Introduction & 'Simple' plots and recap of linear algebra (in python)

Sophie de Buyl

2020 – 21

VRIJE
UNIVERSITEIT
BRUSSEL

Our goal in this course is to make sense out of big datasets.

We will use techniques of machine learning:

- The first part of the course is devoted to unsupervised learning which aims at finding structure in unlabeled datasets $\Rightarrow$ dimensional reduction techniques and clustering.
- The second part is devoted to supervised learning which mostly aims at model building to make predictions or to gain understanding of the system.

Lectures based on “Introduction to statistical learning in R” which is available online at: `https://www.statlearning.com/` (click on "Download the first edition".)

Example of 'Unsupervised' dataset

	0	1	2	3	4	5	6	7	8	9	...	6820	6821	6822
0	0.728671	1.607220	1.325688	1.355688	-0.604845	-0.220654	0.898137	-0.868741	-1.058612	-1.059174	...	-1.030663	-0.358518	-0.238245
1	1.596418	1.753544	0.441686	0.654119	0.911898	1.648748	1.849697	2.226625	-0.095860	-0.477977	...	-0.215657	-0.625720	-0.489938
2	2.190290	-0.016217	-0.349092	0.266465	-1.311310	-0.019322	0.191185	1.988627	1.007979	0.716019	...	0.452274	-0.251651	-0.930304
3	0.682995	-0.375502	1.628079	-0.444299	1.244434	-0.019322	0.408709	0.798057	0.045135	0.119051	...	-1.313667	-0.456479	-0.409013
4	1.151170	-0.581759	0.965145	1.138767	0.361351	-0.033703	0.177590	0.396239	0.550041	2.310550	...	0.718297	-1.048700	-0.728079
5	0.751508	-0.002910	-0.186266	-0.121224	-0.480174	-1.572457	-1.793718	-1.672376	-1.810100	-0.415077	...	-0.577858	-0.358518	-0.930304
6	-1.852001	0.037011	0.348733	-1.201217	0.101621	-1.054745	0.218376	0.530178	-0.682868	-0.163723	...	0.305113	-0.554441	-0.292172
7	-0.390382	-1.120687	-1.000396	0.377234	-0.043828	0.987340	-0.162291	-0.719920	-0.729836	-0.415077	...	-0.147693	-0.509913	-0.642695
8	1.094075	0.037011	2.721339	-0.988910	-0.064606	-2.032645	-2.555051	-1.464026	-0.659384	-0.210852	...	0.191911	0.140194	0.103290
9	1.779209	2.019733	0.697646	0.395695	-1.145083	1.016102	1.577899	0.589707	1.360239	1.768565	...	0.429634	-0.883947	-0.633707
10	0.660157	0.875343	-0.791048	-0.656605	-0.023049	-0.134369	1.523518	1.333813	2.323083	1.312986	...	-1.098584	-0.536629	-0.516866
11	-0.984164	-0.042831	0.395255	-0.259685	-0.874964	0.412105	2.502375	-0.392513	0.679203	0.763148	...	0.157951	0.630000	0.651544
12	-0.024977	-1.453358	-0.069962	0.303388	-0.646401	-0.738366	0.381519	0.321829	0.326943	-0.980625	...	2.580458	2.758432	2.494037
13	1.665020	0.037011	0.418516	-0.407376	0.039286	-0.479510	0.816566	-0.005578	0.937527	0.150471	...	0.282472	-0.429763	-0.364074
14	-0.778624	-1.852564	-0.023440	-1.801212	0.226292	-0.939698	-0.162291	0.173007	-1.387388	0.935954	...	0.973000	0.140194	0.390899
15	-0.436057	-2.557828	-1.581917	-1.219678	-0.210055	0.153249	-0.216671	0.113479	-0.870740	-1.263399	...	0.871119	-0.055729	-0.265209
16	-1.098353	-0.654946	-1.488873	-2.105826	0.413297	-0.163131	-1.113957	-0.213928	1.148883	-0.792109	...	1.935211	1.876780	1.748052

Each line corresponds to a cell, and rows correspond to gene expression within this cell (6822 genes measured per cell). How to make sense / extract information out of high dimensional data sets?

Example of 'Supervised' dataset

In supervised learning we would also have 'outputs':

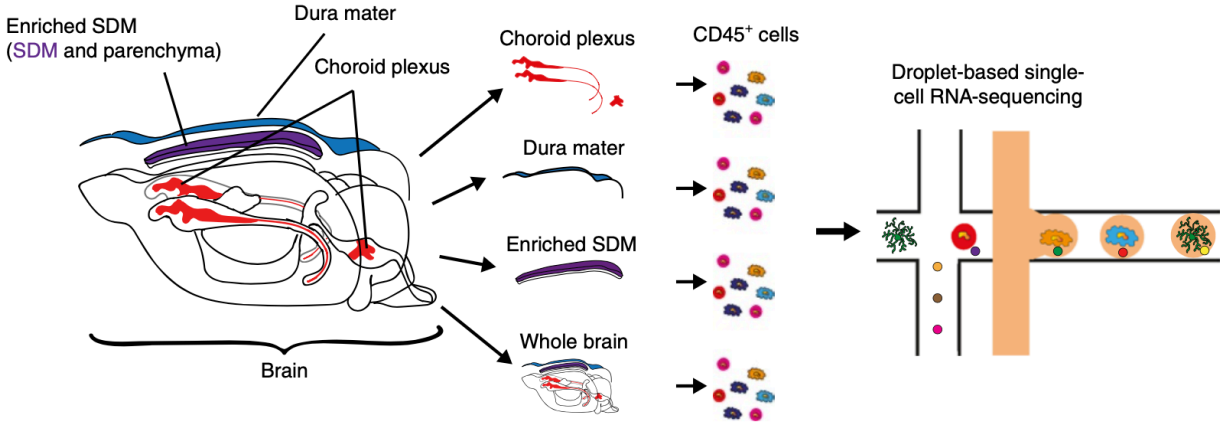
	0	1	2	3	4	5	6	7	...	6820	6821	6822	outputs
0	0.728671	1.607220	1.325688	1.355688	-0.604845	-0.220654	0.898137	-0.868741	...	-1.030663	-0.358518	-0.238245	Microglia
1	1.596418	1.753544	0.441686	0.654119	0.911898	1.648748	1.849697	2.226625	...	-0.215657	-0.625720	-0.489938	Microglia
2	2.190290	-0.016217	-0.349092	0.266465	-1.311310	-0.019322	0.191185	1.988627	...	0.452274	-0.251651	-0.930304	Microglia
3	0.682995	-0.375502	1.628079	-0.444299	1.244434	-0.019322	0.408709	0.798057	...	-1.313667	-0.456479	-0.409013	T-cell
4	1.151170	-0.581759	0.965145	1.138767	0.361351	-0.033703	0.177590	0.396239	...	0.718297	-1.048700	-0.728079	T-cell
5	0.751508	-0.002910	-0.186266	-0.121224	-0.480174	-1.572457	-1.793718	-1.672376	...	-0.577858	-0.358518	-0.930304	T-cell
6	-1.852001	0.037011	0.348733	-1.201217	0.101621	-1.054745	0.218376	0.530178	...	0.305113	-0.554441	-0.292172	T-cell
7	-0.390382	-1.120687	-1.000396	0.377234	-0.043828	0.987340	-0.162291	-0.719920	...	-0.147693	-0.509913	-0.642695	Monocytes
8	1.094075	0.037011	2.721339	-0.988910	-0.064606	-2.032645	-2.555051	-1.464026	...	0.191911	0.140194	0.103290	Monocytes
9	1.779209	2.019733	0.697646	0.395695	-1.145083	1.016102	1.577899	0.589707	...	0.429634	-0.883947	-0.633707	BAM
10	0.660157	0.875343	-0.791048	-0.656605	-0.023049	-0.134369	1.523518	1.333813	...	-1.098584	-0.536629	-0.516866	BAM
11	-0.984164	-0.042831	0.395255	-0.259685	-0.874964	0.412105	2.502375	-0.392513	...	0.157951	0.630000	0.651544	BAM
12	-0.024977	-1.453358	-0.069962	0.303388	-0.646401	-0.738366	0.381519	0.321829	...	2.580458	2.758432	2.494037	Neurophils
13	1.665020	0.037011	0.418516	-0.407376	0.039286	-0.479510	0.816566	-0.005578	...	0.282472	-0.429763	-0.364074	Neutrophils
14	-0.778624	-1.852564	-0.023440	-1.801212	0.226292	-0.939698	-0.162291	0.173007	...	0.973000	0.140194	0.390899	B-cell
15	-0.436057	-2.557828	-1.581917	-1.219678	-0.210055	0.153249	-0.216671	0.113479	...	0.871119	-0.055729	-0.265209	B-cell
16	-1.098353	-0.654946	-1.488873	-2.105826	0.413297	-0.163131	-1.113957	-0.213928	...	1.935211	1.876780	1.748052	B-cell

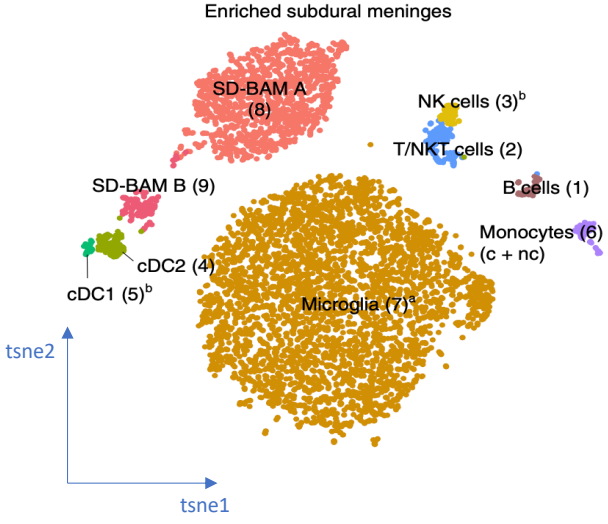
The goal of supervised learning would be typically to predict the output for a new sample for which we only have the features measured, but we do not know which type of cell it is. For instance, we could wonder whether it is a cancer cell or not.

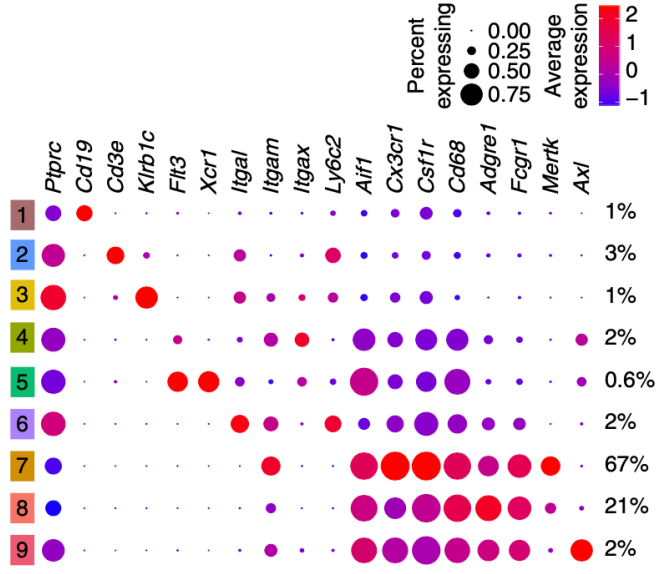
Machine learning is everywhere

- Biology/bioinformatics :
  - protein structure prediction
  - gene prediction (finding location of protein-encoding gene in a DNA sequence)
  - identifying multiple genes involved in rare diseases
  - identification of transcription factors binding sites,..
  - new antibiotic found by using neural networks **`https://www.quantamagazine.org/machine-learning-takes-on-antibiotic-resistance-20200309/`**
  - combining fluorescent labels can be tricky, machine learning can help to visualize multiple structures in real time simultaneously: **`https://www.quantamagazine.org/greg-johnsons-artificial-intelligence-sees-inside-living-cells-20190724/`**
- Medical diagnosis
- Timeseries forecasting
- Image/speech recognition
- Search engines
- Traduction tools, spelling/grammar correctors,...
- ...

Example of single cell gene expression dataset

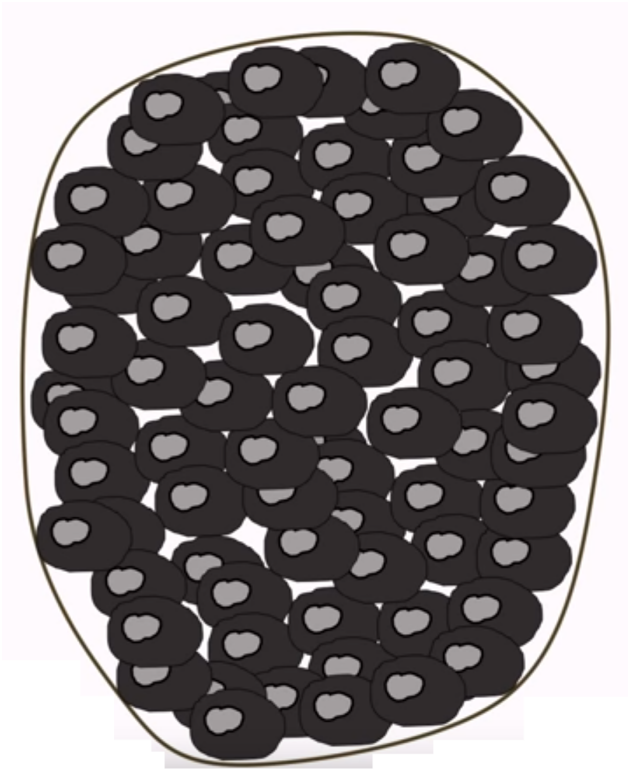




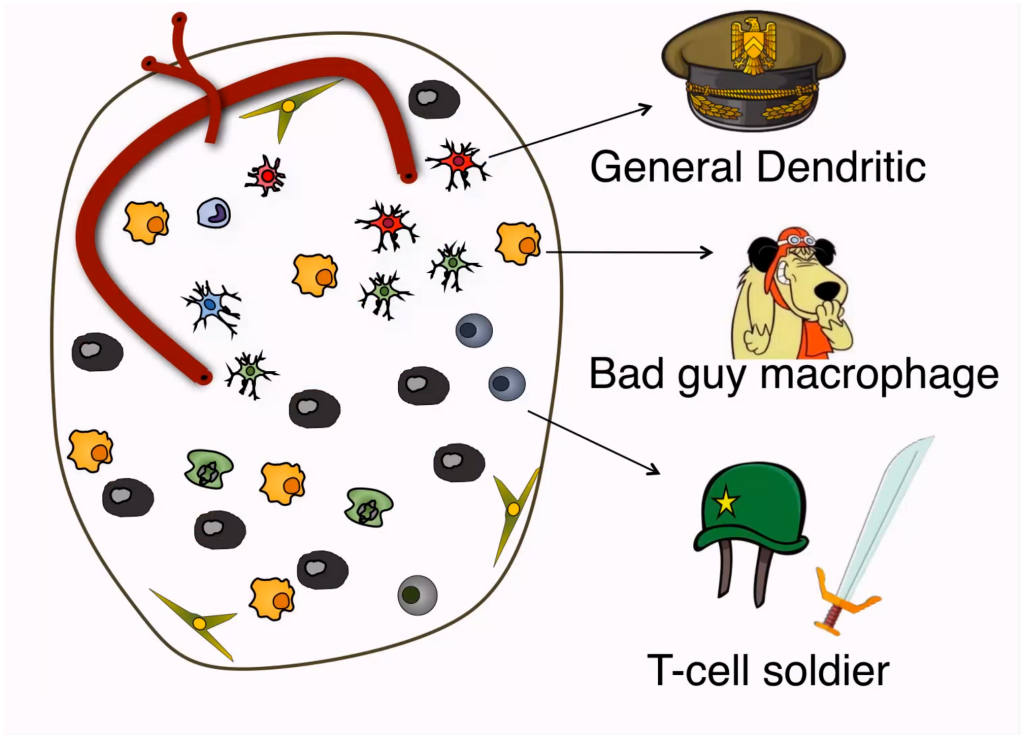


www.brainimmuneatlas.org

Another example of single cell gene expression dataset



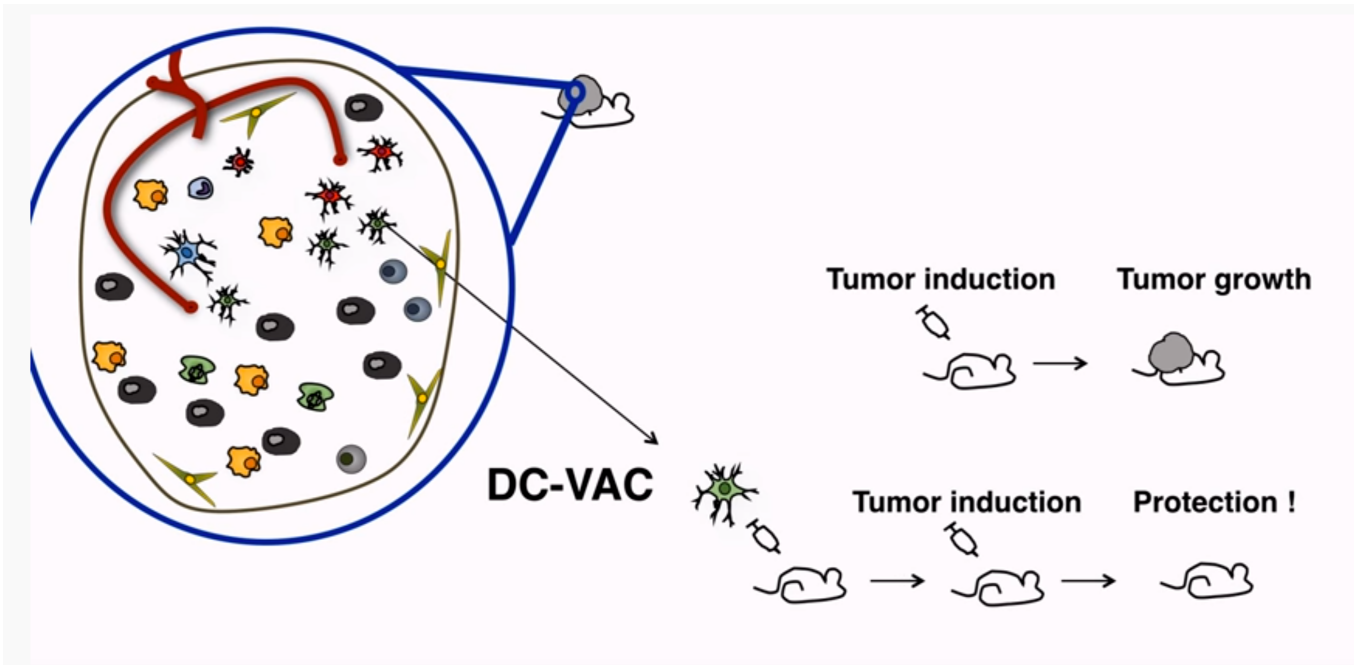
Ceci n'est pas une tumeur



Ceci est une tumeur

From Damya Laoui Science & cocktails talk available at
`https://www.scienceandcocktails.org/en/video/new-ways-to-eradicate-cancer`

Another example of single cell gene expression dataset



The identification of specific immune cells in a tumor led to the development of cancer vaccine.

1. Introduction

2. (Co-)variance of a dataset

3. Bar, box and violin plots

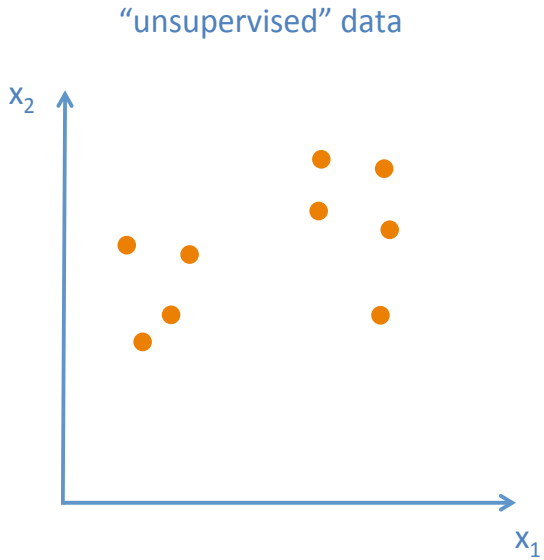
A dataset is a matrix

Our dataset in encoded in a matrix $\boldsymbol{X}$ of size $n \times \boldsymbol{p}$.

The elements of this matrix are denoted by $\boldsymbol{x}^{\ell}_{\boldsymbol{i}}$ and carry two types of indices:

- a 'sample' index ℓ taking its values in $(\mathbf{1}, ..., n)$ (this is the line index of the matrix).
- a 'feature' index $\boldsymbol{i}$ taking its values in $(\mathbf{1}, ..., \boldsymbol{p})$ (this is the column index).

First characterization of the dataset: means and (co-)variances



Before doing any unsupervised learning technique, one can start by characterizing the data set with the following quantities

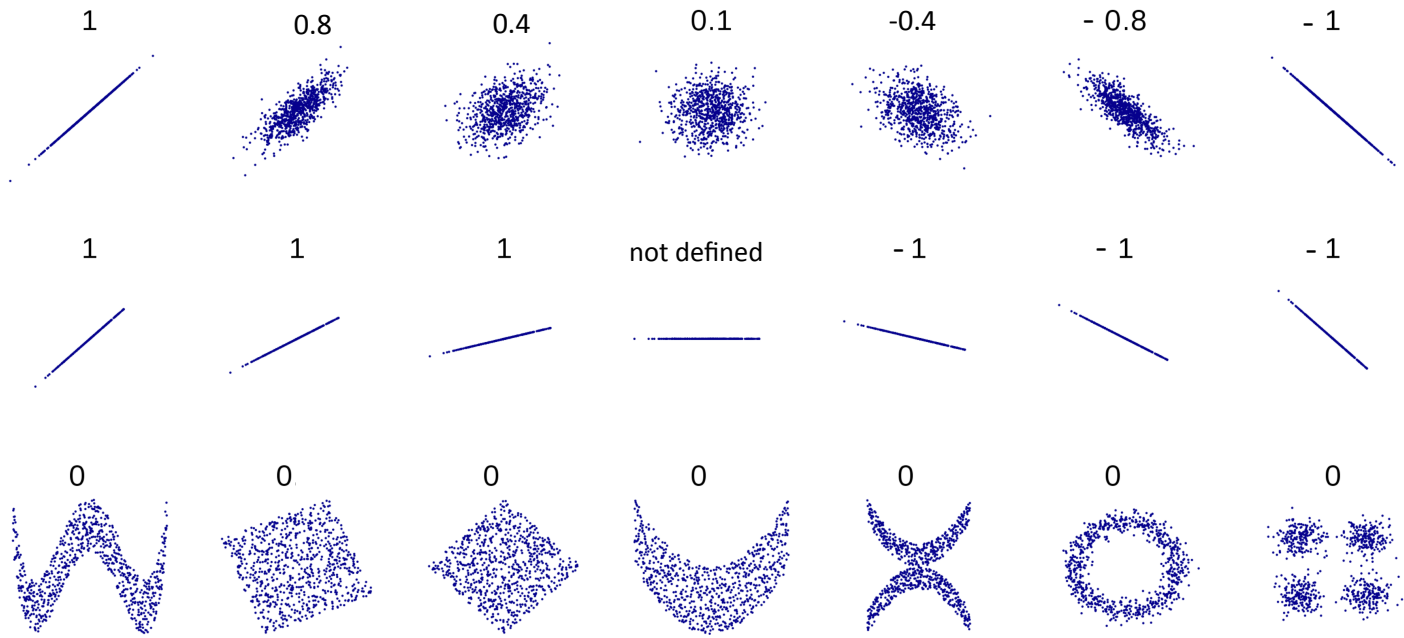
means : $\mu_i = \frac{1}{n} \sum_{\ell=1}^{n} x^\ell{}_i$
variances : $\sigma_i^2 = \frac{1}{n} \sum_{\ell=1}^{n} (x^\ell{}_i - \mu_i)^2$
covariances : $\sigma_{ij} = \frac{1}{n} \sum_{\ell=1}^{n} (x^\ell{}_i - \mu_i)(x^\ell{}_j - \mu_j)$

The covariance measures how two variables vary together. For example, height and weight of giraffes have positive covariance because when one is big the other tends also to be big.Two independent variable will have zero covariance.

Do not confuse correlation and covariance: the correlation is the normalized covariance $\text{corr}(x_1, x_2) = \sigma_{12}/(\sigma_1 \sigma_2)$ and is a number between -1 and 1.

Data point labelled by $\ell, m, \ldots = 1, \ldots, n$; features measured labelled by $i, j, k, \ldots = 1, \ldots, p$

A normalized measure of the linear correlation between 2 variables.



Several sets of (x, y) points and their correlation coefficient corr(x,y).

Top: corr(x,y) reflects the strength and direction of a linear relationship.
Middle: corr(x,y) does not reflects the slope of that relationship.
Bottom: nor many aspects of nonlinear relationships.

rem: the figure in the center has a slope of 0 but in that case the correlation coefficient is undefined because the variance of Y is zero.

[from wikipedia : Pearson correlation coefficient]

The covariance matrix characterizes the variability of a dataset

The covariance matrix $\boldsymbol{\Sigma}$ can be computed from the normalized dataset $\hat{\boldsymbol{X}}$:

$$\boldsymbol{\Sigma} = \frac{1}{n} \hat{\boldsymbol{X}}^T \hat{\boldsymbol{X}} \,.$$

It contains the variances of the variables on the diagonal and the covariances on the off-diagonal elements.

The covariance matrix characterizes the variability of a dataset

$$X = \begin{pmatrix} x^1_1 & x^1_2 & x^1_3 \\ x^2_1 & x^2_2 & x^2_3 \\ x^3_1 & x^3_2 & x^3_3 \end{pmatrix}$$

→ cell #1 for which we have measured the expression of three genes

$$\begin{pmatrix} \mu_1 & \mu_2 & \mu_3 \\ \mu_1 & \mu_2 & \mu_3 \\ \mu_1 & \mu_2 & \mu_3 \end{pmatrix}$$

Σ is all about **variability** and does not look at the mean value of each variable

$$\Rightarrow \hat{X} = \begin{pmatrix} x^1_1 - \mu_1 & x^1_2 - \mu_2 & x^1_3 - \mu_3 \\ x^2_1 - \mu_1 & x^2_2 - \mu_2 & x^2_3 - \mu_3 \\ x^3_1 - \mu_1 & x^3_2 - \mu_2 & x^3_3 - \mu_3 \end{pmatrix}$$ is the normalized dataset

where $\mu_i = \frac{1}{3}(x^1_i + x^2_i + x^3_i)$ is the mean value of the "ith" variable

We can compute the covariance matrix:

$$\Sigma = \frac{1}{n} \hat{X}^T \hat{X}$$

Exercise:

check that $\Sigma = \begin{pmatrix} \sigma_1^2 & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & \sigma_2^2 & \sigma_{23} \\ \sigma_{13} & \sigma_{23} & \sigma_3^2 \end{pmatrix} \Rightarrow$

The covariance matrix Σ contains the variances of each variable on its diagonal and covariances on off-diag elements.

1. Introduction

2. (Co-)variance of a dataset

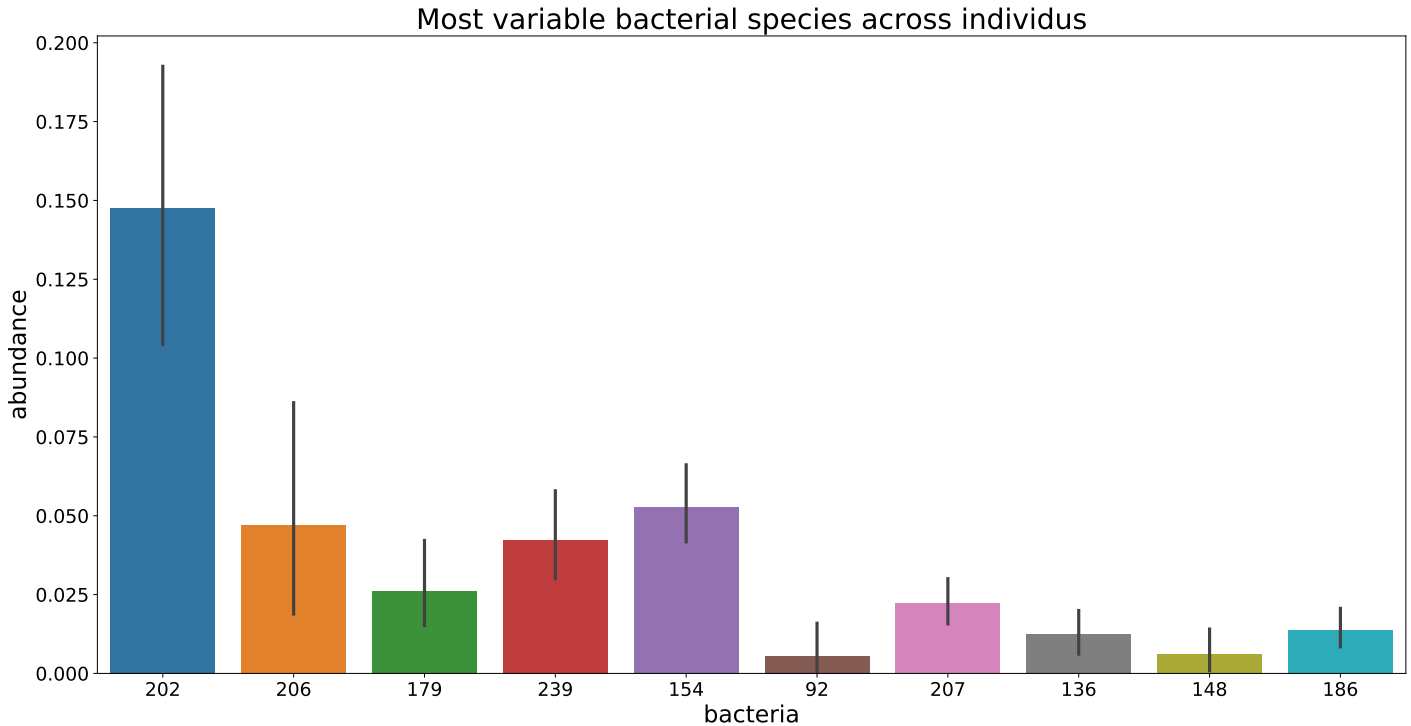
3. Bar, box and violin plots

Example of dataset : the gut microbiome

To illustrate box, bar and violin plots, we will consider a dataset which consists of the abundance of 248 microbial species of 33 individuals, see a snapshot here under:

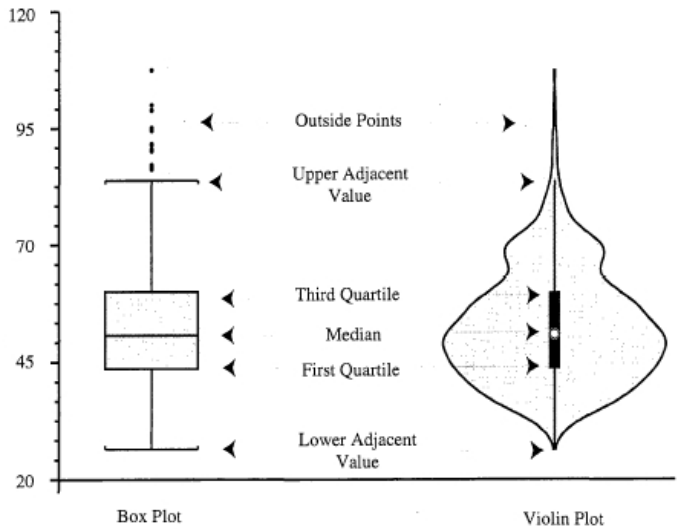
0.0000268	0.0000798	0.0000636	0.0000565	0.000115896	0.0000432	0.0000316	0.00019609	0.00004	0.0000716
0.000615703	0.000541181	0.001046353	0.000665158	0.000689109	0.000437158	0.000512171	0.000408606	0.000397813	0.000482034
0.000510041	0.000068	0.0000724	0.0000687	0.0000645	0.0000416	0.0000714	0.0000388	0.000130294	0.0000472
0.000178808	0.00000586	0.00000189	0.0000147	0.0000544	0.0000849	0.0000117	0.000068	0.0000198	0.00000488
0.000116489	0.000296717	0.000183091	0.000108476	0.0000976	0.0000494	0.000222225	0.000196857	0.000327481	0.000348413
0.001166081	0.000722	0.000718418	0.000507162	0.00055814	0.000568596	0.00067678	0.000475078	0.000552965	0.000549491
0.000258469	0.000334943	0.000395204	0.000251151	0.000166682	0.000299899	0.000201003	0.000251086	0.000343657	0.000321792
0.000475983	0.000336855	0.000446418	0.000371842	0.000326289	0.00057566	0.00037902	0.000729107	0.000322915	0.000404254
0.000587723	0.000615169	0.000573969	0.000582565	0.000425491	0.000473737	0.000519671	0.000501205	0.000323562	0.000506146
0.000056	0	0.0000152	0.0000378	0.0000124	0.0000191	0.0000282	0.0000109	0.0000384	0.00000822
0.000806178	0.000735476	0.000642349	0.00037878	0.00068775	0.000684564	0.000770342	0.000795664	0.00078233	0.000714904
0.00027869	0.0004671	0.00042568	0.000231702	0.000455785	0.000526811	0.000426485	0.000269044	0.000270495	0.0002747
0.000333809	0.000461454	0.000401274	0.000570715	0.000913697	0.000579964	0.000611882	0.000211419	0.000343369	0.000397113

The bar plot : means and variances



A bar plot represents the mean values of the variables (via the height of the boxes) as well as the standard deviation which are by definition the square root of the variances (black lines). Here we selected the bacterial species having the largest variances as showing all 248 bacterial species would not be very readable.

Medians and quartiles



Median: value such that half of the samples have a higher value and half of them have a lower value.

First quartile (Q1) : value between the lowest value and the median of the dataset. 25 percent of the data points lie below this value.

Third quartile (Q3) : value between the median and the highest value of the dataset. 25 percent of the data points lie above this value.

The inter quartile range (IQR) is defined as Q3 - Q1.
The lower adjacent value is Q1 - 1,5 IQR. The upper adjacent value is Q3 + 1,5 IQR.

Samples outside the lower/upper adjacent values can be considered as **outliers**.

The **envelope** around the violin plot represents the probability density of the data.

Introduction & ‘Simple’ plots and recap of linear algebra (in python)

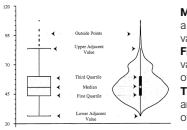
Bar, box and violin plots

Medians and quartiles

Read this short article:
https://towardsdatascience.com/violin-plots-explained-fb1d115e023d

For more info, see also **https://en.wikipedia.org/wiki/Quartile**, **https://en.wikipedia.org/wiki/Violin_plot** and **https://seaborn.pydata.org/generated/seaborn.violinplot.html**

Medians and quartiles



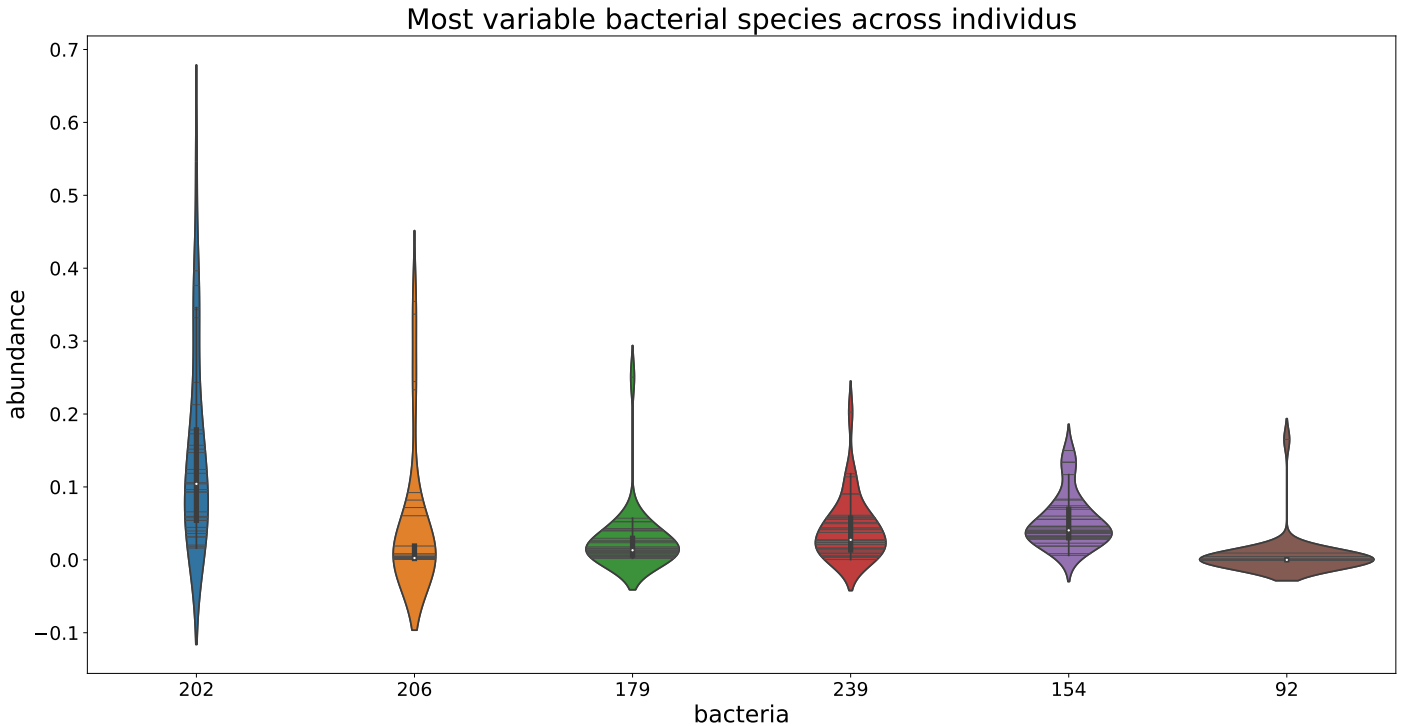
Median: value such that half of the samples have a higher value and half of them have a lower value.
First quartile (Q1) : value between the lowest value and the median of the dataset. 25 percent of the data points lie below this value.
Third quartile (Q3) : value between the median and the highest value of the dataset. 25 percent of the data points lie above this value.

The inter quartile range (IQR) is defined as Q3 - Q1.
The lower adjacent value is Q1 - 1,5 IQR. The upper adjacent value is Q3 + 1,5 IQR.

Samples outside the lower/upper adjacent values can be considered as **outliers**.

The **envelope** around the violin plot represents the probability density of the data.

The box and violin plots : medians and quartiles



We can add all samples on the violin plot to represent more precisely the distribution of the samples for each variable, as done in the above plot with the thin horizontal lines.

We will see how to make those plots in python $\Rightarrow$ python_basic.ipynb .

We will also recall useful python tools: arrays, matrices, for loops, define a function, use elementary plotting functions.

The datasets are encoded in matrices and it will be useful to know linear algebra to manipulate those dataset. In particular we need to:

- compute the covariance matrix of a dataset
- perform coordinates change
- compute eigenvectors and eigenvalues of a matrix
- know the theorem stating that a symmetric matrix can be diagonalised by an orthogonal matrix.