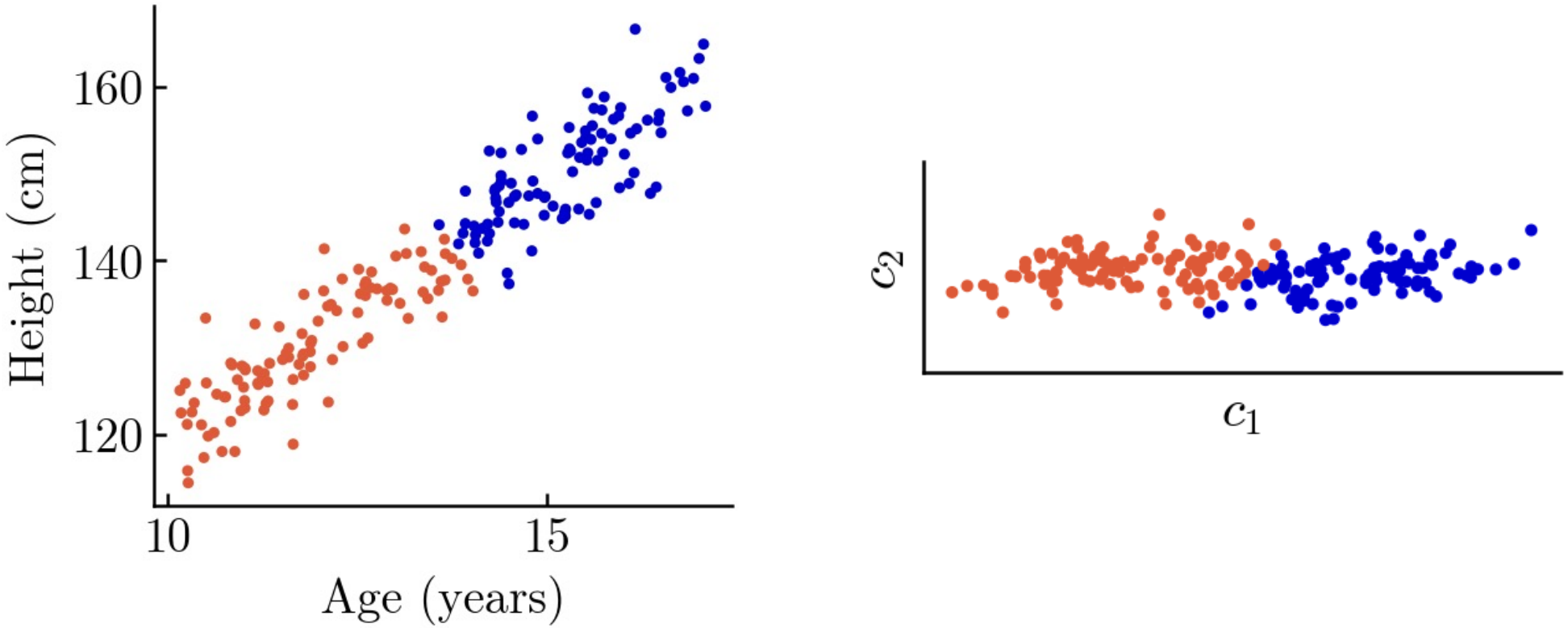
PCA can be used for feature engineering in supervised tasks if we suspect that features are highly correlated



Suppose we have two features, age and height which are correlated, instead of using both features we could just use the 1st principal component

Principal components regression

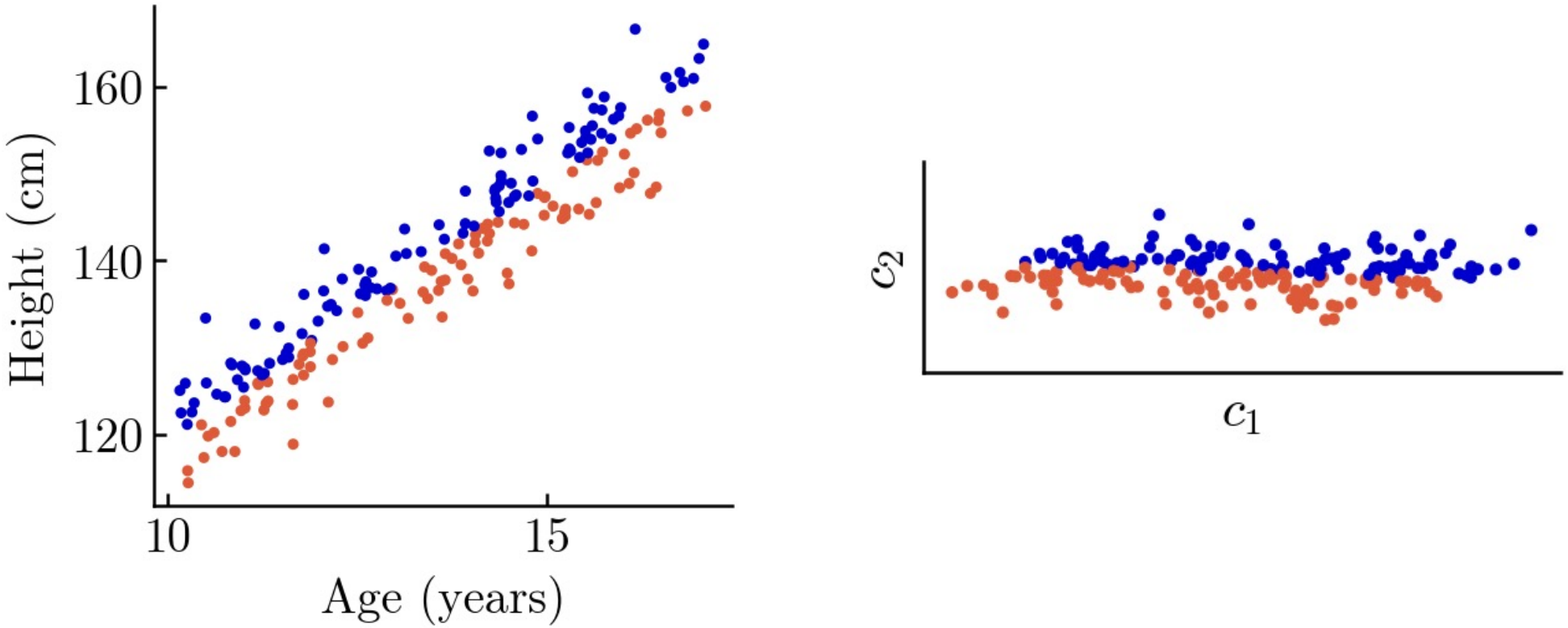
Will PCA work? Success depends on the outcome variable



Using the first principal component with logistic regression will likely work very well

Principal components regression

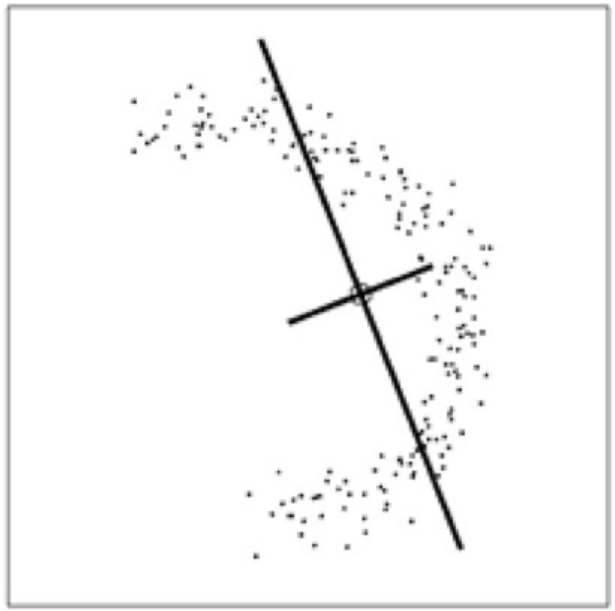
If the outcome variable is orthogonal to the PC things will go wrong



A logistic model with the first component will completely fail!

PCA captures linear variations

PCA works when the relationships between the features are linear (e.g. linearly correlated features)



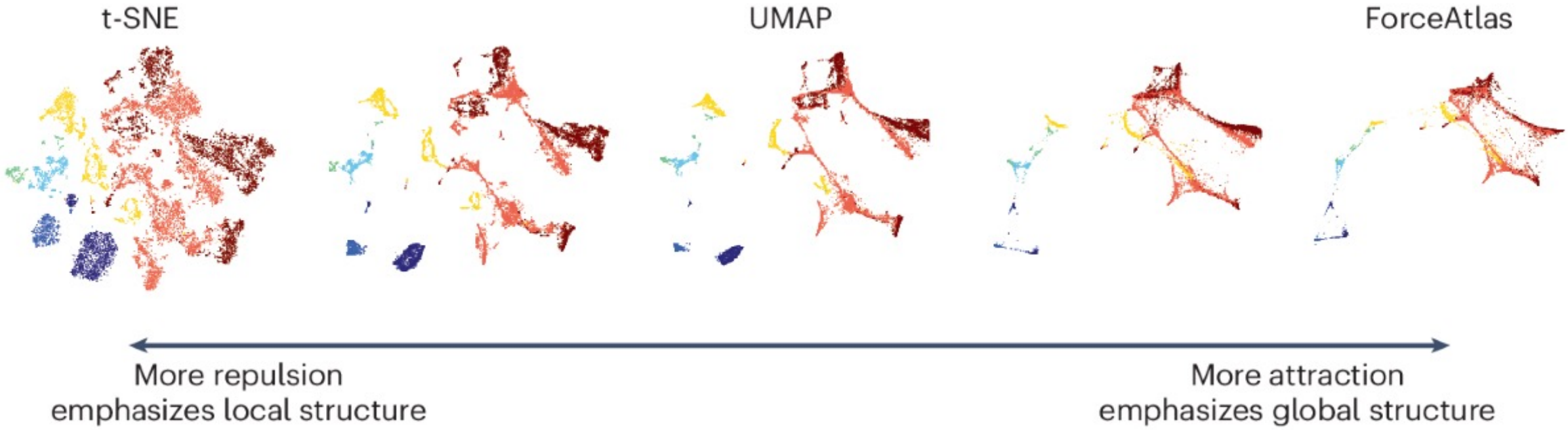
PCA

Principal curve

Here PCs are reasonable but don't capture the key trend – principal curves can be applied instead which generalises PCA by fitting 1D **curves instead of lines**

Non-linear dimensionality reduction

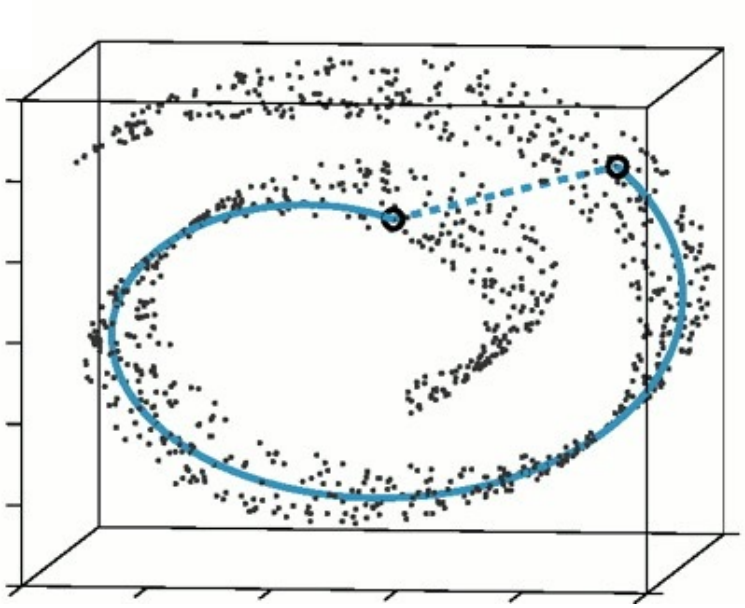
t-distributed stochastic neighbour embedding (t-SNE) and UMAP are two commonly used non-linear approaches



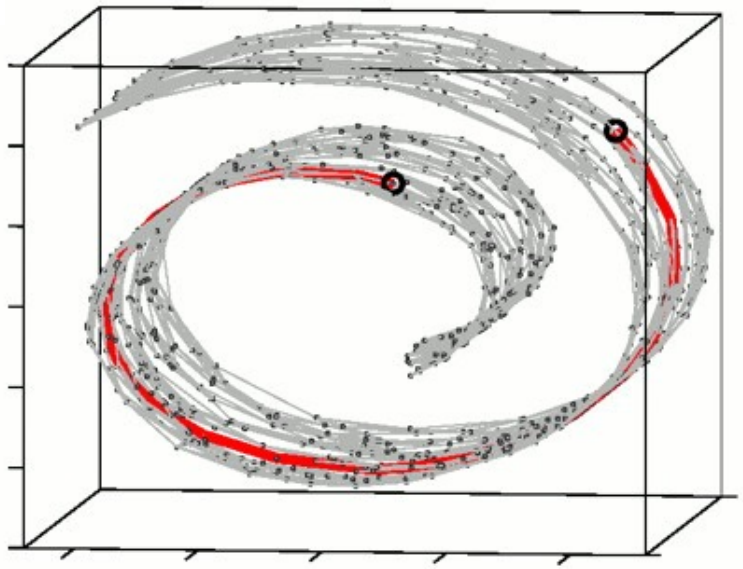
Mainly used for **visualisation purposes** – distances between points should be consistent with distances in the full dataset (Image from 10.1101/2024.04.26.590867)

Manifold learning

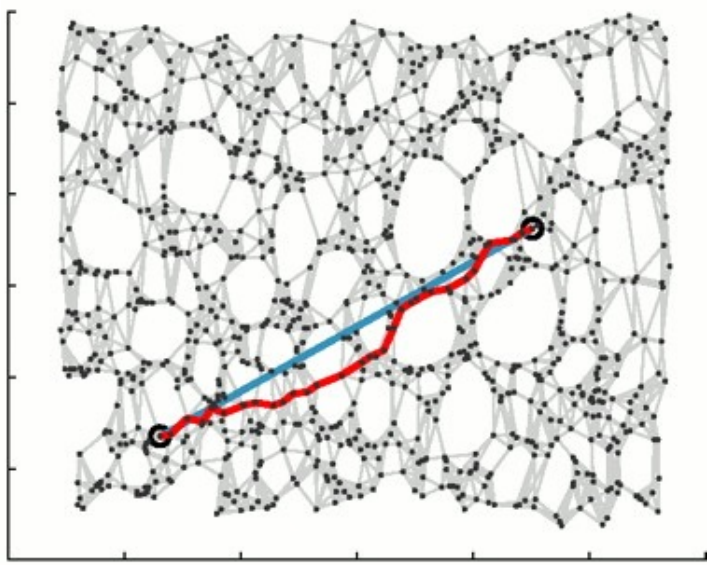
A versatile approach for estimating the intrinsic geometry of data



Goal: Use geodesic distance between points, (relative to manifold)



Estimate manifold using graph – distance given by shortest path



Embed onto 2D so Euclidean distance approximates graph distance

J. B. Tenenbaum, V. de Silva, J. Langford, *Science* 240, 5500 (2000)

Overview of Lecture 3

Unsupervised learning

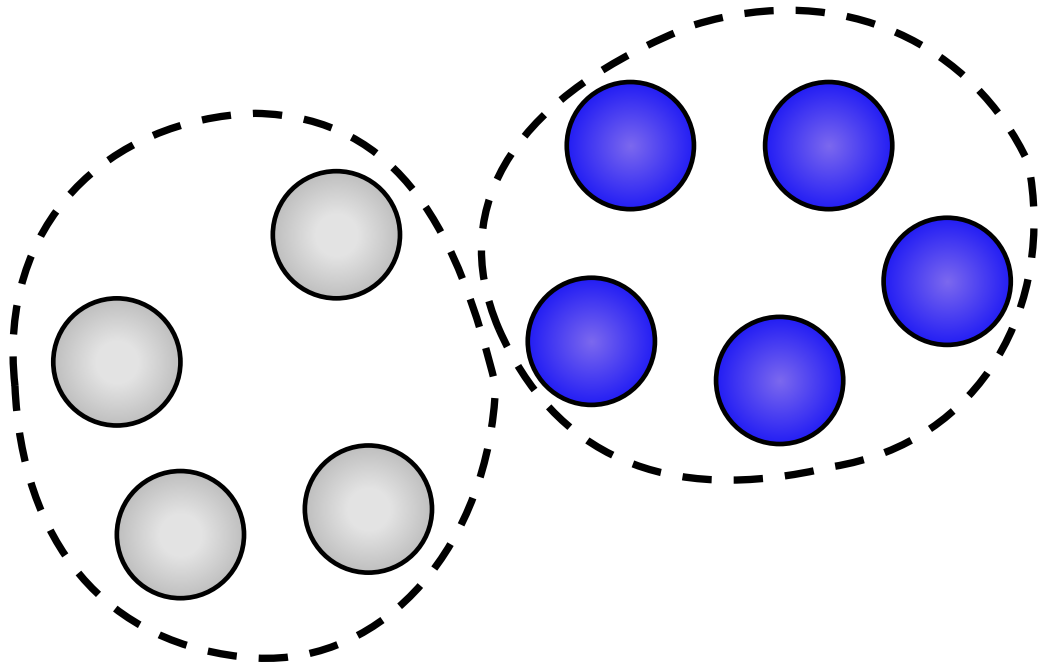
A. Curse of dimensionality

B. Dimensionality reduction

C. Clustering

k-means clustering

Unsupervised model groups data into clusters, where k is the number of clusters identified



Place N observations into K sets

$$\mathbf{C} = \{C_1, C_2, \ldots, C_K\}$$

Datapoints within a cluster should be similar

"Sur la division des corps matériels en parties" H. Steinhaus (1957)

k-means clustering

Main components of a *k*-means model

1. **Initialisation:** choose the number of clusters *k* that you want to identify in your dataset. Randomly assign each point to a cluster

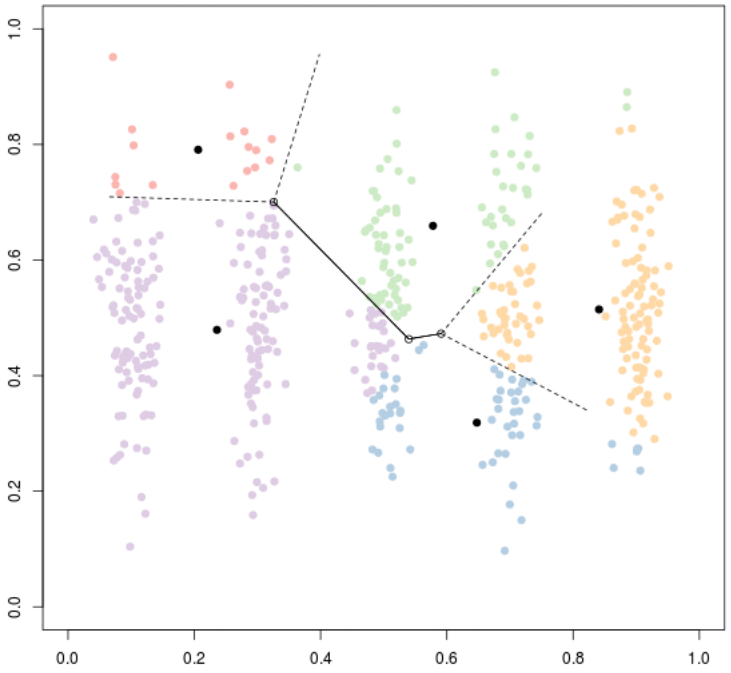
2. **Distance metric**: how the object/data is separated in multi-dimensional space, e.g. Euclidean or Manhattan distance

3. **Centroid**: calculate the centroid (mean location of each cluster)

4. **Assignment**: each point is reassigned to the nearest centroid based on distance. Iterate until the clusters stop changing

k-means clustering

Unsupervised model groups data into clusters, where K is the number of clusters identified



Place N observations into K sets

$$\mathbf{C} = \{C_1, C_2, \ldots, C_K\}$$

Minimise within cluster variance

$$W(C_k) = \sum_{i \in C_k} ||\mathbf{x}_i - \bar{\mathbf{x}}_k||^2$$

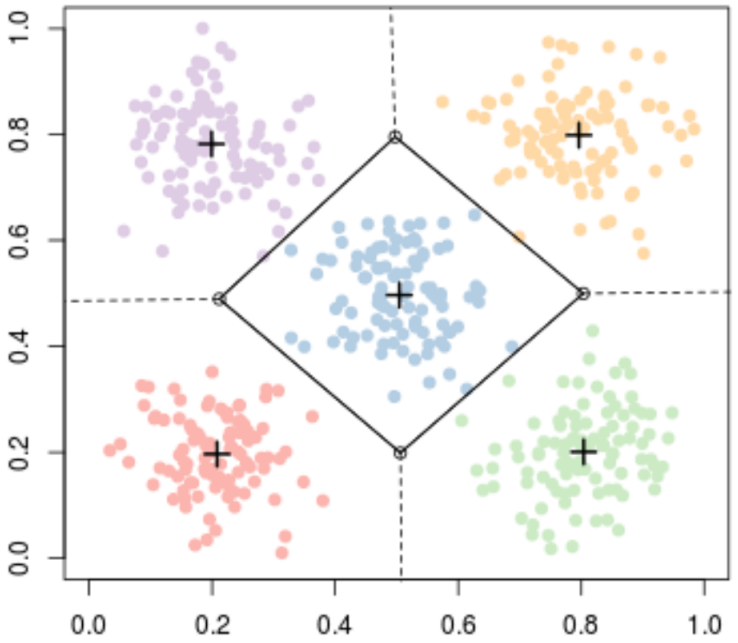
Cluster centroid

An iterative algorithm is used to minimise cluster variance

Animation from https://feakonometrics.hypotheses.org/19156

k-means clustering

Unsupervised model groups data into clusters, where K is the number of clusters identified



Place N observations into K sets

$$\mathbf{C} = \{C_1, C_2, \ldots, C_K\}$$

Minimise within cluster variance

$$W(C_k) = \sum_{i \in C_k} ||\mathbf{x}_i - \bar{\mathbf{x}}_k||^2$$

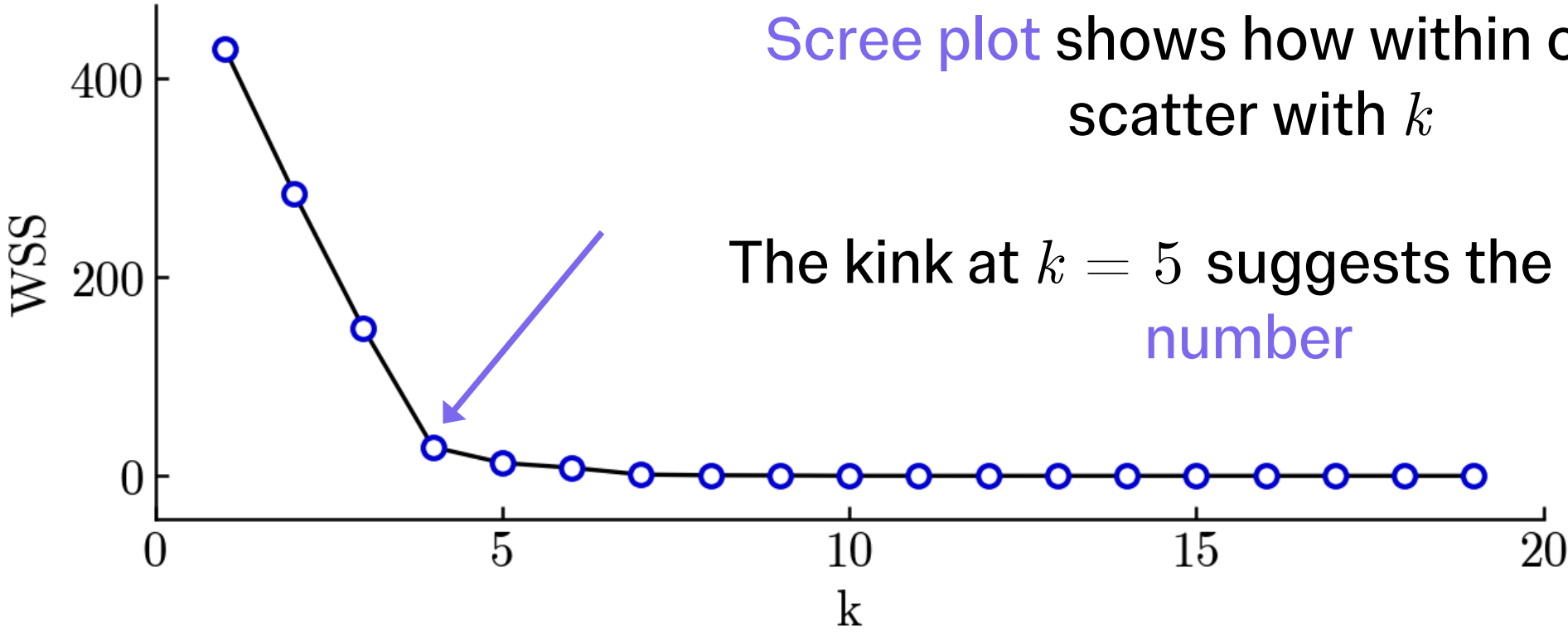
Cluster centroid

An iterative algorithm is used to minimise cluster variance

Animation from https://feakonometrics.hypotheses.org/19156

k-means clustering

k is a hyperparameter. How many clusters to choose?



Scree plot shows how within cluster scatter with k

The kink at $k = 5$ suggests the optimal number

As k increases, the similarity within a cluster increases but in the limit of $k = n$, each cluster is only one data point

k-means clustering

The strength of *k*-means is simplicity, but it has limitations

1. **No dual membership:** even if a data point falls at a boundary, it is assigned to one cluster only

2. **Clusters are discrete**: no overlap or nesting is allowed between clusters

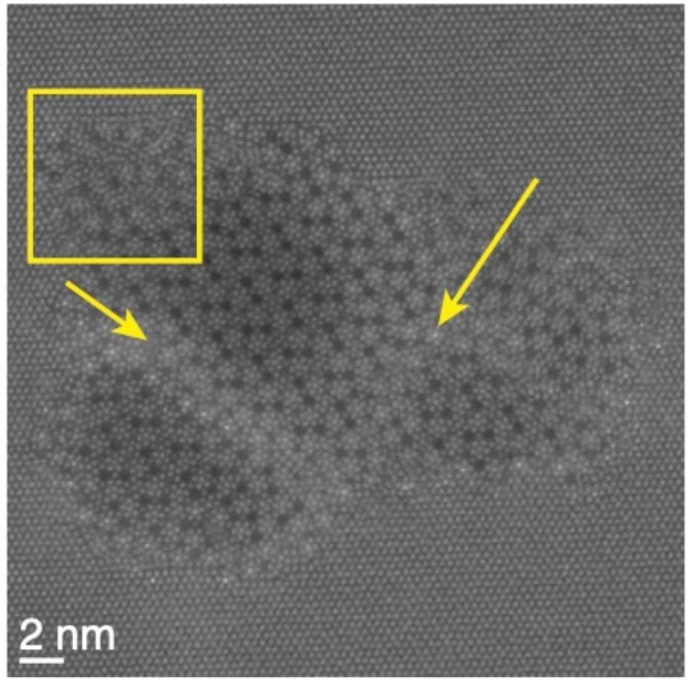
3. **Unpredictability**: *k*-means algorithm is random and only discovers local minima.

Extended techniques such as spectral clustering compute the probability of membership in each cluster

k-means application: microscopy

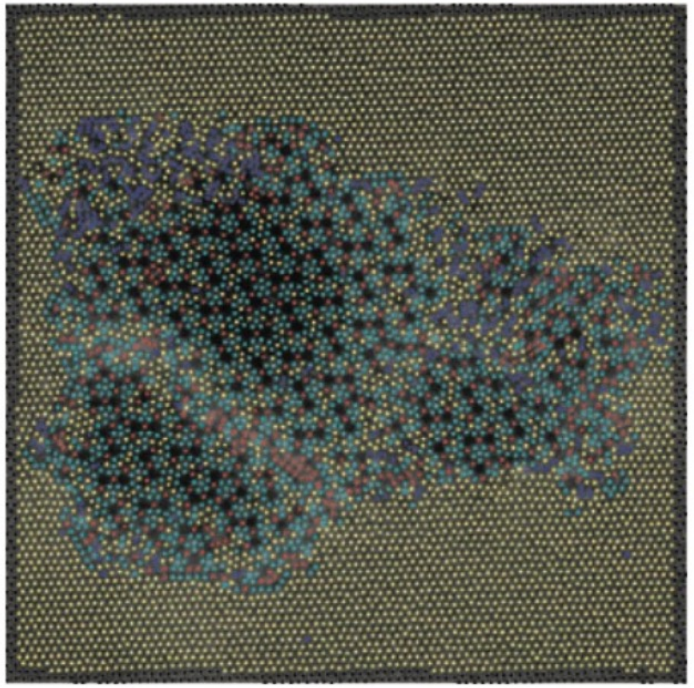
Clustering in STEM images of multicomponent (Mo-V-Te-Ta) oxides

Original data



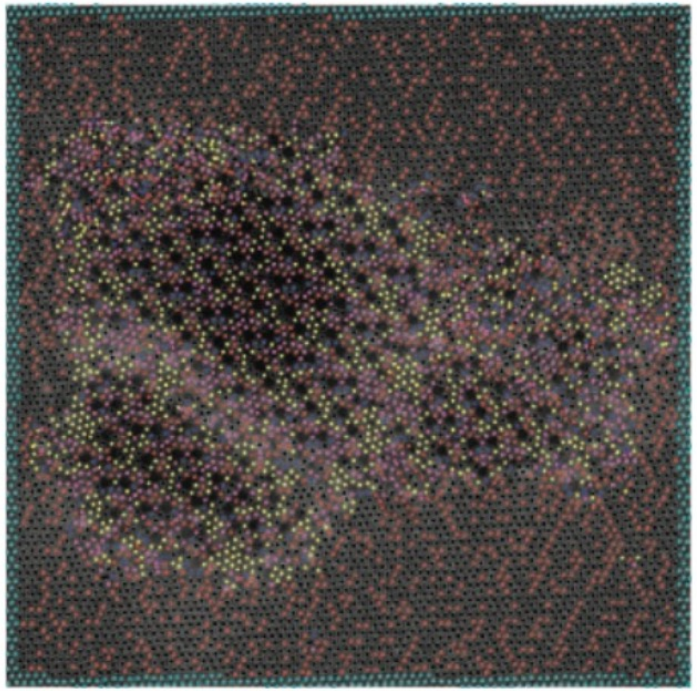
k-means

(k=4, Euclidean distance)



k-means

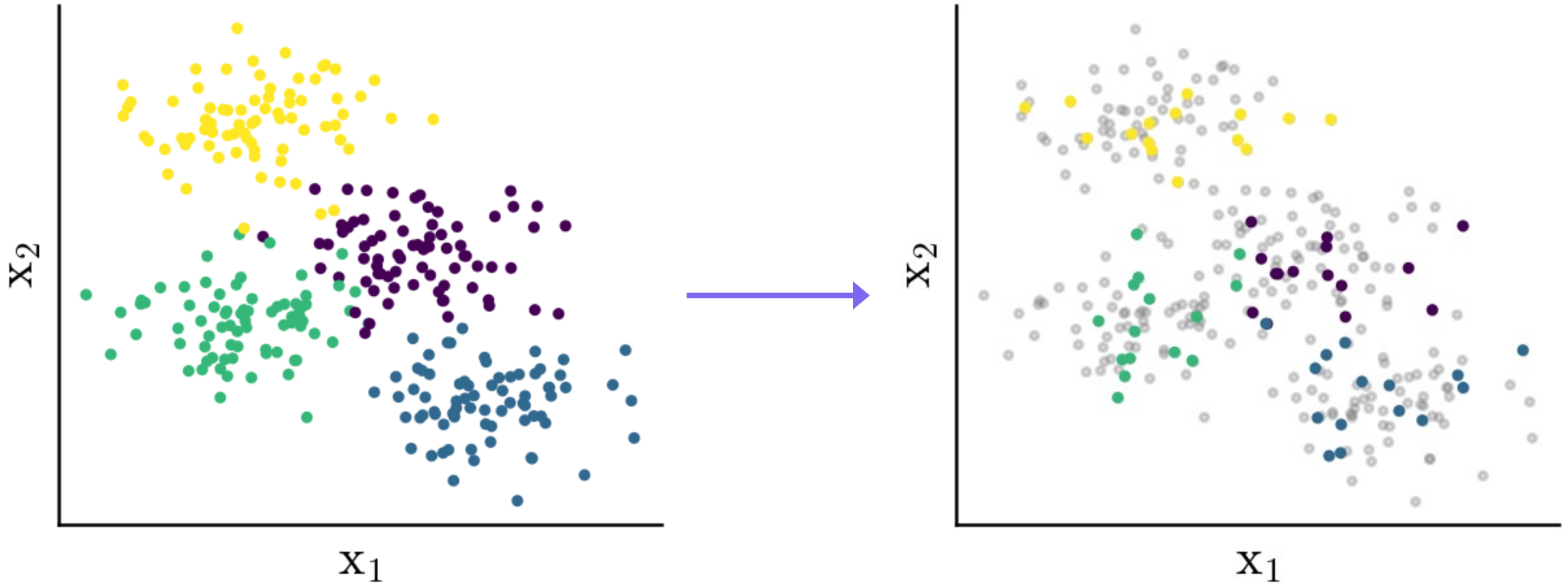
(k=4, angle metric)



A. Belianinov et al., *Nat. Commun.* 6, 7801 (2015

Subsampling from clusters

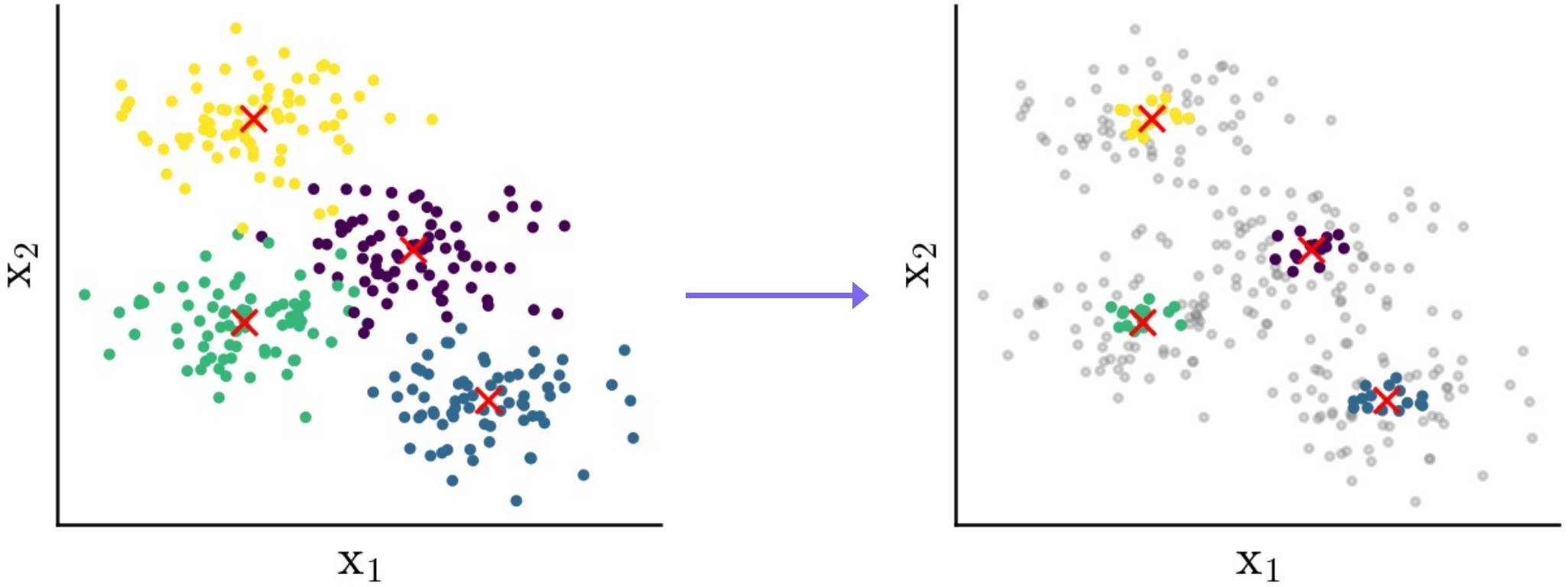
Large datasets can be expensive to process – subsampling helps reduce size while keeping the dataset representative



Random sampling selects points randomly from each cluster and leads to unbiased dataset size reduction

Centroid proximity sampling

Selects points closest to the cluster centre – retains the most representative samples in the dataset



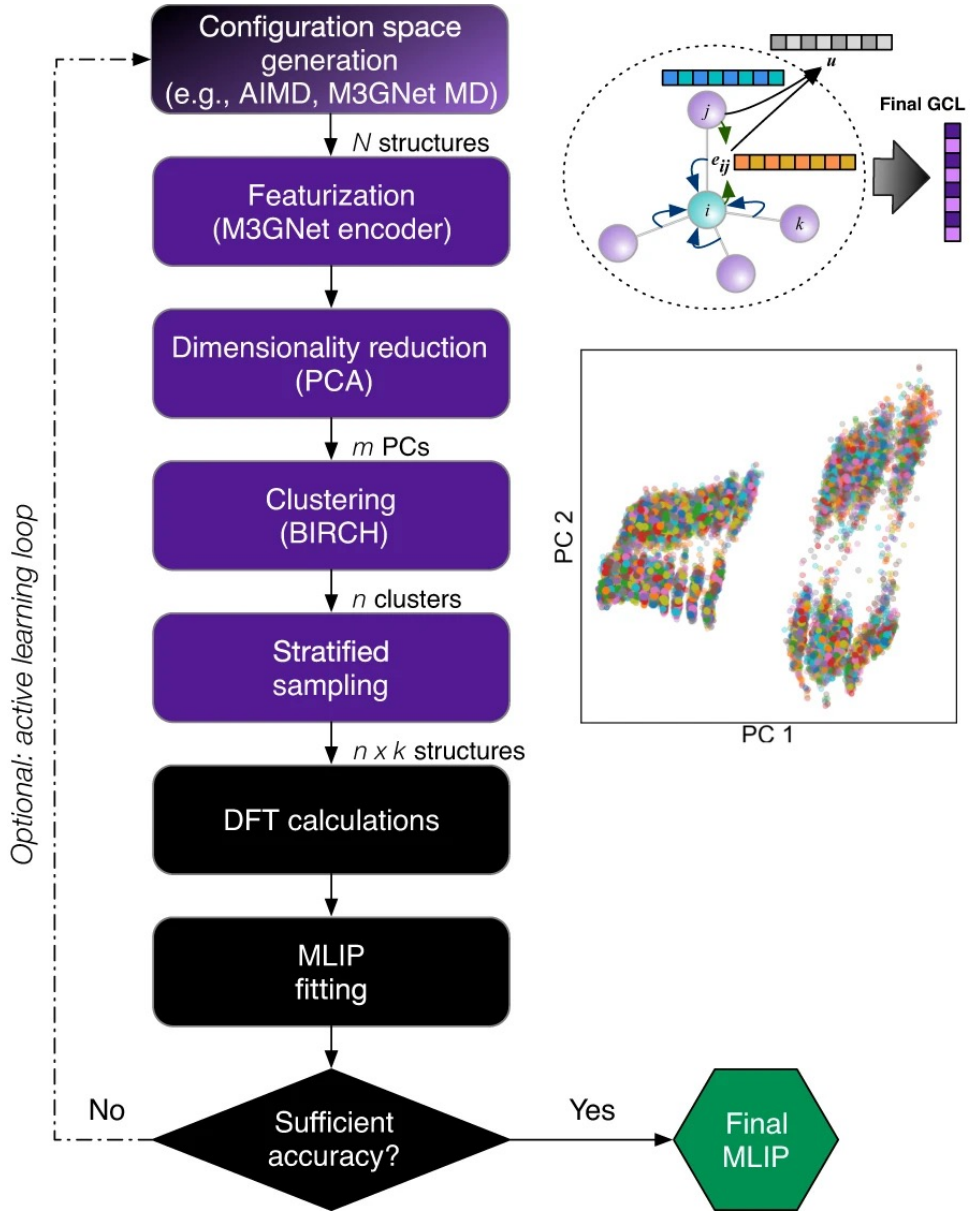
Note that unrepresentative points are not sampled at all, leading to a loss in dataset diversity – most often applied when number of samples per cluster = 1

Cluster sampling for machine learned potentials

Uses stratified sampling to select *k* structures from each cluster

- ***k* = 1: select structure closest to centroid**
- ***k* > 1: sort by distance from centroid; take regular samples**
- ***k* > # samples: all points selected (with duplicates removed)**

J. Qi, T. W. Ko, B. C. Wood, T. A. Pham, S. P. Ong, *npj Computational Materials* 10, 43 (2024)



Many clustering approaches exist

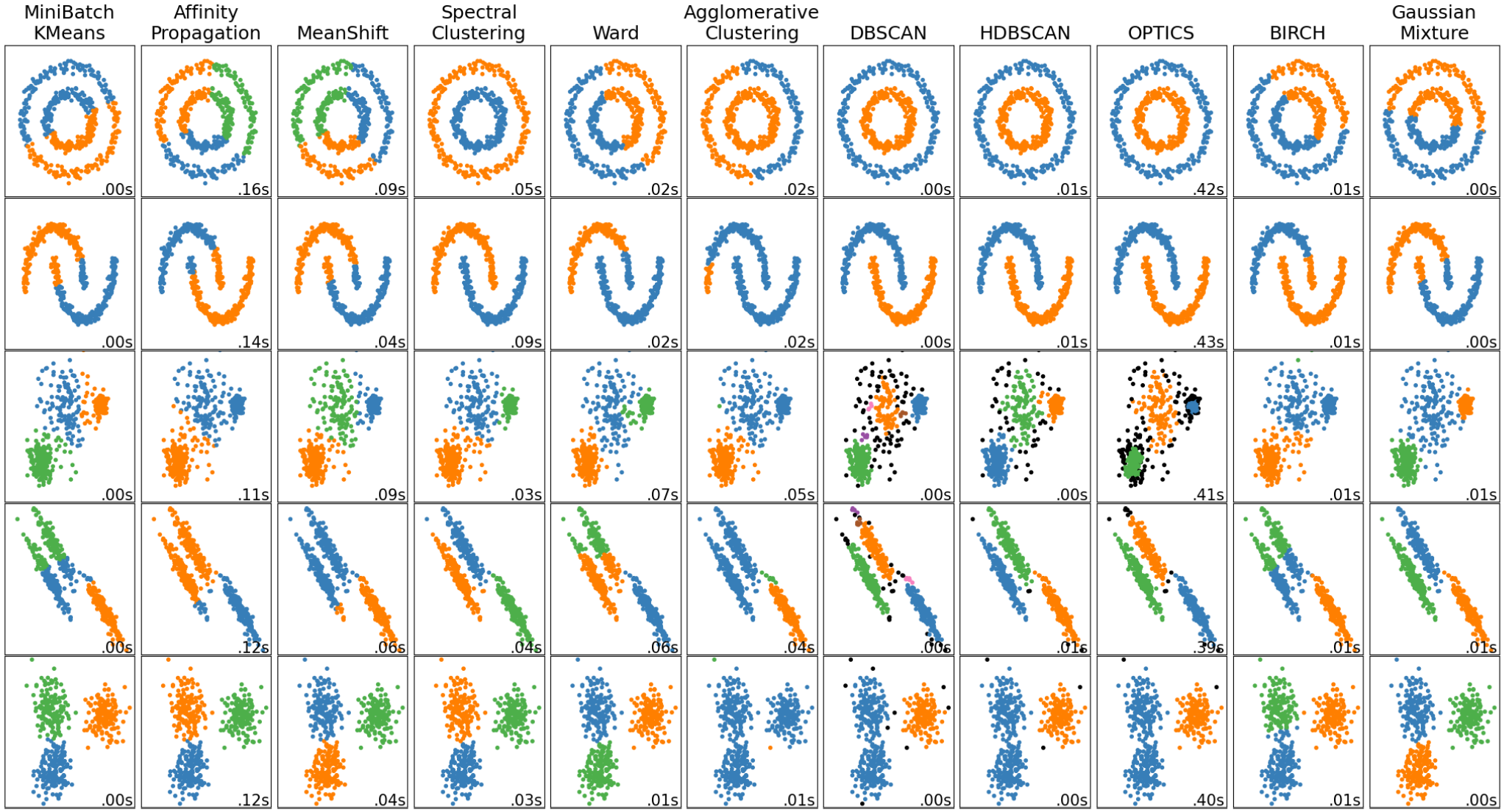
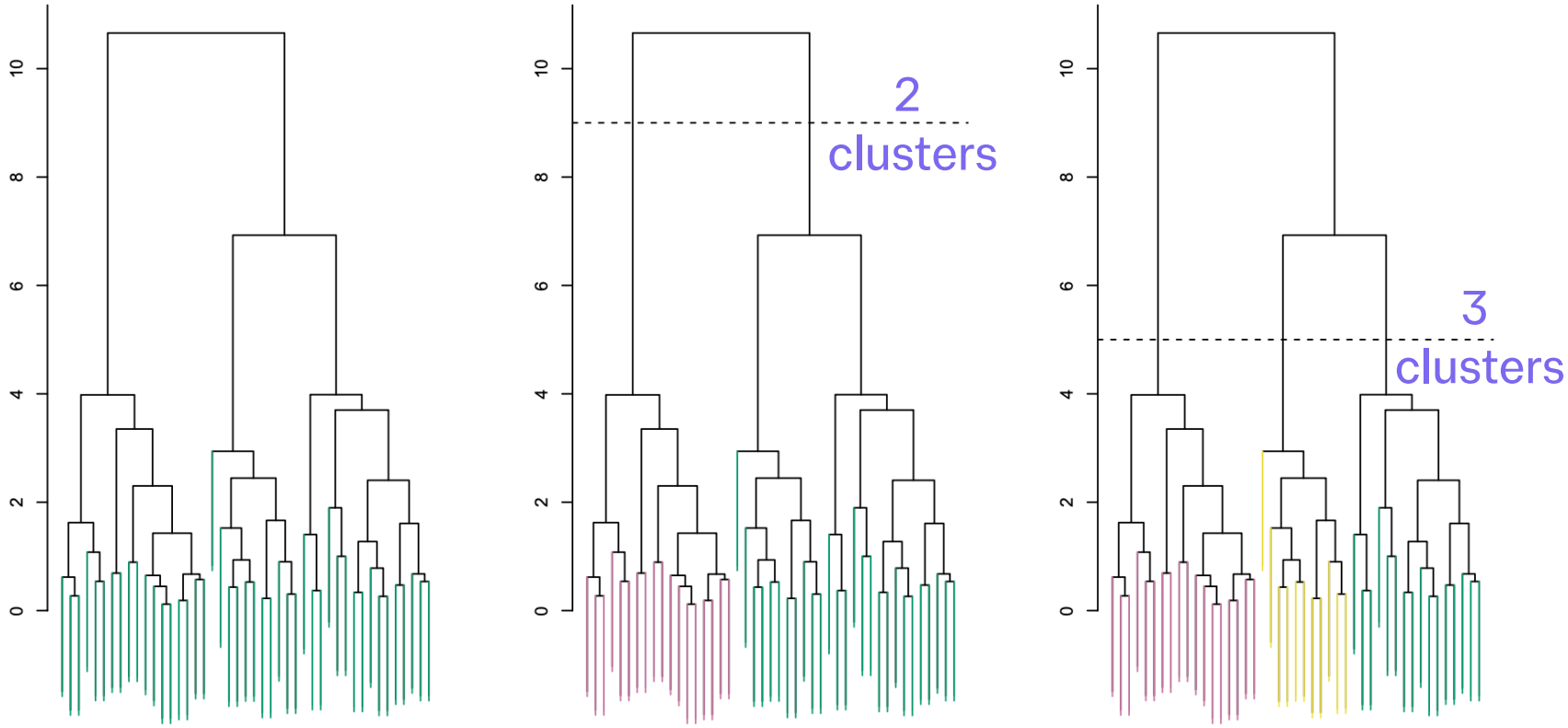


Image from: https://scikit-learn.org/stable/modules/clustering

Hierarchical clustering

Hierarchical clustering avoids the need to specify k in advance and is entirely deterministic – useful for nested data



Agglomerative hierarchical clustering progressively connects nearby points into a dendrogram which can be cut to select the number of clusters

Notes on clustering

Clustering can be a useful tool but should be used wisely

1. **Sensitivity:** small decisions can have large effects on the results (e.g. data standardisation and distance metrics)

2. **Exploration**: recommended to perform clustering with a range of different choices to see what patterns consistently emerge. This can include clustering subsets of the data

3. **Reporting**: take care when discussing clustering results – they should not be taken as absolute truth but be a starting point for hypothesis development and further study on independent data

Lecture outcomes

1. Assess which types of unsupervised learning approaches are suitable for particular problems

2. Explain how principal component analysis works and when it can be applied

3. Explain the use of clustering for unsupervised and supervised learning problems

Slide credits

Many ideas borrowed from Aron Walsh and Sophia Yaliraki (Imperial)