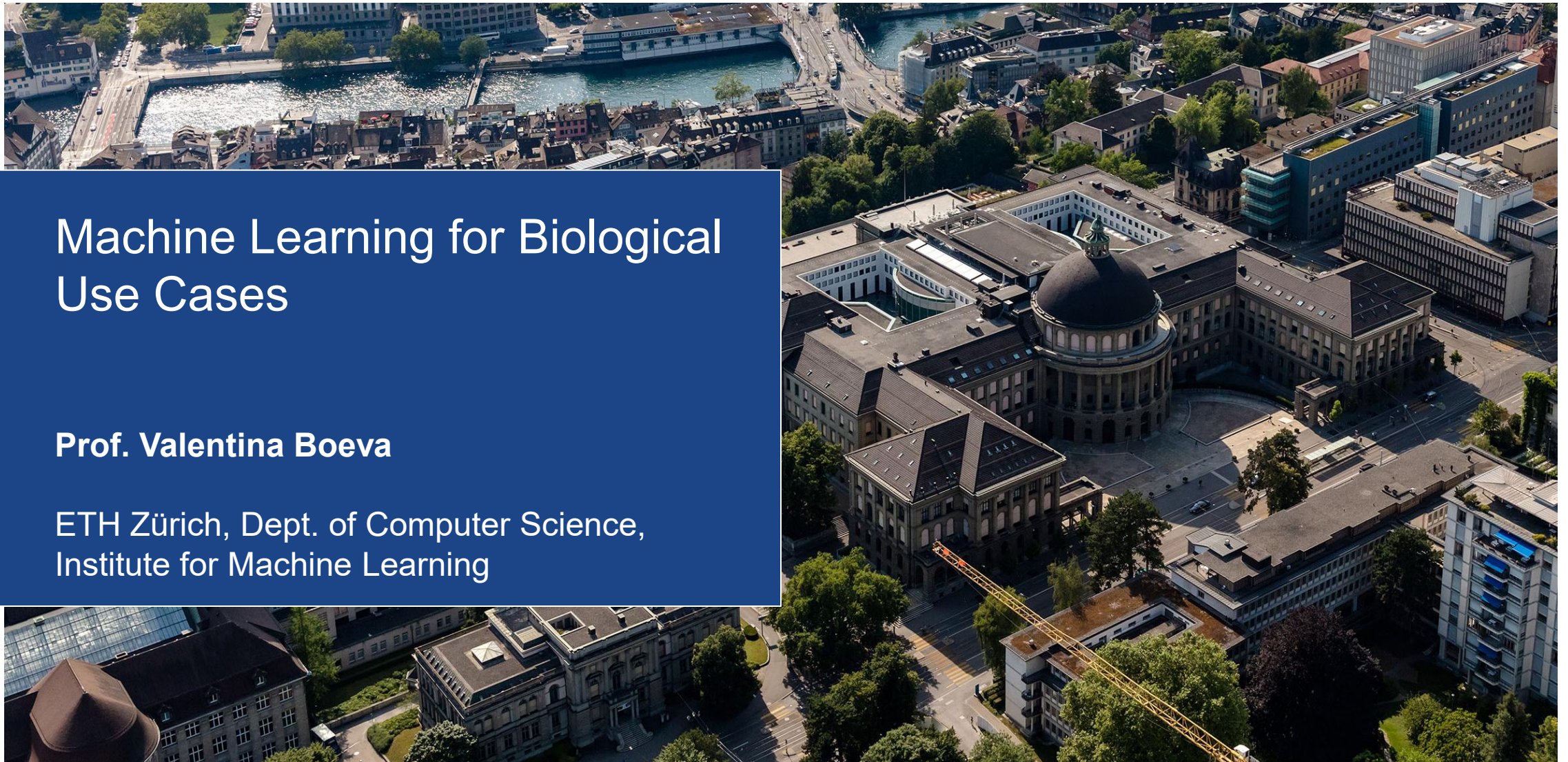


Machine Learning for Biological Use Cases

Prof. Valentina Boeva

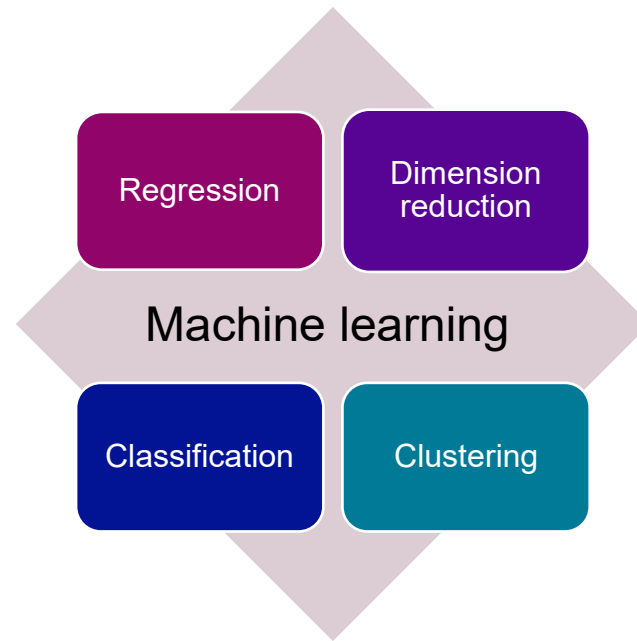
ETH Zürich, Dept. of Computer Science,
Institute for Machine Learning



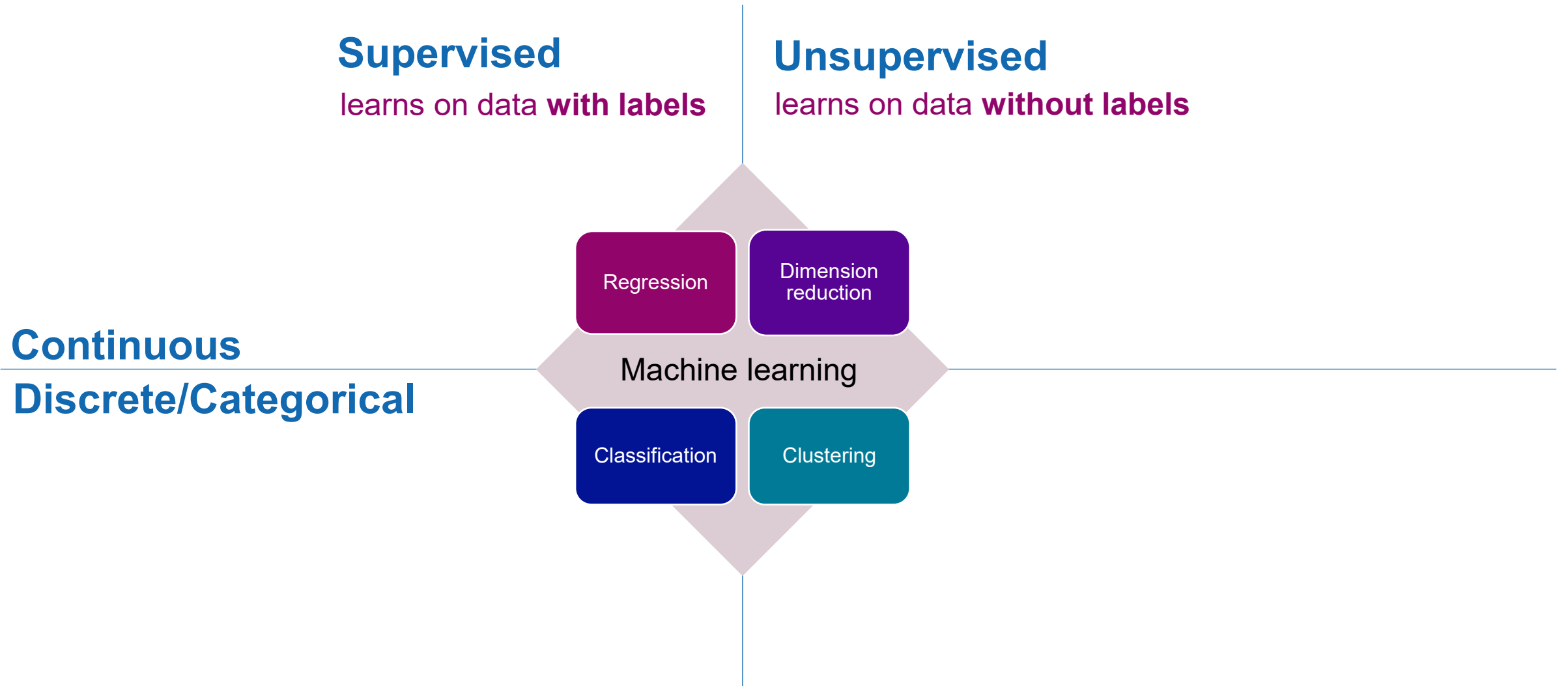
Machine learning



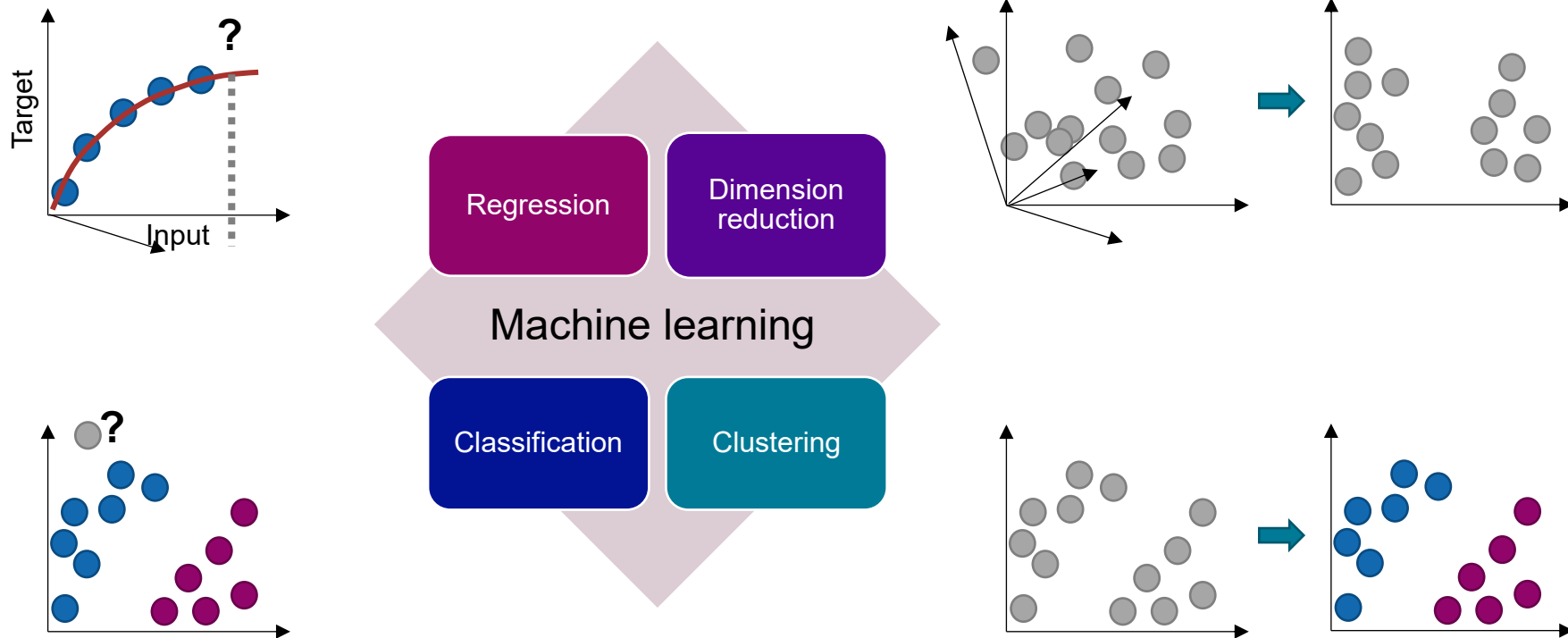
Map of classical machine learning methods



Map of classical machine learning methods



Map of classical machine learning methods

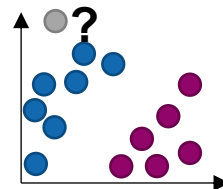
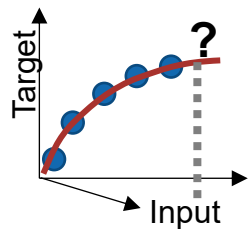


Map of classical machine learning methods

Supervised

learns on data **with labels**

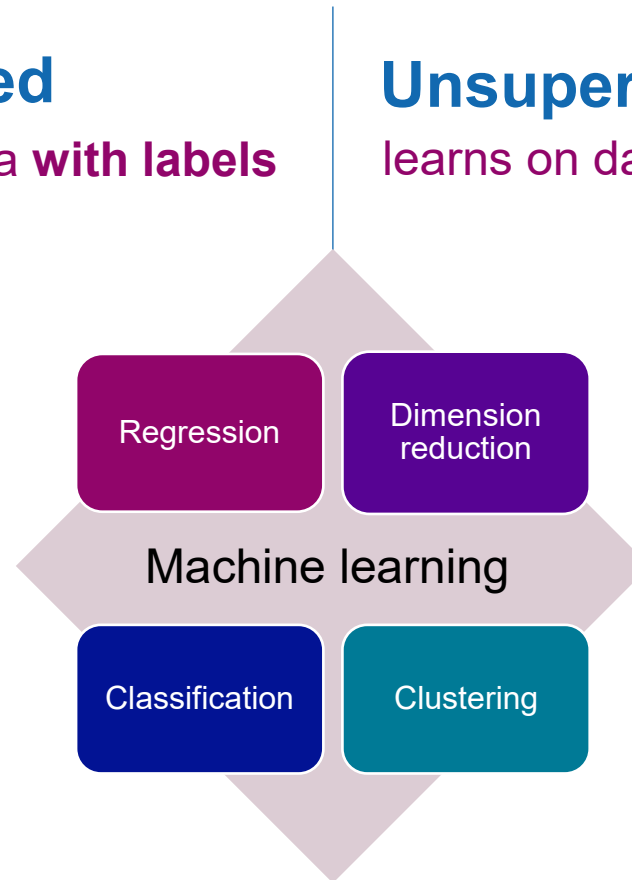
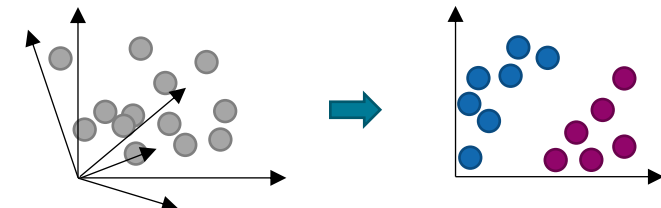
- Automate decisions
 - Primary diagnosis
 - Choice of treatment
- Predict future
 - Treatment response
 - Side effects
- Detect the most “important” features



Unsupervised

learns on data **without labels**

- Understand data structure
- Visualize data in 2D/3D
- Detect hidden features



Types of input data in molecular biology and genetics

Type of data:

- **Genomic:**
 - Variants common in the general population (SNPs)
 - Rare variants
 - Single nucleotide variants, SNV (a.k.a. mutations)
 - Structural aberrations (e.g., translocations, amplifications)
 - Copy number profiles
- **Transcriptomic:**
 - RNA-seq (or expression microarrays) – bulk and single-cell
 - miRNA
- **Epigenetic:**
 - DNA methylation (sequencing or methylation arrays)
 - Histone modifications (ChIP-seq) and open chromatin (ATAC-seq, DHS-seq)
- **Proteomic:**
 - RPPA (bulk), Mass Spec (bulk), CyTOF (single-cell)

Context:

- Common diseases (Alzheimer, asthma, hypertension,...)
- Genetic syndromes (Down syndrome, CHARGE syndrome, ...)
- Cancer

Type of samples:

- Blood samples
- Saliva samples
- Tissue samples
- Maternal blood samples

Omics data are high dimensional

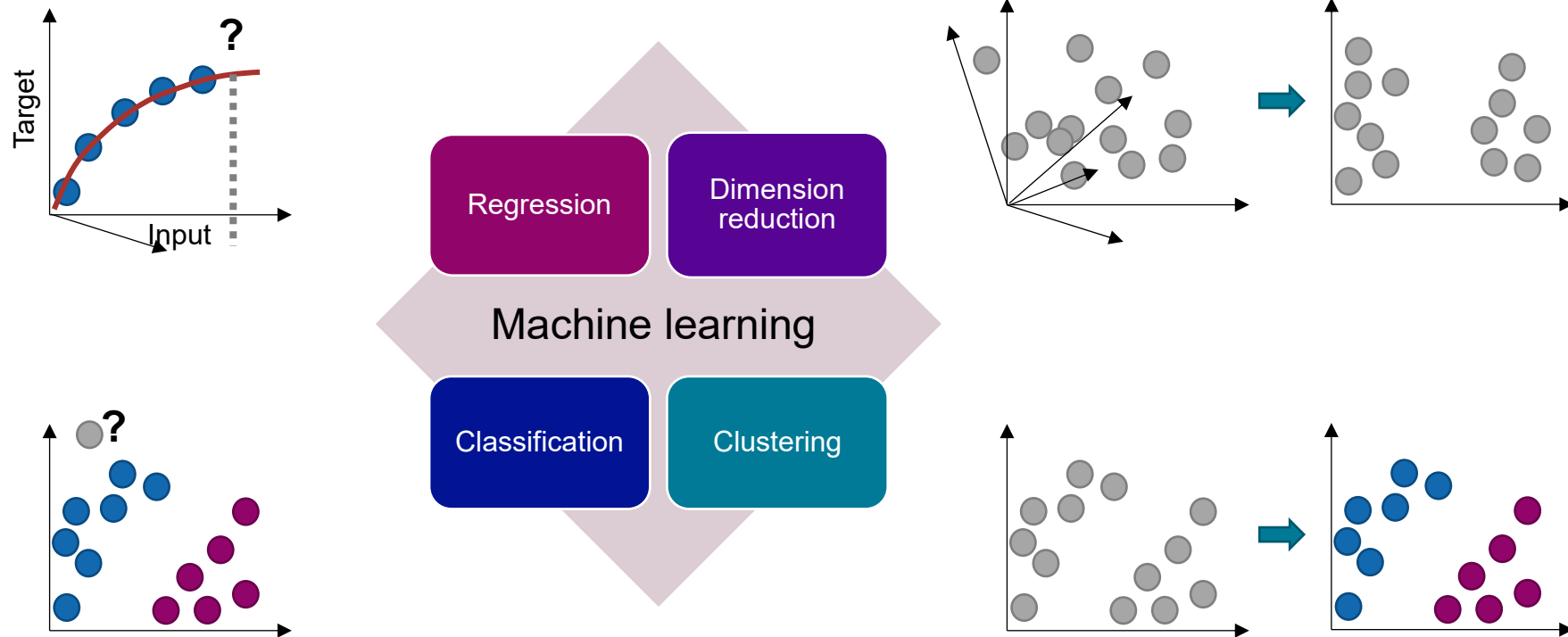
SNPs	Mutations (SNVs)	Structural variants	Copy number alterations	mRNA expression	miRNA expression	DNA methylation
3M-100M	~10-10K	~1-1000	~10-25K	~25K	~1000	27K-28M

+ sometimes these data is complemented with proteomics data (expression of hundreds of proteins)

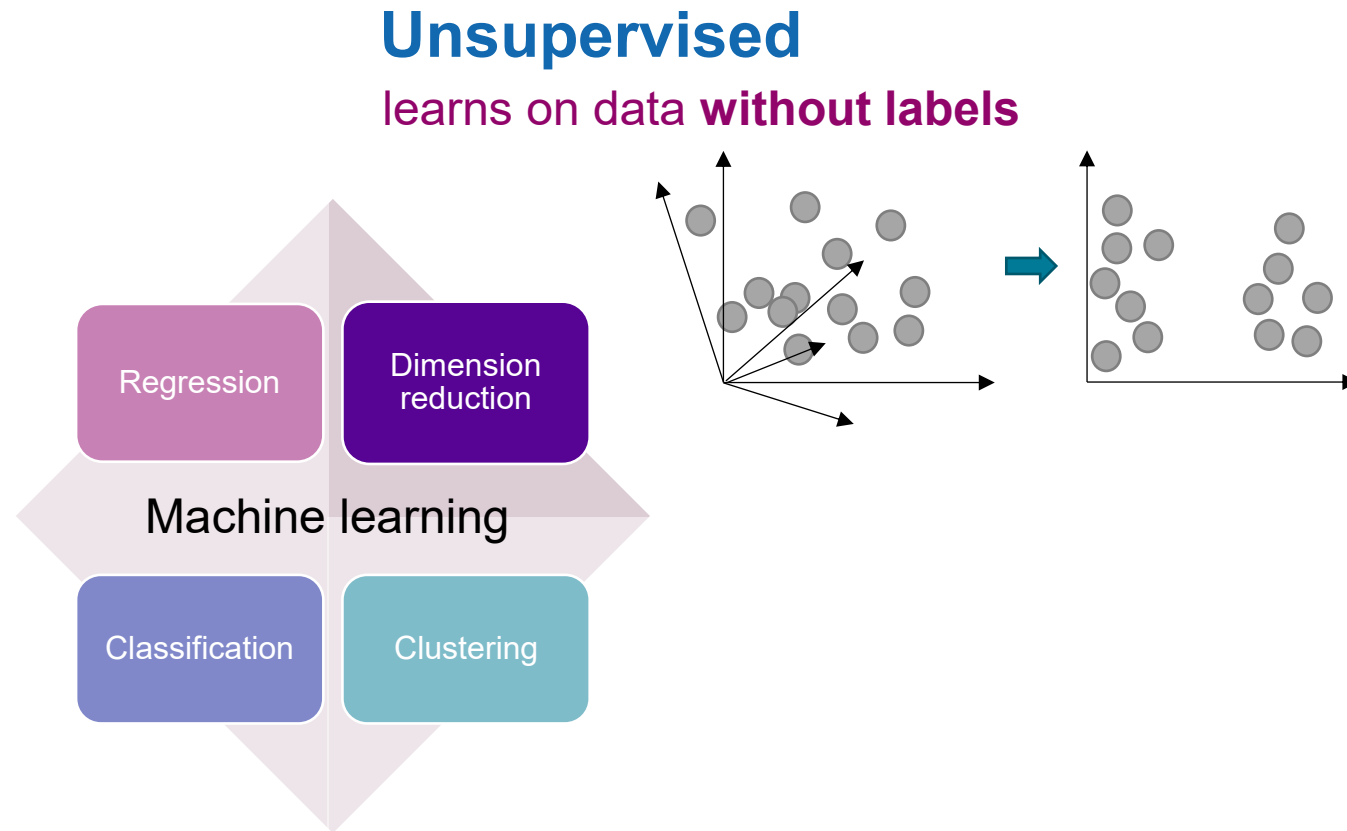
Full –omics dataset millions of observations per patient:

Great challenge to avoid over-fitting and perform feature selection!

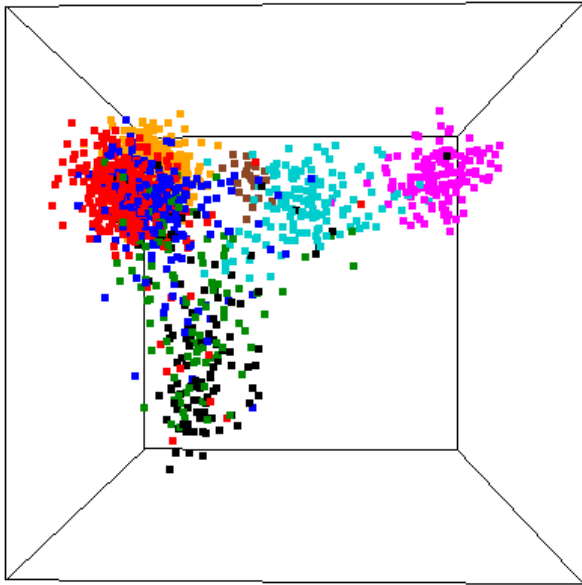
Map of classical machine learning methods



Map of machine learning methods

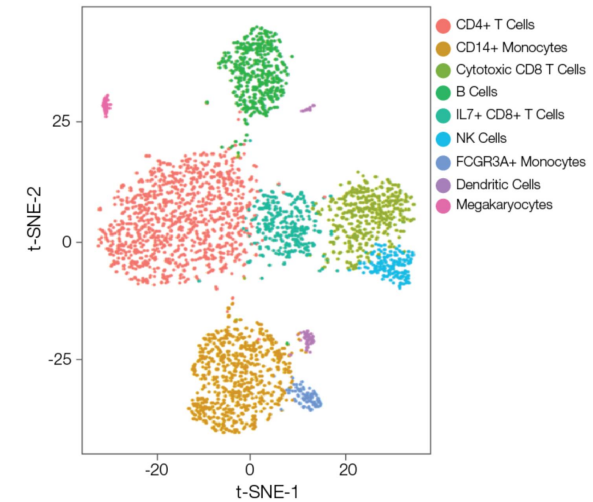
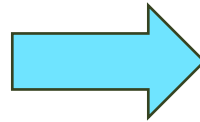


Dimensionality reduction



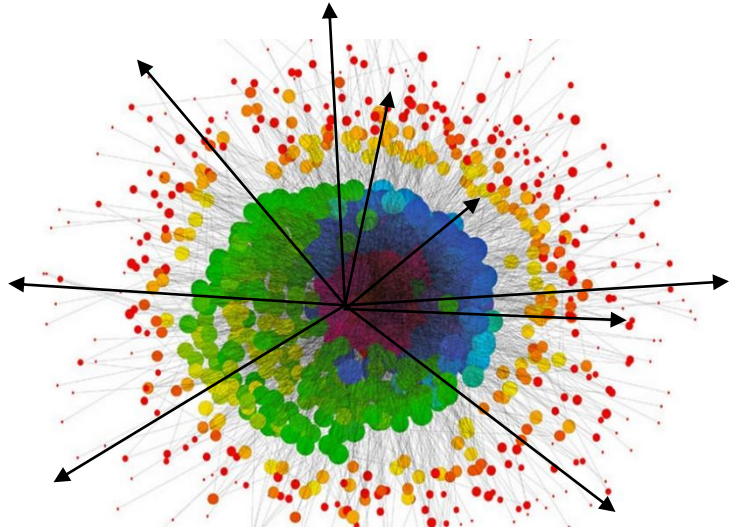
Data points in 3-1000,000 dimensions

**Preserving maximal
information**



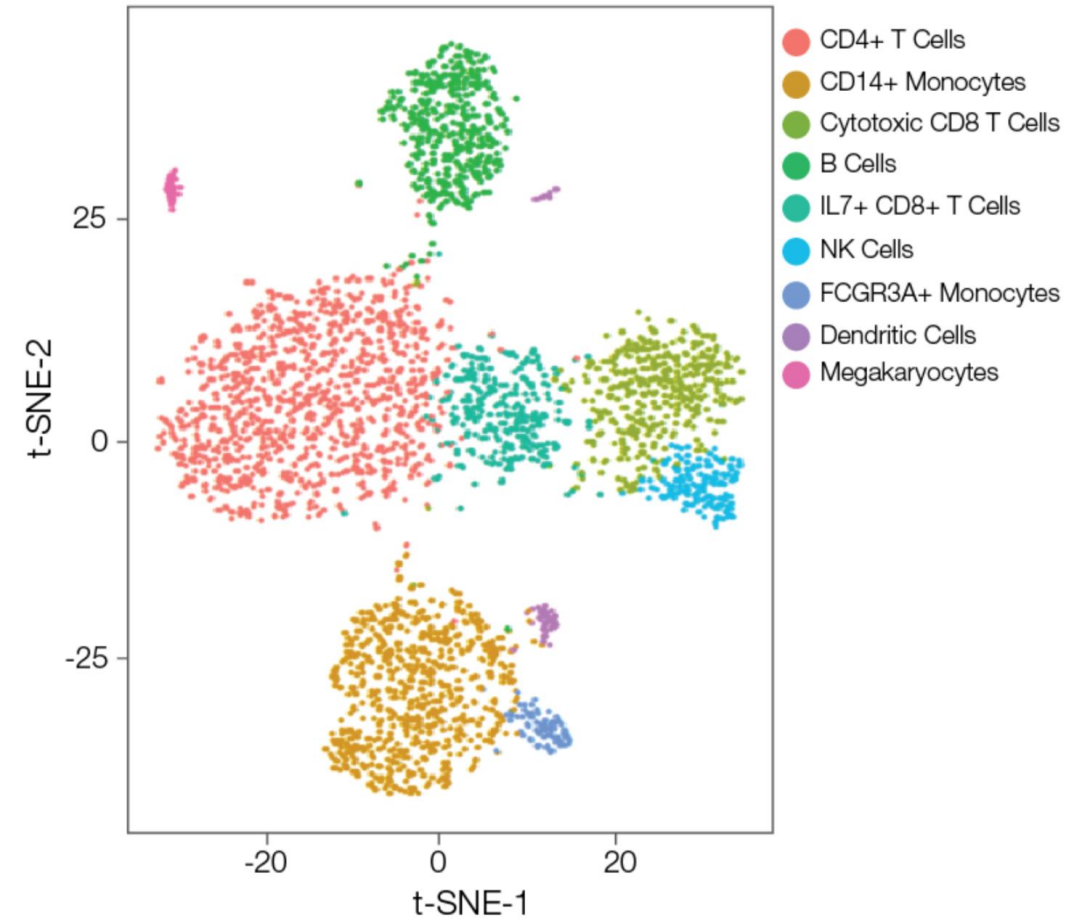
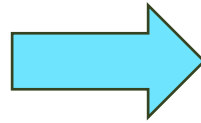
Data points in 2-100 dimensions

Dimensionality reduction



Data points in 3-100,000 dimensions
(e.g., gene expression)

preserving maximal
information

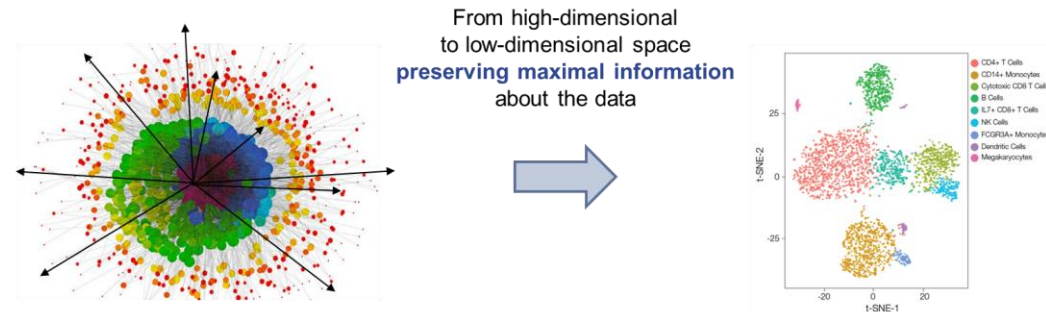


Data points in 2-100 dimensions

Dimensionality reduction

MAIN PURPOSE:

- Visualize data in 2D/3D
- Understand data structure
- Detect hidden features
- Speed up/improve further analysis (e.g., clustering)

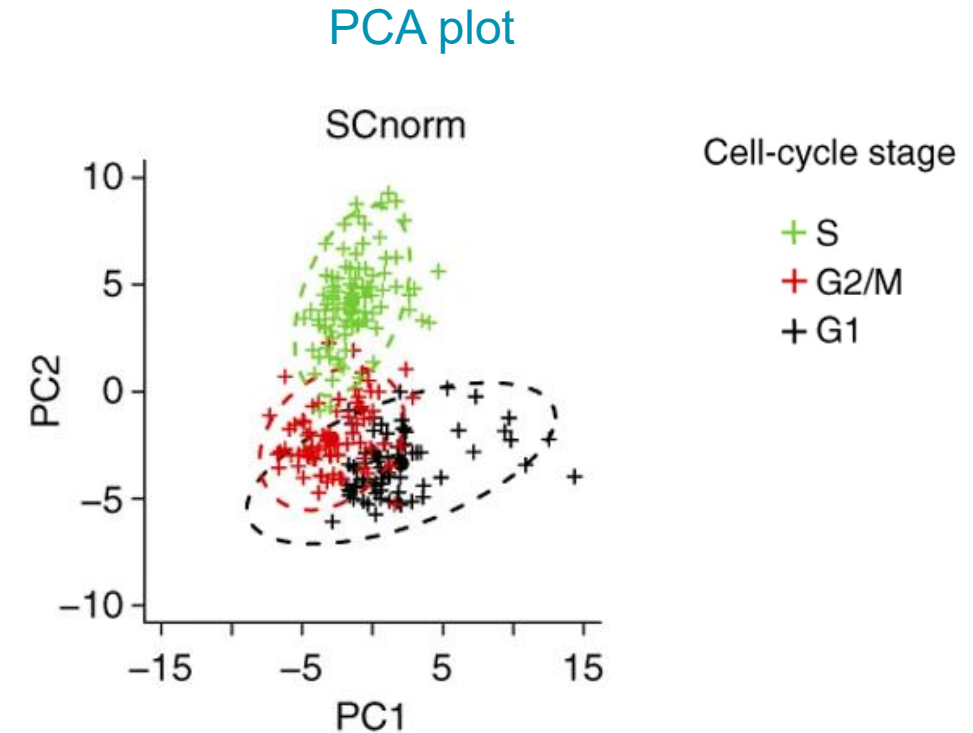


Visualization/dimensionality reduction

Visualization is one of the most important steps in any data analysis

Common visualization strategies:

- **PCA** (principal component analysis)
- **t-SNE** (t-distributed stochastic neighbor embedding)
- **UMAP** (uniform manifold approximation and projection)



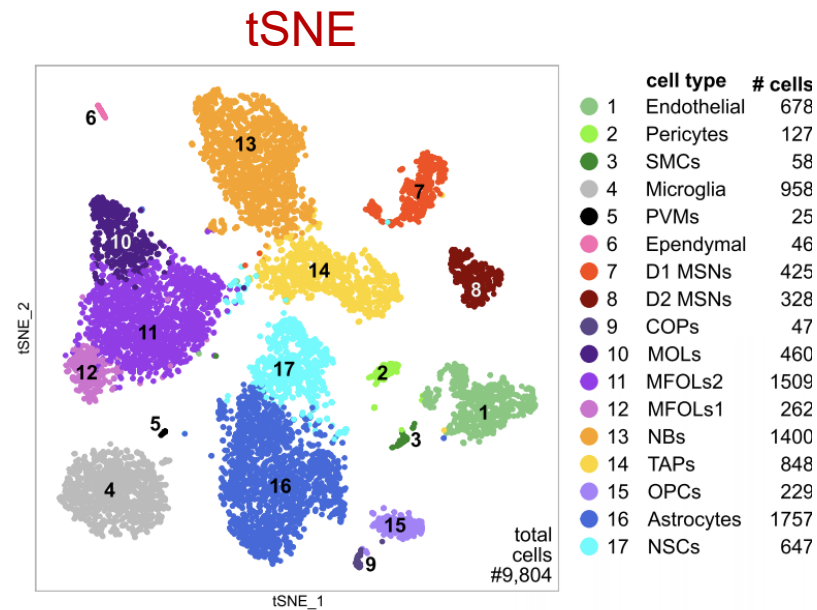
Linear transformation of the data to inspect presence of technical and/or biological biases

Visualization/dimensionality reduction

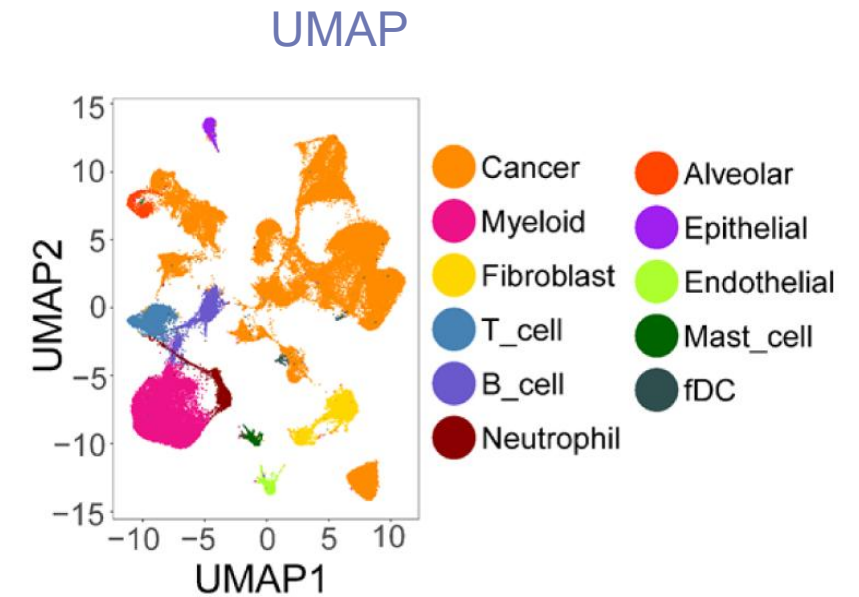
Visualization is one of the most important steps in any data analysis

Common visualization strategies:

- PCA
- t-SNE
- UMAP



From Zywitza, Vera, et al, 2017



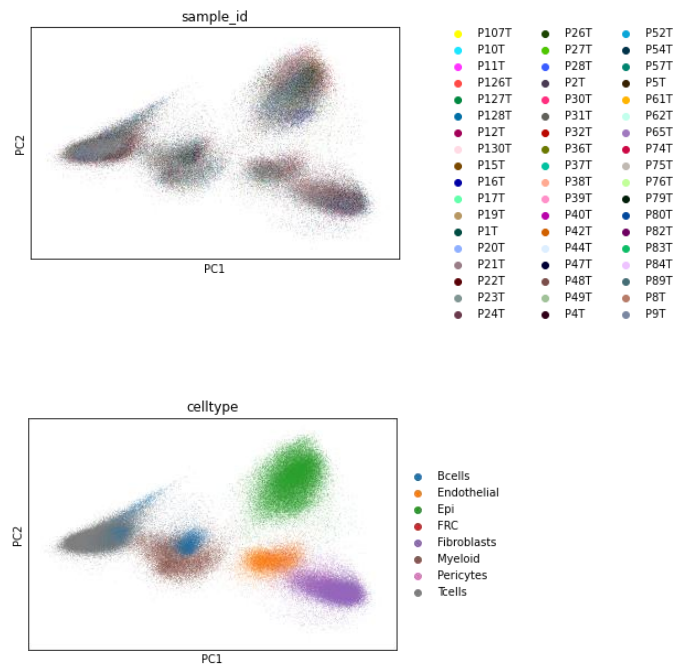
From Wu et al, 2021

Non-linear transformation of the data to better visualize clusters present in the data

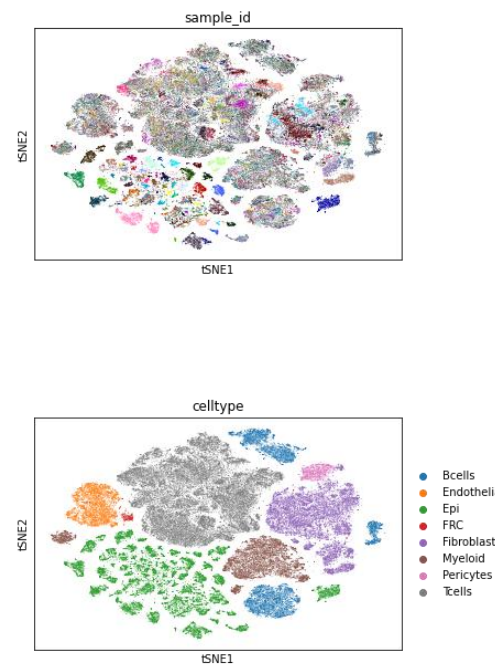
Visualization/dimensionality reduction

- **PCA**: **Linear transformation** of the data to inspect presence of **technical and/or biological biases**
- **t-SNE** and **UMAP**: **Non-linear transformation** of the data to better visualize **clusters** present in the data

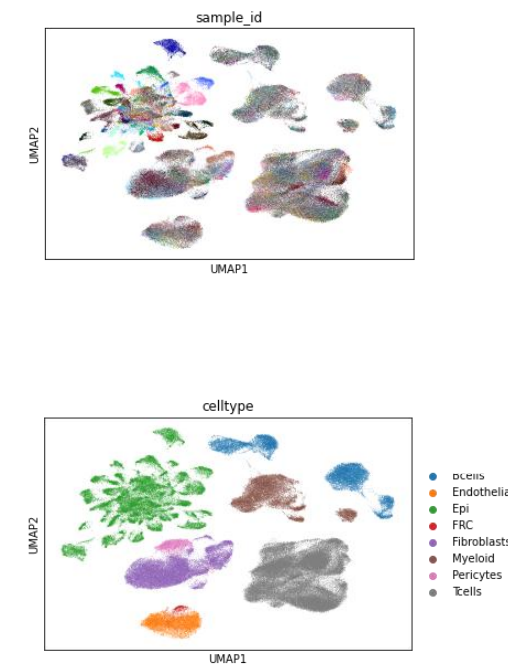
PCA



tSNE



UMAP

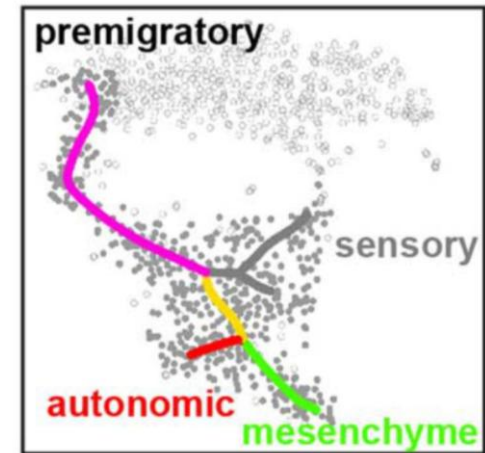
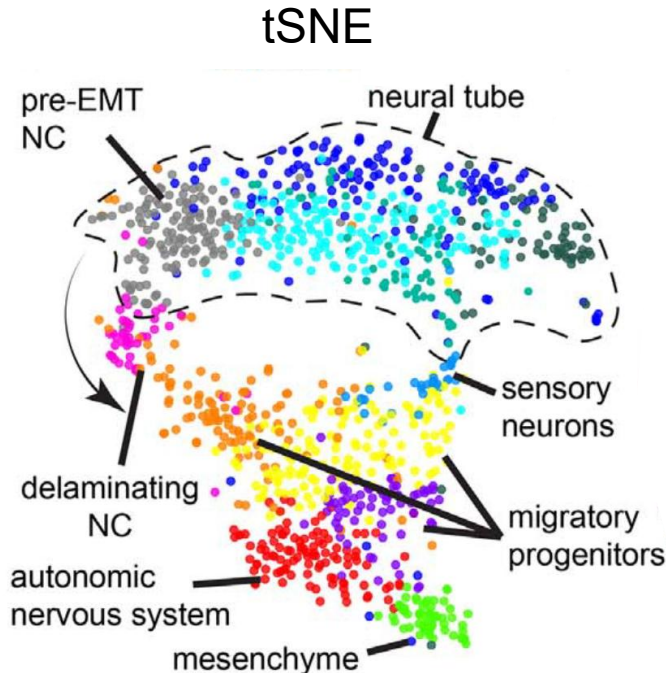


Annotation by sample ID (batch+tumor specific differences)

Annotation by cell type

Examples: Dimension reduction and clustering

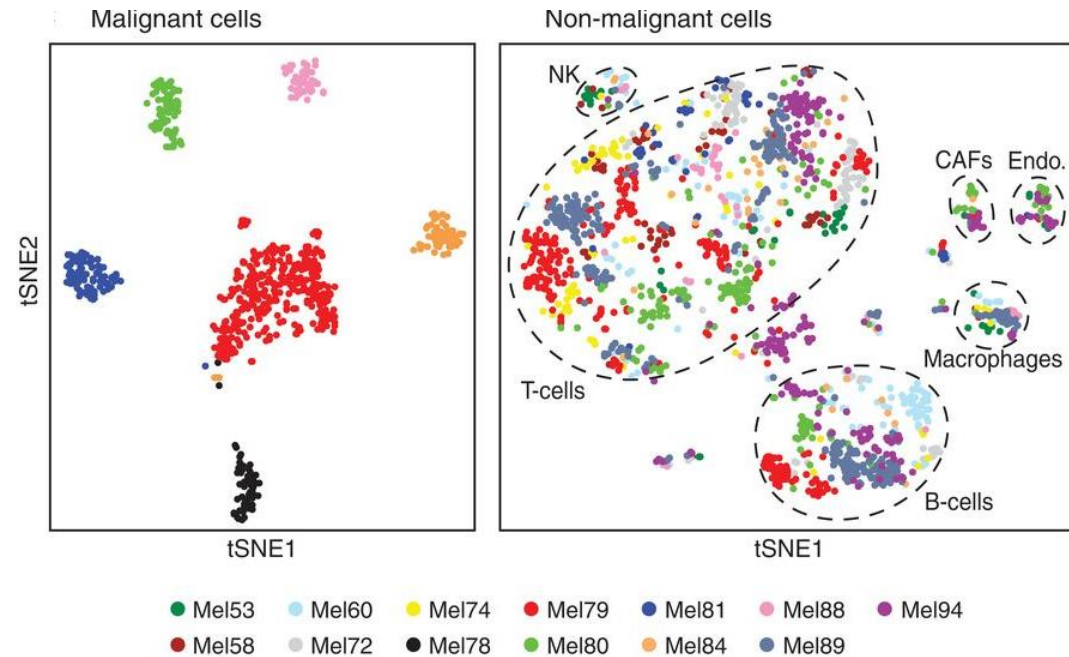
- Single cell transcriptomics: development, cell type heterogeneity and cancer
- Bulk transcriptomics and epigenetics (cancer)
- Bulk transcriptomics: Effect of mutations



Spatiotemporal structure of cell fate decisions in murine neural crest. Soldatov et al., *Science* 364, 971 (2019)

Examples: Dimension reduction and clustering

- Single cell transcriptomics: development, cell type heterogeneity and cancer
- Bulk transcriptomics and epigenetics (cancer)
- Bulk transcriptomics: Effect of mutations

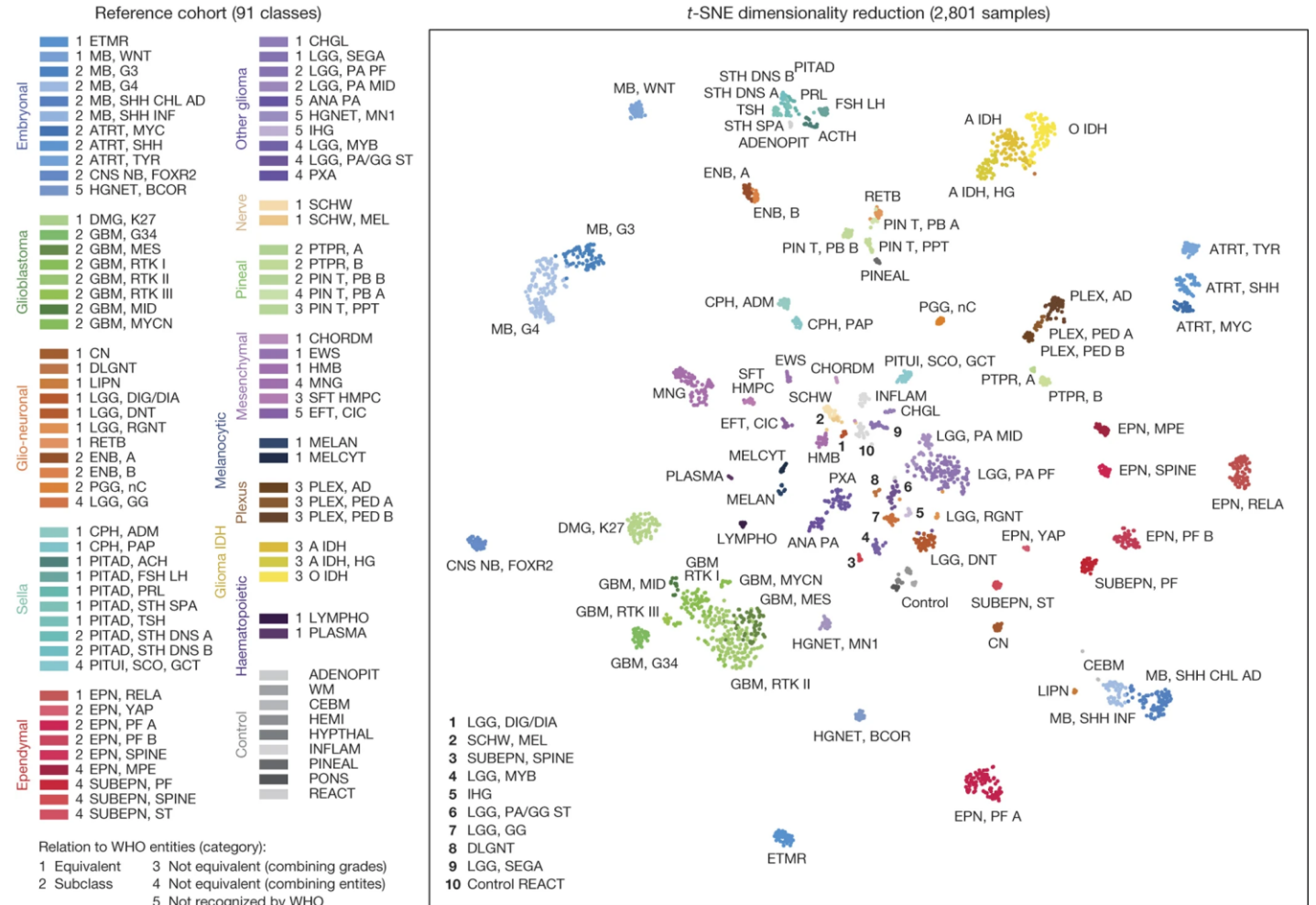


Clusters called by DBScan

Dissecting the multicellular ecosystem of metastatic melanoma by single-cell RNA-seq. Tirosh et al. *Science*. 2016 Apr 8;352(6282):189-96.

Examples: Dimension reduction and clustering

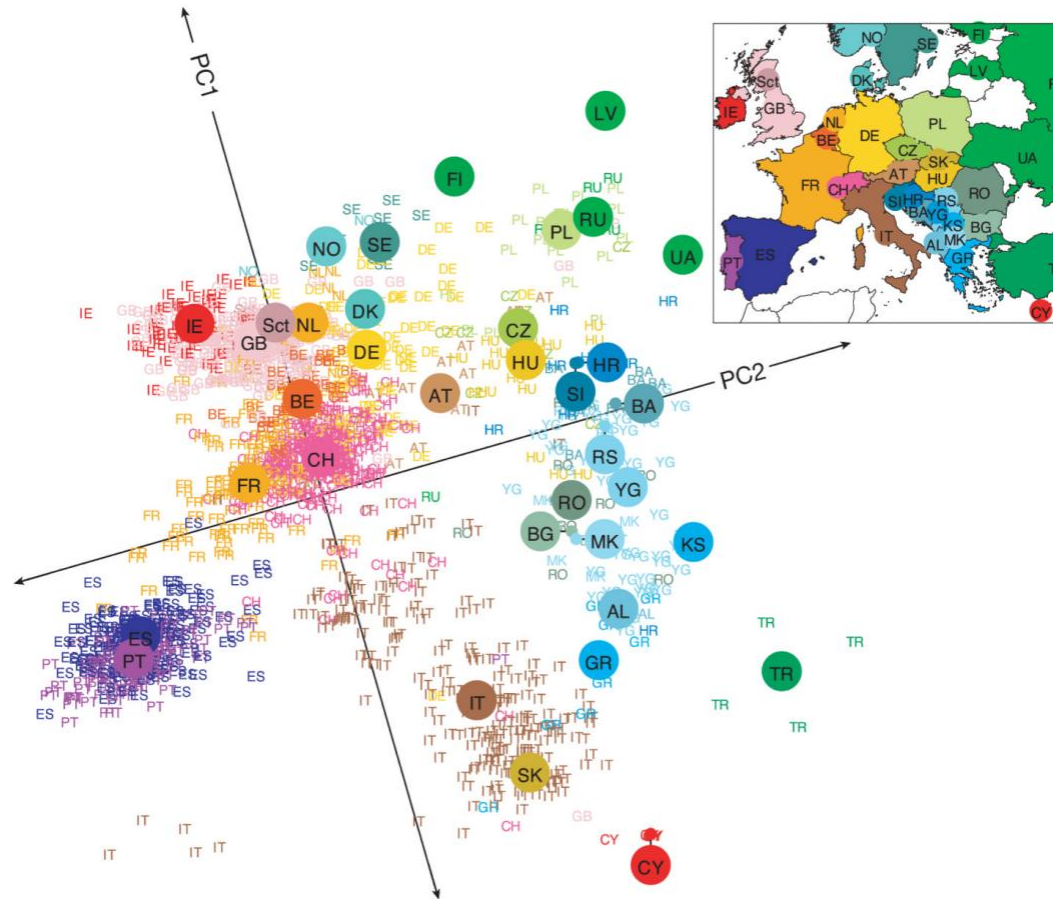
- Single cell transcriptomics: development, cell type heterogeneity and cancer
- Bulk transcriptomics and epigenetics (cancer)
- Bulk transcriptomics: Effect of mutations



DNA methylation-based classification of central nervous system tumours. Capper et al. *Nature* 2018

Examples: Dimension reduction and clustering

- Population genetics

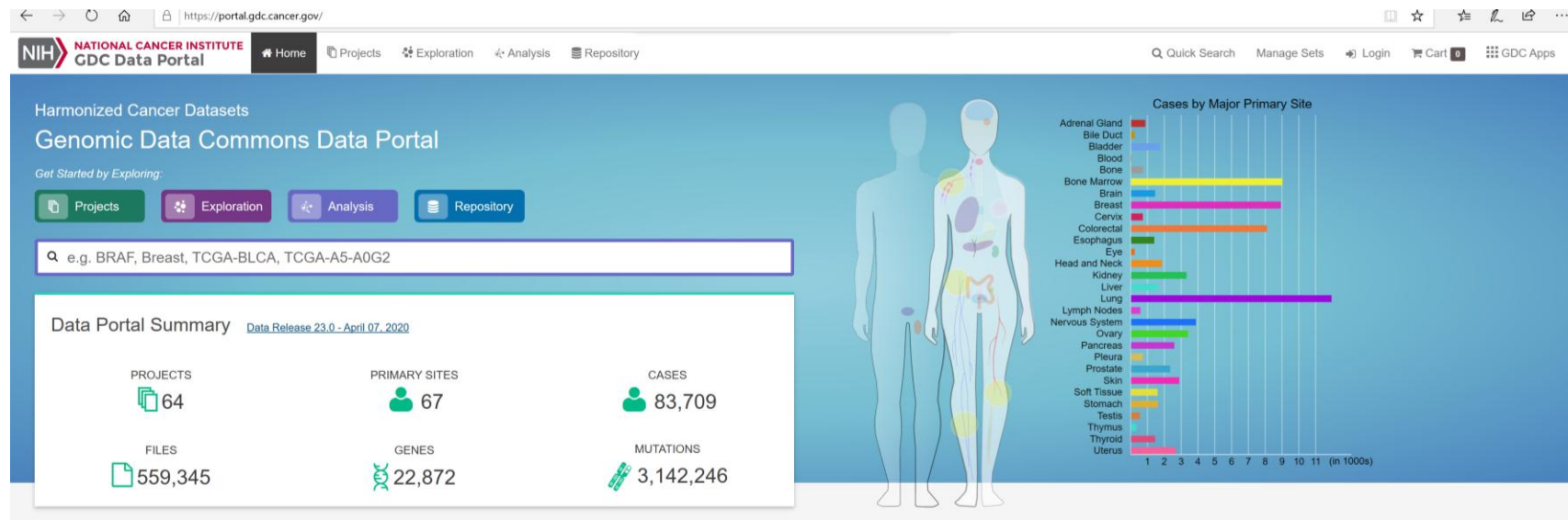


PCA

«Genes mirror geography within Europe» Novembre et al, *Nature*, 2008

Hands-on: gene expression data from several cancer types

- <https://github.com/BoevaLab/Teaching>
- Input: The Cancer Genome Atlas (TCGA) mRNA expression data



Jupyter example: gene expression data from several cancer types

- **Input:** The Cancer Genome Atlas (TCGA) mRNA expression data

- 13 cancer types
- Total number of samples: 4,540
- Number of genes with profiled expression: 20,528

```
In [31]: import numpy as np
import pandas as pd

#read precalculated csv table "TCGA-cancer-DF.zip"
df_noNA=pd.read_csv("https://cloud.inf.ethz.ch/s/c64nqrB4SJKaHDF/download", index_col=0,compression="zip")
df_noNA.head()
```

Out[31]:

	?	?	?	?	?	?	?	?	?	?	...	ZYG11B	ZYX	ZZEF1	ZZZ3
	100130426	100133144	100134869	10357	10431	136542	155060	26823	280660	317712	...				
TCGA-61-1910-01	-0.04126	4.7860	5.040	-6.683	-0.5449	-0.000557	3.3340	2.58300	-0.0893	-0.001292	...	-0.1832	1.6160	0.4057	0.58240
TCGA-61-1728-01	-0.04126	2.0640	2.149	-2.271	-0.6134	-0.000557	0.3176	-0.54440	-0.0893	-0.001292	...	-0.3968	1.3070	-1.4040	0.28320
TCGA-09-1666-01	-0.04126	1.7720	2.805	-2.959	0.4053	-0.000557	0.5211	1.16200	-0.0893	-0.001292	...	0.6730	1.3920	-0.9993	0.67380
TCGA-24-1469-01	-0.04126	2.1020	2.402	-1.069	0.4796	-0.000557	0.1432	-0.08483	0.7185	-0.001292	...	-0.5974	0.7653	-0.4181	-0.07337
TCGA-61-1917-01	-0.04126	0.5883	2.380	-1.513	0.3045	-0.000557	-0.6307	-0.54440	-0.0893	-0.001292	...	-0.4541	1.2990	-1.1600	0.03223

5 rows x 20534 columns

4,540 samples

20,528 genes

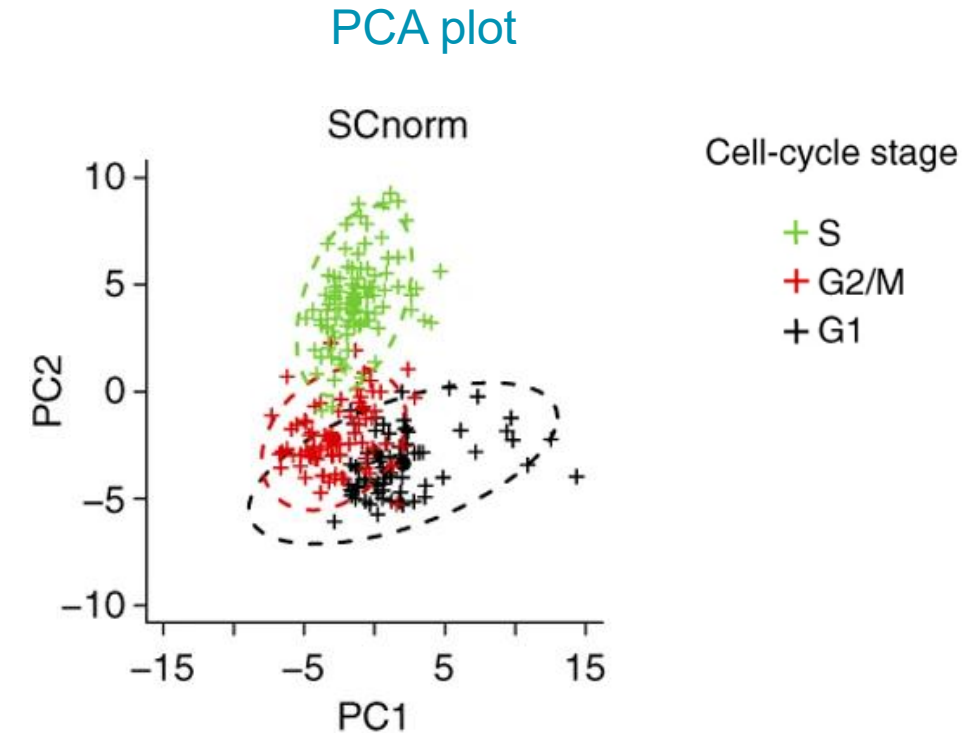
Potential question: How different are transcriptomic profiles of different cancer samples?

Visualization/dimensionality reduction of scRNA-seq

Visualization is one of the most important steps in any data analysis

Common visualization strategies for scRNA-seq data:

- **PCA** (principal component analysis)
- **t-SNE** (t-distributed stochastic neighbor embedding)
- **UMAP** (uniform manifold approximation and projection)

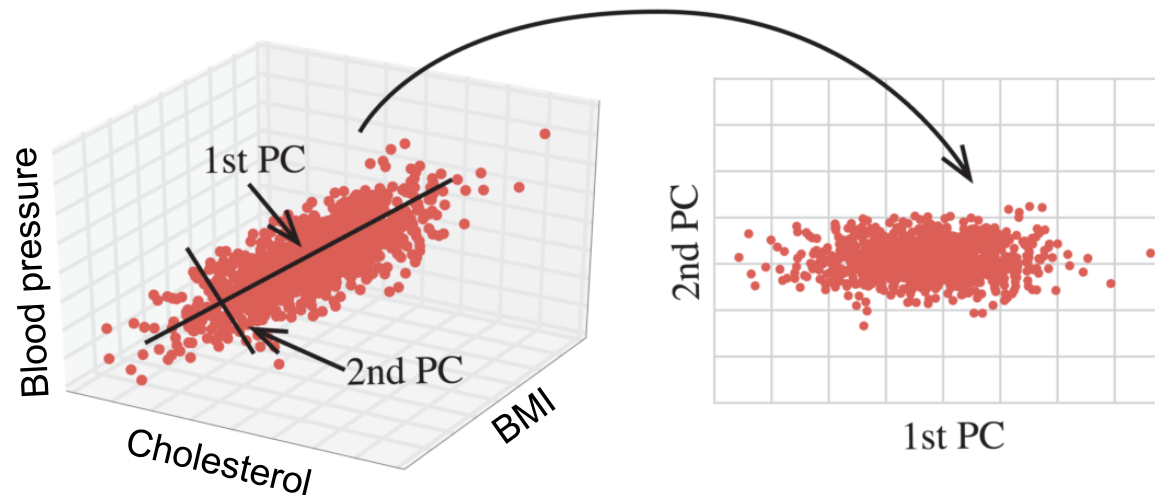


Linear transformation of the data to inspect presence of technical and/or biological biases

Principal component analysis (PCA)

PCA: transforms a set of possibly correlated variables into a **smaller set of uncorrelated variables** called **principal components**.

Each **principal component** is a **linear combination of the original variables** and is constructed to capture the **maximum possible variance in the data**

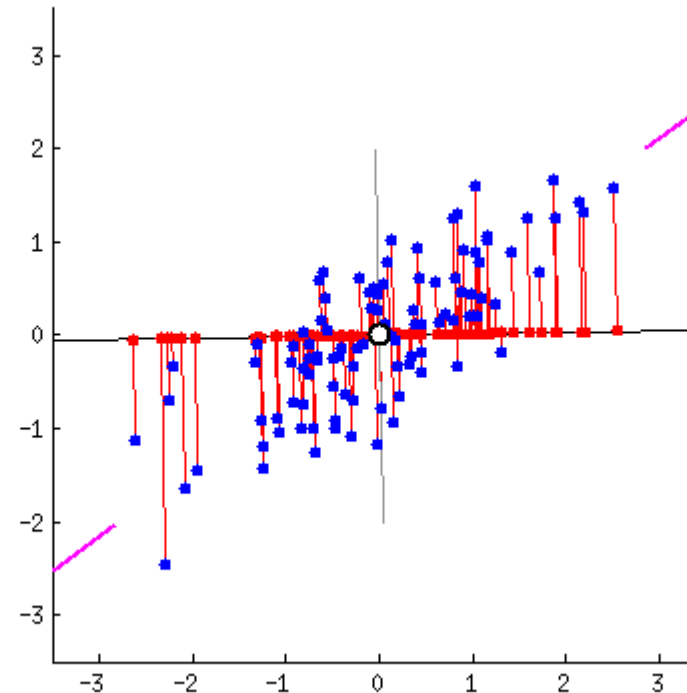


Principal component analysis (PCA)

The **first** principal component:
the line that captures the largest amount of
variance

The **second** principal component captures the
next highest variance, subject to being
uncorrelated (orthogonal) with the first.

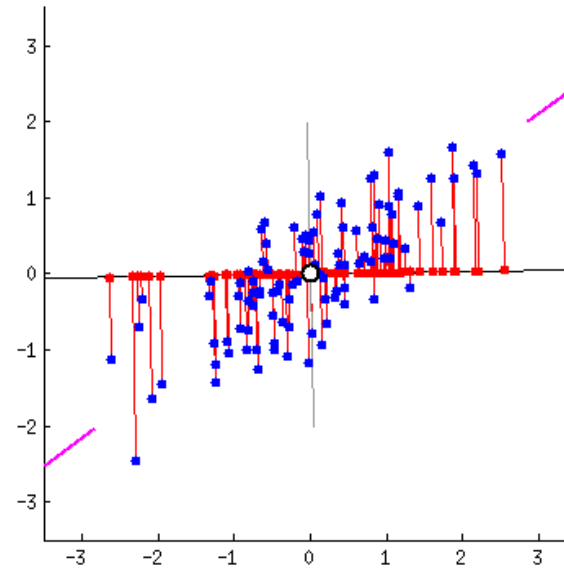
And so on.



From <https://builtin.com/data-science/step-step-explanation-principal-component-analysis>

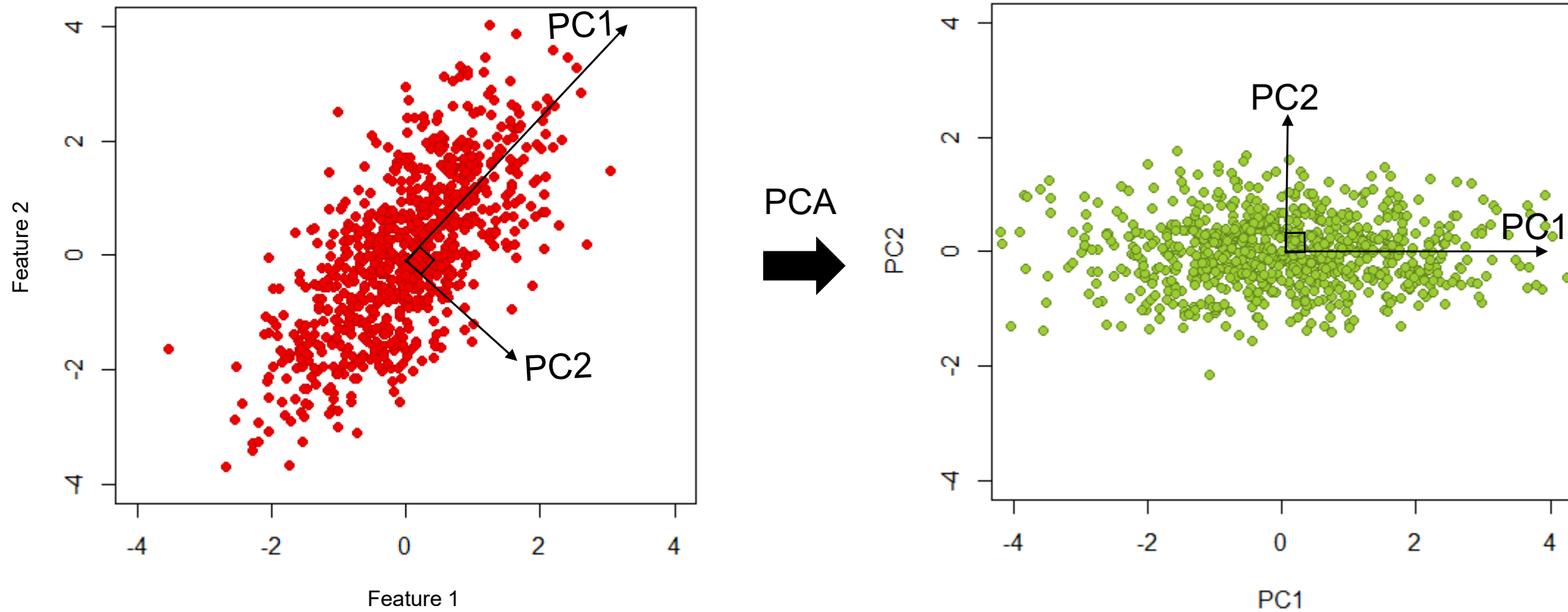
Principal component analysis (PCA)

- Importantly, PCA also minimizes the **projection error** — the difference between the original data and its lower-dimensional representation — ensuring that the reduced data retains as much of the original structure as possible.



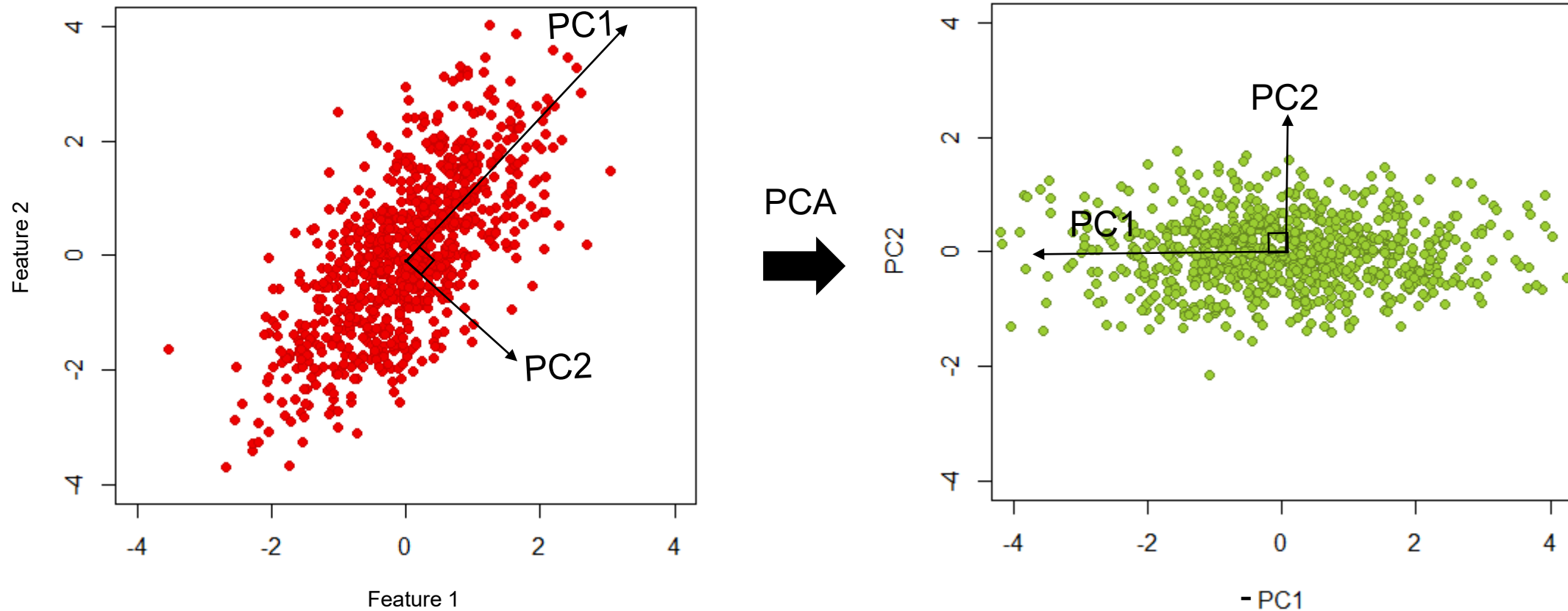
Principal component analysis (PCA)

PCA: an orthogonal linear transformation



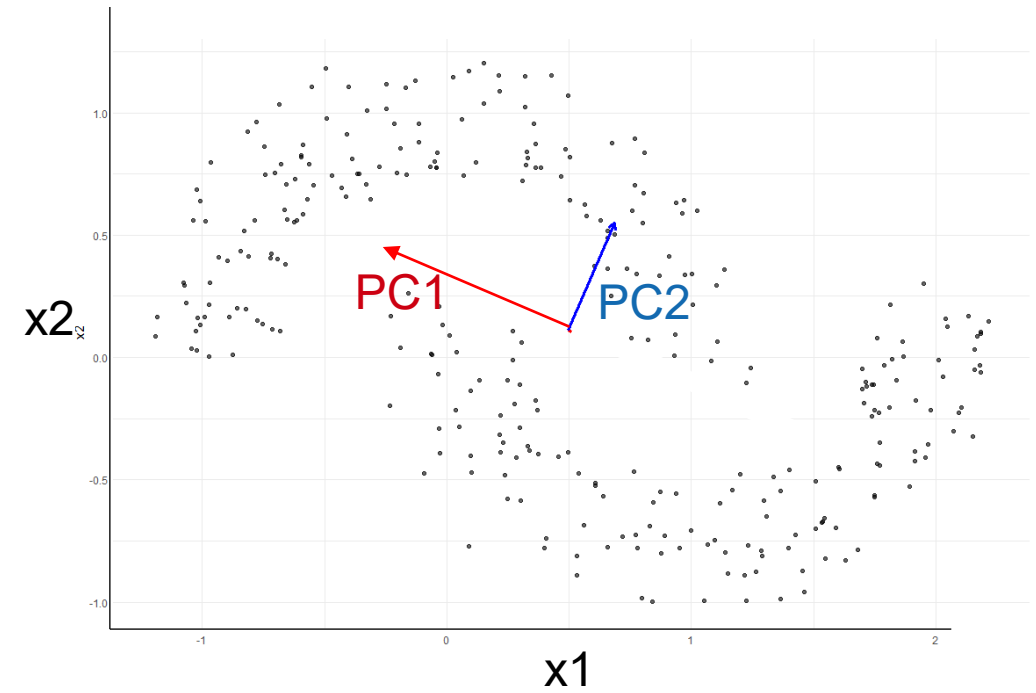
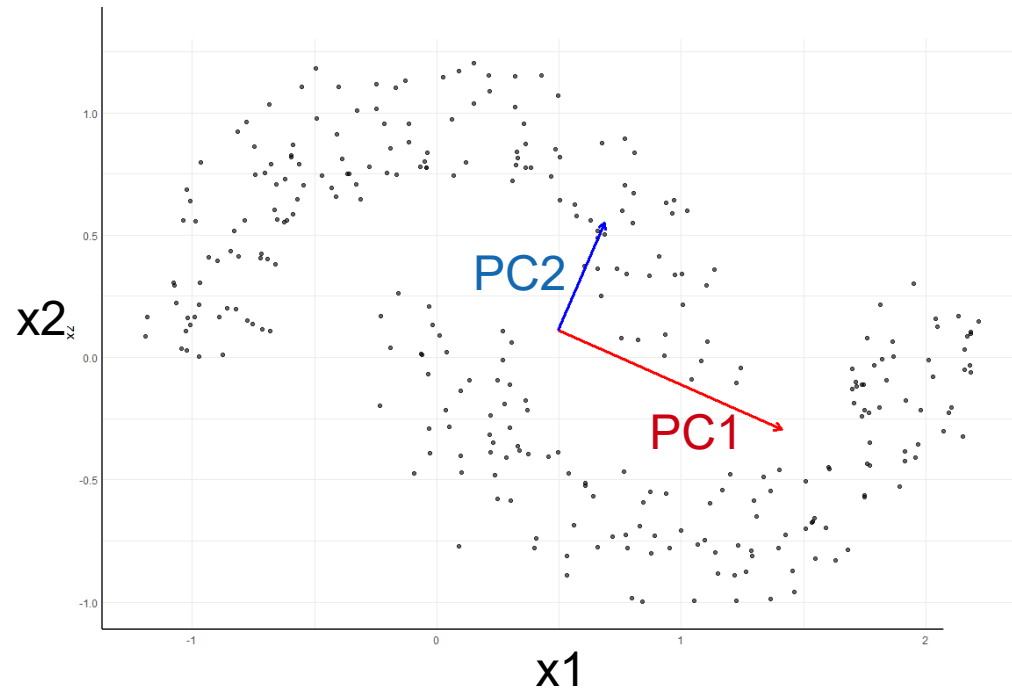
Direction along a PC is chosen arbitrary

PCA: an orthogonal linear transformation



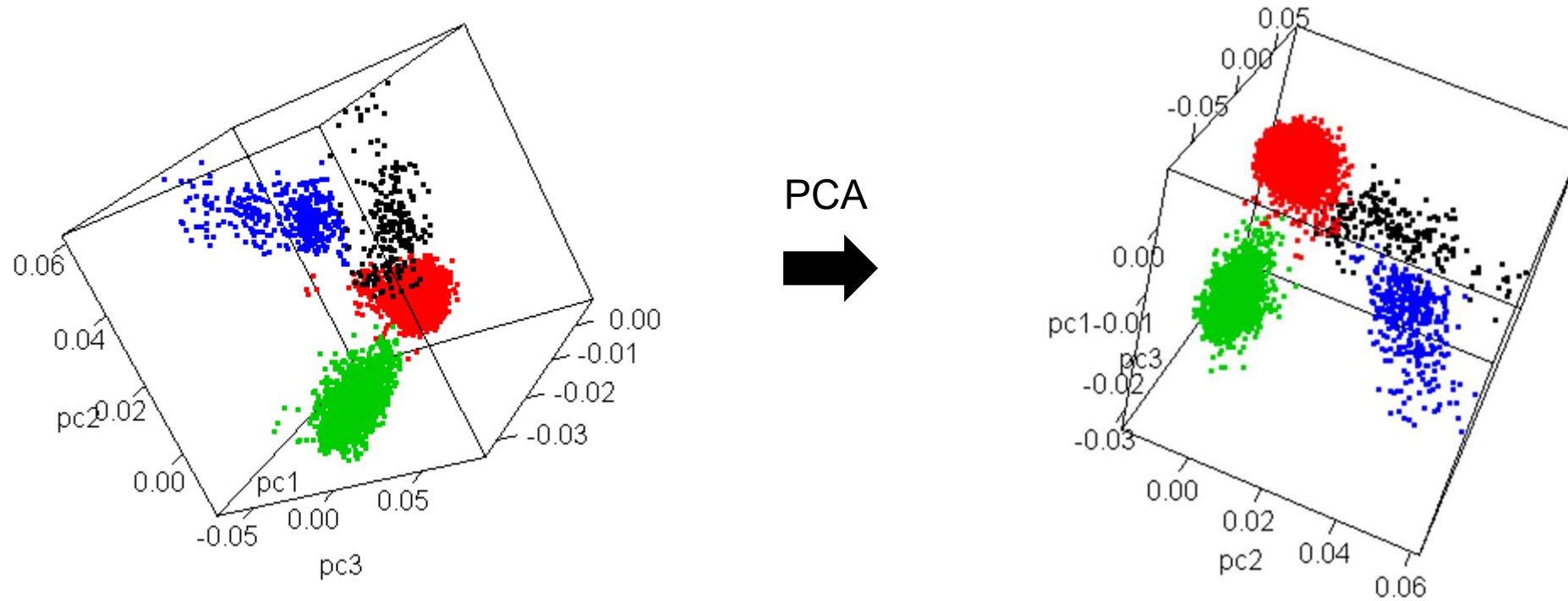
Direction along a PC is chosen arbitrary

Two possible results:



Principal component analysis (PCA)

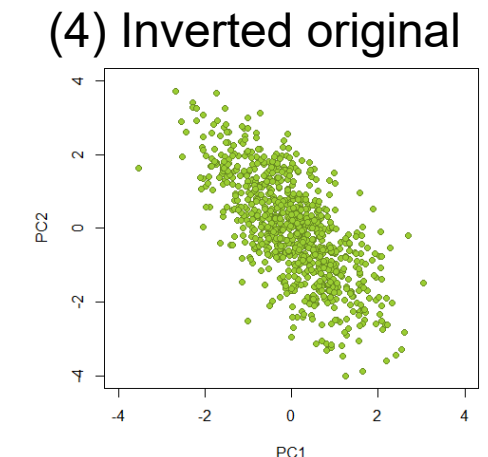
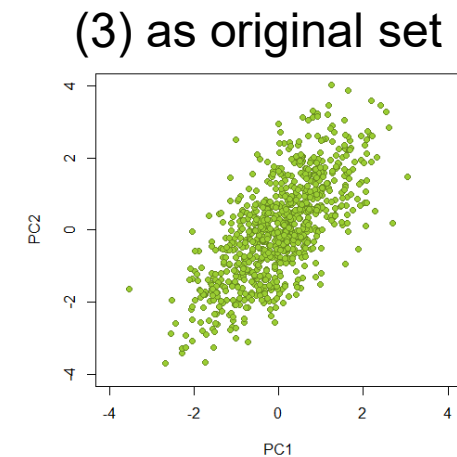
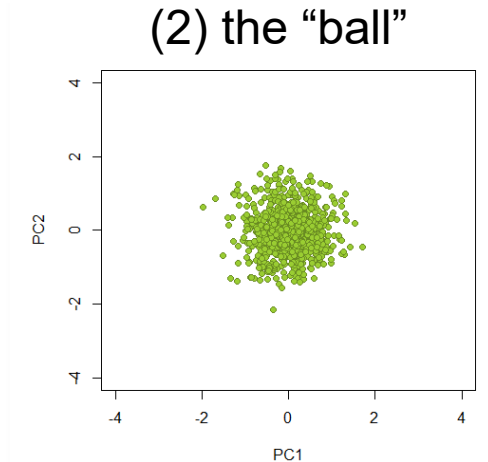
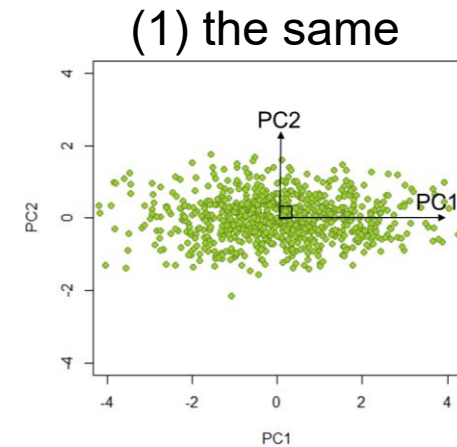
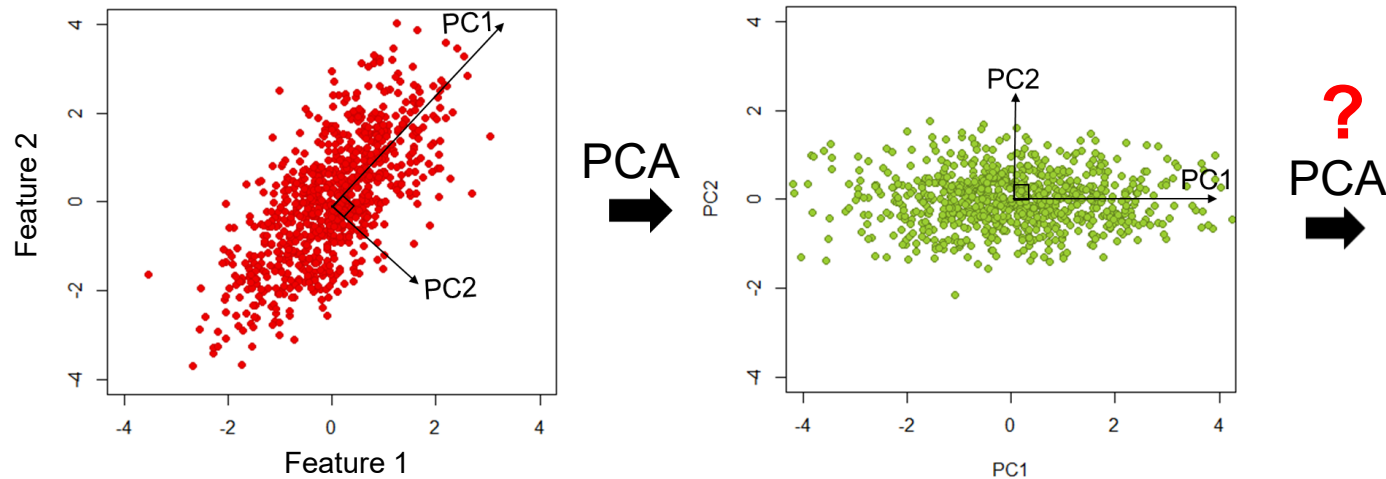
PCA: an orthogonal linear transformation



Principal component analysis (PCA)

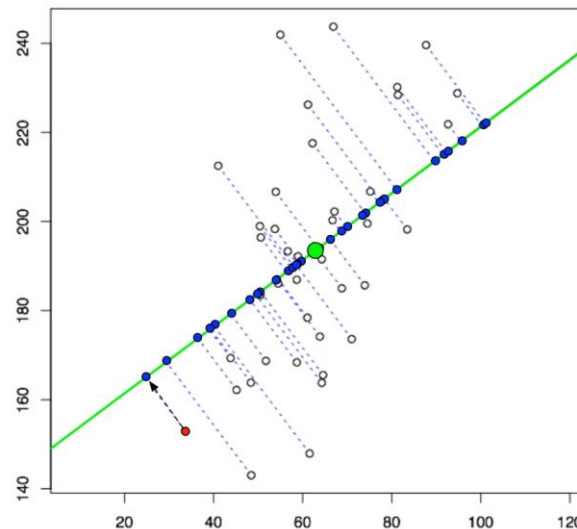
Generally, one standardizes the data along each dimension before applying PCA.

Q: How will look the PCA transformation of a PCA?



Principal component analysis (PCA)

- **PCA**: an **orthogonal linear transformation** that transforms the data to a new coordinate system such that the greatest variance by some scalar projection of the data comes to lie on the first coordinate (called the first principal component), the second greatest variance on the second coordinate, and so on.

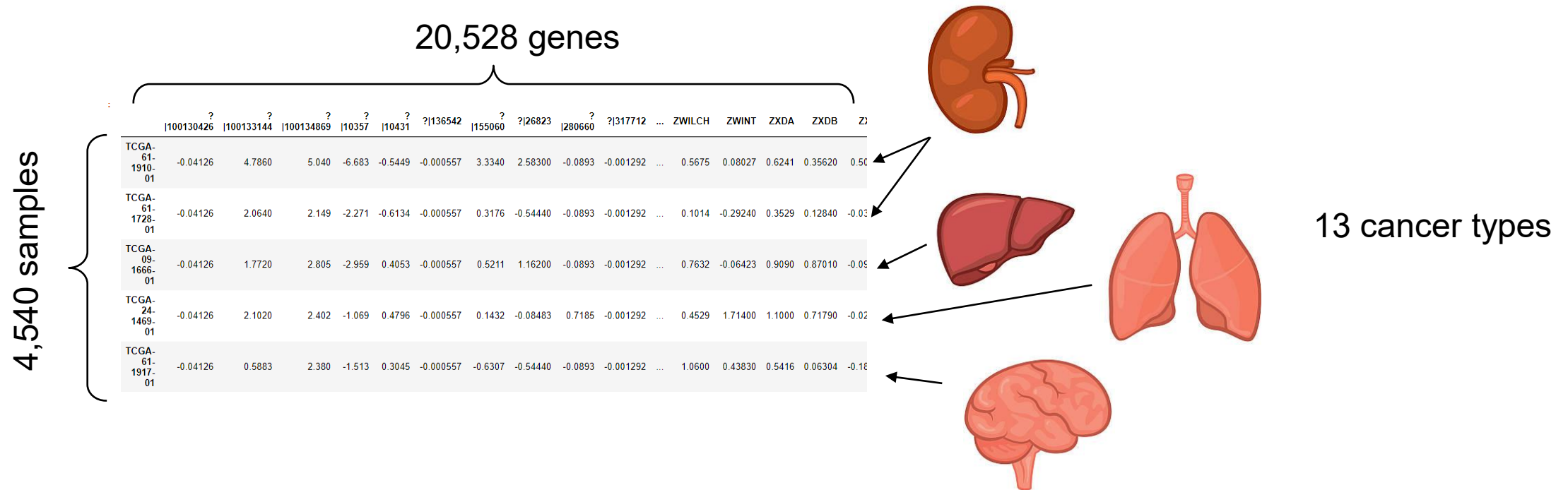


From <https://liorpachter.wordpress.com/2014/05/26/what-is-principal-component-analysis/>

Example: check Jupiter Notebook

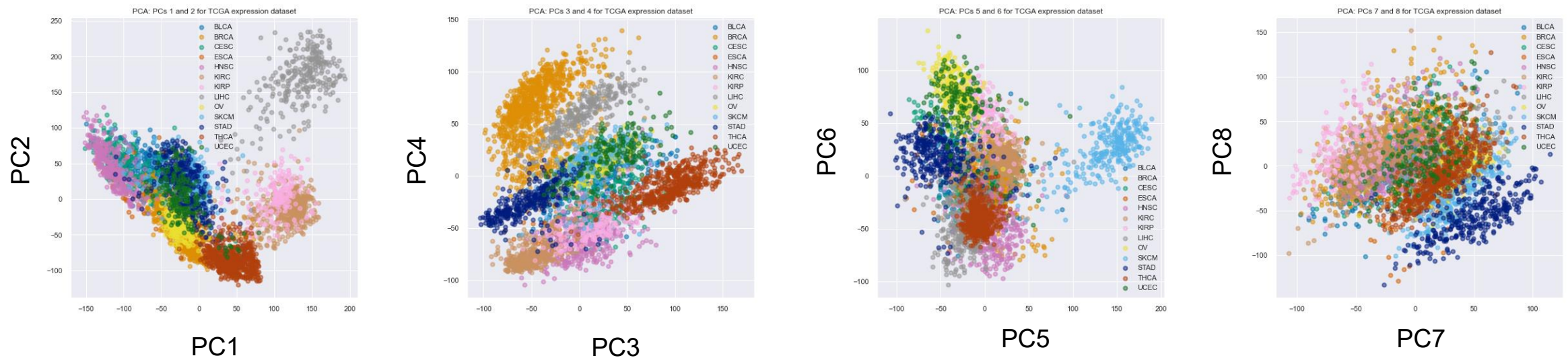
<https://student-jupyter.inf.ethz.ch/>

Input: The Cancer Genome Atlas (TCGA) mRNA expression data



Our task: find the best 2-dimensional visualization for a publication of these data

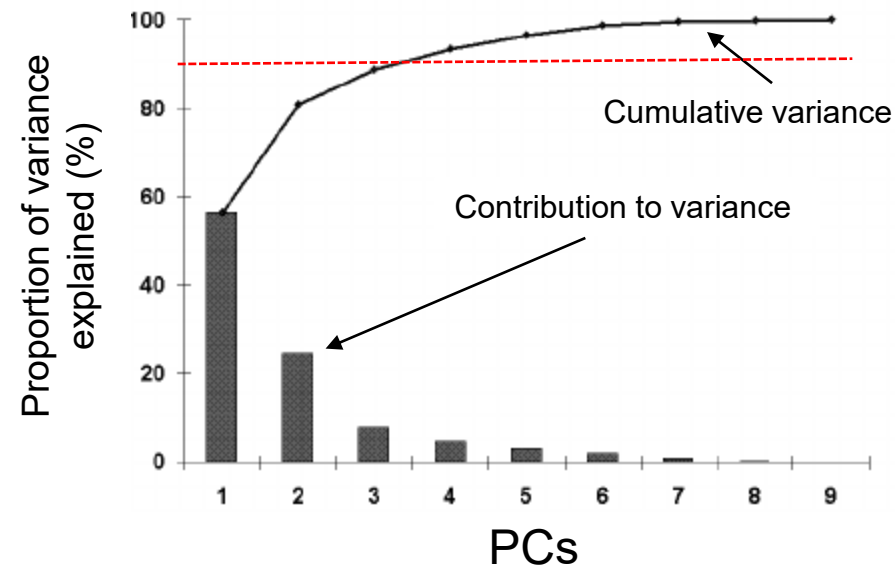
Example from the Jupiter Notebook: PCA on cancer bulk RNA-seq data



Cancer types seem to form distinct groups, but we cannot see them with only first two PCs

Principal component analysis (PCA)

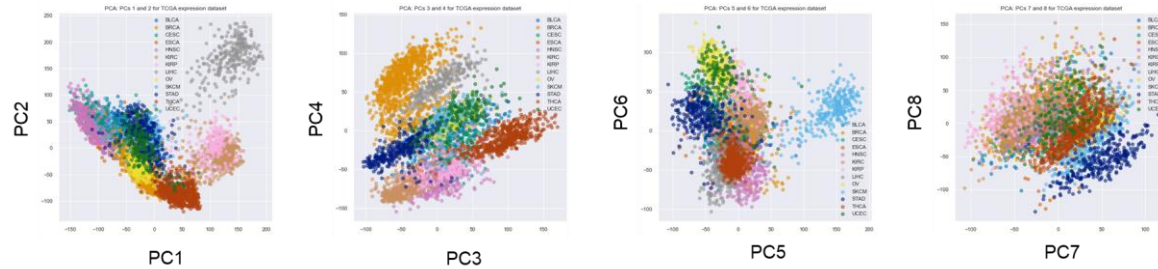
- PCA is a deterministic method, the only parameter one can choose is k for how many principal components to keep for further analysis.
- Choosing k based on the proportion of the variance explained.



$Var_explained_k = \frac{\lambda_k}{\sum_i \lambda_i}$, where λ_k is k^{th} eigenvalue.

Example from the Jupiter Notebook: PCA on cancer bulk RNA-seq data

- Variance explained by the first components

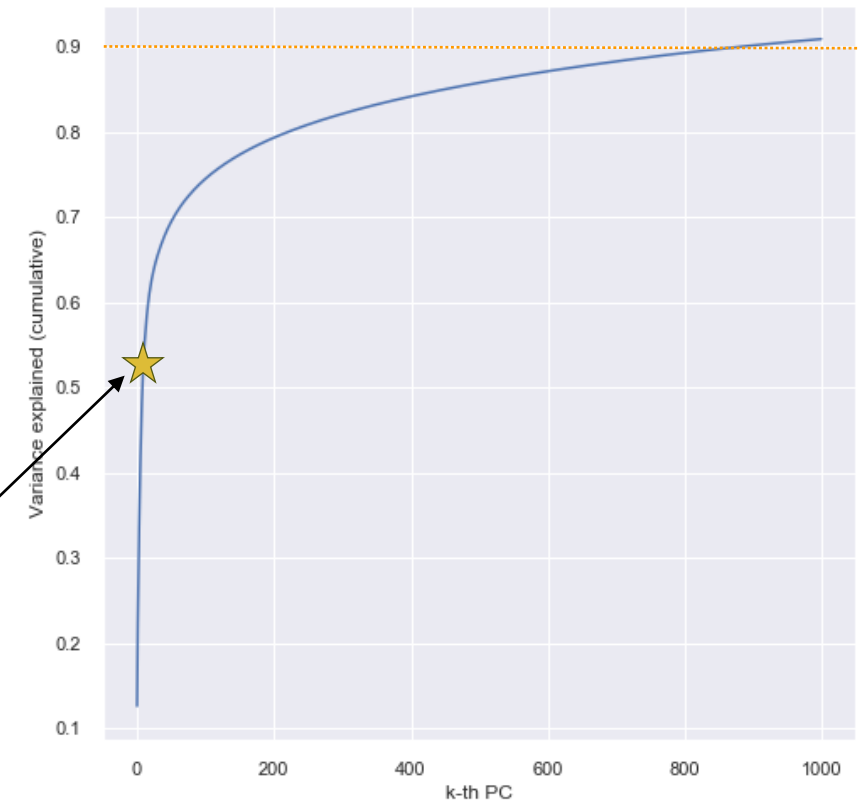


Variance explained by the first 10 PCs:

0.13, 0.09, 0.06, 0.05, 0.05, 0.04, 0.03, 0.03, 0.02, 0.02

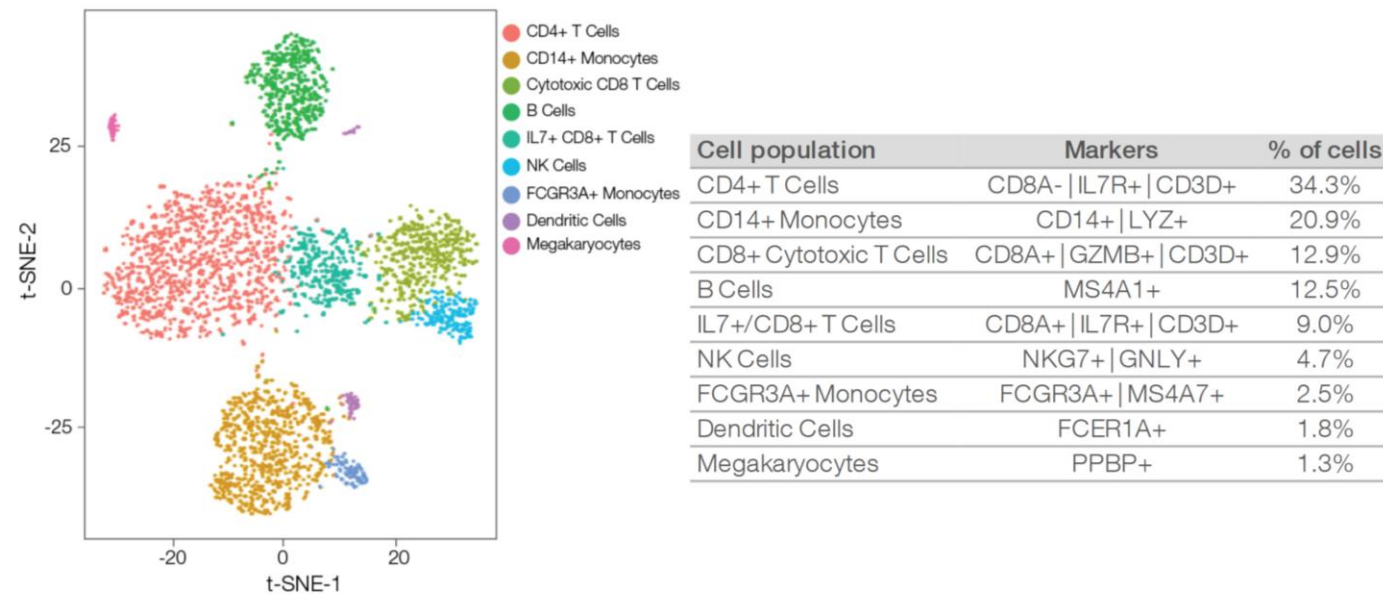
Cumulative variance explained by the first 10 PCs:

0.13, 0.22, 0.28, 0.36, 0.38, 0.42, 0.45, 0.48, 0.51, 0.52



t-distributed Stochastic Neighbor Embedding (tSNE)

- tSNE: nonlinear dimensionality reduction technique, converts similarities between data points to joint probabilities and tries to minimize the Kullback-Leibler divergence between the joint probabilities of the low-dimensional embedding and the high-dimensional data.



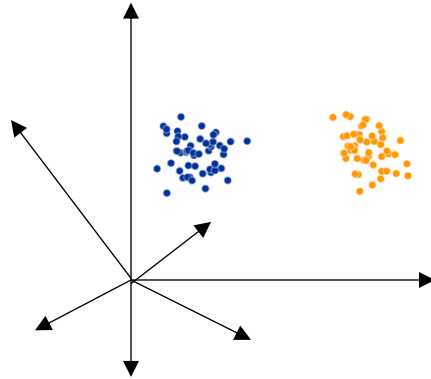
- t-SNE has a cost function that is not convex, i.e., with different initializations we can get different results.

[First introduced by van der Maaten & Hinton paper from 2008.](#)

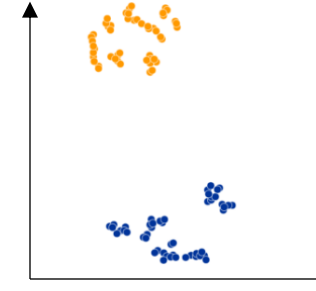
Top level idea of t-SNE

Q: Will t-sne output be identical to the input when transforming 2D data into 2D?

High-dimensional space

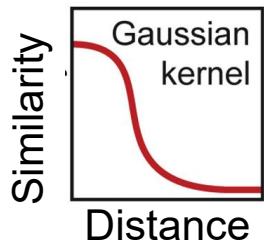


Two-dimensional space



Distance matrix
(distance between each
two vectors x_i and x_j)

	x_1	x_2	...	x_N
x_1	0	0.1	...	0.9
x_2	0.1	0	...	0.8
...	...	...	...	...
x_N	0.9	0.8	...	0



Similarity matrix
(similarity between
 x_i and x_j)

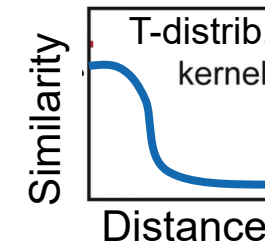
	x_1	x_2	...	x_N
x_1	1	0.7	...	0.1
x_2	0.7	1	...	0.2
...	...	...	...	...
x_N	0.1	0.2	...	1

Match
using
gradient
descent



Similarity matrix
(similarity between
 x_i and x_j)

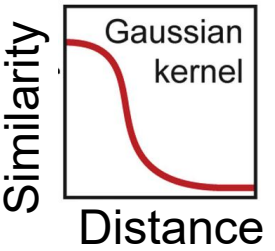
	x_1	x_2	...	x_N
x_1	1	0.6	...	0.2
x_2	0.6	1	...	0.4
...	...	...	...	...
x_N	0.2	0.3	...	1



Distance matrix
(distance between each
two vectors x_i and x_j)

	x_1	x_2	...	x_N
x_1	0	0.2	...	0.9
x_2	0.2	0	...	0.7
...	...	...	...	...
x_N	0.9	0.7	...	0

Zoom on the t-SNE method



t-SNE (2008)

$p_{i|j}$ – the similarity of datapoint x_j to datapoint x_i : the conditional probability, $p_{j|i}$, that x_i would pick x_j as its neighbor if neighbors were picked in proportion to their probability density under a Gaussian centered at x_i .

q_{ij} – joint probabilities of the low-dimensional embedding – follow a Student t-distribution with one degree of freedom (which is the same as a Cauchy distribution)

C – the cost function to minimize: the sum of Kullback-Leibler divergences

“similarity” between two datapoints i and j in the high-dimensional space

Related to the user-specified Perplexity parameter

$$p_{j|i} = \frac{\exp(-\|x_i - x_j\|^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2 / 2\sigma_i^2)}$$

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n}$$

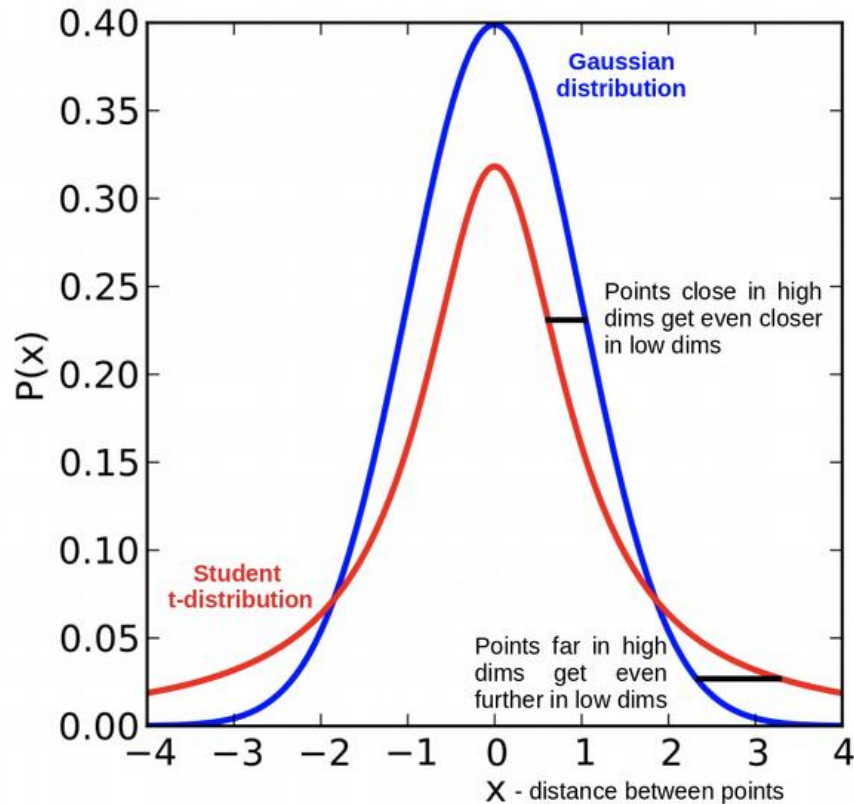
$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq l} (1 + \|y_k - y_l\|^2)^{-1}}$$

“similarity” between two datapoints i and j in the 2D space

$$C = KL(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

Zoom on the t-SNE method

t-SNE (2008)



“similarity” between two datapoints i and j in the high-dimensional space

Related to the user-specified Perplexity parameter

$$p_{j|i} = \frac{\exp(-\|x_i - x_j\|^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2 / 2\sigma_i^2)}$$

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n}$$

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq l} (1 + \|y_k - y_l\|^2)^{-1}}$$

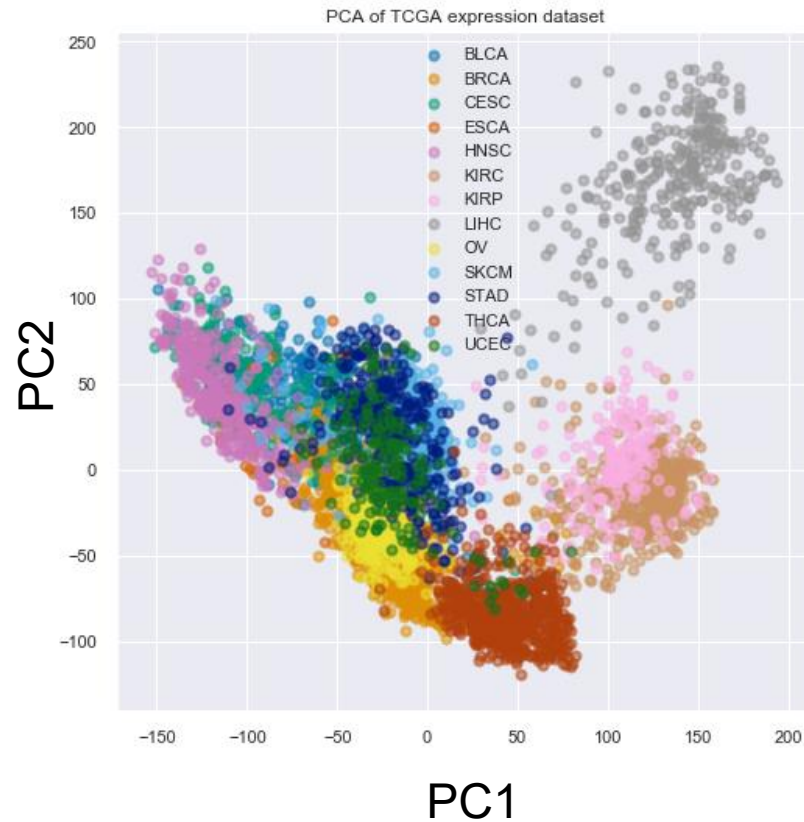
“similarity” between two datapoints i and j in the 2D space

$$C = KL(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

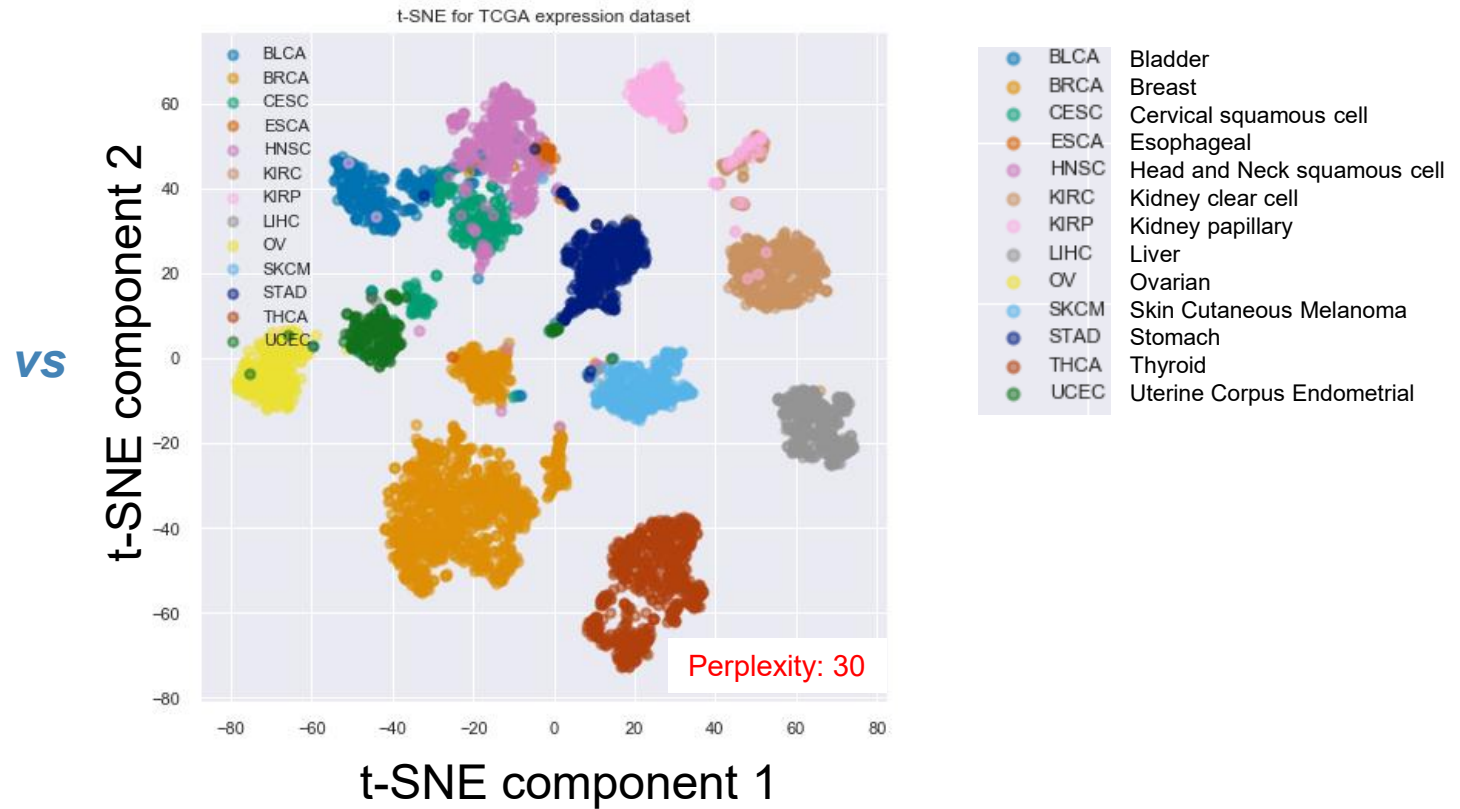
t-distr. solves “crowding problem”

Example from the Jupiter Notebook: t-SNE

PCA



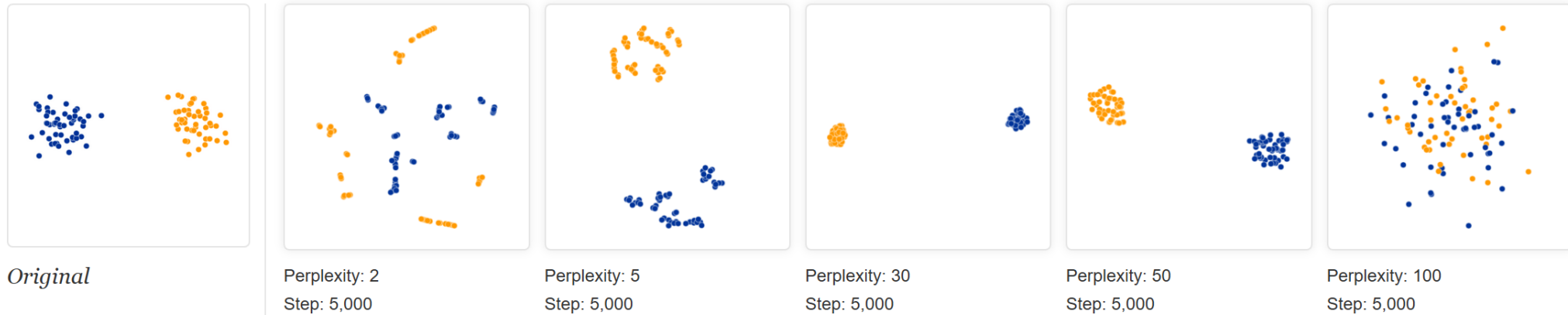
t-SNE



t-distributed Stochastic Neighbor Embedding (tSNE)

tSNE method's the most important parameter:

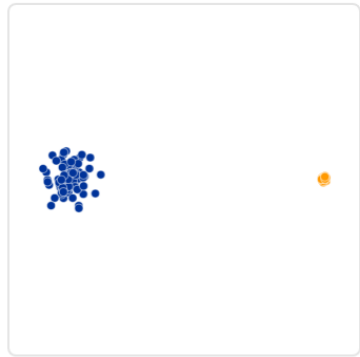
- **Perplexity**, scikit-learn recommended range: [5, 50], default: 30



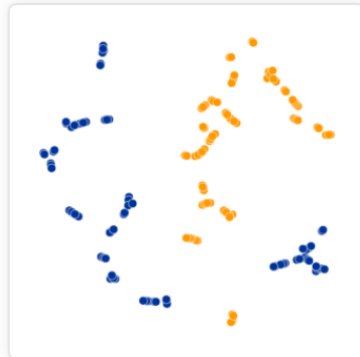
t-distributed Stochastic Neighbor Embedding (tSNE)

tSNE method's particularities:

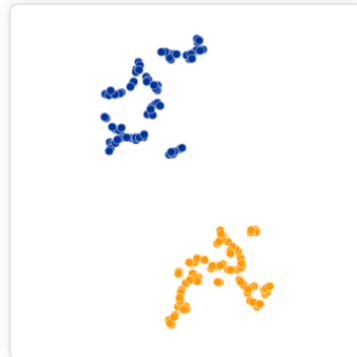
- Cluster sizes in a t-SNE plot mean nothing



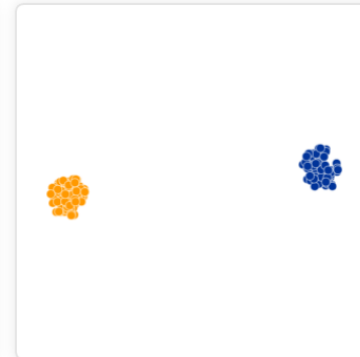
Original



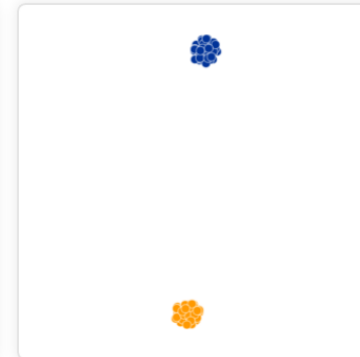
Perplexity: 2
Step: 5,000



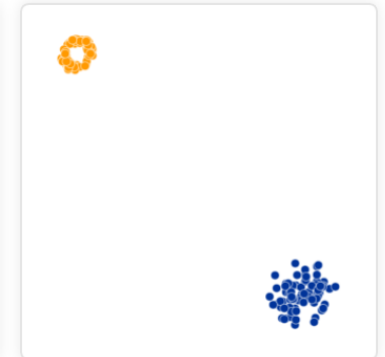
Perplexity: 5
Step: 5,000



Perplexity: 30
Step: 5,000



Perplexity: 50
Step: 5,000



Perplexity: 100
Step: 5,000

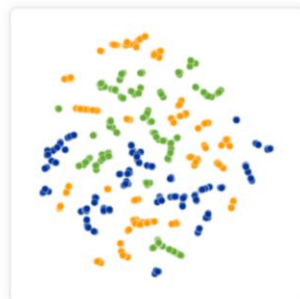
t-distributed Stochastic Neighbor Embedding (tSNE)

tSNE method's particularities:

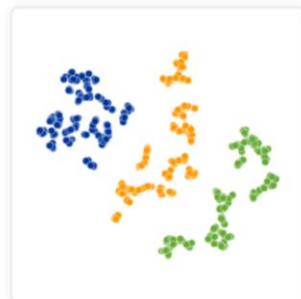
- Cluster sizes in a t-SNE plot mean nothing
- Distances between clusters might not mean anything



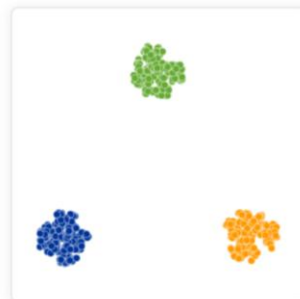
Original



Perplexity: 2
Step: 5,000



Perplexity: 5
Step: 5,000



Perplexity: 30
Step: 5,000



Perplexity: 50
Step: 5,000



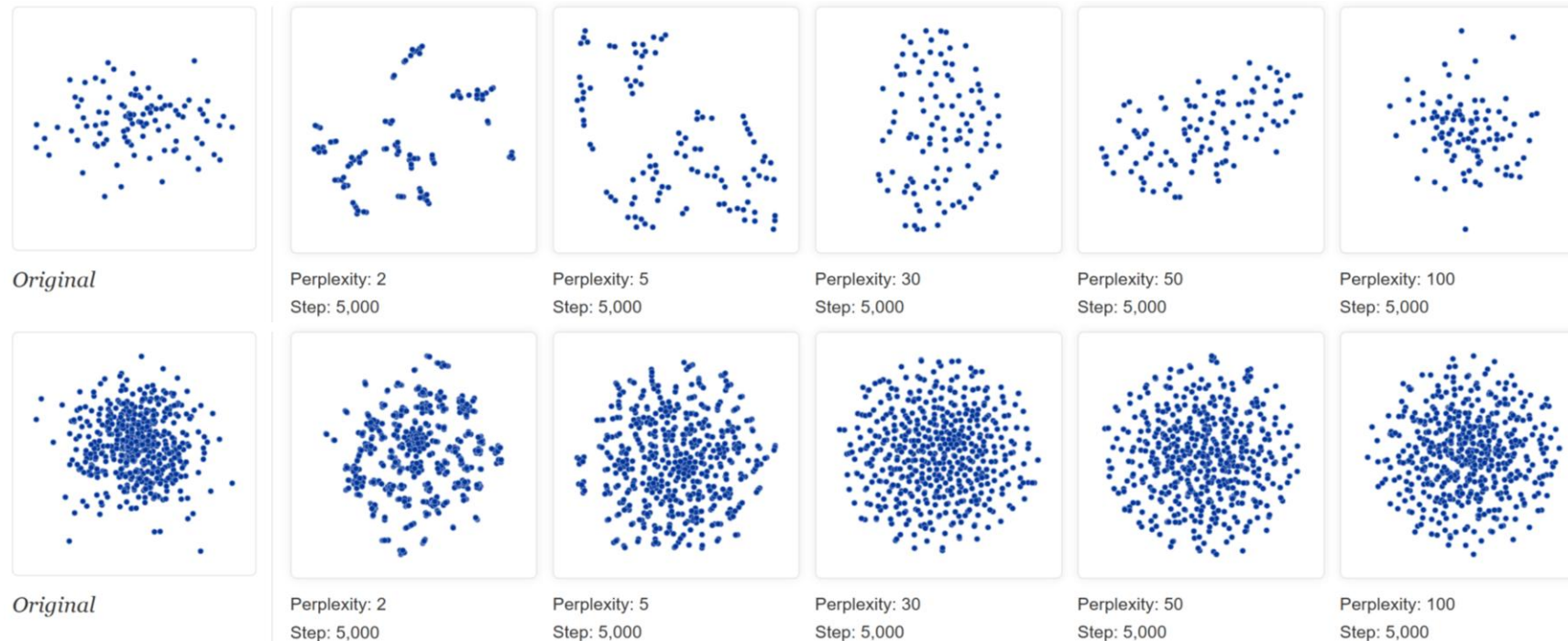
Perplexity: 100
Step: 5,000

200 points/cluster

t-distributed Stochastic Neighbor Embedding (tSNE)

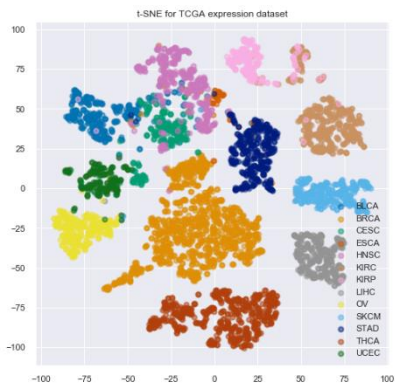
tSNE method's particularities:

- Cluster sizes in a t-SNE plot mean nothing
- Distances between clusters might not mean anything
- Random noise does not always look random, and sometimes one can see some shapes

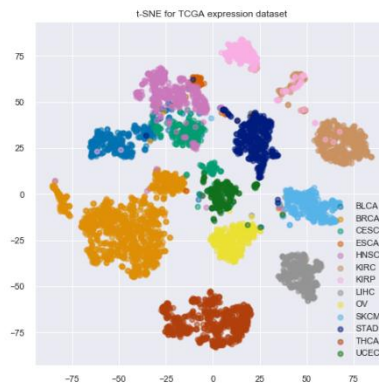


What would you choose as the best value of perplexity in this example?

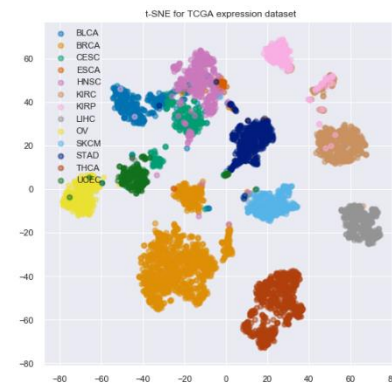
● BLCA	Bladder
● BRCA	Breast
● CESC	Cervical squamous cell
● ESCA	Esophageal
● HNSC	Head and Neck squamous cell
● KIRC	Kidney clear cell
● KIRP	Kidney papillary
● LIHC	Liver
● OV	Ovarian
● SKCM	Skin Cutaneous Melanoma
● STAD	Stomach
● THCA	Thyroid
● UCEC	Uterine Corpus Endometrial



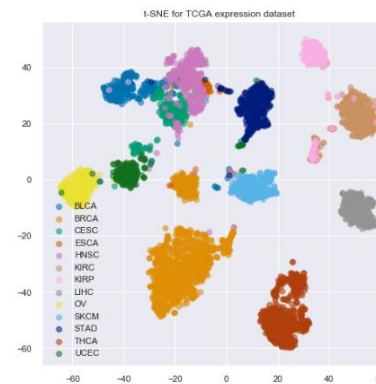
Perplexity: 10



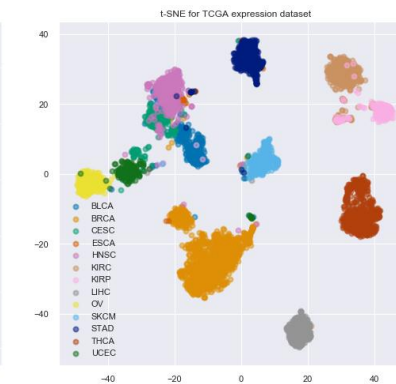
20



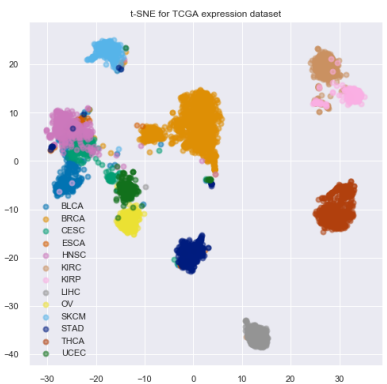
30



50

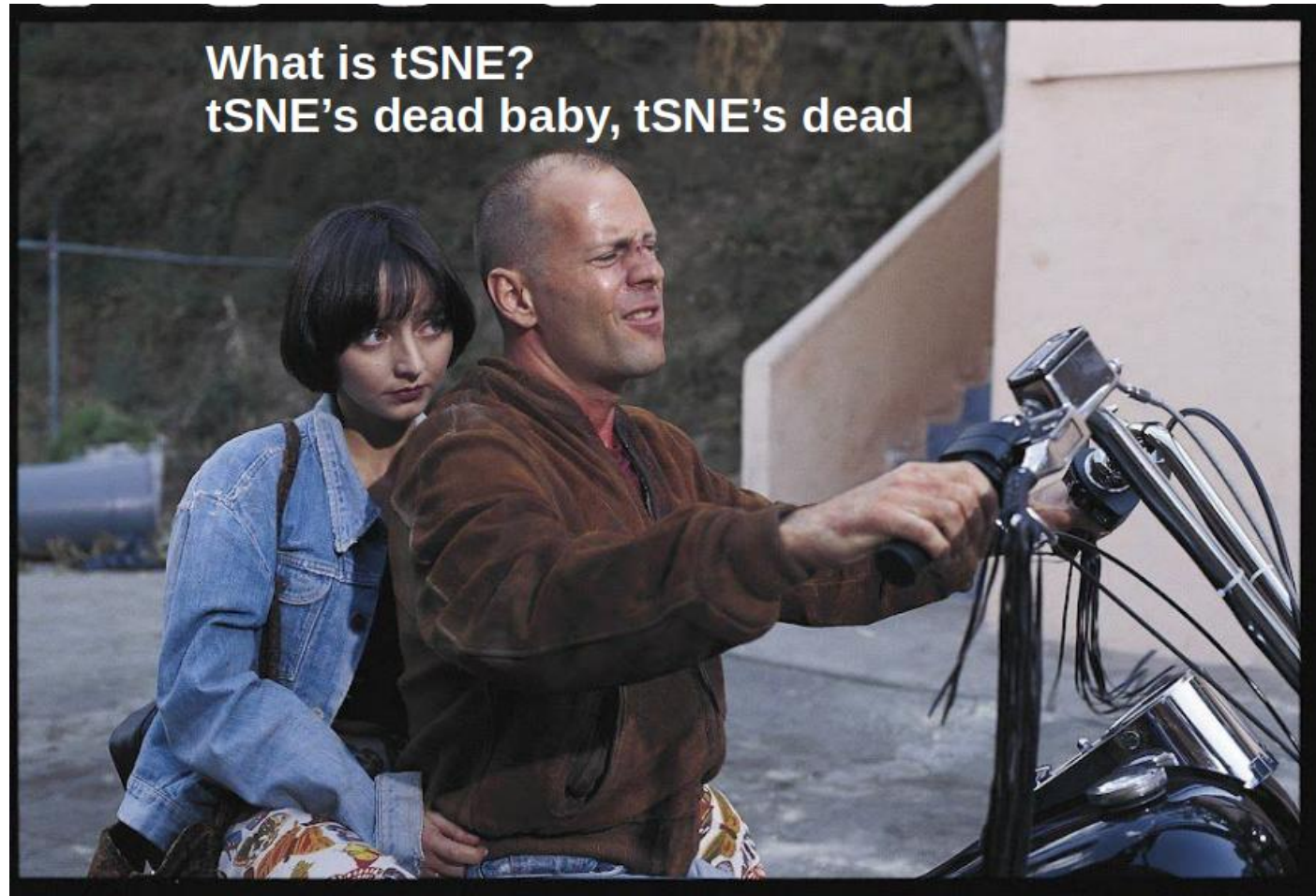


100



200

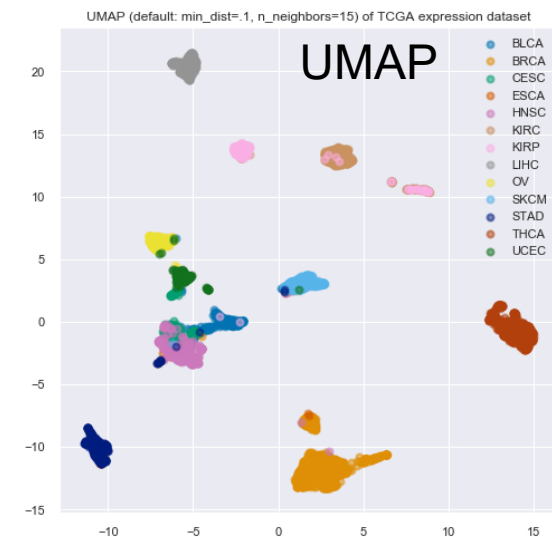
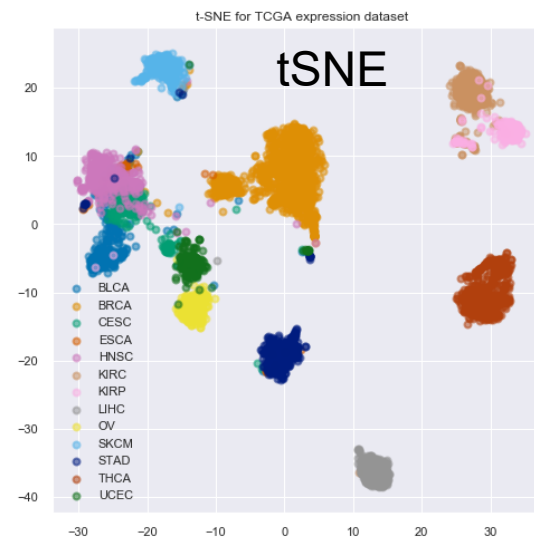
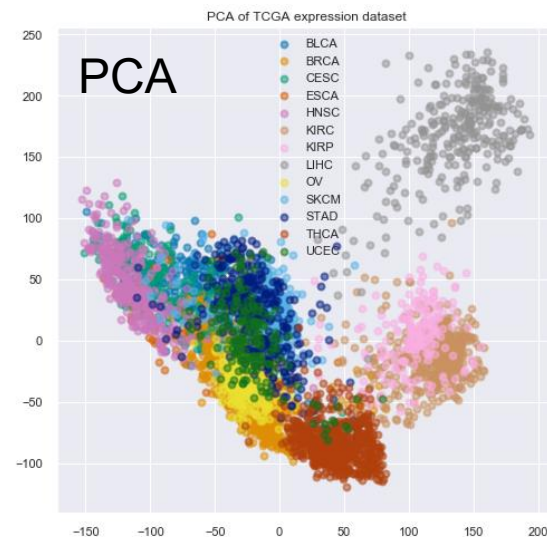
Uniform Manifold Approximation and Projection (UMAP)



Uniform Manifold Approximation and Projection (UMAP)

UMAP: nonlinear dimensionality reduction technique. Idea is similar to tSNE, but

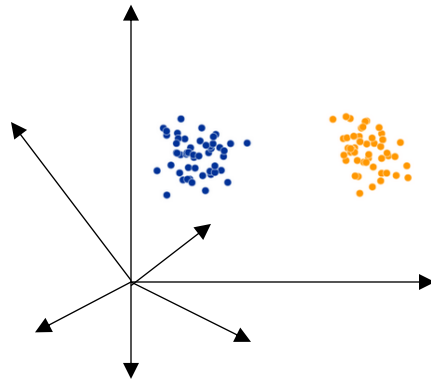
- Much faster
- Not limited to the first 2-3 dimensions
- Uses binary cross-entropy as a cost function instead of the KL-divergence
- Better preserves global structure
- Uses the number of nearest neighbors and min-dist correction instead of perplexity



First introduced by [McInnes, L, Healy, J, ArXiv e-prints 1802.03426, 2018](https://arxiv.org/abs/1802.03426)

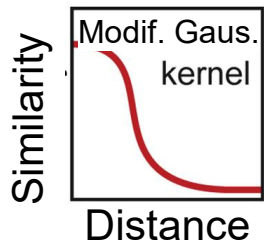
Top level idea of UMAP

High-dimensional space



Distance matrix
(distance between each
two vectors x_i and x_j)

	x_1	x_2	...	x_N
x_1	0	0.1	...	0.9
x_2	0.1	0	...	0.8
...	...	...	...	...
x_N	0.9	0.8	...	0



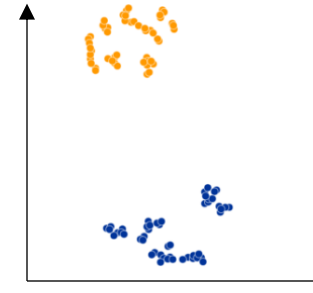
Similarity matrix
(similarity between
 x_i and x_j)

	x_1	x_2	...	x_N
x_1	1	0.7	...	0.1
x_2	0.7	1	...	0.2
...	...	...	...	...
x_N	0.1	0.2	...	1

Match
using
gradient
descent

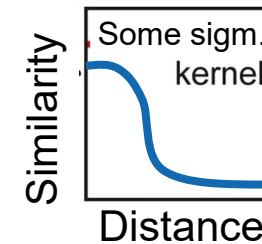


Two-dimensional space



Similarity matrix
(similarity between
 x_i and x_j)

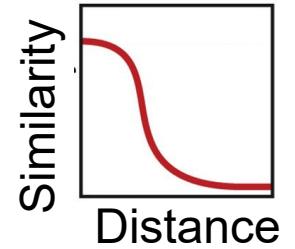
	x_1	x_2	...	x_N
x_1	1	0.6	...	0.2
x_2	0.6	1	...	0.4
...	...	...	...	...
x_N	0.2	0.3	...	1



Distance matrix
(distance between each
two vectors x_i and x_j)

	x_1	x_2	...	x_N
x_1	0	0.2	...	0.9
x_2	0.2	0	...	0.7
...	...	...	...	...
x_N	0.9	0.7	...	0

Uniform Manifold Approximation and Projection (UMAP)



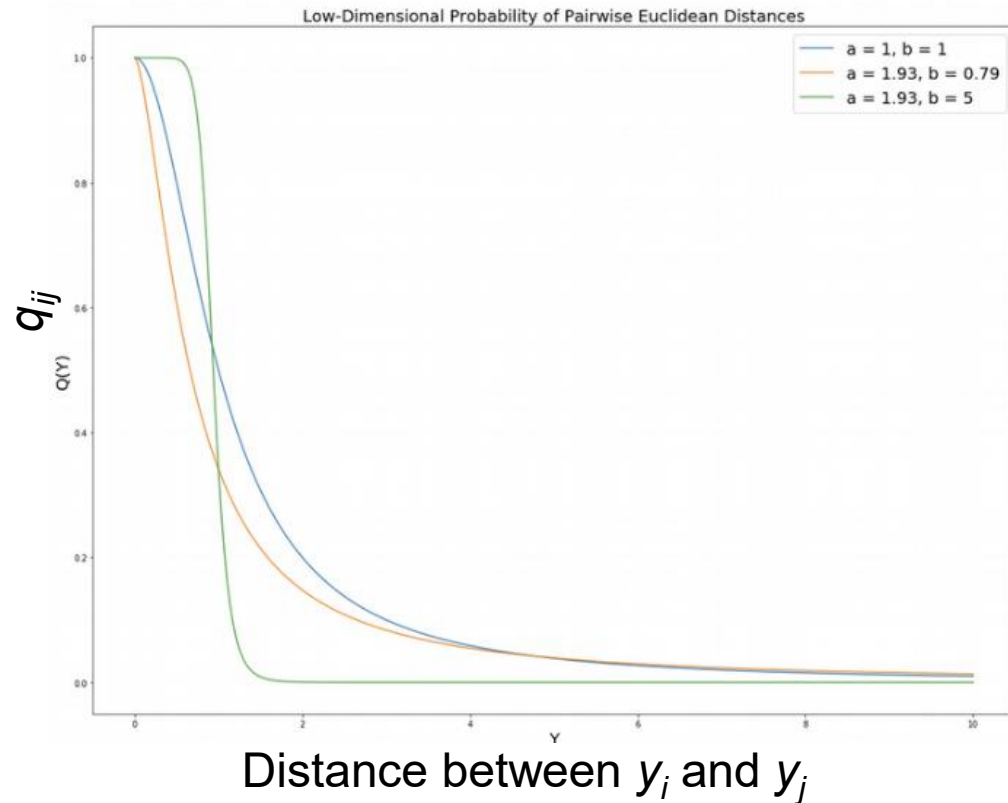
$$p_{i|j} = e^{-\frac{d(x_i, x_j) - \rho_i}{\sigma_i}}$$

Any distance can be plugged into UMAP, not only Euclidean distances.
 ρ_i is the distance from each i -th data point to its first nearest neighbor.

UMAP uses a slightly different symmetrization of the high-dimensional probability:

$$p_{ij} = p_{i|j} + p_{j|i} - p_{i|j}p_{j|i}$$

Uniform Manifold Approximation and Projection (UMAP)



$$p_{i|j} = e^{-\frac{d(x_i, x_j) - \rho_i}{\sigma_i}}$$

$$p_{ij} = p_{i|j} + p_{j|i} - p_{i|j}p_{j|i}$$

$$q_{ij} = \left(1 + a(y_i - y_j)^{2b}\right)^{-1}$$

q_{ij} is modeled as a family of curves, similar to student t-distribution.

Uniform Manifold Approximation and Projection (UMAP)

High-dimensional space

$$p_{i|j} = e^{-\frac{d(x_i, x_j) - \rho_i}{\sigma_i}}$$
$$p_{ij} = p_{i|j} + p_{j|i} - p_{i|j}p_{j|i}$$

Low-dimensional space

$$q_{ij} = \left(1 + a(y_i - y_j)^{2b}\right)^{-1}$$

$$CE(X, Y) = \sum_i \sum_j \left[p_{ij}(X) \log \left(\frac{p_{ij}(X)}{q_{ij}(Y)} \right) + (1 - p_{ij}(X)) \log \left(\frac{1 - p_{ij}(X)}{1 - q_{ij}(Y)} \right) \right]$$

Cross-entropy is minimized between the two “distributions”

Keep in mind the distributions q_{ij} are not normalized - is one of the steps that makes UMAP much faster than t-SNE.

Uniform Manifold Approximation and Projection (UMAP)

UMAP's the most important parameters:

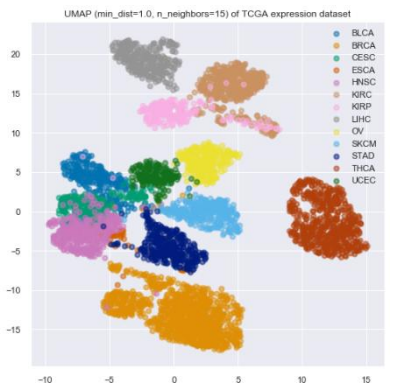
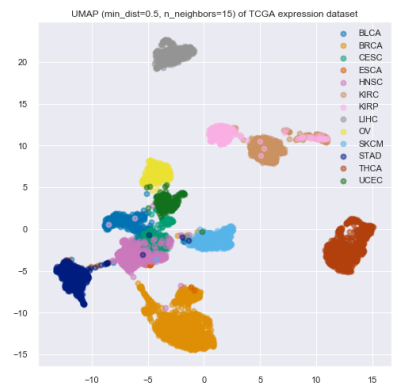
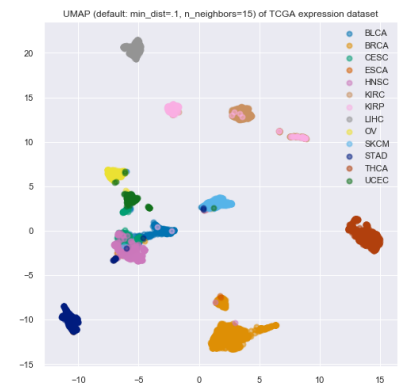
min_dist, recommended range: [0.1, 1], default: 0.1

- Controls how tightly the embedding is allowed compress points together. Larger values ensure embedded points are more evenly distributed, while smaller values allow the algorithm to optimise more accurately with regard to local structure. Sensible values are in the range 0.001 to 0.5, with 0.1 being a reasonable default.

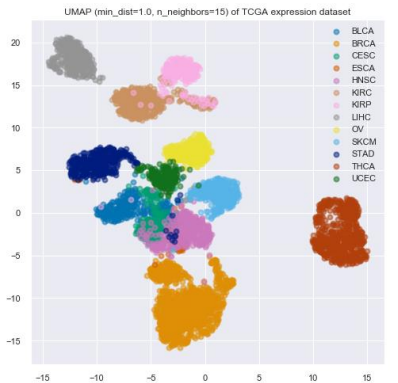
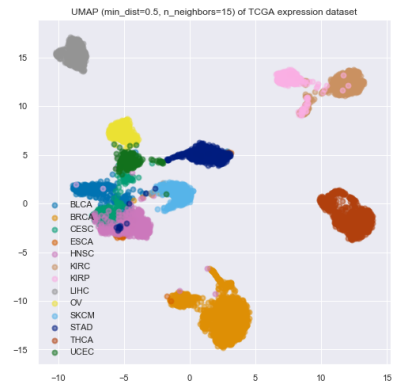
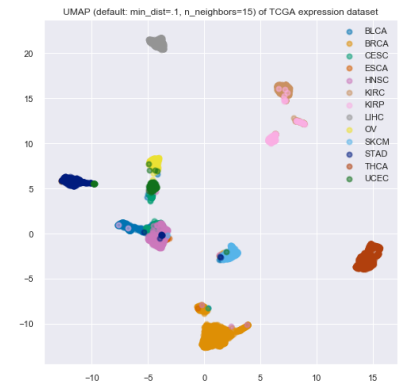
n_neighbors, recommended range: [2, 100], default: 15

- Determines the number of neighboring points used in local approximations of manifold structure. Larger values will result in more global structure being preserved at the loss of detailed local structure. In general, this parameter should often be in the range 5 to 50, with a choice of 10 to 15 being a sensible default.

What would you choose as the best value of **min_dist** and **n_neighbors** for this example?



n_neighbors: 15



n_neighbors: 50

Min_dist: 0.1

0.5

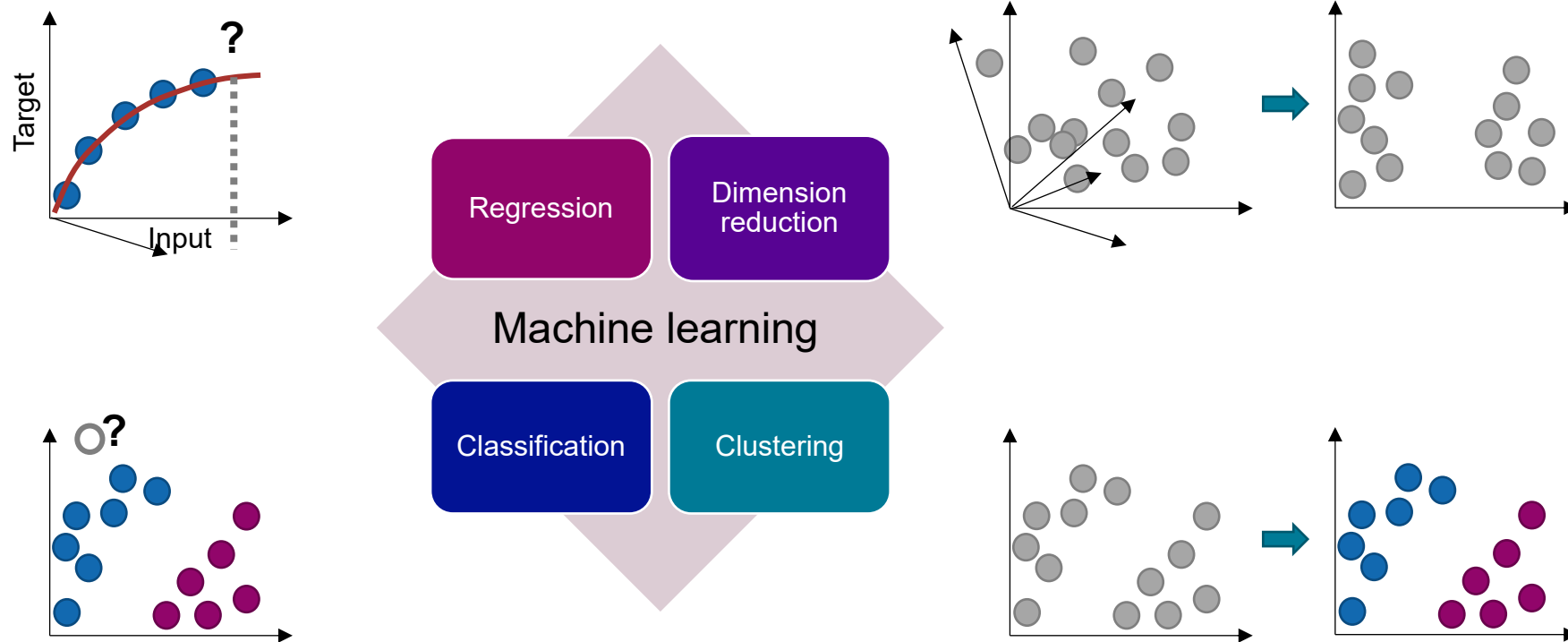
1

- | | |
|--------|-----------------------------|
| ● BLCA | Bladder |
| ● BRCA | Breast |
| ● CESC | Cervical squamous cell |
| ● ESCA | Esophageal |
| ● HNSC | Head and Neck squamous cell |
| ● KIRC | Kidney clear cell |
| ● KIRP | Kidney papillary |
| ● LIHC | Liver |
| ● OV | Ovarian |
| ● SKCM | Skin Cutaneous Melanoma |
| ● STAD | Stomach |
| ● THCA | Thyroid |
| ● UCEC | Uterine Corpus Endometrial |

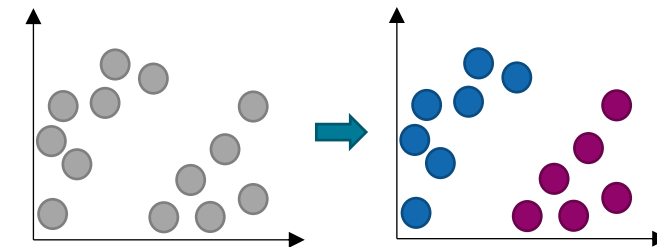
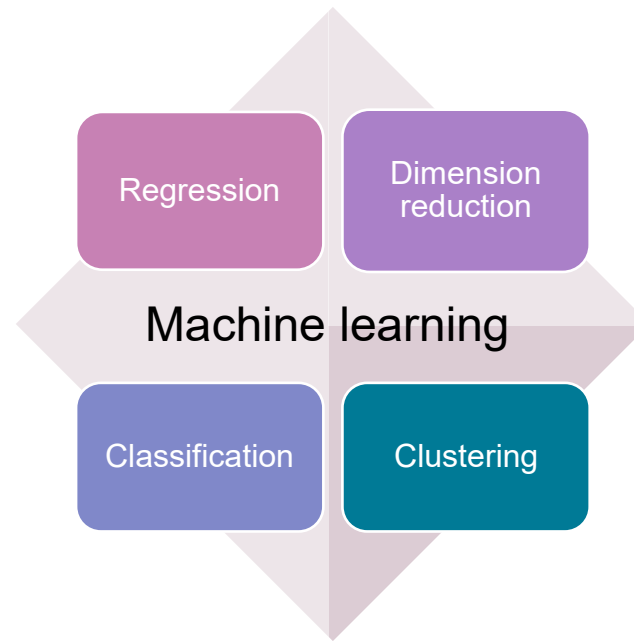
Take home message: dimensionality reduction

- One should often try different methods with different parameters to get different views on our data
- Random noise does not always look random, and sometimes one can see shapes
- Projections on the first n principal components can be used as input (instead of the original X) to other dimension reduction methods such as tSNE to reduce execution time.

Map of classical machine learning methods

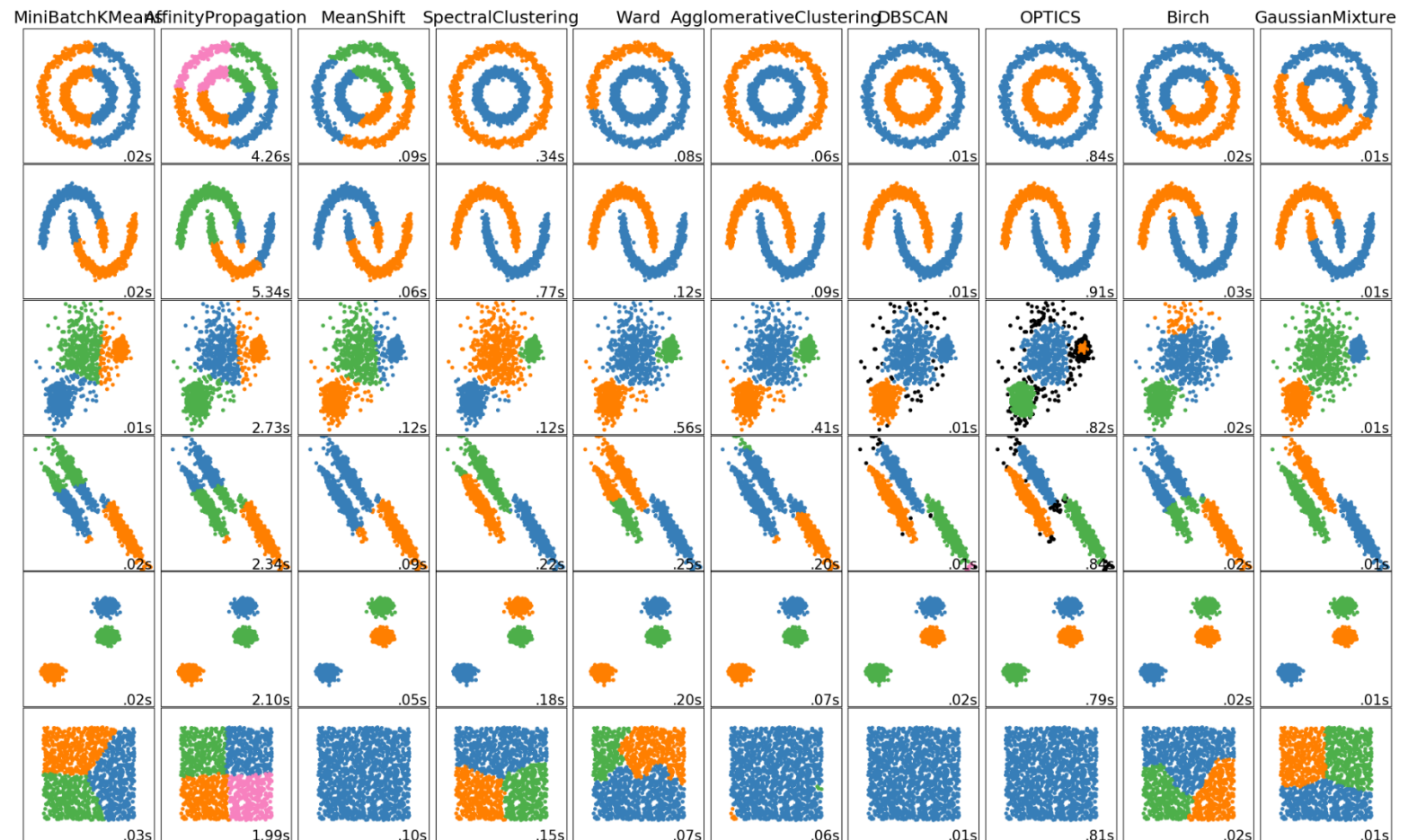


Map of classical machine learning methods



Clustering methods

- K-means
- Gaussian mixture models
- Spectral clustering
- Hierarchical clustering

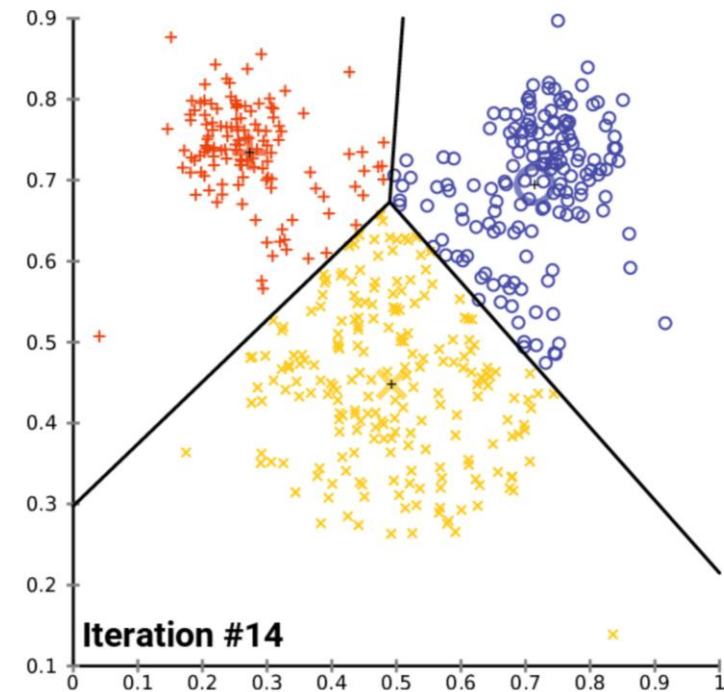
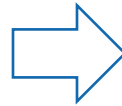
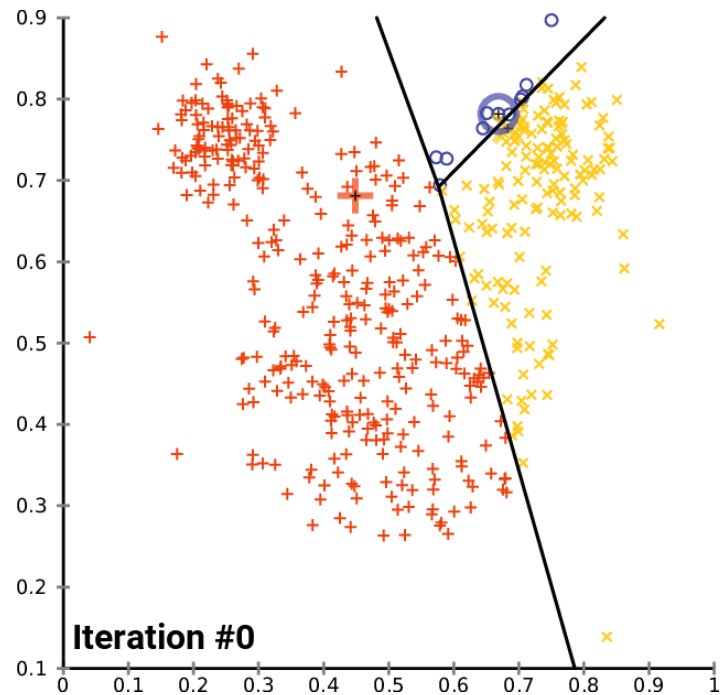


From <https://scikit-learn.org/stable/modules/clustering.html>

K-means

Attempts to minimize
$$L = \sum_{\substack{k=1 \\ \bar{x}_i \in C_k}}^K \|\bar{x}_i - \bar{\mu}_k\|^2$$

- **k-means**: aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean (cluster centers)

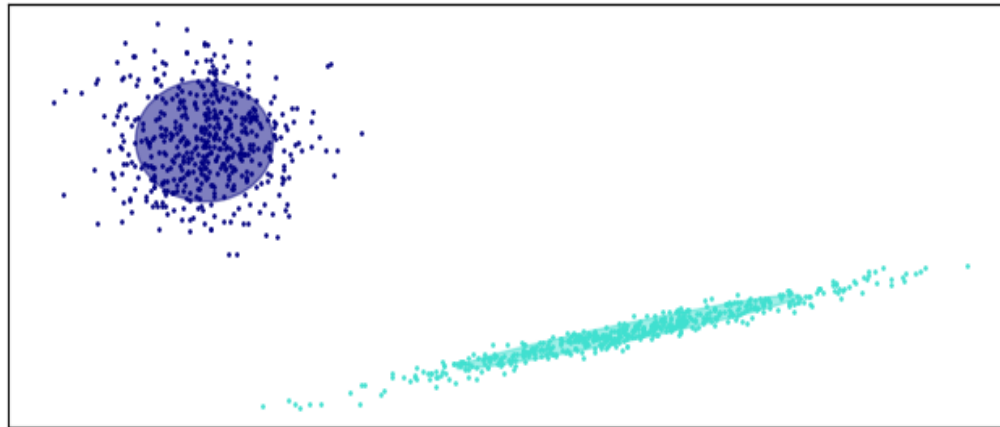


- k-means clustering tends to find clusters of comparable spatial extent

Gaussian mixture models (GMM)

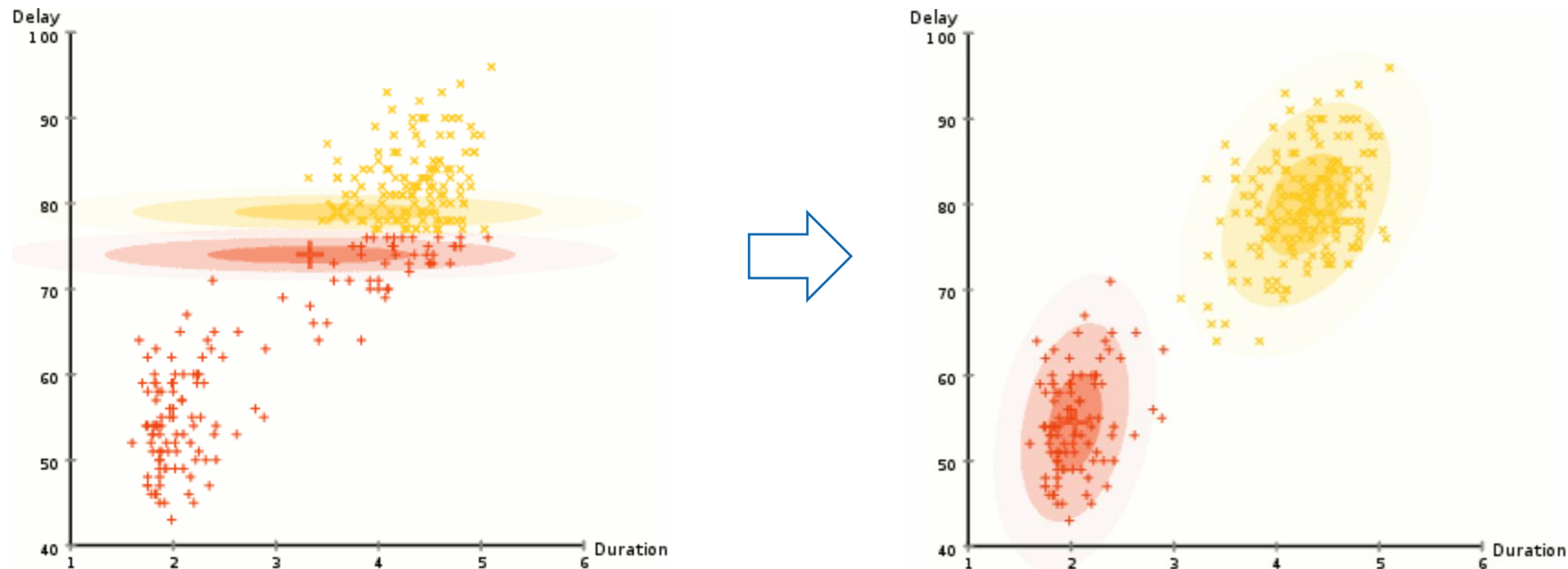
Attempts to maximize $L(\{\pi_k, \mu_k, \Sigma_k\}_{k=1}^K) = - \sum_{i=1}^N \log \left(\sum_{k=1}^K \pi_k \mathcal{N}(x_i | \mu_k, \Sigma_k) \right)$

GMM: probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters



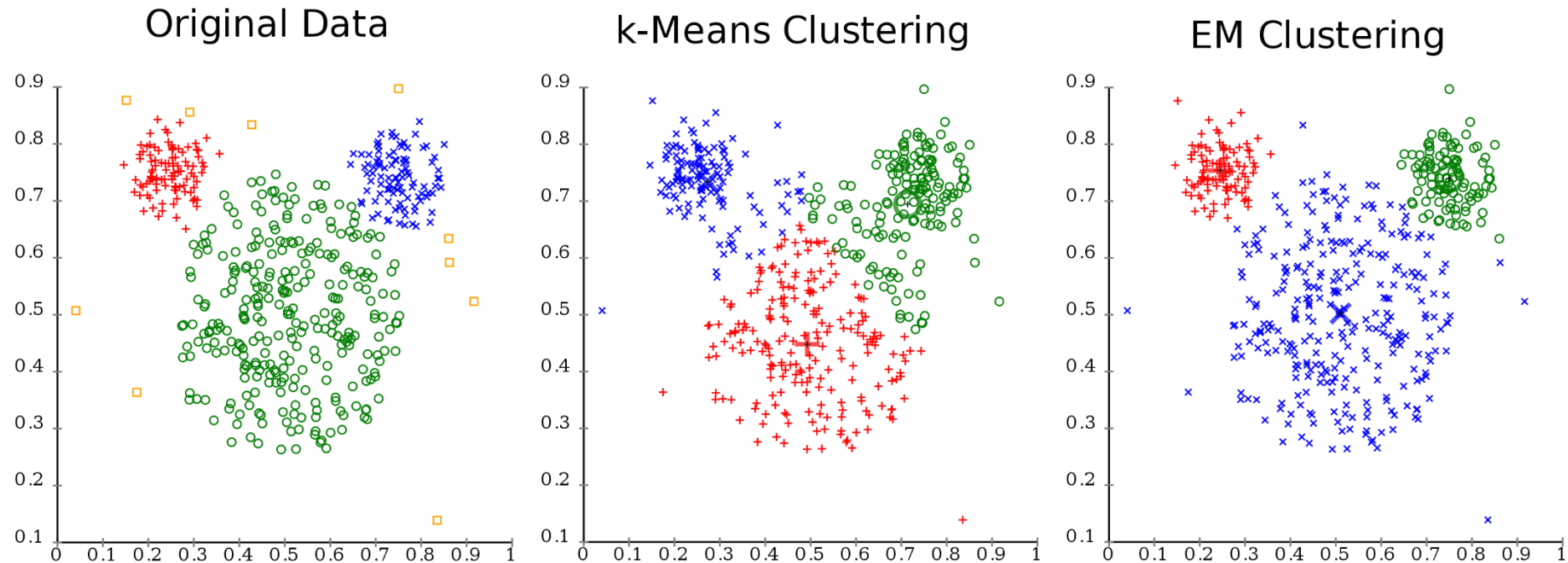
Gaussian mixture models (GMM)

GMM: probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters



k-means clustering tends to find clusters of comparable spatial extent, while the GMM expectation-maximization mechanism allows clusters to have different shapes.

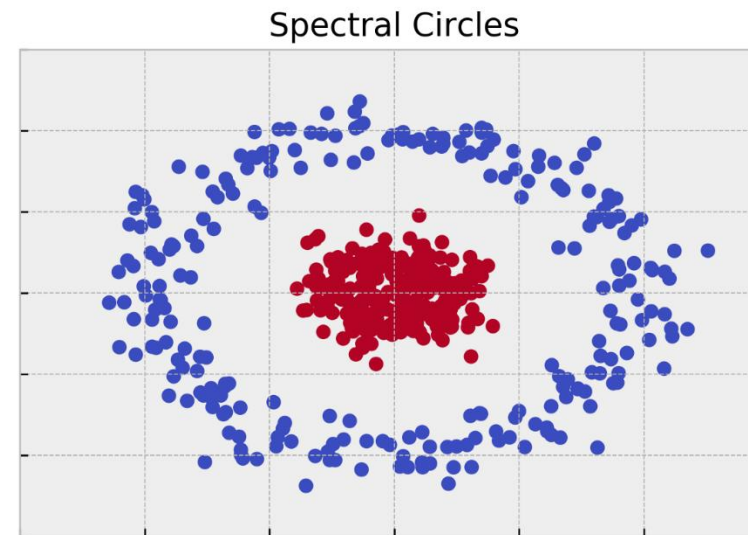
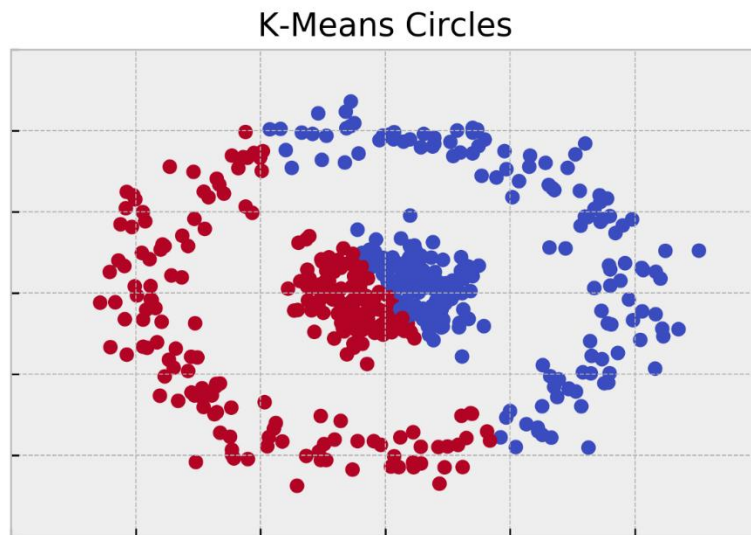
Different cluster analysis results on "mouse" data set:



Also, GMMs support mixed membership

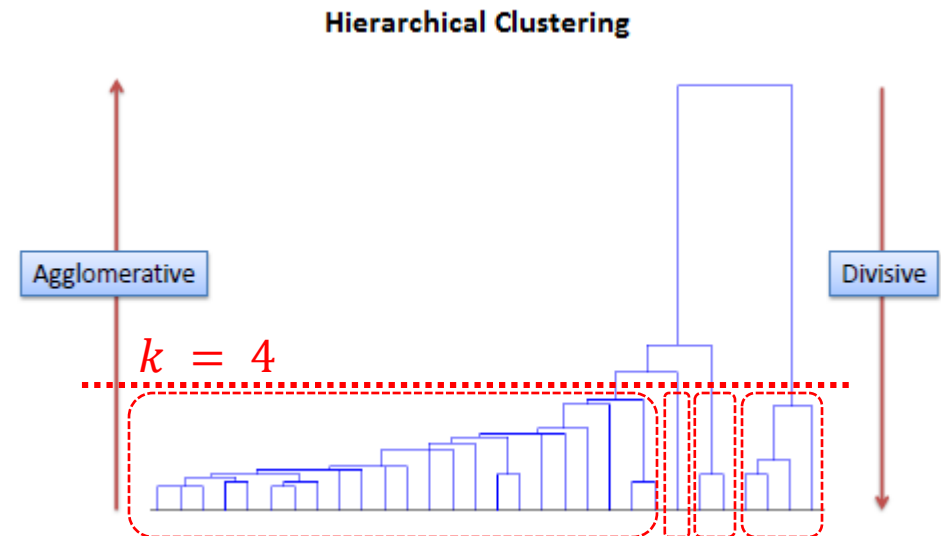
Spectral clustering: relies on the assumption that “close” points should belong in the same cluster

- Spectral clustering: uses a standard clustering method (e.g., k-means) on relevant eigenvectors of a Laplacian matrix L of symmetric data similarity matrix A . – can also use k-nearest neighbors graphs for construction of A .
- $L := D - A$, where D is the diagonal matrix, such as $D_{ii} = \sum_j A_{ij}$.



Hierarchical clustering

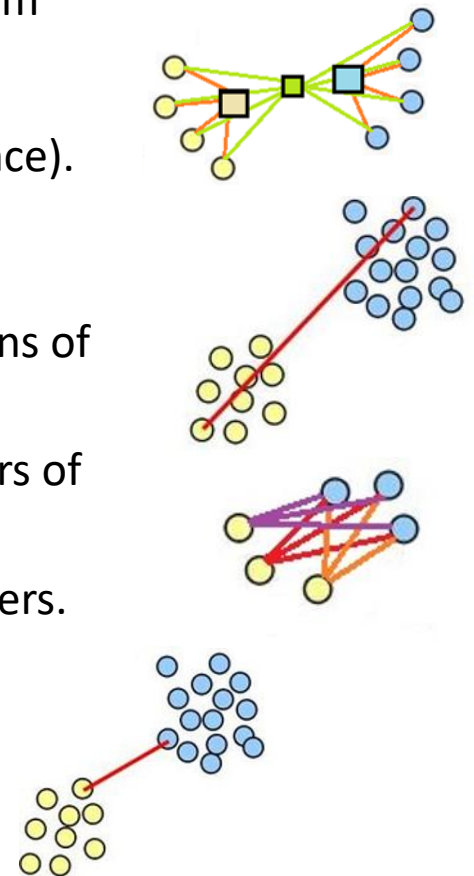
- Hierarchical clustering: family of clustering algorithms that build nested clusters by merging or splitting them successively. This hierarchy of clusters is represented as a tree (or dendrogram).
- **Agglomerative approach:** This is a "bottom-up" approach: each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy.
- **Divisive approach:** This is a "top-down" approach: all observations start in one cluster, and splits are performed recursively as one moves down the hierarchy.



Hierarchical clustering: important parameters

You choose:

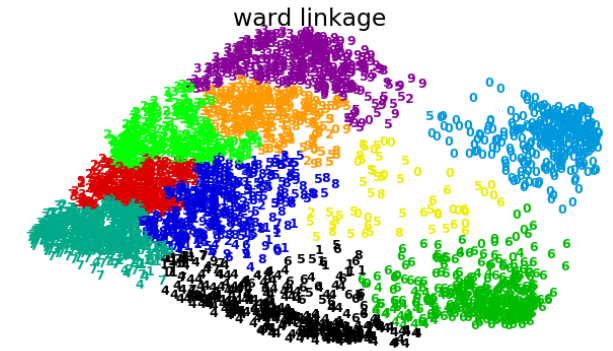
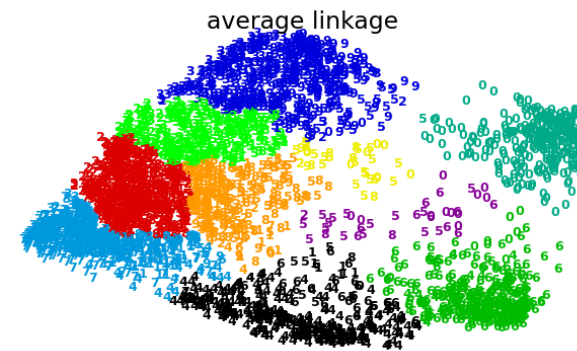
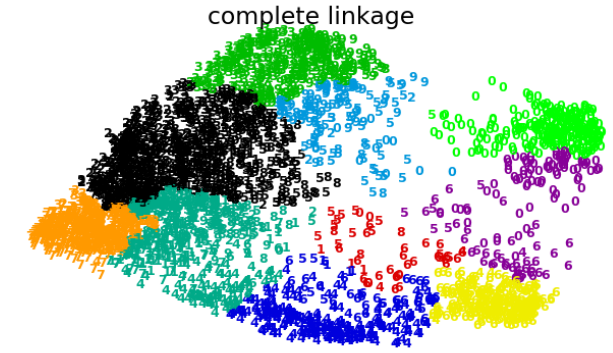
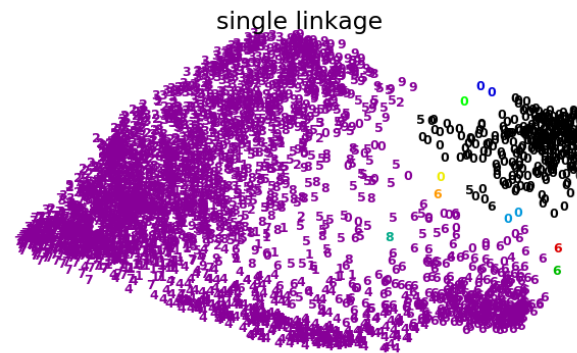
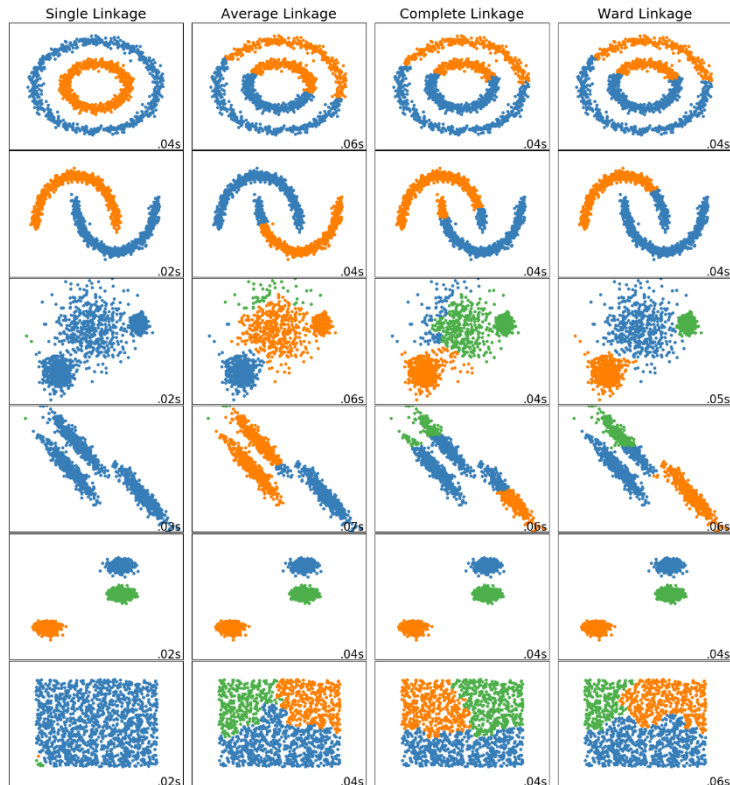
- **Distance metric** (between observations): Euclidean, Squared Euclidean, Manhattan, Maximum
- **Linkage criterium** (distance between sets of observations):
 - “**Ward**” minimizes the sum of squared differences within all clusters (within-cluster variance). It is a variance-minimizing approach and in this sense is similar to the k-means objective function but tackled with an agglomerative hierarchical approach.
 - “**Maximum**” or “complete linkage” minimizes the maximum distance between observations of pairs of clusters.
 - “**Average linkage**” minimizes the average of the distances between all observations of pairs of clusters.
 - “**Single linkage**” minimizes the distance between the closest observations of pairs of clusters.



<https://blog.tdwi.eu/hierarchical-clustering-in-python/>

Hierarchical clustering: important parameters

- **Linkage criterium** (distance between sets of observations):

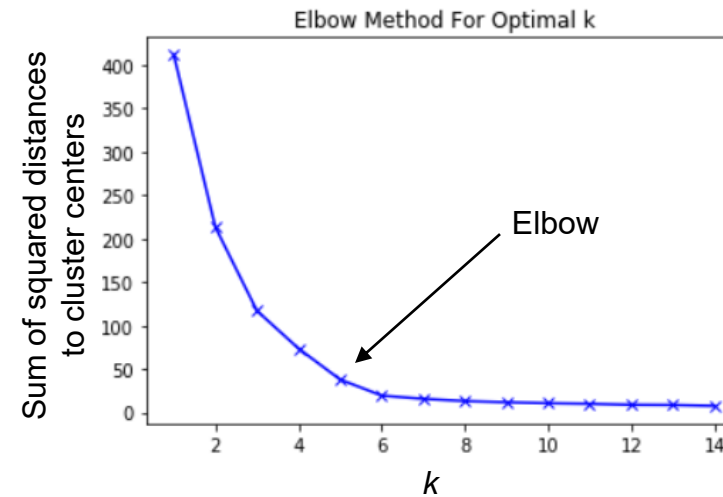


“Ward” gives the most regular cluster sizes

From <https://scikit-learn.org/stable/modules/clustering.html#hierarchical-clustering>
https://scikit-learn.org/stable/auto_examples/cluster/plot_digits_linkage.html

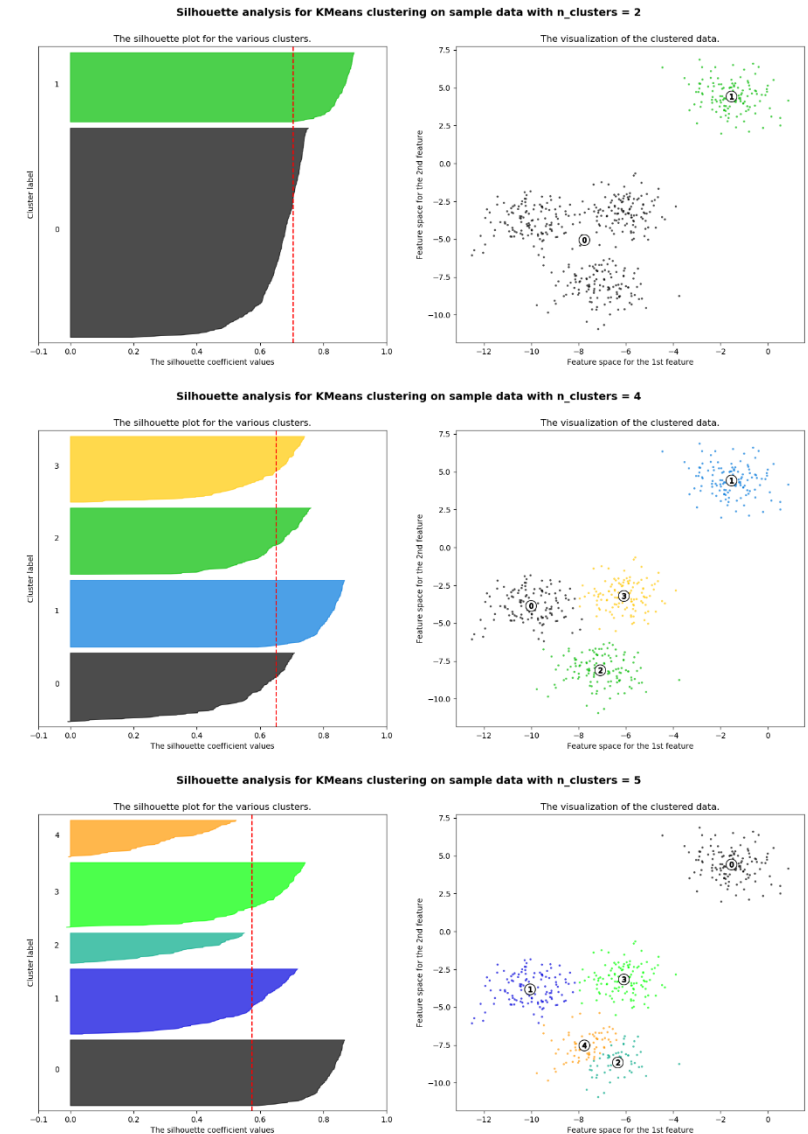
How to choose the best number of clusters k ?

- Elbow method (for sum of squared distances to cluster centers)
 - How find the “elbow”? By eye or <https://github.com/arvkevi/kneed>
- Silhouette analysis
- BIC and AIC



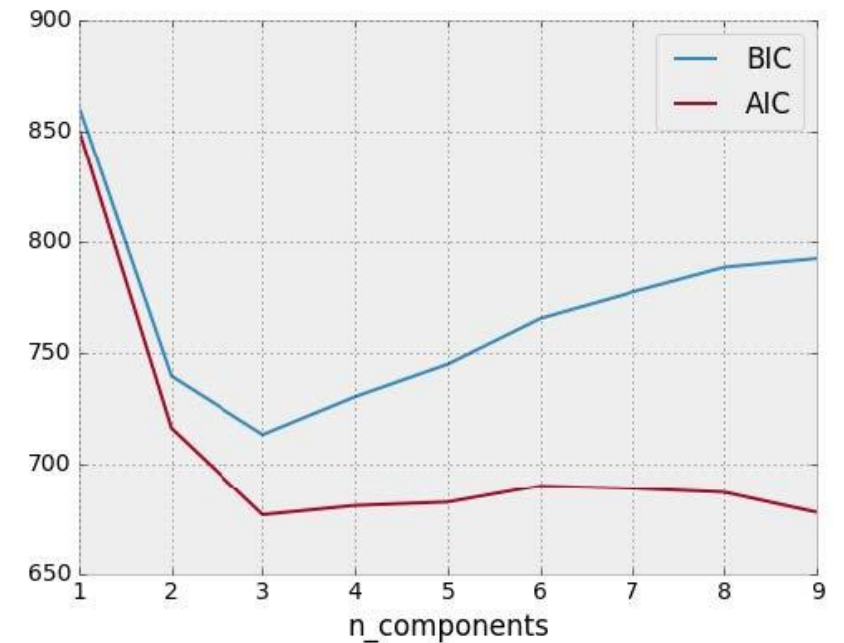
How to choose the best number of clusters k?

- Elbow method
- Silhouette analysis
 - Plots silhouette score: a measure of how close each point in one cluster is to points in the neighboring clusters, in $[-1, 1]$.
 - Values near +1 indicate that the sample is far away from the neighboring clusters. A value of 0 indicates that the sample is on very close to the decision boundary between two neighboring clusters and negative values indicate that those samples might have been assigned to the wrong cluster.



How to choose the best number of clusters k?

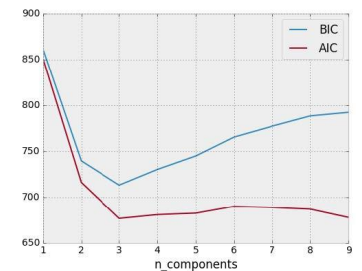
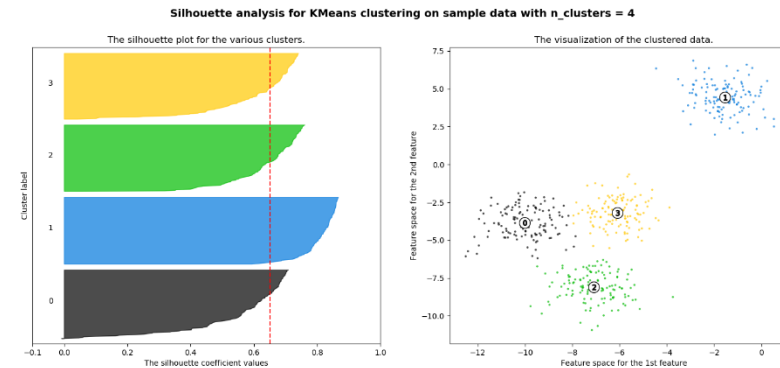
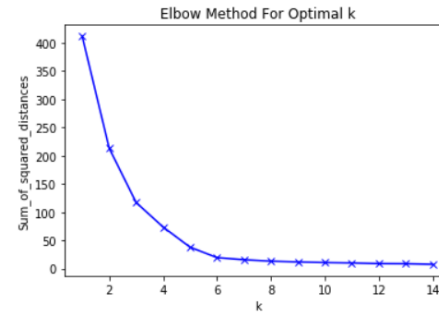
- Elbow method
- Silhouette analysis
- Akaike information criterion (AIC) or Bayesian information criterion (BIC)
 - The BIC generally penalizes free parameters more strongly than the AIC



From <https://sites.northwestern.edu/msia/2016/12/08/k-means-shouldnt-be-our-only-choice/>

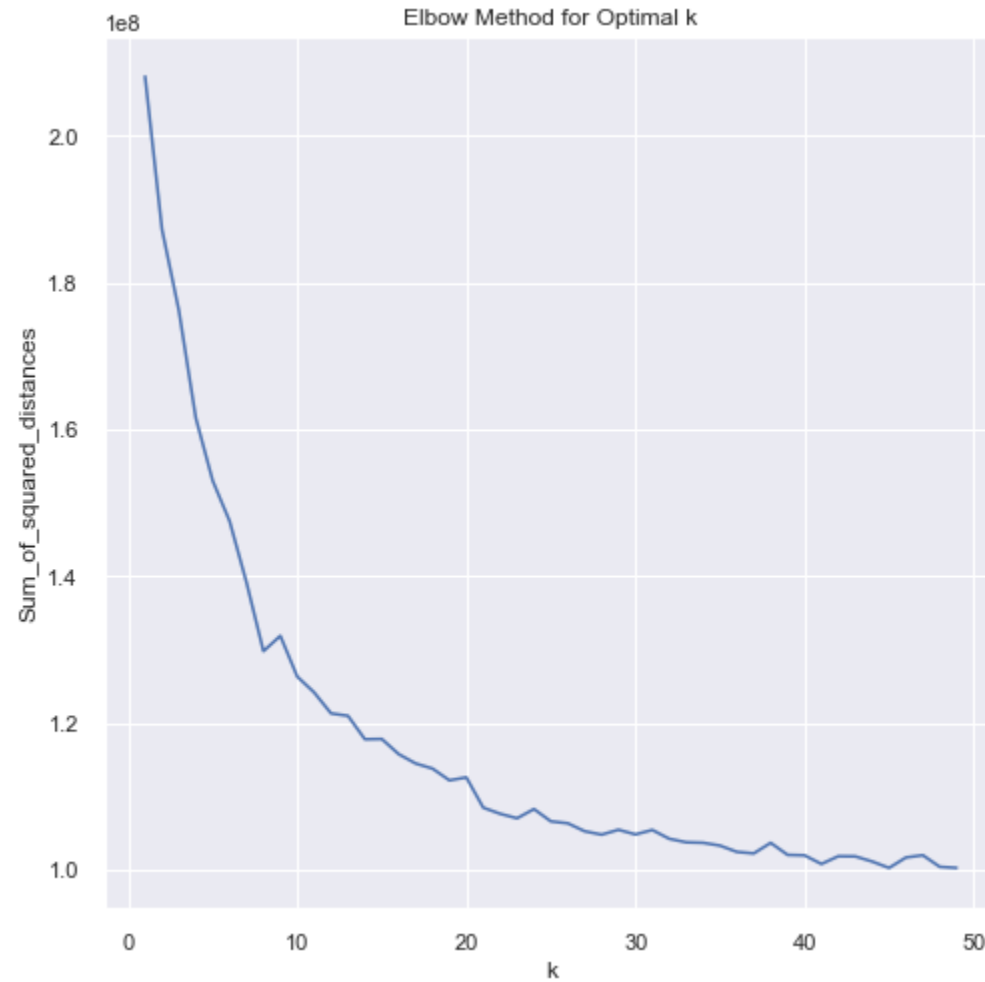
How to choose the best number of clusters k?

- Elbow method
- Silhouette analysis
- BIC and AIC



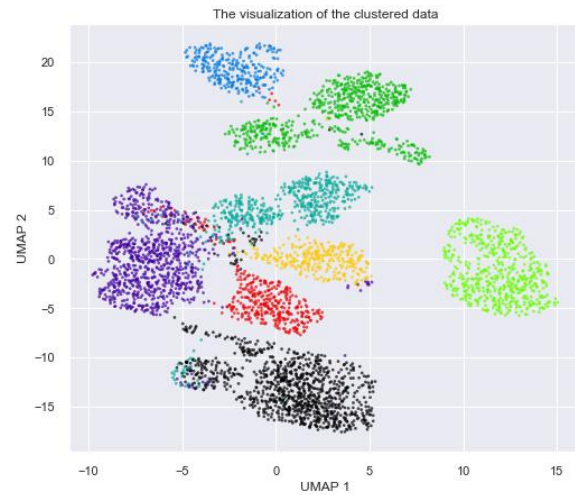
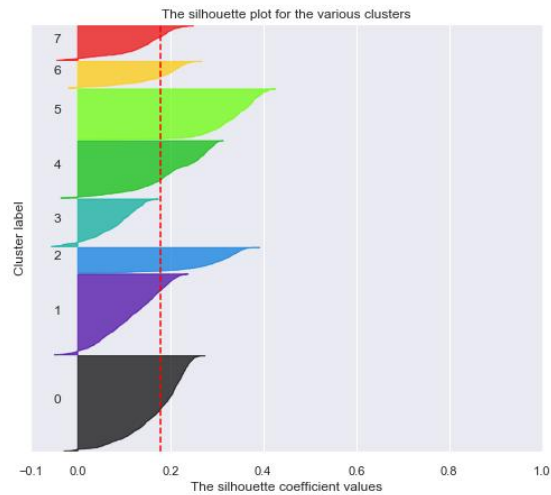
Let's go to our hands on exercise to see how it works!

Which value of k would you choose?
(Mini-batch k-means clustering method)



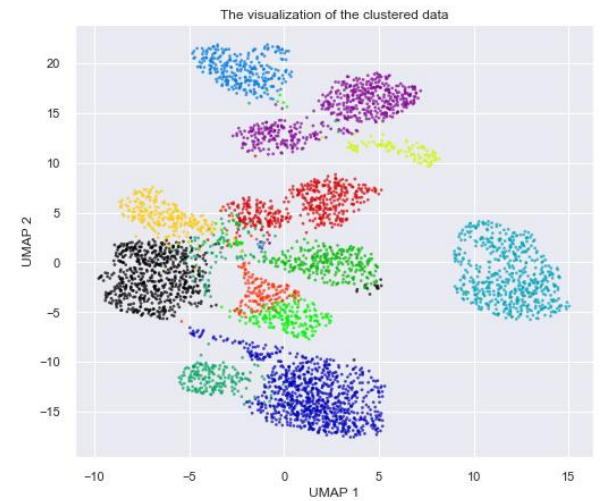
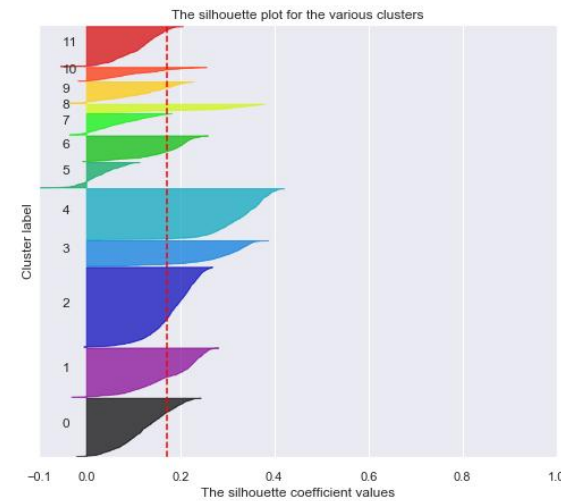
Which value of k would you choose?

Silhouette analysis for KMeans clustering on sample data with n_clusters = 8

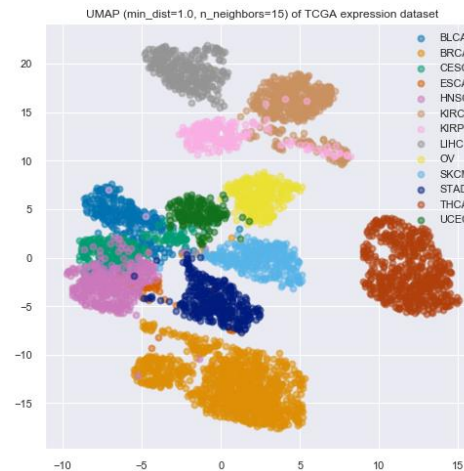


$k = 8$

Silhouette analysis for KMeans clustering on sample data with n_clusters = 12



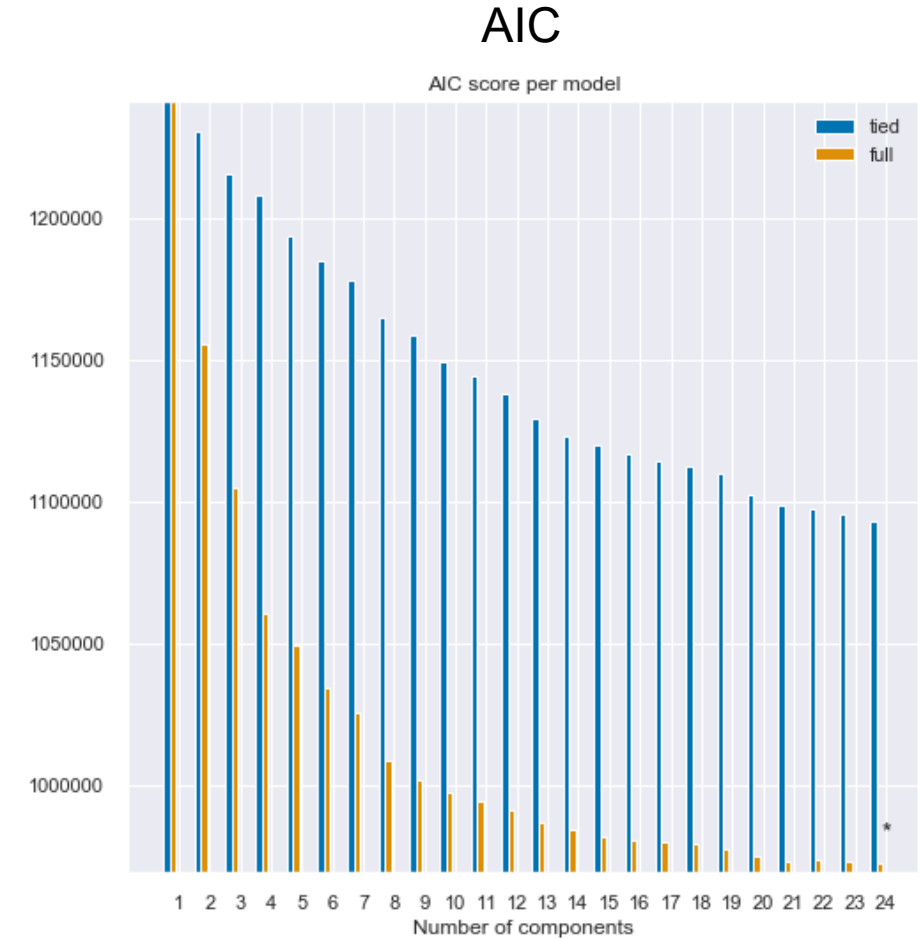
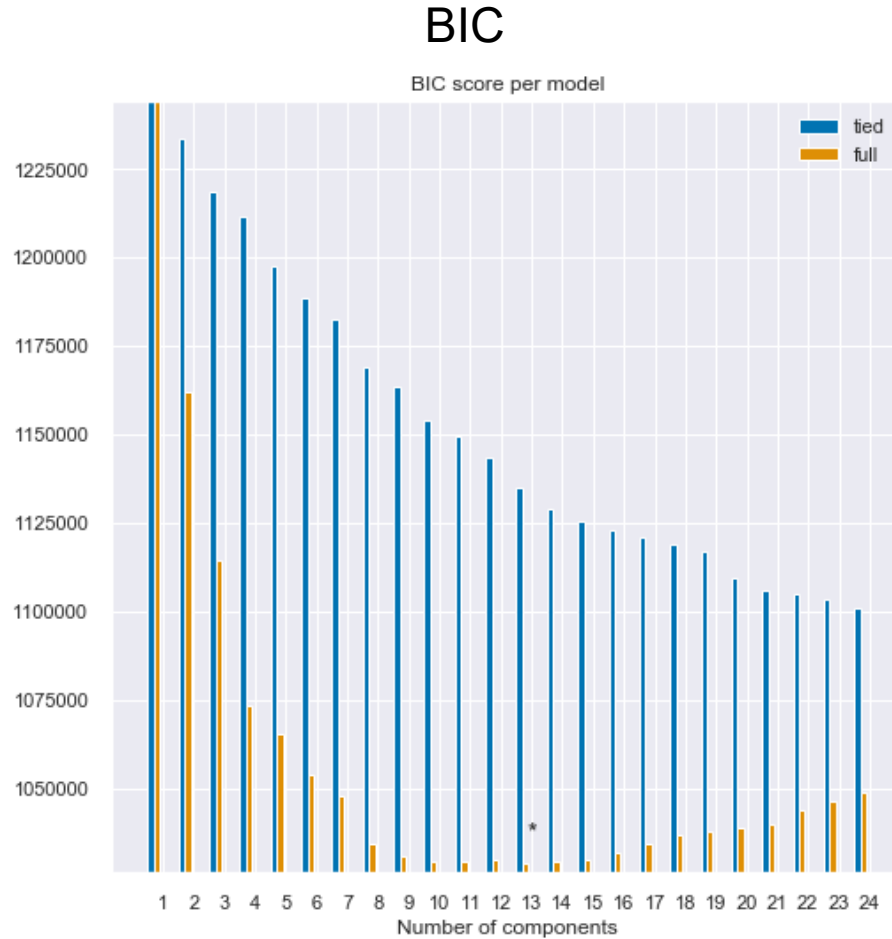
$k = 12$



True types

- | | |
|--------|-----------------------------|
| ● BLCA | Bladder |
| ● BRCA | Breast |
| ● CESC | Cervical squamous cell |
| ● ESCA | Esophageal |
| ● HNSC | Head and Neck squamous cell |
| ● KIRC | Kidney clear cell |
| ● KIRP | Kidney papillary |
| ● LIRC | Liver |
| ● OV | Ovarian |
| ● SKCM | Skin Cutaneous Melanoma |
| ● STAD | Stomach |
| ● THCA | Thyroid |
| ● UCEC | Uterine Corpus Endometrial |

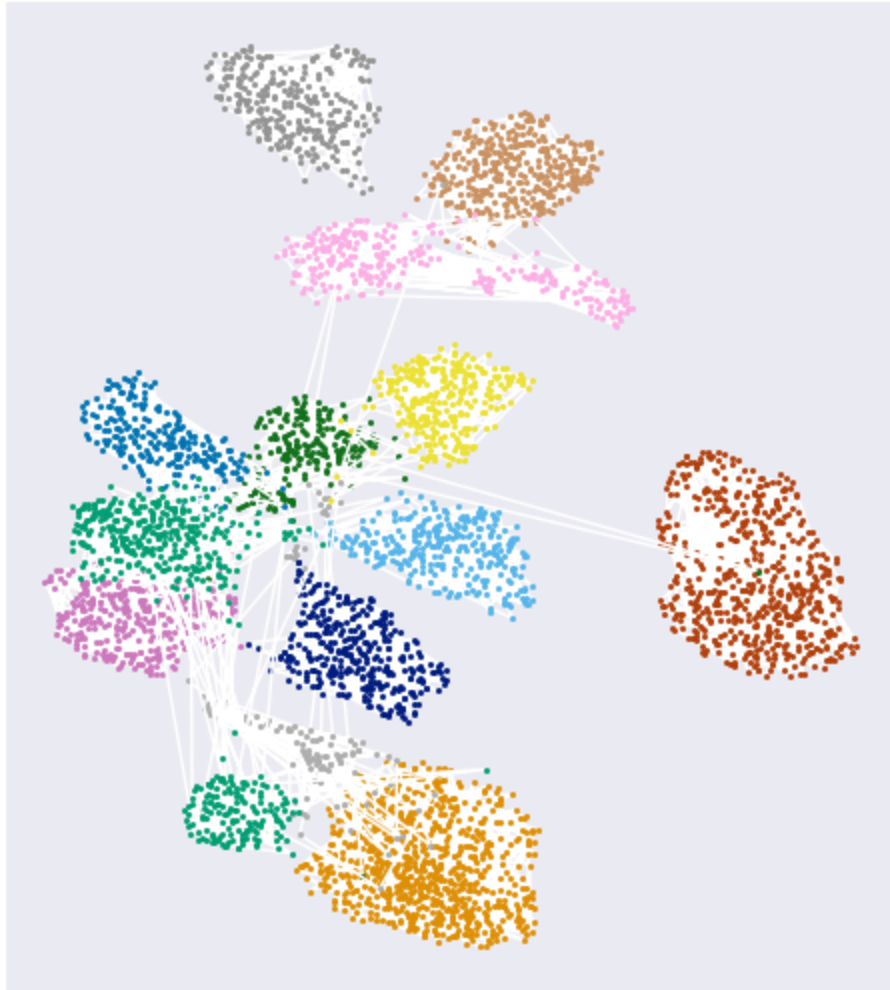
Which value of k would you choose? (Gaussian mixture model)



Which value of k would you choose? (Gaussian mixture model)

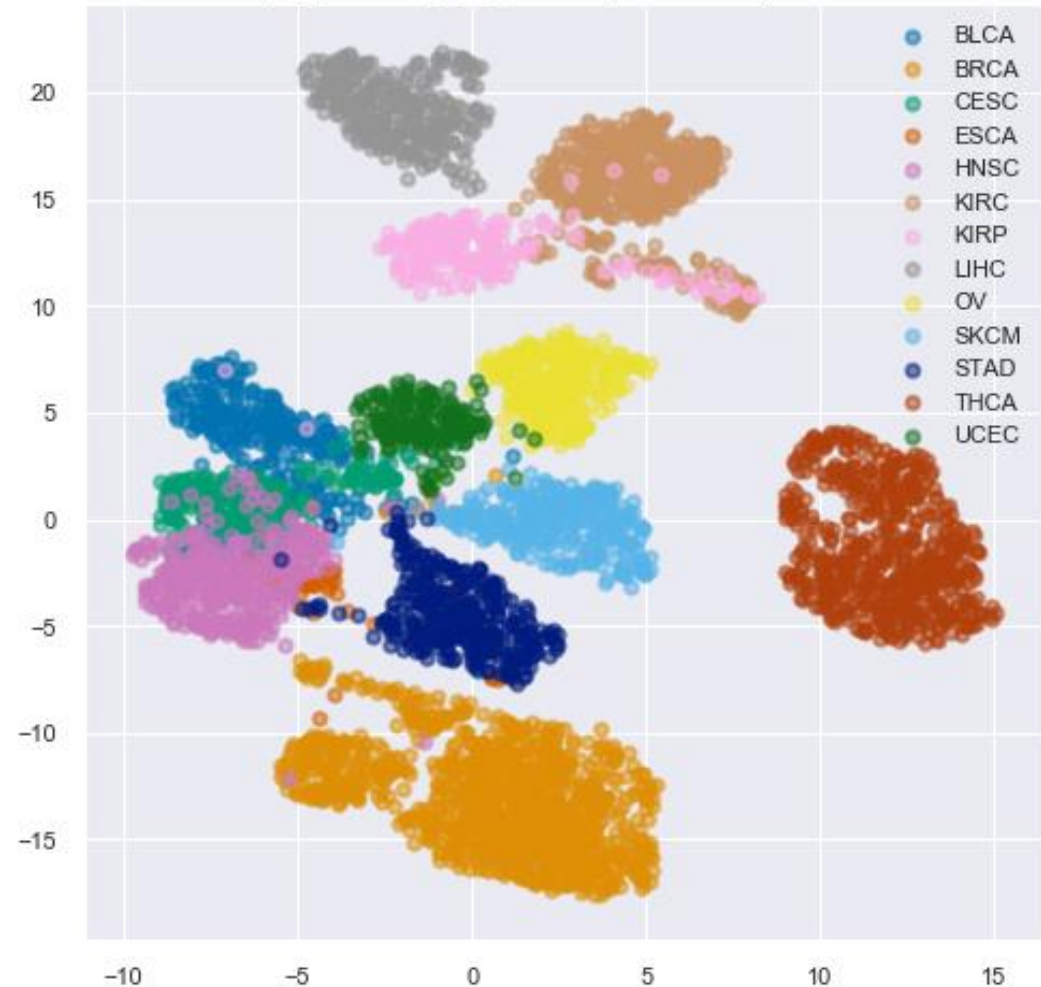
Best solution according to BIC: $k=13$

Gaussian mixture models for clustering TCGA expression dataset



True labels

UMAP (min_dist=1.0, n_neighbors=15) of TCGA expression dataset



NMI and ARI allows to see which clustering model is the best on our data set

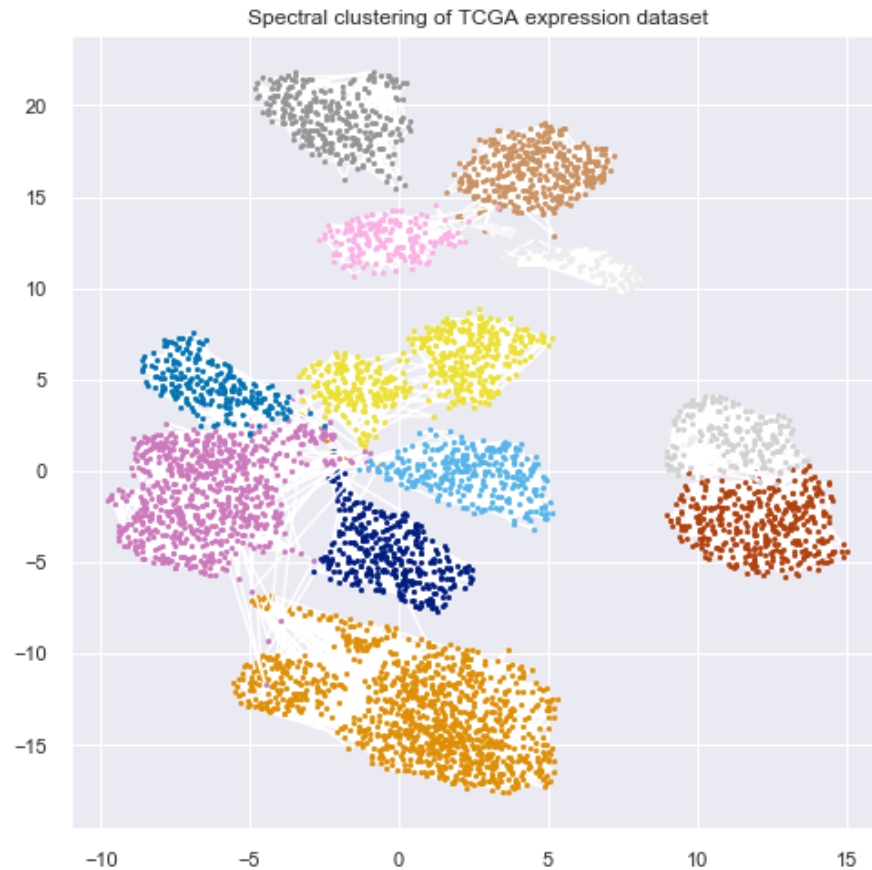
When true labels are available (rarely the case though):

NMI = Normalized Mutual Information }
ARI = Adjusted rand index } measure mutual dependence between the true and predicted labels
(can be also used as a measure of similarity between two clusterings)

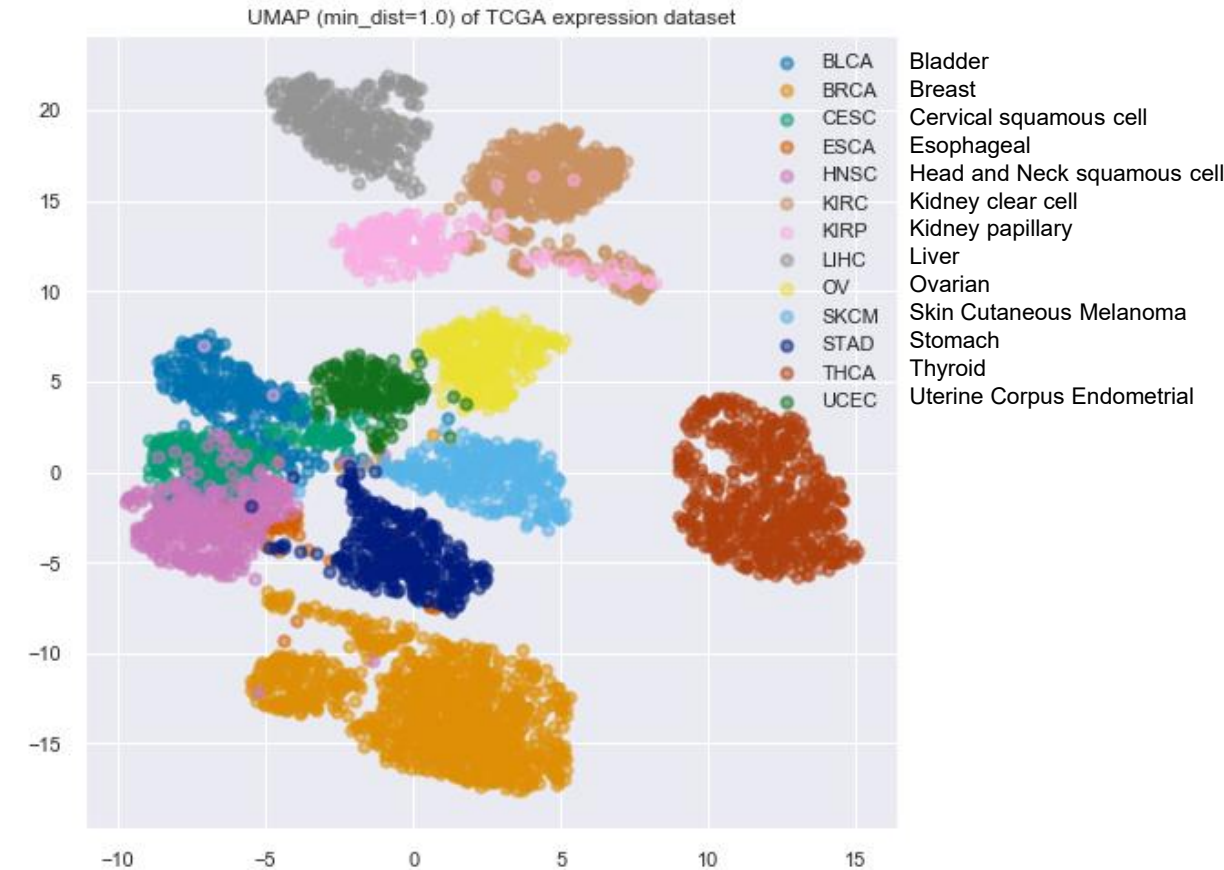
Method	k	ARI	NMI
Mini-batch k-means	8	0.7479	0.8102
Mini-batch k-means	19	0.7426	0.8179
Gaussian mixture model	13	0.7600	0.8479
Spectral clustering	13	0.8148	0.8671
Hierarchical clustering (linkage='ward')	13	0.7394	0.8379

Spectral clustering result (k=13)

Best solution according to NMI



True labels



Take home message: clustering

- The choice of the clustering method should be advised by data structure
 - Visualize the data first
 - E.g, ellipsoids => GMM; sophisticated connected components => structural clustering
- Choosing the number of clusters can be done using:
 - Elbow method
 - Silhouette method
 - BIC or AIC
- Different methods for the estimation of the number of clusters provide different results
 - E.g., BIC is more conservative than AIC

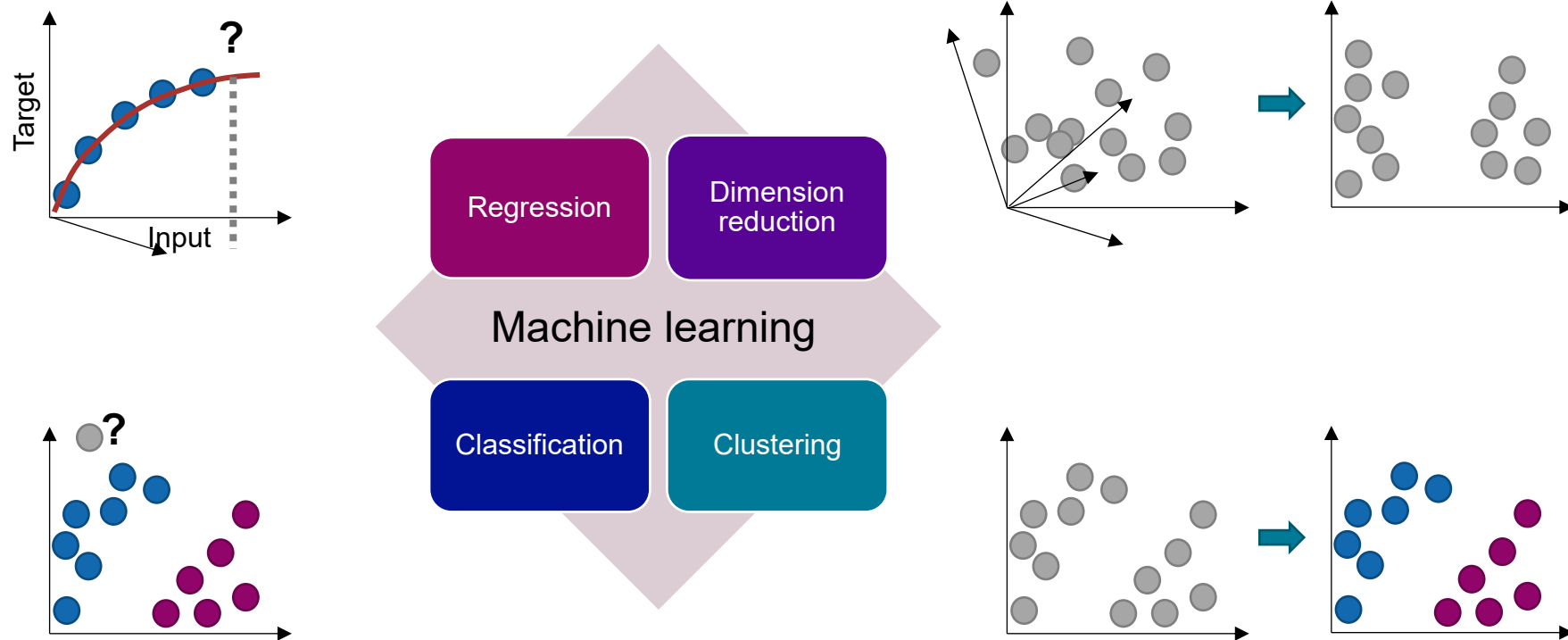
What we did not cover today

- Graph-based clustering methods (Leiden and Louvain algorithms, commonly used on scRNA-seq data)
- Topic models/LDA
- Autoencoder-based methods for dimensionality reduction
- Integration of different data types (e.g., clusters of clusters)

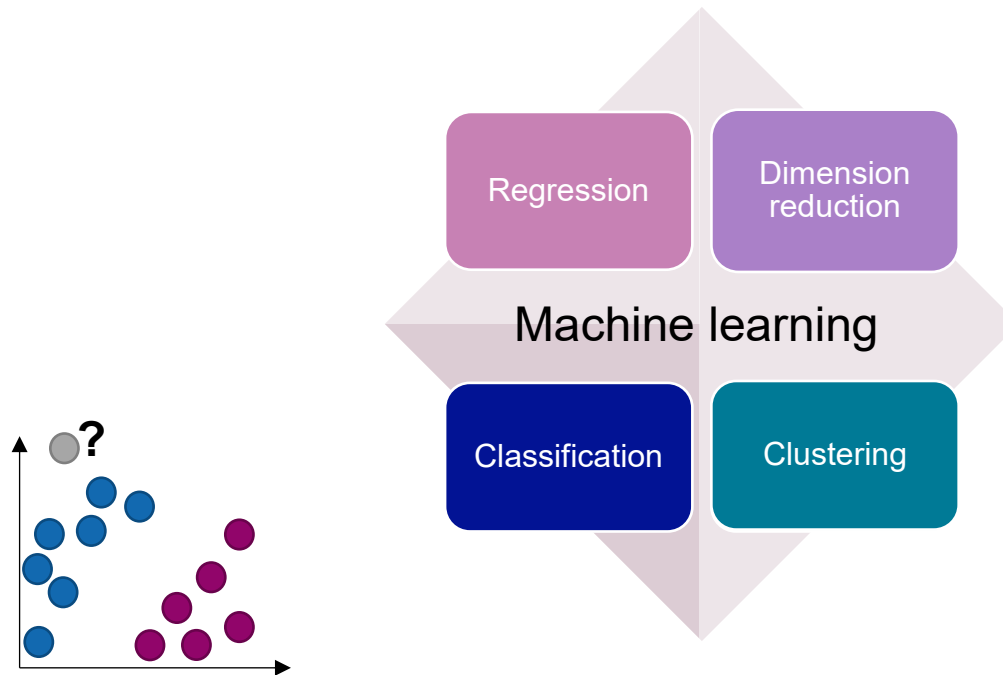
Take home message

- Dimensionality reduction can be used as a step prior to clustering
- One usually tries different dimensionality reduction techniques to choose the one that fits the expectations
- The choice of clustering method should match the data structure

Map of classical machine learning methods

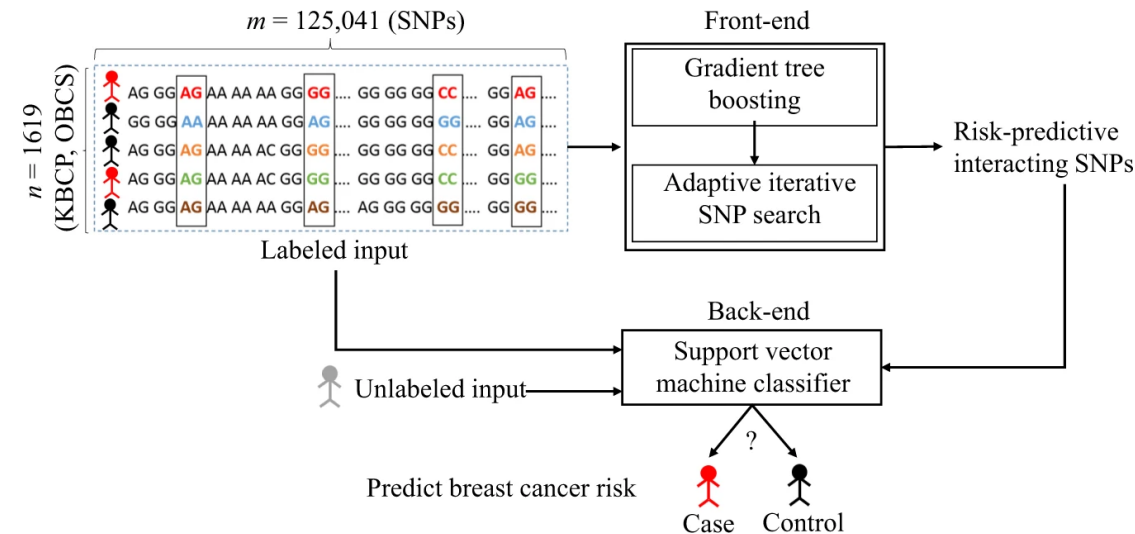


Map of classical machine learning methods



Classification: Biological examples

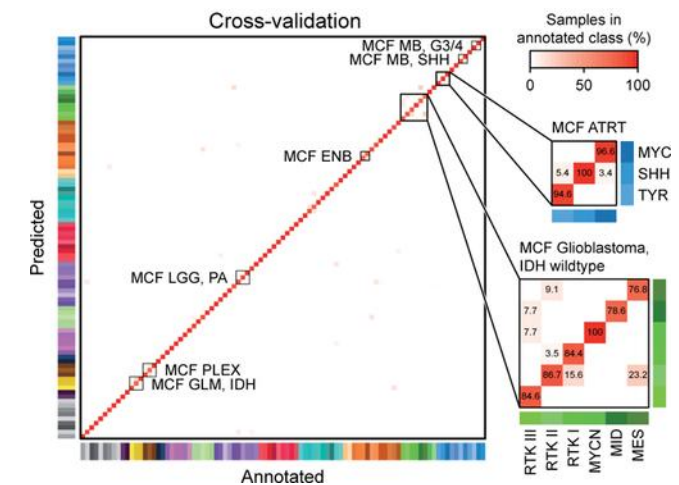
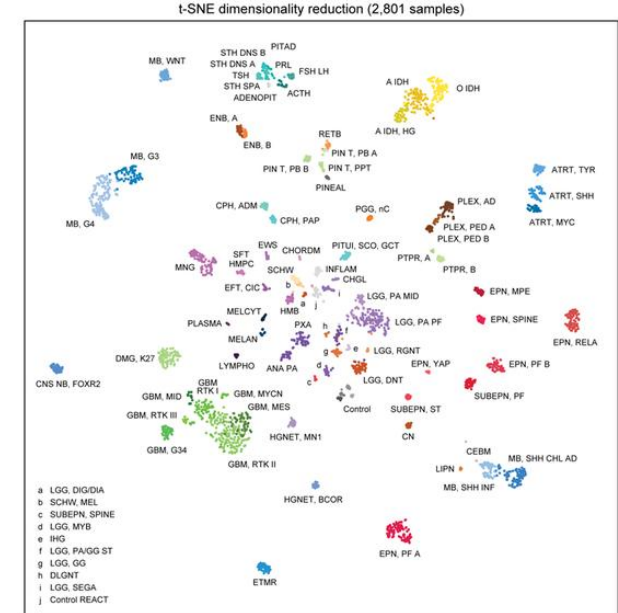
- Prediction of risk groups
 - **Machine learning identifies interacting genetic variants contributing to breast cancer risk [...]** Behravan et al., *Sci Rep.* 2018
- Primary diagnosis
 - **DNA methylation-based classification of central nervous system tumours.** Capper et al. *Nature* 2018



Proposed breast cancer risk prediction approach using identified risk-predictive interacting SNPs

Classification: Biological examples

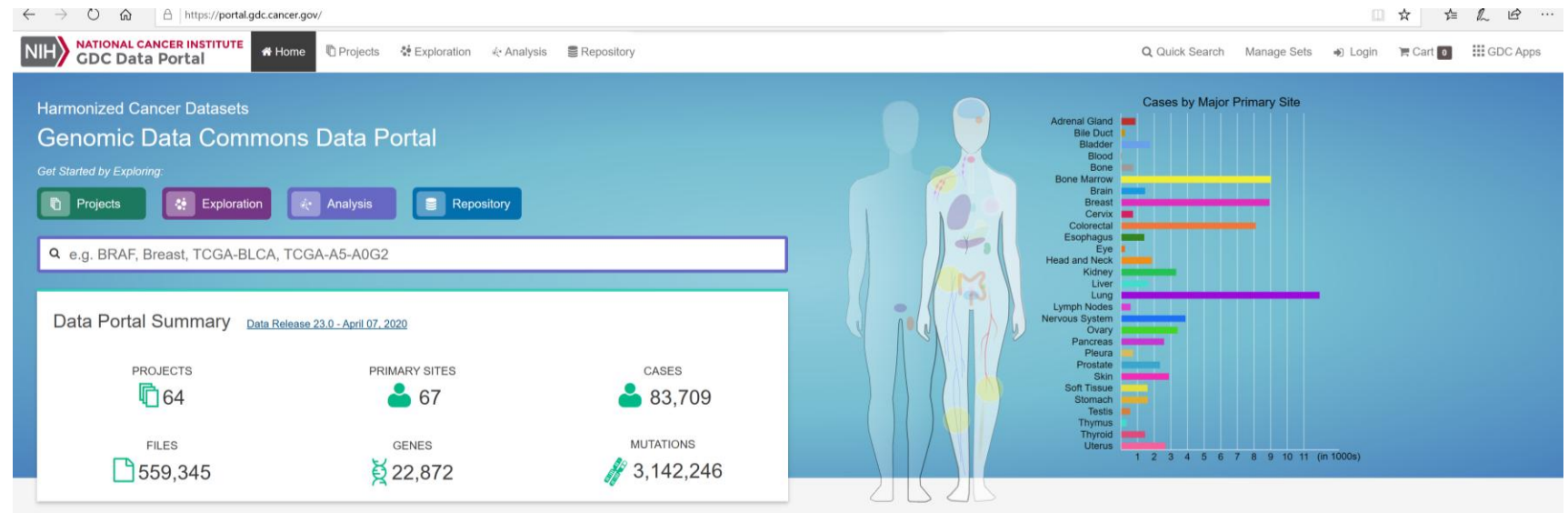
- Prediction of risk groups
 - Machine learning identifies interacting genetic variants contributing to breast cancer risk [...] Behravan et al., *Sci Rep.* 2018
- Primary diagnosis
 - DNA methylation-based classification of central nervous system tumours. Capper et al. *Nature* 2018



Hands-on:

1. Primary cancer diagnosis from gene expression
2. Breast cancer patients' stratification based on gene expression

- **Input:** The Cancer Genome Atlas (TCGA) mRNA expression data

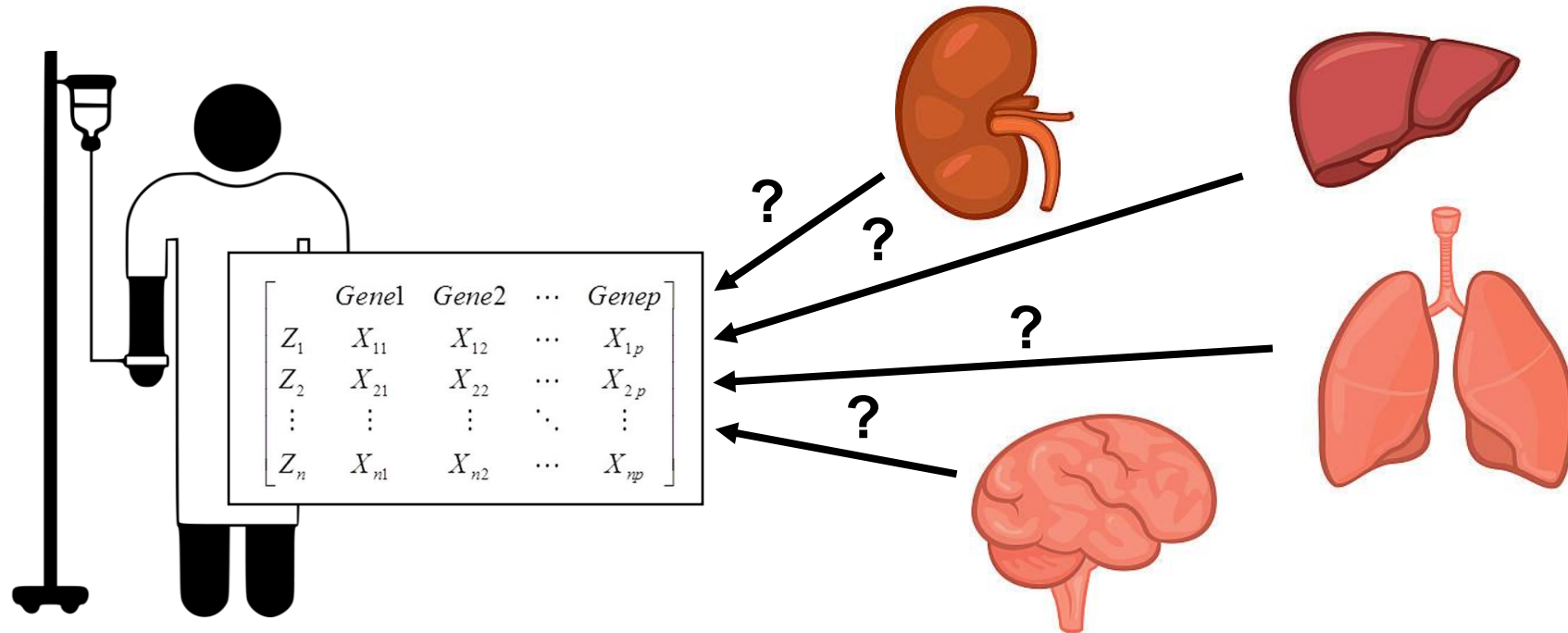


Hands-on:

1. Primary cancer diagnosis from gene expression
2. Breast cancer patients' stratification based on gene expression

- **TASK 1:**

Given mRNA expression, predict cancer type

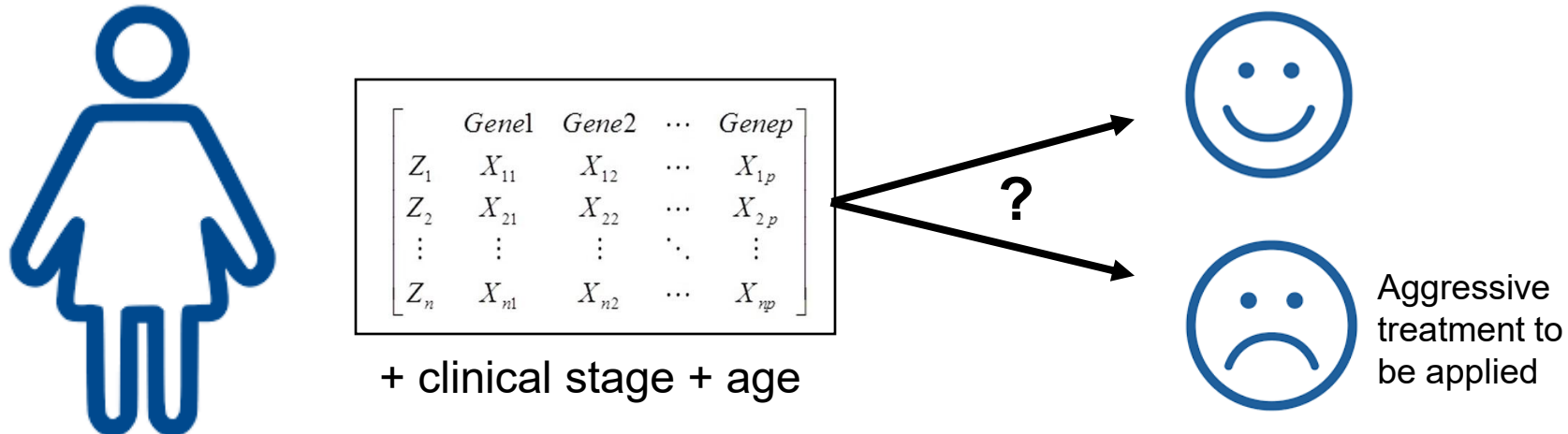


Hands-on:

1. Primary cancer diagnosis from gene expression
2. Breast cancer patients' stratification based on gene expression

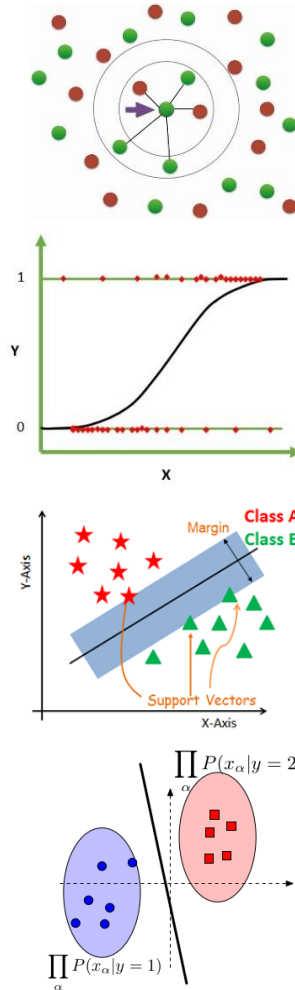
- **TASK 2:**

Given mRNA expression (and clinical data: stage, age), stratify patients according to good and bad prognosis

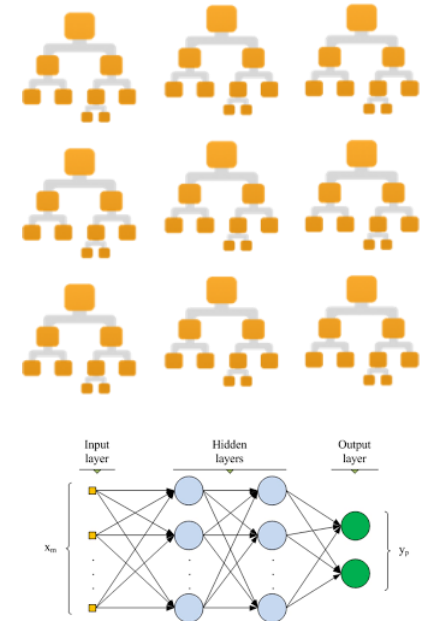


Standard approaches for classification

- k Nearest Neighbors (k-NN)
- Logistic Regression
- Logistic Regression with L1+L2 (Elastic Net) penalty
- Support Vector Machines (SVM)
- Naive Bayes (Gaussian)

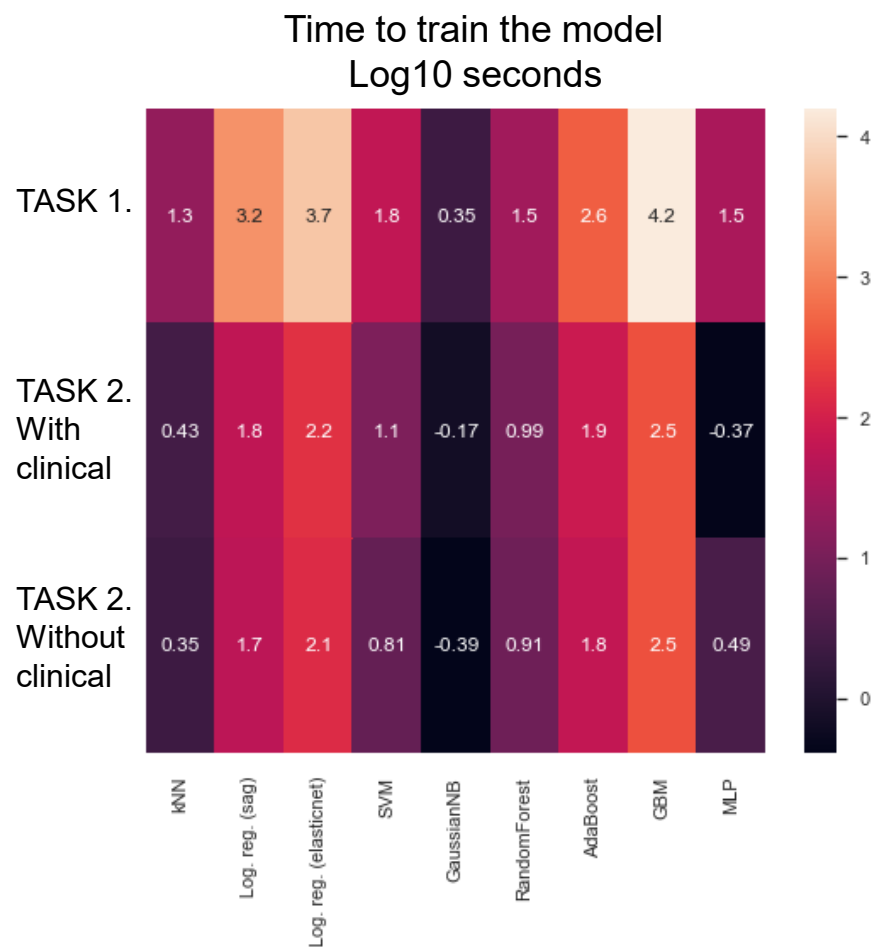


- Random Forest
- AdaBoost
- Gradient Tree Boosting (gradient boosting machine, GBM)
- Multi-layer perceptron (MLP)



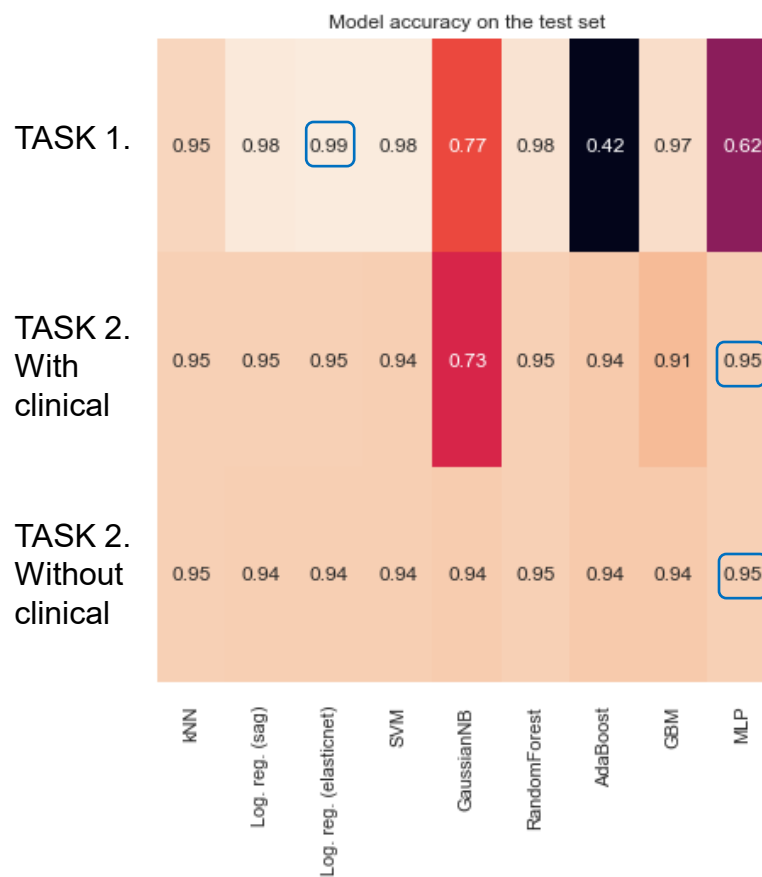
Hands-on:

1. Primary cancer diagnosis from gene expression
2. Breast cancer patients' stratification based on gene expression



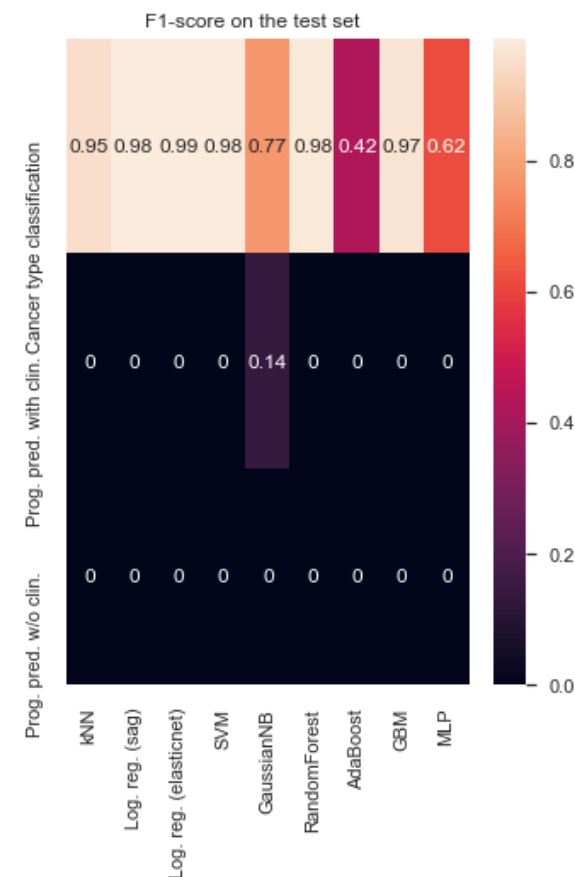
$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

Fraction predicted correctly



F1 score: harmonic mean of recall and precision

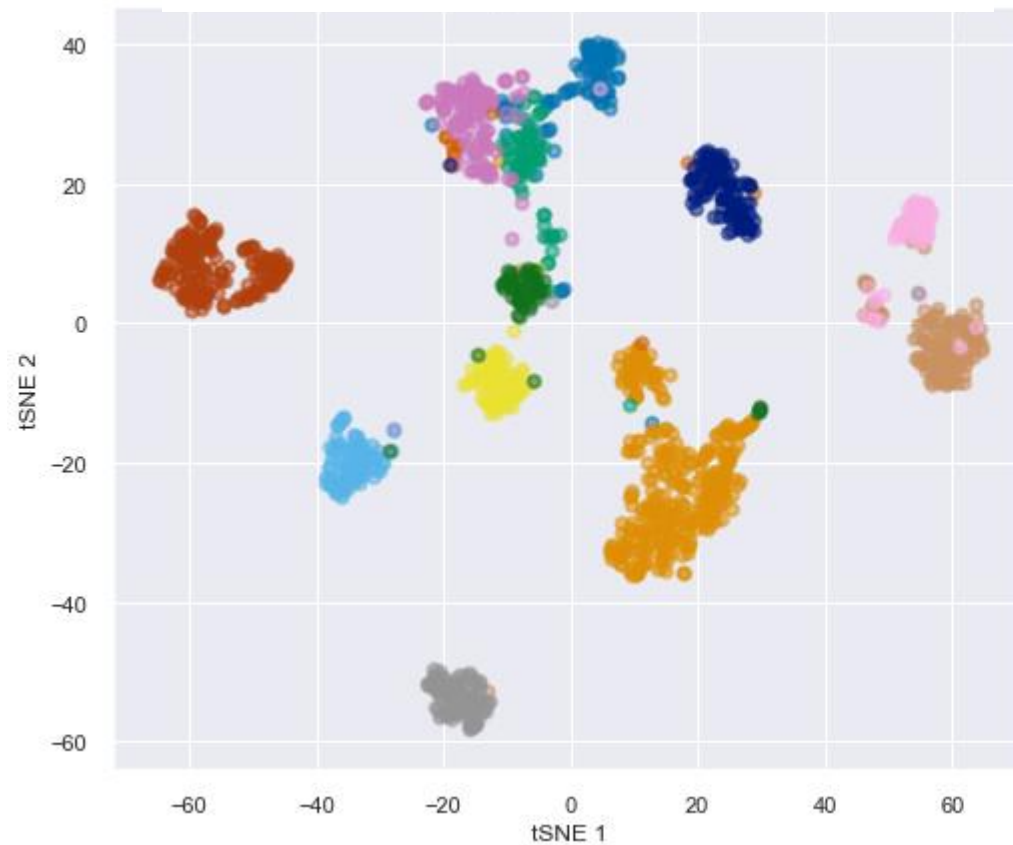
$$F1 = \frac{2}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}} = \frac{2 * (\text{precision} * \text{recall})}{\text{precision} + \text{recall}}$$



Hands-on:

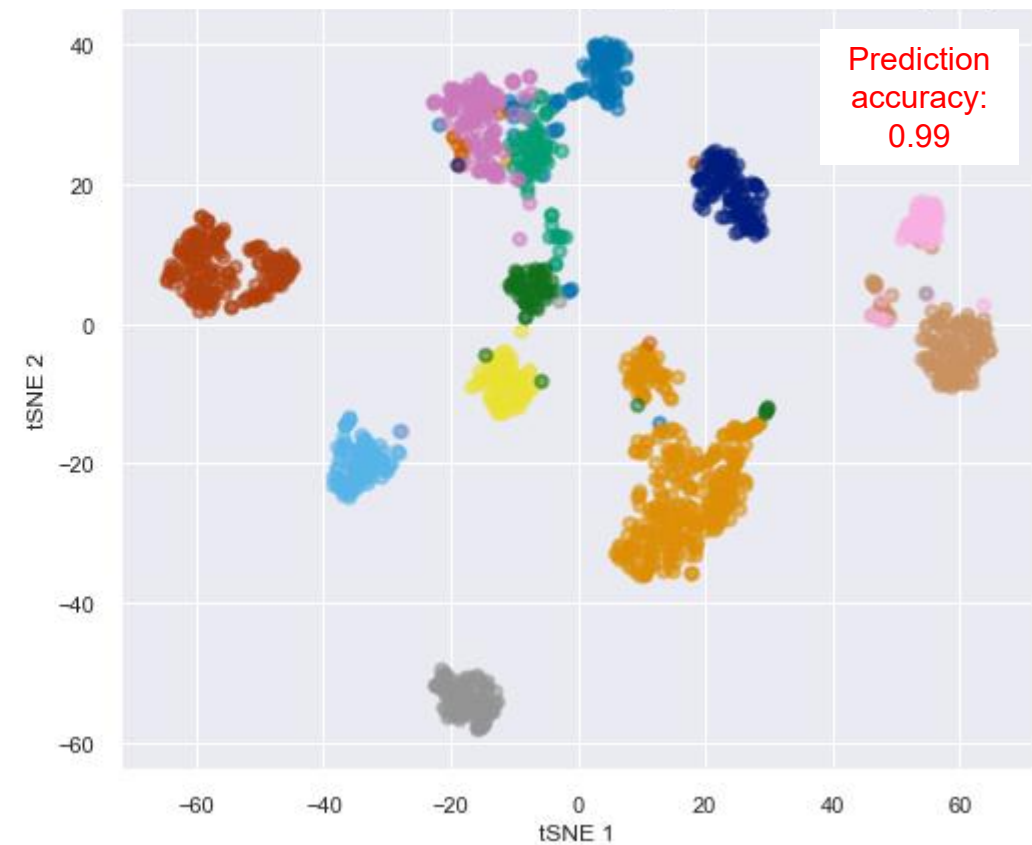
1. Primary cancer diagnosis from gene expression
2. Breast cancer patients' stratification based on gene expression

True values, tumor classification task



Validation set (colors correspond to true cancer types)

Best solution: Elastic net logistic regression



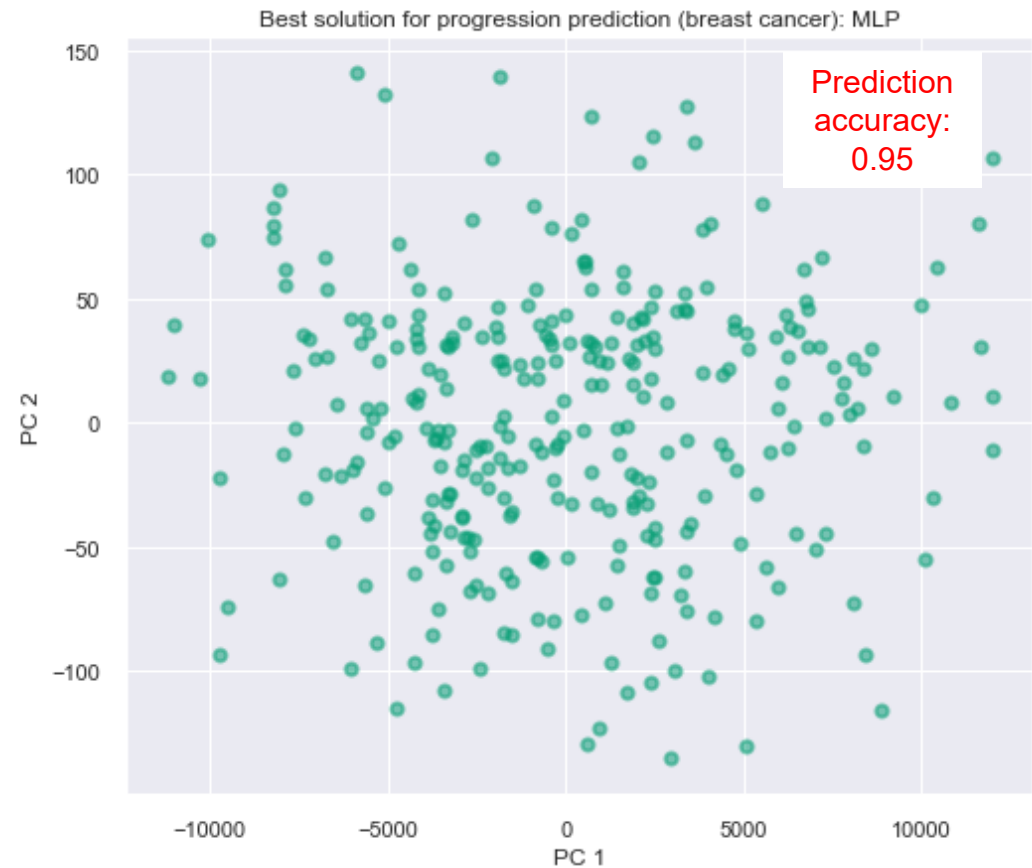
Elastic net on the validation set (colors correspond to predictions)

Hands-on:

1. Primary cancer diagnosis from gene expression
2. Breast cancer patients' stratification based on gene expression



Validation set (true colors)

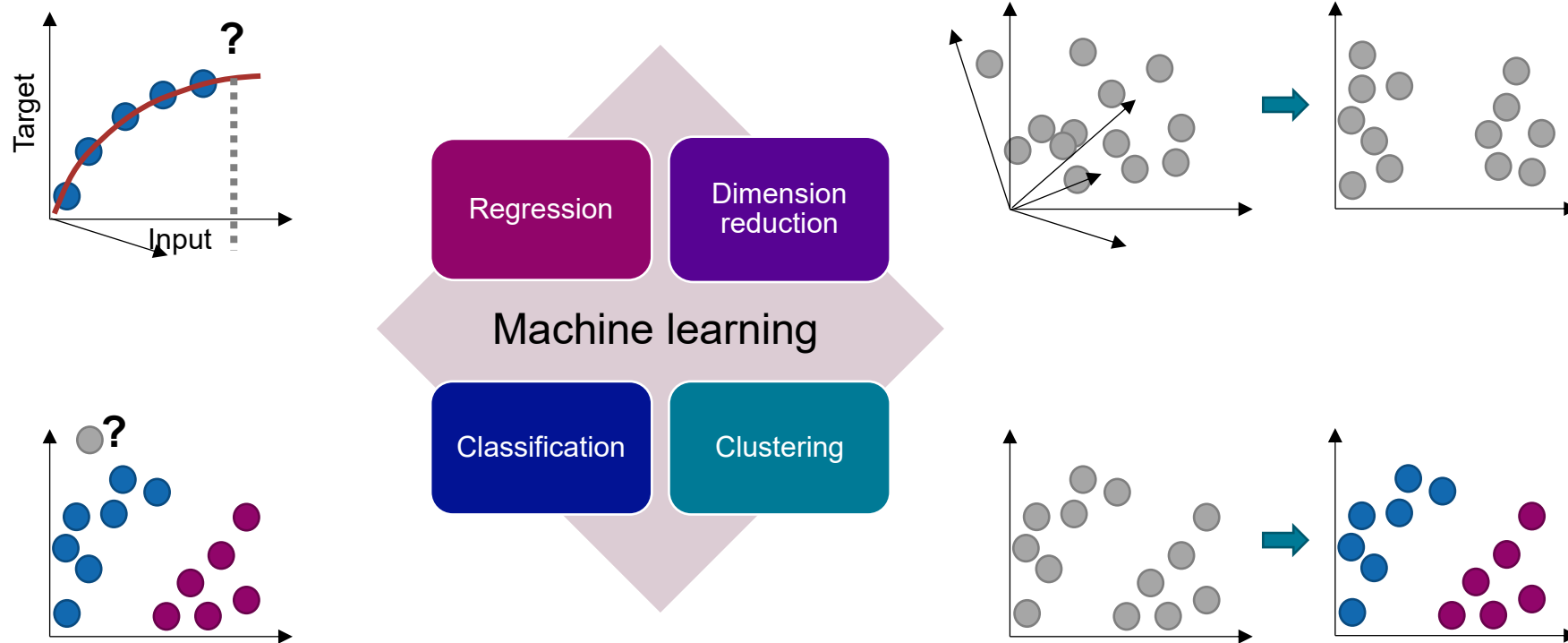


multi-layer perceptron on the validation set

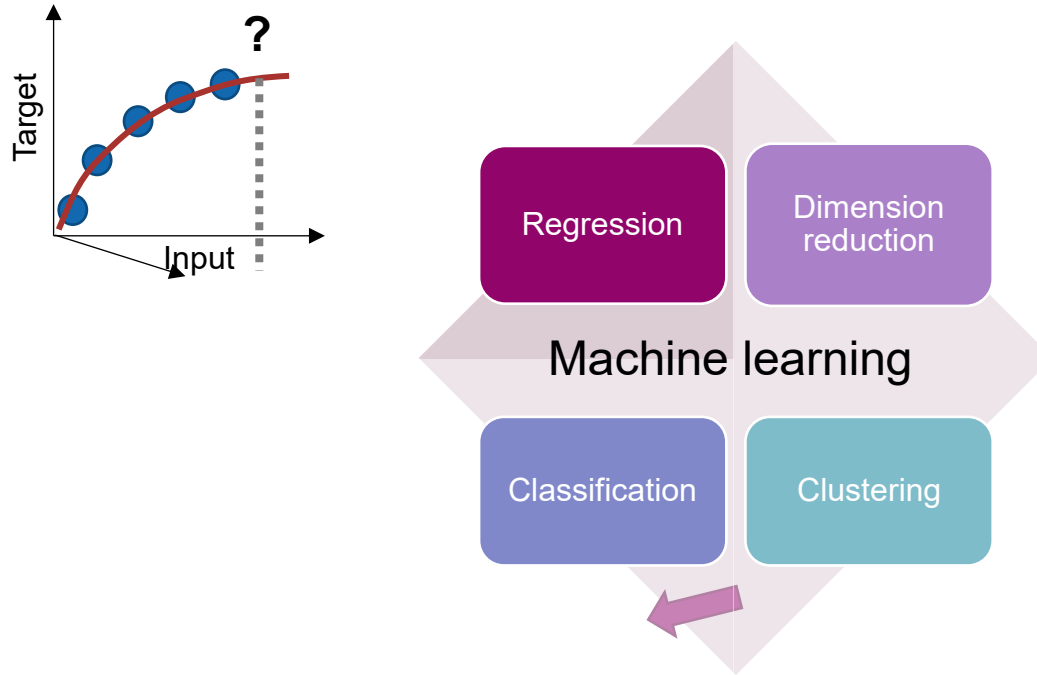
Take home message: Classification

- Linear and non-linear models can provide similar prediction accuracy (TASK1)
- Classification on imbalanced groups with low information content may fail (TASK2)
 - Study your data first
 - Check data summary
 - Visualize your data
 - Use the right evaluation metrics (e.g., precision and recall)
 - Consider redesigning your task

Map of classical machine learning methods

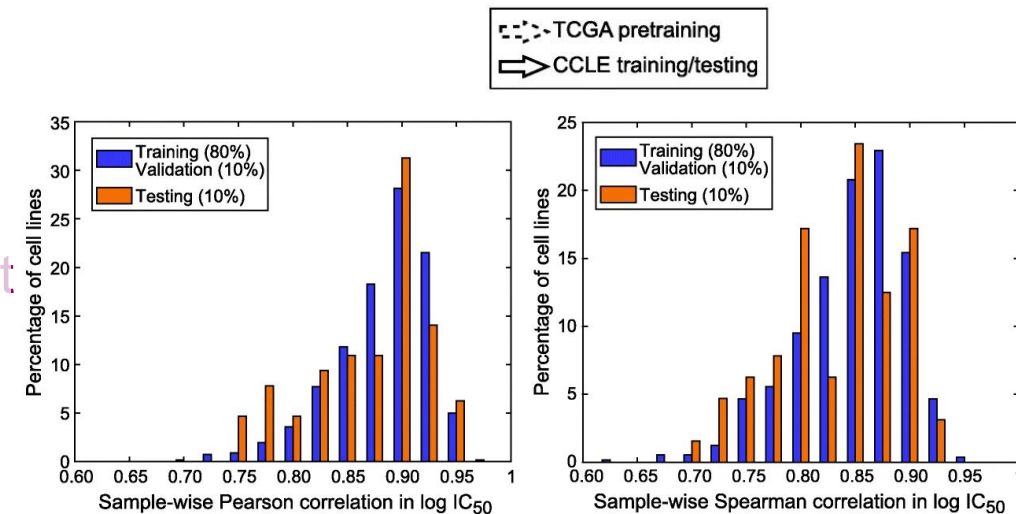
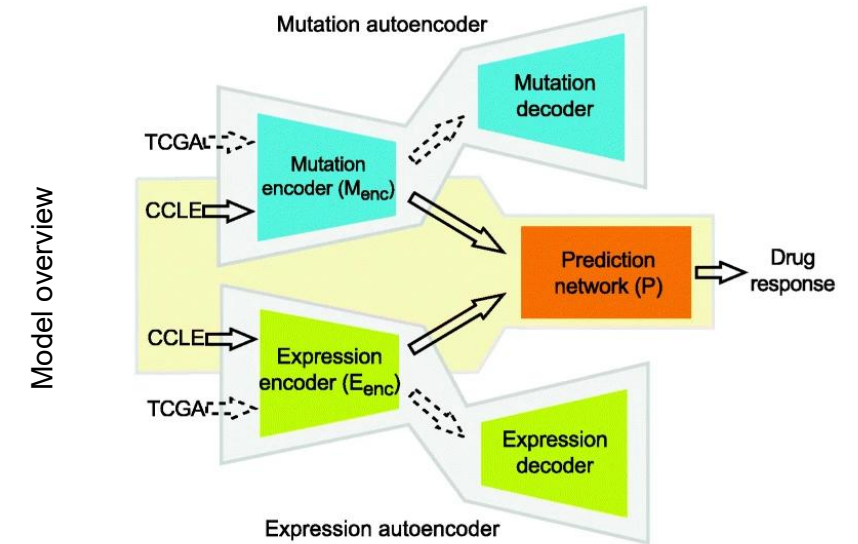


Map of classical machine learning methods



Regression: Biological examples

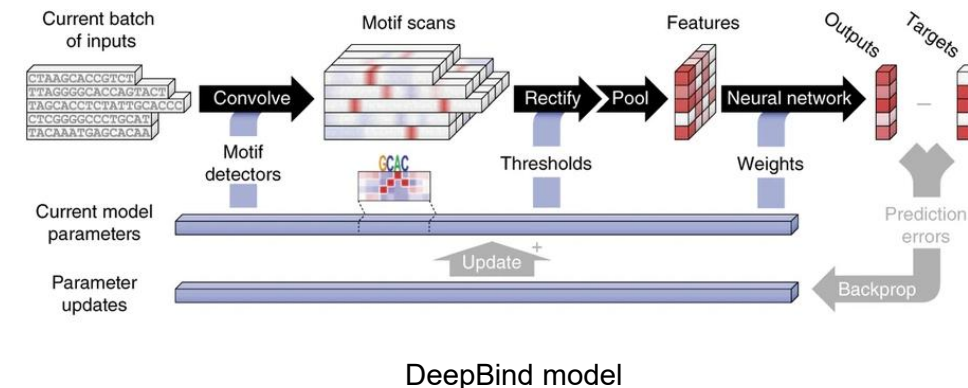
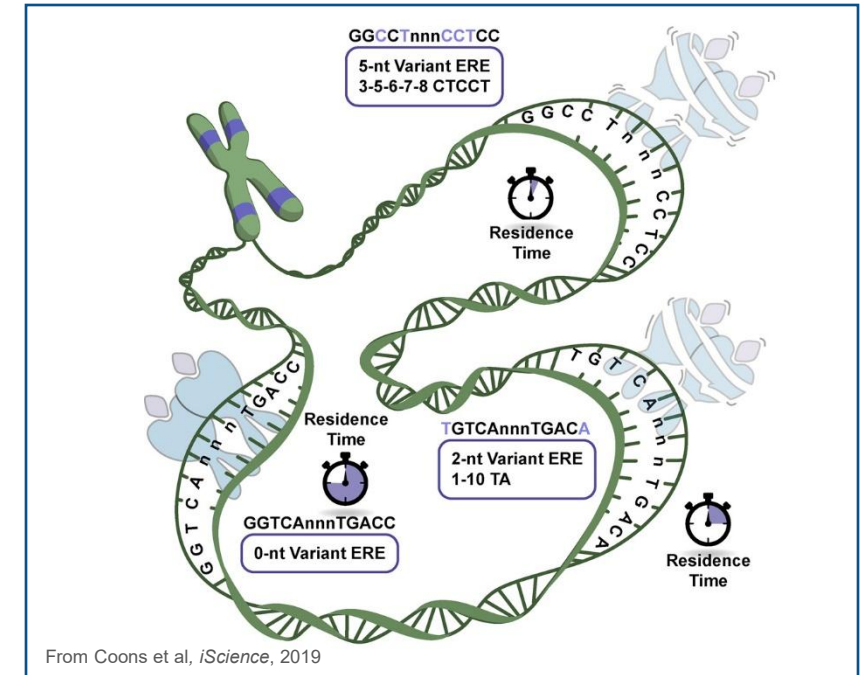
- Prediction of treatment efficiency / drug response
 - **Predicting drug response of tumors from integrated genomic profiles by deep neural networks.** Chiu et al., *BMC Med. Genomic*, 2019
- Prediction of molecular/cellular properties (e.g., protein-DNA binding affinities)
 - **Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning.** Alipanahi et al., *Nature Biotech.* 2015



Sample-wise Pearson and Spearman correlation between imputed and predicted IC_{50} data of CCLE samples

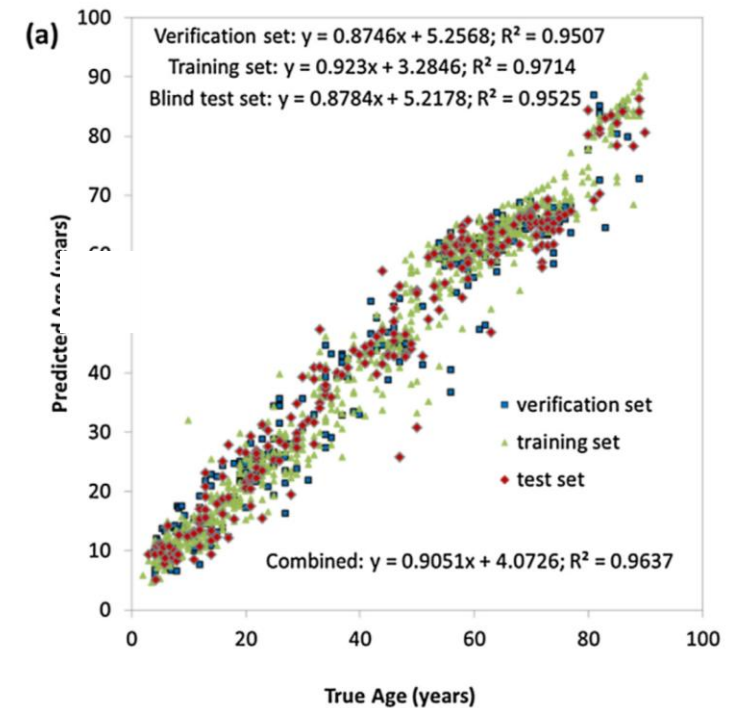
Regression: Biological examples

- Prediction of treatment efficiency / drug response
 - **Predicting drug response of tumors from integrated genomic profiles by deep neural networks.** Chiu et al., *BMC Med. Genomic*, 2019
- Prediction of molecular/cellular properties (e.g., protein-DNA binding affinities)
 - **Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning.** Alipanahi et al., *Nature Biotech.* 2015



Regression: Biological examples

- Age prediction from DNA methylation (e.g., for forensics)
 - **DNA methylation-based forensic age prediction using artificial neural networks and next generation sequencing.** Vidaki et al. *Forensic Sci Int Genet.* 2017



Hands-on: Prediction of protein concentration based on mRNA data (Breast cancer samples)

1. Concentration of HER2 (coded by the *ERBB2* gene)
2. Concentration of p53 (coded by the *TP53* gene)

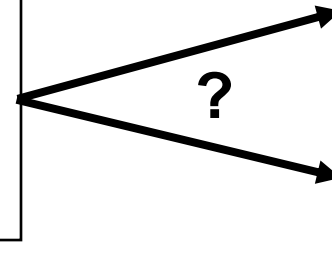
<https://github.com/BoevaLab/Teaching>

TASK 1:



	<i>Gene1</i>	<i>Gene2</i>	...	<i>Gene_p</i>
Z_1	X_{11}	X_{12}	...	X_{1p}
Z_2	X_{21}	X_{22}	...	X_{2p}
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
Z_n	X_{n1}	X_{n2}	...	X_{np}

+ clinical stage + age



High level of
HER2 protein

Low level of
HER2 protein

Treatment with HER2
inhibitors: Herceptin
(trastuzumab) or
Tykerb (lapatinib)

The **HER2** protein is coded by the **ERBB2** gene,
frequently amplified in human breast cancer

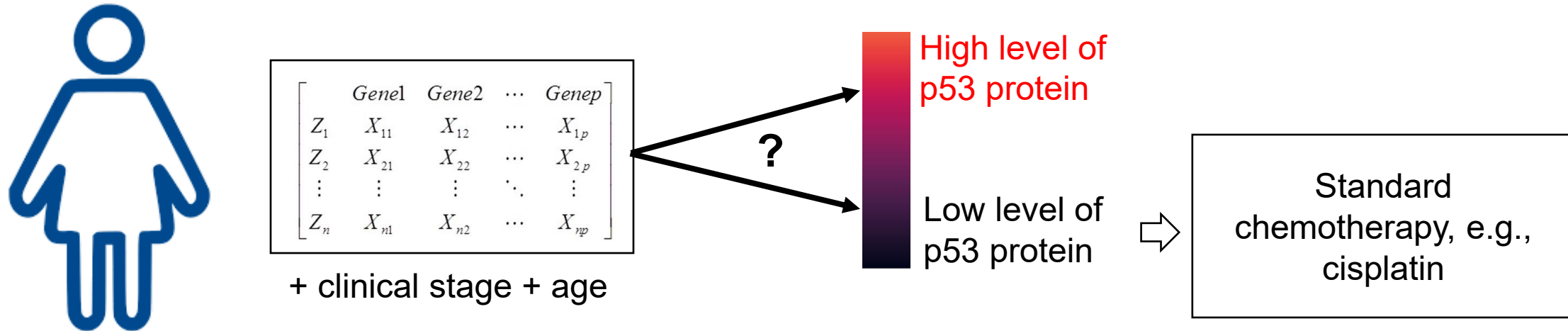
Target variable (y): Reverse Phase Protein Array (*RPPA*) value of HER2 presence in tumor cells

Hands-on: Prediction of protein concentration based on mRNA data (Breast cancer samples)

1. Concentration of HER2 (coded by the *ERBB2* gene)
2. Concentration of p53 (coded by the *TP53* gene)

<https://github.com/BoevaLab/Teaching>

TASK 2:

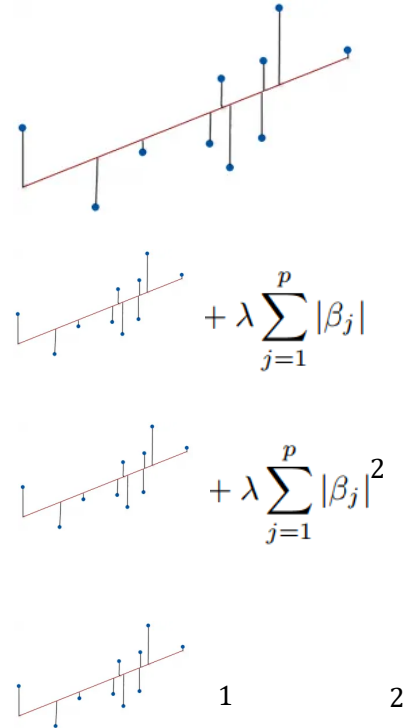


The **p53** protein is coded by the **TP53 gene**, frequently deleted, mutated or repressed in human cancers

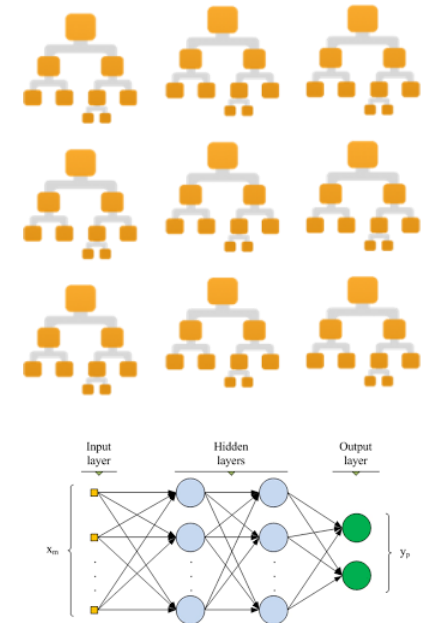
Target variable (y): Reverse Phase Protein Array (*RPPA*) value of p53 presence in tumor cells

Classic regression models

- Ordinary Least Squares
- Lasso (L1 penalty on model coefficients)
- Ridge (L2 penalty on model coefficients)
- Elastic Net (L1 and L2 penalty on model coefficients)

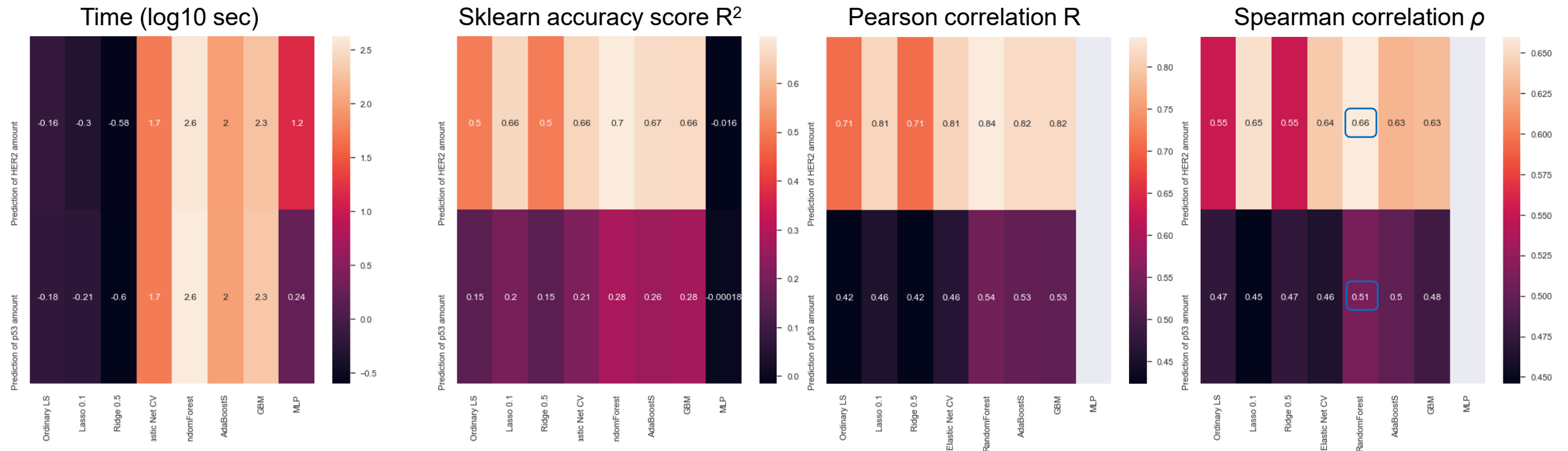


- Random Forest
- AdaBoost
- Gradient Tree Boosting (gradient boosting machine, GBM)
- Multi-layer perceptron (MLP)



Hands-on: Prediction of protein concentration based on mRNA data (Breast cancer samples)

1. Concentration of HER2 (coded by the *ERBB2* gene)
2. Concentration of p53 (coded by the *TP53* gene)



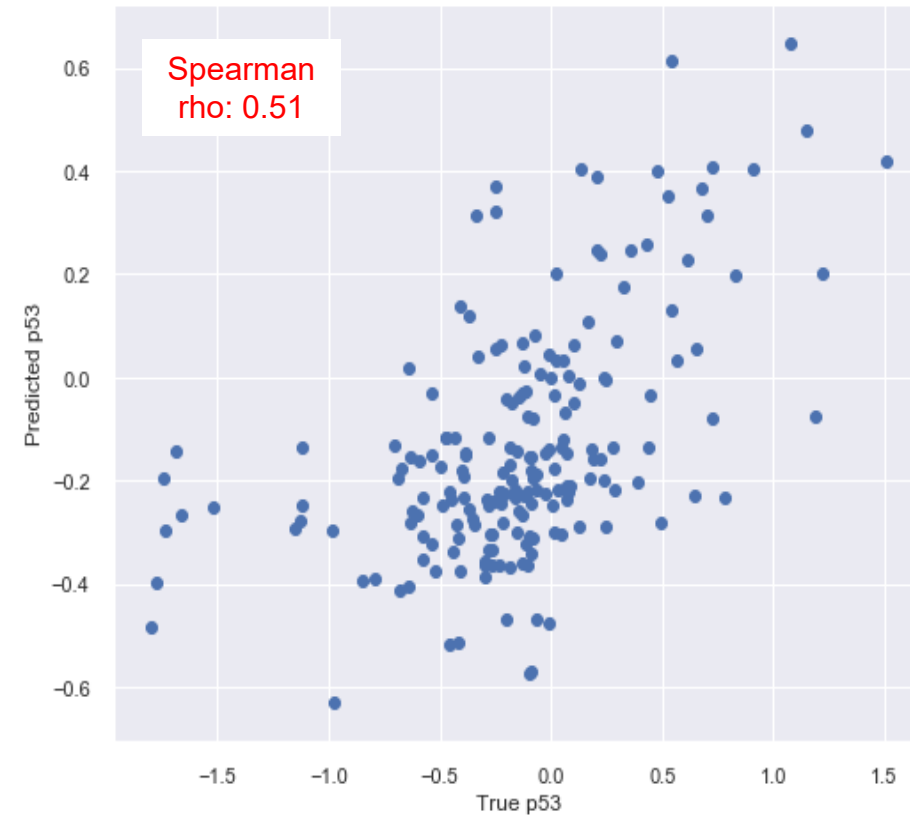
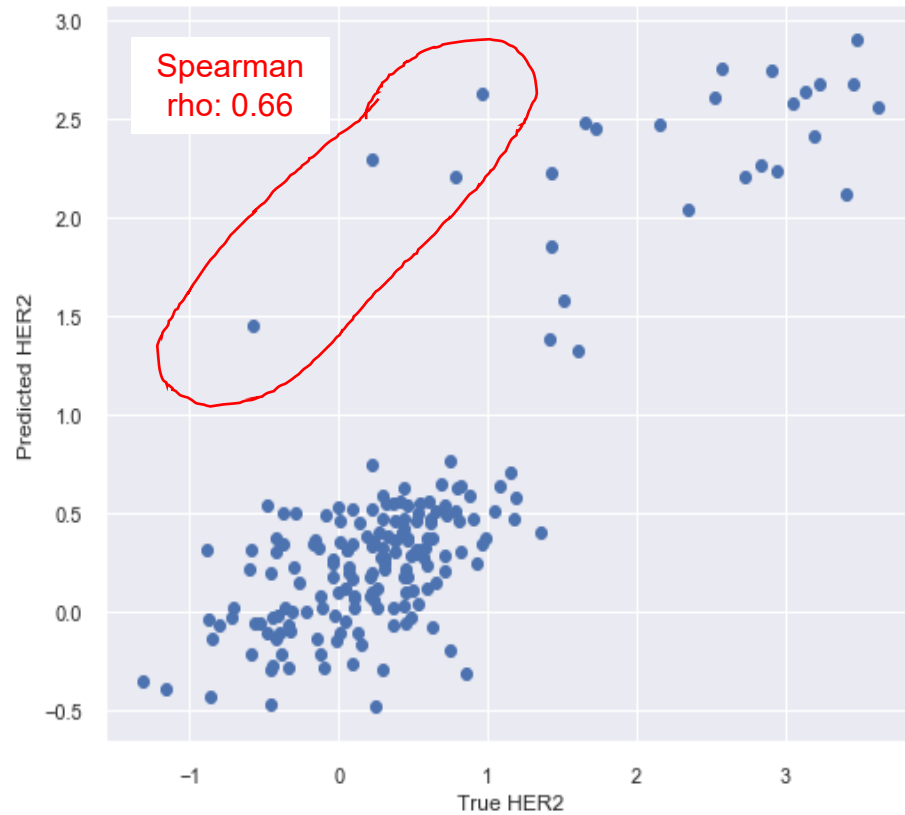
Coefficient of determination $R^2 = \left(1 - \frac{RSS}{\sum(y_i - \bar{y})^2}\right)$

$Spearman \rho (HER2, ERBB2) = 0.63$

$Spearman \rho (p53, TP53) = 0.27$

Hands-on: Prediction of protein concentration based on mRNA data (Breast cancer samples)

1. Concentration of HER2 (coded by the *ERBB2* gene)
2. Concentration of p53 (coded by the *TP53* gene)

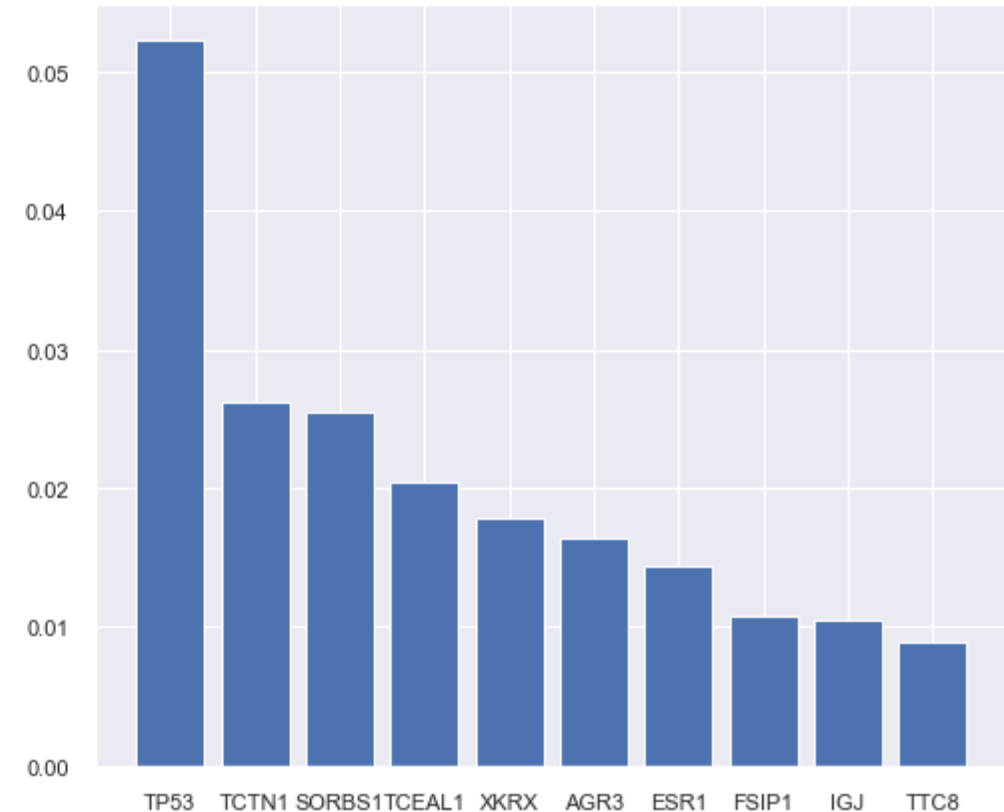
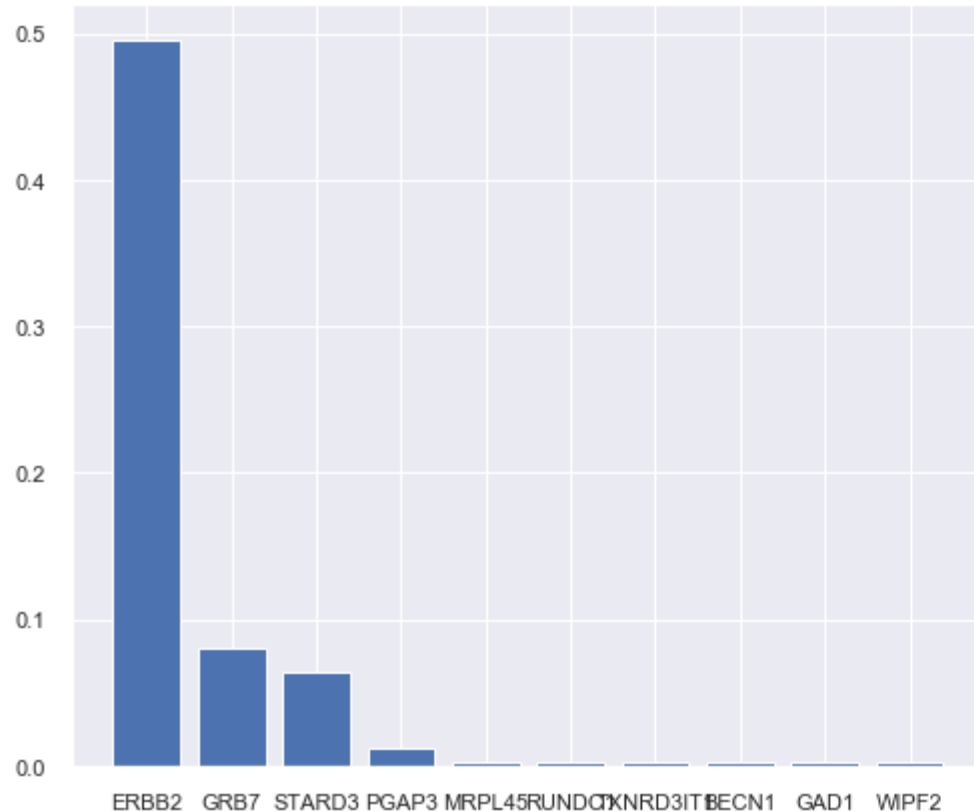


Random Forest predictions

Hands-on: Prediction of protein concentration based on mRNA data (Breast cancer samples)

1. Concentration of HER2 (coded by the *ERBB2* gene)
2. Concentration of p53 (coded by the *TP53* gene)

FEATURE IMPORTANCE:

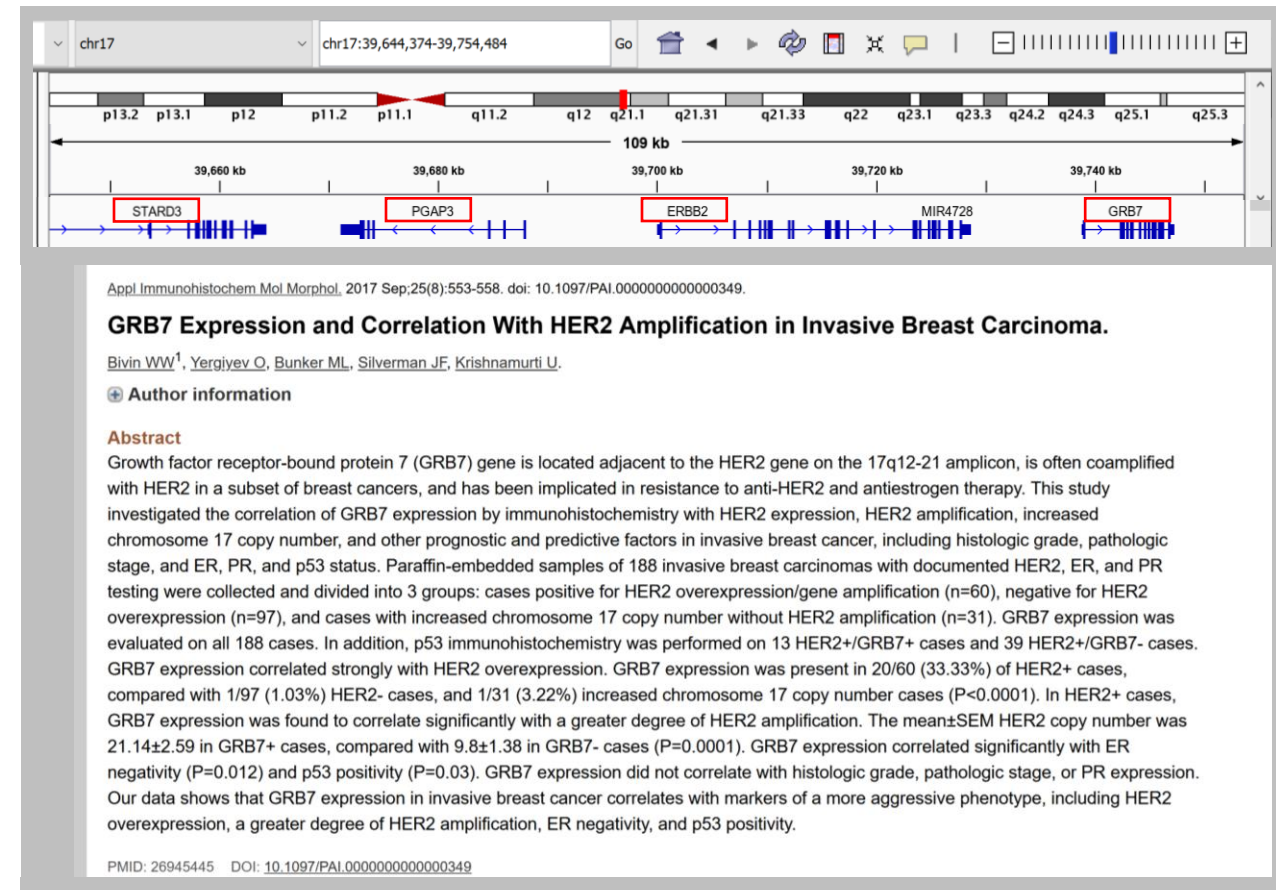
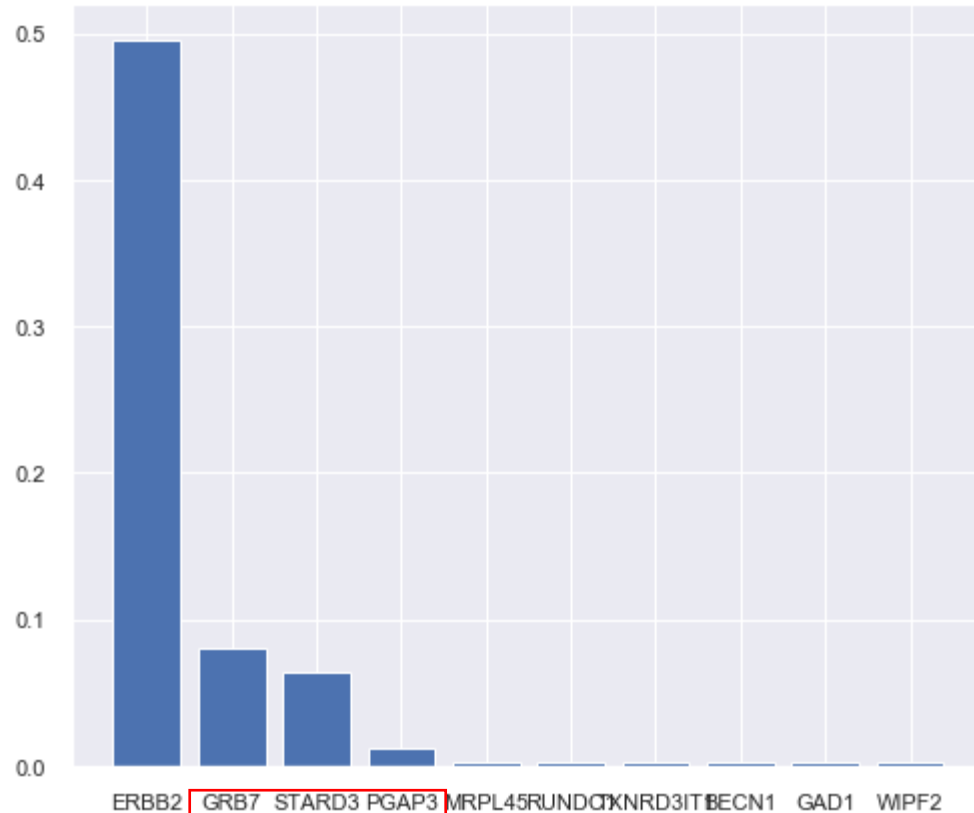


Random Forest predictions

Hands-on: Prediction of protein concentration based on mRNA data (Breast cancer samples)

1. Concentration of HER2 (coded by the *ERBB2* gene)
2. Concentration of p53 (coded by the *TP53* gene)

FEATURE IMPORTANCE:



Hands-on: Prediction of protein concentration based on mRNA data (Breast cancer samples)

1. Concentration of HER2 (coded by the *ERBB2* gene)
2. Concentration of p53 (coded by the *TP53* gene)

FEATURE IMPORTANCE:



Review

The p53 Pathway in Glioblastoma

Ying Zhang ^{1,†}, Collin Dube ^{1,†}, Myron Gibert Jr. ^{1,†}, Nichola Cruickshanks ¹, Baomin Maeve Coughlan ¹, Yanzi Yang ¹, Initha Setiady ¹, Ciana Deveau ¹, Karim Saoud ¹, Cassandra Grello ¹, Madison Oxford ¹, Fang Yuan ¹ and Roger Abounader ^{1,2,3,*}

“ARF deletion [from p53 pathway] is correlated with overexpression of tectonic family member 1 (TCTN1), a protein involved in a diverse range of cellular processes, including promotion of GBM cell proliferation”.

www.impactjournals.com/oncotarget/ Oncotarget, 2017, Vol. 8, (No. 6), pp: 9108-9122

Research Paper

SORBS1 suppresses tumor metastasis and improves the sensitivity of cancer to chemotherapy drug

Lele Song ^{1,2}, Renxu Chang ^{1,2}, Cheng Dai ^{1,2}, Yanjun Wu ^{1,2}, Jingyu Guo ^{1,2}, Meiyan Qi ¹, Wu Zhou ³, Lixing Zhan ¹

“Silencing of SORBS1 [...] attenuates chemical drug sensitivity especially that to cisplatin, by inhibition of p53 in **breast cancer cells**.”

Meng et al. Journal of Translational Medicine 2014, 12:288
http://www.translational-medicine.com/content/12/1/288

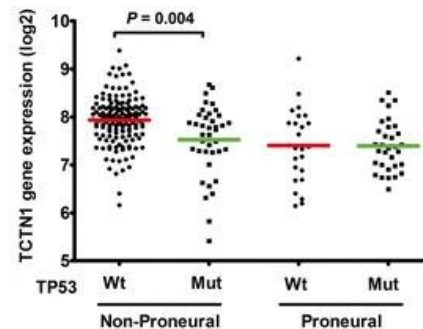


RESEARCH

Open Access

Expression and prognostic significance of TCTN1 in human glioblastoma

Delong Meng ^{1†}, Yuanyuan Chen ^{1†}, Yingjie Zhao ¹, Jingkun Wang ¹, Dapeng Yun ¹, Song Yang ², Juxiang Chen ³, Hongyan Chen ¹ and Daru Lu ^{1*}



0.05

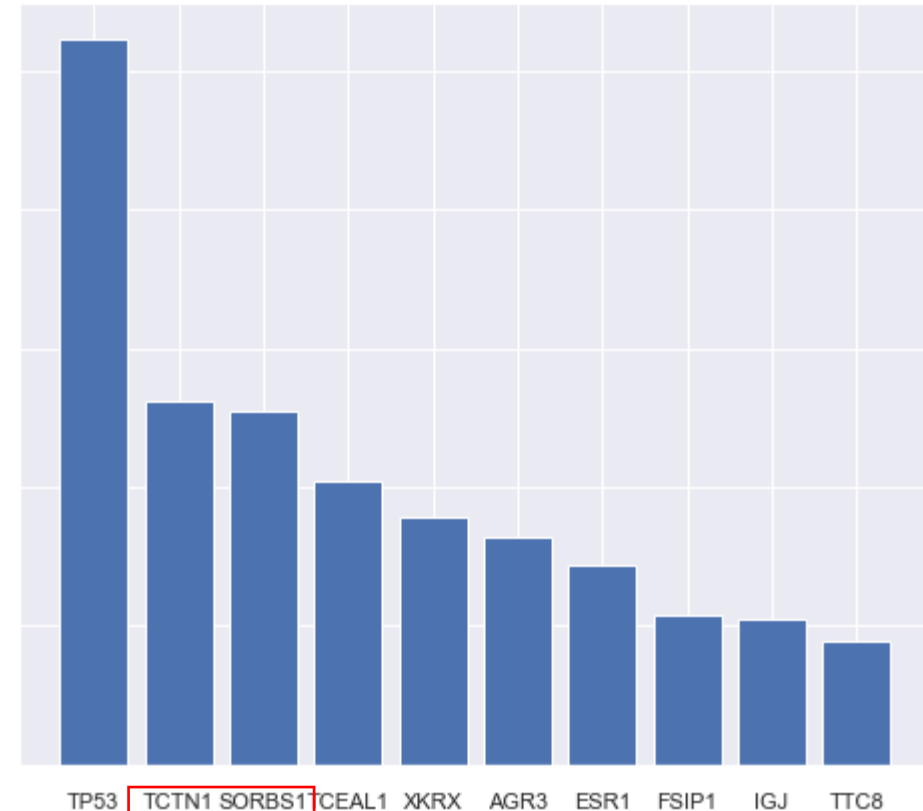
0.04

0.03

0.02

0.01

0.00

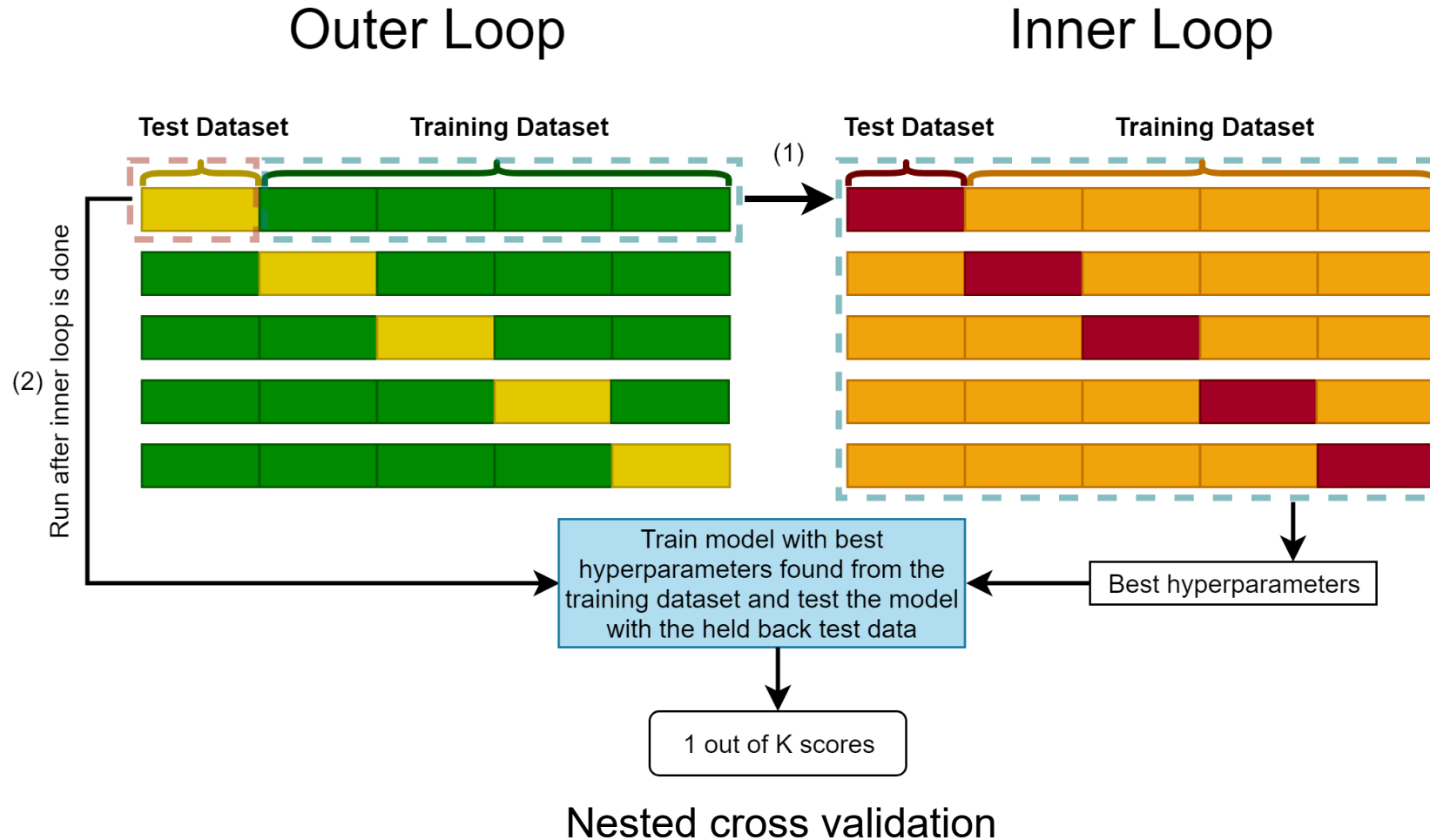


Take home message: Regression

- Regularized methods generally work better
 - Regularization may prevent over-fitting and select “important” features
- Regularized linear methods may provide accuracy similar to non-linear methods
- Neural networks do not always win
- Checking the feature importance may provide insights into biological mechanisms

Selection of hyperparameters via cross validation

What we did not do today, but in real life we should do it:



Take home message

- Classification and regression are extremely widely used in biology and medicine to automatize decisions of clinicians (diagnosis, choice of treatment) and predict treatment response and side effects
- The accuracy of predictions depends a lot on the information present in the data rather than on the ML method used
 - In our hands-on exercises the difference in accuracy between linear and non-linear methods varied between 0.5% and 15%
- There is no method that works the best in any situation
- Cross validation should be always applied to select the best hyperparameters
- In real life, one should compare a model built on omics data (+ clinical) with a model built using clinical variables only

Thank you for your attention!

Professor Valentina Boeva
valentina.boeva@inf.ethz.ch

ETH Zurich
Dept of Computer Science
CAB G32.2
Universitatstrasse 6
8092 Zurich, Switzerland

www.boevalab.com

