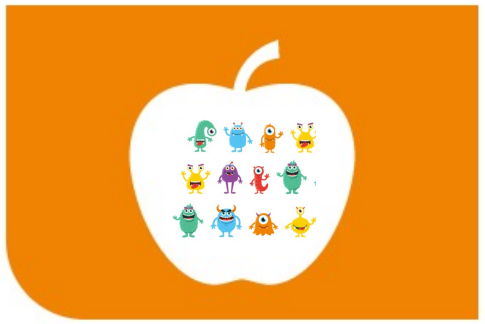
https://github.com/IshtarMM/spp_course22

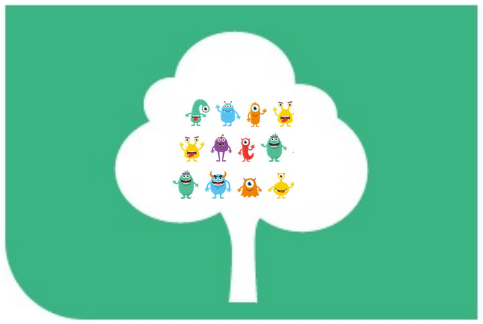
Computational aspects of planning and performing studies on microbiome interactions

Maryam Mahmoudi
Prof. Dr. Eric Kemen
Microbial Interactions in Plant Ecosystems Center for Plant Molecular Biology (ZMBP)
University of Tuebingen

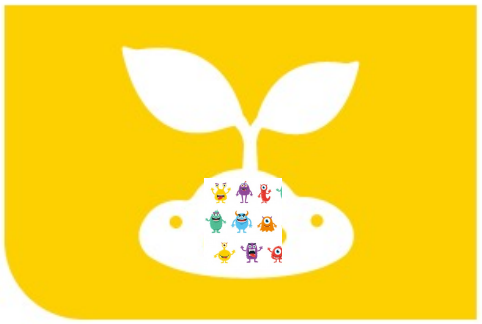
Microbes are everywhere



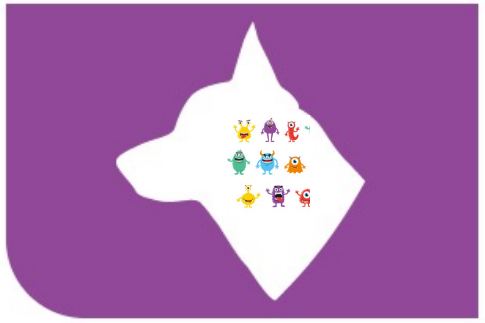
Food



Plant

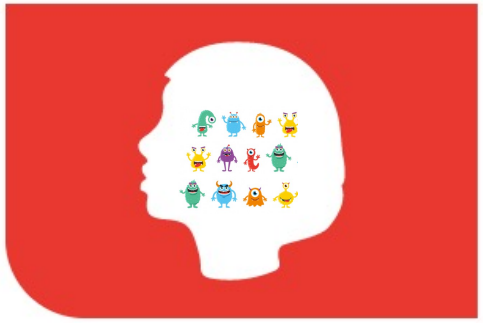


Soil



Animal

Marine



Human

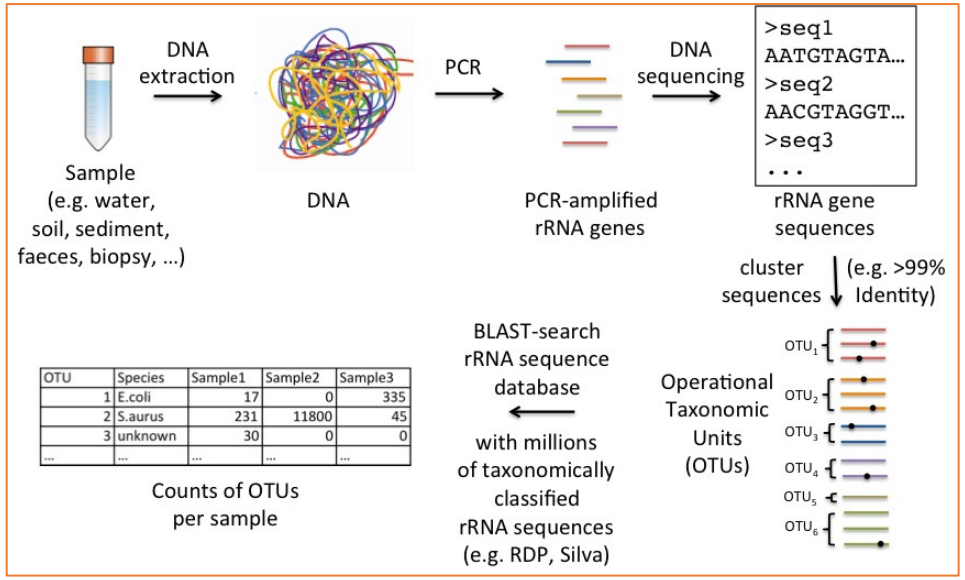
Illumina-based amplicon sequencing to study microbial diversity

How can we study microbial diversity in a sample i.e. identify the phylogenetic groups present in that sample?

Cultured based

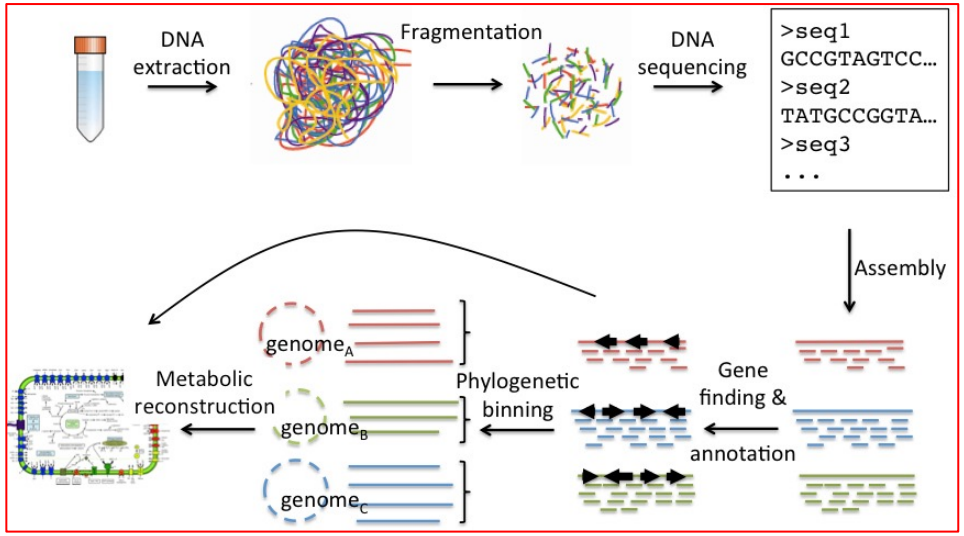
Genome sequencing

Amplicon sequencing



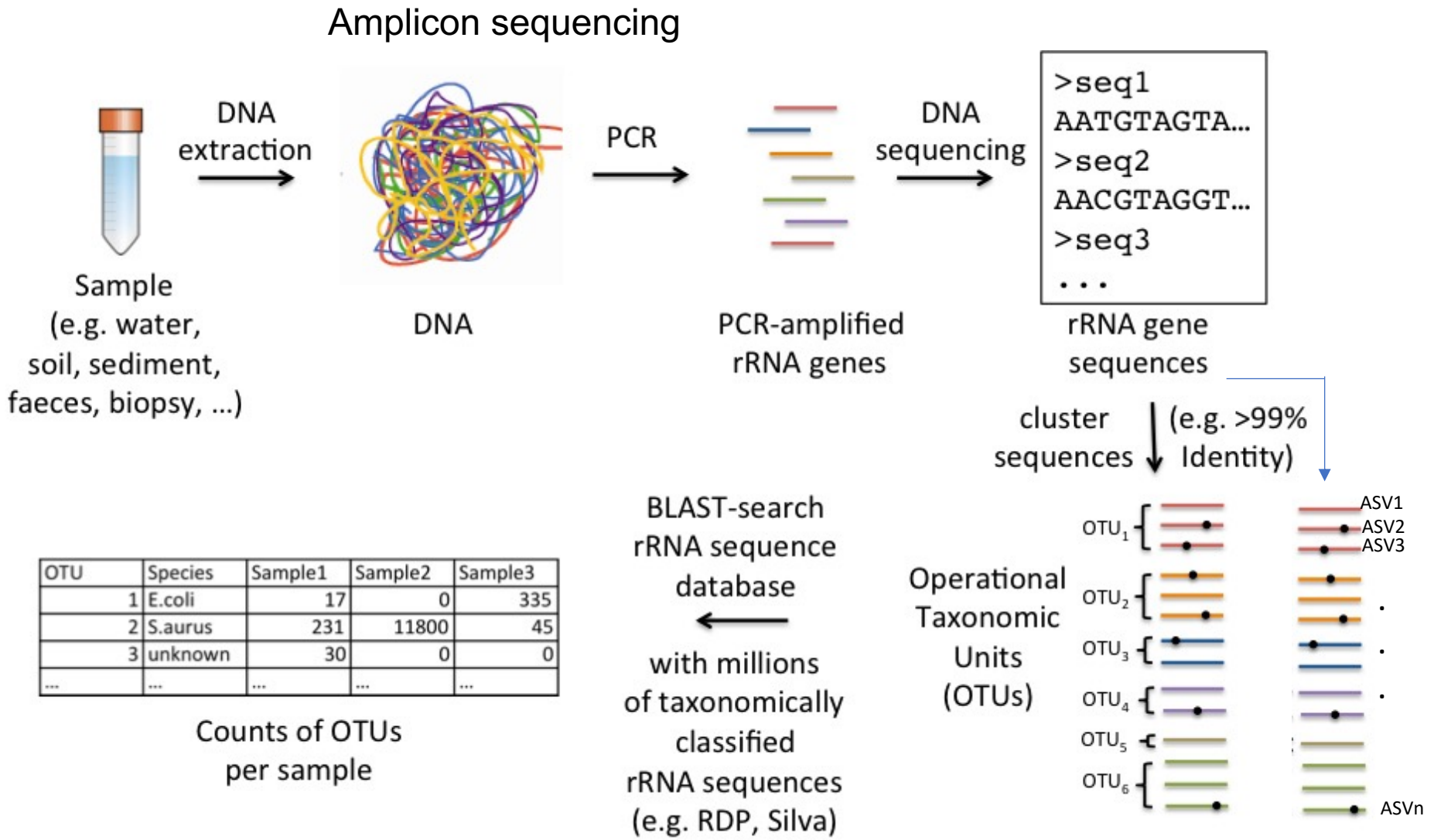
OTU	Species	Sample1	Sample2	Sample3
1	E.coli	17	0	335
2	S.aurus	231	11800	45
3	unknown	30	0	0
...	...	...	...	...

Metagenomic sequencing



Illumina-based amplicon sequencing to study microbial diversity

How can we study microbial diversity in a sample i.e. identify the phylogenetic groups present in that sample?



OTU	Species	Sample1	Sample2	Sample3
1	E.coli	17	0	335
2	S.aurus	231	11800	45
3	unknown	30	0	0
...	...	...	...	...

Counts of OTUs per sample

Overview of amplicon sequencing data(format and analysis)

Sample's sequencing

fastq files

Header | Sequence | Quality

```
@HWI-ST227:389:C4WA2ACXX:7:1204:2272:59979
GGAGGAAGGTCCTCGCTCCTCTTTCATATAAGGGAAATGGCTGAAT
+
FFFFHHHHHHHJIJJJJJJJJIJJJIGIGIGGIJJIJIJJJJJJIII
```

demultiplex
(split samples by barodes)

fastqc/multiqc
plotQuality (dada2)

quality filter/trim
(remove adapters/primers)

fasta files

```
>HISEQ2500:282:1:1101:1220:1944 1
ATCGGATCG...
```

generate OTUs

chimera removal

resolve ASVs

output format

Count table

	sample1	sample2	Sample n
OTU/ASV1	1000	0	1
OTU/ASV2	20	800	265
OTU/ASV m	0	0	10

Taxonomy of each OUT/ASV's

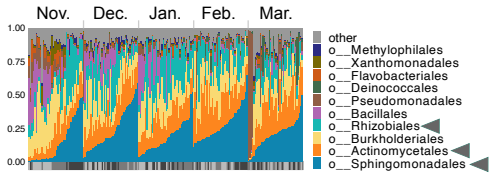
	Taxonomy
OTU/ASV1	Bacteria; Sphingomonas;
OTU/ASV2	Bacteria;unknown
OTU/ASV m	Bacteria;Pseudomonas

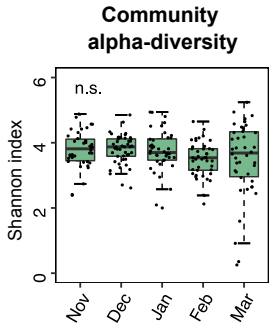
Fasta file

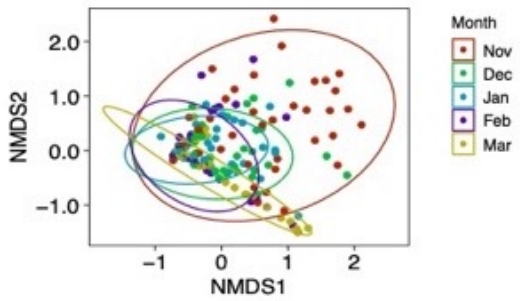
```
>OTU/ASV1
ATCGATCG…

>OTU/ASV1
TCGAATCG…

>OTU/ASVn
TATCGACG…
```

Analyses

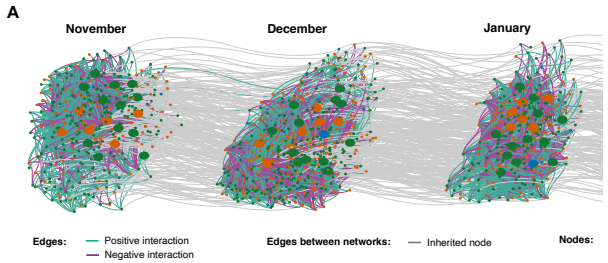
Diversity and composition



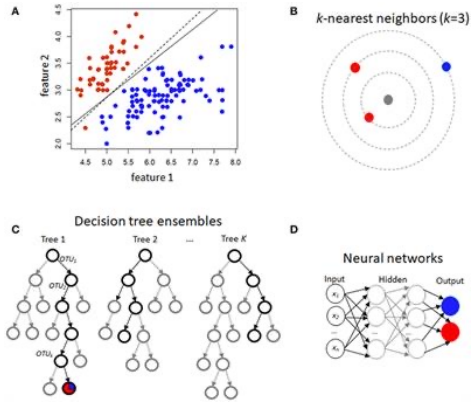




Correlation network



Machine learning



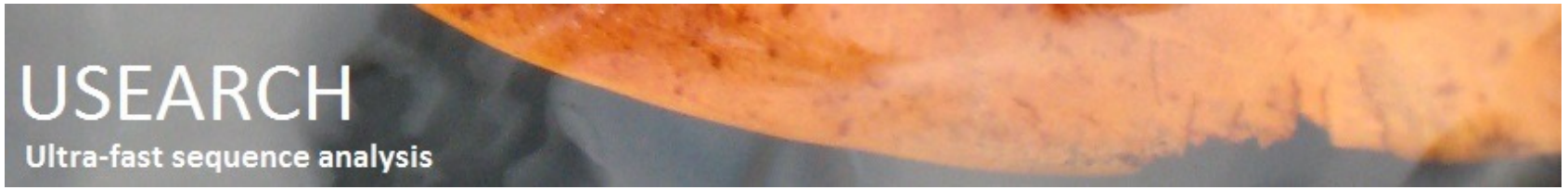
Tools for amplicon sequence data analyses

Mothur

qiime2

DADA2

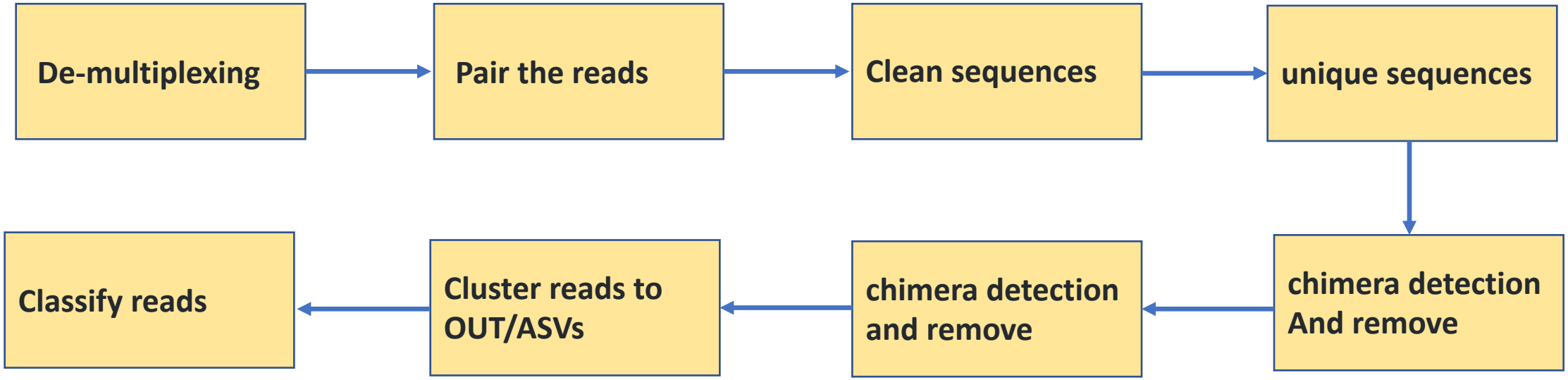
Amplicon Sequencing. Exactly. Version 1.14



http://www.mothur.org http://www.qiime.org

three OTU-level flows (QIIME-uclust, MOTHUR, and USEARCH-UPARSE) and three ASV-level (DADA2, Qiime2-Deblur, and USEARCH-UNOISE3). We tested workflows with different quality control options, clustering algorithms, and cutoff parameters on a mock community as well as on a large (N = 2170) recently published fecal sample dataset from the multi-ethnic HELIUS study. We assessed the sensitivity, specificity, and degree of consensus of the different outputs. DADA2 offered the best sensitivity, at the expense of decreased specificity compared to USEARCH-UNOISE3 and Qiime2-Deblur. USEARCH-UNOISE3 showed the best balance between resolution and specificity. OTU-level USEARCH-UPARSE and MOTHUR performed well, but with lower specificity than ASV-level pipelines. QIIME-uclust produced large number of spurious OTUs as well as inflated alpha-diversity measures and should be avoided in future studies. This study provides guidance for researchers using amplicon sequencing to gain biological insights.

Workflow amplicon sequencing analysis using Mothur pipeline



De-multiplexing

Normally you get the sequences from all samples together and you have to divide them this is called de-multiplexing. The first step is then to divide all.

```
Forward.fastq Reverse.fastq Index.fastq
```

```
B5.145  File_FastqInfo.B5.145_TGGAAGATGAGT.forward.fastq        File_FastqInfo.B5.145_TGGAAGATGAGT.reverse.fastq
B5.146  File_FastqInfo.B5.146_GAACTGTATCTC.forward.fastq        File_FastqInfo.B5.146_GAACTGTATCTC.reverse.fastq
B5.147  File_FastqInfo.B5.147_TGATAGTGAGGA.forward.fastq        File_FastqInfo.B5.147_TGATAGTGAGGA.reverse.fastq
```

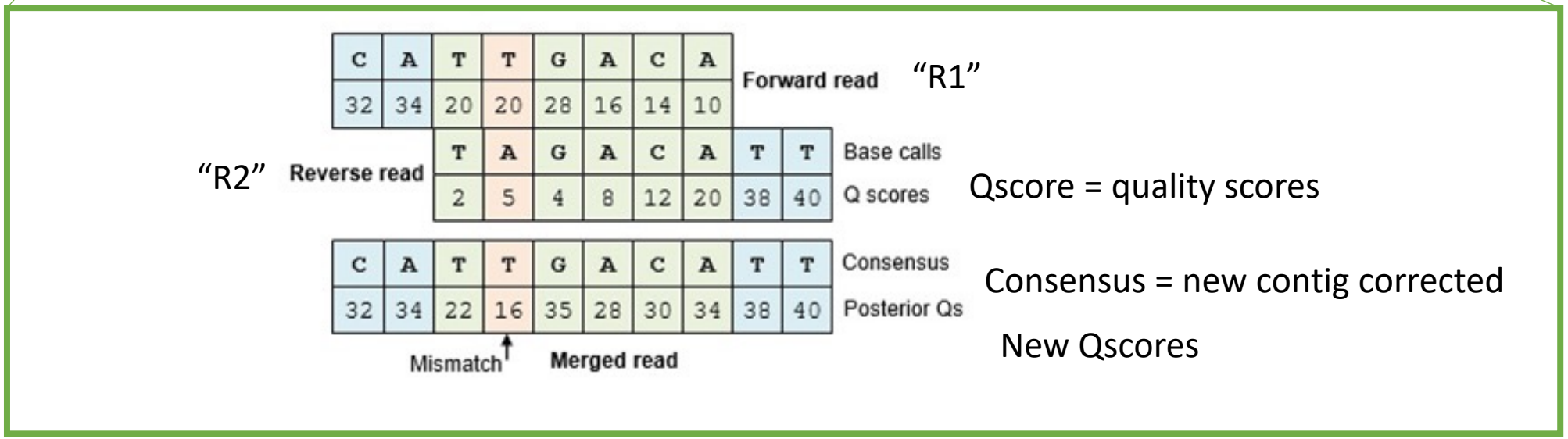
Creating contigs from paired-end reads

Forward read “R1”

Reverse read “R2”

Pair single reads

Contig creation and base correction (overlapping region)

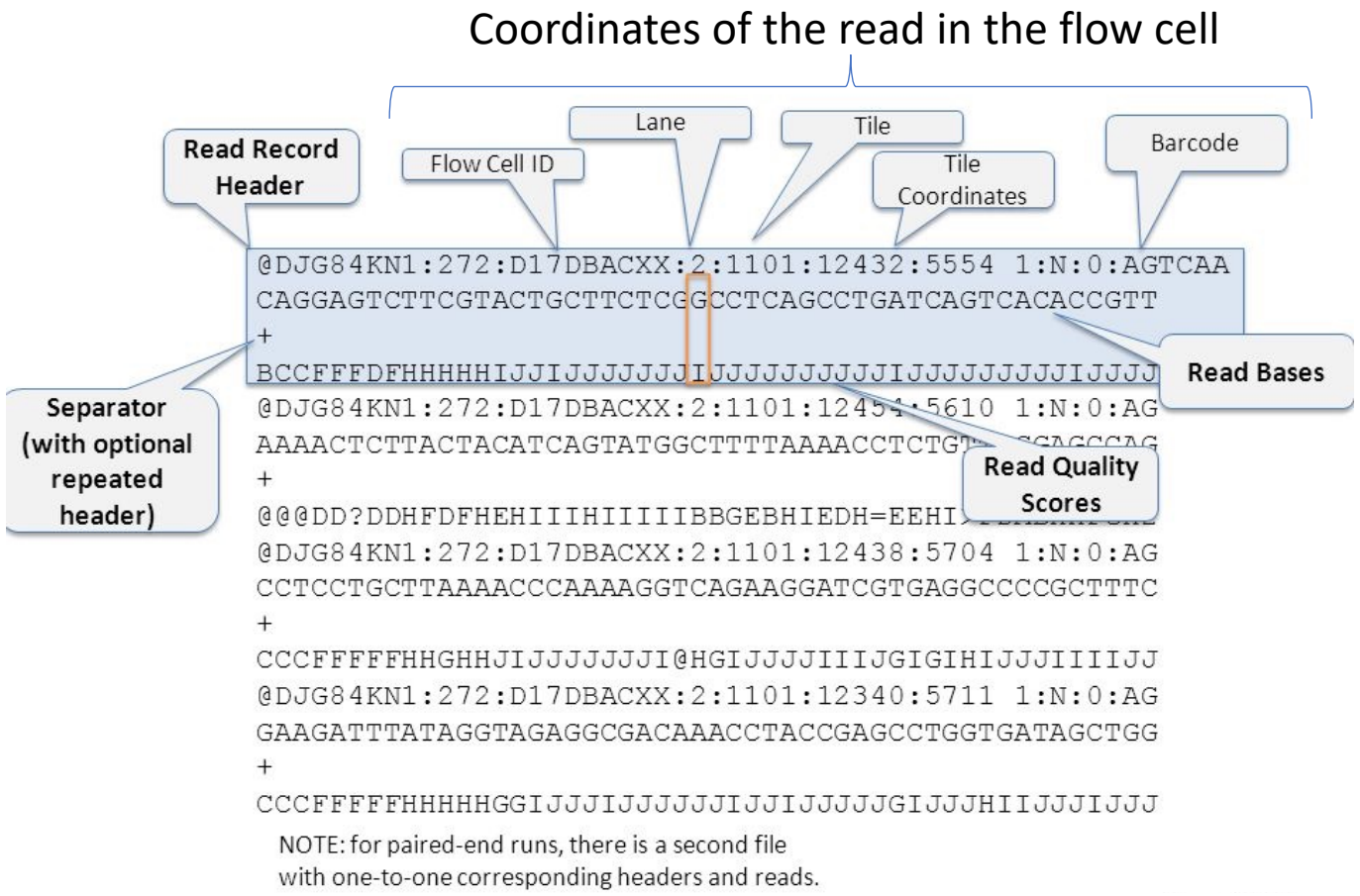


paired-end sequencing of the ~300 bp V5 region of the 16S rRNA gene was performed. The sequencing was done from either end of each fragment. Because the reads are about 300 bp in length, this results in a significant overlap between the forward and reverse reads in each pair. We will combine these pairs of reads into *contigs*.

Clean sequences

Illumina sequencing : output files

3 fastq files : Forward, Reverse and Index
fastq format = FASTa + Quality score of each base



quality score (Qscore) of each base = symbol

Table 1 ASCII Characters Encoding Q-scores 0–40

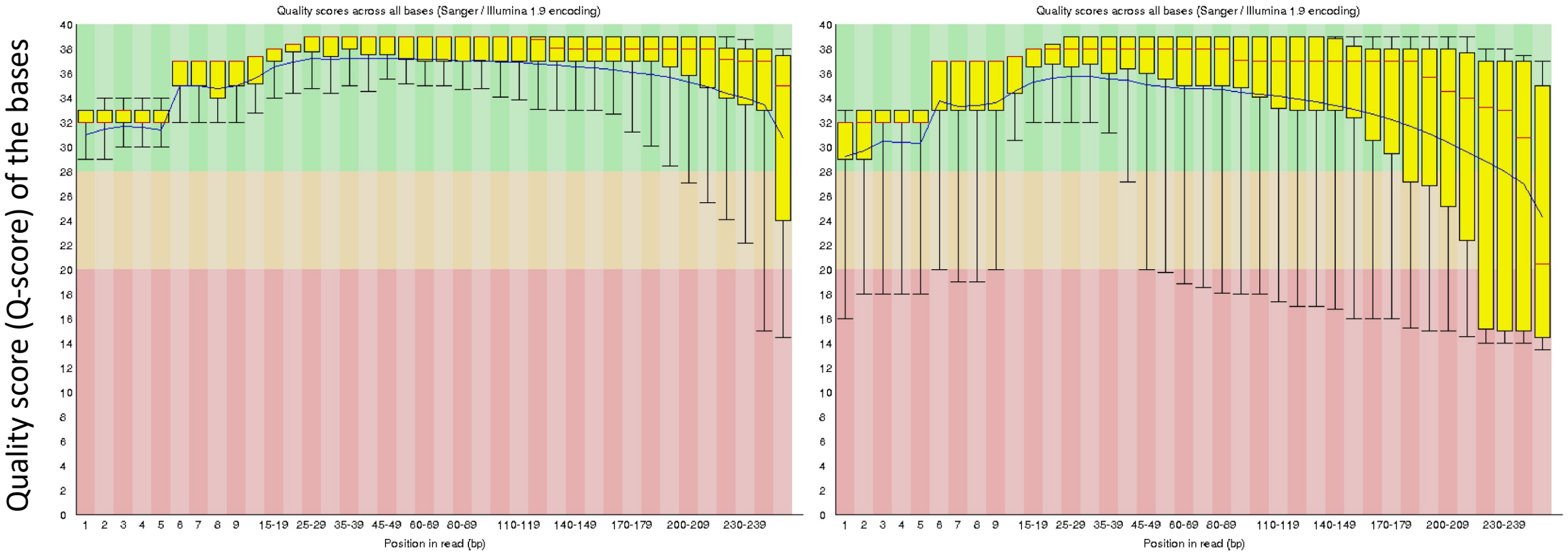
Symbol	ASCII Code	Q-Score	Symbol	ASCII Code	Q-Score	Symbol	ASCII Code	Q-Score
!	33	0	/	47	14	=	61	28
"	34	1	0	48	15	>	62	29
#	35	2	1	49	16	?	63	30
$	36	3	2	50	17	@	64	31
%	37	4	3	51	18	A	65	32
&	38	5	4	52	19	B	66	33
'	39	6	5	53	20	C	67	34
(	40	7	6	54	21	D	68	35
)	41	8	7	55	22	E	69	36
*	42	9	8	56	23	F	70	37
+	43	10	9	57	24	G	71	38
,	44	11	:	58	25	H	72	39
-	45	12	;	59	26	I	73	40
.	46	13	<	60	27			

Qscore = **−10 log10 P**
P= error probability

e.g. 1% probability of error = Qscore of 20

Clean sequences

Quality score and sequence length



Forward reads

Sequenced bases

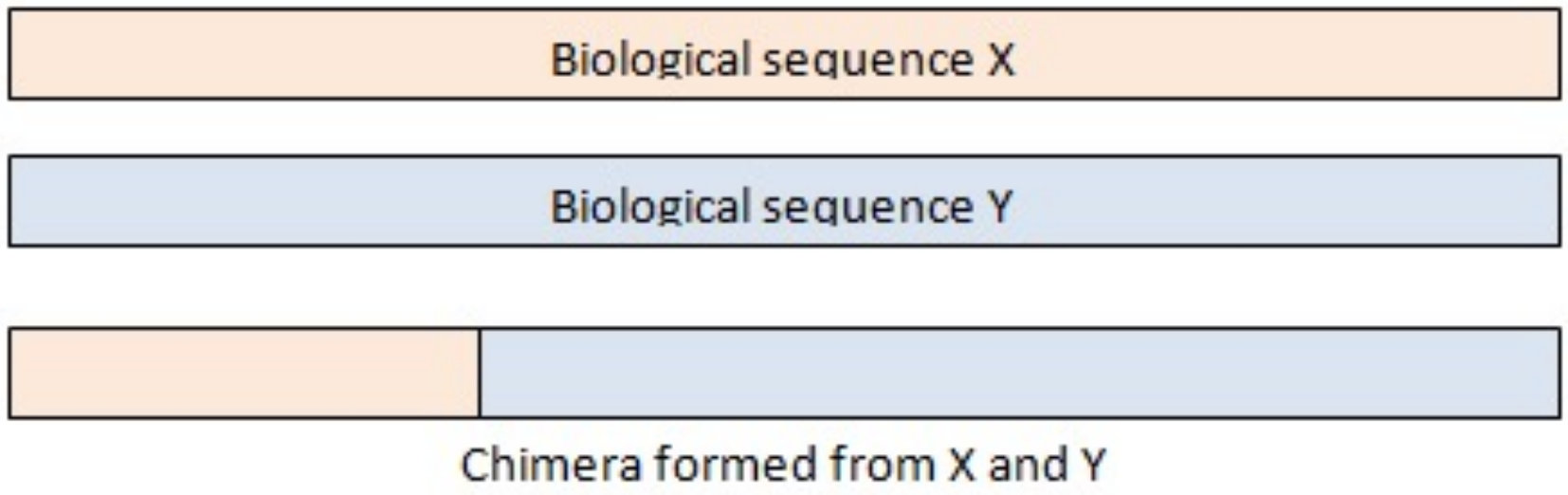
Reverse reads

→ sequencing quality (precision) decreases with each polymerization cycle

→ the central bases overlapping between the two reads have to be corrected

chimera detection And remove

Chimeras are artifact sequences formed by two or more biological sequences incorrectly joined together.



Cluster reads To OTUs

What is OTU Clustering?

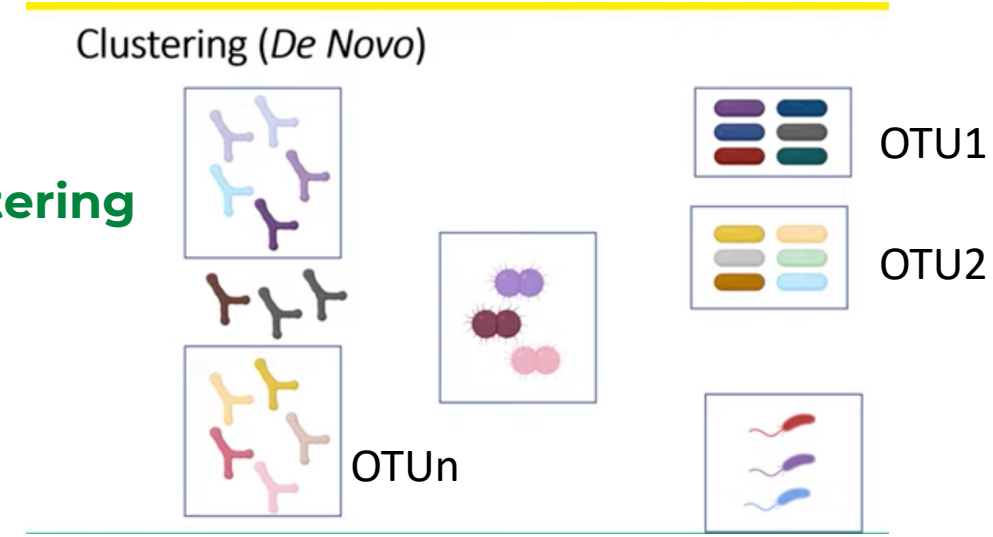
In order to minimize the risks of sequencer error in targeted sequencing, clustering approaches were initially developed. Clustering approaches are based upon the idea that related/similar organisms will have similar target gene sequences and that rare sequencing errors will have a trivial contribution, if any, to the consensus sequence for these clusters, or operating taxonomic units (OTUs)[1].

There are three basic methods to generate OTUs from sequencing data, with clusters often being generated using a similarity threshold of 97% sequence identity. This approach carries with it the risk that multiple similar species can be grouped into a single OTU, with their individual identifications being lost to the abstract of a cluster. Alternatively, some have tried the approach of requiring extremely high levels of sequence identity to minimize the risk of losing diversity to clustering, with thresholds closer to 100% being used, but this creates a significant risk of identifying sequencing errors as new species and false diversity[2].

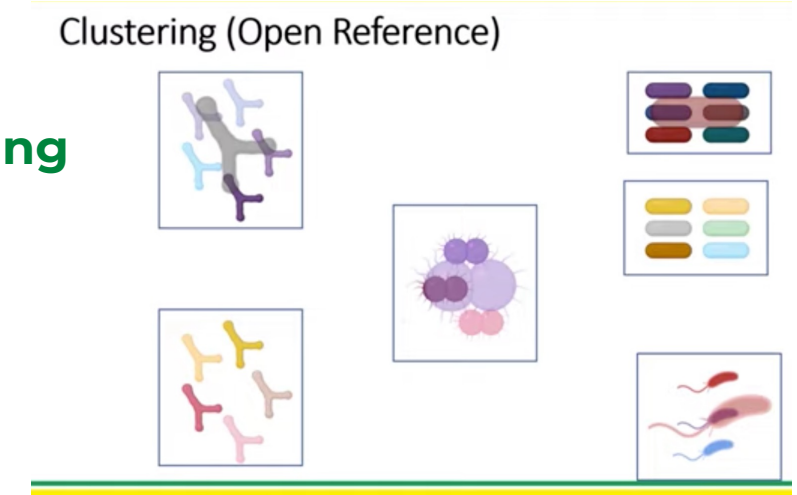
Cluster reads To OTUs

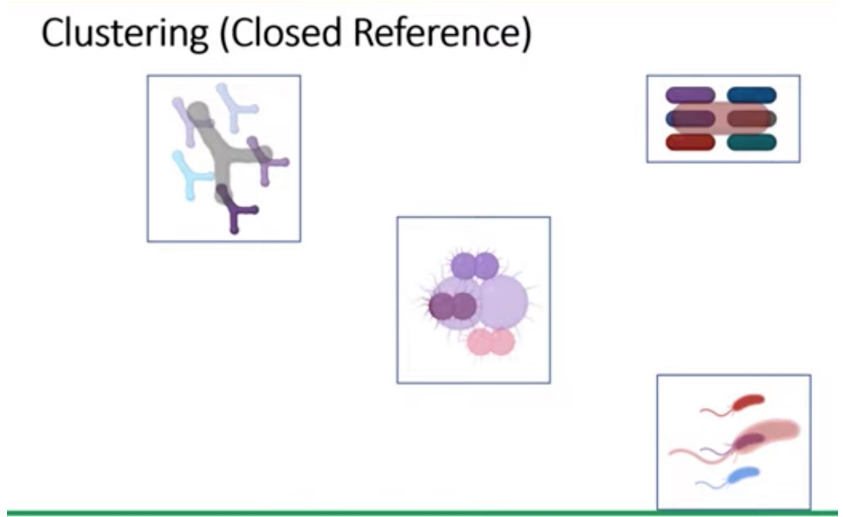
OTUs(Operational taxonomic unit) or clusters with sequence similarity(97 or 99 percent)

Reference-free OTU Clustering



Reference-based OTU Clustering





https://www.zymoresearch.com/blogs/blog/microbiome-informatics-otu-vs-asv

Operational Taxonomic Unit (OTU) Approach

We know some of these sequences arose from error/artifact.

Combine extremely similar sequences (usually 97% or more identity) to minimize the effects of observed errors. Then treat each OTU as a representative sequence.

- OTUs can be represented by a representative consensus sequence
- Closed reference OTUs are fast to create, but are subject to reference bias
- *De novo* OTUs are free of reference bias, but computationally expensive and can change with changed samples
- Open reference OTUs are in between, with sequence similar to reference behaving like closed reference and more novel sequences behaving like *de novo*

https://www.zymoresearch.com/blogs/blog/microbiome-informatics-otu-vs-asv

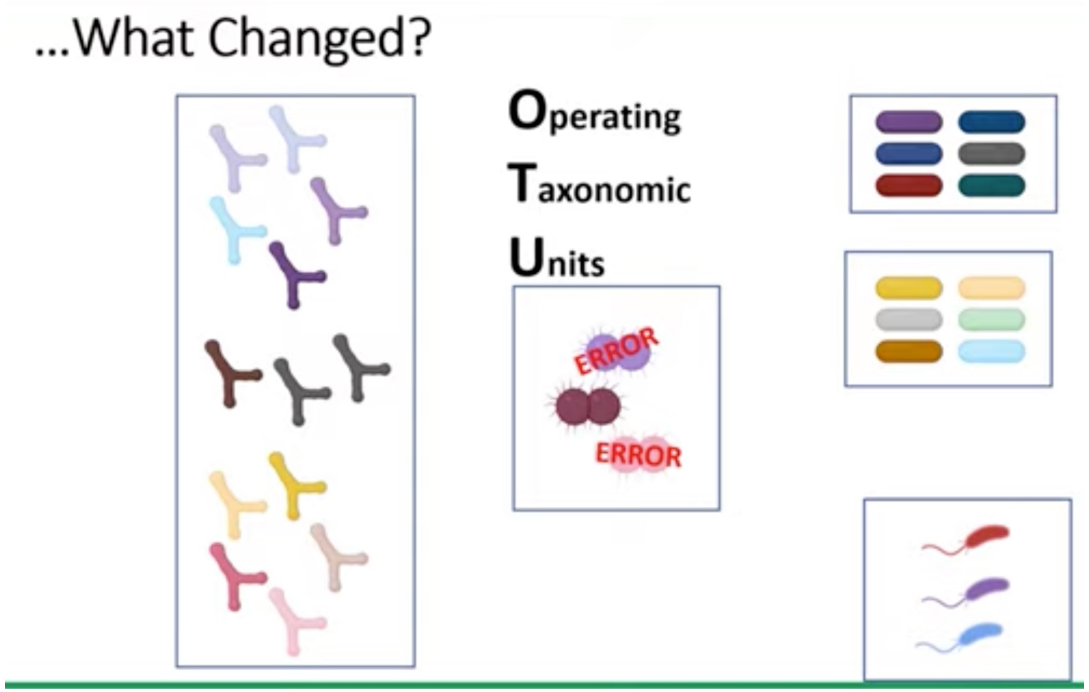
What is ASV Analysis?

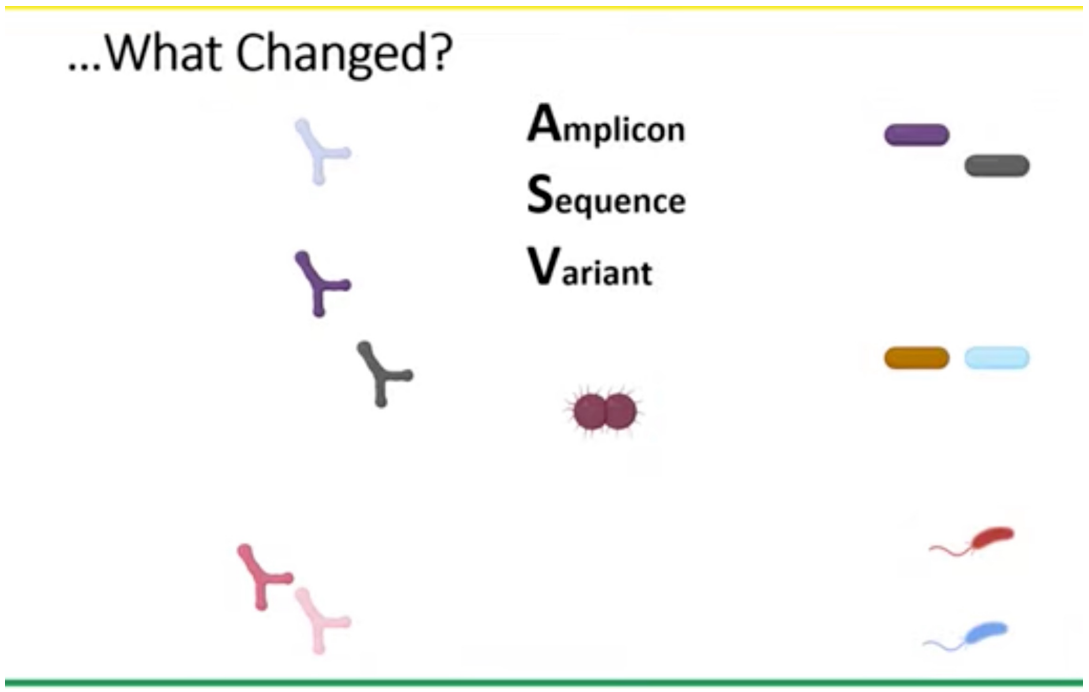
While OTU clustering approaches attempt to blur similar sequences into an abstracted consensus sequence, thus minimizing the influence of any sequencing errors within the pool of reads, **the Amplicon Sequence Variant (ASV)** approach attempts to go the opposite direction. **The ASV approach will start by determining which exact sequences were read and how many times each exact sequence was read**. These data will be combined with an error model for the sequencing run, enabling the comparison of similar reads to **determine the probability that a given read at a given frequency is not due to sequencer error**. This creates, in essence, a p-value for each exact sequence, where the null-hypothesis is equivalent to that exact sequence being a consequence of sequencing error.

Following this calculation, sequences are filtered according to some threshold value for confidence, leaving behind a collection of exact sequences having a defined statistical confidence. Because these are exact sequences, generated without clustering or reference databases, **ASV results can be readily compared between studies using the same target region**. Additionally, a given target gene sequence should always generate the same ASV and a given ASV, being an exact sequence, can be compared to a reference database at a much higher resolution allowing for more precise identification down to the species level and even potentially beyond[3].

The ASV :

approach will start by determining which exact sequences were read and how many times each exact sequence was read. These data will be combined with an error model for the sequencing run, enabling the comparison of similar reads to determine the probability that a given read at a given frequency is not due to sequencer error





https://www.zymoresearch.com/blogs/blog/microbiome-informatics-otu-vs-asv

Amplicon Sequence Variant (ASV) Approach

What is the statistical support for each sequence's existence?

Throw out amplicon sequences that lack strong statistical support for not being artifacts of sequencing. Cost: potential loss of real sequence that was present at very low levels.

- No representative consensus sequence
  - Each sequence is supported as being present in the sample
- Potentially higher resolution, no potential to combine multiple "real" sequences into an abstract
- Generated without the use of a reference, no risk of reference bias.
- May also be called an ESV (exact sequence variant) or zOTU (zero-radius OTU)

https://www.zymoresearch.com/blogs/blog/microbiome-informatics-otu-vs-asv

OTU *vs.* ASV

OTU	ASV
Can be subject to reference bias	Reference is not used until taxonomy assignment
OTU tables cannot be combined between studies	ASV tables can be compared across studies
Represented by a consensus sequence	Represented by an exact sequence
Can represent multiple species with different sequences	If it represents multiple species, it is because they share the sequence
Subject to chimeric sequences	Subject to chimeric sequences
Chimera detection can be complex and may require reference bias	Chimera detection is simple and reference-free

https://www.zymoresearch.com/blogs/blog/microbiome-informatics-otu-vs-asv

Output of analysis

OTU/ASV table

OTU	Sample1	Sample2	Sample3	SampleN
Otu_0002	86	27	4	4
Otu_0003	75	98	1	1
Otu_0004	13	8	17	4
Otu_0005	2	12	69	0
Otu_0006	24	8	98	6
Otu_0007	13	65	150	15
Otu_0008	2	1	0	184
Otu_0009	464	13	179	22
Otu_0010	293	43	9	46

Taxonomy table

OTU	Kingdom	Phylum	Class	Order	Familly	Genus	Species
Otu_0002	k__Bacteria	p__Proteobacteria	c__Alphaproteobacteria	o__Rhizobiales	f__Methylobacteriaceae	g__Methylobacterium	s__adhaesivum
Otu_0003	k__Bacteria	p__Proteobacteria	c__Alphaproteobacteria	o__Sphingomonadales	f__Sphingomonadaceae	g__Sphingomonas	g__Sphingomonas_unclassified
Otu_0004	k__Bacteria	p__Proteobacteria	c__Betaproteobacteria	o__Burkholderiales	f__Oxalobacteraceae	f__Oxalobacteraceae_unclassified	f__Oxalobacteraceae_unclassified
Otu_0005	k__Bacteria	p__Actinobacteria	c__Actinobacteria	o__Actinomycetales	f__Microbacteriaceae	g__Rathayibacter	s__caricis
Otu_0006	k__Bacteria	p__Firmicutes	c__Bacilli	o__Bacillales	f__Bacillaceae	g__Bacillus	g__Bacillus_unclassified
Otu_0007	k__Bacteria	p__Proteobacteria	c__Alphaproteobacteria	o__Sphingomonadales	f__Sphingomonadaceae	g__Sphingomonas	g__Sphingomonas_unclassified
Otu_0008	k__Bacteria	p__Proteobacteria	c__Gammaproteobacteria	o__Pseudomonadales	f__Pseudomonadaceae	g__Pseudomonas	g__Pseudomonas_unclassified
Otu_0009	k__Bacteria	p__Proteobacteria	c__Gammaproteobacteria	o__Pseudomonadales	f__Pseudomonadaceae	g__Pseudomonas	s__viridiflava
Otu_0010	k__Bacteria	p__Proteobacteria	c__Betaproteobacteria	o__Burkholderiales	f__Comamonadaceae	g__Variovorax	s__paradoxus

Sample info table

Sample	Soil PH	Year	Temp	Day	Season
Sample1	7	2015	20	17	Spring
Sample2	6	2015	2	17	Fall
Sample3	8	2017	21	17	Spring
Sample4	6	2015	24	17	Spring
SampleN	5	2016	0	17	Fall

Access to files and some protocols:

https://github.com/IshtarMM/spp_course22

What is Colab?

Colab, or "Colaboratory", allows you to write and execute Python/R in your browser, with

- Zero configuration required
- Access to GPUs free of charge
- Easy sharing

Colab is basically a free Jupyter notebook environment running wholly in the cloud. Most importantly, Colab does not require a setup, plus the notebooks that you will create can be simultaneously edited by your team members – in a similar manner you edit documents in Google Docs.

For analyzing data using R programming in colab:

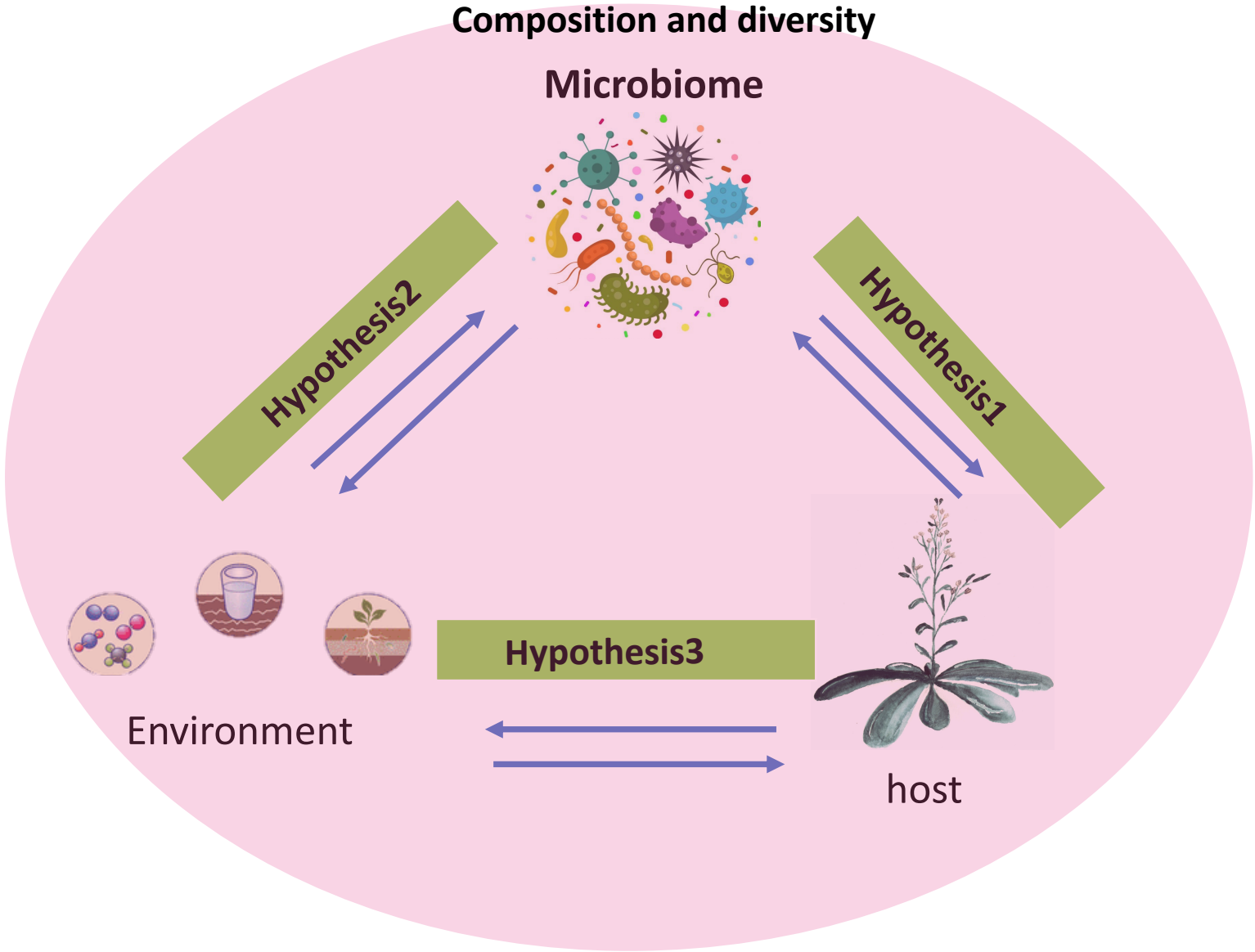
1. open this link in your browser:

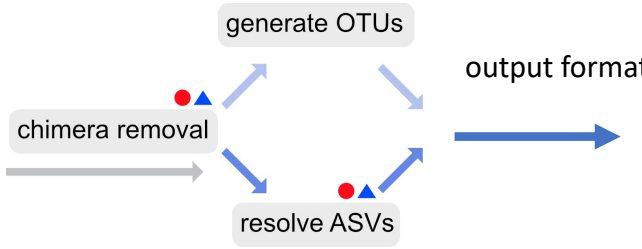
https://github.com/IshtarMM/spp_course22

2. Download …

3. Open the file in the Google Colab

Dynamic Interactions among environment, microbiome and host for the research hypotheses in microbiome studies





output format

Count table

	sample1	sample2	Sample n
OTU/ASV1	1000	0	1
OTU/ASV2	20	800	265
OTU/ASV m	0	0	10

Taxonomy of each OUT/ASV's

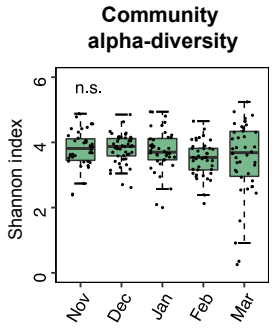
	Taxonomy
OTU/ASV1	Bacteria; Sphingomonas;
OTU/ASV2	Bacteria;unknown
OTU/ASV m	Bacteria;Pseudomonas

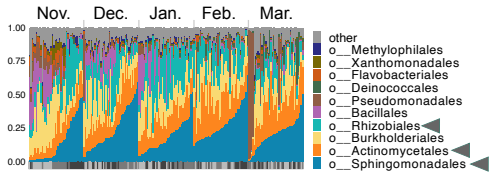
Fasta file

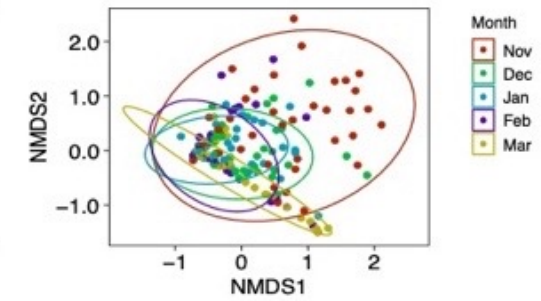
```
>OTU/ASV1
ATCGATCG…

>OTU/ASV1
TCGAATCG…

>OTU/ASVn
TATCGACG…
```

Analyses

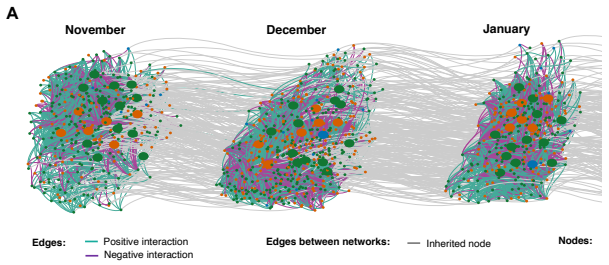
Diversity and composition



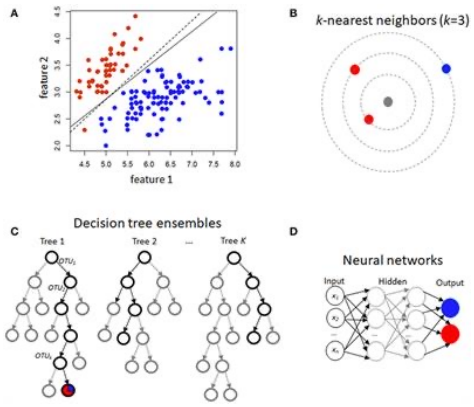




Correlation network



Machine learning



Data analysis : α-diversity indexes

α-diversity (within the sample): number of microbial species within the sample

Shannon index:

$$H = \sum_{i=1}^{S} - (P_i * \log(P_i))$$

$P_i = n/N$

n=number of individuals of each species

N=total number of individuals

S=total number of species

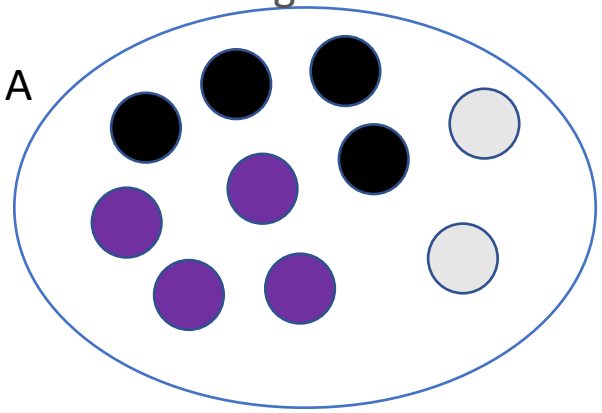
log - Usually the natural logarithm

Takes into account the total number of species (S) and their abundance (n) in the sample relative to all individuals from all species in the sample (N). Sensitive to dominant species.

What happens if the purple species is very abundant like 100?

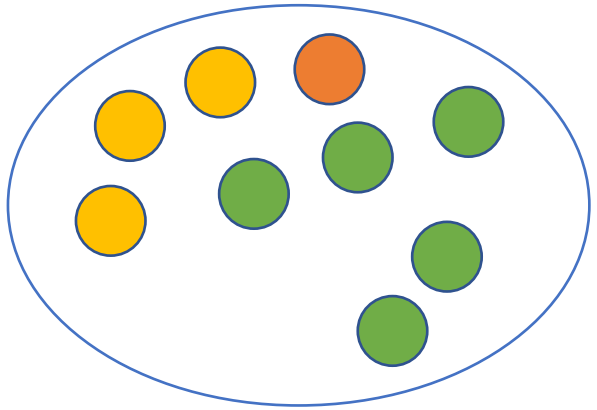
→ = 0.31 →the index decreases if there are dominant species (very abundant)

Sample A

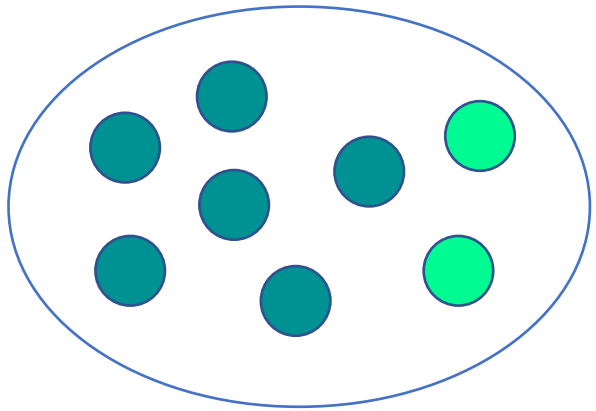


alpha= -(4/10*ln(4/10)+ 4/10*ln(4/10)+ 2/10*ln(2/10))=1.052

Sample B



Sample C

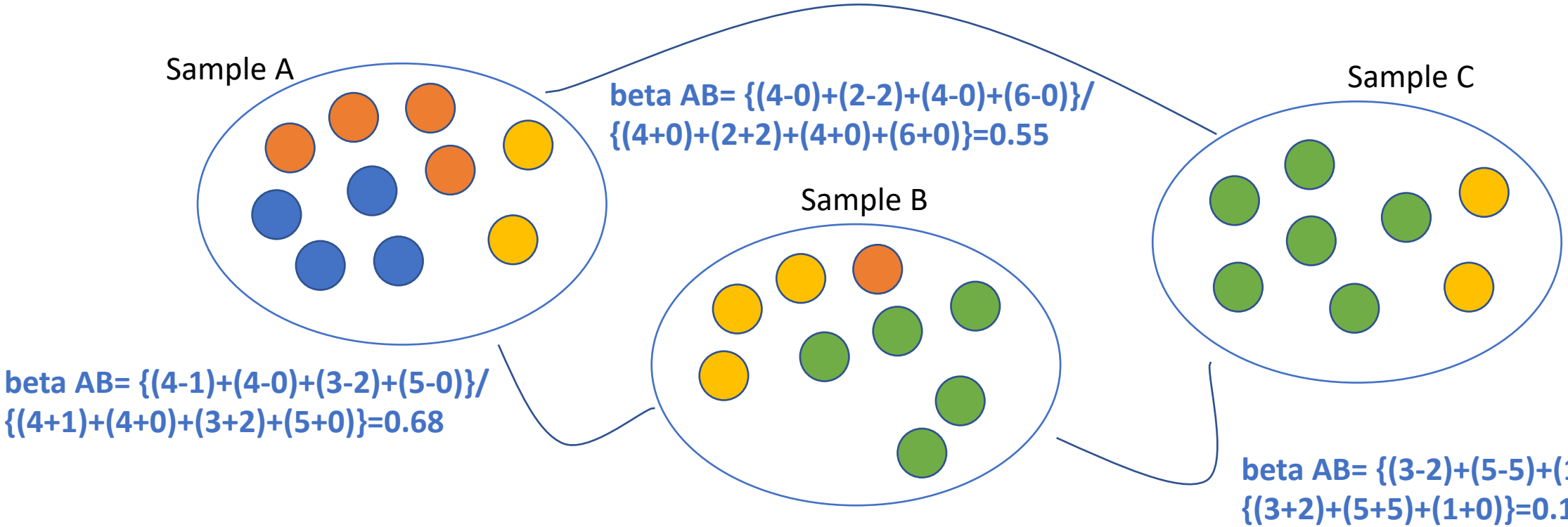


alpha= -(6/8*ln(6/8)+ 2/8*ln(2/8))=0.562

Data analysis : β-diversity metrics → distances

β-diversity metrics allow us to compare samples based on their differences and calculate distances between samples

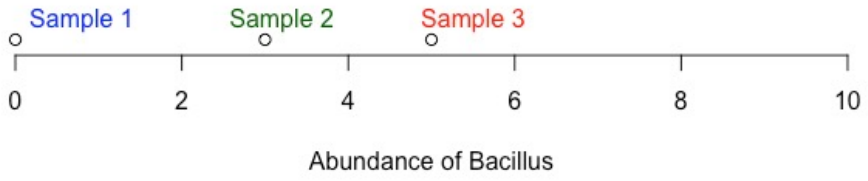
Bray-Curtis dissimilarities: for all species ∑ |abundance in sample A - abundance in sample B|/ ∑ |abundance in sample A + abundance in sample B|



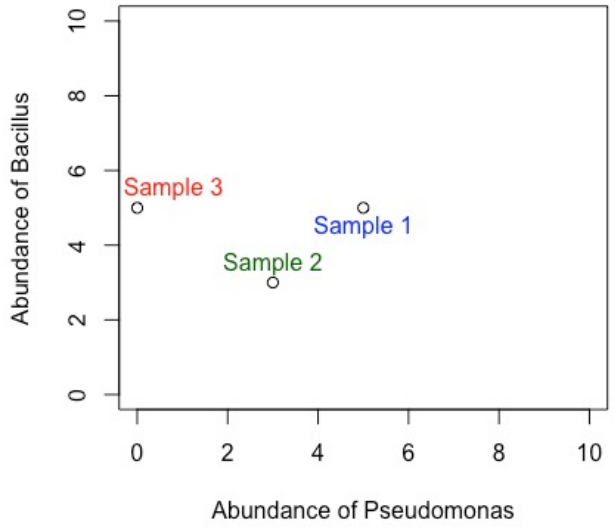
Which are the most similar samples? How are these results different from the Sorensen distance?

Dimention reduction approches for microbiome data

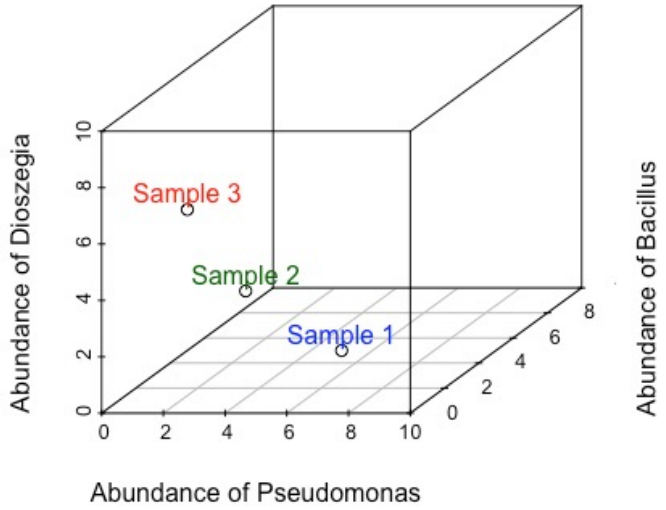
1



2

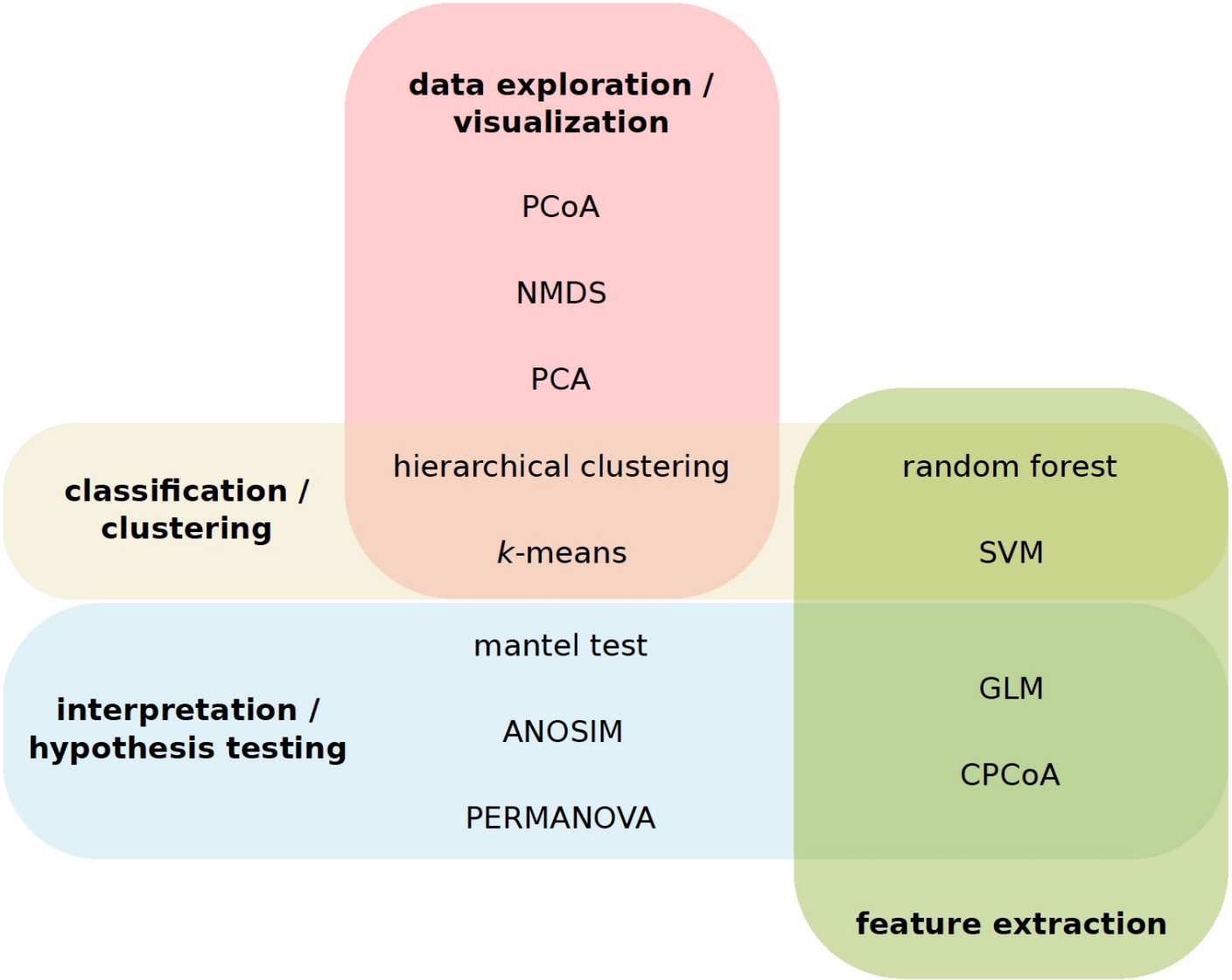


3



4.....1000

Overview of common multivariate analysis methods in microbial ecology



Ruben

Ordination methods in microbial ecology

PCA (Principal Component Analysis; Pearson, 1901)

consists on rotating the original system of coordinates to maximize dispersion
input are coordinates of datapoints in a high-dimensional space
most widely used and simple (fast) ordination method
R function: prcomp (stats)

PCoA (Principal Coordinate Analysis; Gower, 1966)

similar to PCA but first transforms distances into coordinates in a new space
input are pairwise distances between datapoints
popular in microbial ecology because it allows employing various distances
R function: cmdscale (stats)

NMDS (Non-metric Multidimensional Scaling; Kruskal et al., 1964)

numerical rather than analytical method (slow(er), non-deterministic)
number of dimensions *k* are chosen *a priori*
all variance of the data is used to distribute points in a *k*-dimensional space
(Euclidean) distances in the new space are monotonically related to original distances
R function: isoMDS (MASS)

CPCoA (Constrained Principal Coordinate Analysis; Legendre and Legendre, 1998)

similar to PCoA but attempts to maximize separation between groups (env. variables)
used to address specific hypotheses (e.g. significant differences among groups)
statistical test of hypothesis by permutation procedures
R function: capscale (vegan)

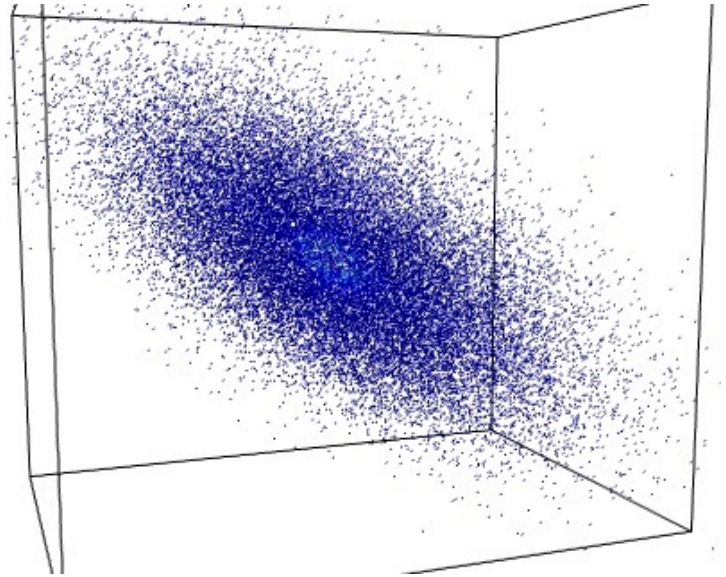
Ruben

Data analysis : β-diversity visualization

- **Principal Component Analysis**

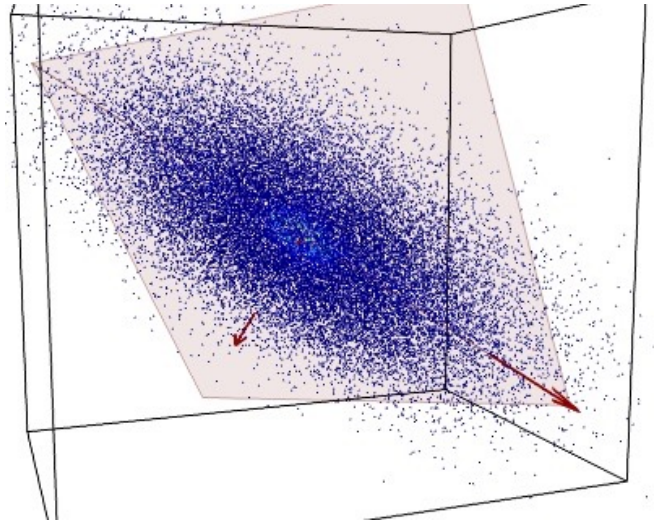
Data reduction to 2 dimensions allows the easy visualization of the data

This is how many samples and 3 species would look like in a 3D representation



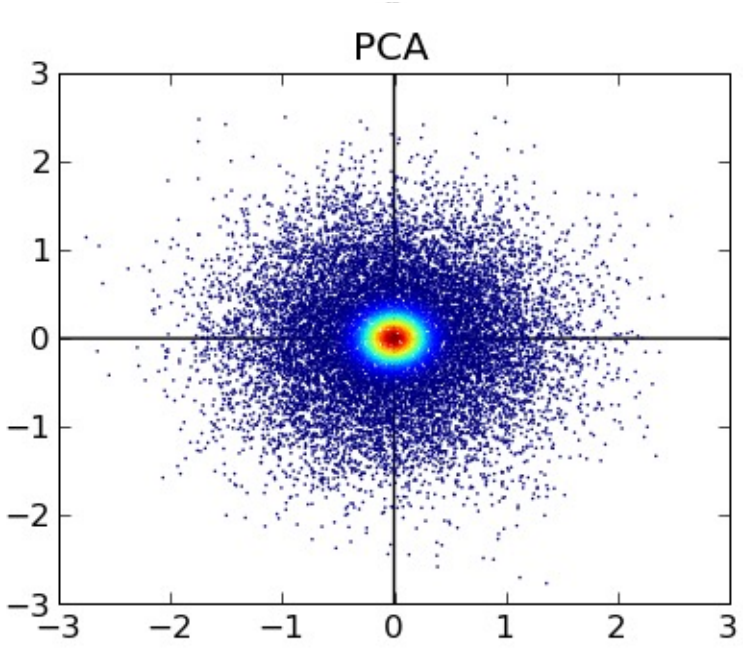
It's hard to see inside this cloud

To simplify we need to reduce the data

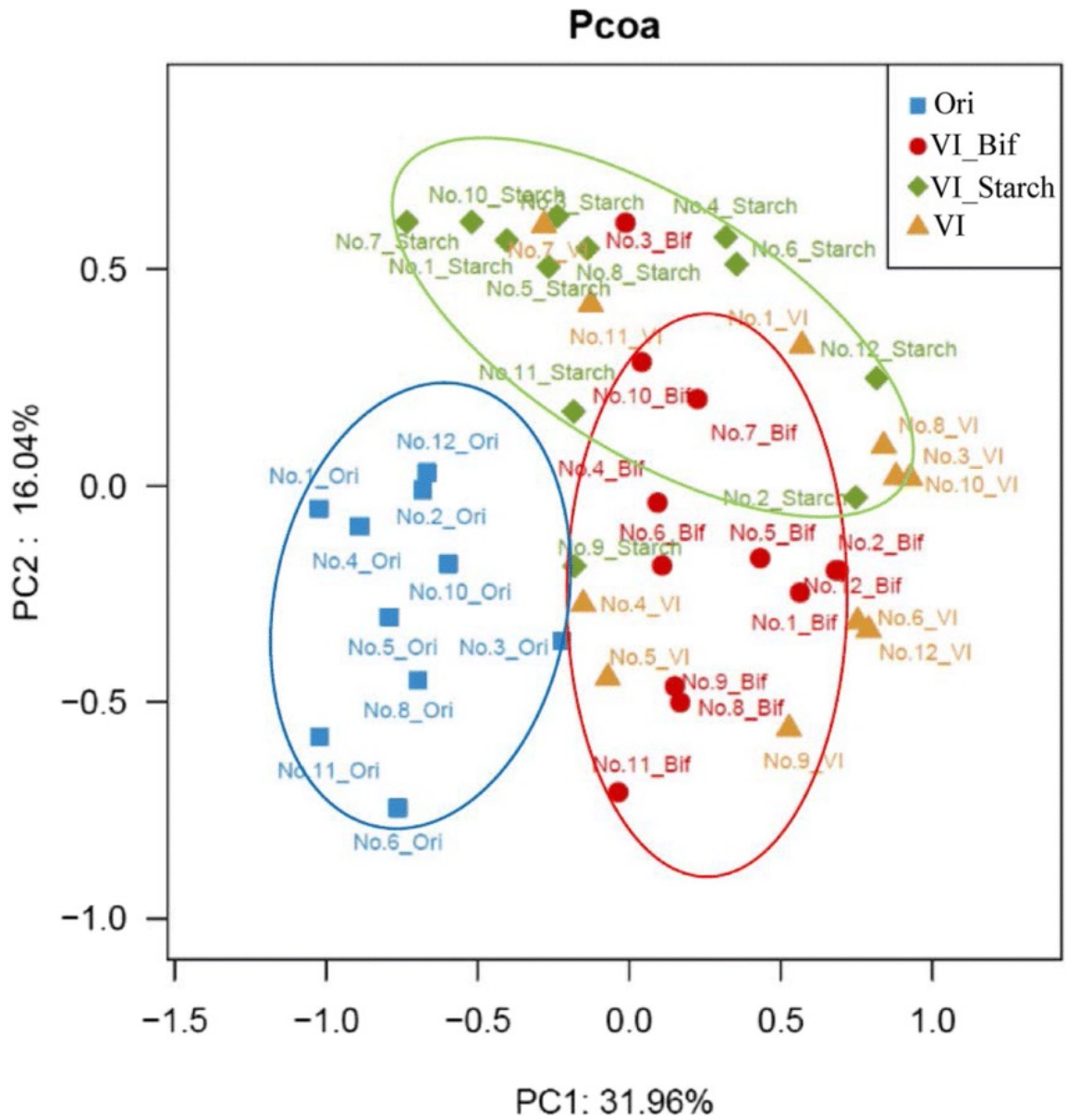


Find the best plane to compress the data to 2 dimensions

The best plane is the one capturing most of the "spread" of the cloud (data variance)



In the 2 dimensions visualization we can see inside the cloud


Pcoa
Ori
VI_Bif
VI_Starch
VI
PC2 : 16.04%
PC1: 31.96%
0.5
0.0
−0.5
−1.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
No.10_Starch
No.4_Starch
No.7_Starch
No.3_Bif
No.6_Starch
No.1_Starch
No.8_Starch
No.5_Starch
No.1_VI
No.11_VI
No.12_Starch
No.11_Starch
No.10_Bif
No.7_Bif
No.8_VI
No.3_VI
No.10_VI
No.12_Ori
No.4_Bif
No.1_Ori
No.2_Starch
No.2_Ori
No.6_Bif
No.5_Bif
No.2_Bif
No.4_Ori
No.9_Starch
No.10_Ori
No.12_Bif
No.4_VI
No.1_Bif
No.6_VI
No.5_Ori
No.3_Ori
No.12_VI
No.5_VI
No.9_Bif
No.8_Ori
No.8_Bif
No.11_Ori
No.11_Bif
No.9_VI
No.6_Ori

Ordination methods in microbial ecology

PCA (Principal Component Analysis; Pearson, 1901)

consists on rotating the original system of coordinates to maximize dispersion

input are coordinates of datapoints in a high-dimensional space

most widely used and simple (fast) ordination method

R function: prcomp (stats)

PCoA (Principal Coordinate Analysis; Gower, 1966)

similar to PCA but first transforms distances into coordinates in a new space

input are pairwise distances between datapoints

popular in microbial ecology because it allows employing various distances

R function: cmdscale (stats)

NMDS (Non-metric Multidimensional Scaling; Kruskal et al., 1964)

numerical rather than analytical method (slow(er), non-deterministic)

number of dimensions *k* are chosen *a priori*

all variance of the data is used to distribute points in a *k*-dimensional space

(Euclidean) distances in the new space are monotonically related to original distances

R function: isoMDS (MASS)

CPCoA (Constrained Principal Coordinate Analysis; Legendre and Legendre, 1998)

similar to PCoA but attempts to maximize separation between groups (env. variables)

used to address specific hypotheses (e.g. significant differences among groups)

statistical test of hypothesis by permutation procedures

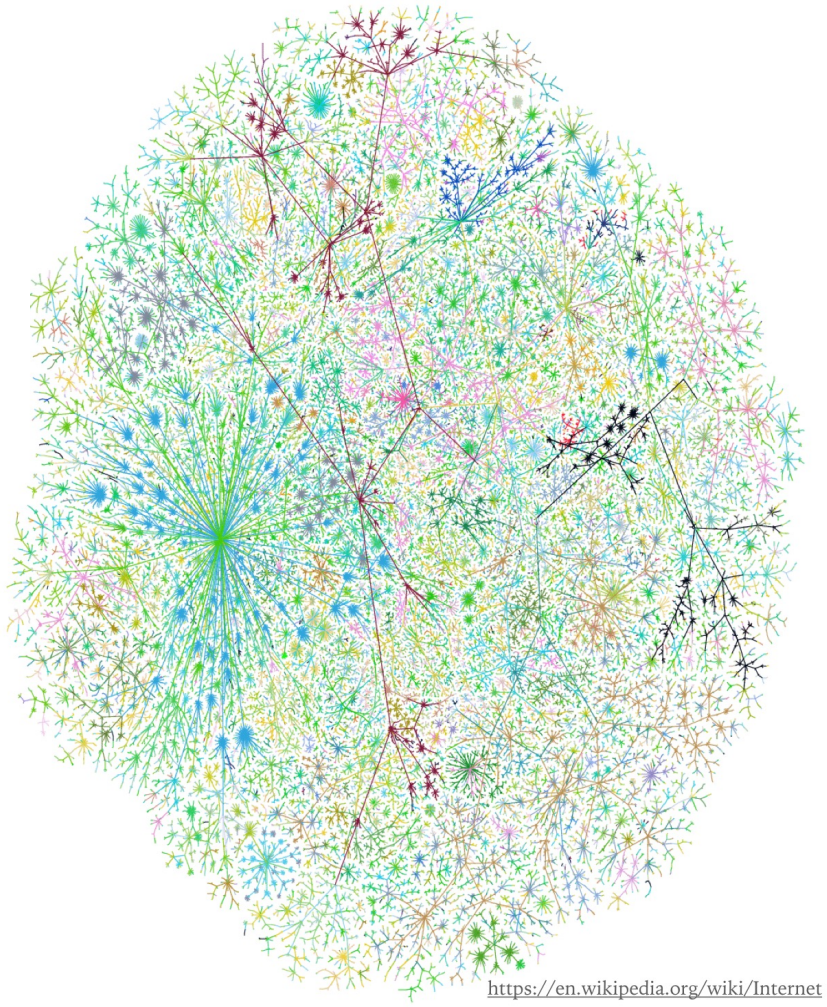
R function: capscale (vegan)

MICROBIAL NETWORKS

What we can and can't learn

ALL AROUND US

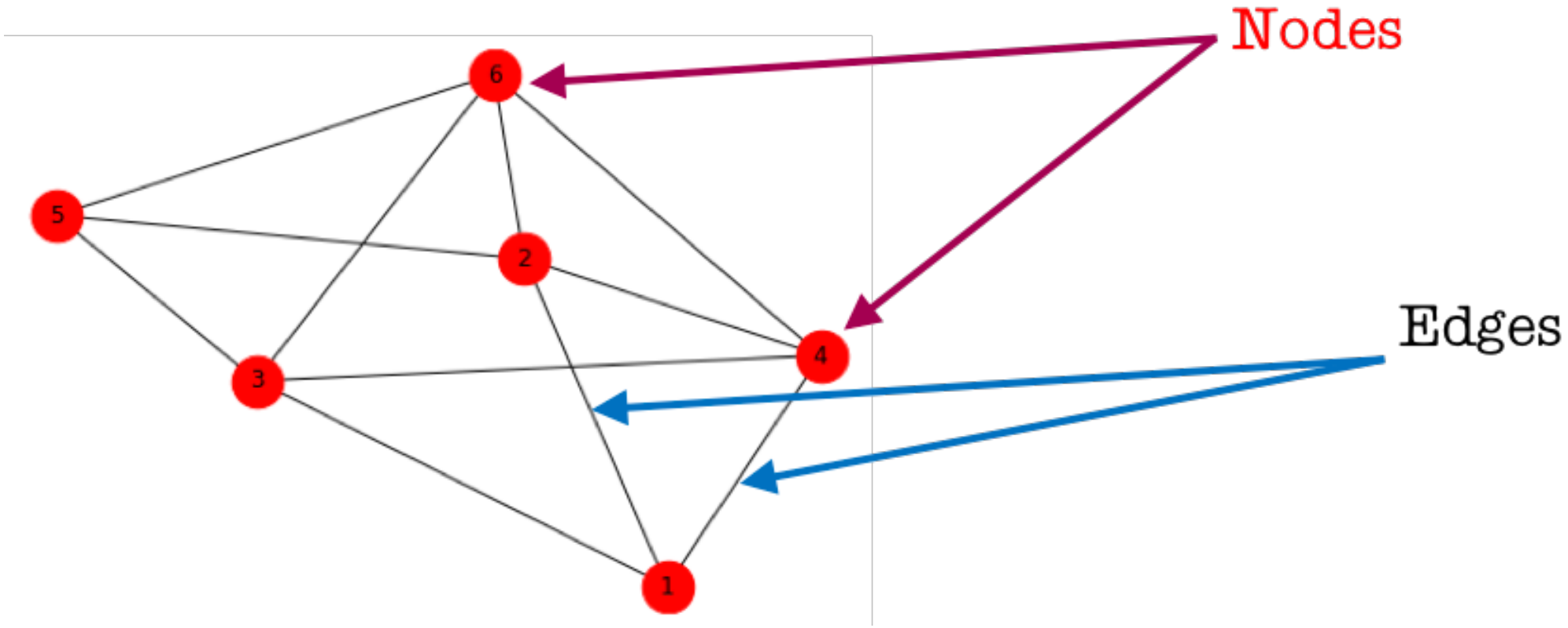
- 6 hand shakes
- Internet
- SN
- Paper citations
- Foodwebs
- Genes
- Microbes



https://en.wikipedia.org/wiki/Internet

What is a Network?

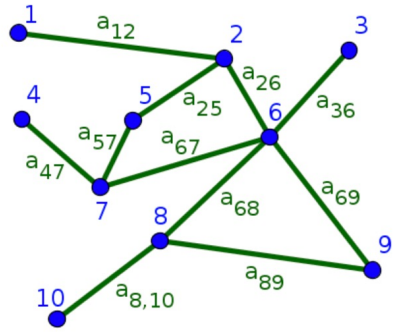
A network refers to a structure representing a group of objects/people and relationships between them. It is also known as a graph in mathematics.

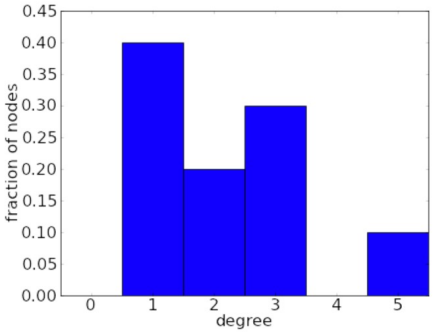


The degree distribution of a network
Suggested background

A network can be an exceedingly complex structure, as the connections among the nodes can exhibit complicated patterns. One challenge in studying complex networks is to develop simplified measures that capture some elements of the structure in an understandable way.

$P_{\deg}(k)$ = fraction of nodes in the graph with degree k.





For this undirected network, the degrees are
$k_1 = 1$,
$k_2 = 3$,
$k_3 = 1$,
$k_4 = 1$,
$k_5 = 2$,
$k_6 = 5$,
$k_7 = 3$,
$k_8 = 3$,
$k_9 = 2$, and
$k_{10} = 1$. Its degree distribution is
$P_{\deg}(1) = 2/5$,
$P_{\deg}(2) = 1/5$,
$P_{\deg}(3) = 3/10$,
$P_{\deg}(5) = 1/10$, and all other
$P_{\deg}(k) = 0$.

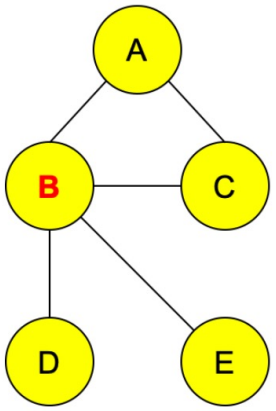
Closeness centrality

Closeness centrality is a useful measure that estimates how fast the flow of information would be through a given node to other nodes.

Closeness centrality measures how short the shortest paths are from node *i* to all nodes. It is usually expressed as the normalised inverse of the sum of the topological distances in the graph (see equation at the top of Figure 28). This sum is also known as the farness of the nodes. Sometimes closeness centrality is also expressed simply as the inverse the farness (13, 14). In the example shown on the bottom half of the figure, you can see the distances matrix for the graph on the left and the calculations to get the closeness centrality on the right. Node *B* is the most central node according to these parameters.

$$CC(i) = \frac{N-1}{\sum_j d(i,j)}$$

where
$i \neq j$,
d_{ij} is the length of the shortest path between nodes i and j in the network,
N is the number of nodes.



	A	B	C	D	E
A	0	1	1	2	2
B	1	0	1	1	1
C	1	1	0	2	2
D	2	1	2	0	2
E	2	1	2	2	0

farness $\sum_{j=1}^{n} d(i,j)$	$CC(i) = \frac{N-1}{\sum_j d(i,j)}$
6	(5-1)/6= 0.67
4	1.00
6	0.67
7	0.57
7	0.57

N = 5 (# of nodes)

sums the distance from a node to every other node, and graph centrality

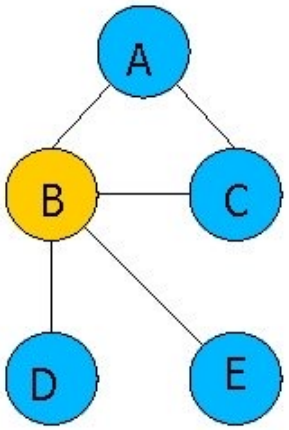
Betweenness centrality

Betweenness centrality is based on communication flow. Nodes with a high betweenness centrality are interesting because they lie on communication paths and can control information flow.
These nodes can represent important proteins in signalling pathways and can form targets for drug discovery.

$$\mathrm{BC}(v) = \sum_{s,t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

betweenness centrality (4) measures the fraction of shortest paths passing through a vertex.

Betweenness centrality



- Shortest paths are:
  - AB, AC, ABD, ABE, BC, BD, BE, CBD, CBE, DBE

$$\rho(A,B,D)=1; \quad \rho(A,D)=1$$
$$\rho(A,B,E)=1; \quad \rho(A,E)=1$$
$$\rho(C,B,D)=1; \quad \rho(B,D)=1$$
$$\rho(C,B,E)=1; \quad \rho(C,E)=1$$
$$\rho(D,B,E)=1; \quad \rho(D,E)=1$$

  - B has a BC of 5

KEYSTONES, UMBRELLAS, APEX

- Keystone:
  - “A species with disproportionately large effect on its natural environment related to its abundance ”
- Umbrellas:
  - “A species that, when protected, protects indirectly other species that make up ecological communities of its habitat”
- Apex (predator):
  - “A species which does not have natural predators in nature, and therefore is on top of the foodchain”

2.Network Inference:

How the network is born and how to interpret it

SIMILARITY, DISSIMILARITY AND CORRELATIONS: WHAT TO CHOOSE?

- 1. Look at the data, and spot weaknesses and strengths,
- 2. Have a clear goal of what we want to find, and which properties do we want to analyse

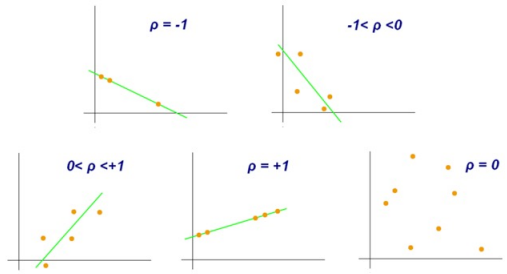
Similarities

Mutual information

one point through another

Correlations

Pearson



Spearman

Kendall

parameters and linearity

sensitive to compositionality,

resistant to outliers

Dissimilarities

Bray-curtis

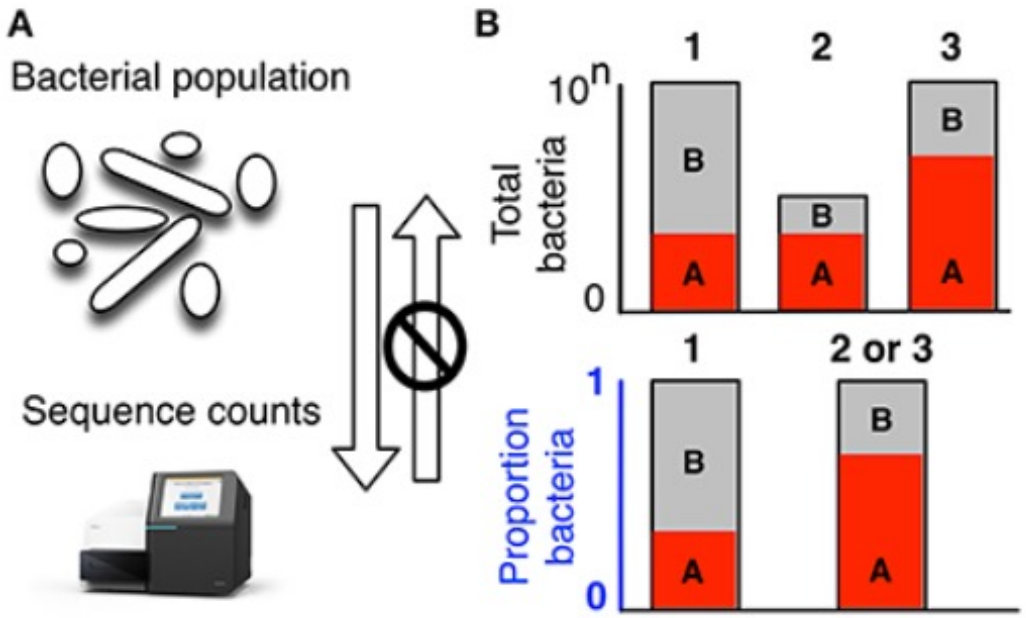
Euclidean

Kullback-Leibler

sensitive to outliers,

resistant to compositionality

Figure 1



High-throughput sequencing data are compositional. (A) illustrates that the data observed after sequencing a set of nucleic acids from a bacterial population cannot inform on the absolute abundance of molecules. The number of counts in a high throughput sequencing (HTS) dataset reflect the proportion of counts per feature (OTU, gene, etc.) per sample, multiplied by the sequencing depth. Therefore, only the relative abundances are available. The bar plots in (B)show the difference between the count of molecules and the proportion of molecules for two features, A (red) and B (gray) in three samples. The top bar graphs show the total counts for three samples, and the height of the color illustrates the total count of the feature. When the three samples are sequenced we lose the absolute count information and only have relative abundances, proportions, or “normalized counts” as shown in the bottom bar graph. Note that features A and B in samples 2 and 3 appear with the same relative abundances, even though the counts in the environment are different. The table below in (C) shows real and perceived changes for each sample if we transition from one sample to another.

SparCC (Friedman and Alm, 2012) is particularly designed to deal with compositional data, as it is based on Aitchison's log-ratio analysis

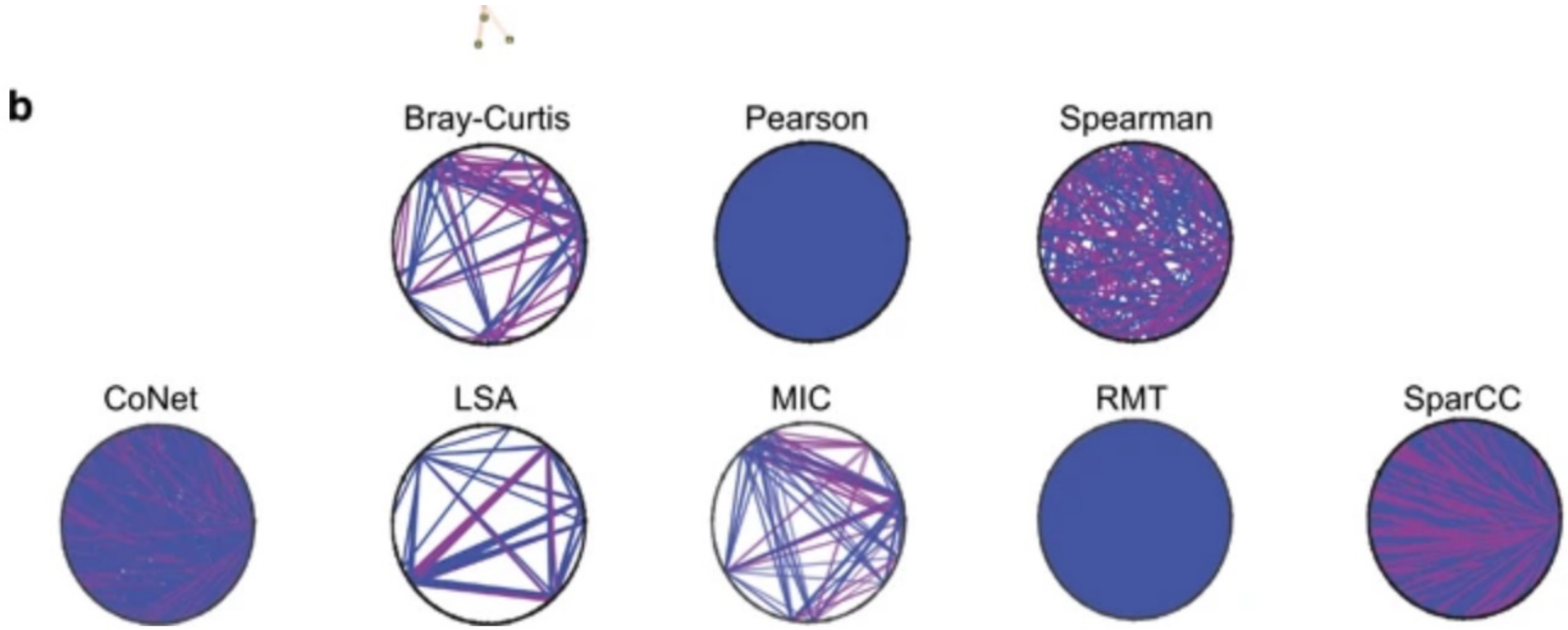
Like most compositional data analysis techniques, SparCC is based on the log-ratio transformation:

$$y_{ij} = \log \frac{x_i}{x_j} = \log x_i - \log x_j, \tag{1}$$

where x_i is the fraction of OTU i. This transformation carries several advantages: First, the new variables y_{ij} contain information regarding the true abundances of OTUs, as the ratio of fractions is equal to the ratio of the true abundances. Second, unlike the fractions themselves, the ratio of the fractions of two OTUs is independent of which other OTUs are included in the analysis, a property termed subcompositional coherence. Third, this transformation is mathematically convenient, as the new variables y_{ij} are no longer limited to the simplex, but are free to assume any real value. Taking the logarithm removed the positivity constraint, and induces (anti) symmetry in the treatment of the variables.

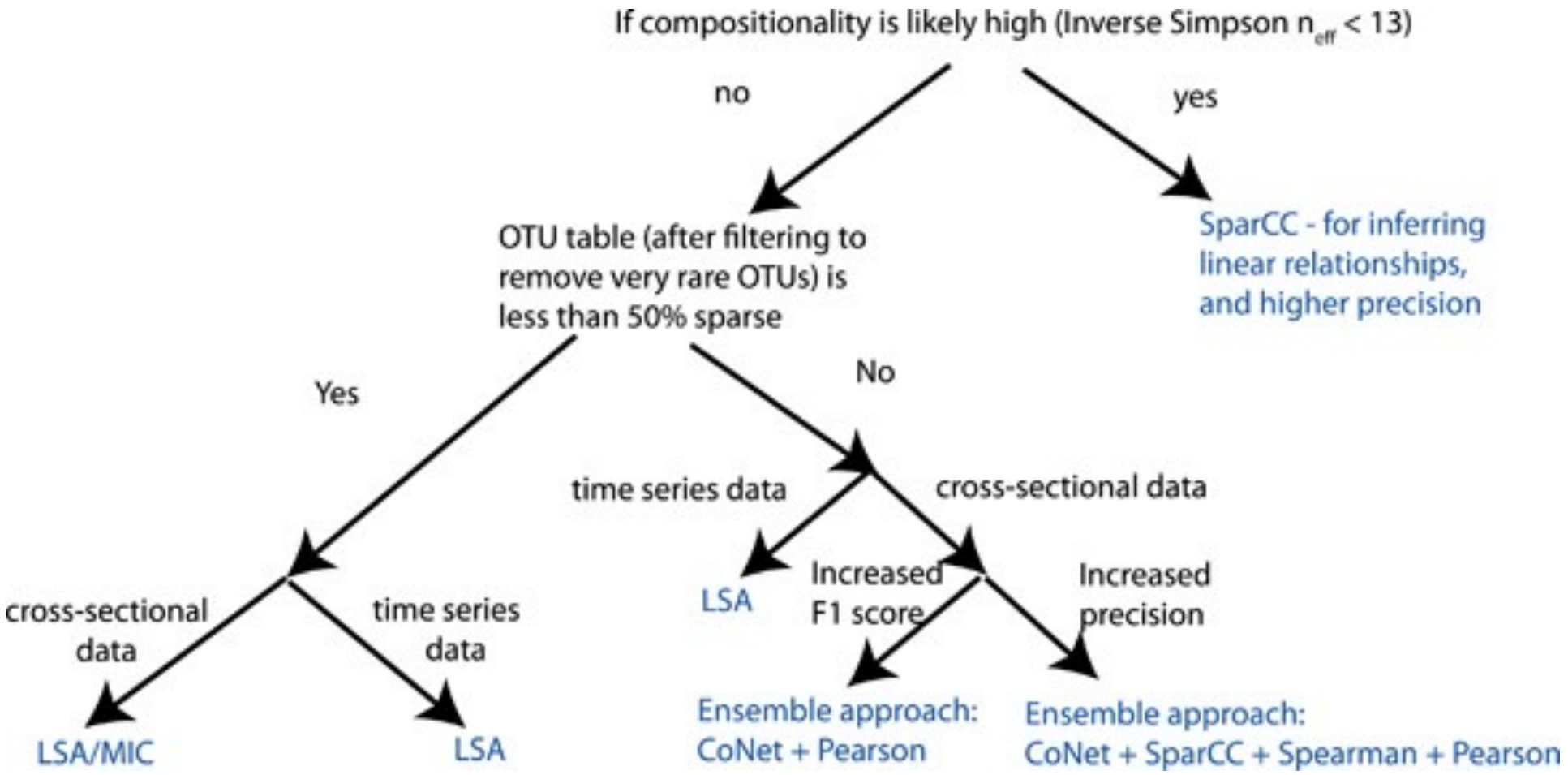
To describe the dependencies in a compositional dataset, Aitchison suggested using the quantity

$$t_{ij} \equiv \mathrm{Var}\left[\log \frac{x_i}{x_j}\right] = \mathrm{Var}\left[y_{ij}\right],$$



Overview and motivation of correlation network technique benchmarking. (**a**) Mathematical properties of microbial communities naturally present in the environment are simulated in different feature × sample tables. These tables are evaluated for significant feature correlation networks by different metrics and toolkits. The networks are then assessed for accuracy. (**b**) Correlation tools find very different significant pairs on the same data set. A blue (pink) line connects significant positively (negatively) correlated OTU pairs.

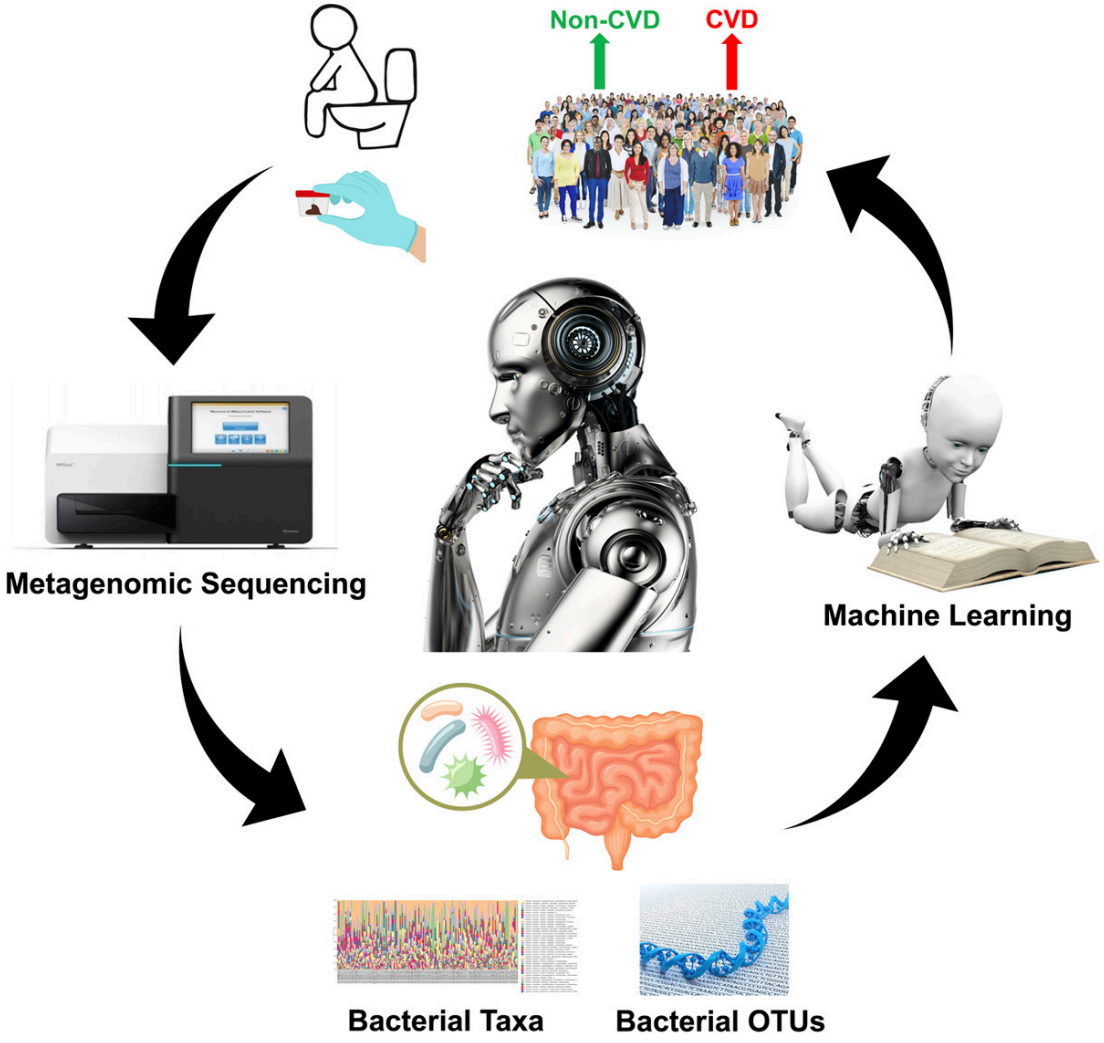
Correlation detection strategies in microbial data sets vary widely in sensitivity and precision

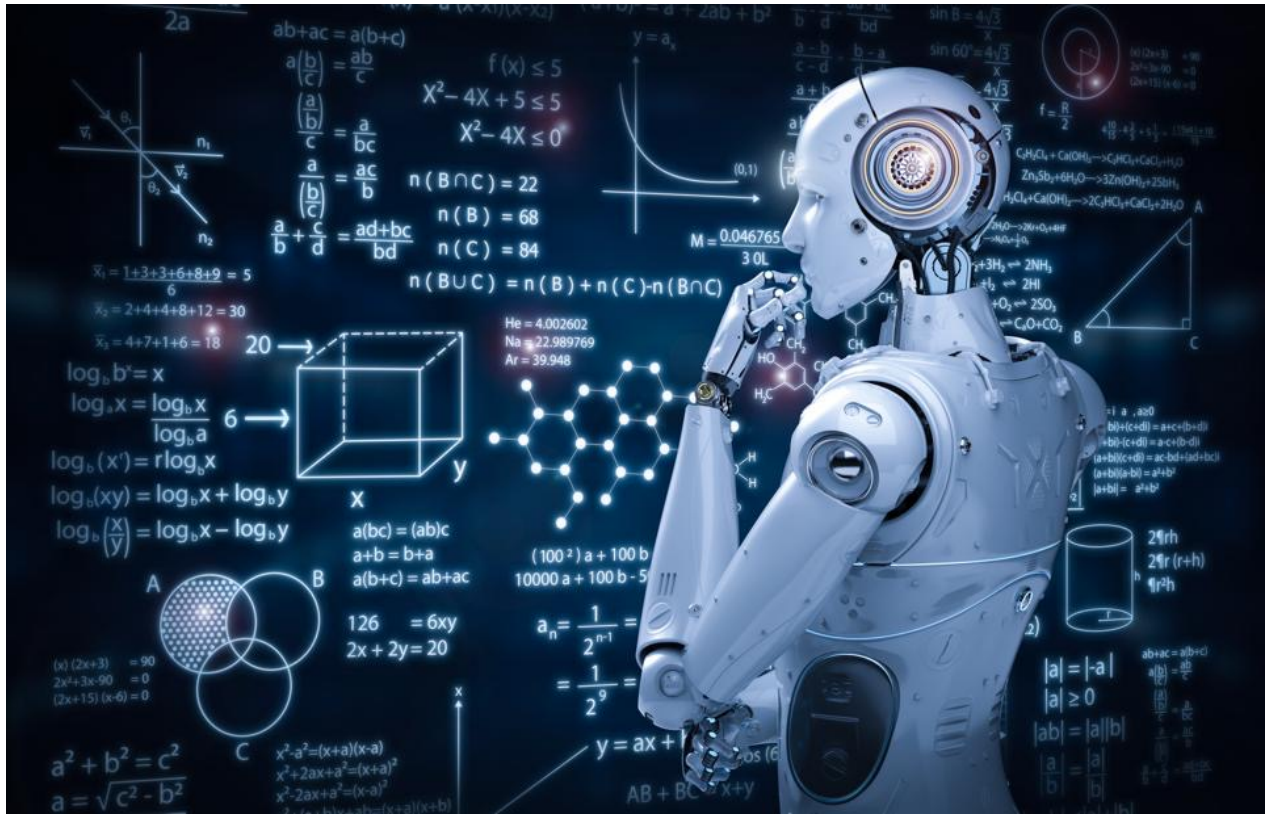


TYPE OF NETWORKS II: QUESTION TIME

- What should we consider when looking for hubs? what centrality measures?
- Once we find a hub, what are we looking at?
- In which case a hub can be an umbrella species?
- In which case a hub can be a keystone?
- What ecological conclusion can we draw after identifying a hub?
- Can you conceive experiments to assess those?

Machine learning methods for microbiome studies





Part of the images are from Shutterstock.com (Artists: Rawpixel.com; Odrik; Julia Pankin; Phonlamai Photo; Vasilyeva Larisa; Sarah Holmlund; Mopic).

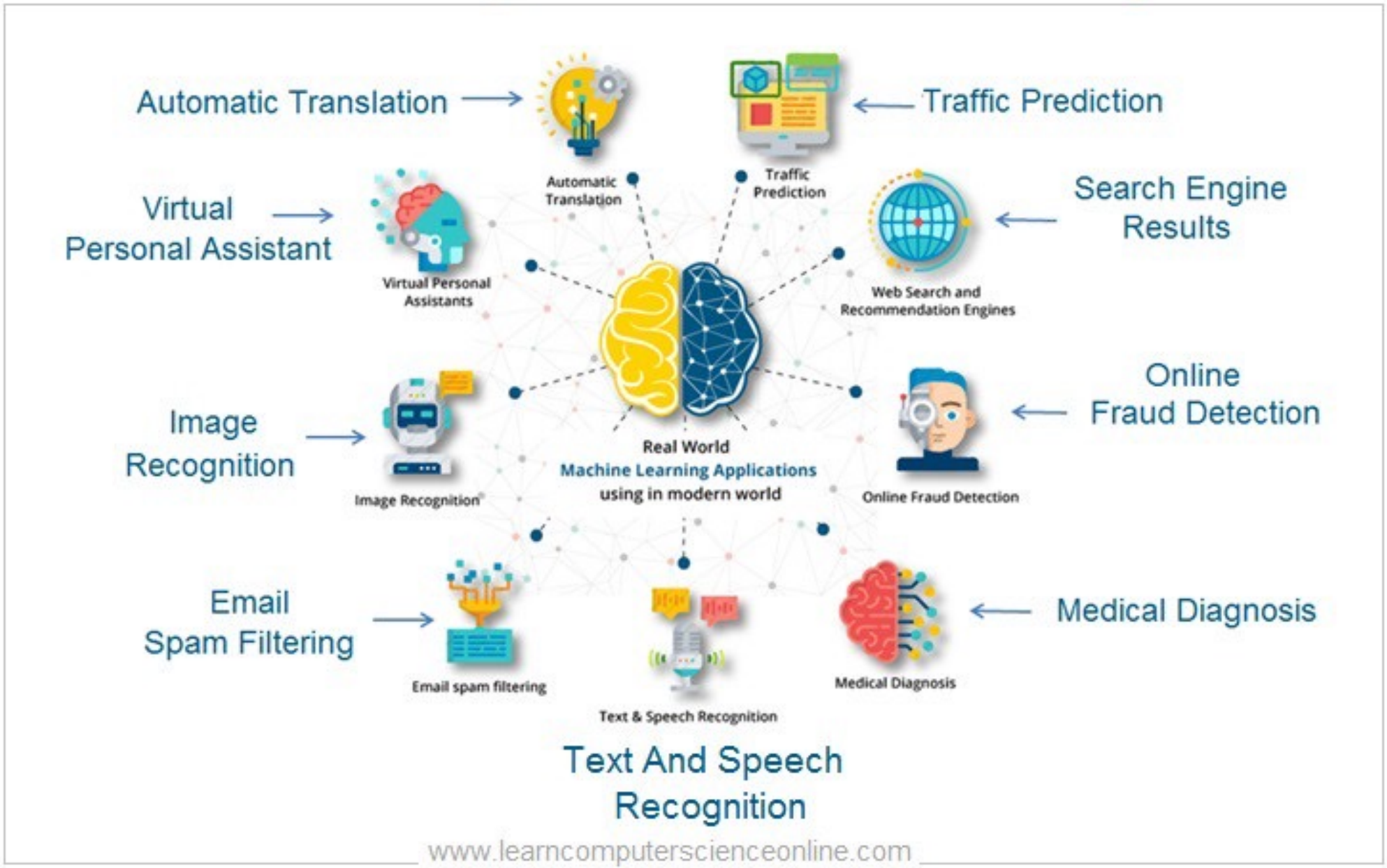
What is Machine learning:

Machine learning (ML) is defined as a discipline of artificial intelligence (AI) that provides machines the ability to automatically learn from data and past experiences to identify patterns and make predictions with minimal human intervention

Why is Machine Learning Important?

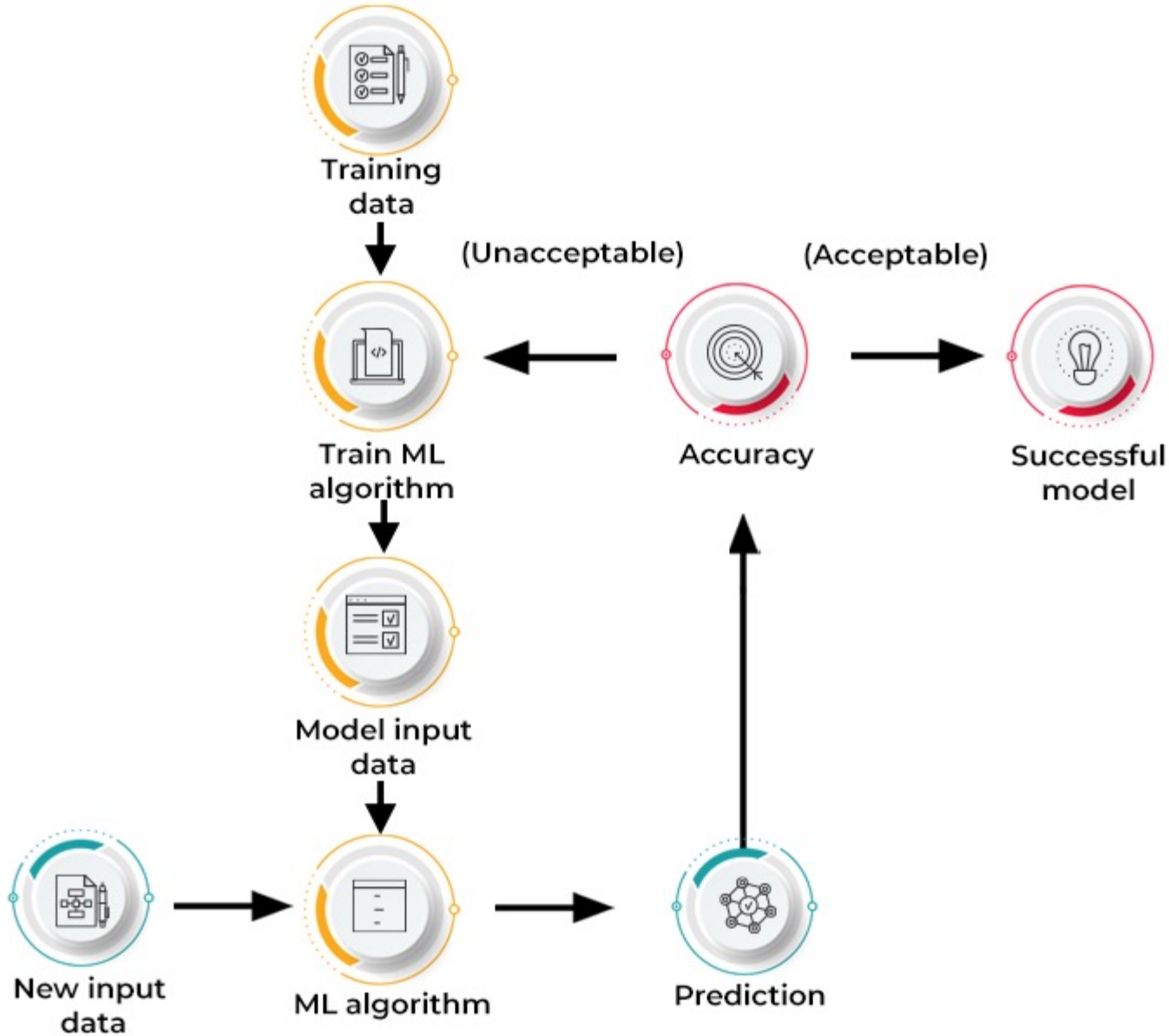
Why use machine learning? Machine learning is growing in importance **due to increasingly enormous volumes and variety of data**, the access and affordability of computational power, and the availability of high speed Internet. These digital transformation factors make it possible for one to **rapidly and automatically develop models that can quickly and accurately** analyze extraordinarily large and complex data sets.

Real World Applications Of Machine Learning



TOOLBOX™

HOW DOES MACHINE LEARNING WORK?



TOOLBOX™

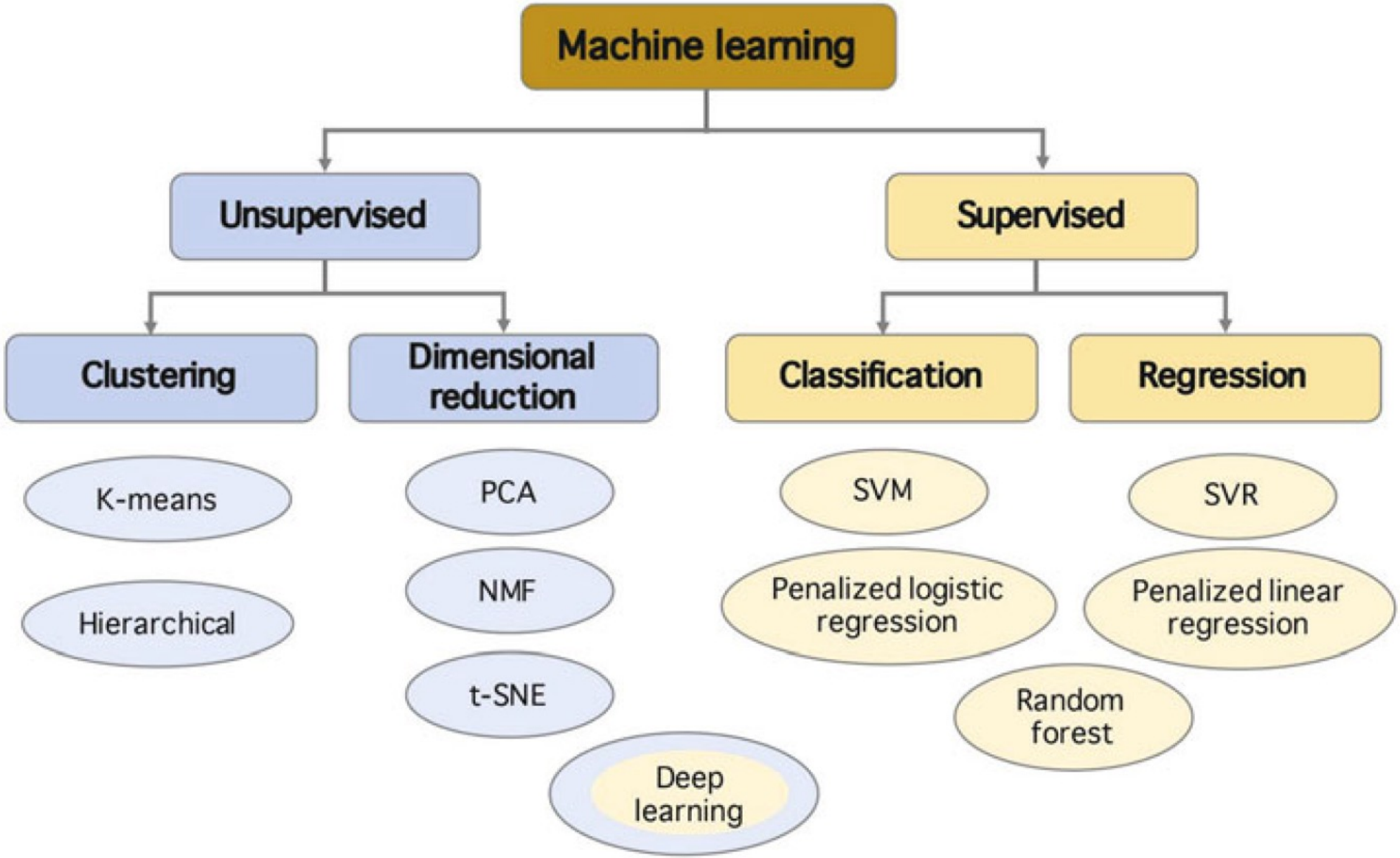
TYPES OF MACHINE LEARNING

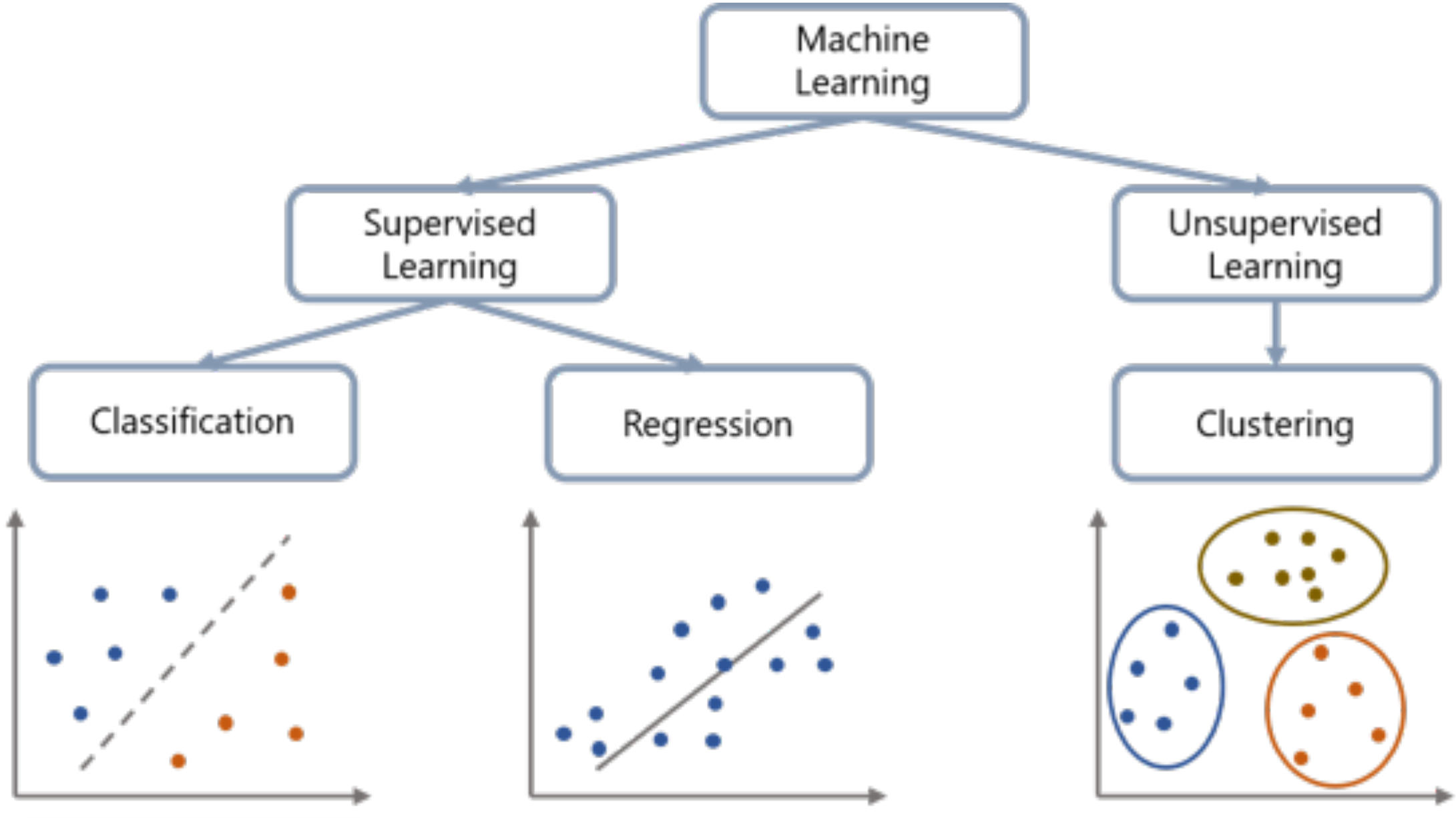
Supervised Machine Learning

Unsupervised Machine Learning

Semi-Supervised Learning

Reinforcement Learning


Machine learning
Unsupervised
Supervised
Clustering
Dimensional reduction
Classification
Regression
K-means
Hierarchical
PCA
NMF
t-SNE
Deep learning
SVM
Penalized logistic regression
Random forest
SVR
Penalized linear regression


Machine Learning
Supervised Learning
Unsupervised Learning
Classification
Regression
Clustering

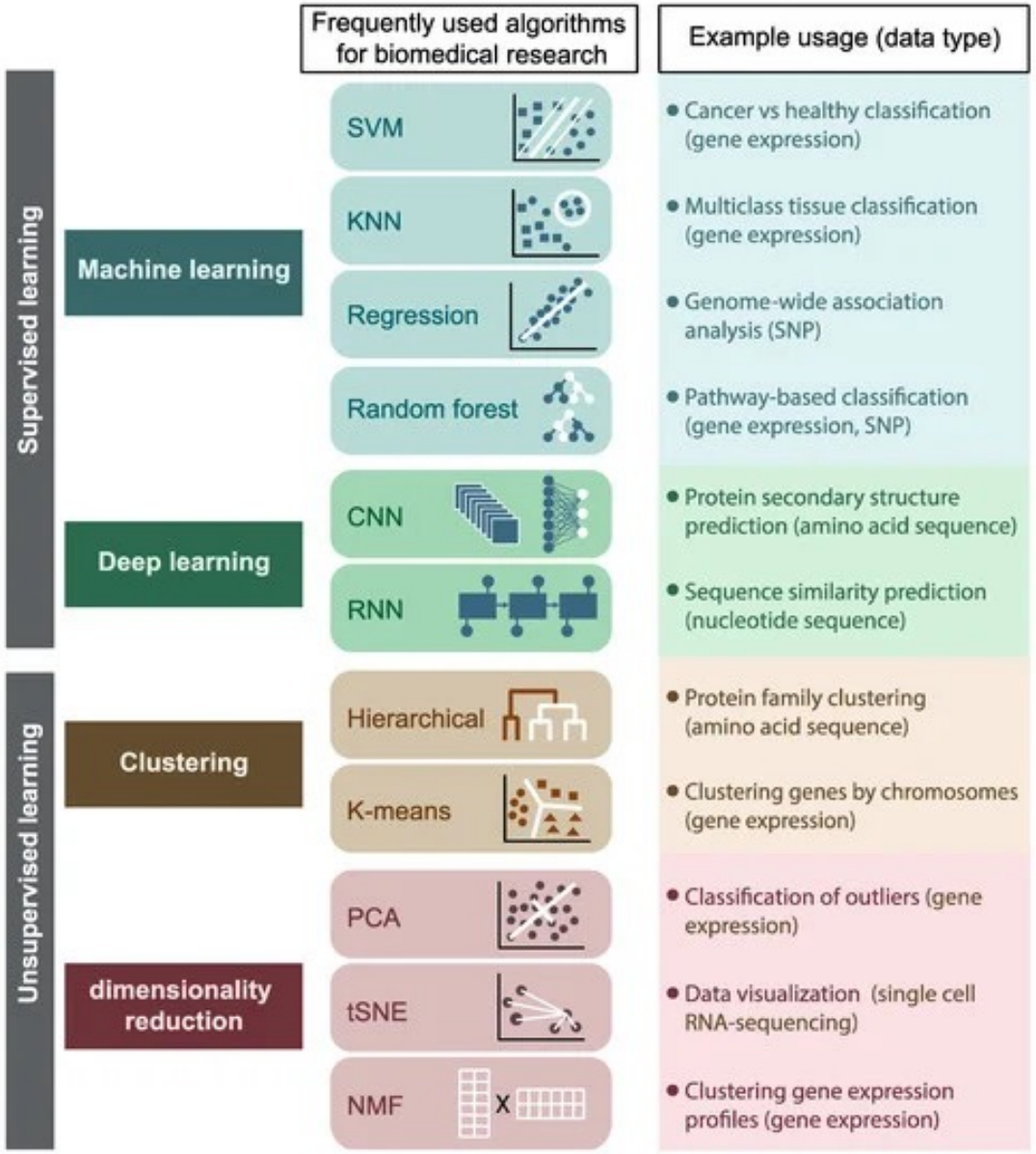
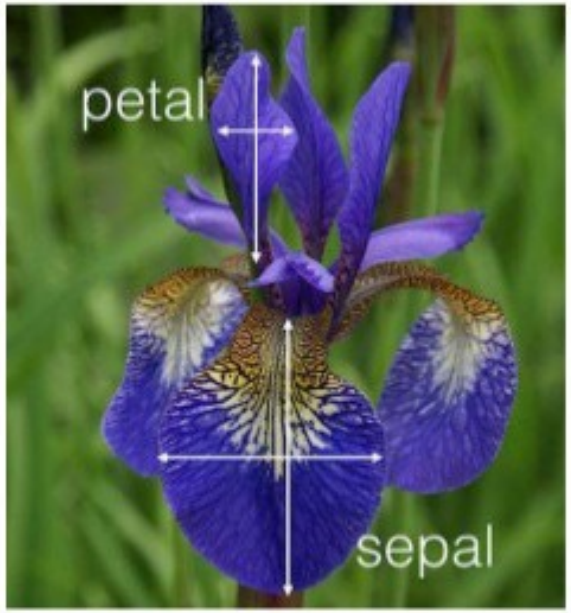


Figure 1. Machine learning algorithms frequently used in bioinformatics research. An example of the usage of each algorithm and the respective input data are indicated on the right. Abbreviations: SVM, support vector machines; KNN, K-nearest neighbors; CNN, convolutional neural networks; RNN, recurrent neural networks; PCA, principal component analysis; t-SNE, t-distributed stochastic neighbor embedding, NMF, non-negative matrix factorization.

https://doi.org/10.3390/ijms22062903

Supervised learning *classification* problem

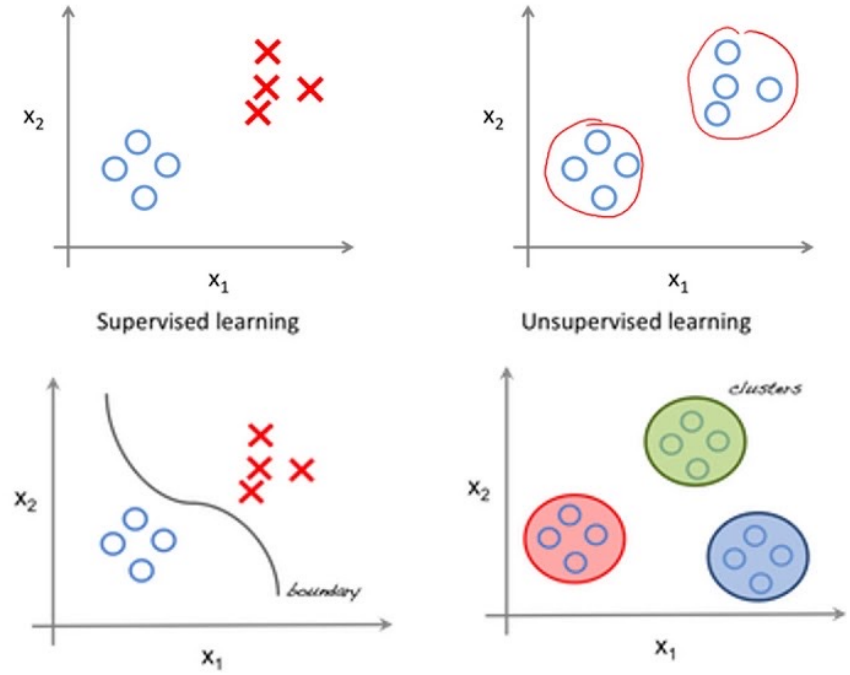
(using the Iris flower data set)



Training / test data

Features				Labels
Sepal length	Sepal width	Petal length	Petal width	Species
5.1	3.5	1.4	0.2	Iris setosa
4.9	3.0	1.4	0.2	Iris setosa
7.0	3.2	4.7	1.4	Iris versicolor
6.4	3.2	4.5	1.5	Iris versicolor
6.3	3.3	6.0	2.5	Iris virginica
5.8	3.3	6.0	2.5	Iris virginica

Supervised Learning | Unsupervised Learning



scikit-learn

Machine Learning in Python

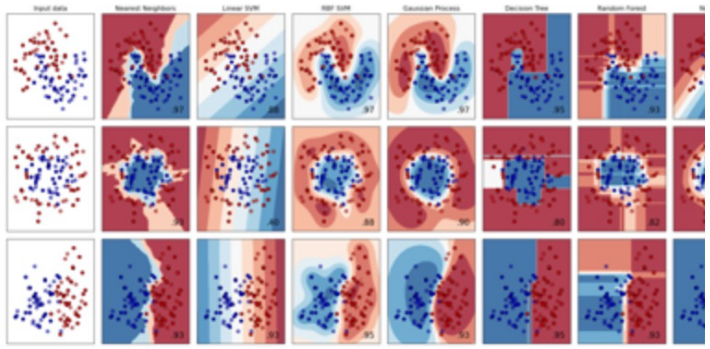
Getting Started | Release Highlights for 1.1 | GitHub

- Simple and efficient tools for predictive data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

Classification

Identifying which category an object belongs to.

Applications: Spam detection, image recognition.
Algorithms: SVM, nearest neighbors, random forest, and more...

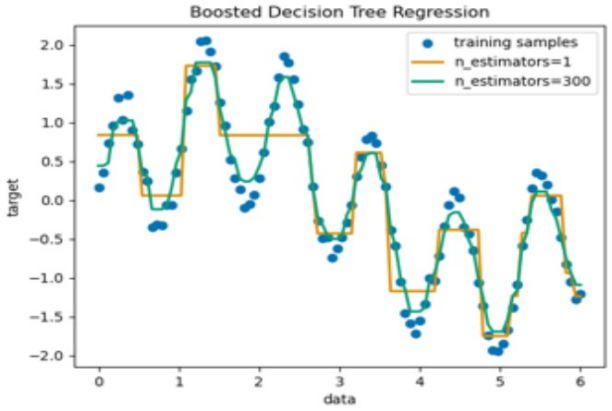


Examples

Regression

Predicting a continuous-valued attribute associated with an object.

Applications: Drug response, Stock prices.
Algorithms: SVR, nearest neighbors, random forest, and more...

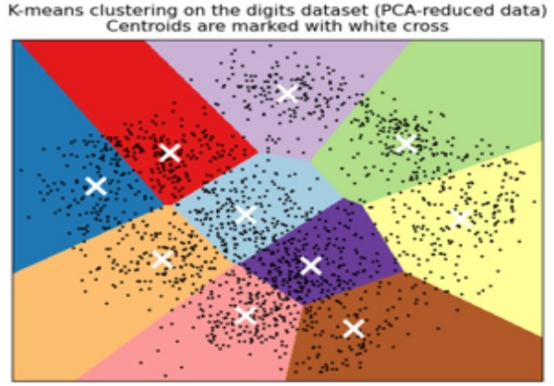


Examples

Clustering

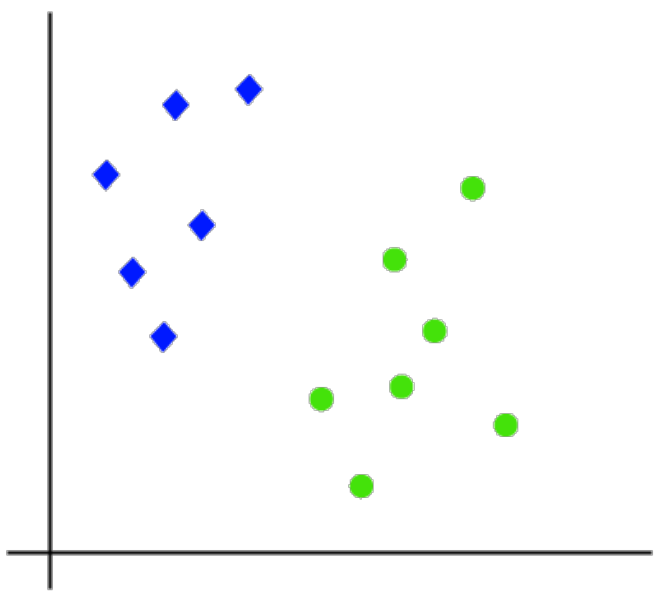
Automatic grouping of similar objects into sets.

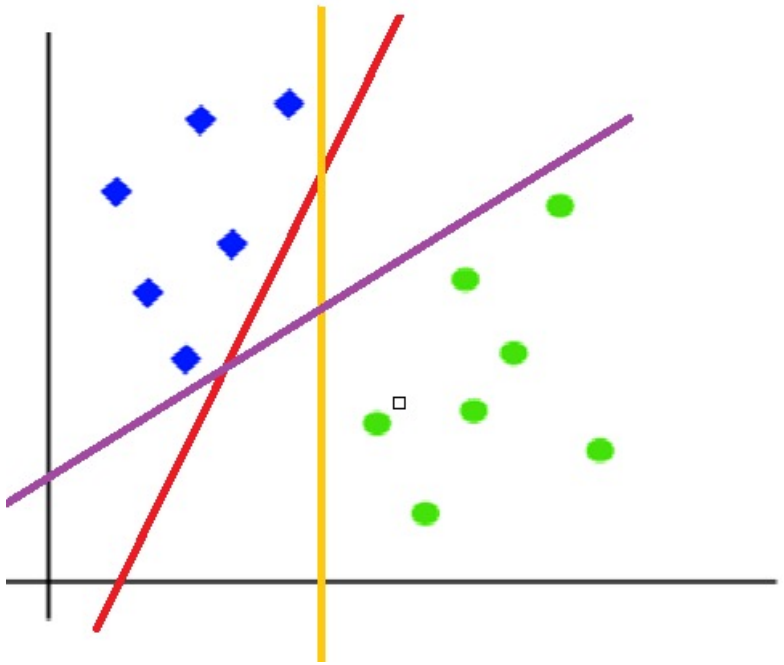
Applications: Customer segmentation, Grouping experiment outcomes
Algorithms: k-Means, spectral clustering, mean-shift, and more...

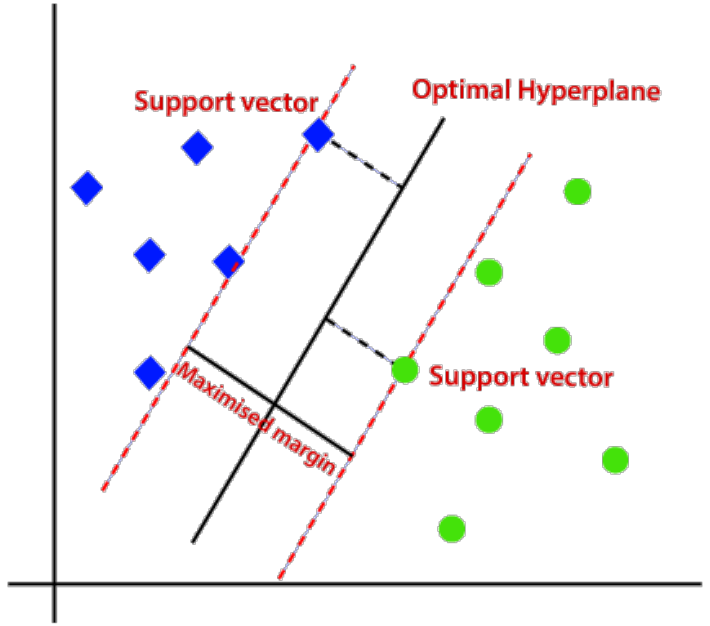


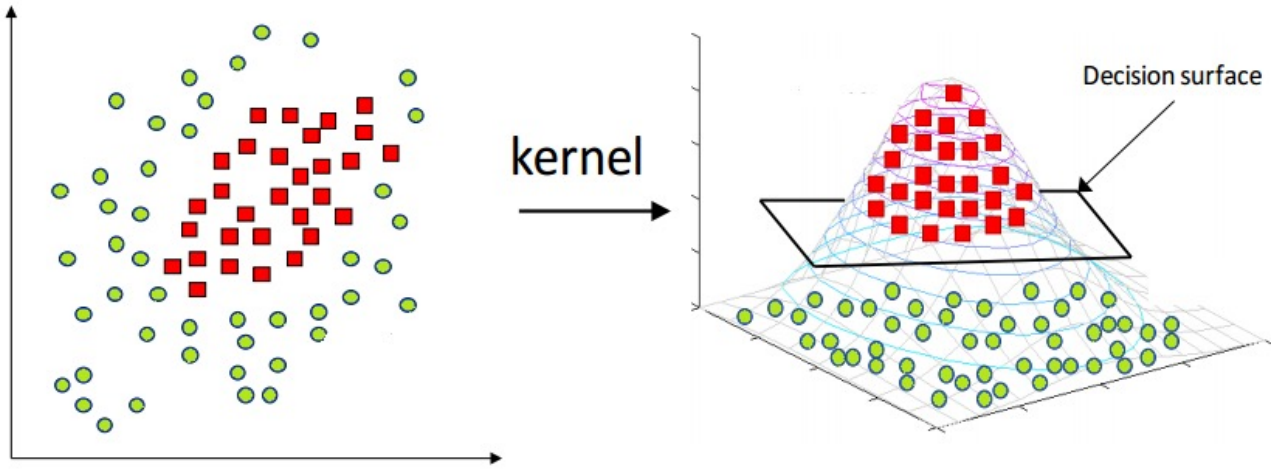
Examples

Classification using Support vector machine



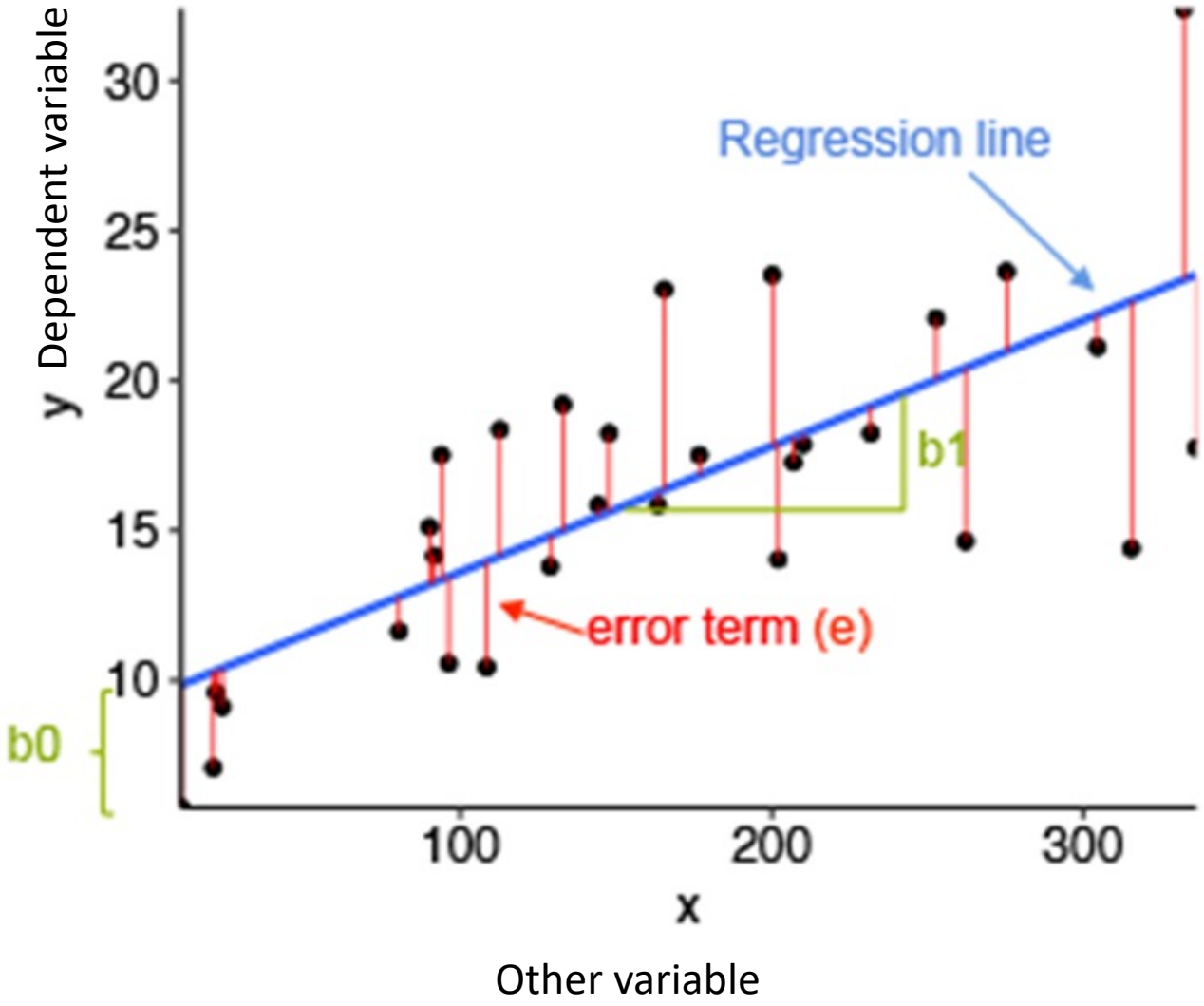


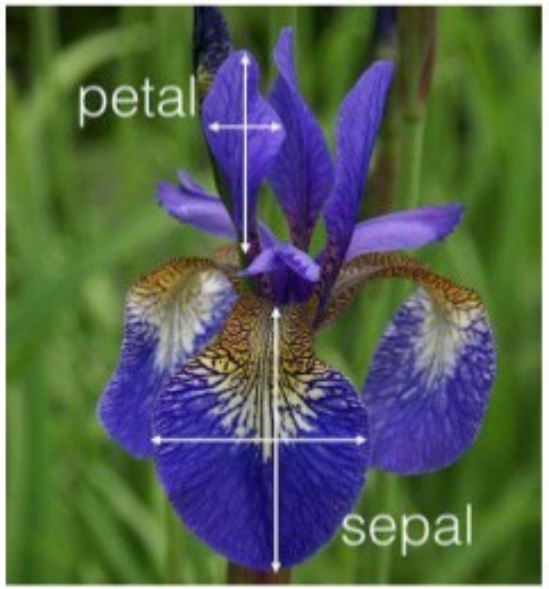




Regression:

relationship between one dependent variable (usually denoted by Y) and a series of other variables (known as independent variables).





Tra
Featu

Sepal length	Sepal width
5.1	3.5
4.9	3.0
7.0	3.2
6.4	3.2
6.3	3.3
5.8	3.3

Simple linear regression:

$$Y = a + bX + u$$

Multiple linear regression:

$$Y = a + b_1X_1 + b_2X_2 + b_3X_3 + ... + b_tX_t + u$$

where:

Y = The dependent variable you are trying to predict or explain

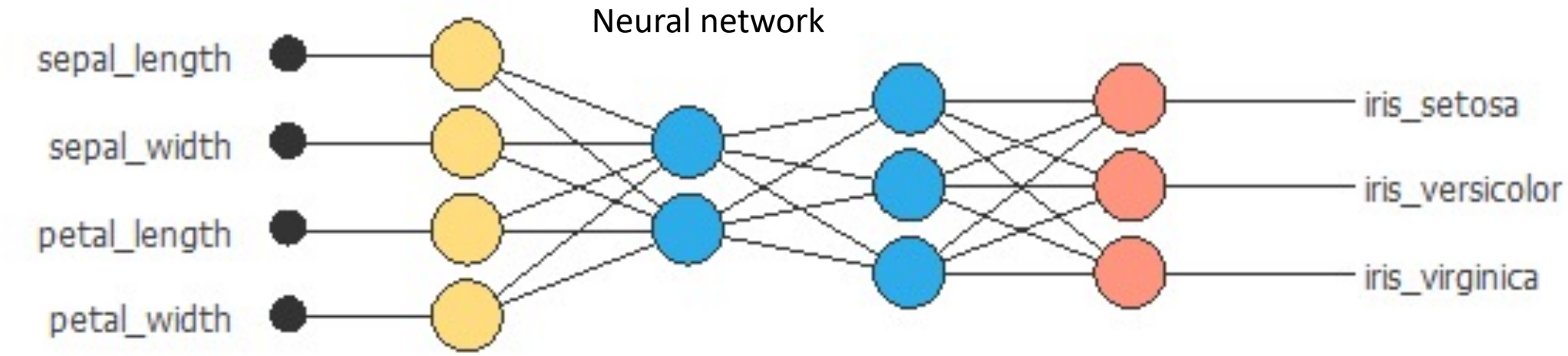
X = The explanatory (independent) variable(s) you are using to predict or associate with Y

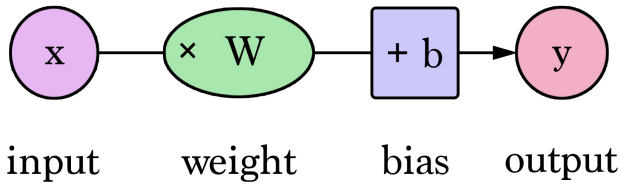
a = The y-intercept

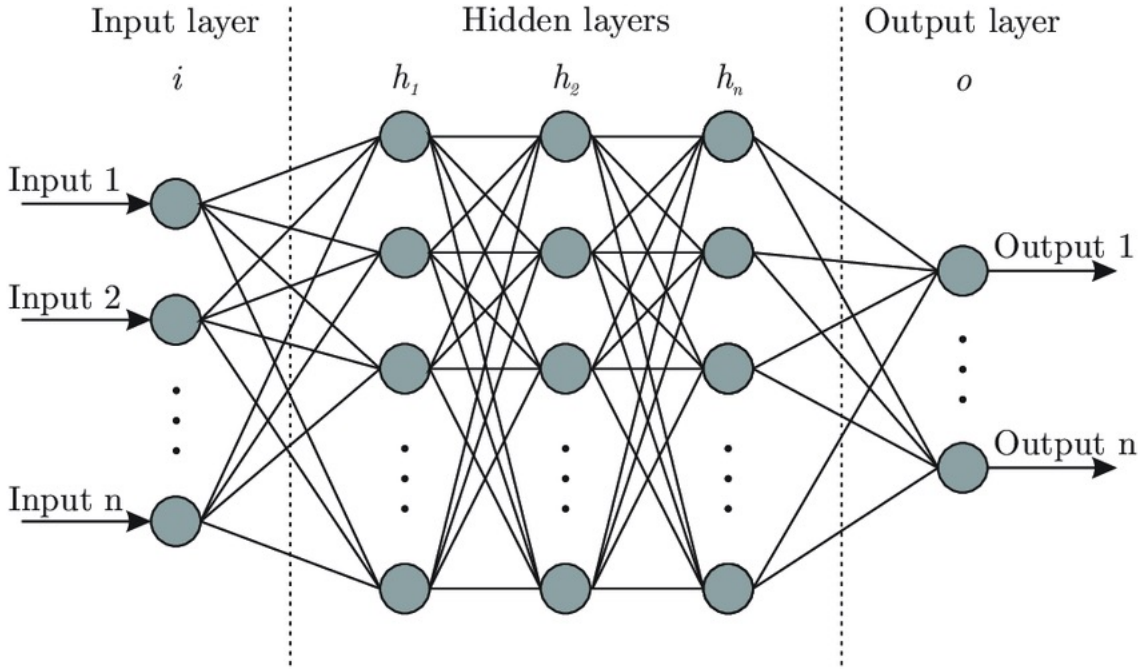
b = (beta coefficient) is the slope of the explanatory variable(s)

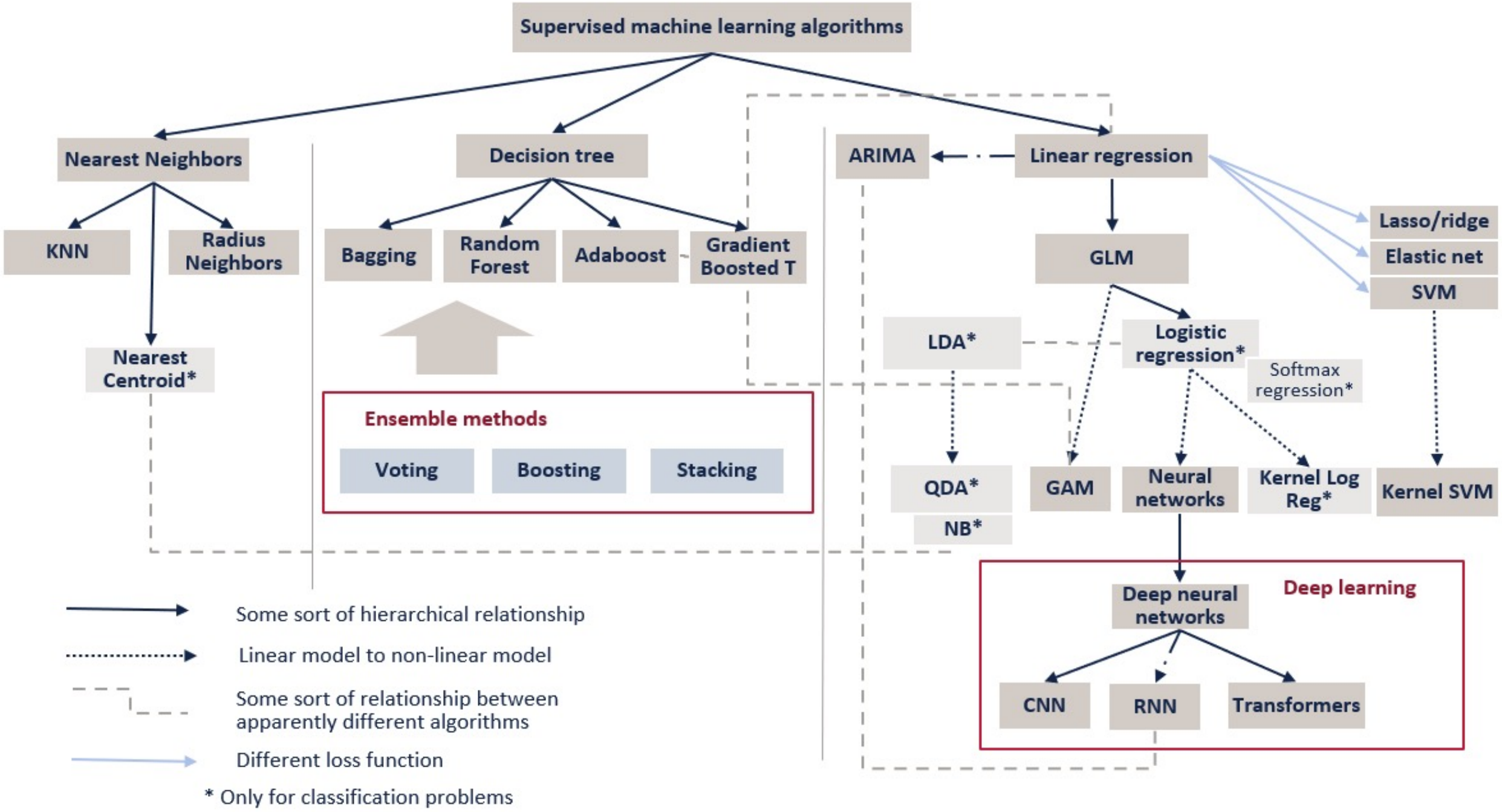
u = The regression residual or error term

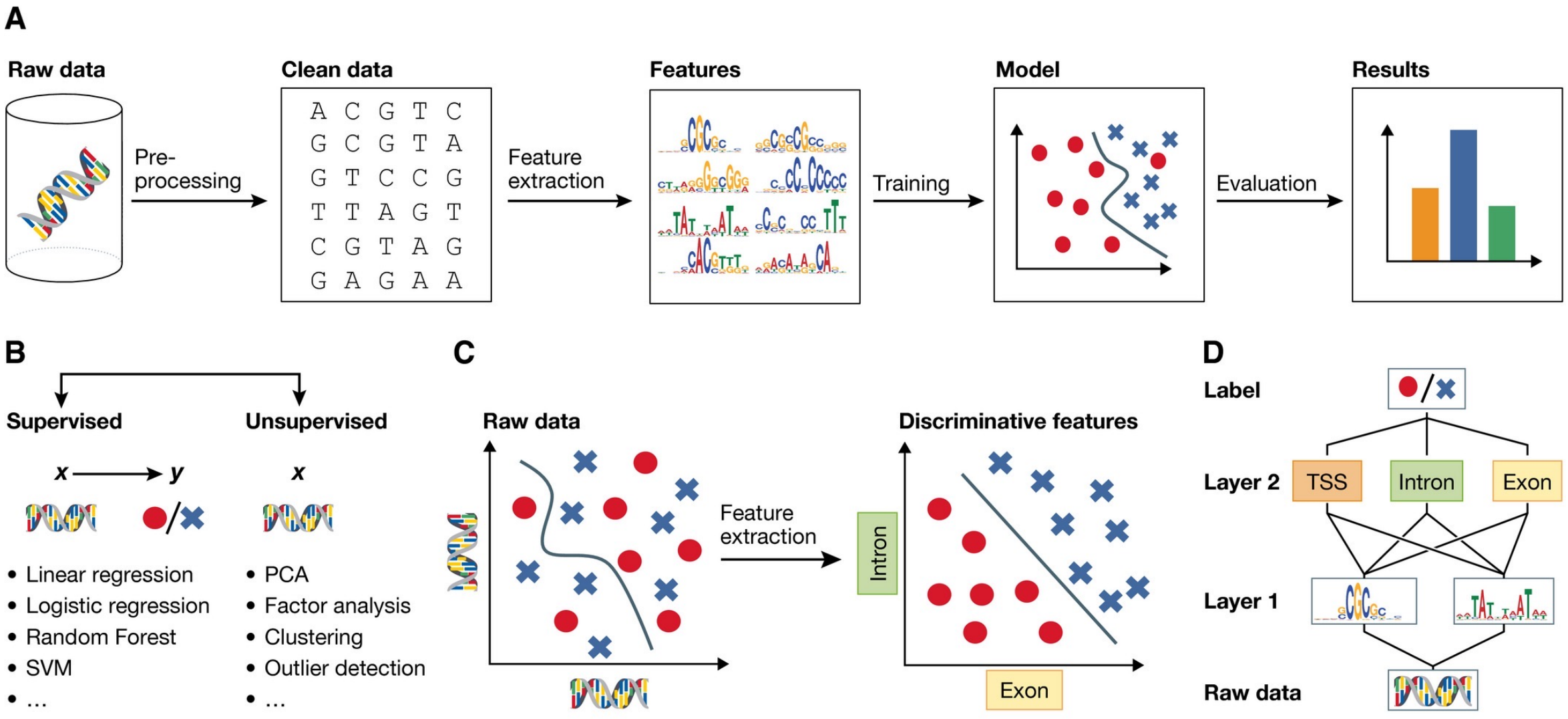
Neural network








Supervised machine learning algorithms
Nearest Neighbors
KNN
Radius Neighbors
Nearest Centroid*
Decision tree
Bagging
Random Forest
Adaboost
Gradient Boosted T
Ensemble methods
Voting
Boosting
Stacking
ARIMA
Linear regression
Lasso/ridge
Elastic net
SVM
GLM
LDA*
Logistic regression*
Softmax regression*
QDA*
NB*
GAM
Neural networks
Kernel Log Reg*
Kernel SVM
Deep learning
Deep neural networks
CNN
RNN
Transformers
Some sort of hierarchical relationship
Linear model to non-linear model
Some sort of relationship between apparently different algorithms
Different loss function
* Only for classification problems


A
Raw data
Pre-
processing
Clean data
A C G T C
G C G T A
G T C C G
T T A G T
C G T A G
G A G A A
Feature
extraction
Features
Training
Model
Evaluation
Results
B
Supervised
Unsupervised
x
y
x
• Linear regression
• Logistic regression
• Random Forest
• SVM
• …
• PCA
• Factor analysis
• Clustering
• Outlier detection
• …
C
Raw data
Feature
extraction
Discriminative features
Intron
Exon
D
Label
Layer 2
TSS
Intron
Exon
Layer 1
Raw data

Deep learning for genome