3000788 Intro to Comp Molec Biol

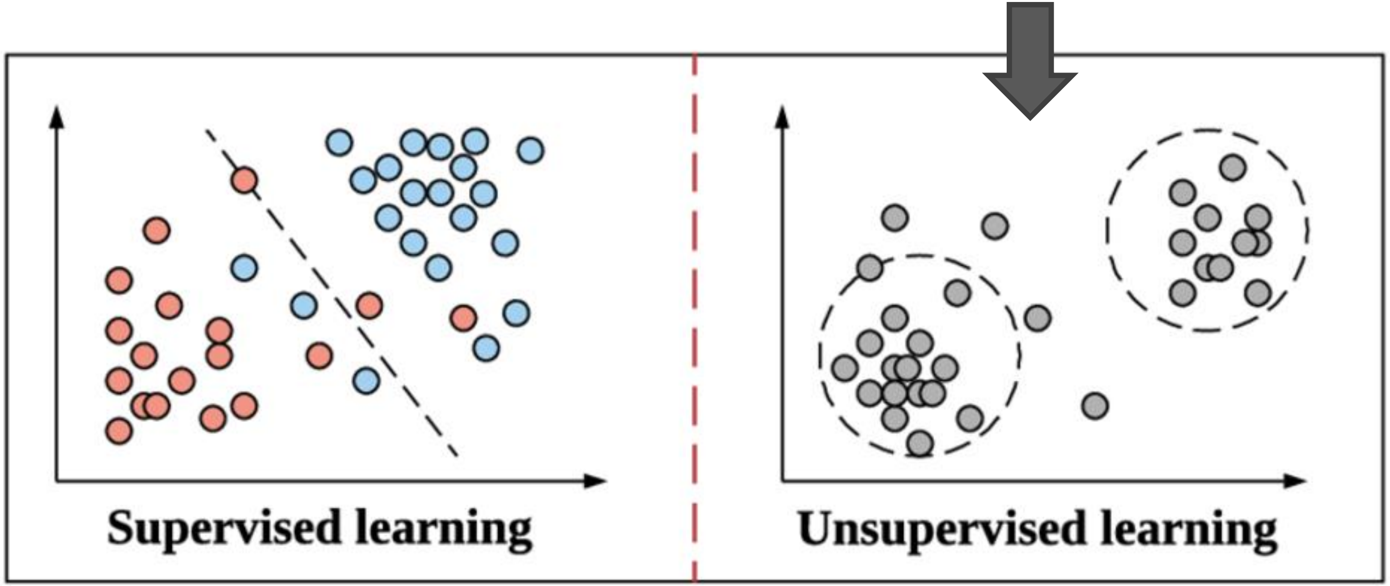
Lecture 25: Unsupervised learning

November 10, 2022

Sira Sriswasdi, PhD

- Research Affairs
- Center of Excellence in Computational Molecular Biology (CMB)
- Center for Artificial Intelligence in Medicine (CU-AIM)

Machine learning paradigms

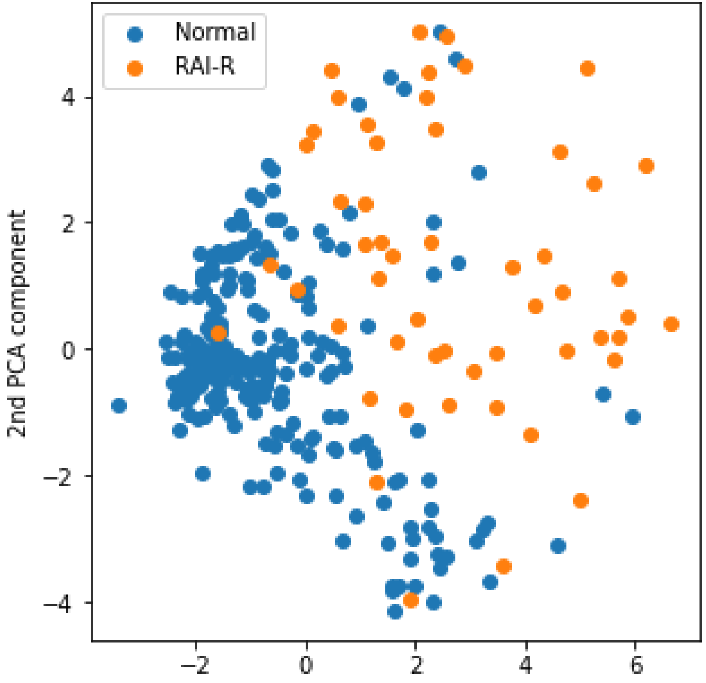


Qian, B. et al. “Orchestrating the Development Lifecycle of Machine Learning-Based IoT Applications: A Taxonomy and Survey”

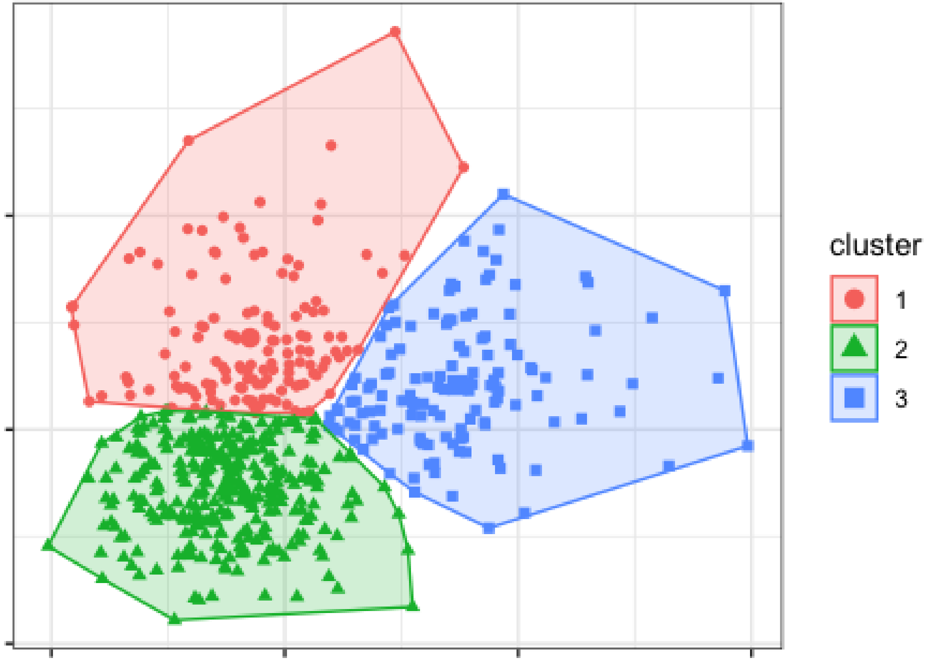
- Identify robust patterns that can be generalized to new data

Two primary branches of unsupervised learning

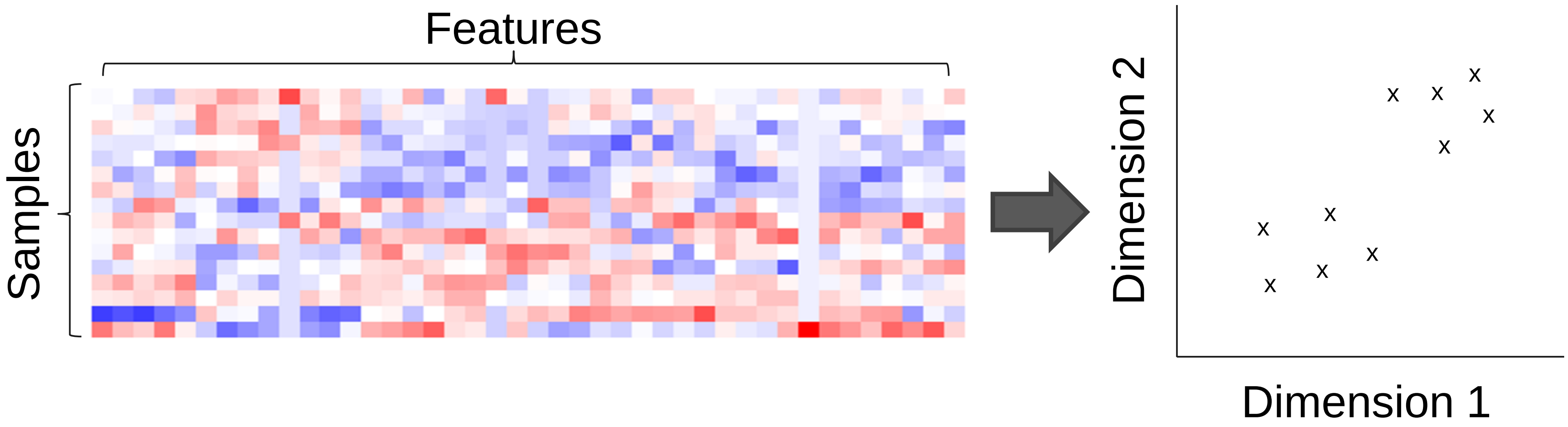
Dimensionality Reduction



Clustering

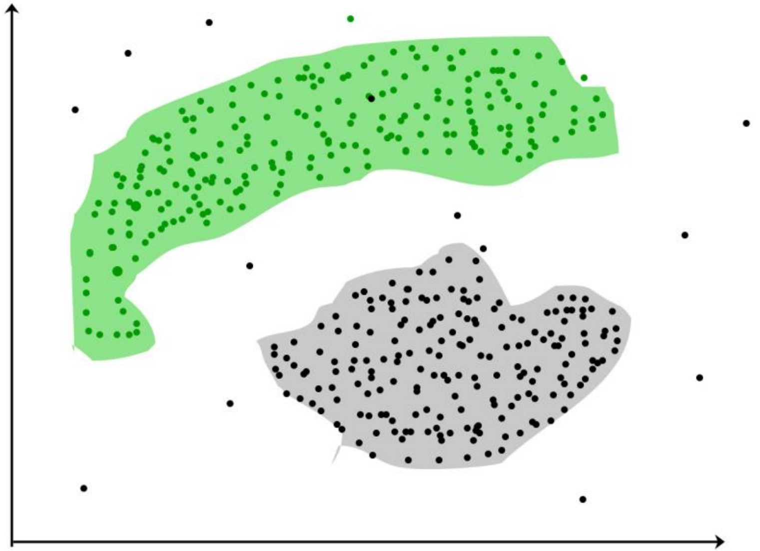


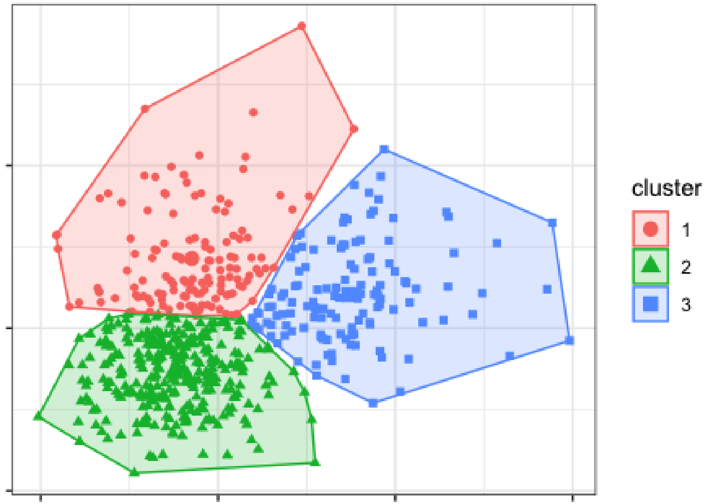
Dimensionality reduction



- Understand data distribution & gauge the difficulty of supervised learning
- Visualize on high-dimensional data on 2D-3D

Clustering

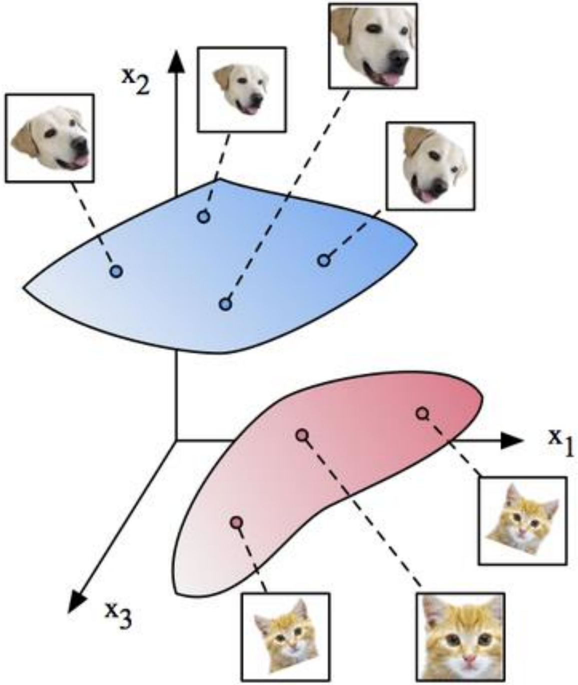




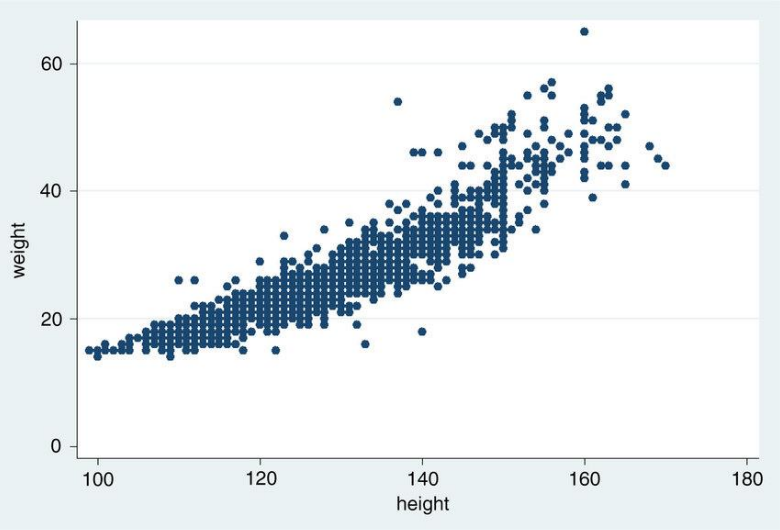
- Identify data subgroups
  - Predict shared characteristics
  - Generate hypothesis

Dimensionality reduction

Manifold hypothesis



“Real-world, high-dimensional data lie on some low-dimensional manifolds”

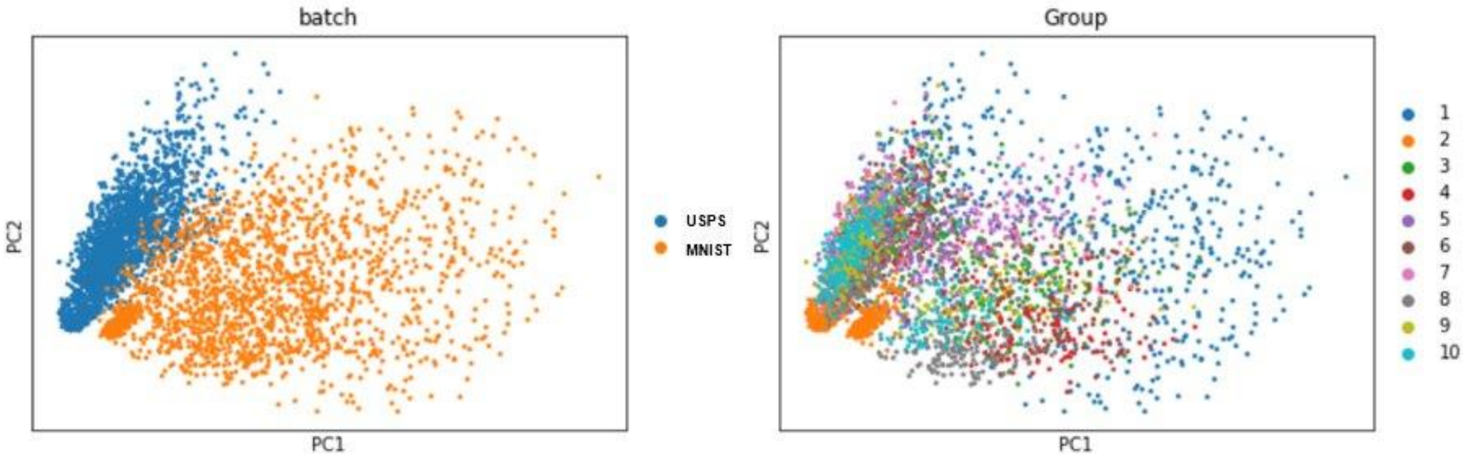


Chung, S. et al. “Classification and Geometry of General Perceptual Manifolds”

Nordin, P. et al. Global Health Action 7:25351 (2014)

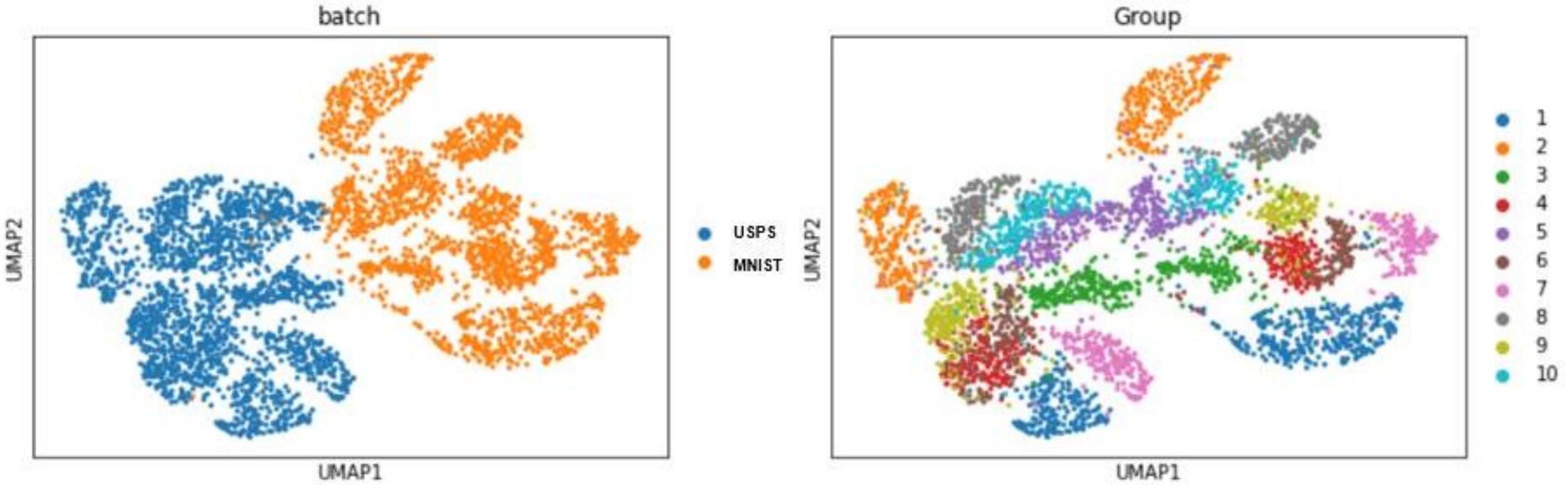
An example: Digit datasets

*adapted from doi:10.1109/TKDE.2017.2669193



https://twitter.com/lkmklsmn/status/1436357177887895555

A powerful 2D visualization

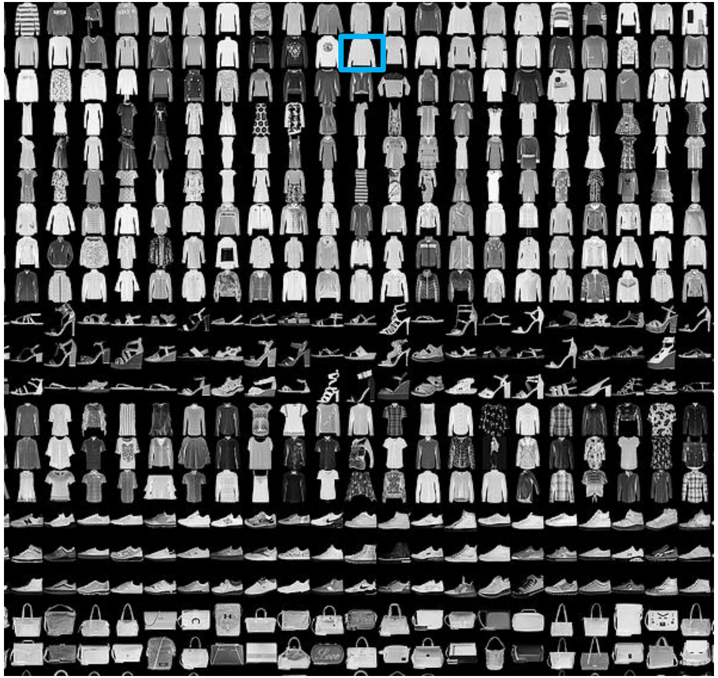


https://twitter.com/lkmklsmn/status/1436357177887895555

- Both data source and digit identity can be distinguished

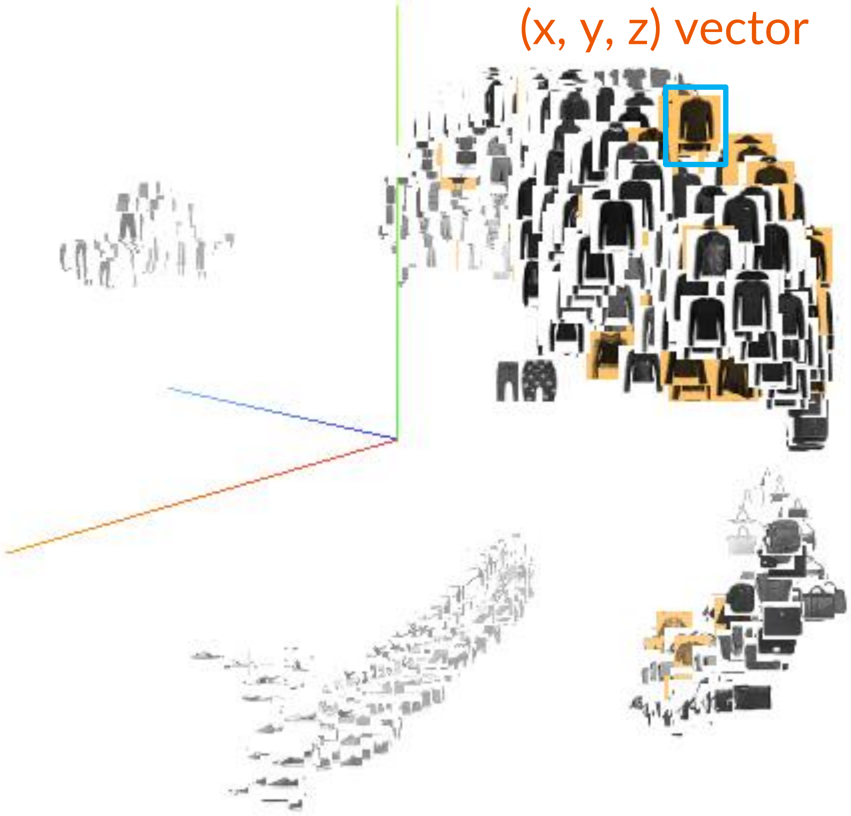
Another example: Fashion MNIST

28x28 pixel image



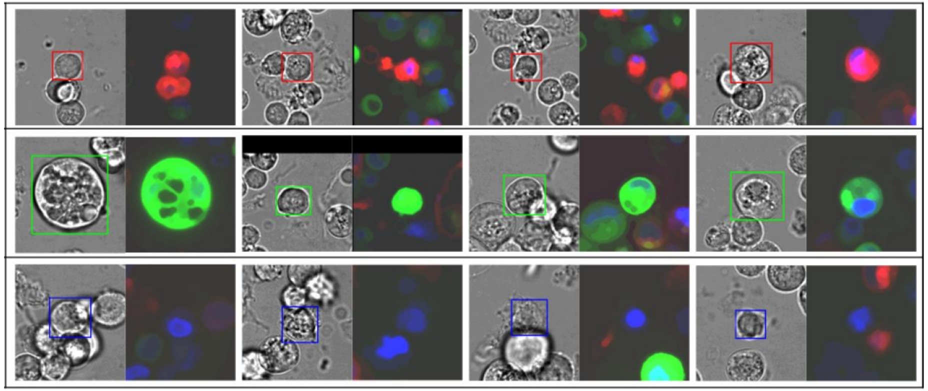
https://github.com/zalandoresearch/fashion-mnist

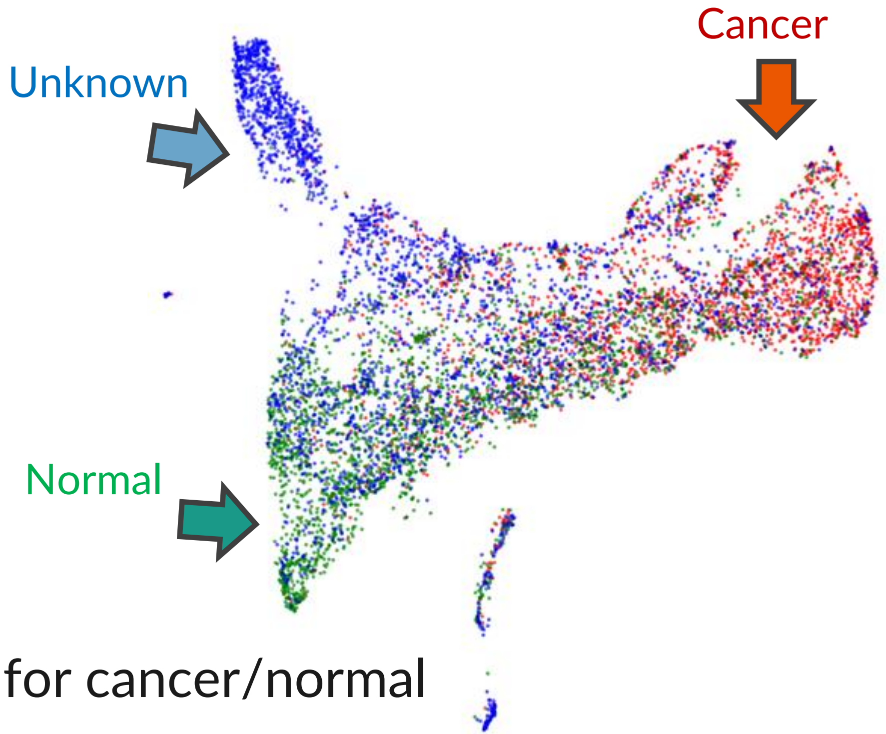
(x, y, z) vector



https://pair-code.github.io/understanding-umap/

2D visualization for cell images

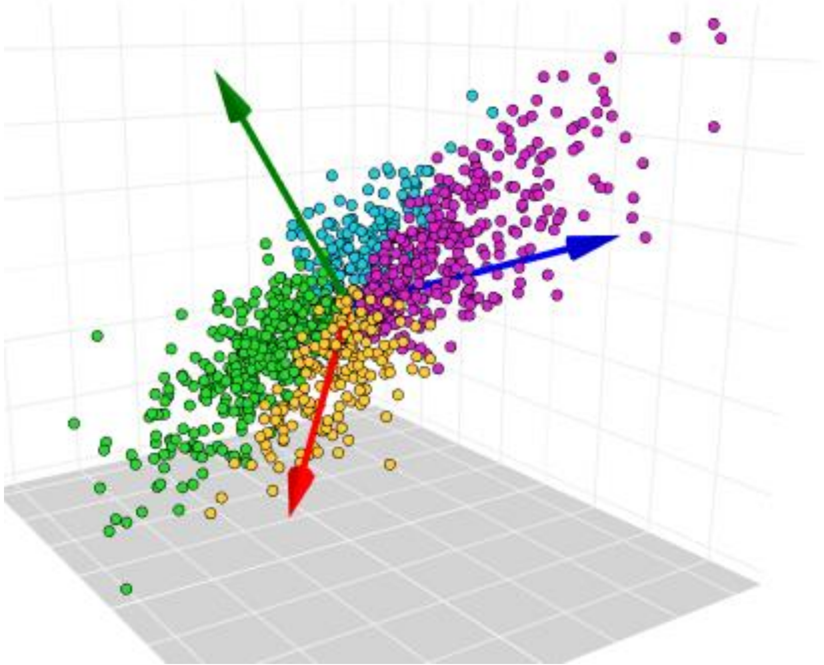




- Extracted from deep learning model for cancer/normal cell type classification

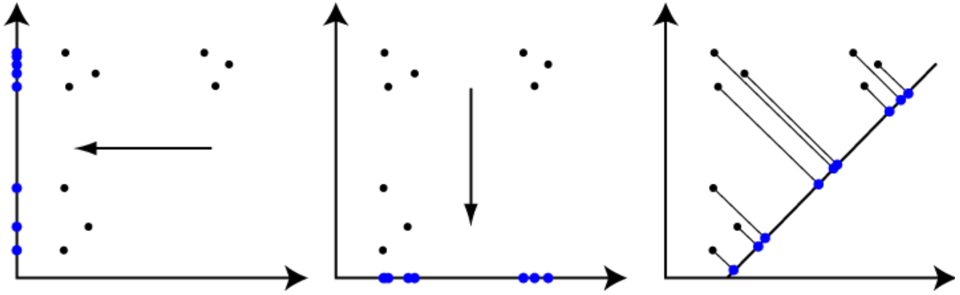
Principal component analysis (PCA)

Variance is information



https://towardsdatascience.com/principal-component-analysis-pca-explained-visually-with-zero-math-1cbf392b9e7d

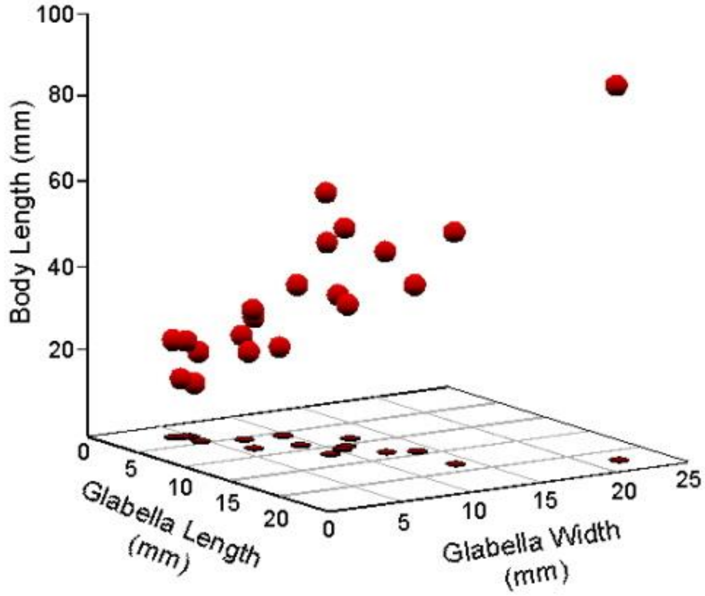
Projection

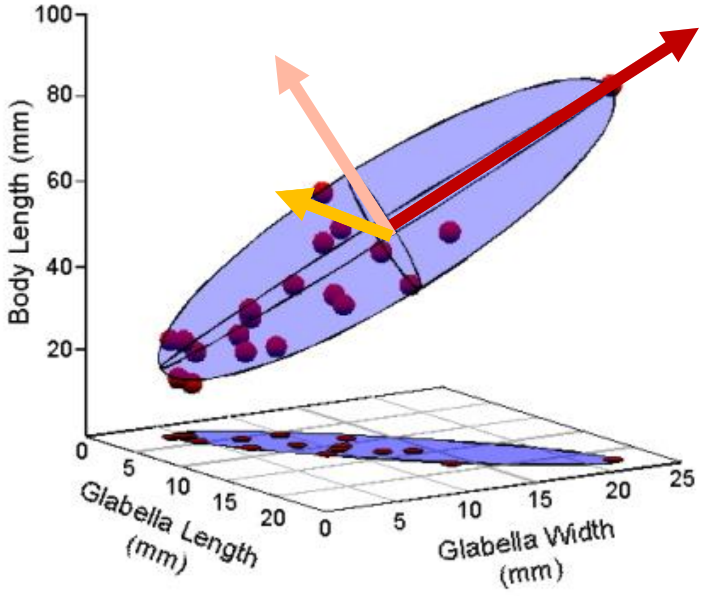


https://shapeofdata.wordpress.com/2013/04/16/visualization-and-projection/

- High variances = more power to distinguish groups of data points

PCA identifies directions with high variances

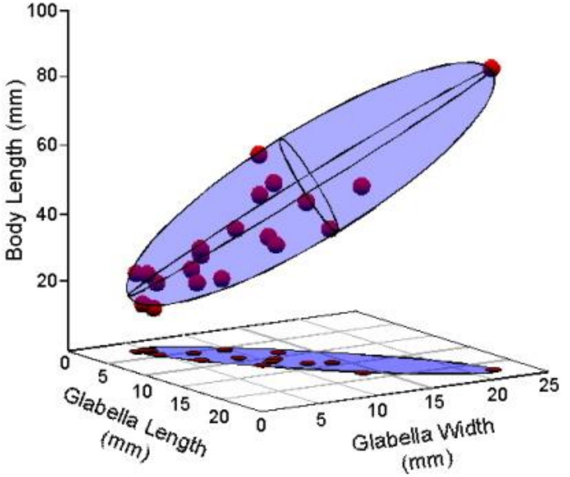


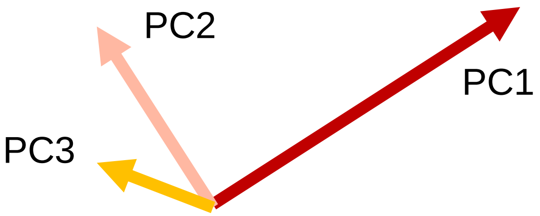


Source: the paleontological association

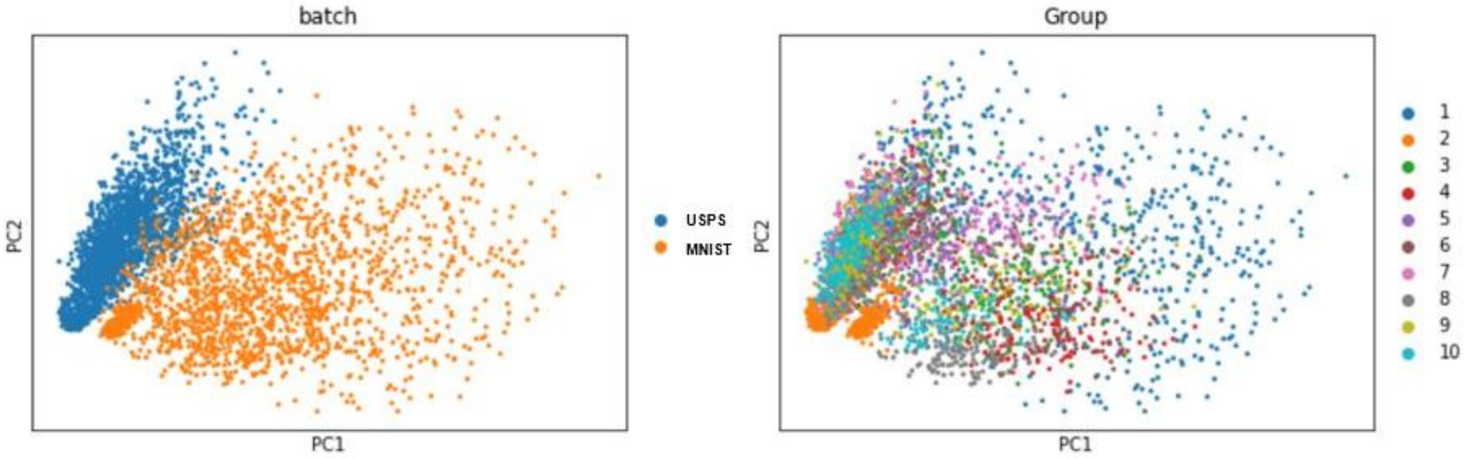
Interpretations of PCA algorithm

- Fitting the data with an *n*-dimensional ellipsoid
  - Axes = principal component (PC) direction
  - Axis lengths = magnitude of variances captured
- Orthogonal linear transformation
  - New axes are rotations of the original axes
  - PC1 $= w_1x_1 + w_2x_2 + \cdots + w_nx_n$
  - PC2 $= v_1x_1 + v_2x_2 + \cdots + v_nx_n$
  - w_i's and v_i's are called loadings





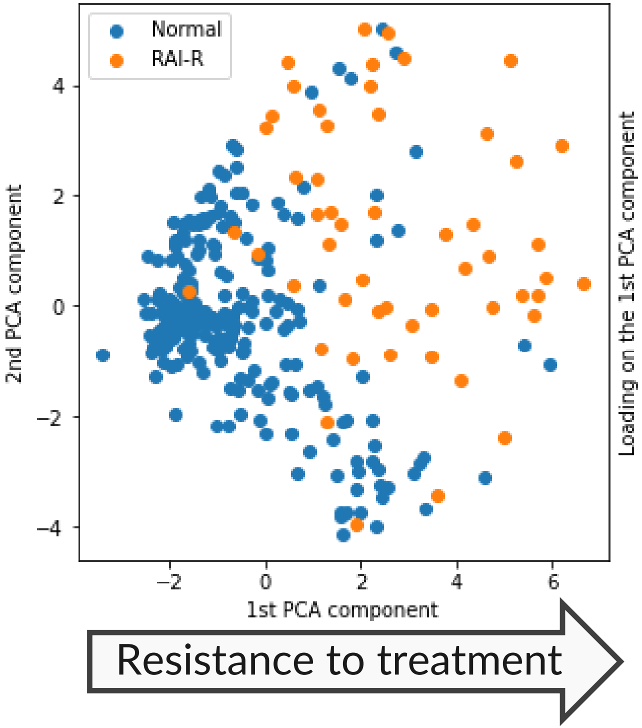
Interpretation of PCA result

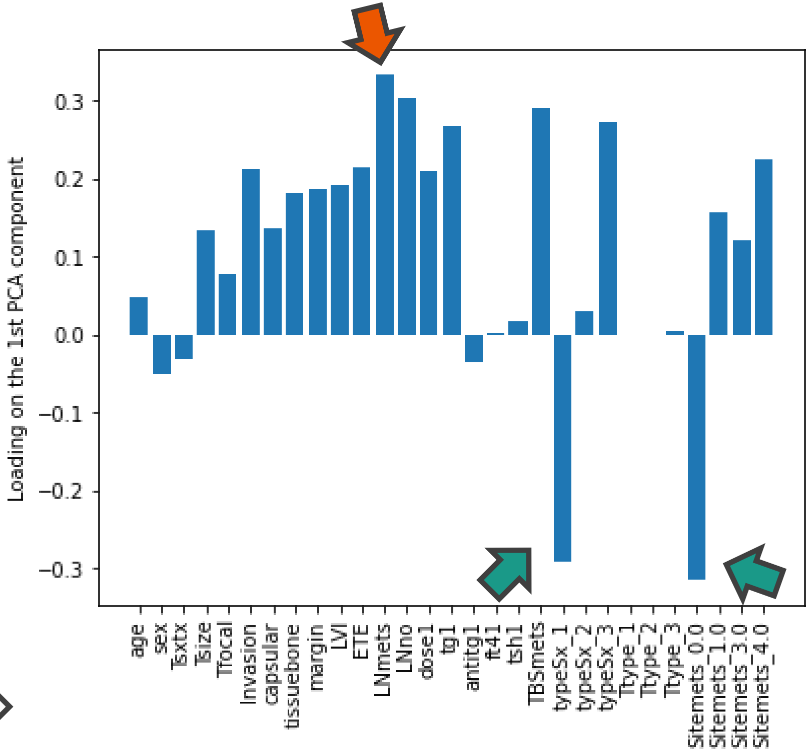


https://twitter.com/lkmklsmn/status/1436357177887895555

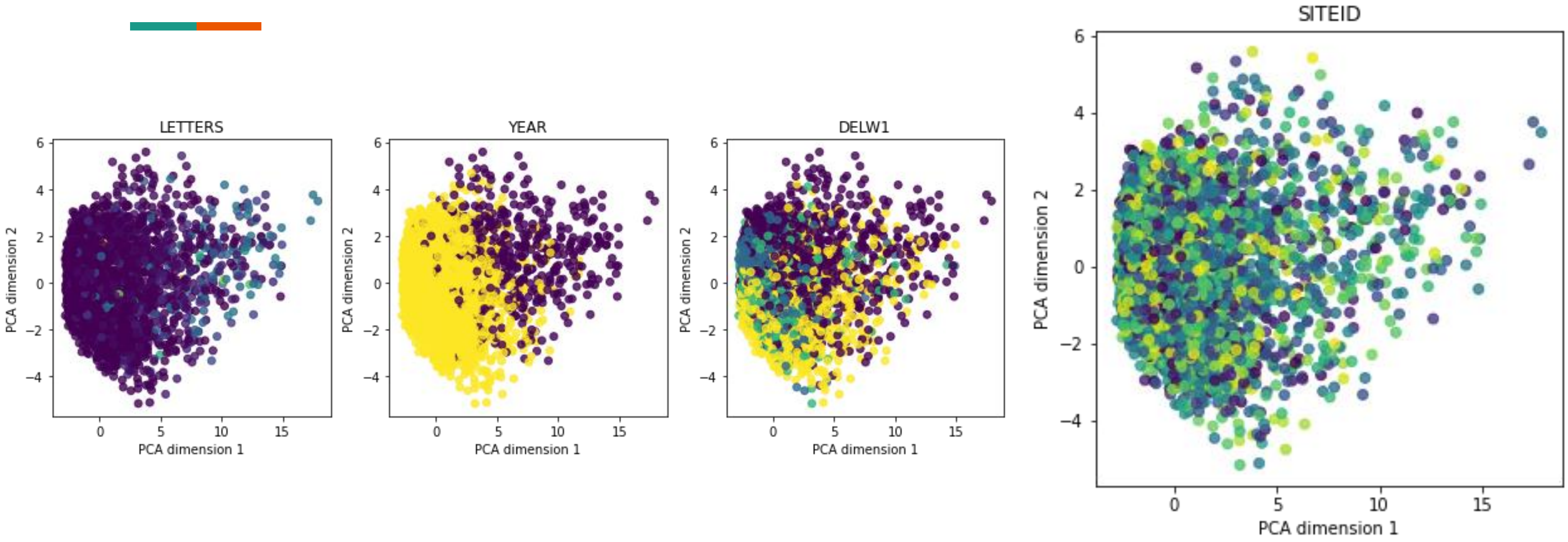
- PC1 captures the variance between data sources
- PC2 somewhat captures the variance between digit identity

Interpreting loadings on individual PC



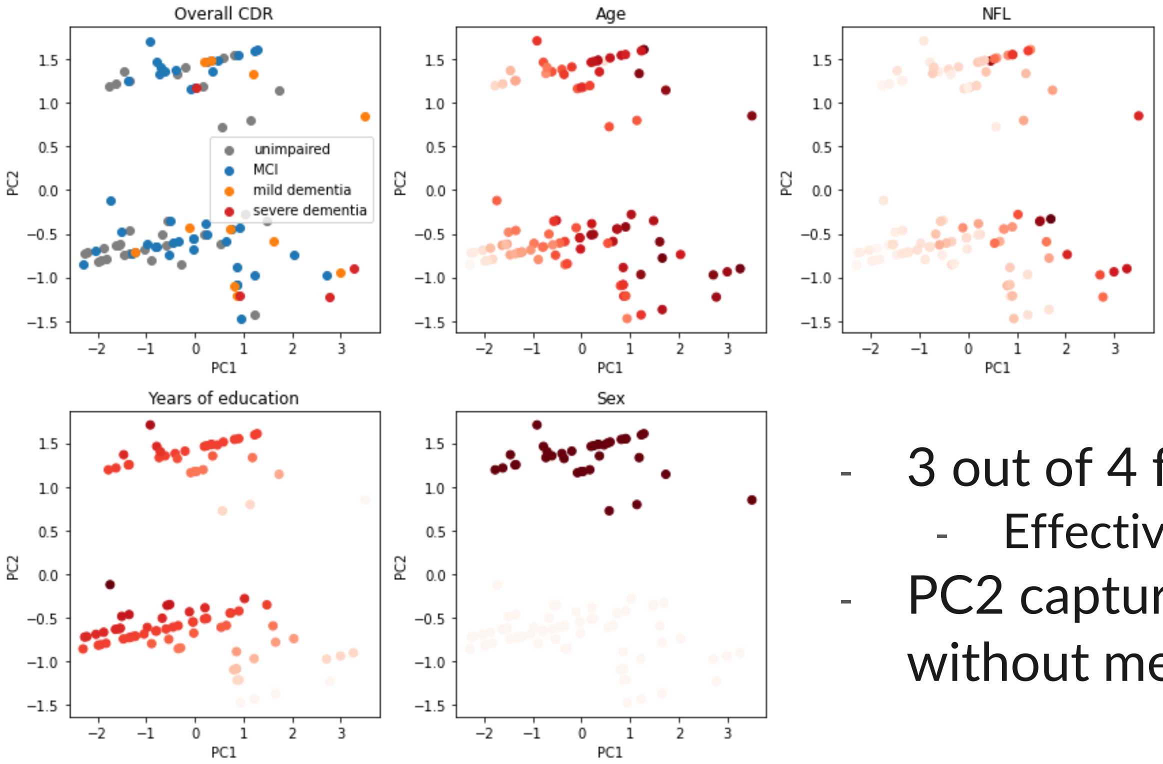


Exploring PCA results



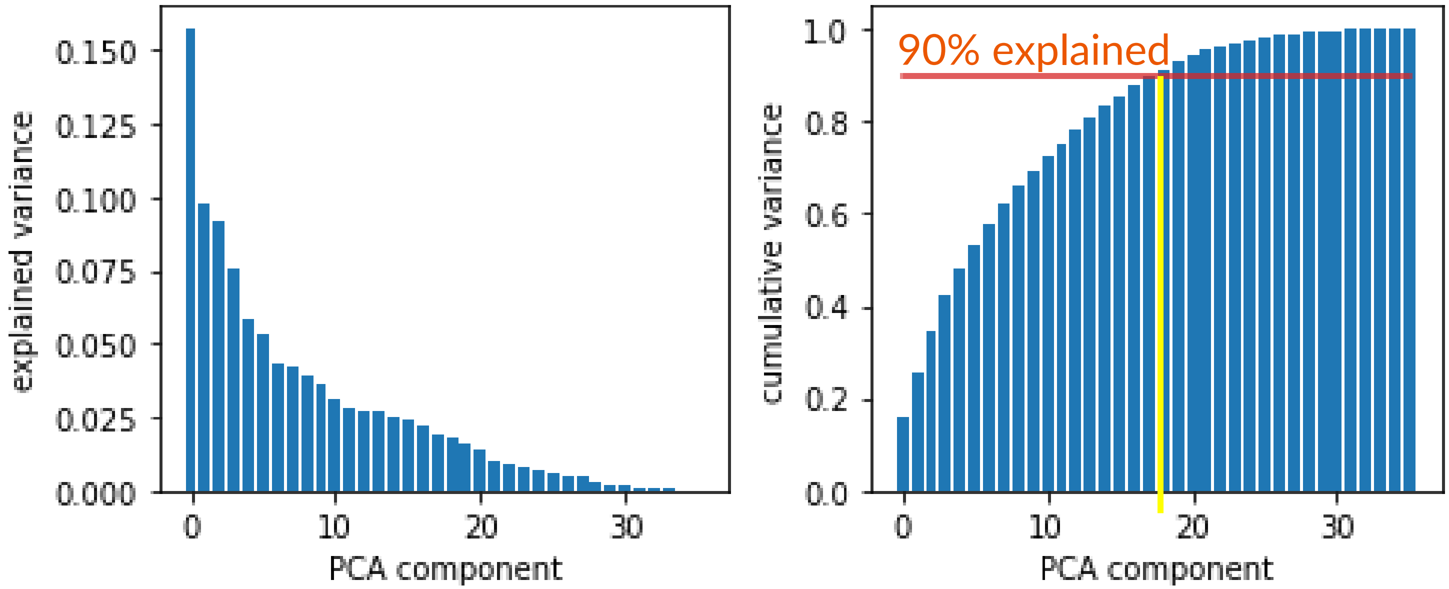
- Color by feature values to understand how PCA group data points
- Color by potential confounding factors

Be careful of correlated features



- 3 out of 4 features are correlated
  - Effective dimension is 2
- PC2 captures the gender difference without meaningful interpretation

PCA for dimensionality reduction

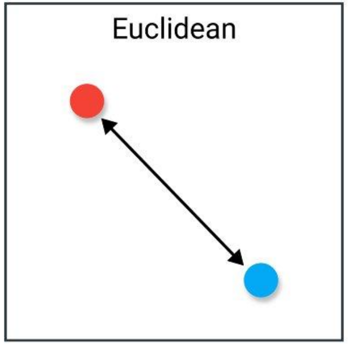


- By default, PCA retains the number of dimensions
- We can select only the first *k* PC for downstream analyses

Pros and cons of PCA

- Each PC can be interpreted from the loadings
- Highly correlated features tend to be grouped into the same PC
- PCA is a good initial dimensionality reduction step

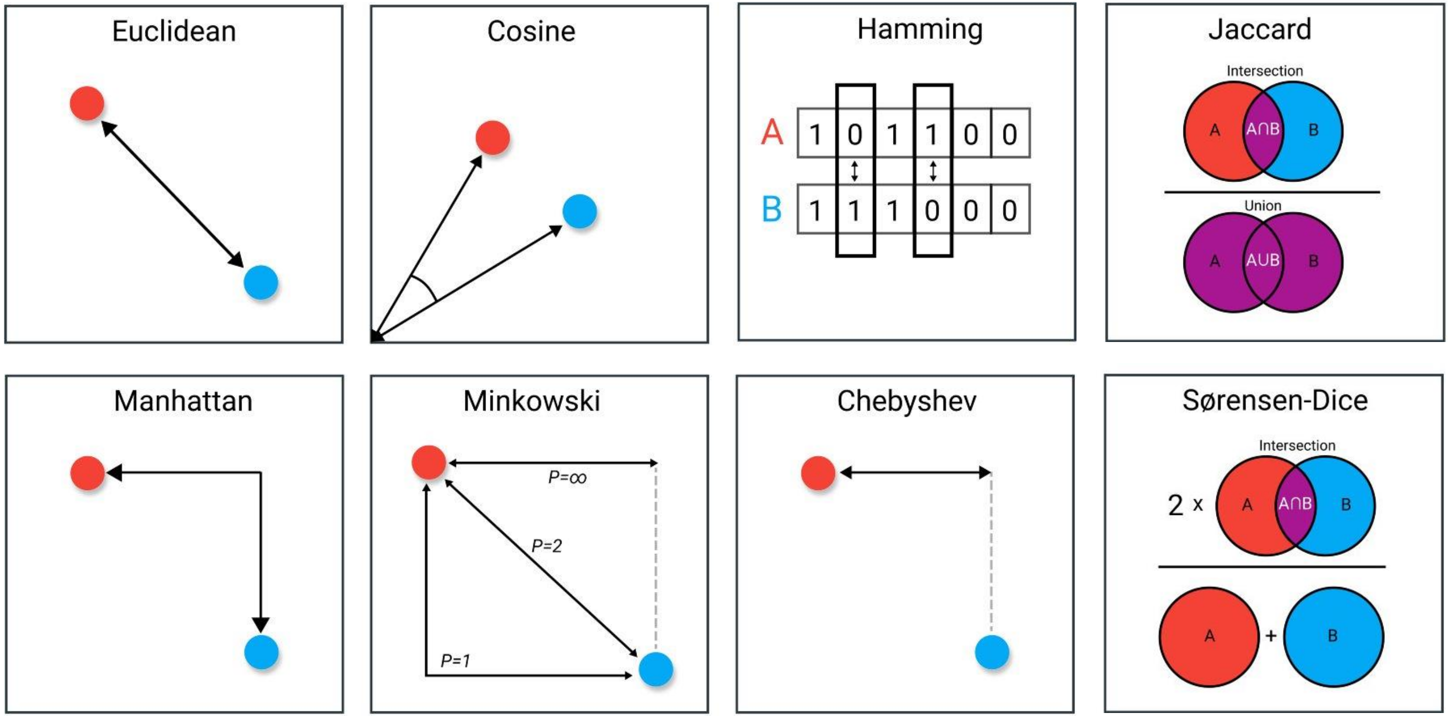
- PCA strictly preserves Euclidean distance
    - But some datasets require different distance metric!



https://towardsdatascience.com/9-distance-measures-in-data-science-918109d069fa

Multidimensional Scaling (MDS)

Distances



https://towardsdatascience.com/9-distance-measures-in-data-science-918109d069fa

Pairwise distance matrix

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1		.62	.15	.15	.82	.25	.65	.33	.45	.61	.73	.84	.99	.41	.20	.24	.26	.34	.67	.15	.01	.35	.73	.51
2	.77		.48	.50	.41	.52	.39	.29	.20	.02	.12	.25	.50	.28	.43	.41	.43	.35	.25	.49	.62	.37	.25	.18
3	.36	.45		.02	.69	.16	.52	.20	.32	.48	.58	.71	.88	.29	.05	.09	.12	.19	.52	.03	.15	.23	.58	.36
4	.36	.48	.07		.70	.14	.52	.22	.33	.49	.60	.72	.90	.30	.07	.10	.13	.20	.53	.01	.15	.22	.58	.37
5	.83	.21	.53	.58		.65	.22	.56	.47	.43	.35	.34	.67	.44	.65	.64	.66	.57	.21	.69	.83	.48	.24	.41
6	.43	.41	.12	.15	.47		.45	.28	.35	.52	.61	.73	.96	.31	.17	.17	.19	.23	.49	.13	.25	.19	.55	.37
7	.68	.20	.34	.35	.34	.30		.42	.35	.40	.38	.44	.78	.31	.48	.47	.48	.40	.18	.51	.65	.30	.25	.30
8	.47	.34	.21	.26	.39	.16	.31		.13	.29	.40	.52	.72	.15	.15	.16	.20	.16	.38	.21	.33	.17	.43	.22
9	.54	.26	.24	.28	.33	.20	.23	.10		.19	.30	.40	.66	.14	.27	.25	.28	.20	.29	.33	.45	.21	.34	.16
10	.74	.05	.42	.45	.23	.38	.18	.31	.22		.13	.26	.50	.28	.43	.41	.43	.35	.26	.49	.62	.37	.27	.19
11	.90	.15	.58	.61	.28	.55	.28	.49	.40	.18		.17	.50	.37	.53	.51	.53	.45	.22	.59	.73	.45	.19	.24
12	.20	.59	.25	.27	.65	.31	.54	.30	.36	.56	.73		.48	.46	.66	.65	.67	.58	.30	.72	.85	.55	.28	.39
13	.32	.45	.16	.21	.52	.21	.40	.17	.22	.42	.60	.14		.69	.84	.83	.84	.77	.65	.90	1.0	.81	.61	.66
14	.38	.41	.15	.19	.47	.15	.34	.11	.16	.37	.54	.21	.09		.25	.25	.26	.19	.30	.29	.42	.15	.37	.21
15	.41	.39	.11	.18	.45	.10	.32	.11	.16	.36	.54	.26	.14	.11		.07	.11	.16	.48	.07	.20	.19	.53	.31
16	.48	.35	.13	.14	.47	.15	.22	.22	.19	.32	.47	.35	.23	.19	.16		.06	.11	.46	.11	.24	.20	.51	.29
17	.40	.40	.06	.10	.49	.08	.30	.16	.19	.37	.54	.27	.16	.12	.08	.10		.10	.48	.14	.25	.23	.52	.30
18	.46	.33	.12	.16	.43	.13	.24	.14	.12	.30	.47	.30	.17	.12	.10	.08	.08		.39	.20	.34	.18	.43	.22
19	.93	.20	.59	.61	.30	.55	.27	.52	.43	.22	.10	.76	.63	.57	.55	.47	.55	.48		.52	.67	.33	.10	.21
20	.35	.48	.04	.04	.57	.13	.36	.24	.27	.45	.61	.25	.18	.17	.15	.14	.08	.15	.61		.16	.21	.58	.37
21	.06	.77	.35	.33	.84	.43	.67	.48	.54	.74	.90	.23	.34	.38	.42	.46	.39	.45	.92	.33		.36	.73	.51
22	.52	.26	.19	.23	.37	.18	.20	.14	.08	.23	.40	.35	.21	.16	.15	.12	.15	.08	.42	.23	.52		.40	.24
23	1.0	.25	.66	.68	.33	.62	.35	.58	.50	.28	.12	.83	.69	.64	.62	.55	.62	.55	.08	.69	.99	.49		.23
24	.87	.18	.53	.54	.30	.49	.20	.47	.38	.19	.13	.71	.58	.52	.50	.41	.49	.42	.07	.55	.86	.37	.15	

Metric

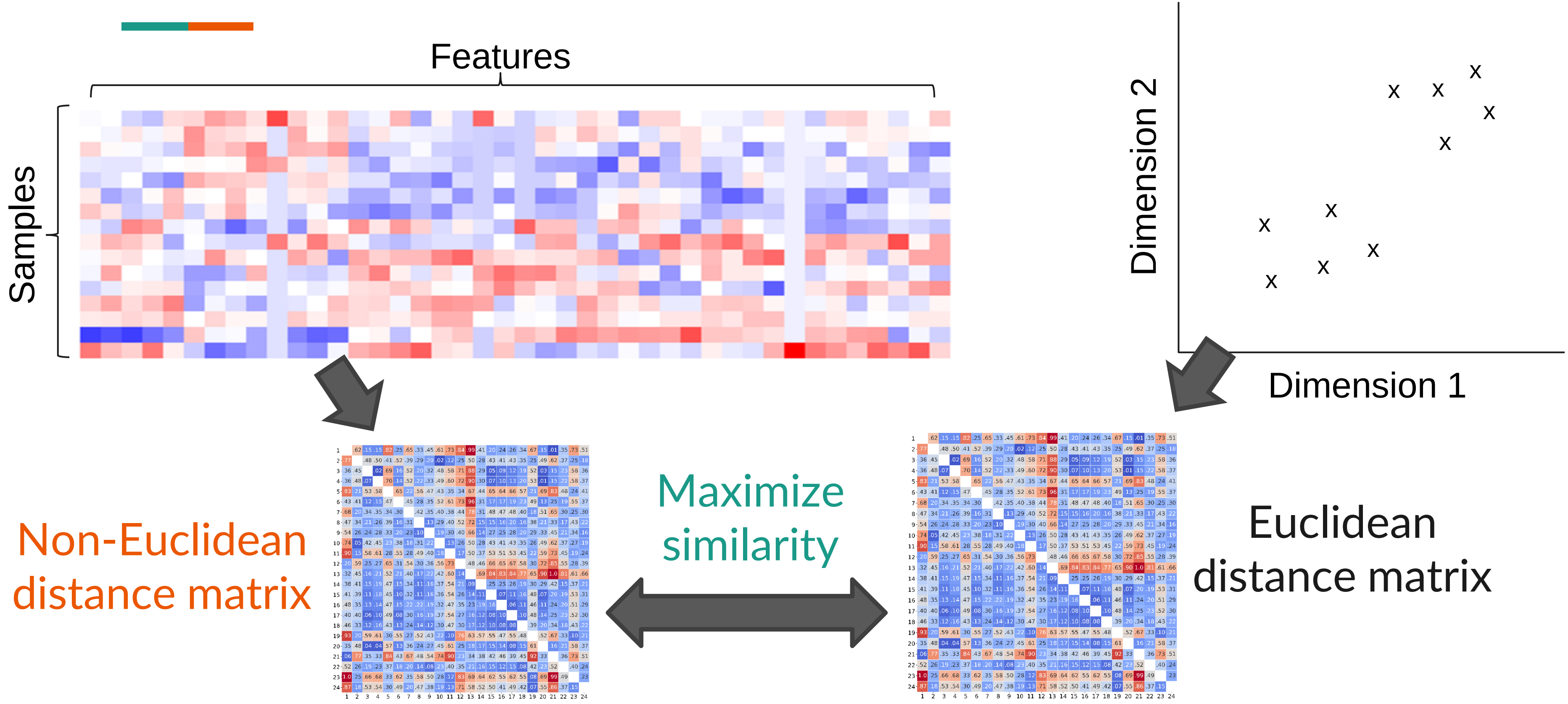
- $D(i, j)$ = distance between sample i and sample j
- $D(i, i) = 0$
- $D(i, j) = 0$ iff $i = j$
- $D(i, j) = D(j, i)$
- $D(i, j) + D(j, k) \geq D(i, k)$

Non-metric

- Any user-defined dissimilarity

de Nobel, J. et al. “Explorative Data Analysis of Time Series based Algorithm Features of CMA-ES Variants” April 2021

Principal Coordinate Analysis (PCoA)



PCoA algorithm sketch

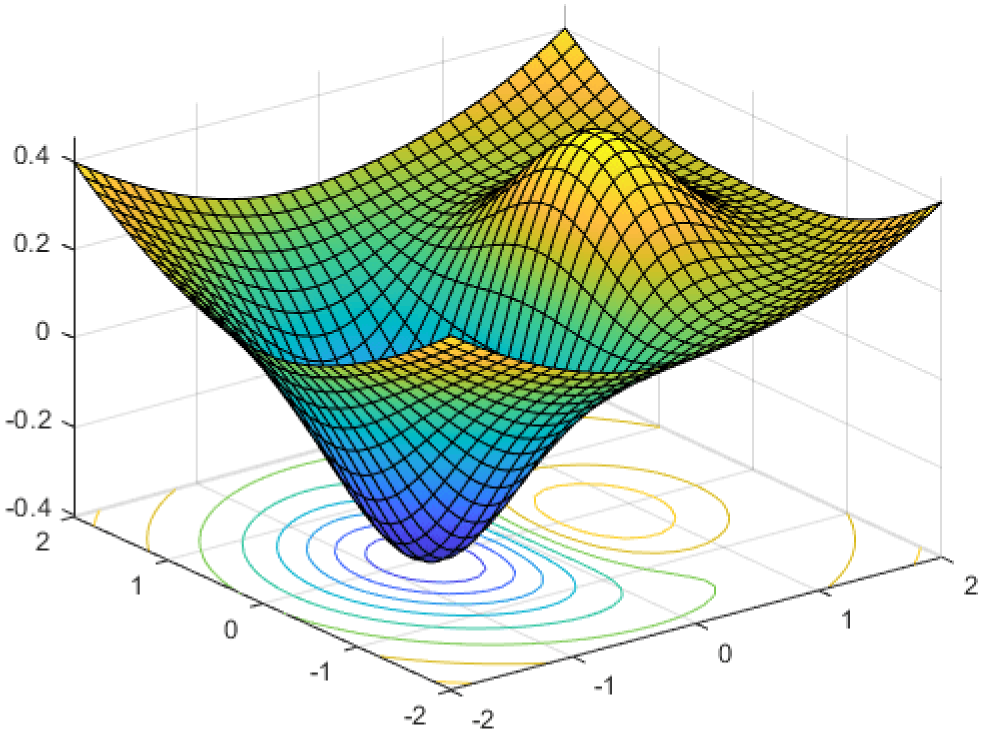
- Calculate pairwise distance matrix *D* for the original data

- For a 2D projection of the data: sample *i* projected onto (x_i, y_i)
    - Calculate pairwise distance matrix *D'* for the projection
    - Each element of *D'* is a function of (x_i, y_i)'s

- Calculate a similarity score between *D* and *D'*
    - Such as Person's correlation
    - This similarity score is a function of (x_i, y_i)'s

- Find (x_i, y_i)'s that maximize this score!

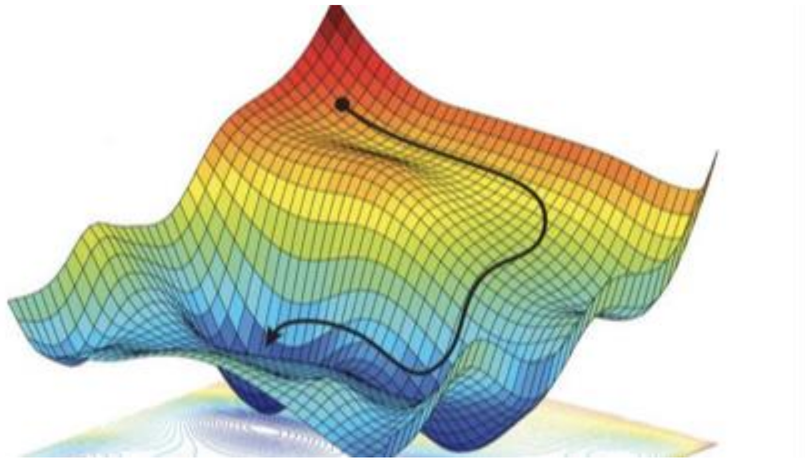
How to optimize a function?



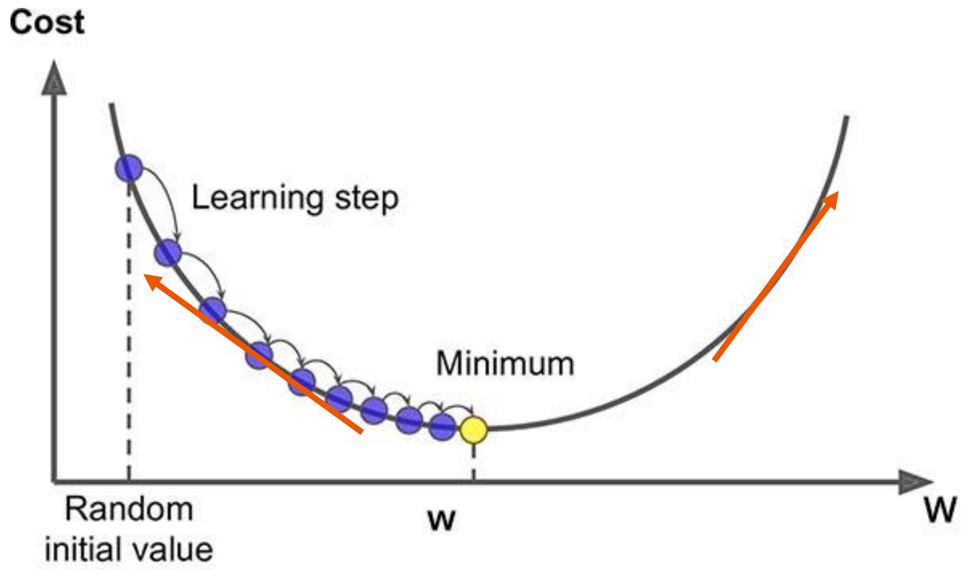
https://es.mathworks.com/help/optim/ug/optimization-toolbox-tutorial.html

- Find $(x_1, x_2, ..., x_n)$ that minimize $f(x_1, x_2, ..., x_n)$
- At minimum, the slope is zero in all directions
- Take derivative of each variable and set to zero
  - $\frac{\delta f}{\delta x_1} = 0$
  - $\frac{\delta f}{\delta x_2} = 0$
  - *n* equations with *n* variables

Gradient descent

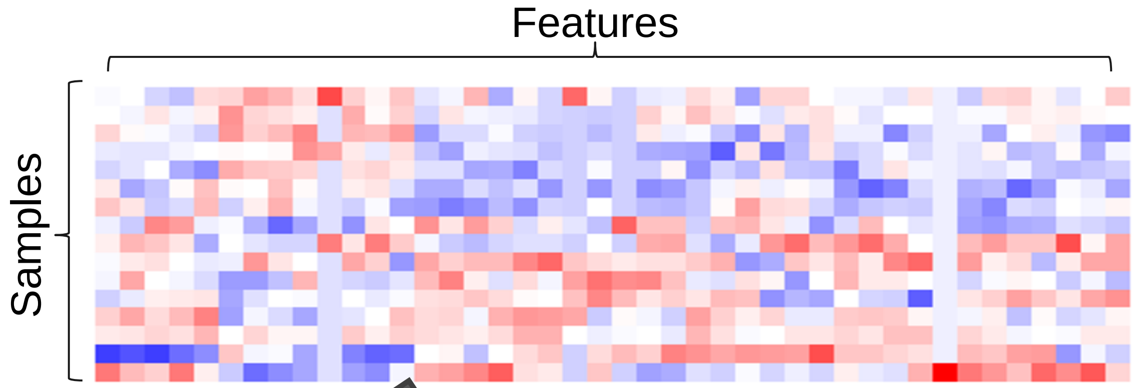


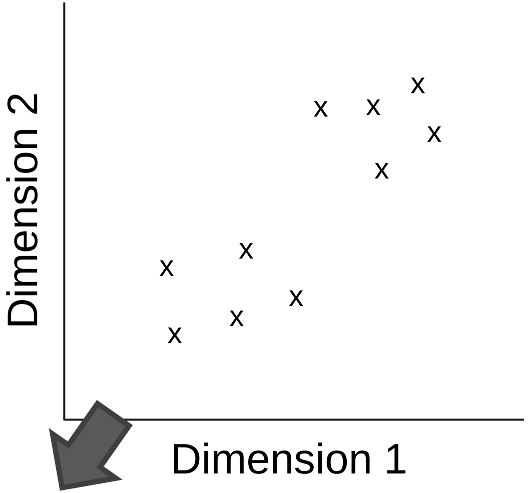
https://medium.com/analytics-vidhya/gradient-descent-b0dc1af33517



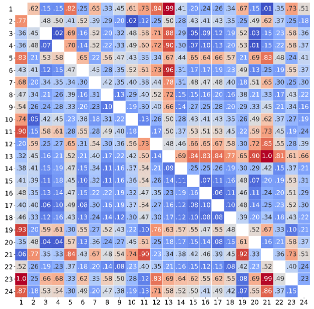
- Slope tells us if the function is increasing or decreasing if we increase x_i
    - So, we can update x_i accordingly

Generalized MDS

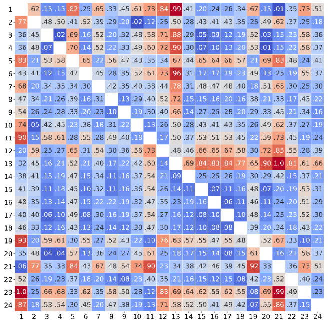




Any concept of distance



Maximize similarity



Any concept of distance

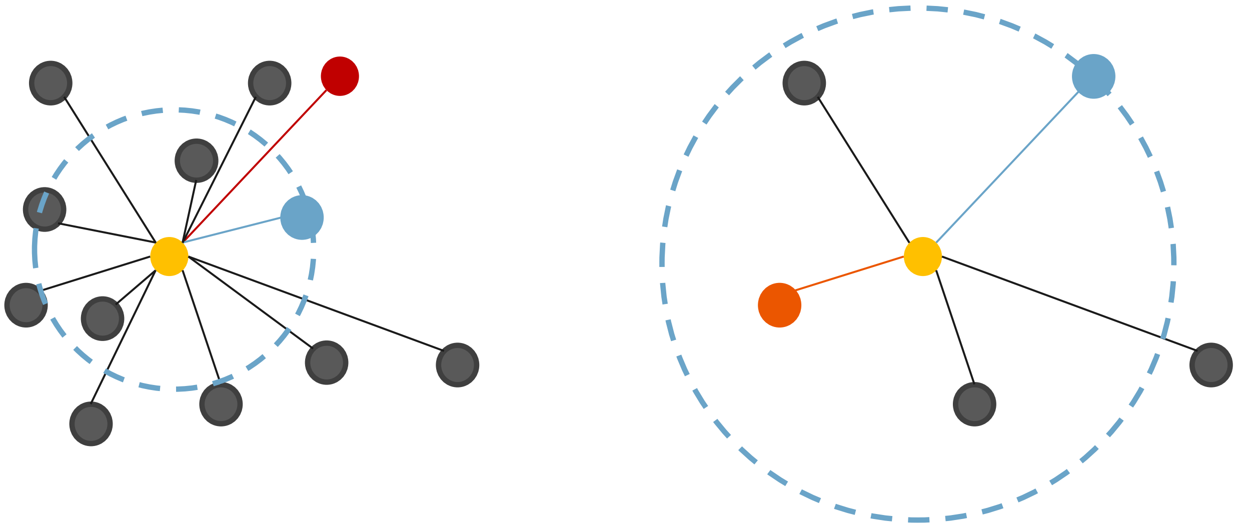
Limitation of PCA and MDS

- A single definition of distance metric is used throughout the data space

- What if some data groups are noisier than the others?
  - Difference in data density

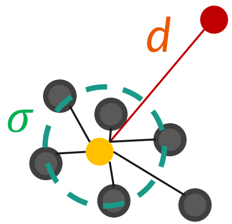
t-distributed stochastic neighbor embedding (*t*-SNE)

Measuring data density



- Distance to the *k*-th nearest neighbor reflects data density
  - Small distance in dense area
  - Large distance in sparse area

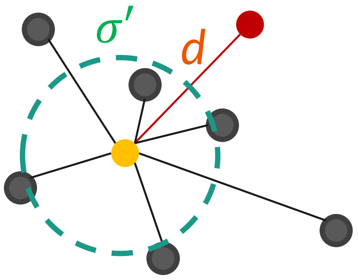
Probability of being a neighbor



score(o | o) = probability that o would pick o as neighbor
under a **normal distribution** center at o

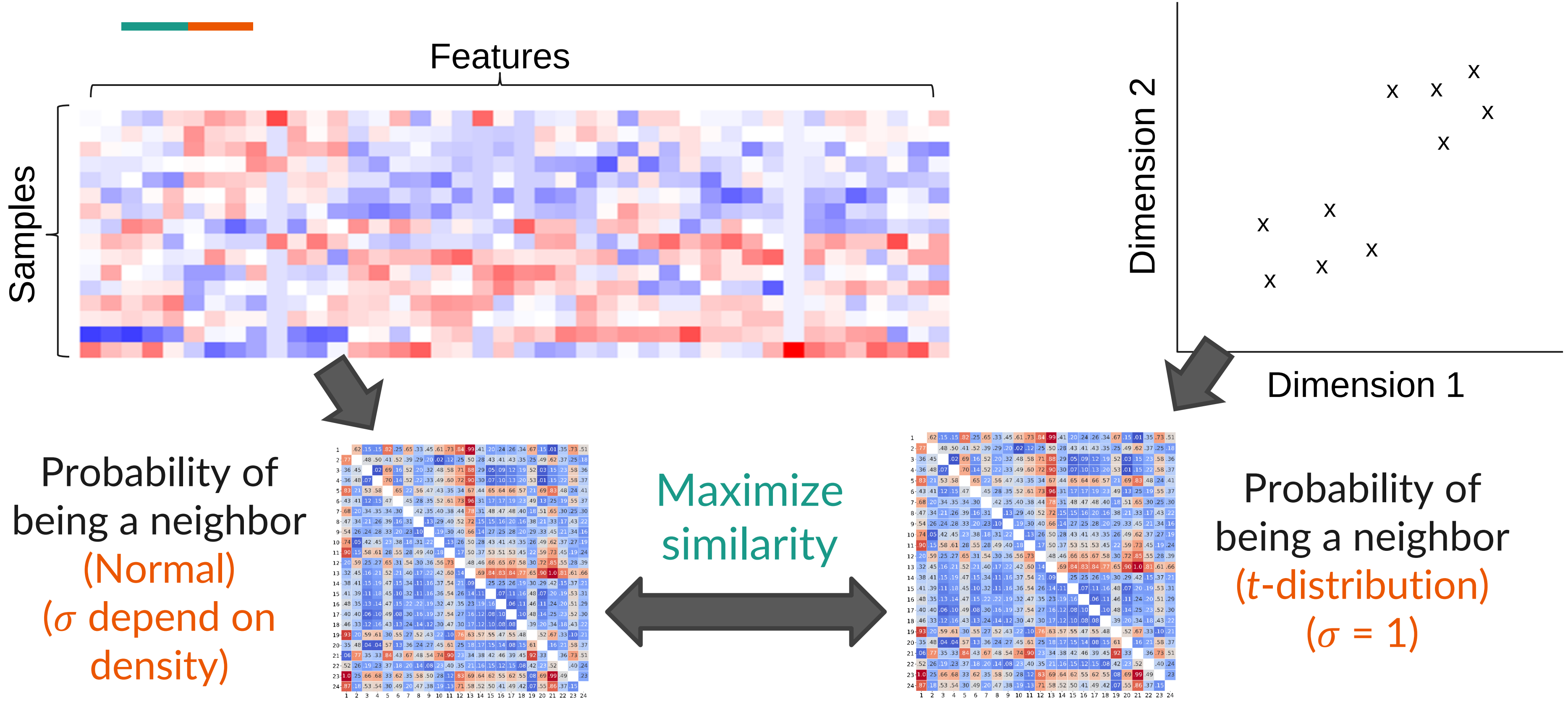
$$= \frac{e^{-\frac{d^2}{2\sigma^2}}/\sigma}{\sum e^{-\frac{(\text{dist}(o,o))^2}{2\sigma^2}}/\sigma}$$

o = other data points



- Same distance d normalized against density σ and distances to other nearby data points o

Finding the optimal projection for *t*-SNE



A similarity score for probability distribution

- Kullback-Leibler (KL) divergence
- Measure how distribution P differs from another distribution Q
  - $D_{KL}(P \,||\, Q) = -\sum_i \sum_j p_{j|i} \log_2 \frac{p_{j|i}}{q_{j|i}}$

- P = Probability of neighbor from the original data
- Q = Probability of neighbor from the projection

- Solve for the best $q_{j|i}$'s using gradient descent

Why *t*-distribution for the projection?

- *t*-distribution has fatter tails

- Allow data points to be projected far away from each other

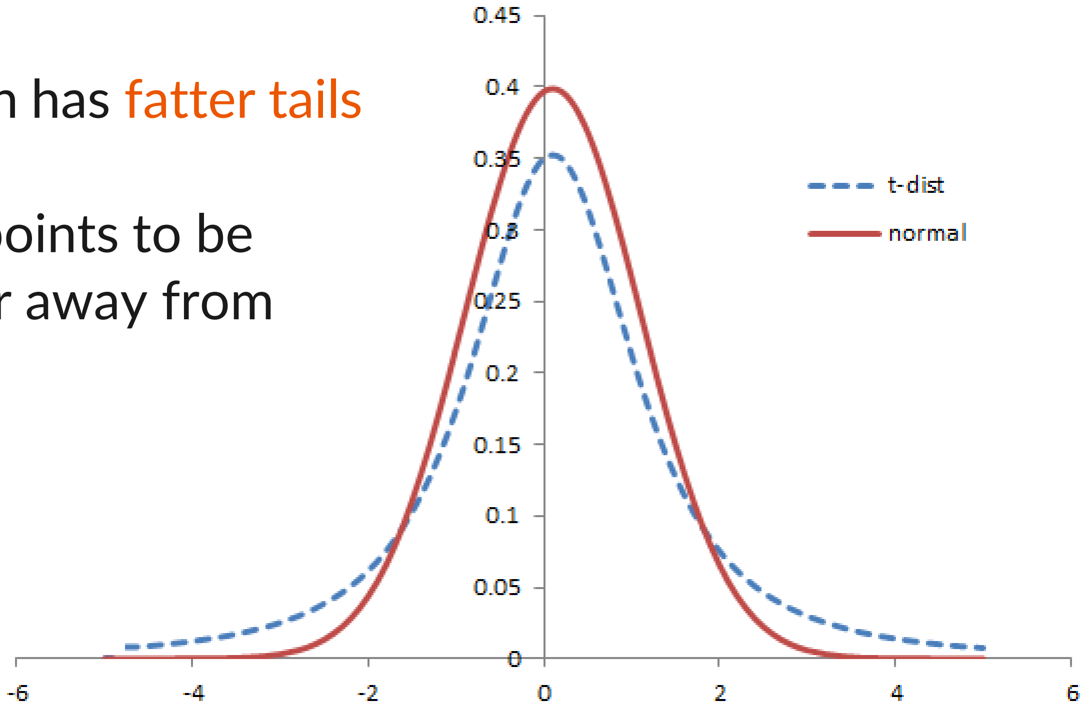
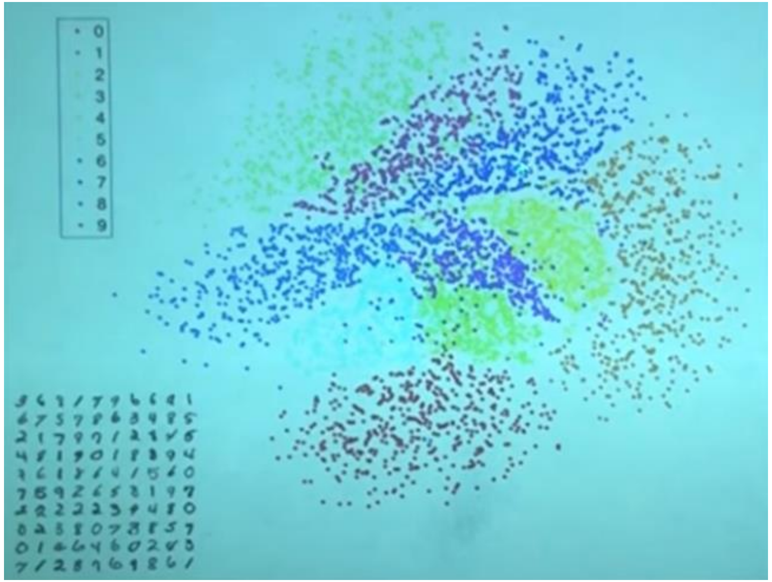
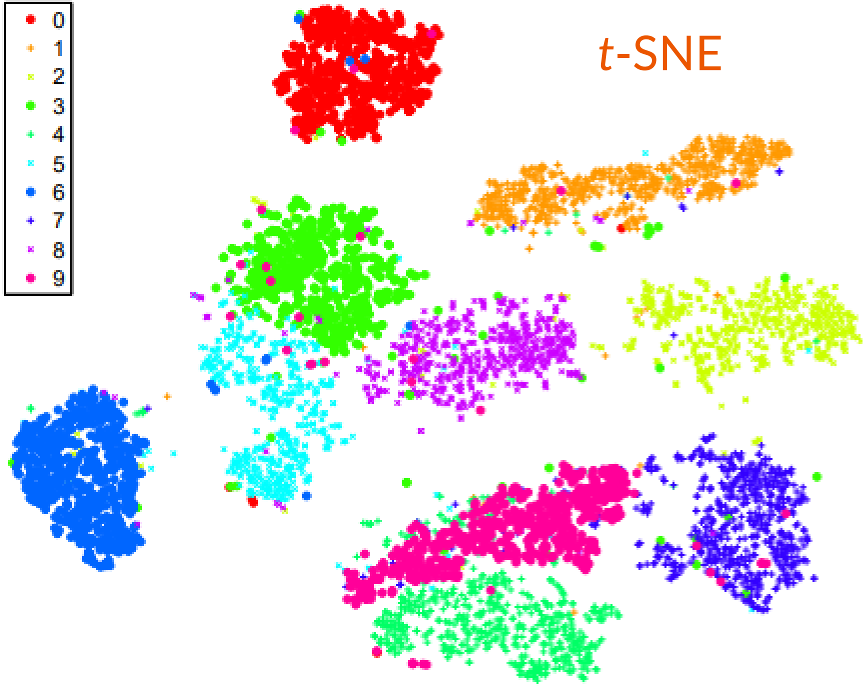


Image from riskprep.com

Impact of *t*-distribution

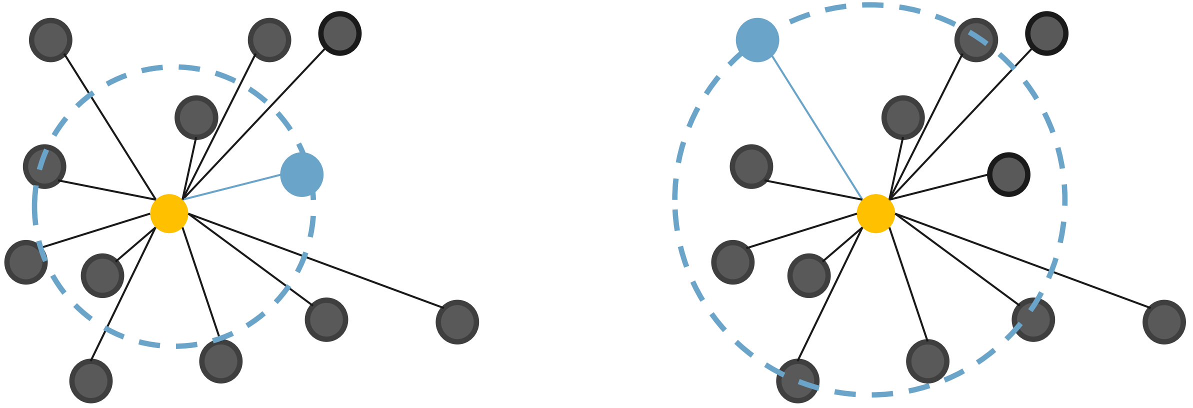


SNE (Normal → Normal)



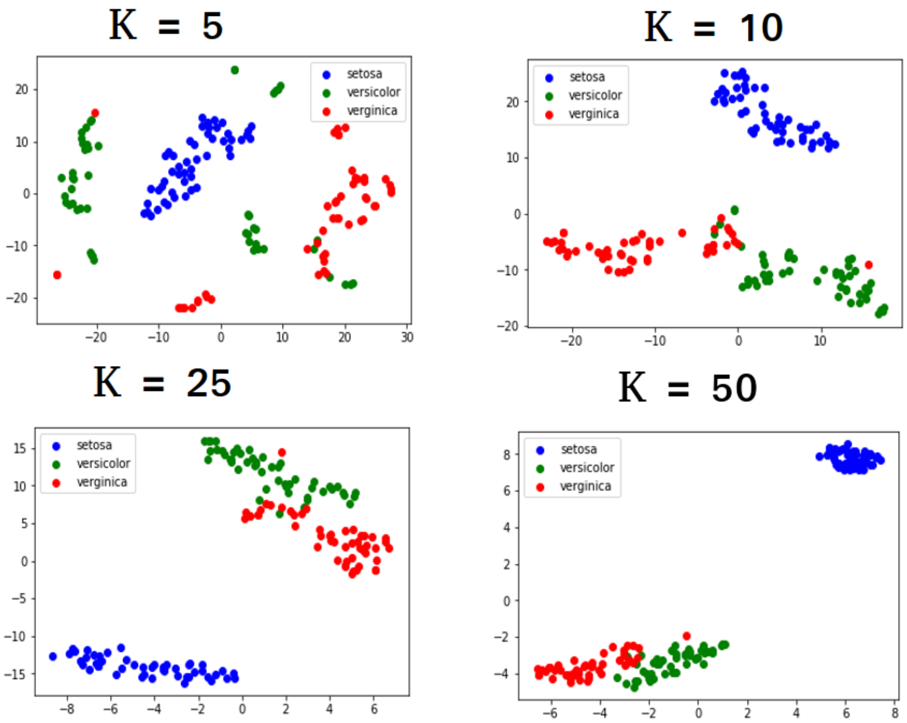
Maaten, L. and Hinton, G. J of Machine Learning Research 9:2579-2605 (2008)

Perplexity



- How many nearest neighbors to consider to normalize data density?
    - Perplexity parameters

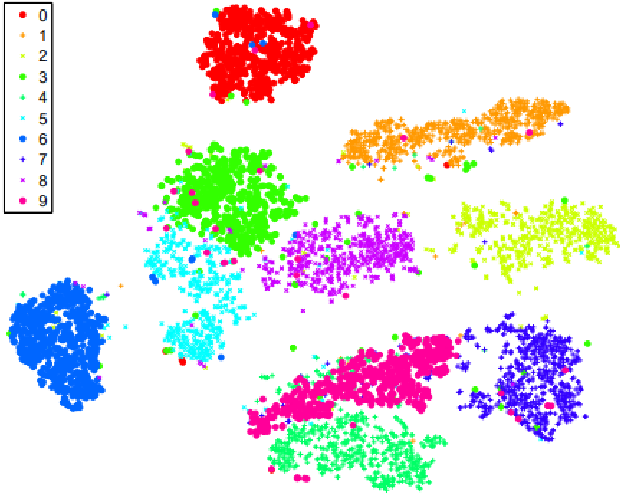
Impact of perplexity



- Too small perplexity = a lot of scatted data groups
- Try varying the perplexity and identify patterns that consistently appear

Source: blog.paperspace.com/dimension-reduction-with-t-sne/

Pros and cons of *t*-SNE



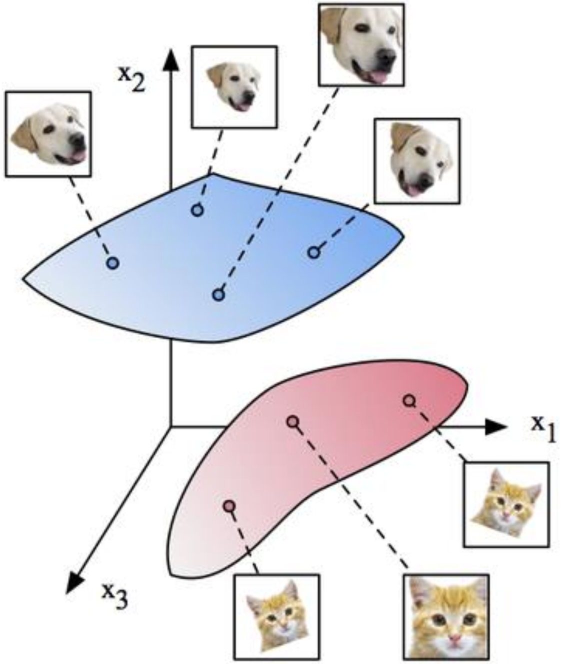
- Capture qualitative neighbor relationship
- Normalize data density

- Recompute every time new data is added
- Lose long-range relationship
- Axes of the resulting projection have no meaning
  - Don’t use *t*-SNE coordinates for clustering or interpretation

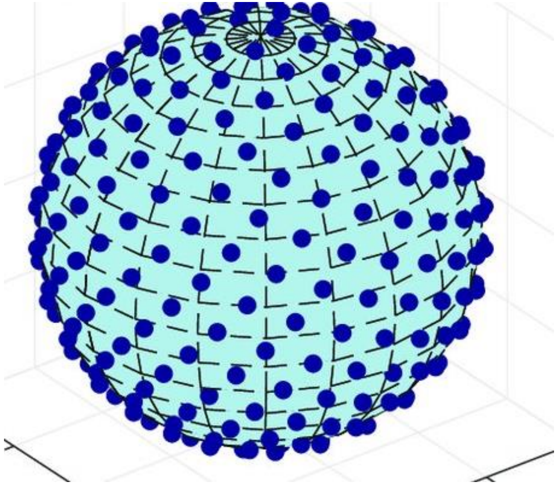
Maaten, L. and Hinton, G. J of Machine Learning Research 9:2579-2605 (2008)

Uniform manifold approximation and projection (UMAP)

Two key assumptions



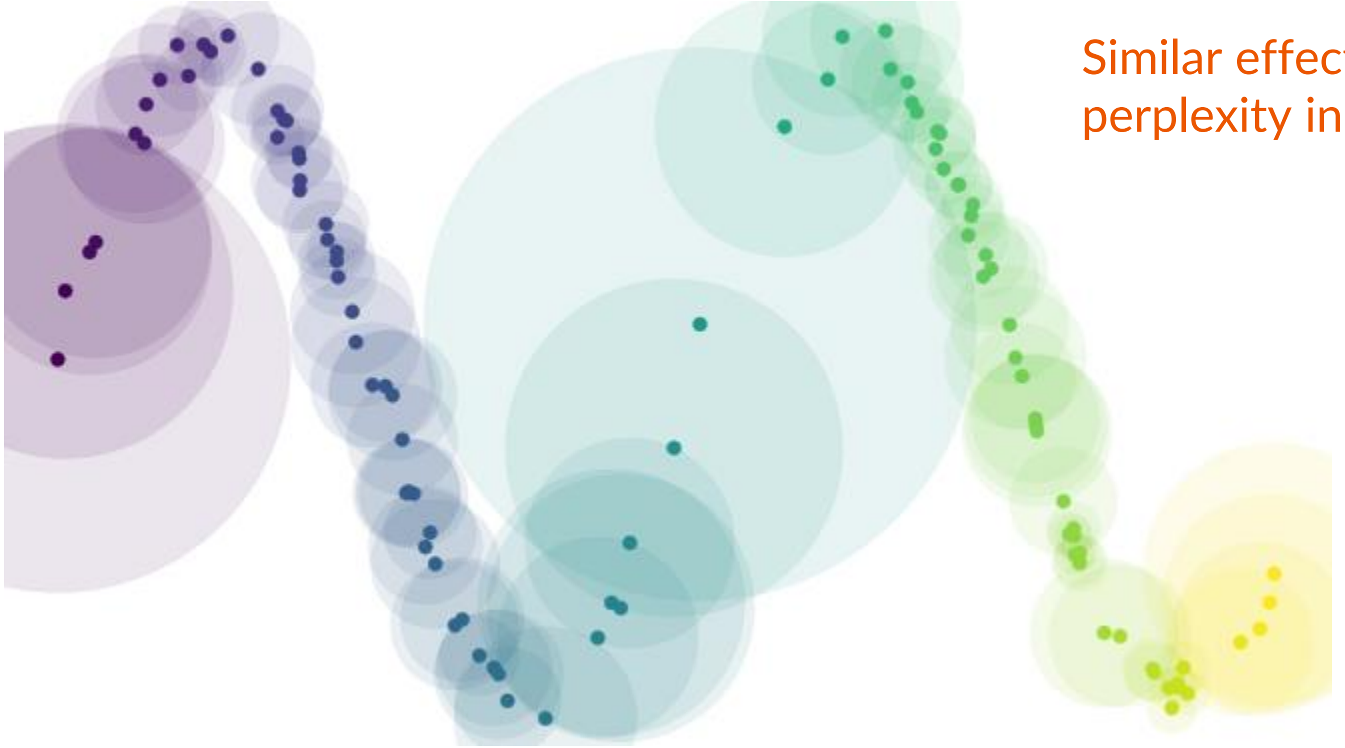
Chung, S. et al. “Classification and Geometry of General Perceptual Manifolds”



Ali, A. et al. IEEE Access PP(99):1 (2021)

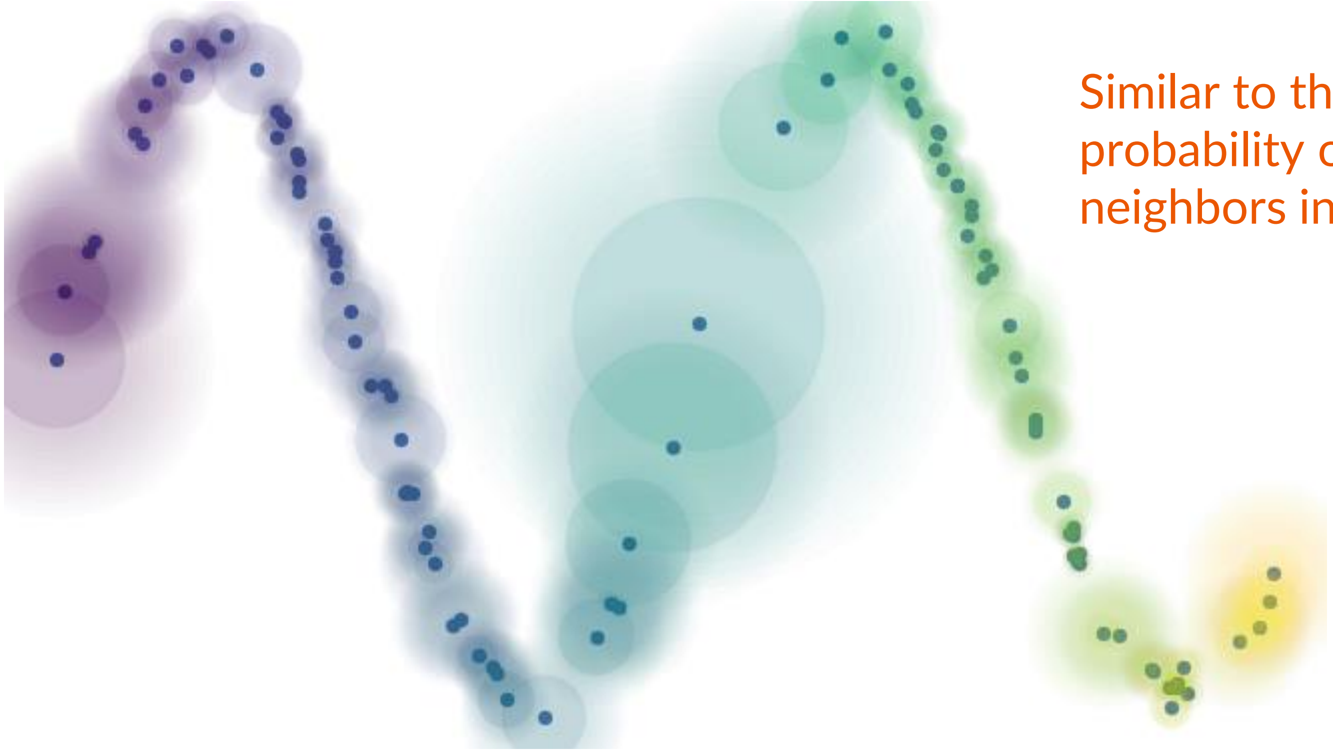
- Data came from multiple manifolds
- Data points were sampled uniformly

Uniform sampling = similar distance to *k*-th neighbor



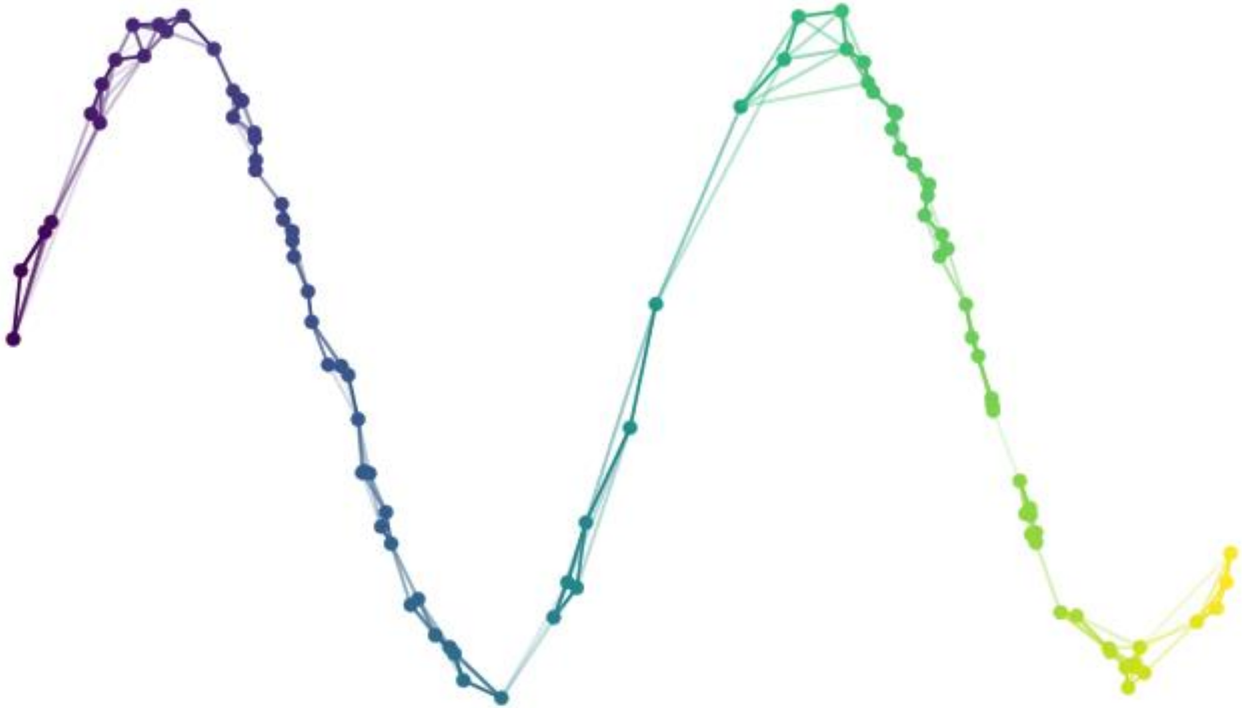
Similar effect as perplexity in t-SNE

Adding uncertainty between faraway data points

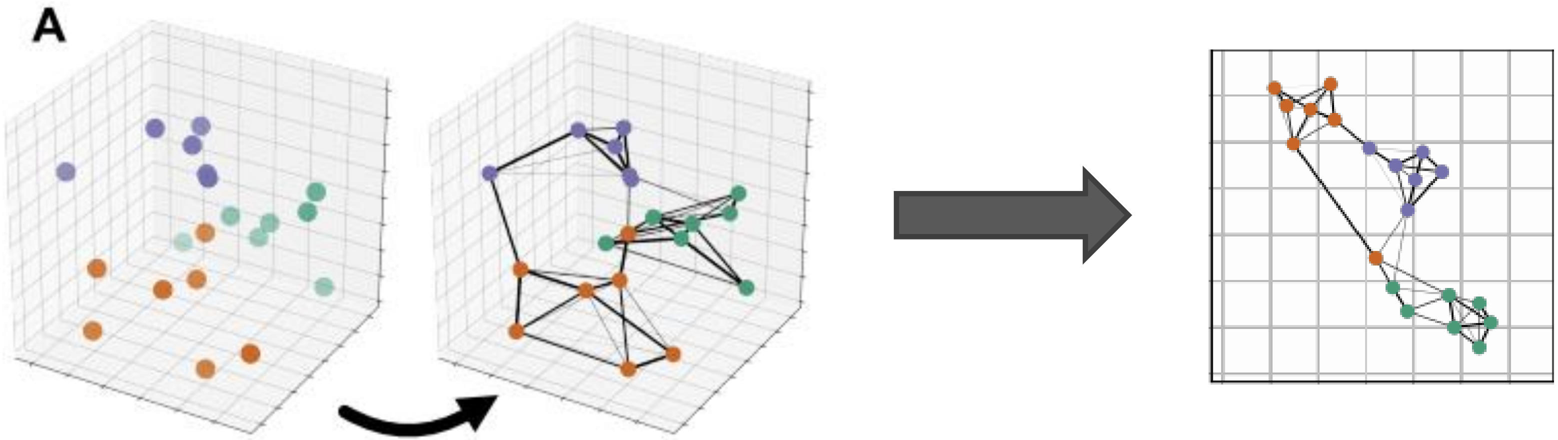


Similar to the probability of being neighbors in *t*-SNE

Network representation of neighbor relationship



Projecting network representation



Sainburg, T. *et al.*, Neural Comput 33(11):2881-2907 (2021)

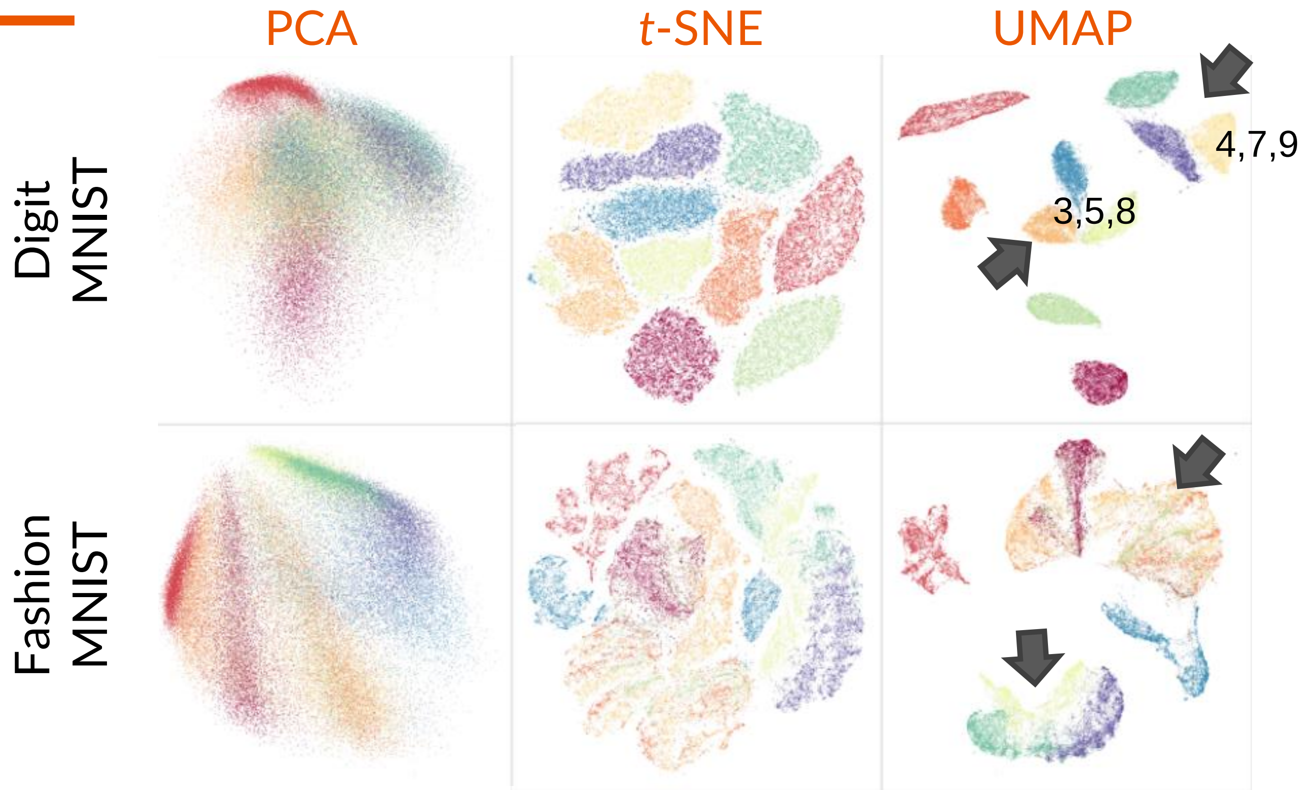
- Preserve scores on edges: probability of being neighbors

Another similarity score for probability distribution

- Cross-entropy
  - $\mathrm{CE}(P \mid\mid Q) = \sum_i \sum_j p_{i,j} \log_2 \frac{p_{i,j}}{q_{i,j}} + \sum_i \sum_j (1 - p_{i,j}) \log_2 \frac{(1-p_{i,j})}{(1-q_{i,j})}$

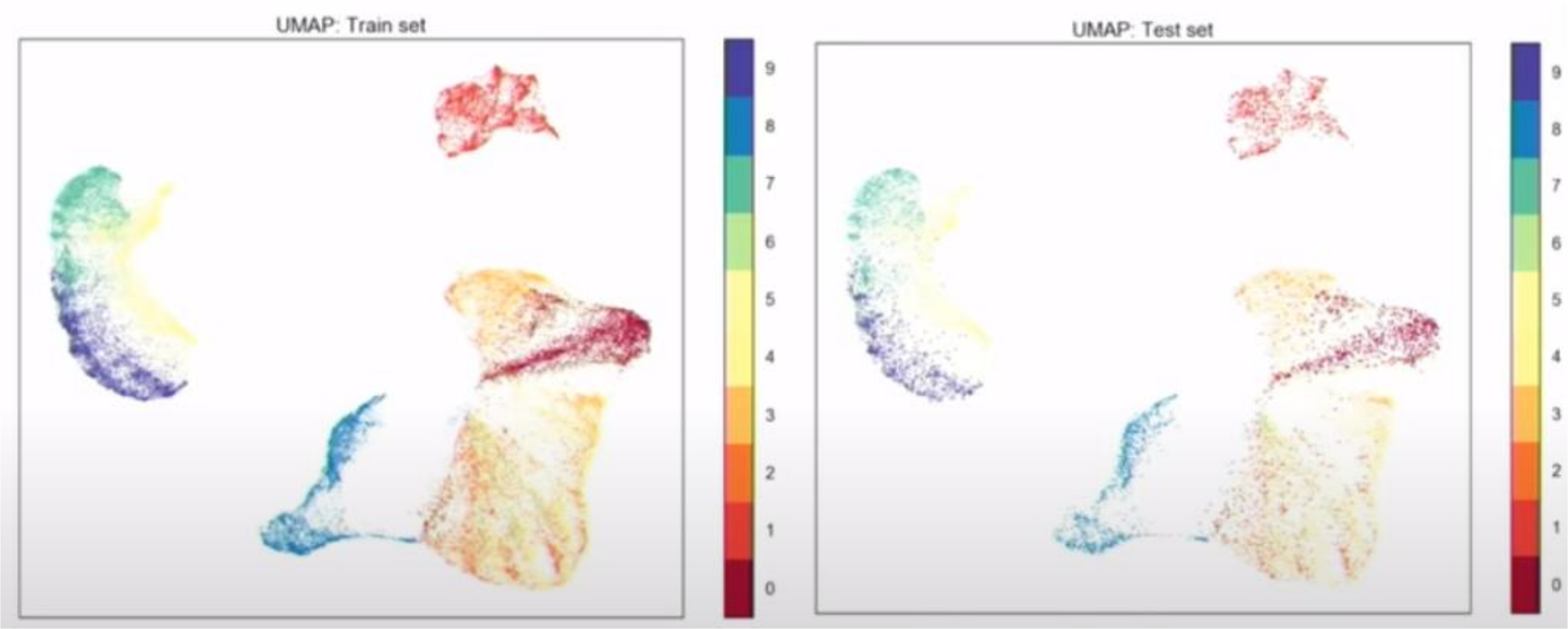
- KL divergence only has the first term
- Cross-entropy considers both when $p_{i,j}$ is high (similar data points) and when $p_{i,j}$ is low (distant data points)

Power of UMAP



McInnes, L., Healy, J. and Melville, J. “UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction”

UMAP can transform new data points



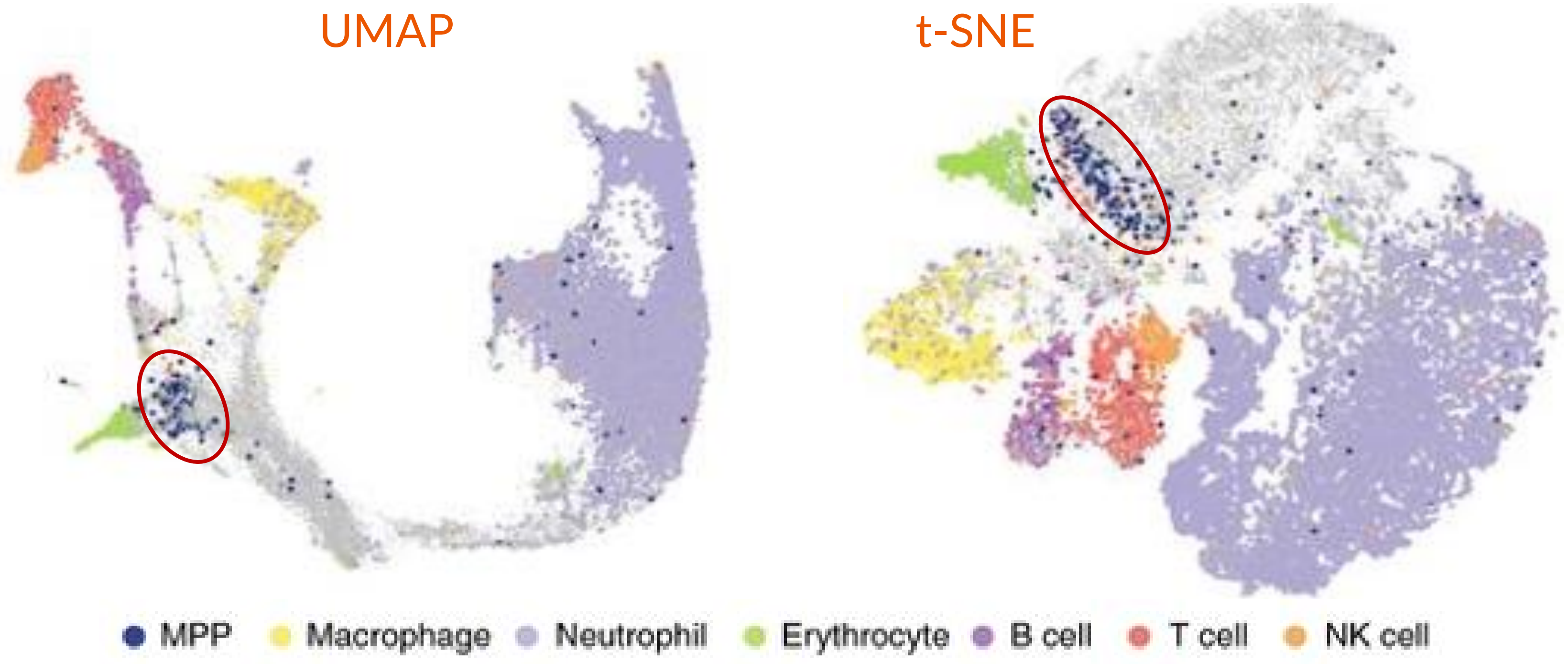
UMAP presentation by Dr. McInnes

Pros and cons of UMAP

- Can capture long-range relationship
- Can be applied to new data points without recomputing

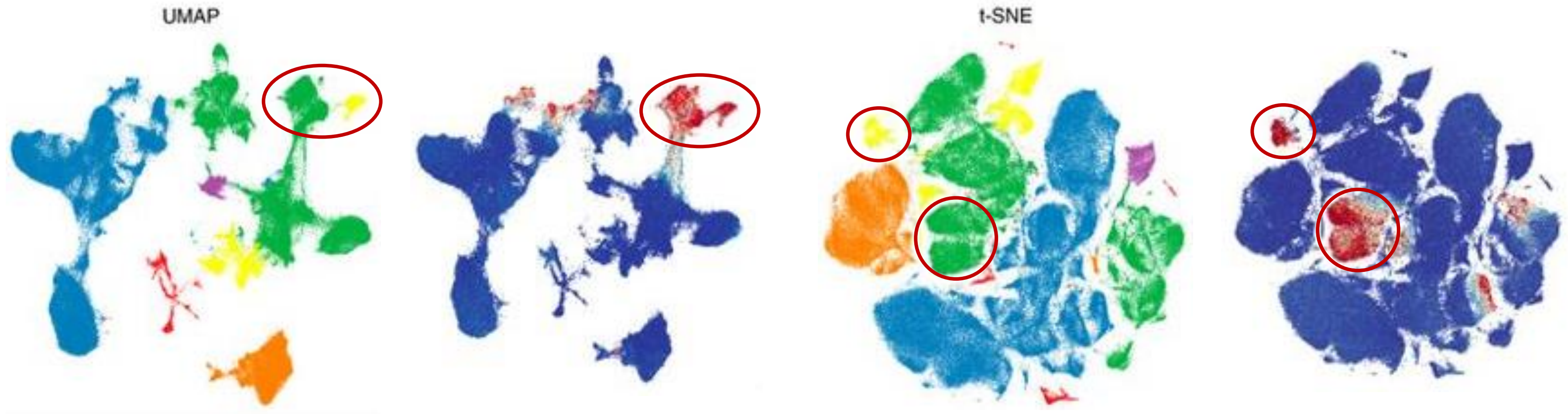
- Require a strong assumption of uniform sampling

t-SNE vs UMAP on biological data



Becht, E. et al. Nature Biotechnology 37:38-44 (2019)

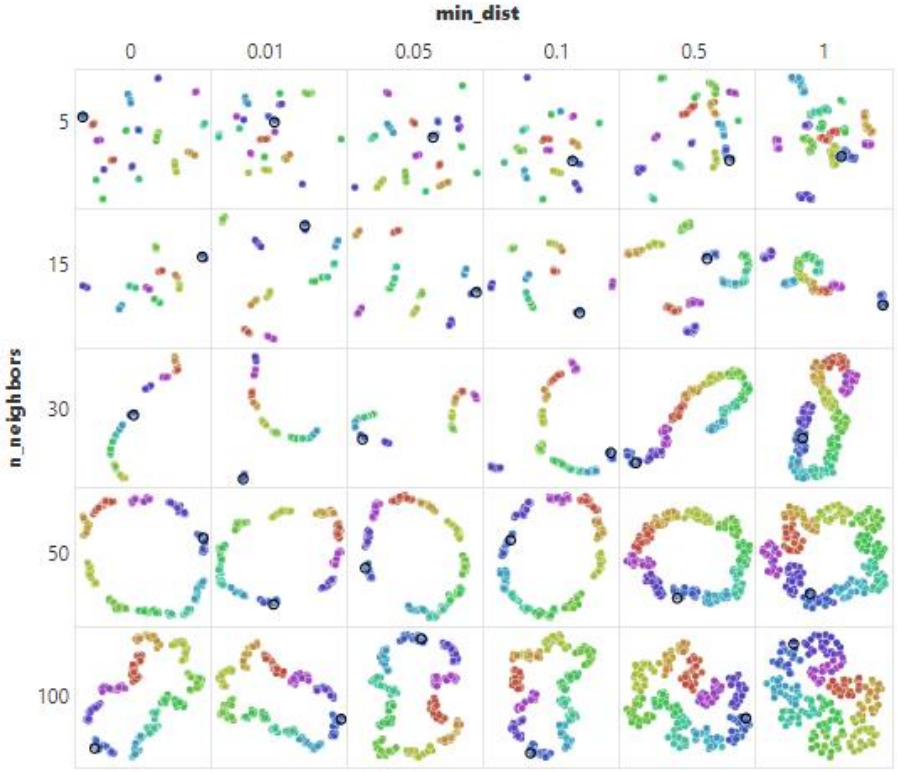
t-SNE vs UMAP on biological data



Becht, E. et al. Nature Biotechnology 37:38-44 (2019)

- Both are equally good at detecting individual data groups
- But UMAP is better at capturing transitions across data groups

Customizing UMAP outputs



- Number of neighbors (n_neighbors) is perplexity

- Minimum distant for placing similar data point (min_dist) is for adjusting the scale of visualization

Source: https://pair-code.github.io/understanding-umap/

Clustering

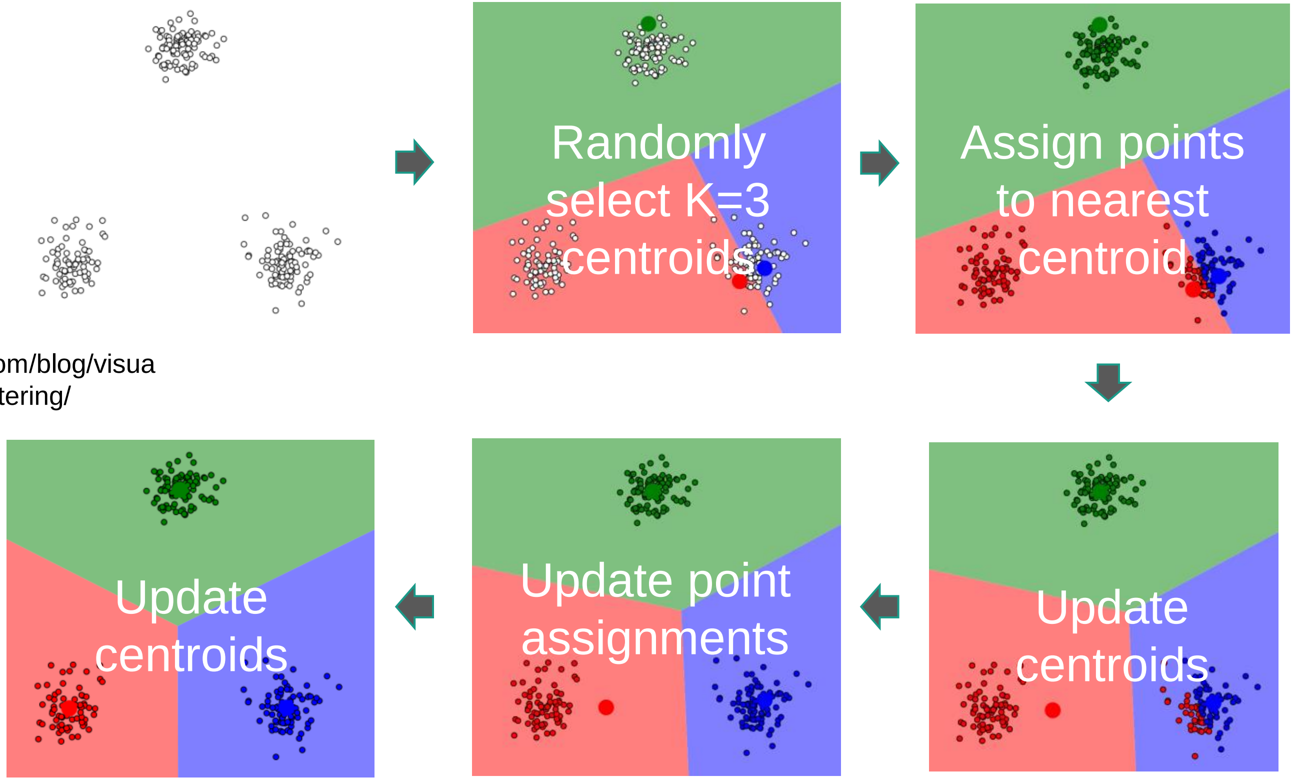
The heart of clustering

- **Goal**: Group similar data point together

- How to define similarity?
  - Distance: Between two data points
  - Linkage: Between groups of data points

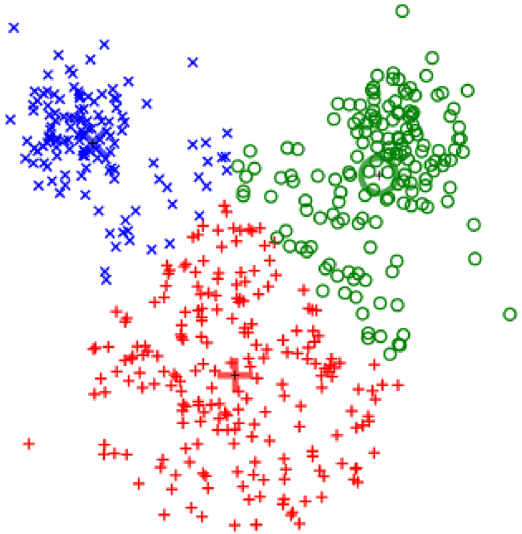
- How many clusters is appropriate?
  - Within-cluster (small) versus between-cluster (large) distance

An example: *k*-mean clustering



www.naftaliharris.com/blog/visualizing-k-means-clustering/

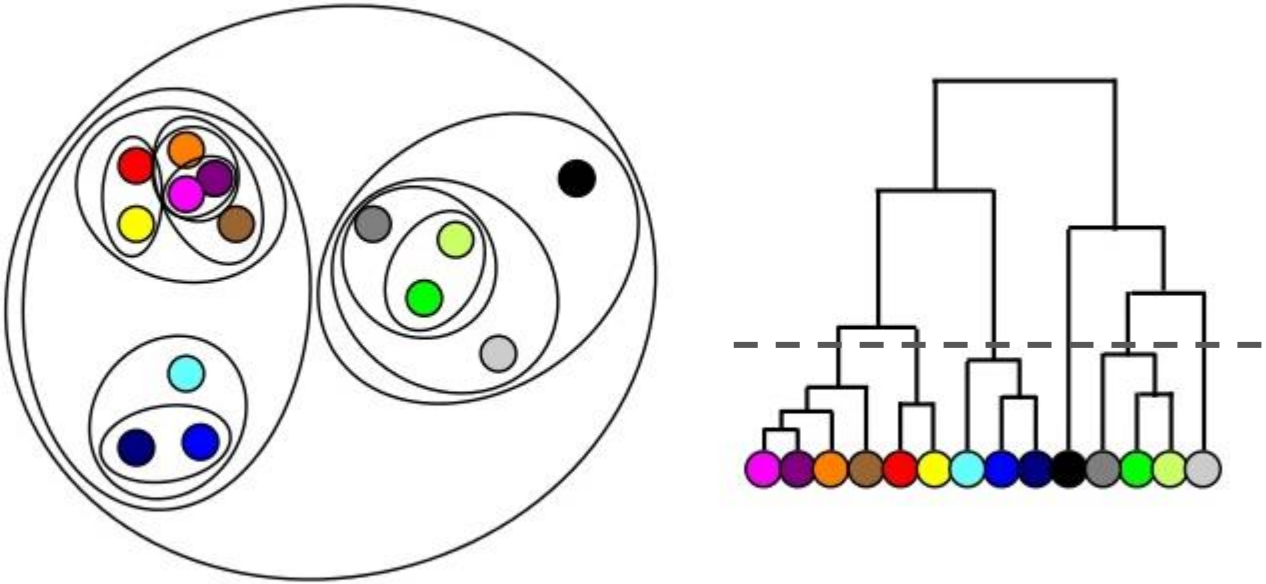
Limitation of *k*-mean



https://en.wikipedia.org/wiki/K-means_clustering

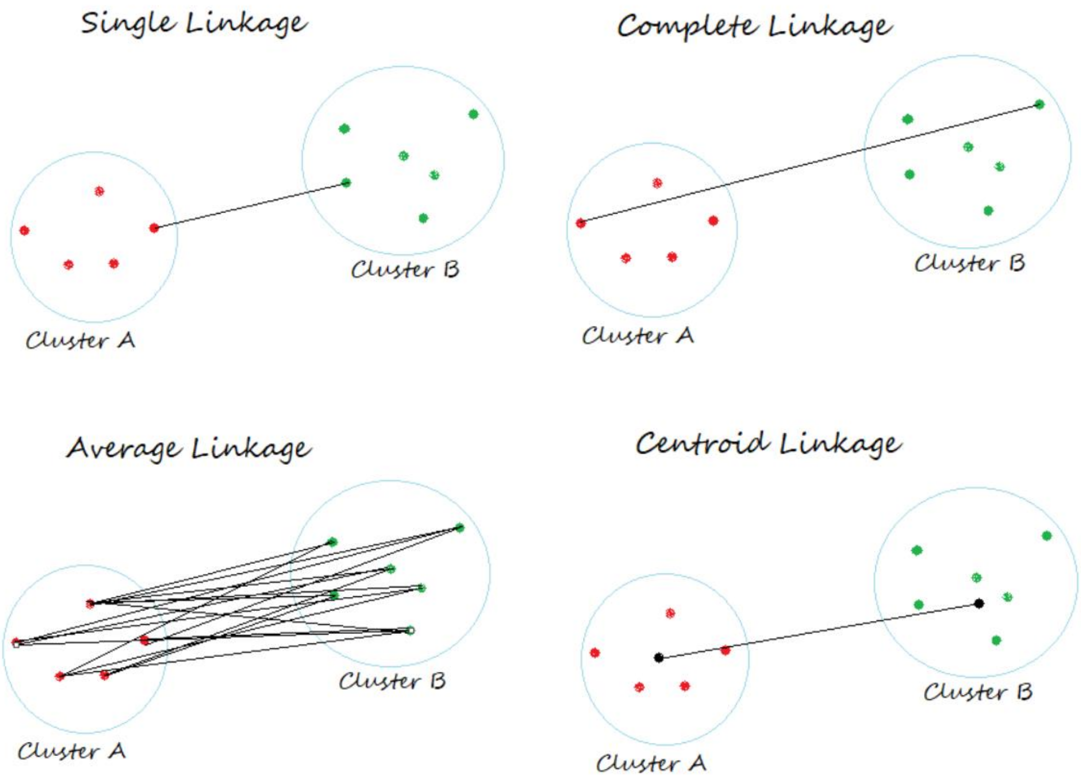
- Assume Euclidean distance
- Assume that clusters are of equal radius
- The initial guess of the locations of *k* means can affect the final clusters
  - Repeat multiple times

Agglomerative / Hierarchical clustering



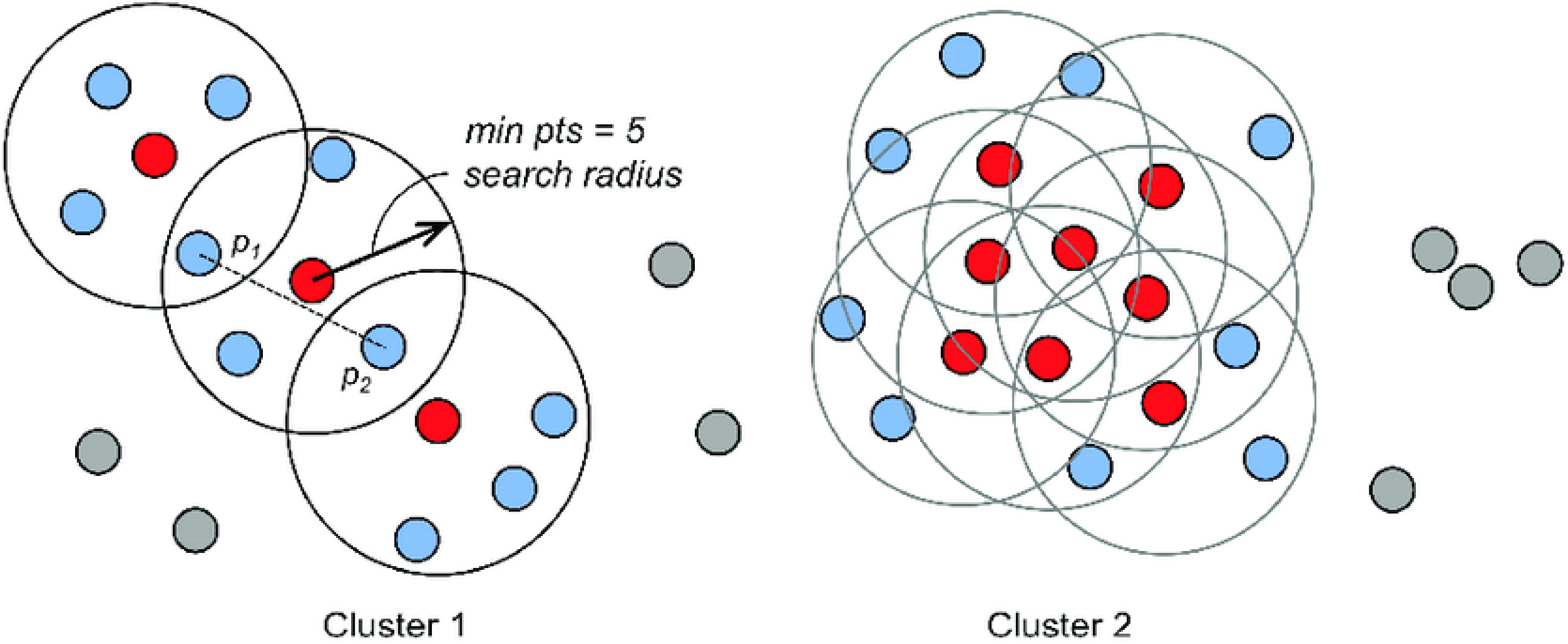
Source: www.slideshare.net/ElenaSgis/data-preprocessing-and-unsupervised-learning-methods-in-bioinformatics

Linkage = distance metric for groups of data points



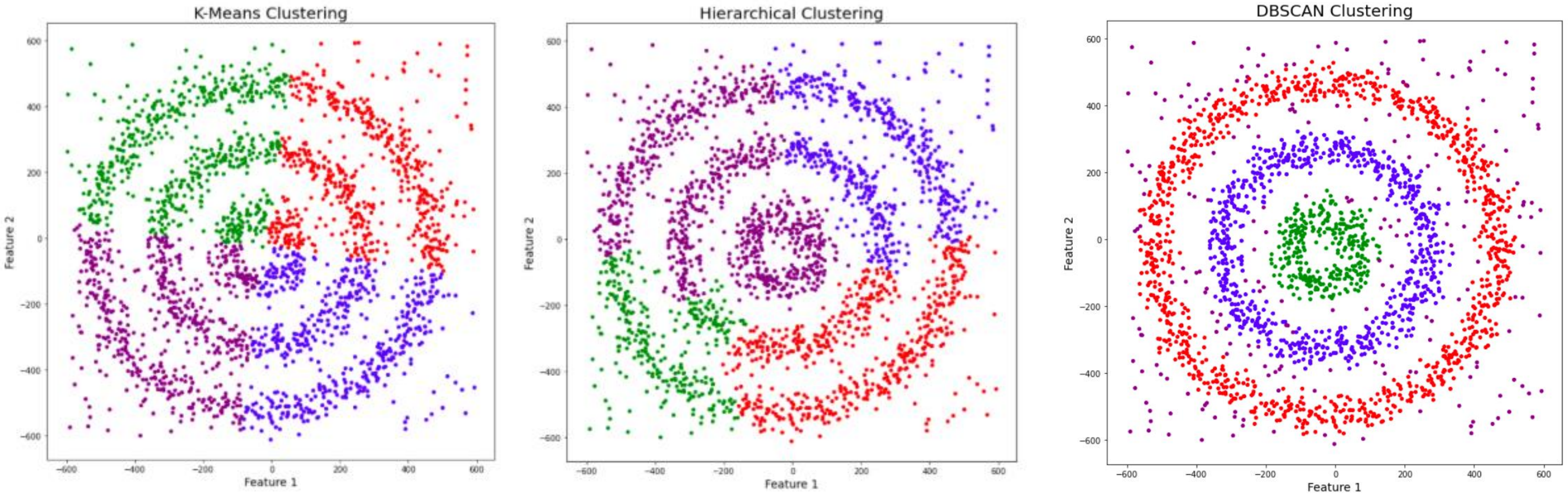
https://www.analyticsvidhya.com/blog/2021/06/single-link-hierarchical-clustering-clearly-explained/

DBSCAN: A density-based technique



Difrancesco, P.-M. Remote Sensing 12:1885 (2020)

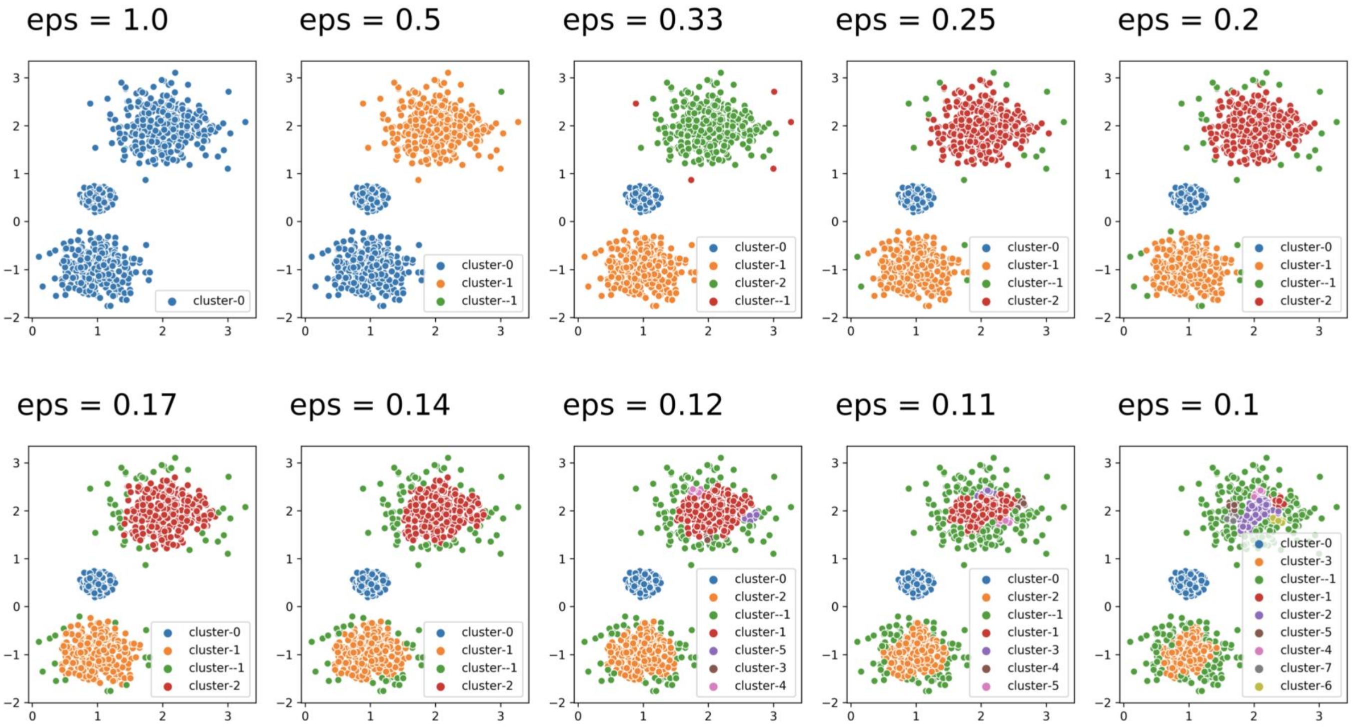
DBSCAN can handle complex cluster shape



https://www.analyticsvidhya.com/blog/2020/09/how-dbscan-clustering-works/

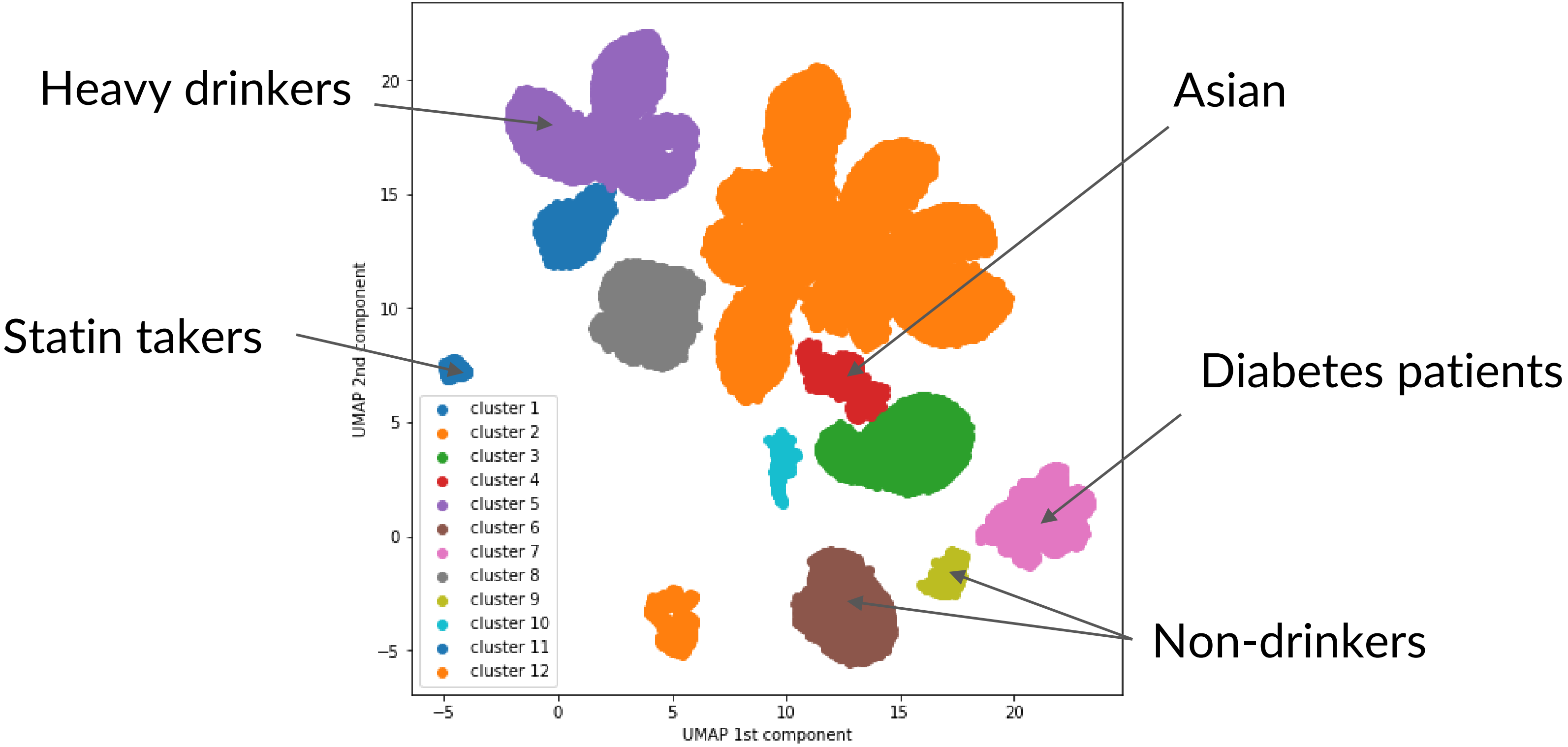
- Distance-based techniques assume that data are spread in all directions

Tuning DBSCAN

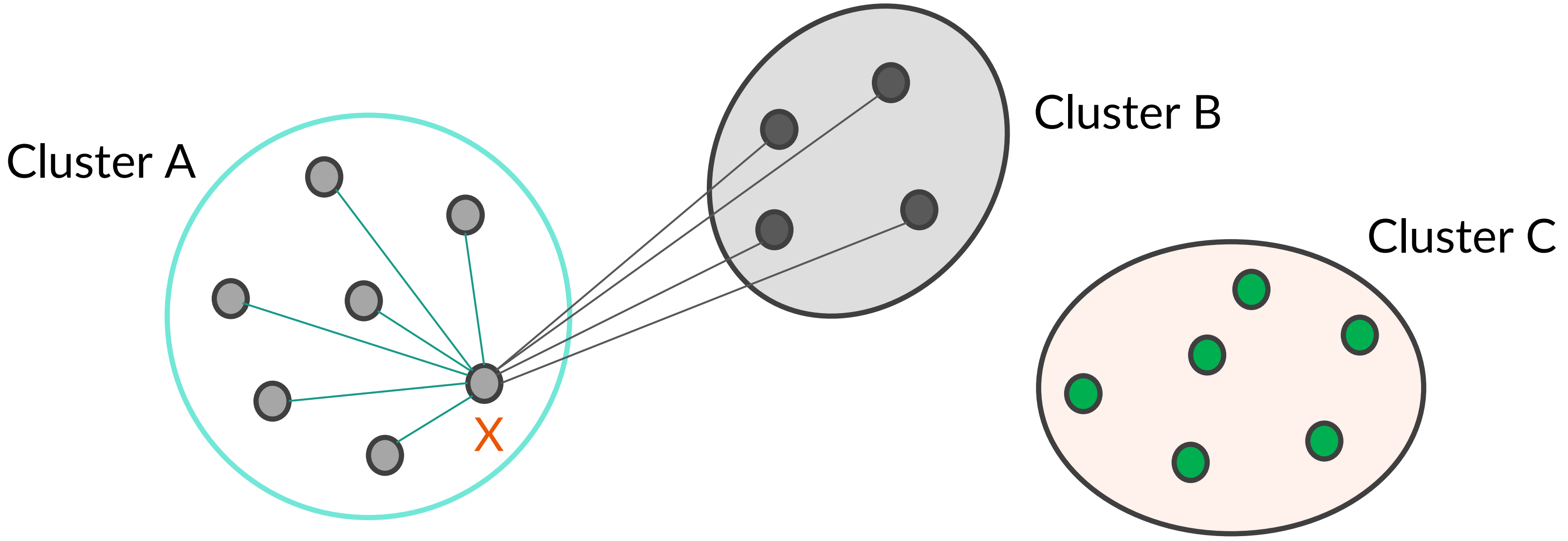


https://towardsdatascience.com/

An example: DBSCAN on UK Biobank data

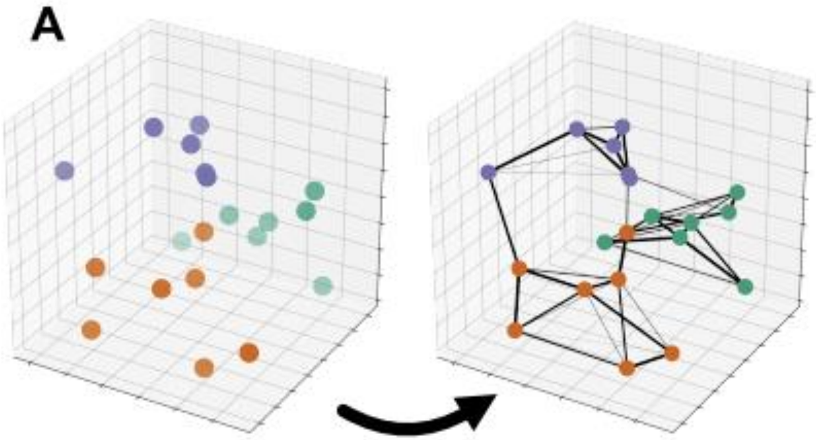


Silhouette score

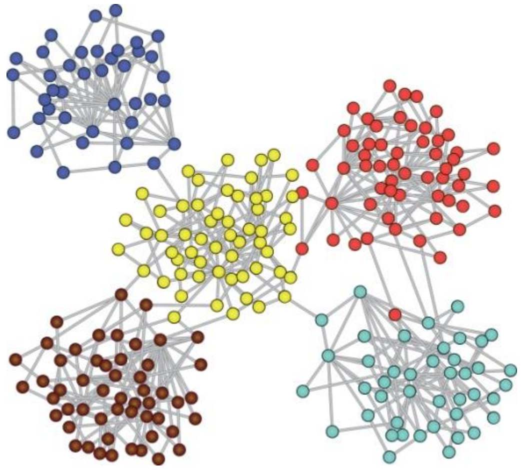


- Compare distances from X to other members of cluster A versus distances from X to members of cluster B (the closest cluster from A)

Spectral clustering and network clustering



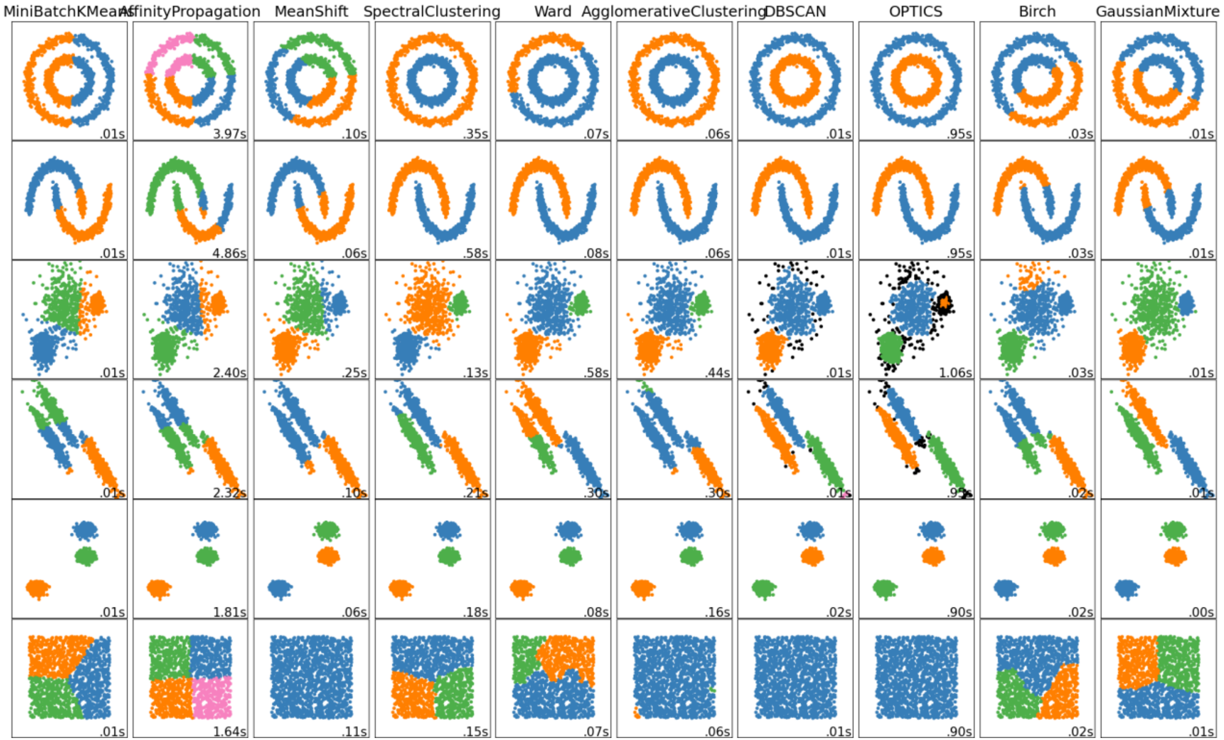
Sainburg, T. *et al.*, Neural Comput 33(11):2881-2907 (2021)



https://github.com/topics/graph-clustering

- View distance matrix as network
- Apply some threshold on the distance to create sparse network
- Split network into modules with dense edges

No one-size fits all



https://scikit-learn.org/0.23/auto_examples/cluster/plot_cluster_comparison.html

Summary

- High-dimensional data can often be simplified onto a 2D/3D visualization
- Explore distribution of feature values on the 2D/3D plot

- Picking appropriate distance metric is the key!
    - Using all features vs informative features
    - $\text{Euclidean}(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \cdots + (x_n - y_n)^2}$

- Unsupervised learning needs domain knowledge for soft validation

Any question?

- Next Tuesday, we will do a demo for unsupervised learning

- Next Thursday, we won’t have class (APEC special holiday), but I will upload a video on supervised learning as scheduled
  - The next problem set will be our last (PS9)
  - Then, you will have to pick papers to present for your final