Genomics Data

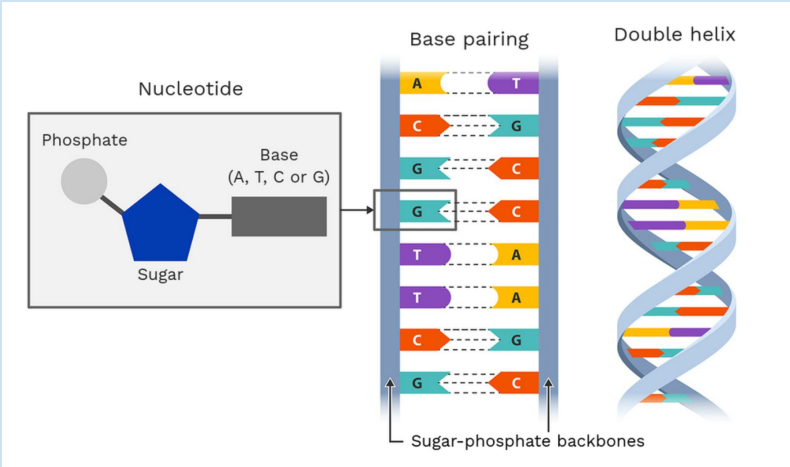
Generation, management, and analysis

TU Omics

Introduction to cell biology

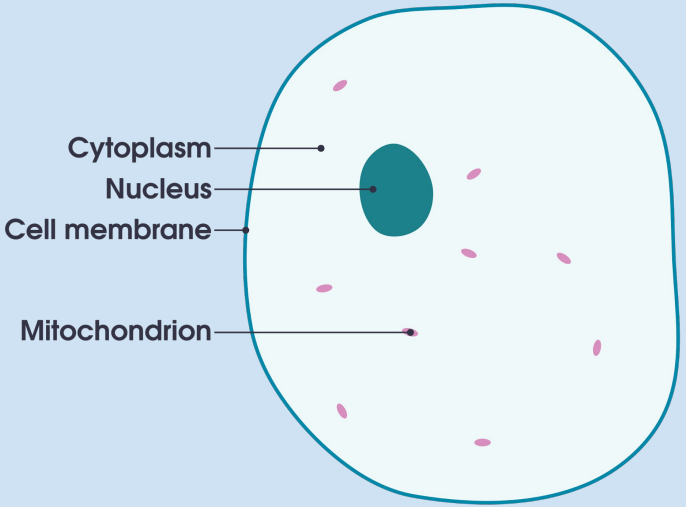
Basics of cell biology (1)

- **Genetic information is stored in DNA**
- **DNA**: long molecule formed of two chains of building blocks (nucleotides)
- **Genome**: complete set of DNA
- **Gene**: segment of DNA encoding a particular product (protein)
    - Human genome contains about 25,000 genes



Basics of cell biology (2)

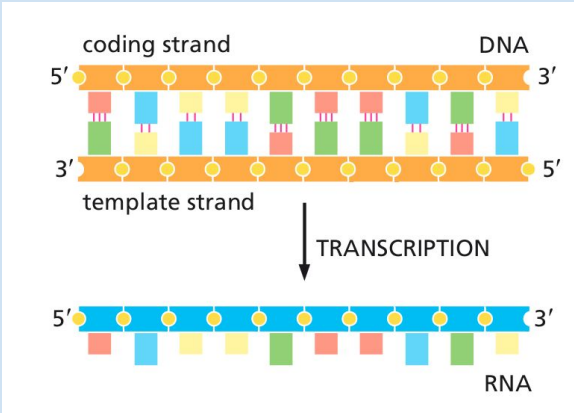
- **Cells**: fundamental units of life
- DNA is stored in the nuclei of cells
- **All cells** in an organism contain the **same DNA**
  - However, cells have diverse functions
  - Different cell types ***express*** different genes
  - Developmental history, intra- and extra-cellular conditions drive these differences



Basics of cell biology (3)

- **Central dogma of molecular biology**: genetic information flows from DNA to protein
- DNA is *transcribed* into mRNA transcripts
    - **Transcriptome**: complete set of mRNA transcripts in a population of cells
- mRNA is *translated* into proteins: functional molecules with diverse roles



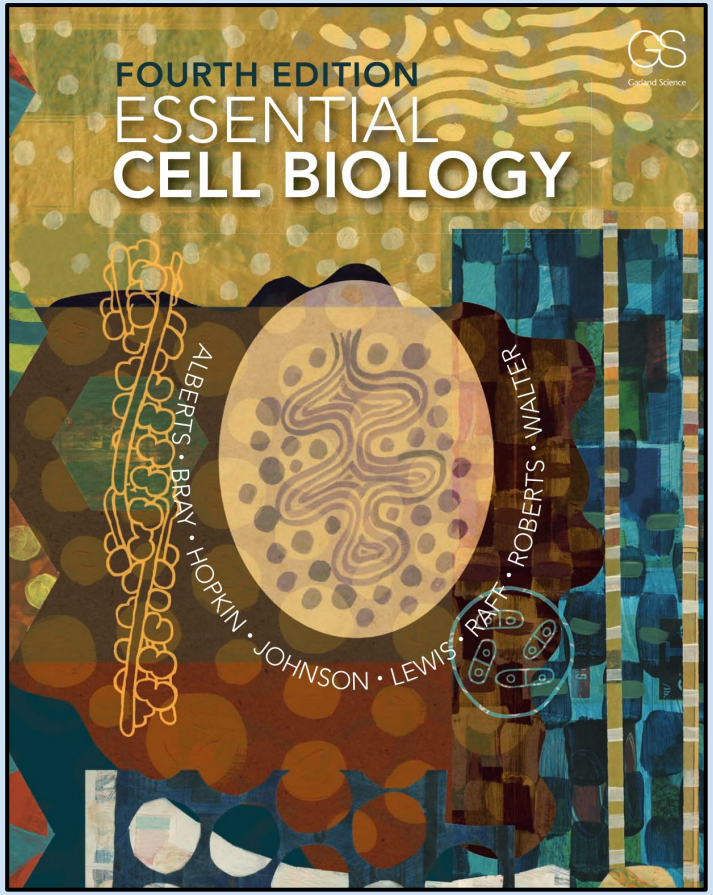


For more information

Alberts, B., Bray, D., Hopkin, K., Johnson, A. D., Lewis, J., Raff, M., Roberts, K., & Walter, P. (2013). *Essential cell biology* (4th ed.). Garland Publishing.

Free pdf online

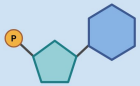
Chapters : 1, 4, 5, 7, 8



Omics Data

What are omics data?

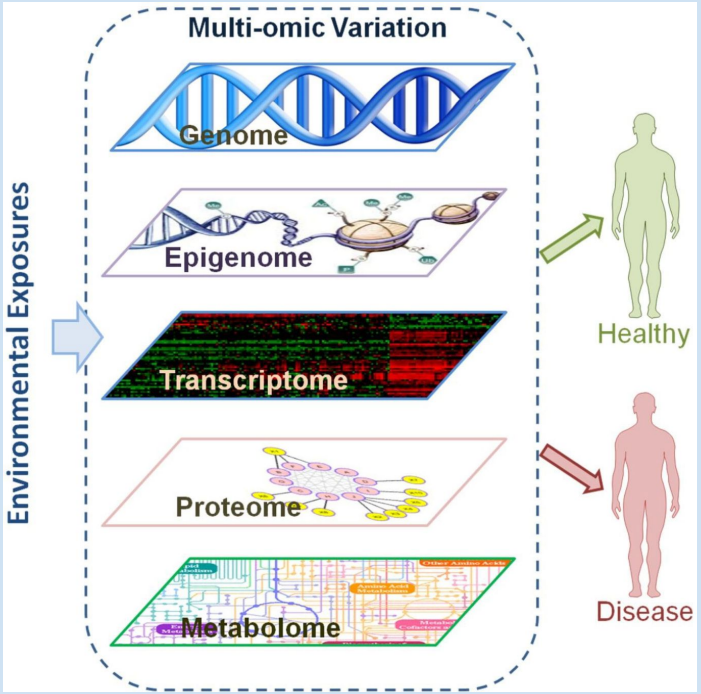
-omics ≈ all parts considered collectively

	Field	Molecules concerned	Short Description
	Genomics	DNA	Study of genetic mutations
	Epigenomics	DNA	Study of chemical modifications to the genome
	Transcriptomics	mRNA	Study of gene expression
	Proteomics	Protein	Study of proteins
	Metabolomics	Metabolites	Study of by-products of cellular metabolism

Why study omics data?

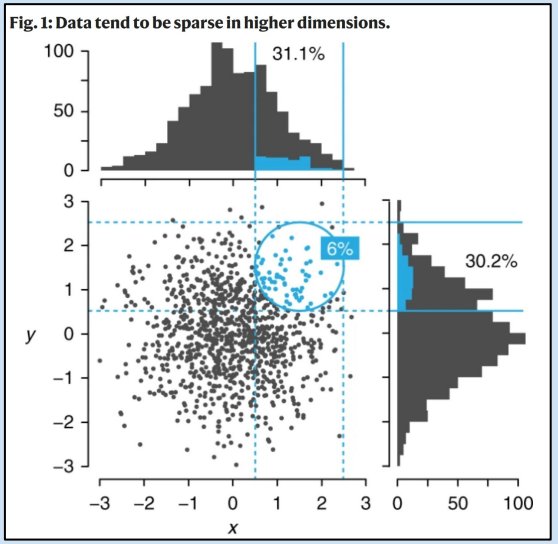
- **Understand underlying processes linking exposures and diseases**
  - Biomarkers
  - New routes for clinical development
  - Systems-level insight: hypothesis generation

- **Predict** disease states/treatment response
  - Personalised medicine



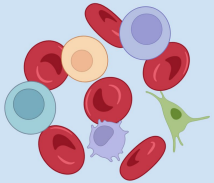
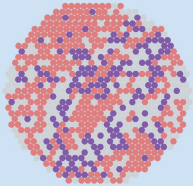
The curse(s) of dimensionality

1. **Data sparsity**: true signal “drowned” in dimension
2. **Multicollinearity**: redundant variables
3. **False discoveries**: large number of candidate variables + low amount of true positives = high rate of false discoveries
4. **Overfitting :** coincidental patterns in the data which do not generalise



Transcriptomic data

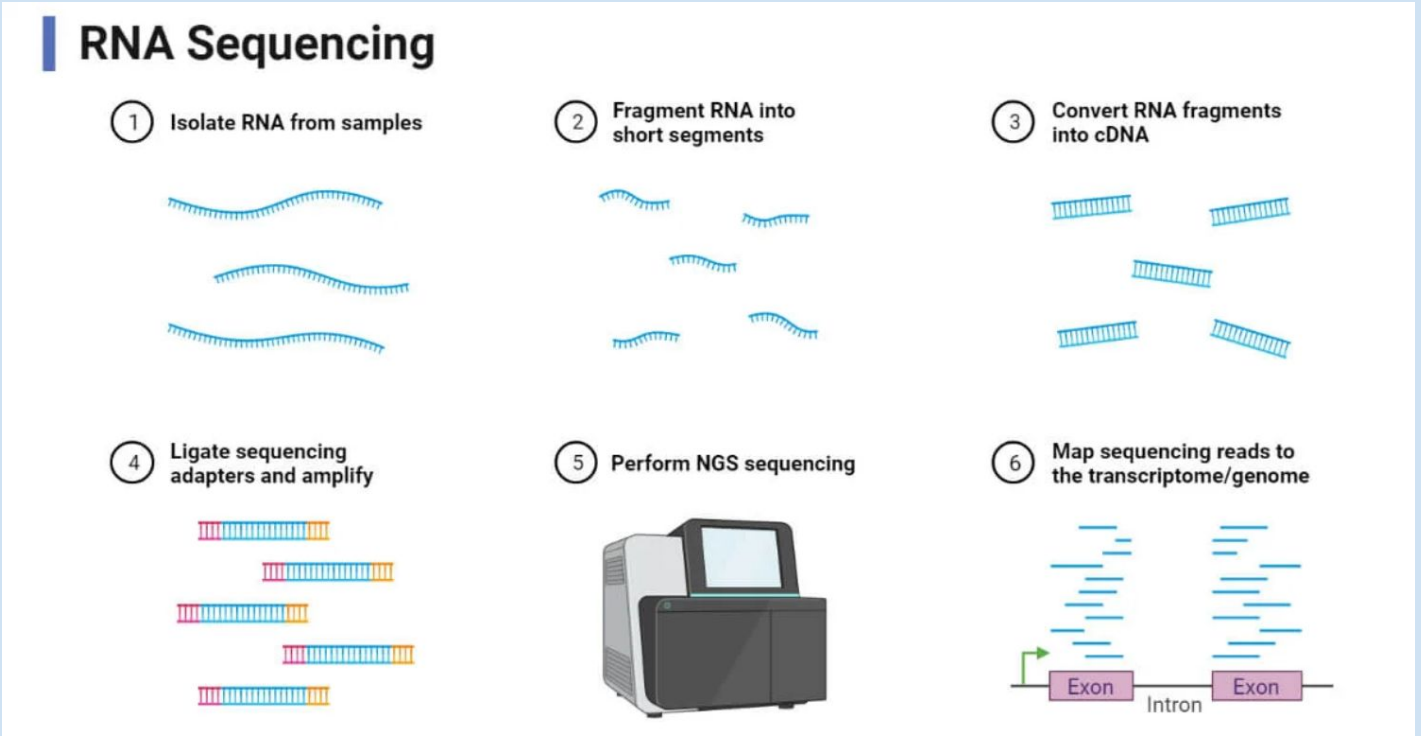
Types of transcriptomic data

	Bulk	Single-cell	Spatial
			
Description	Expression across mixed population of cell types	Expression of individual cells	Expression of individual cells and their location in a tissue sample
Strengths	Cheap, simple analysis, good for large effect sizes	High resolution, insight into cell heterogeneity and rare cell types	Preserves biological structure, spatial interactions
Limitations	Rare cell types expression drowned out	More expensive, complex analysis	Most expensive, complex analysis
Cost per sample	$200-$500	$1,000-$3,000	$5,000-$10,000

Technologies for bulk transcriptomic data

	Microarray	RNA-Sequencing
Description	Former standard, measures gene expression through fluorescent intensities on a pre-specified chip	Current standard, transcript sequences are read directly
Data type	Continuous data	Count data
Strengths	Cheaper, large quantity of public data	Detects all transcripts without need to specify genes beforehand, wider dynamic range
Limitations	Restricted to transcripts pre-specified on the chip, lower dynamic range	More expensive, count data requires tailored methods

RNA-seq data generation



What do transcriptomics data look like?

Genes (P)

Samples (N)

	ENSG00000000003	ENSG00000000005	ENSG00000000419	ENSG00000000457
Berry_London_Sample1	5	0	681	699
Berry_London_Sample2	14	0	596	809
Berry_London_Sample3	1	0	212	294
Berry_London_Sample4	6	0	619	665
Berry_London_Sample5	4	0	324	322
Berry_London_Sample6	6	0	297	251
Berry_London_Sample7	8	0	719	787
Berry_London_Sample8	18	0	855	872

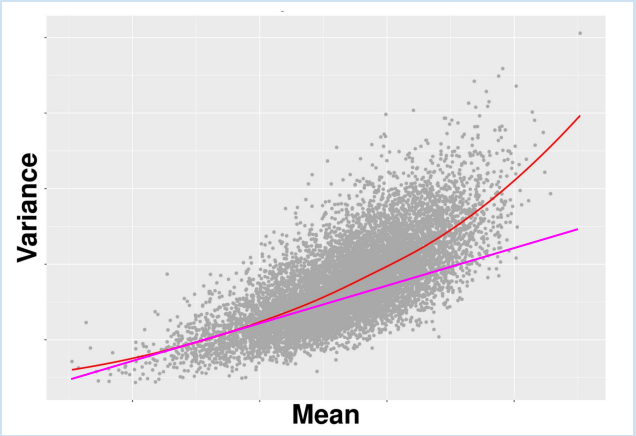
Transcript counts

(reads)

Sequencing depth / library size : total number of reads in a sample

Statistical problems in RNA-seq data

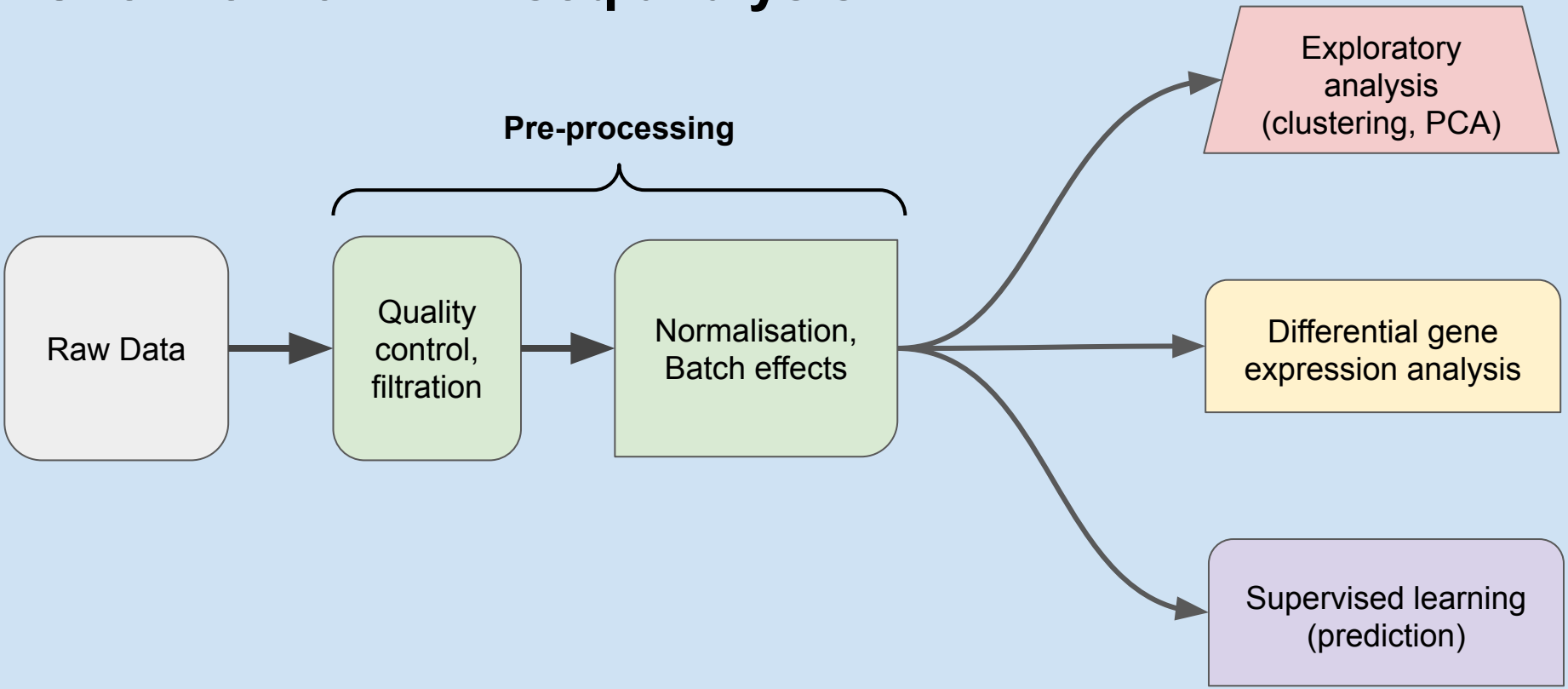
- **Curses of dimensionality**
  - Sparsity, multicollinearity, inflated false positives, overfitting
- **Count data requires special methods**
  - Discrete, non-negative
- **Heteroskedastic & overdispersed**
  - Mean depends on variance
- **Large dynamic range**
  - Typical range : 0 - 2,000,000



Mean vs variance across genes

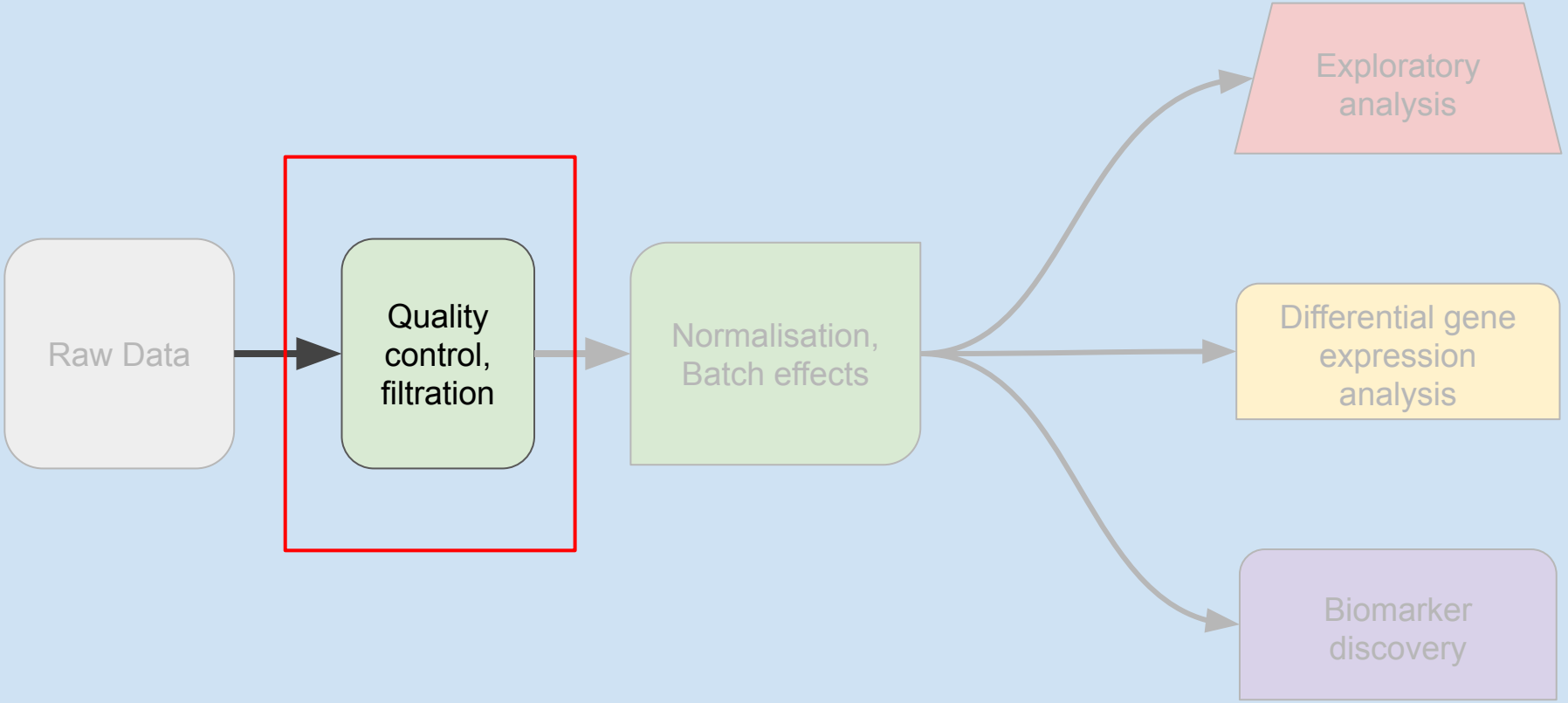
(simulated data)

Overview of RNA-seq analysis



Pre-processing of RNA-seq Data

Quality Control & Filtration

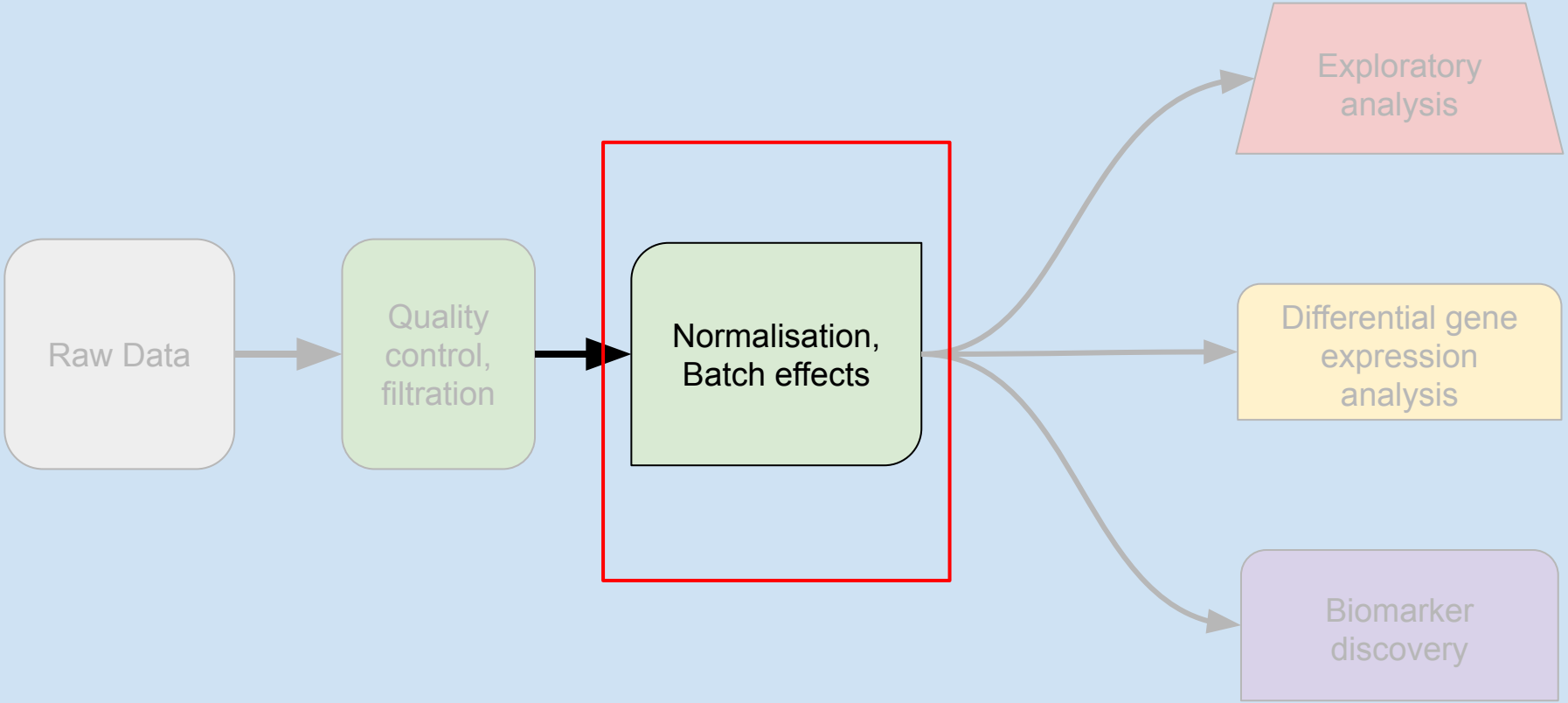

Raw Data
Quality control, filtration
Normalisation, Batch effects
Exploratory analysis
Differential gene expression analysis
Biomarker discovery

Quality control

- **RIN (RNA integrity number)**
  - Value 1-10: quality of sequenced RNA
  - Samples with low RIN removed from study
  - RIN >=8 common requirement
- **Sequencing depth / library size**
  - Number of reads in a sample
  - Higher depth = more precise detection of weakly expressed genes
  - Samples with lower than 1,000,000 reads could be removed
- **Gene filtration**
  - Non-protein coding genes
  - Very weakly expressed genes (less precise)
  - Constantly expressed genes (e.g. never expressed)

Pre-processing of RNA-seq Data

Normalisation


Raw Data
Quality control, filtration
Normalisation, Batch effects
Exploratory analysis
Differential gene expression analysis
Biomarker discovery

Library size normalisation

- Sequencing yields counts which mix two sources of information
  - **True biological abundance of each gene**
  - Technical effects
- Normalisation seeks to remove technical effects

Library size normalisation

- Sequencing yields counts which mix two sources of information
  - **True biological abundance of each gene**
  - Technical effects
- Normalisation seeks to remove technical effects

Sample	Exposure	Gene 1	Gene 2	Gene 3	Library Size
A	Yes	300	300	400	1,000

Library size normalisation

- Sequencing yields counts which mix two sources of information
  - **True biological abundance of each gene**
  - Technical effects
- Normalisation seeks to remove technical effects

Sample	Exposure	Gene 1	Gene 2	Gene 3	Library Size
A	Yes	300	300	400	1,000
B	No	600	600	800	2,000

Library size normalisation

- Sequencing yields counts which mix two sources of information
  - **True biological abundance of each gene**
  - Technical effects
- Normalisation seeks to remove technical effects

Sample	**Exposure**	**Gene 1**	**Gene 2**	**Gene 3**	**Library Size**
A	Yes	300	300	400	1,000
B	No	600	600	800	2,000
True effect?		No	No	No	

Library size normalisation

- Sequencing yields counts which mix two sources of information
  - **True biological abundance of each gene**
  - Technical effects
- Normalisation seeks to remove technical effects

Sample	Exposure	Gene 1	Gene 2	Gene 3	Library Size
A	Yes	300	300	400	1,000
B	No	600	600	800	2,000
True effect?		No	No	No	

No gene is differentially expressed, but more reads were sampled in B

A simple solution: normalise by library size

Idea : Multiply all values by a ***normalisation factor*** to render them comparable

- use ***normalisation factors*** equal to the library size
- Entries are now relative *abundance* of each gene

Sample	Exposure	Gene 1	Gene 2	Gene 3	Library Size
A	Yes	0.3	0.3	0.4	1
B	No	0.3	0.3	0.4	1
True effect?		No	No	No	

Counts as proportions of total reads sampled

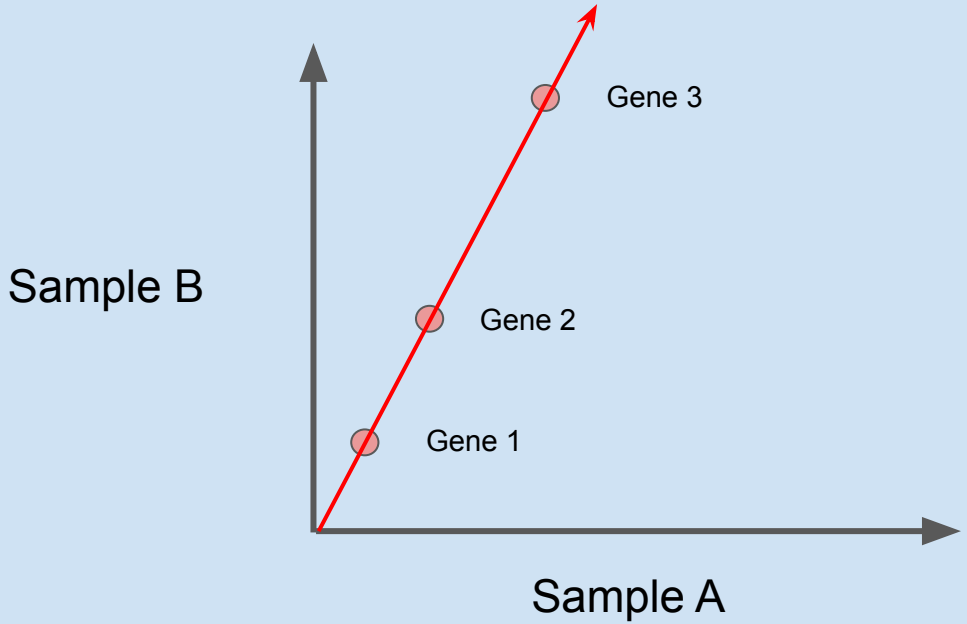
Counts-per-million

- Counts-per-million (CPM) multiplies these proportions by 1,000,000
  - Often this is accompanied by a log2 transformation (log2-CPM)

Sample	Exposure	Gene 1	Gene 2	Gene 3	Library Size
A	Yes	300,000	300,000	400,000	1,000,000
B	No	300,000	300,000	400,000	1,000,000
True effect?		No	No	No	

Counts in CPM

Graphical representation of CPM



Slope of red line =
normalisation factor

A problem with CPM

- **Truly differentially abundant genes** change the **relative abundance** of genes with no true effect

Sample	Exposure	Gene 1	Gene 2	Gene 3	Library Size
A	Yes	300	300	1400	2,000

A problem with CPM

- **Truly differentially abundant genes** change the **relative abundance** of genes with no true effect

Sample	Exposure	Gene 1	Gene 2	Gene 3	Library Size
A	Yes	300	300	1400	2,000
B	No	300	300	400	1,000

A problem with CPM

- **Truly differentially abundant genes** change the **relative abundance** of genes with no true effect

Sample	Exposure	Gene 1	Gene 2	Gene 3	Library Size
A	Yes	300	300	1400	2,000
B	No	300	300	400	1,000
True effect?		No	No	Yes	

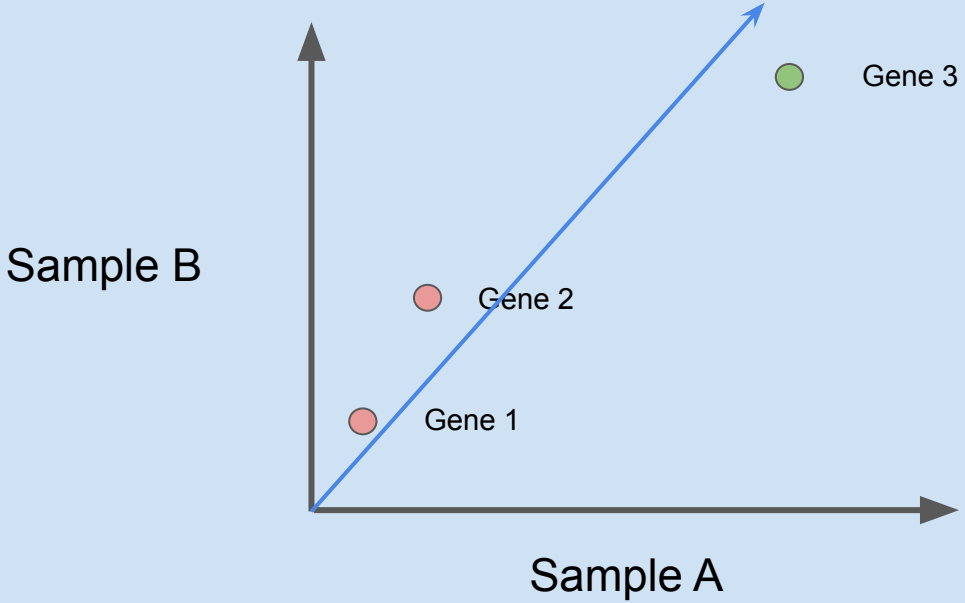
A problem with CPM

Sample	Exposure	Gene 1	Gene 2	Gene 3	Library Size
A	Yes	150,000	150,000	700,000	1,000,000
B	No	300,000	300,000	400,000	1,000,000
True effect?		No	No	Yes	

Counts in CPM

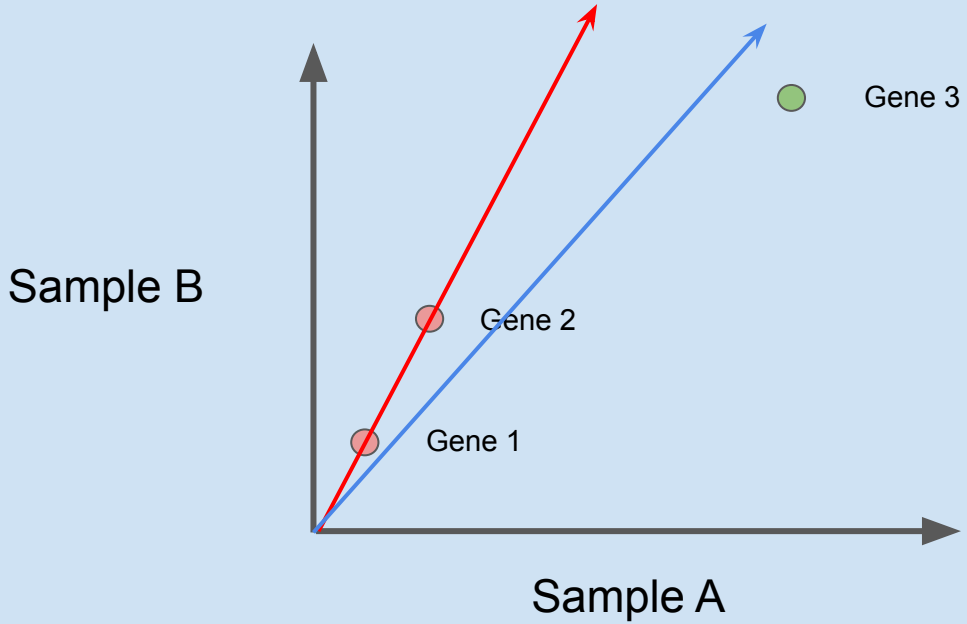
Genes 1 and 2 have no true effect, but appear artificially lower in expression due to the influence of gene 3 in sample A

Graphical representation of the problem with CPM



Slope of blue line = normalisation factor (biased by over-expression of gene 3 in sample A)

Graphical representation of the problem with CPM



Ideally, we would like to estimate the red line (normalisation factors while excluding gene 3)

Trimmed-mean-of-M-Values normalisation

TMM estimates normalisation factors while ***trimming*** (i.e. excluding) extreme values

- **Assumes that most genes are not differentially abundant**
- It calculates normalisation factors while excluding extremely high or low values
  - These normalisation factors are called *M-values, and represent the ratio of relative abundances between exposure groups*

$$M_g = \frac{Count_{g,A}}{library_A} \div \frac{Count_{g,B}}{library_B}$$

Worked example: TMM

Sample	Exposure	Gene 1	Gene 2	Gene 3	Library Size
A	Yes	300	300	1400	2,000
B	No	300	300	400	1,000
M-value					
True effect?		No	No	Yes	

Worked example: TMM

1) Divide counts by library size

Sample	Exposure	Gene 1	Gene 2	Gene 3	Library Size
A	Yes	0.15	0.15	0.7	1
B	No	0.3	0.3	0.4	1
M-value					
True effect?		No	No	Yes	

Worked example: TMM

2) Calculate M-values: mean expression in exposed / mean expression unexposed

Sample	Exposure	Gene 1	Gene 2	Gene 3	Library Size
A	Yes	0.15	0.15	0.7	1
B	No	0.3	0.3	0.4	1
M-value		0.5	0.5	1.75	
True effect?		No	No	Yes	

Worked example: TMM

3) *Trim* M-values: exclude extreme M-values (1.75)

Sample	Exposure	Gene 1	Gene 2	Gene 3	Library Size
A	Yes	0.15	0.15	0.7	1
B	No	0.3	0.3	0.4	1
M-value		0.5	0.5	1.75	
True effect?		No	No	Yes	

Worked example: TMM

4) Calculate the *trimmed mean of M-values* (0.5 x 0.5 /2 = 0.5)

Sample	**Exposure**	**Gene 1**	**Gene 2**	**Gene 3**	**Library Size**
A	Yes	0.15	0.15	0.7	1
B	No	0.3	0.3	0.4	1
M-value		0.5	0.5	1.75	
True effect?		No	No	Yes	

Worked example: TMM

5) Multiply raw counts by TMM factor (0.5)

Sample	Exposure	Gene 1	Gene 2	Gene 3
A	Yes	150	150	700
B	No	150	150	200
True effect?		No	No	Yes

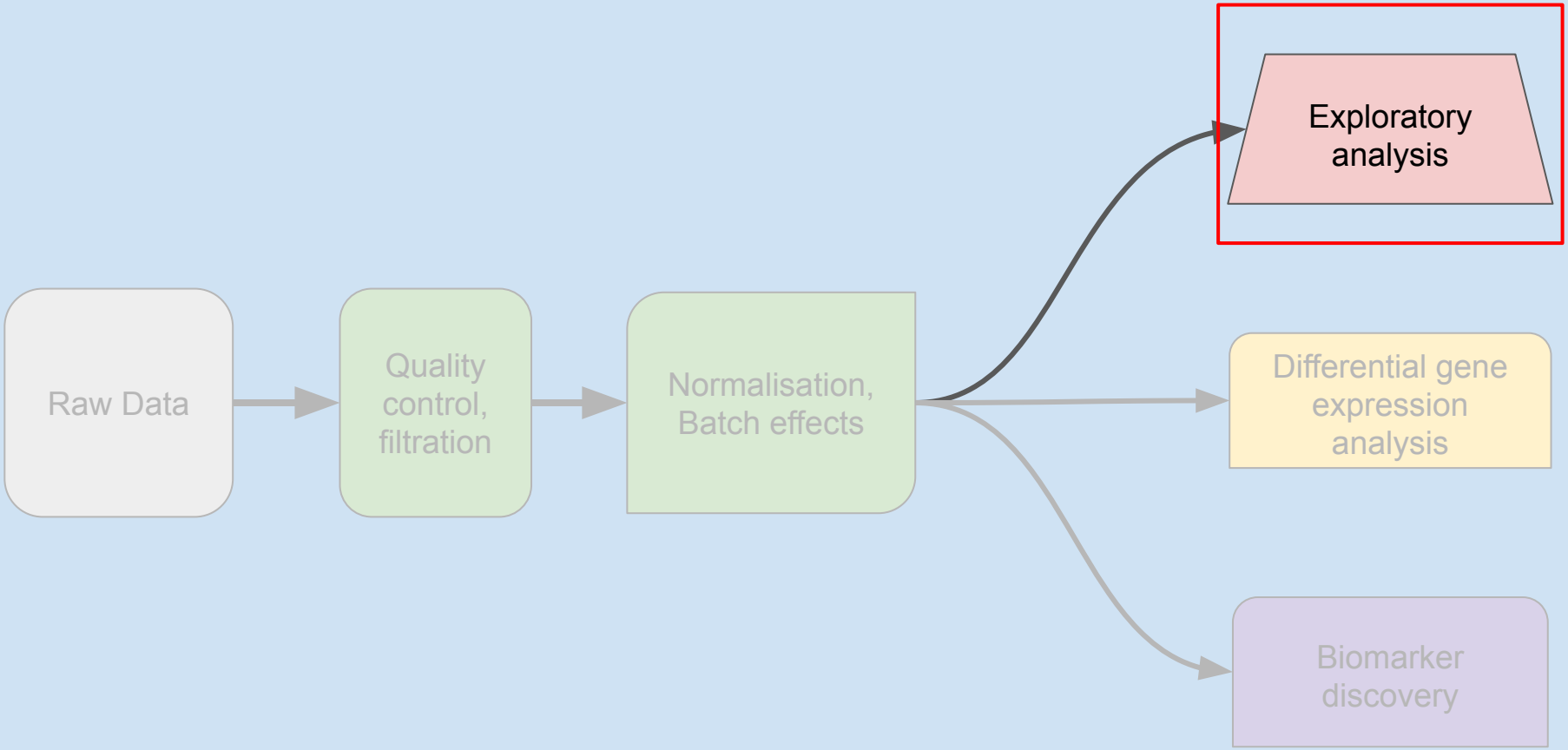
Post-normalisation differences between samples A and B reflect the true effects of the exposure

Summary on normalisation

- Count data combines biological effects with technical effects
- Normalisation is a first step to isolate the biological effects
- **CPM** normalises based on **library size**
    - This can be biased by the presence of differentially abundant genes
- **TMM-normalisation excludes genes with extreme** relative abundances

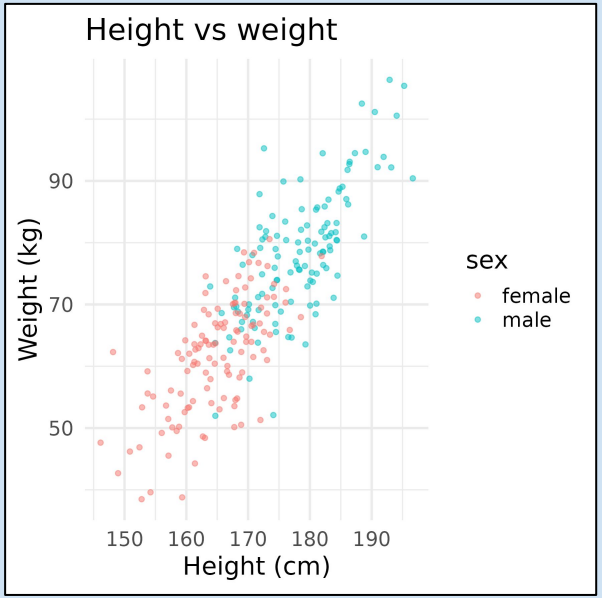
Robinson MD, Oshlack A. A scaling normalization method for differential expression analysis of RNA-seq data. Genome Biol. 2010;11(3):R25. doi: 10.1186/gb-2010-11-3-r25.

Exploratory Analysis :
Principal Component Analysis


Raw Data
Quality control, filtration
Normalisation, Batch effects
Exploratory analysis
Differential gene expression analysis
Biomarker discovery

Variability is information

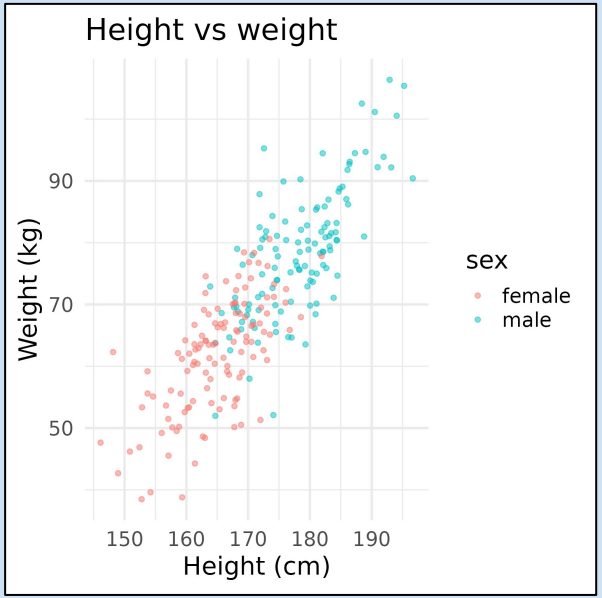
- **Statistics is all about exploring this variability!**



A lot of the **variation in height and weight** is ***explained*** **by sex**

Variability is information

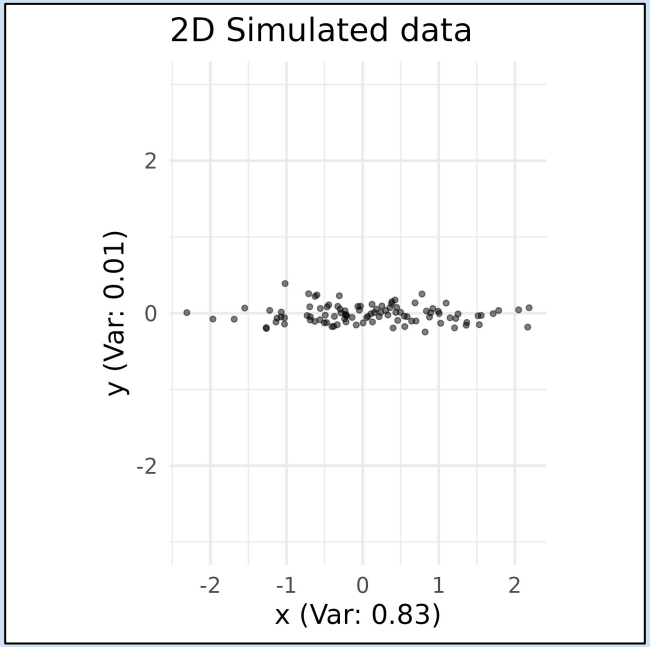
- **Statistics is all about exploring this variability!**



Visual inspection is limited to <3 dimensions

- We cannot explore 1000s of variables in this way
- **PCA** gives an **efficient** way to **represent** the variability in the data in only a **few dimensions**!

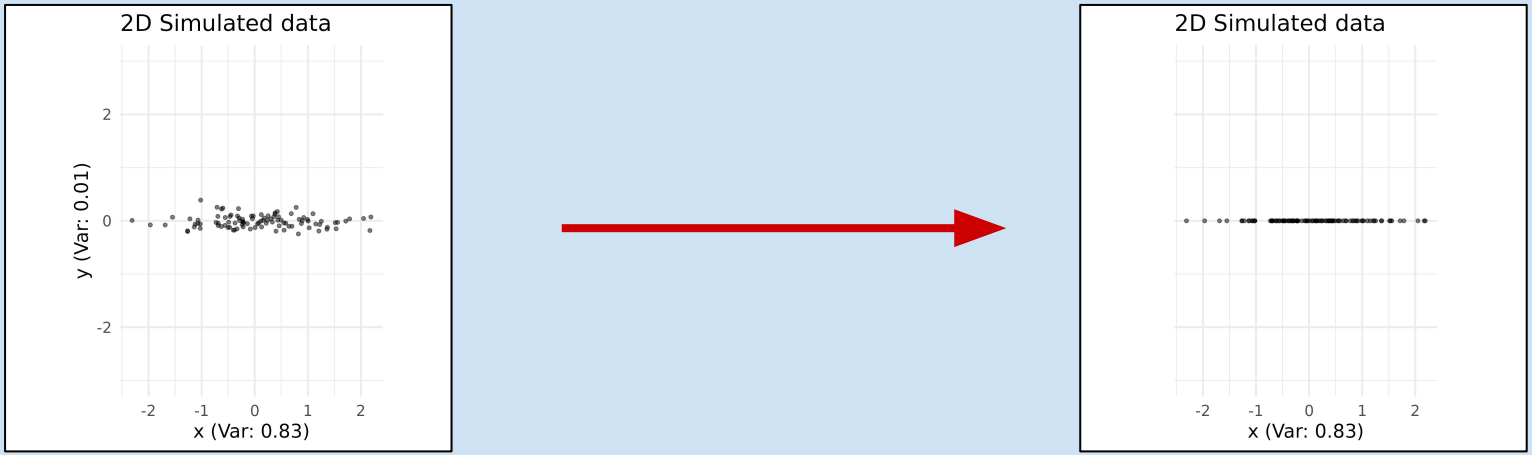
An illustrative example (1)



Could we represent this data in a lower dimension without losing much information?

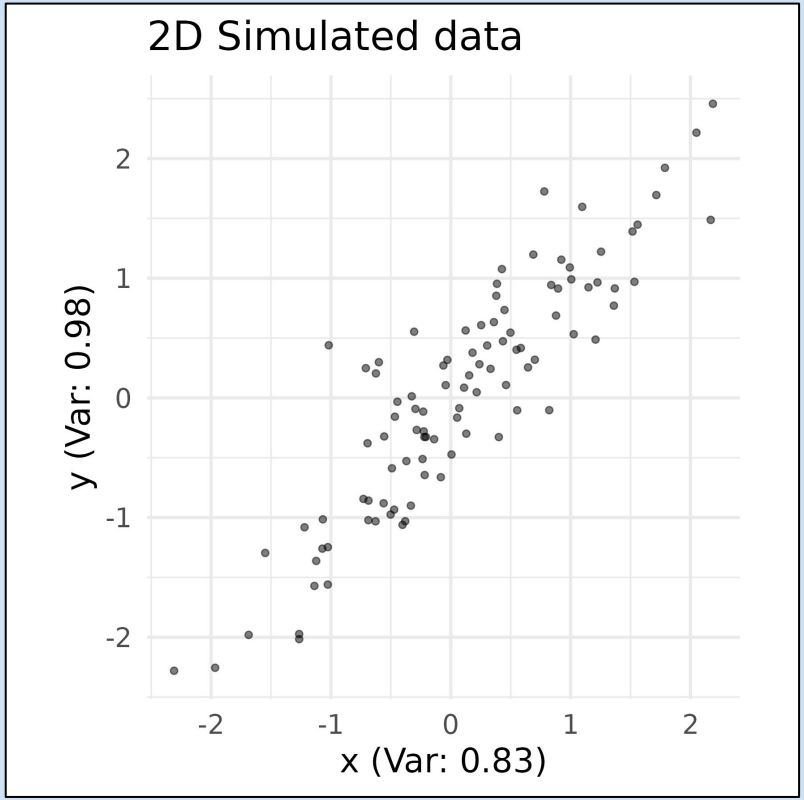
An illustrative example (2)

- **Variability is information!**
- **Most of the information** is contained in the **x-coordinates**
  - The proportion of variability explained by x is 0.83/(0.83+0.01) = 99%
- Could we discard the y coordinates and represent the data in 1D?



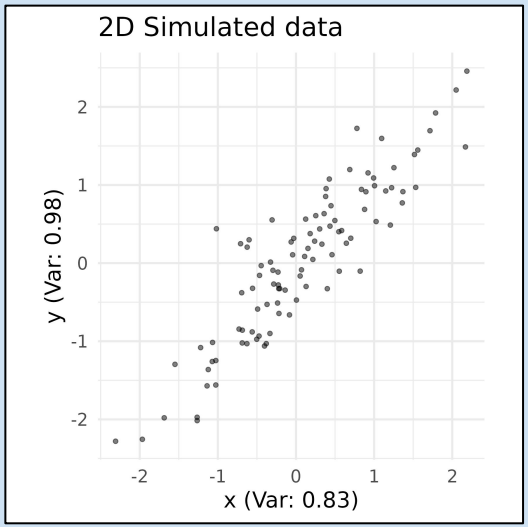
Another example

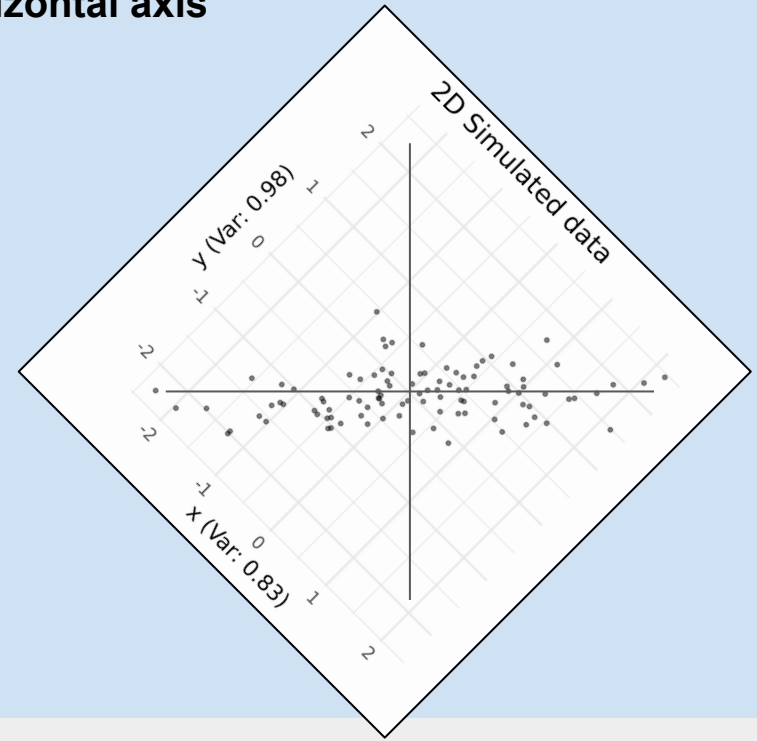
- This time, data varies across both x and y
- However, x and y are correlated - they contribute similar information
- **Could we represent this 2D data in 1D** without losing much information?



Intuition of PCA (1)

- If we rotate our axes, we have a similar situation to the first example…
- **Most of the variability is now captured by the horizontal axis**
- This is what PCA does!



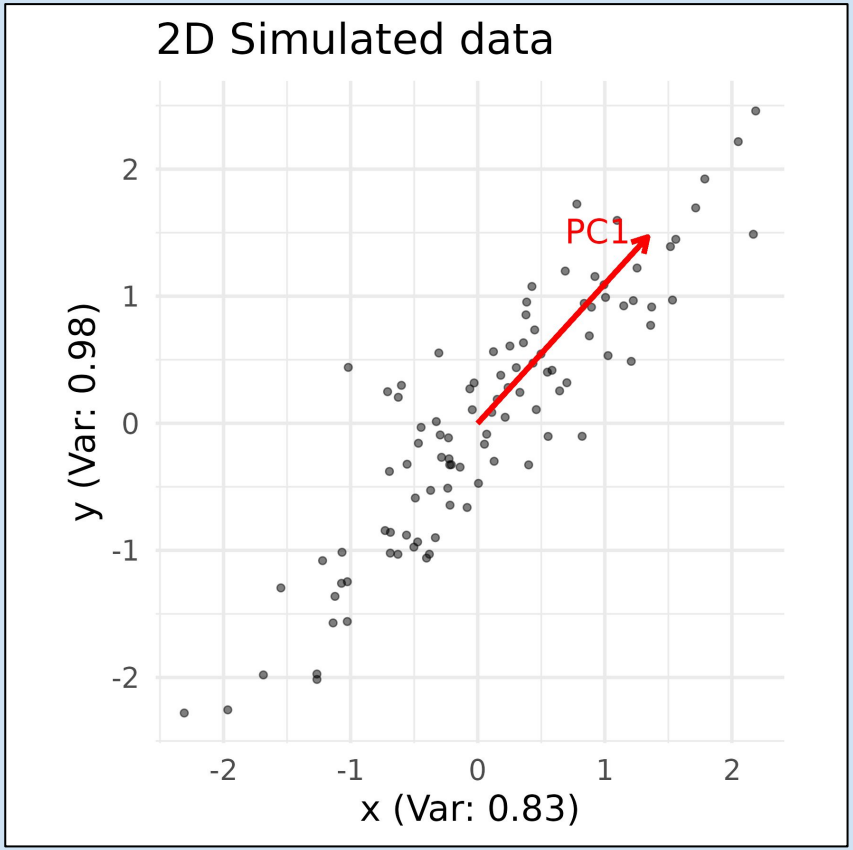


Intuition of PCA (2)

PCA finds the directions of greatest variability

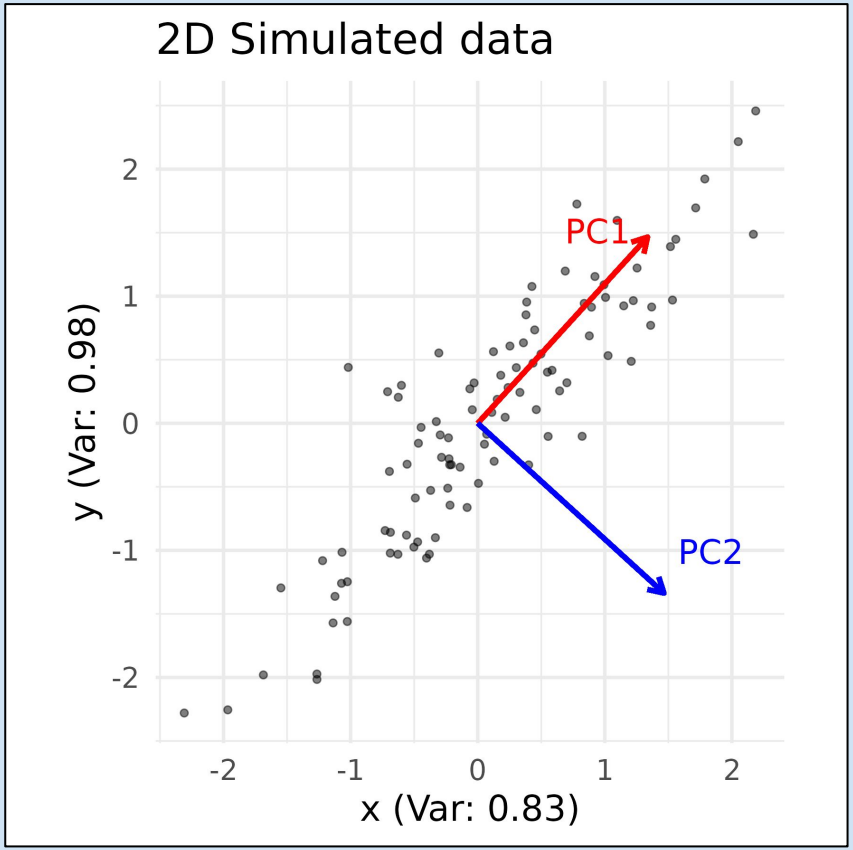
Principal components = these directions (vectors)

The first principal component (PC1) = direction of most variability



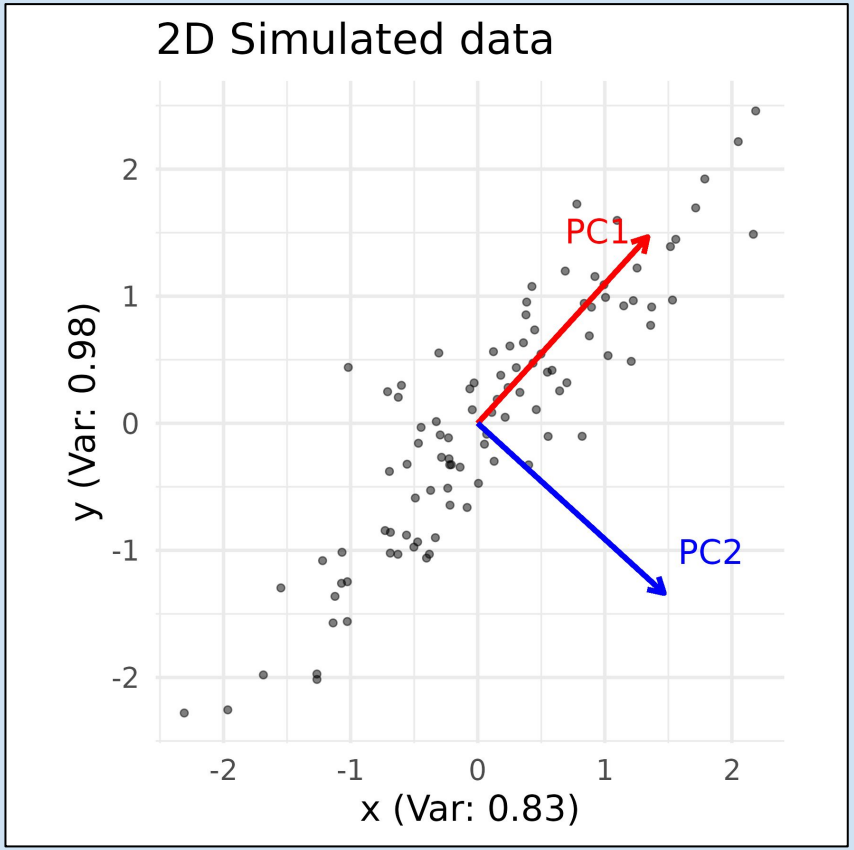
Intuition of PCA (3)

- The second principal component (PC2) is **orthogonal to PC1** and is the direction of the **second most variability**
- These 2 redefined axes are equivalent to “rotating the data”



How were the PCs formed?

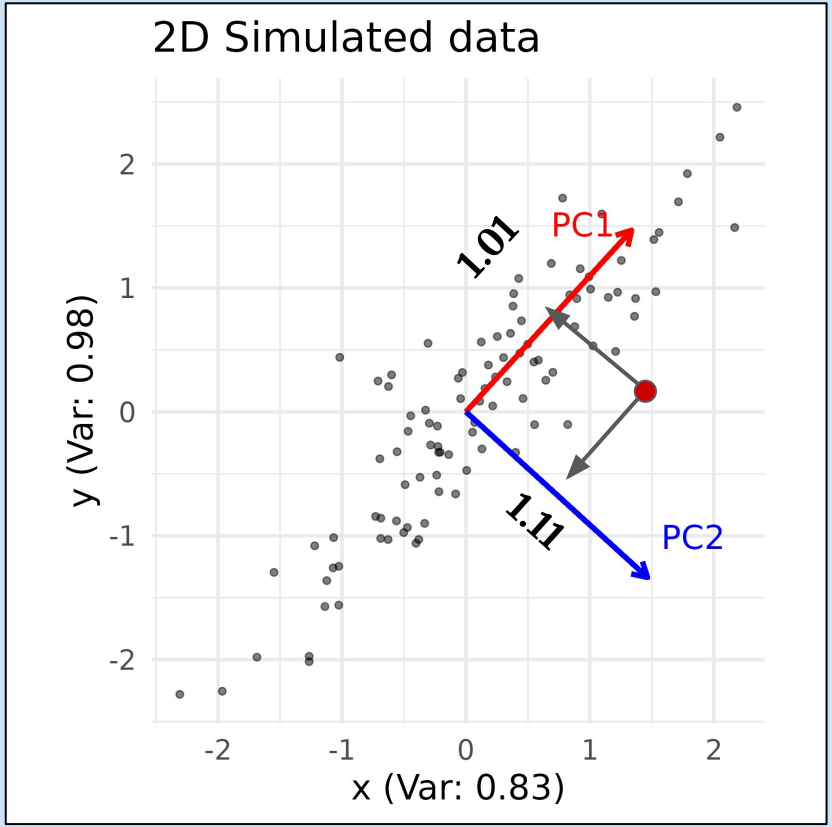
- **Linear combinations** of the x and y coordinates
- In this example :
  - PC1 = 0.67x + 0.74y
  - PC2 = 0.74x - 0.67y



How to *project* the data onto the PCs?

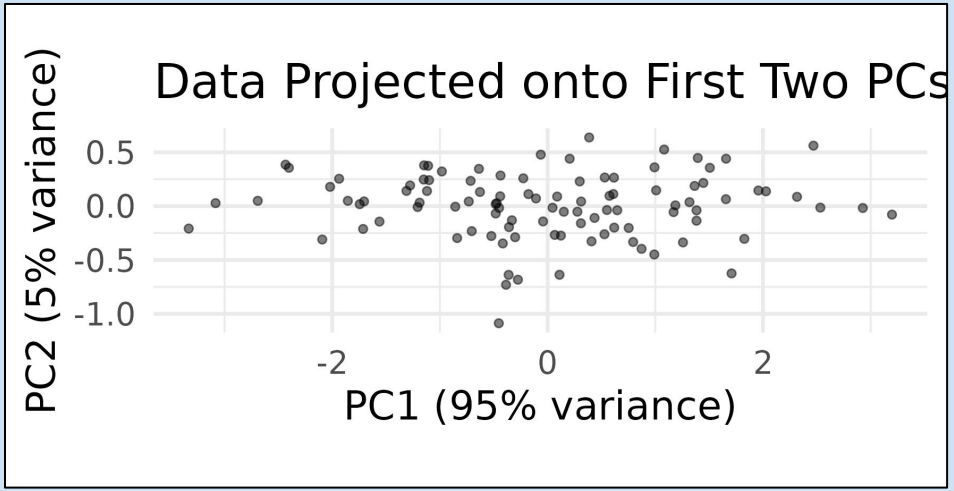
- PC1 = 0.67x + 0.74y
- PC2 = 0.74x - 0.67y
- If **(x,y) = (1.5,0)**
- Its coordinates on the new axes are

  (PC1, PC2) = (1.01, 1.11)



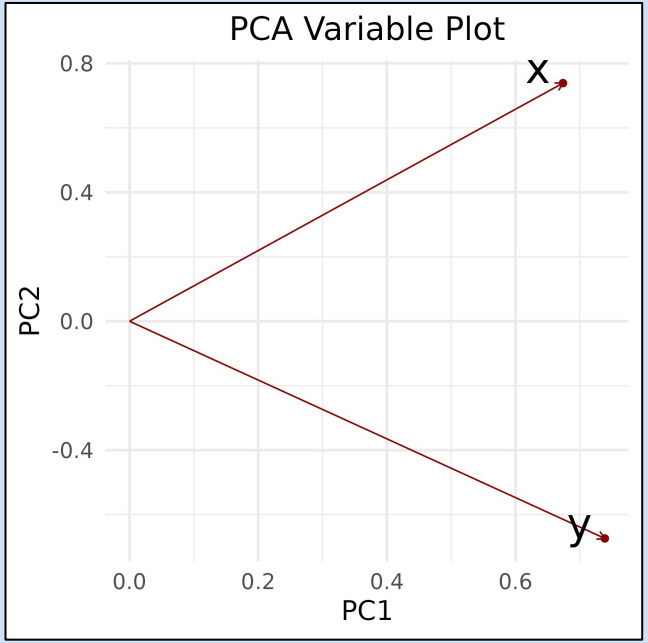
Samples projected onto PCs

- We can represent our data on our new “principal axes”
  - i.e. rotating our original data to new axes
  - **Important :** the data stays 2-dimensional
- **Almost all of the variability** in the data is explained by the PC1 values



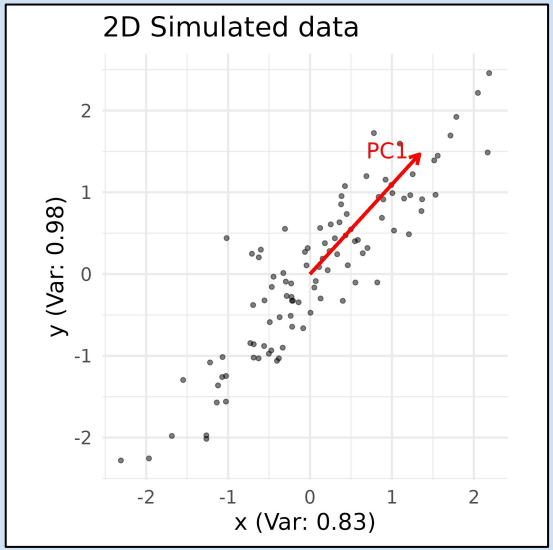
Variables contribution to PCs

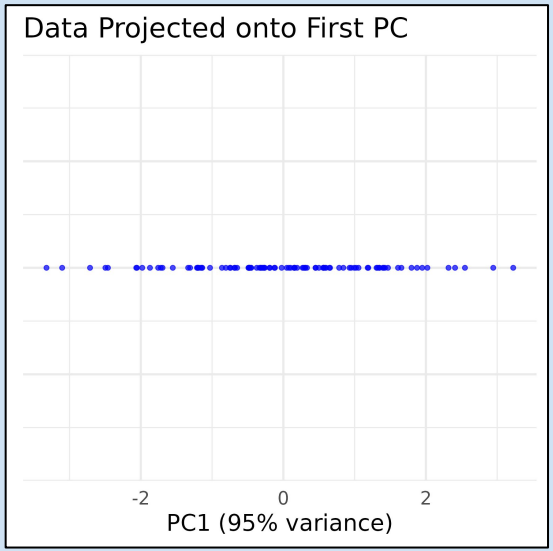
- Common to explore how variables contribute to the PCs
- This shows us the **contributions** (weights) of each variable to each PC, e.g. here
  - PC1 = 0.67x + 0.74y
  - PC2 = 0.74x - 0.67y
- **Length : strength of contribution** to PC
- **Direction :** how the variable relates to each PC
  - Variables in same direction are positively correlated
  - Variables in opposite direction negatively correlated
  - Orthogonal variables uncorrelated



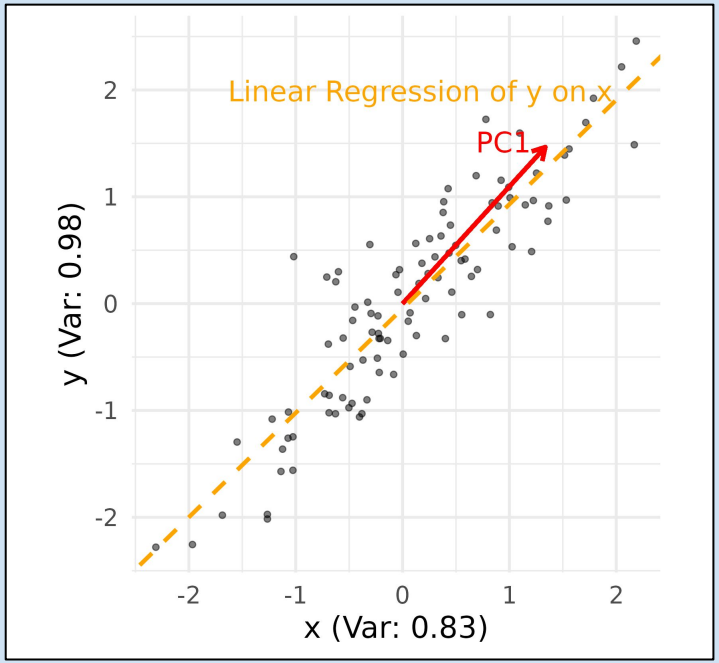
Can we reduce the dimension?

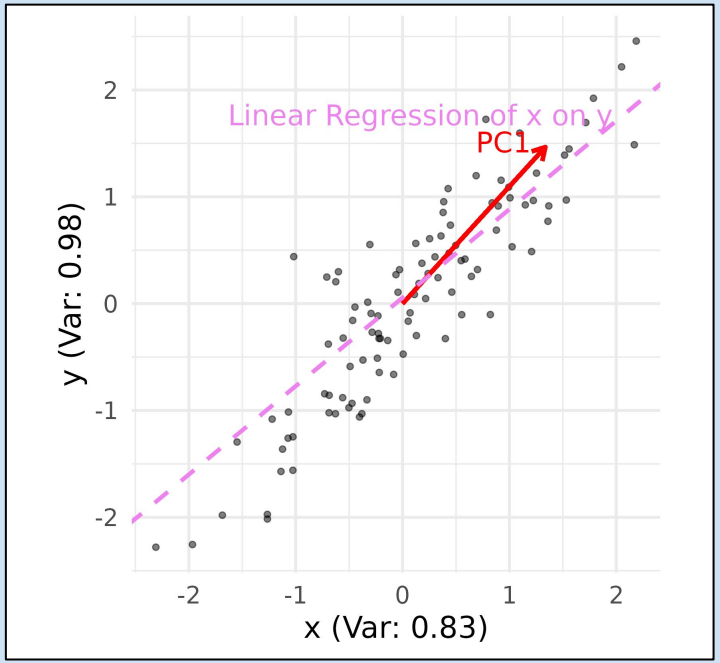
- We can discard PC2 and only represent the data points on PC1, keeping 95% of the variability in the data explained
- We have now ***reduced the dimension*** of our data from 2 to 1





Note : PCs are NOT linear regression!





What happens in higher dimensions?

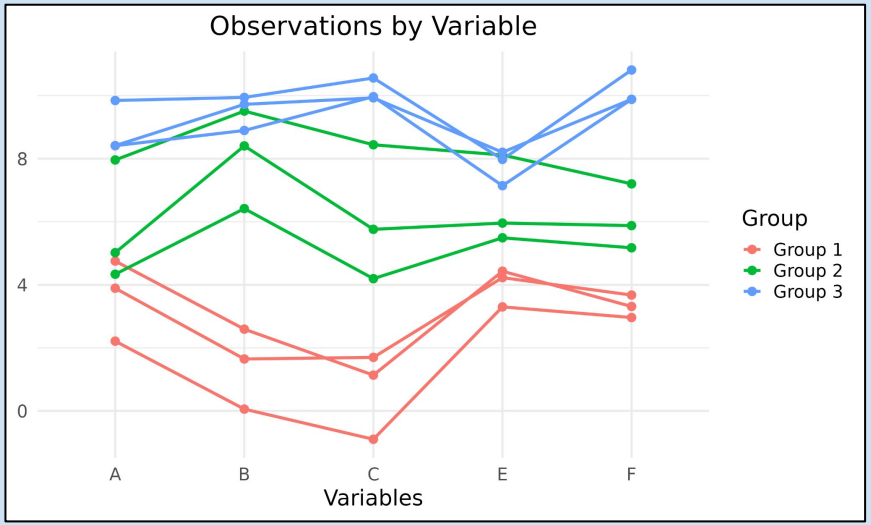
If we start with p-variables, PCA will compute p-components as follows

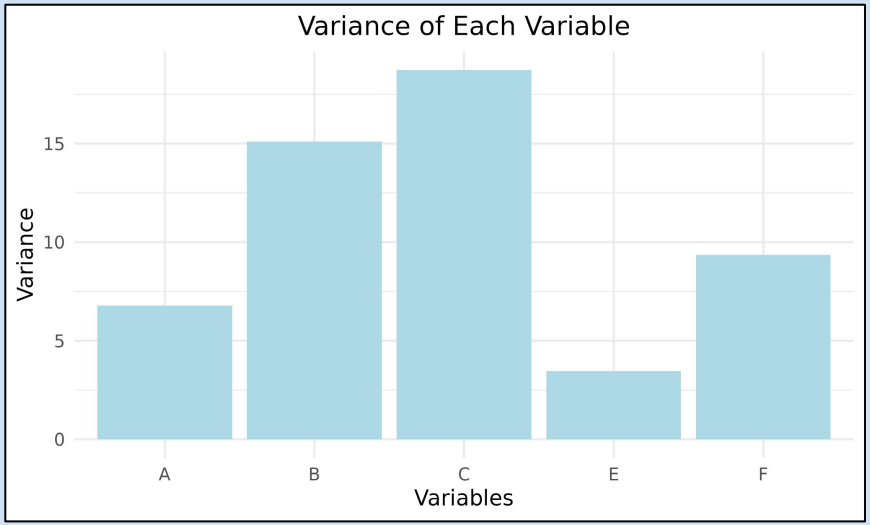
1. PC1 : direction of most variability
2. PC2 : **orthogonal** (i.e. at a right angle) to PC1, direction of most variability
3. PC3 : **orthogonal** to PC1 AND PC2, direction of most variability
4. …
5. PC(p) : **orthogonal** to PC1,...,PC(p-1), direction of most variability

So, **PCA does not directly reduce the dimension**, it just concentrates the variability in the first few variables (PCs)

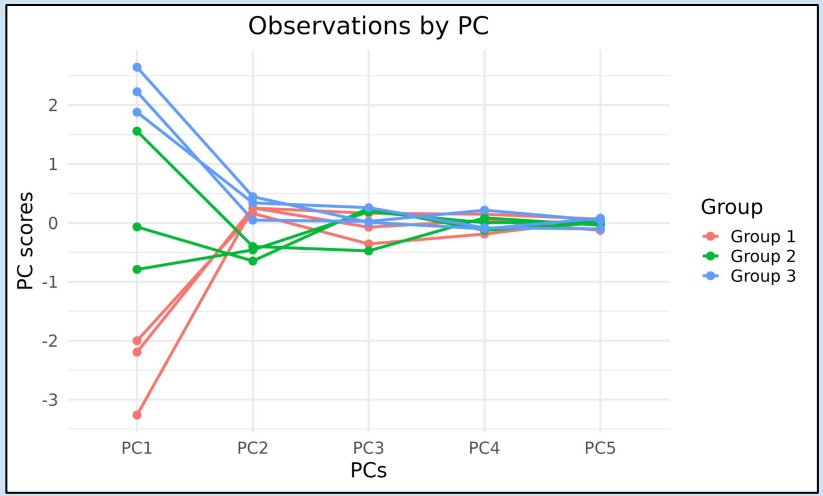
An example in higher-dimension

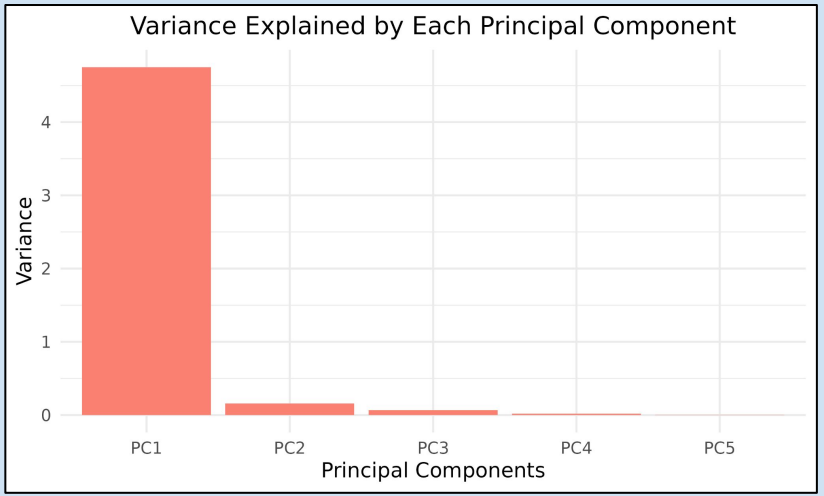
- 9 observations of 5 variables
- 3 categorical exposure groups





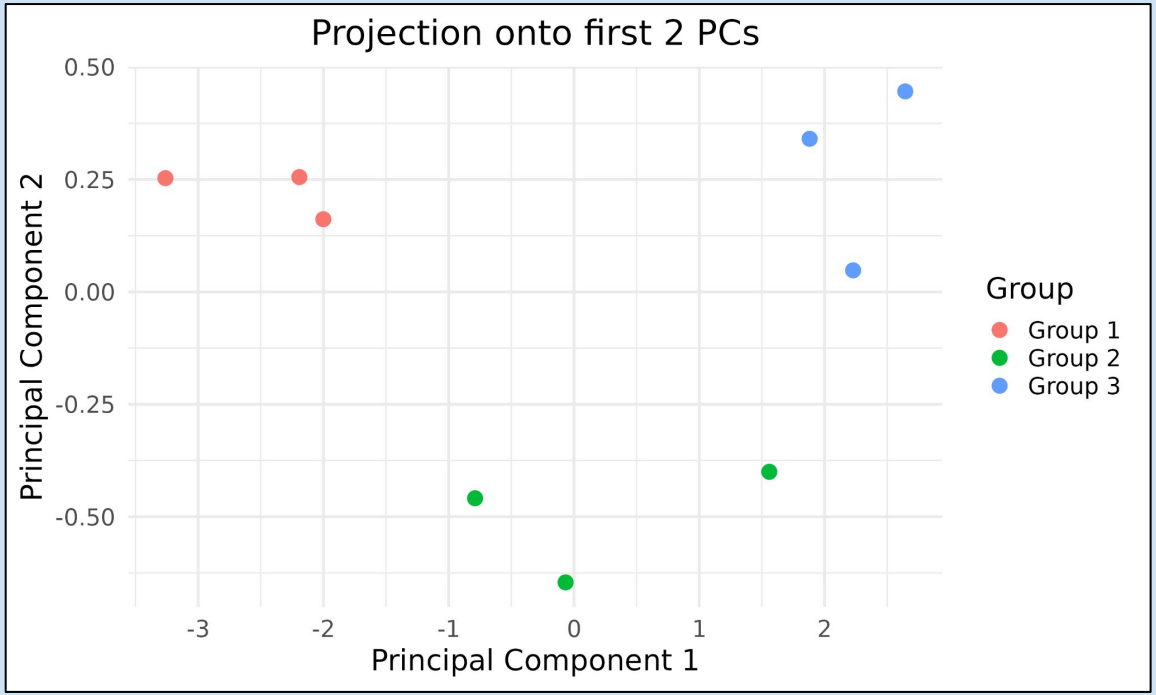
Observations after PCA





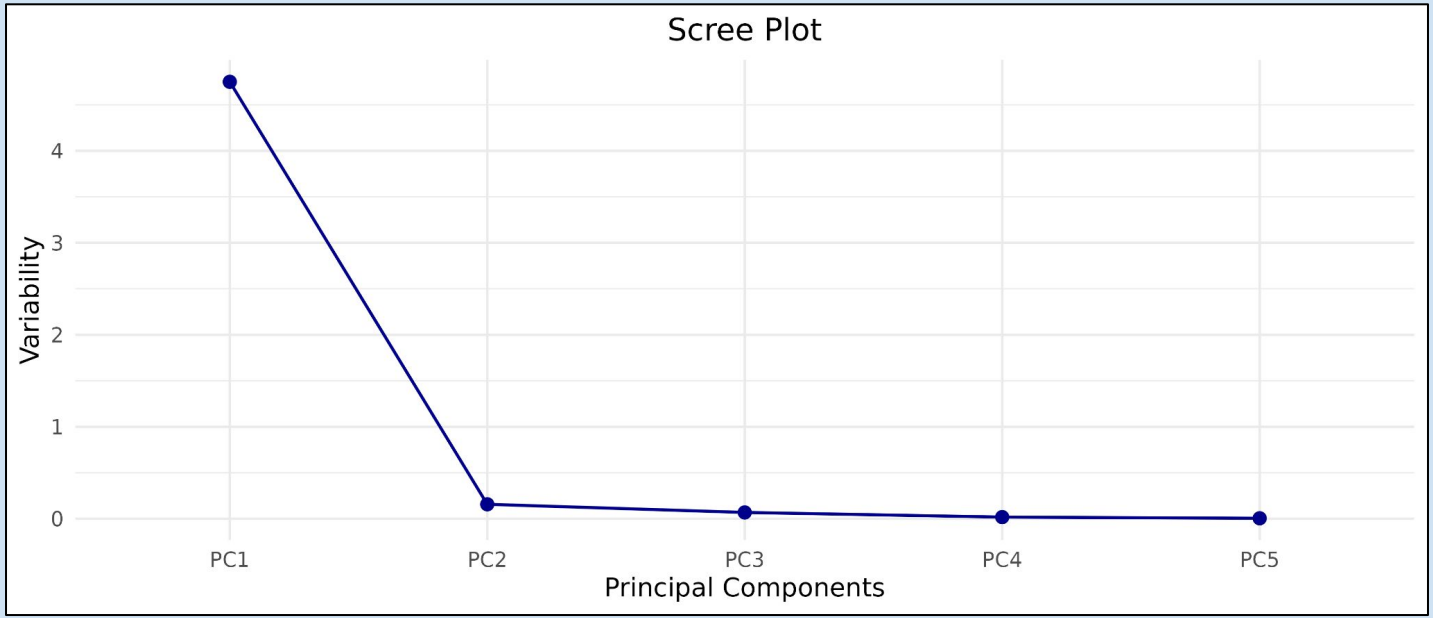
Projection of observations on the first 2 PCs

- PCA can help identify clusters
- The groups ***cluster together*** when plotted on the first 2 PCs



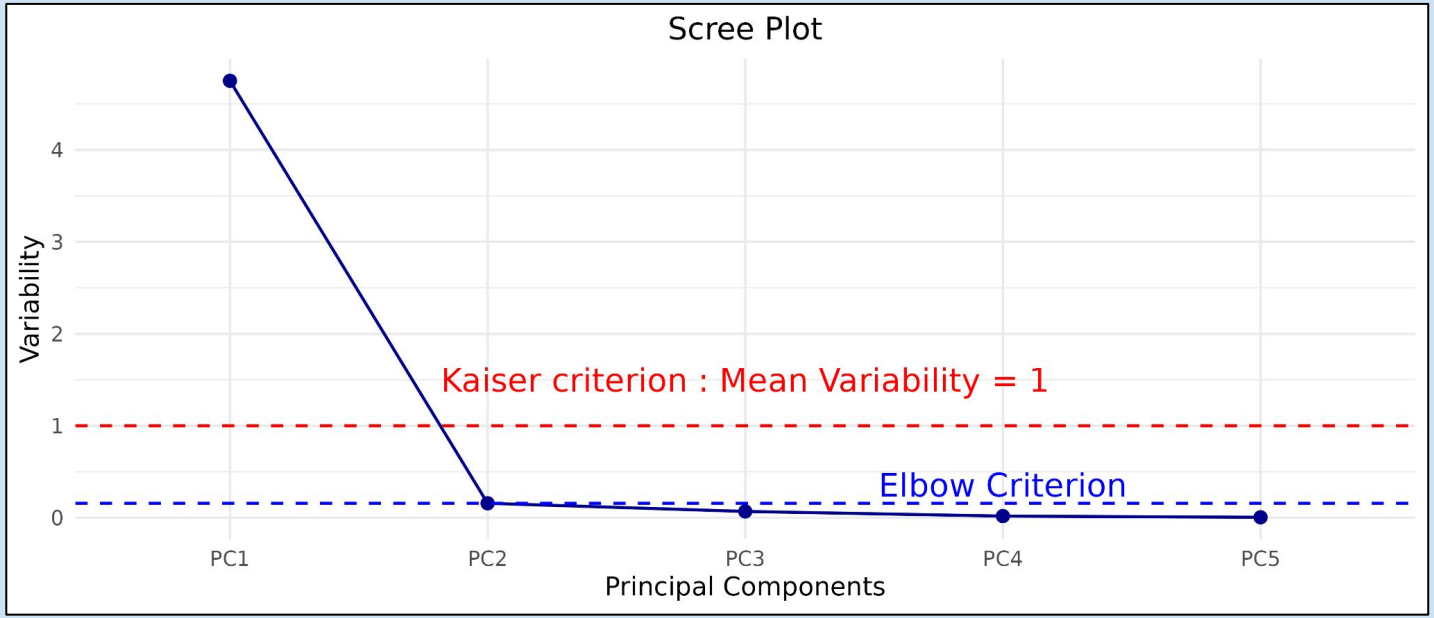
How to choose how many PCs to keep?

- Plot variability of ordered PCs



How do we choose how many PCs to keep?

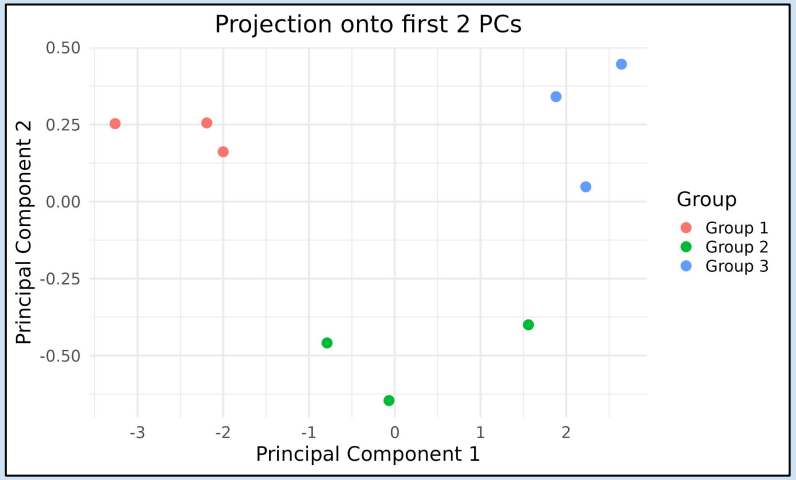
- **Kaiser criterion :** Keep PCs accounting for more variability than average
- **Elbow/ Catell criterion :** Find a visual “elbow” in the scree plot



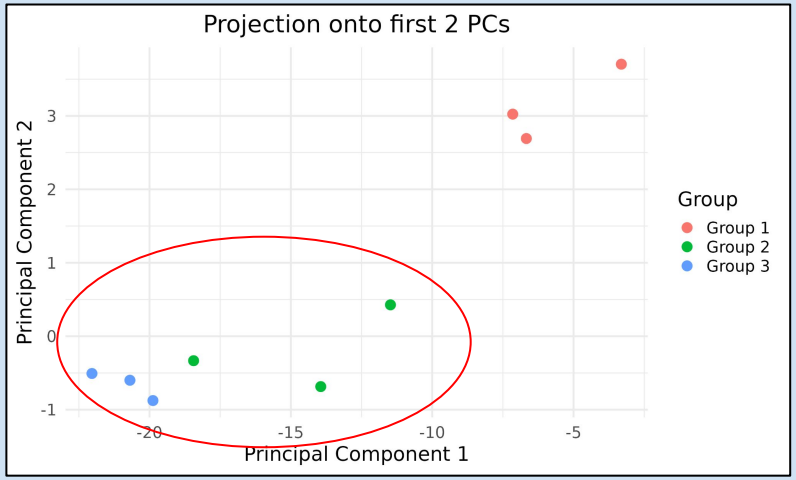
PCA depends on the scale of the data

- **Scaled data :** all variables have variance 1
- **Centered data :** all variables have mean 0
- **Always center and scale your data before PCA**

Centered and scaled data

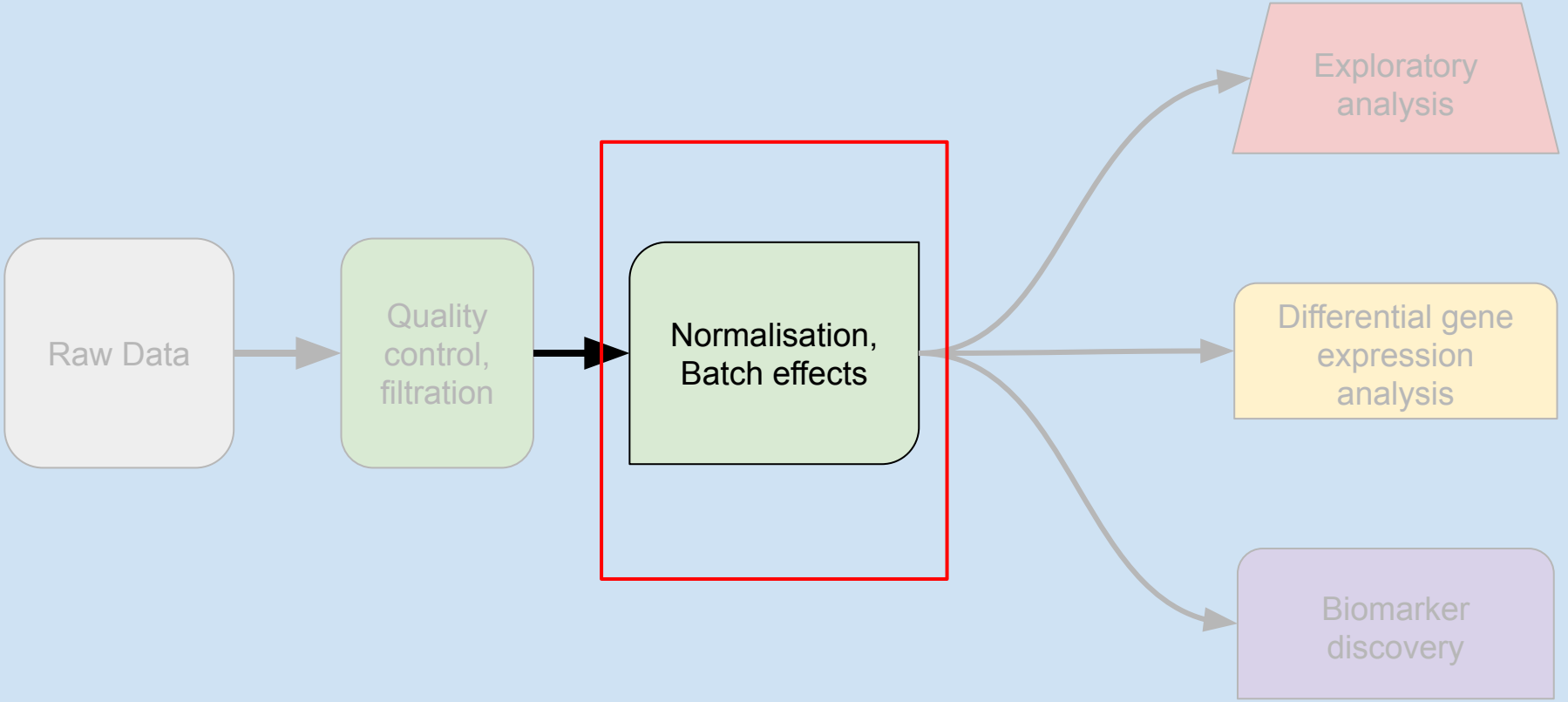


Raw data



Pre-processing of RNA-seq Data

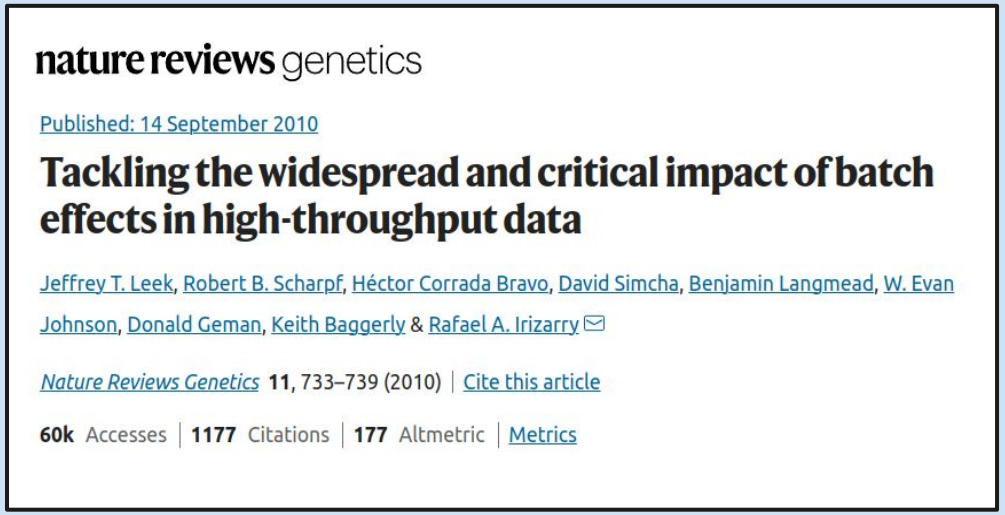
Batch effects


Raw Data
Quality control, filtration
Normalisation, Batch effects
Exploratory analysis
Differential gene expression analysis
Biomarker discovery

Batch Effects

Effects in the data from **non-biological sources**

- Experiment date
- Instrument used
- Atmospheric conditions
- Technician
- Laboratory



nature reviews genetics

Published: 14 September 2010

Tackling the widespread and critical impact of batch effects in high-throughput data

Jeffrey T. Leek, Robert B. Scharpf, Héctor Corrada Bravo, David Simcha, Benjamin Langmead, W. Evan Johnson, Donald Geman, Keith Baggerly & Rafael A. Irizarry

Nature Reviews Genetics **11**, 733–739 (2010) | Cite this article

60k Accesses | **1177** Citations | **177** Altmetric | Metrics

Unaccounted batch effects can dramatically alter the validity of findings

How to detect and correct batch effects?

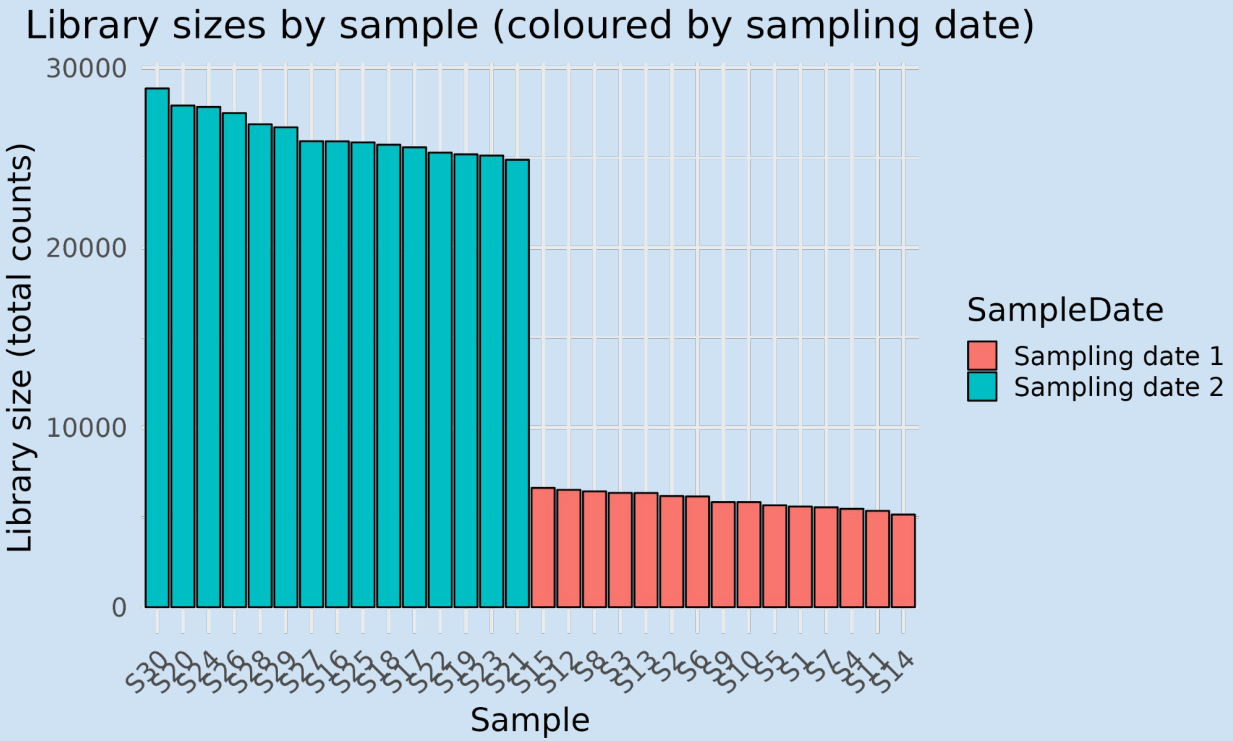
Detection

- Exploratory analysis!
  - **Visualise reduced-dimension data as a function of batch variables (PCA)**
  - Principal variance component analysis (PVCA)

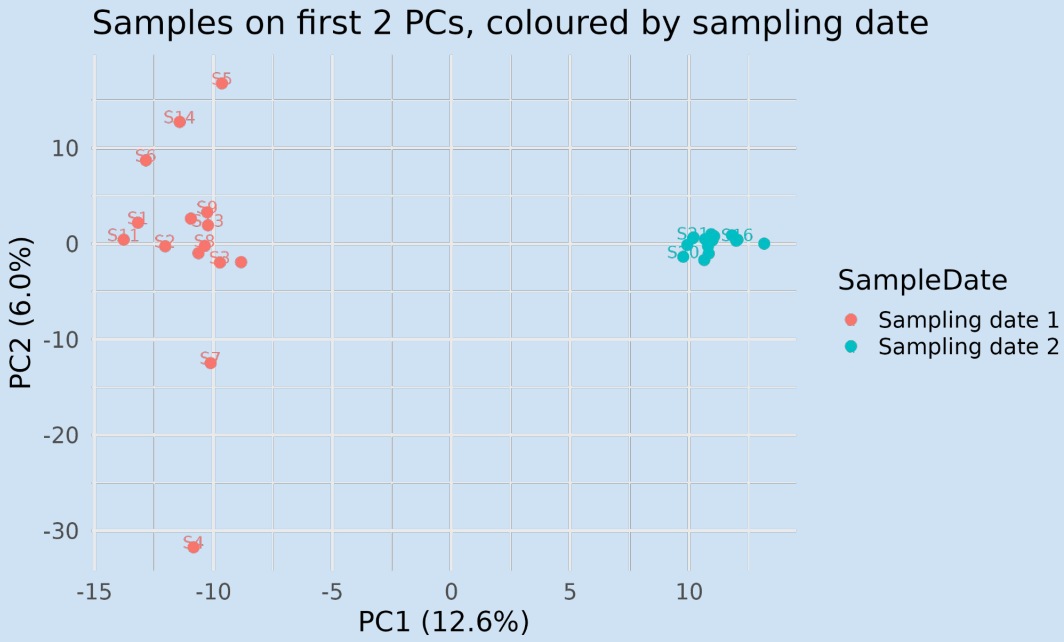
Correction

- Estimate and remove batch effect prior to analysis : e.g. **ComBat-Seq package** in R
  - No detail given here but we will see this in the practical

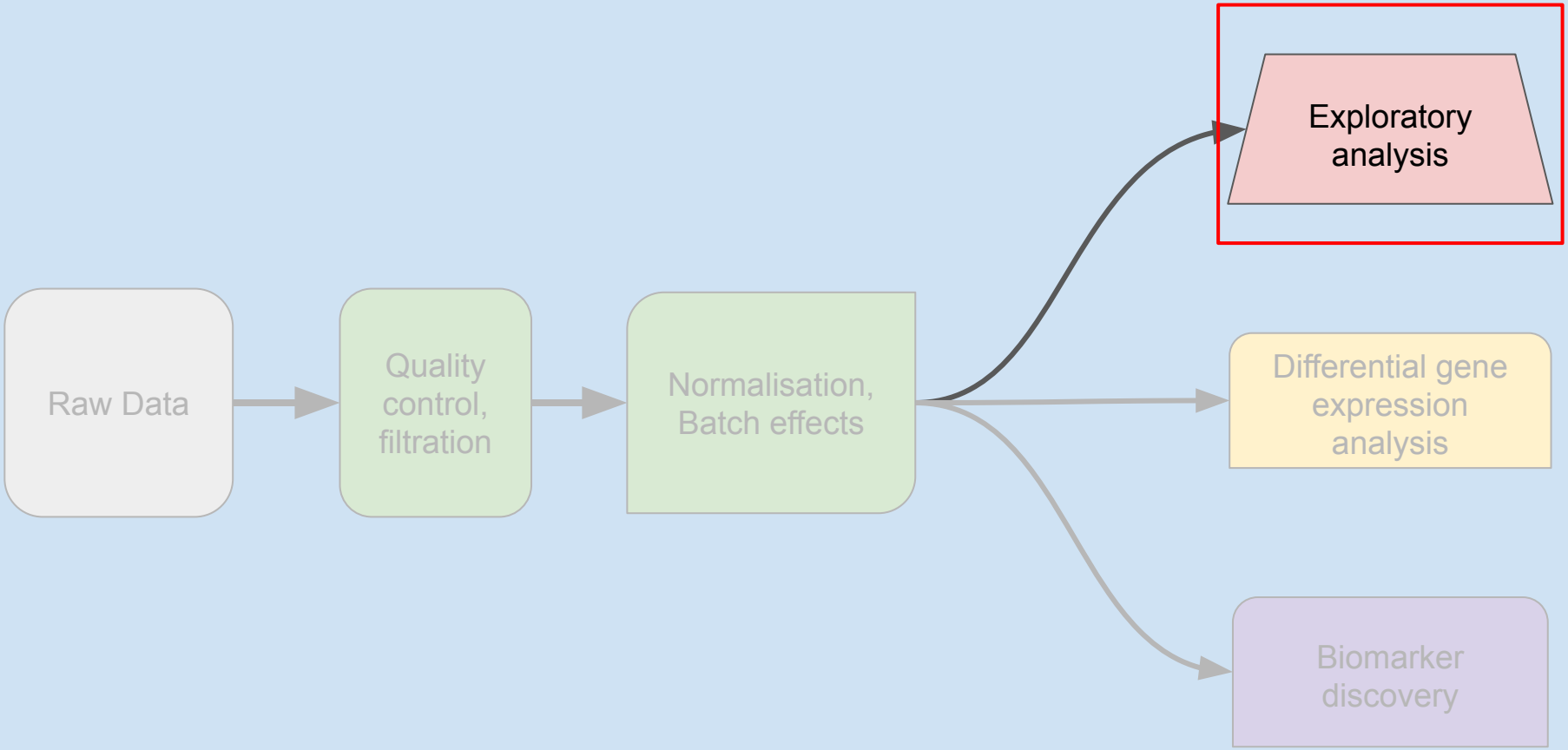
Example of a batch effect



Detecting batch effect with PCA

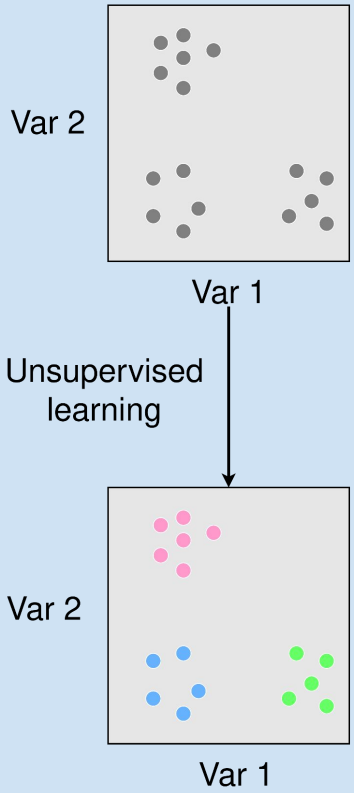


Exploratory Analysis :
Hierarchical Clustering


Raw Data
Quality control, filtration
Normalisation, Batch effects
Exploratory analysis
Differential gene expression analysis
Biomarker discovery

Unsupervised learning

- Exploring patterns in data
- No “correct” response
- PCA, clustering

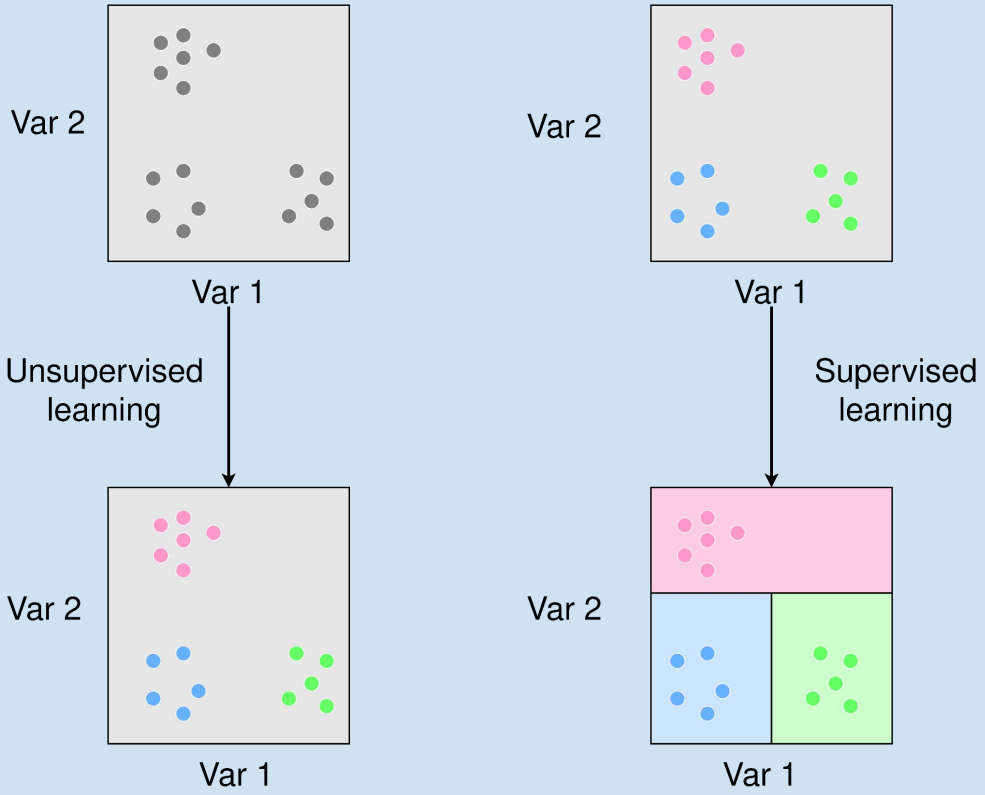


Unsupervised learning

- Exploring patterns in data
- No “correct” response
- PCA, clustering

Supervised learning

- Learn the relationship between predictors and a response
- Linear regression, random forest,...

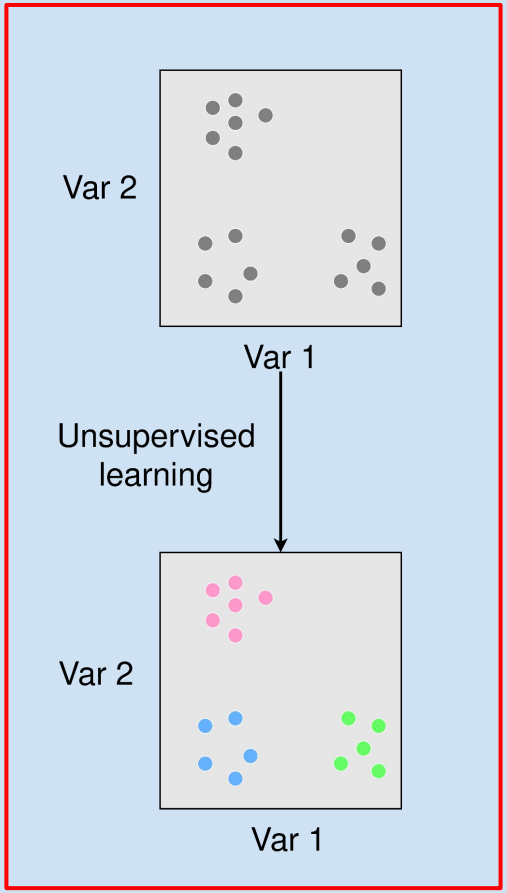


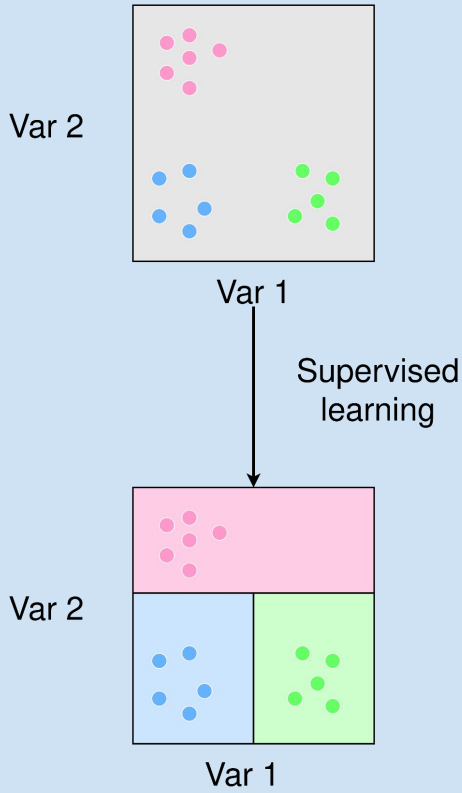
Unsupervised learning

- Exploring patterns in data
- No “correct” response
- PCA, clustering

Supervised learning

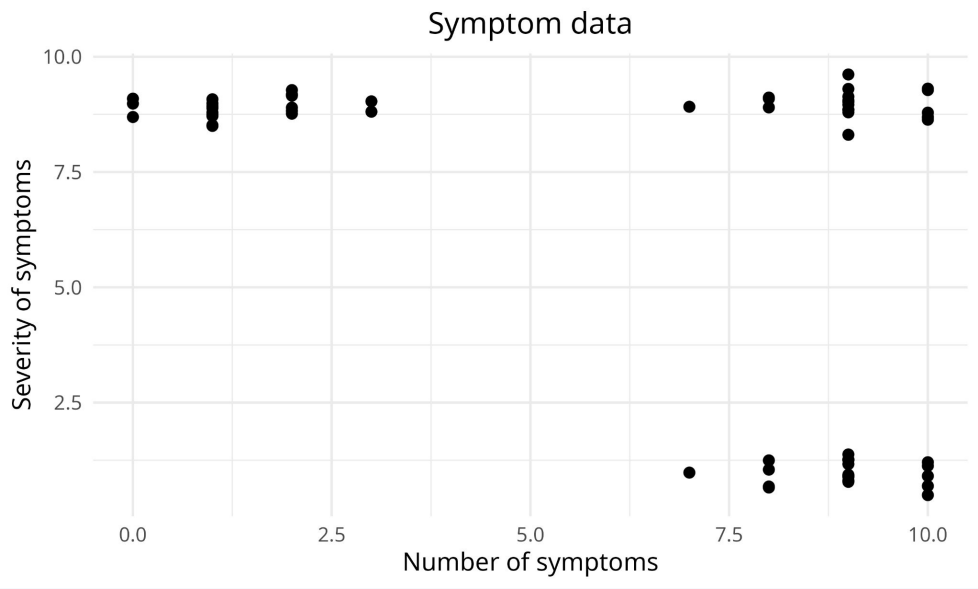
- Learn the relationship between predictors and a response
- Linear regression, random forest,...





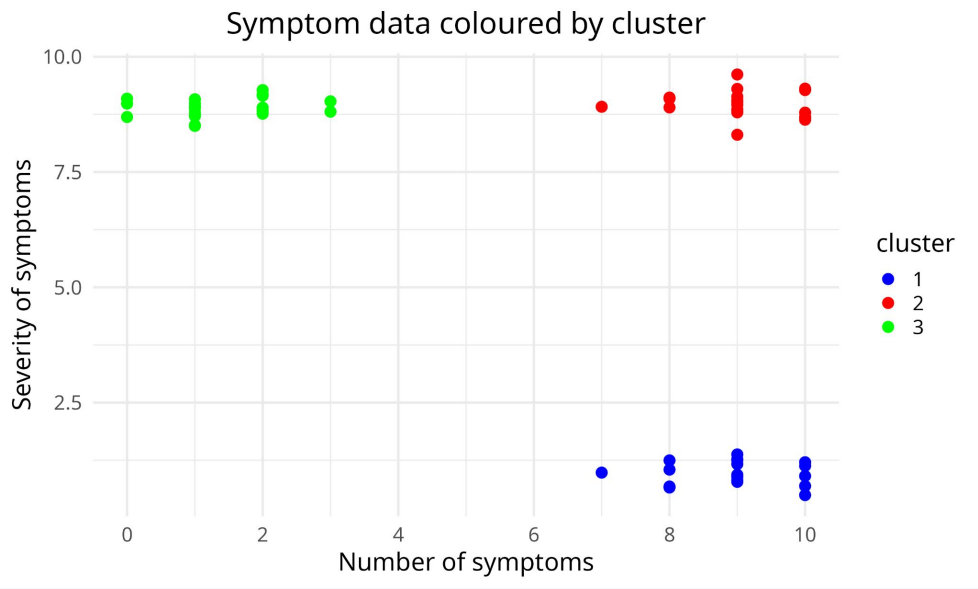
Motivation

***Unsupervised learning* to infer groups** (clusters) from data



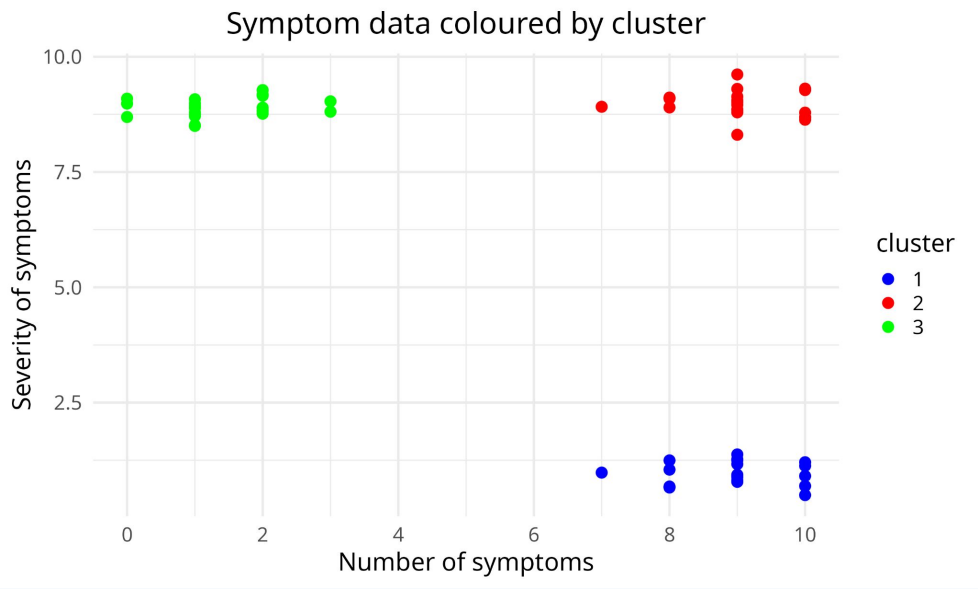
Motivation

***Unsupervised learning* to infer groups** (clusters) from data



Motivation

***Unsupervised learning* to infer groups** (clusters) from data



- Data within a cluster is similar
- Clusters are different from each other

What does *similar* mean?

What does *similar* mean?

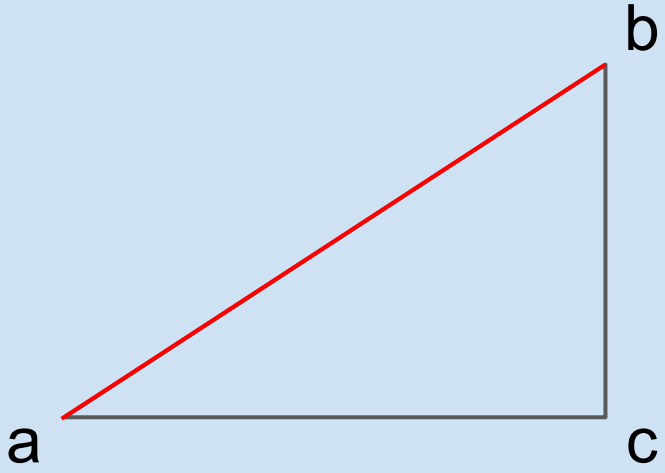
Similarity between data points is formalised with ***distance metrics***

What does *similar* mean?

Similarity between data points is formalised with ***distance metrics***

For a point (a,b), a **distance metric is a function d(,) such that**

- $d(a,b) \geq 0$
- $d(a,b) = 0$ if and only if $x = y$
- $d(a,b) = d(b,a)$
- $d(a,c) \leq d(a,b) + d(b,c)$



Euclidean distance

Data points are p-dimensional vectors:

$\mathbf{a} = (a_1,...,a_p)$, $\mathbf{b} = (b_1,...,b_p)$

Euclidean distance

Data points are p-dimensional vectors:

a = $(a_1,...,a_p)$, **b** = $(b_1,...,b_p)$

Euclidean distance is defined as

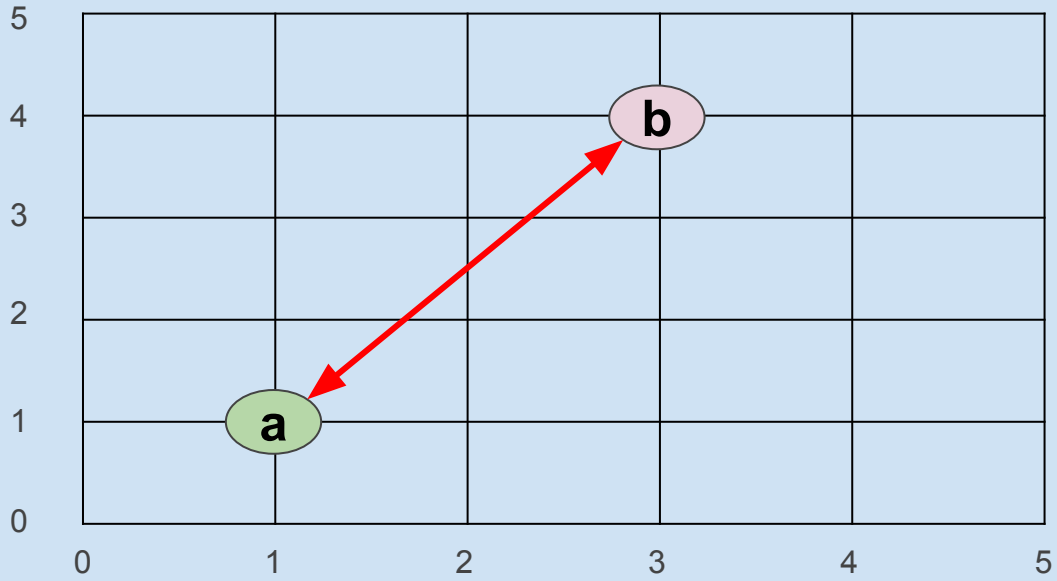
$$d_{\text{euclidean}}(\boldsymbol{a}, \boldsymbol{b}) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \cdots + (a_p - b_p)^2}$$

$$= \sqrt{\sum_{i=1}^{p} (a_i - b_i)^2}$$

Euclidean distance in 2 dimensions

a = (1, 1)

b = (3, 4)

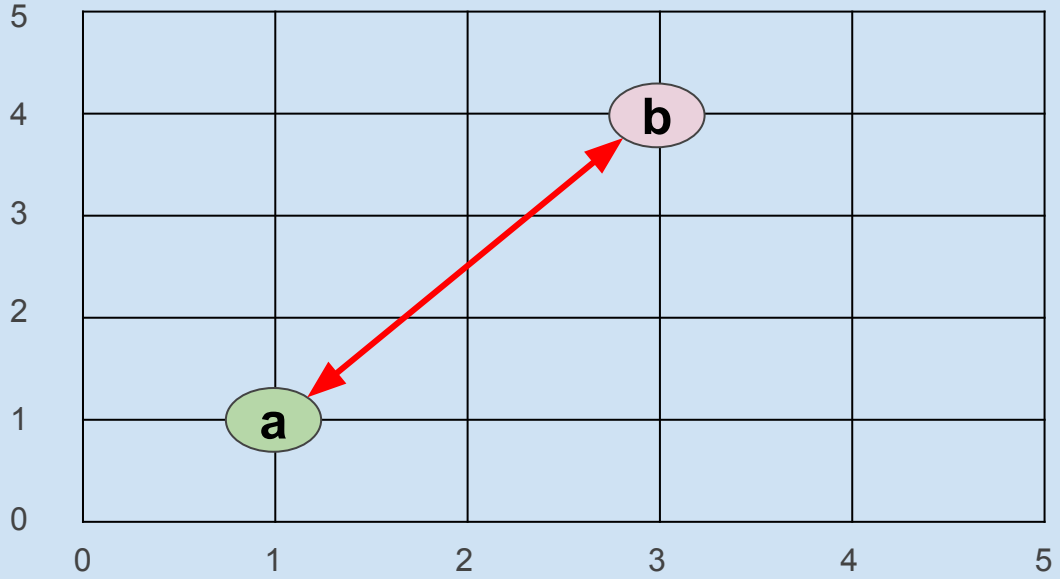


Euclidean distance in 2 dimensions

a = (1, 1)

b = (3, 4)

$$
\begin{aligned}
d_{\text{euclidean}}(\boldsymbol{a}, \boldsymbol{b}) &= \sqrt{(1-3)^2 + (1-4)^2} \\
&= \sqrt{(-2)^2 + (-3)^2} \\
&= \sqrt{4+9} \\
&= \sqrt{13}
\end{aligned}
$$



Manhattan distance

a = $(a_1,...,a_p)$, **b** = $(b_1,...,b_p)$

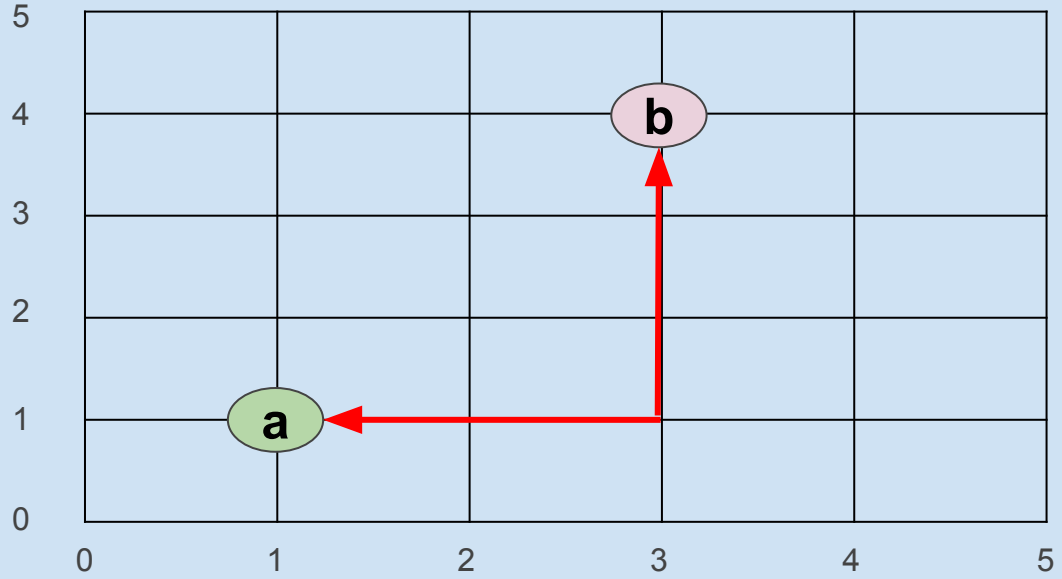
$$d_{\text{manhattan}}(\boldsymbol{a}, \boldsymbol{b}) = |a_1 - b_1| + |a_2 - b_2| + \cdots + |a_p - b_p|$$
$$= \sum_{i=1}^{p} |a_p - b_p|$$

Manhattan distance in 2 dimensions

a = (1, 1)

b = (3, 4)

$$
\begin{aligned}
d_{\text{manhattan}}(\boldsymbol{a}, \boldsymbol{b}) &= |1 - 3| + |1 - 4| \\
&= |-2| + |-3| \\
&= 2 + 3 \\
&= 6
\end{aligned}
$$



Maximum distance

a = $(a_1,...,a_p)$, **b** = $(b_1,...,b_p)$

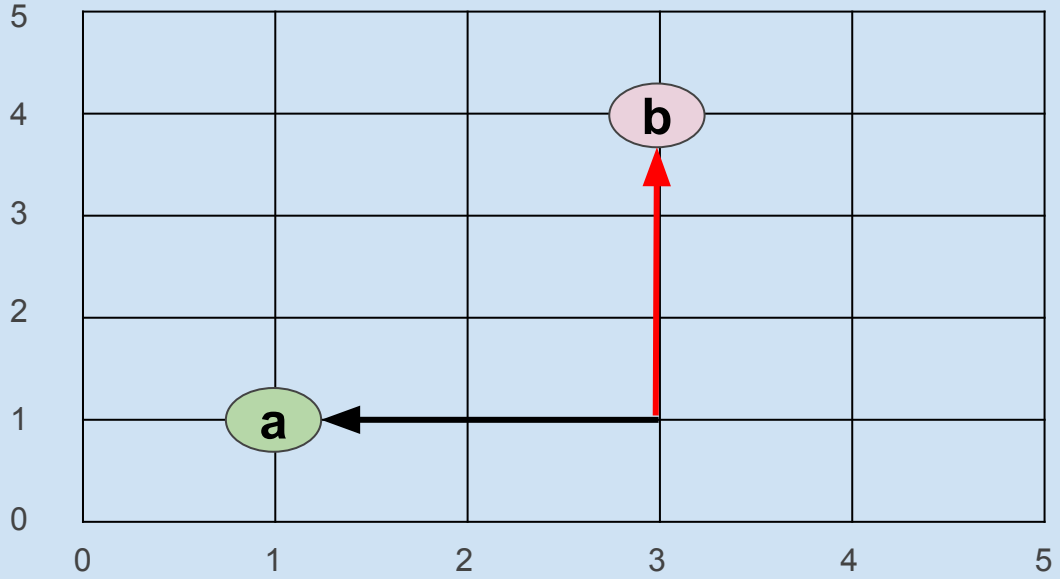
$$d_{\max}(\boldsymbol{a},\boldsymbol{b}) = \max(|a_1 - b_1|, |a_2 - b_2|, \cdots, |a_p - b_p|)$$
$$= \max_{i \in 1,...,p} |a_p - b_p|$$

Maximum distance in 2 dimensions

a = (1, 1)

b = (3, 4)

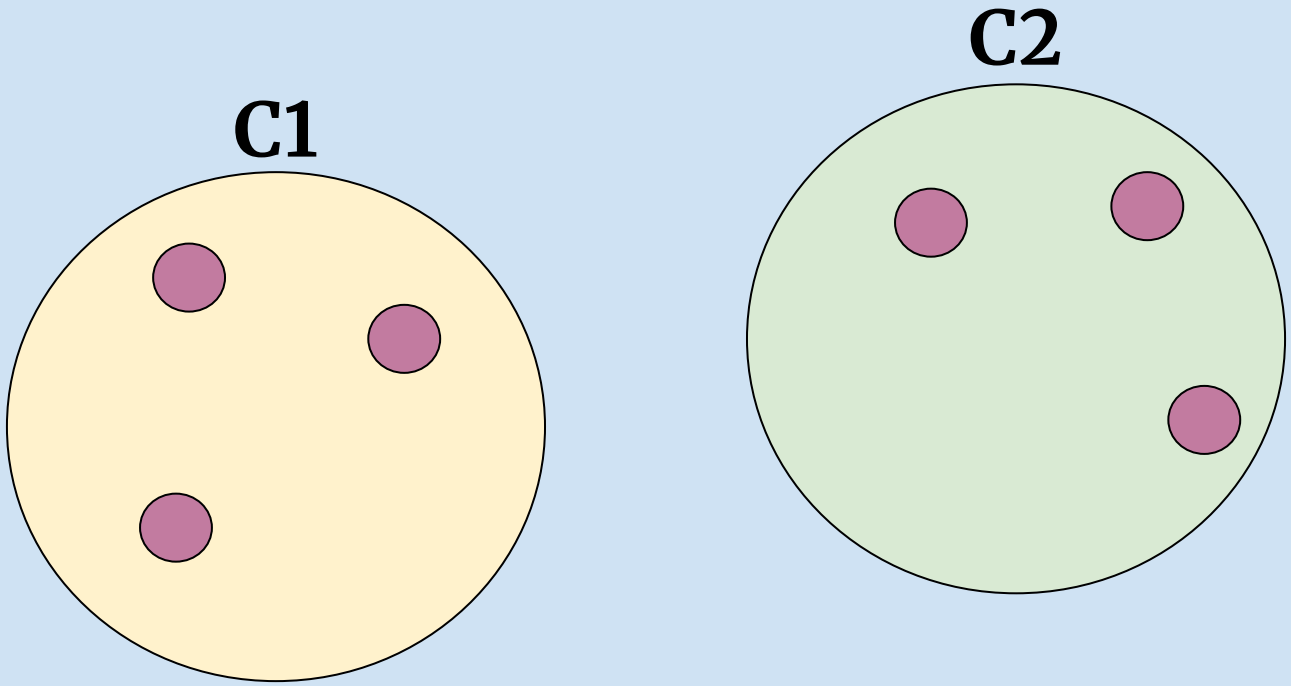
$$
\begin{aligned}
d_{\max}(\boldsymbol{a}, \boldsymbol{b}) &= \max(|1-3|, |1-4|) \\
&= \max(2, 3) \\
&= 3
\end{aligned}
$$



Distance metrics summary

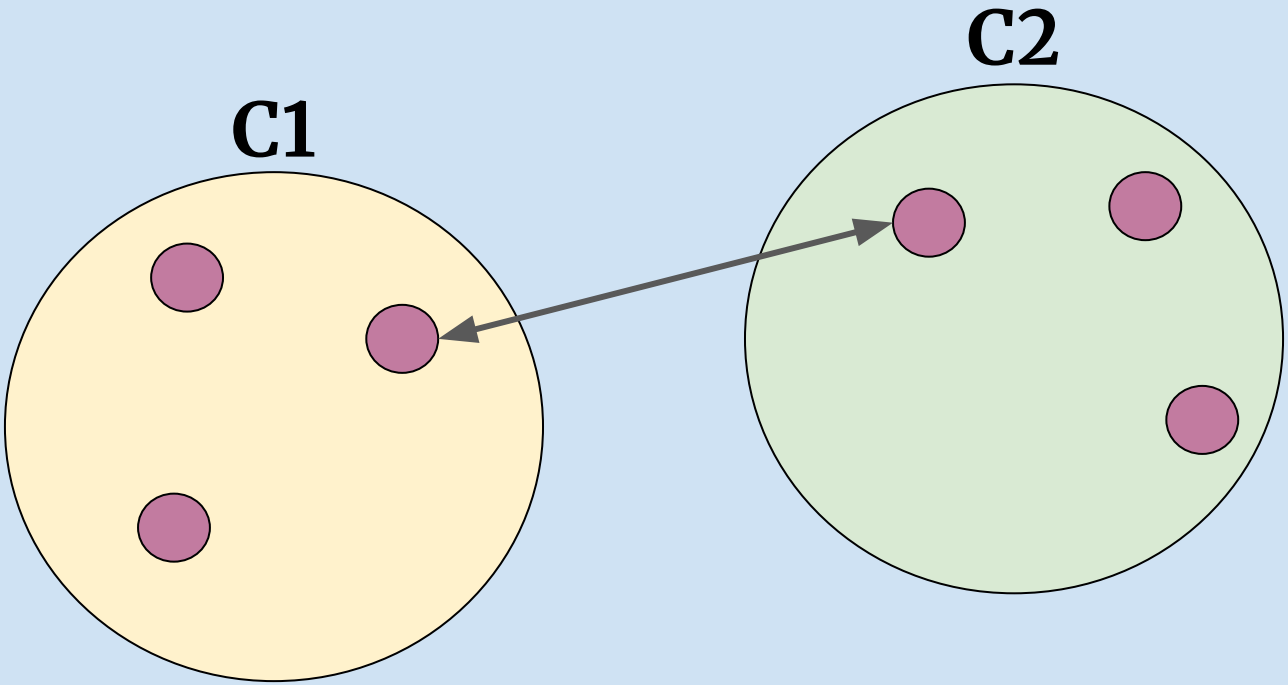
- Data points are represented mathematically by p-dimensional vectors
- **Distance metrics formalise similarity between data points**
  - A distance metric is a function taking two points as inputs and outputting a single value
  - Valid distance metrics must respect certain conditions
- We introduced three different distance metrics
  - Euclidean distance is the direct route between points
  - Manhattan distance is the distance travelled along a grid
  - Maximum distance is the maximum of the paths along a grid

How can we define distance between clusters?



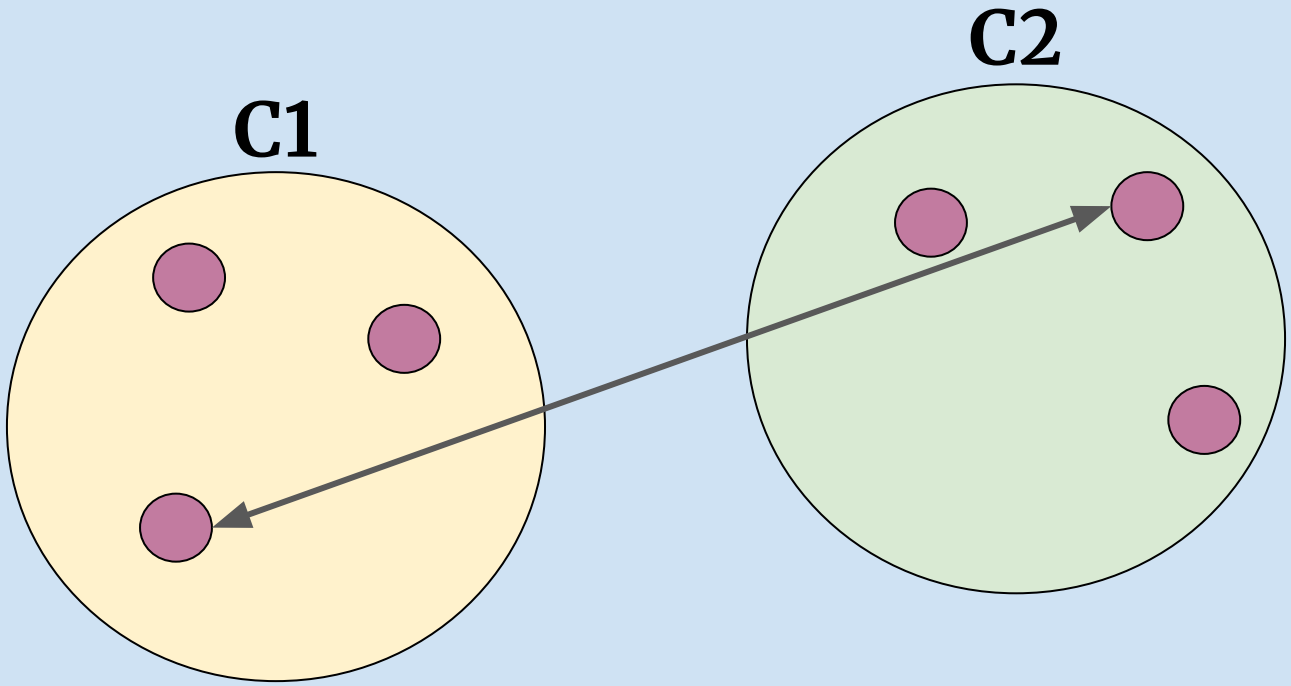
Single linkage

Distance between closest points



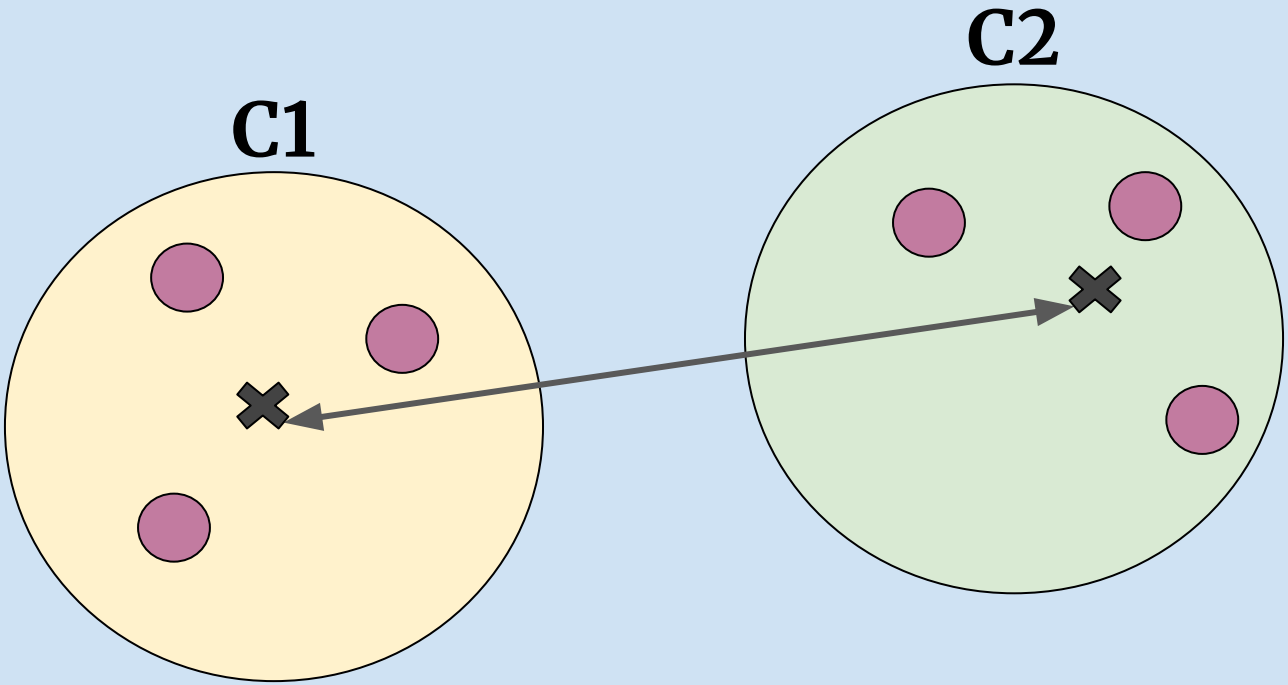
Complete linkage

Distance between farthest points



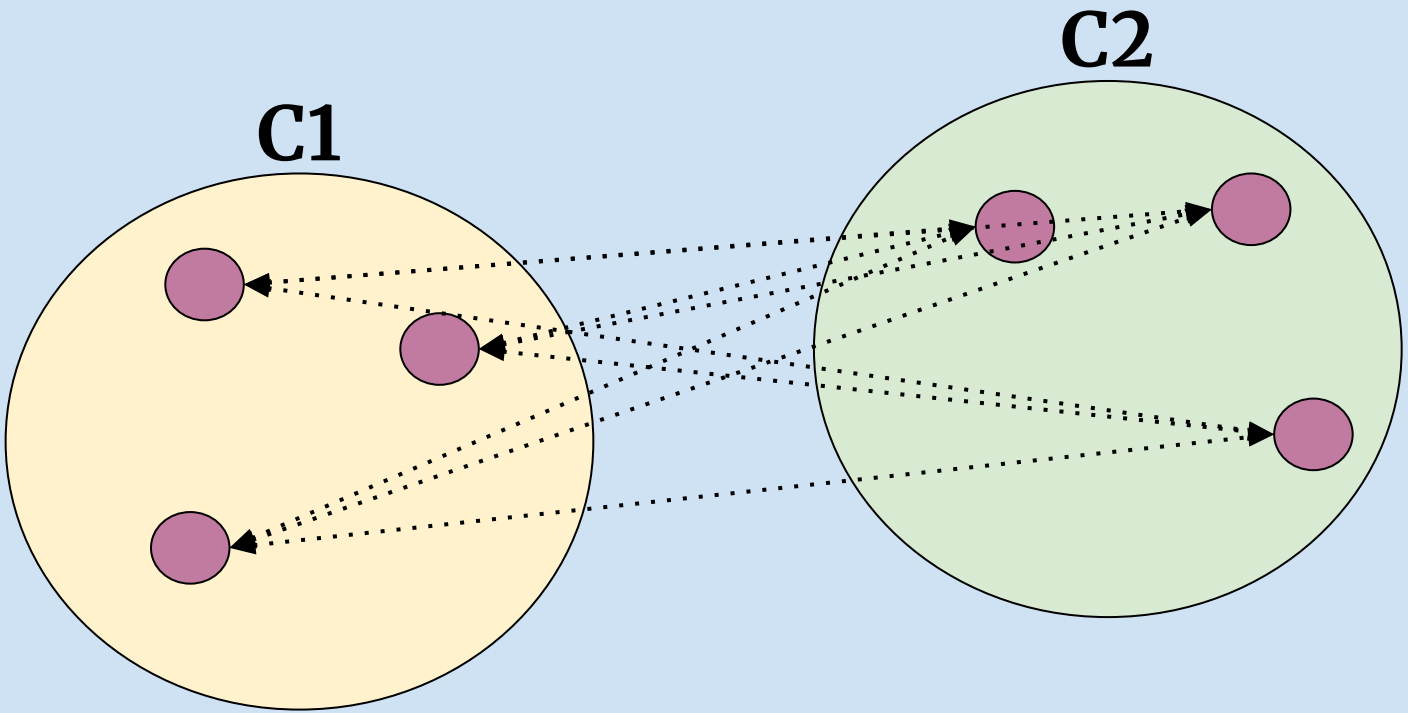
Centroid linkage

Distance between cluster “centers”



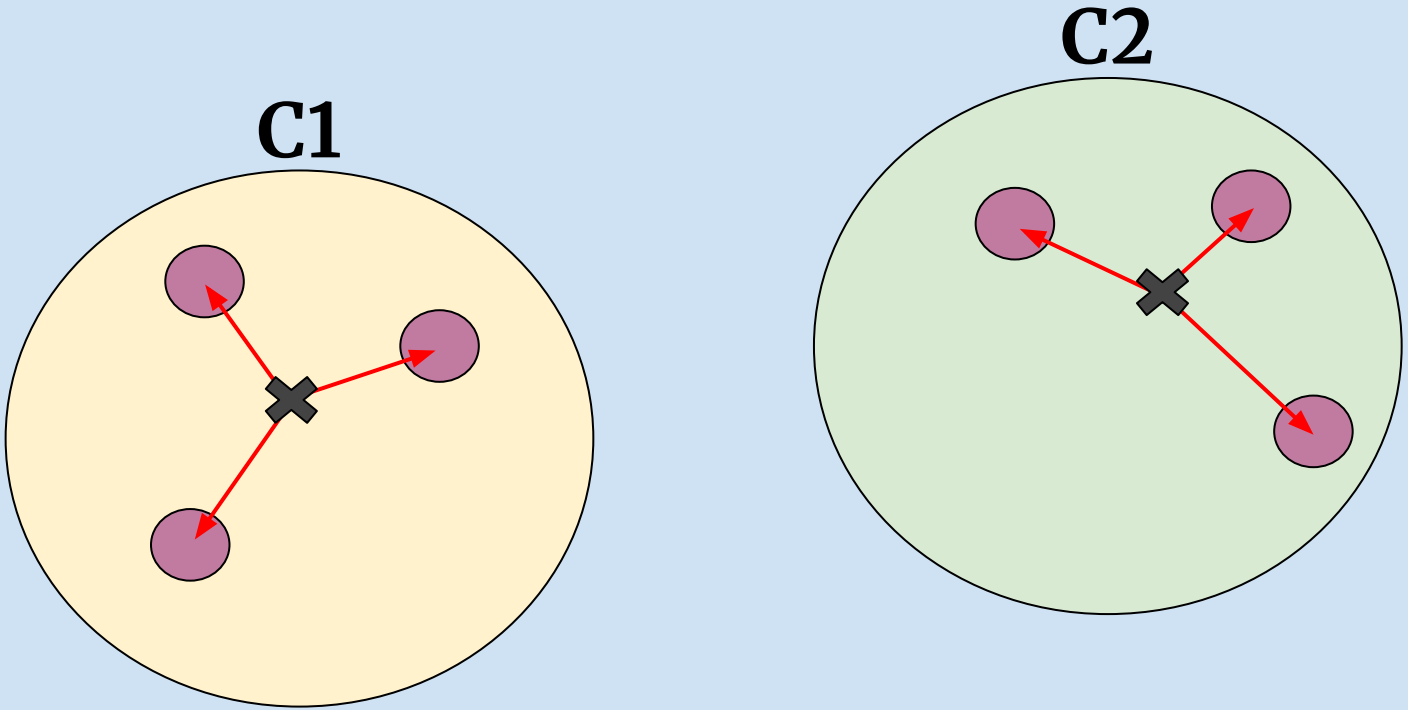
Average linkage

Average distance between all points



Ward's method

Average within-cluster variance



What is *Hierarchical* clustering?

Algorithms which sort observations into a nested *hierarchies*

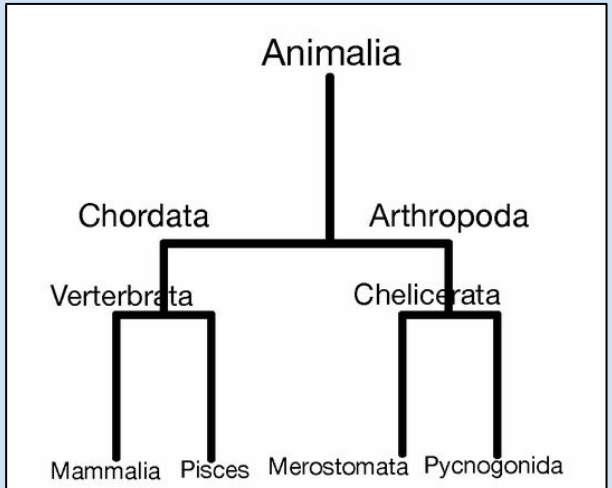


Figure 5.16: A snippet of Linnus' taxonomy that clusters organisms according to feature similarities.

Hierarchical clustering

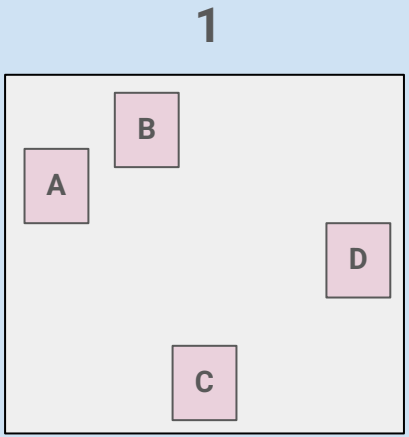
Goal: organise samples into clusters without choosing the number of clusters beforehand

This requires

1. A dataset of samples and features
2. A **distance metric** defining similarity between samples
3. A **linkage function** defining similarity between clusters

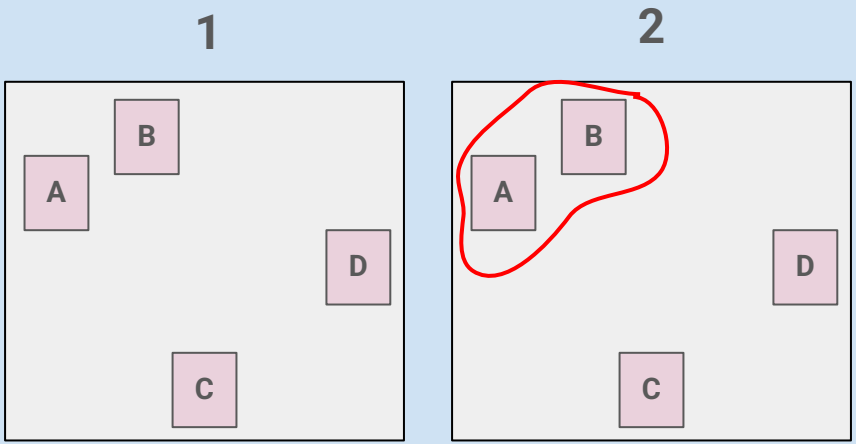
Agglomerative Hierarchical Clustering

Agglomerative = start with each point as its own cluster, finish with all points in one cluster



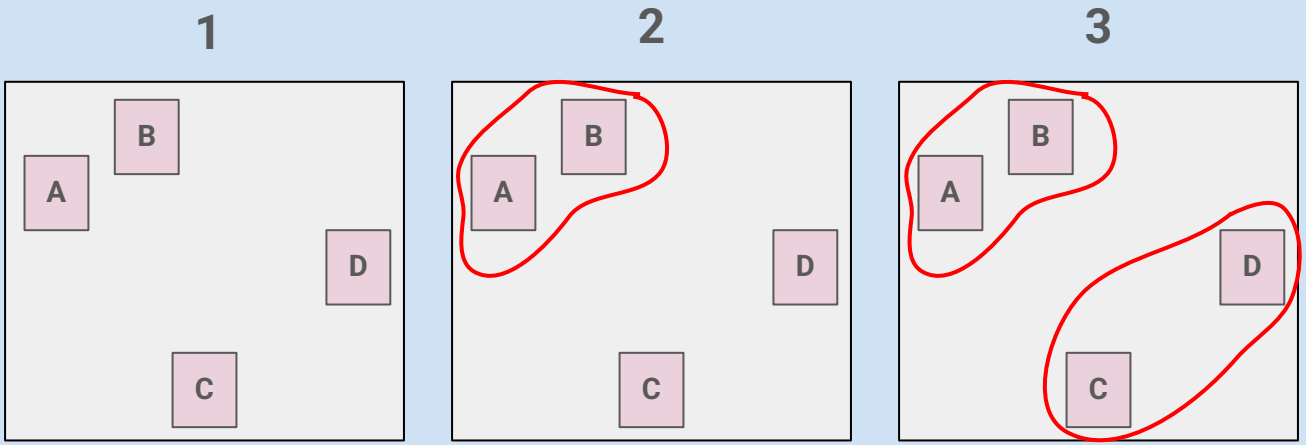
Agglomerative Hierarchical Clustering

Agglomerative = start with each point as its own cluster, finish with all points in one cluster



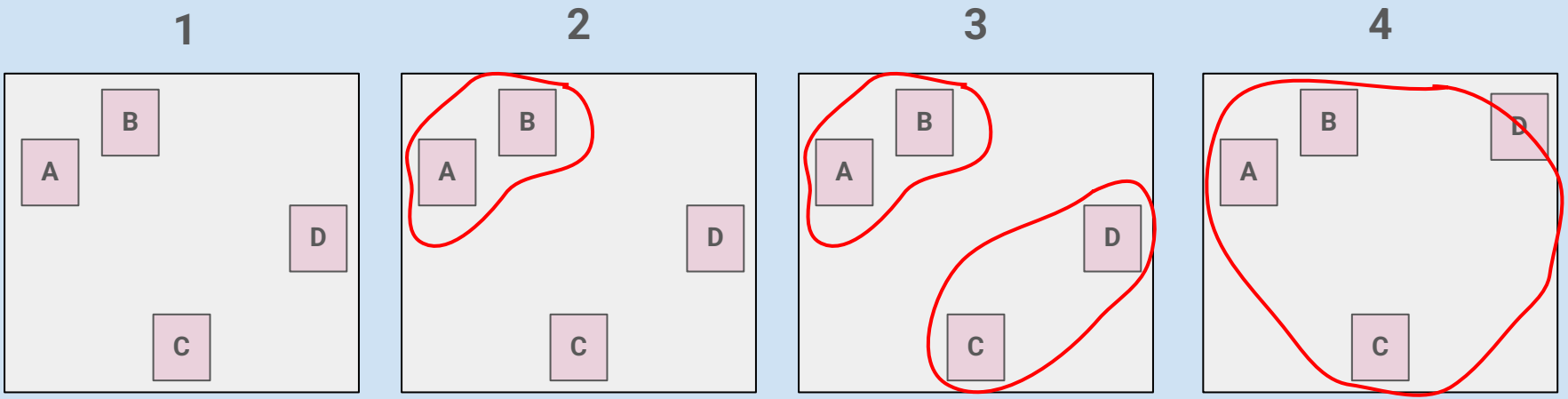
Agglomerative Hierarchical Clustering

Agglomerative = start with each point as its own cluster, finish with all points in one cluster



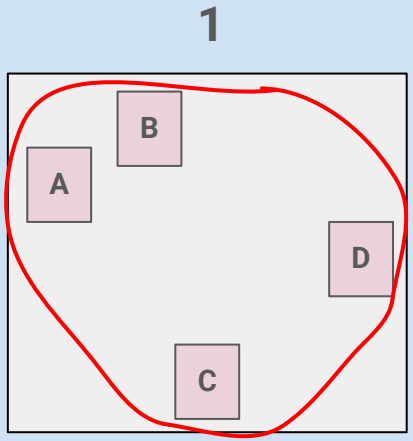
Agglomerative Hierarchical Clustering

Agglomerative = start with each point as its own cluster, finish with all points in one cluster



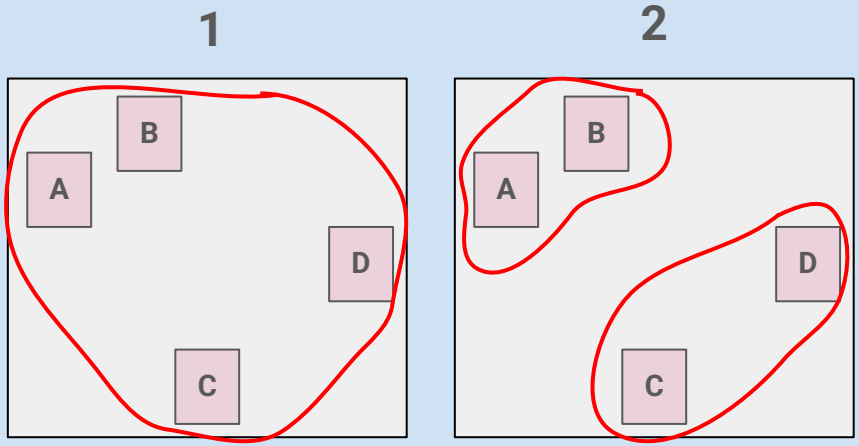
Divisive Hierarchical Clustering

Divisive = start with all points in one cluster, finish with each point as its own cluster



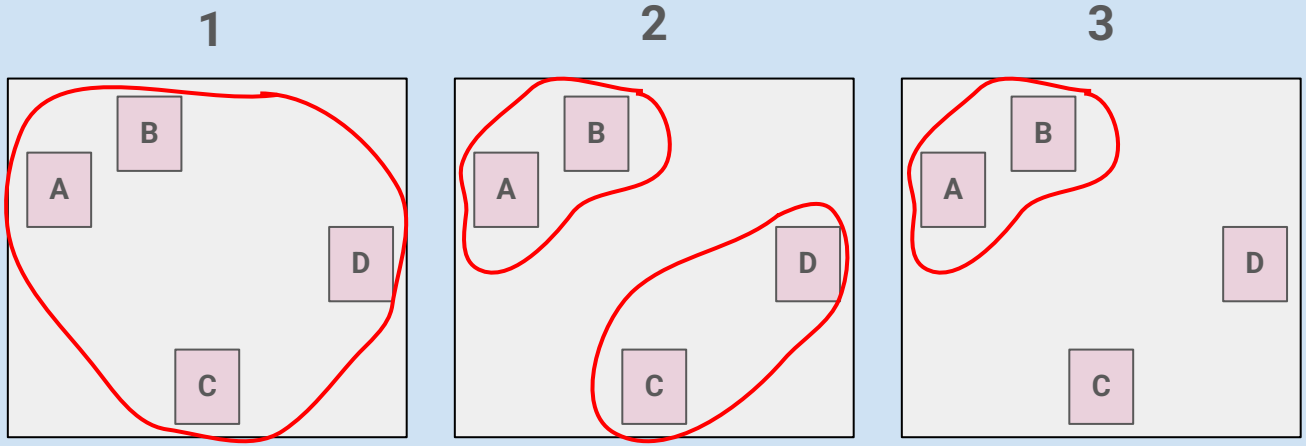
Divisive Hierarchical Clustering

Divisive = start with all points in one cluster, finish with each point as its own cluster



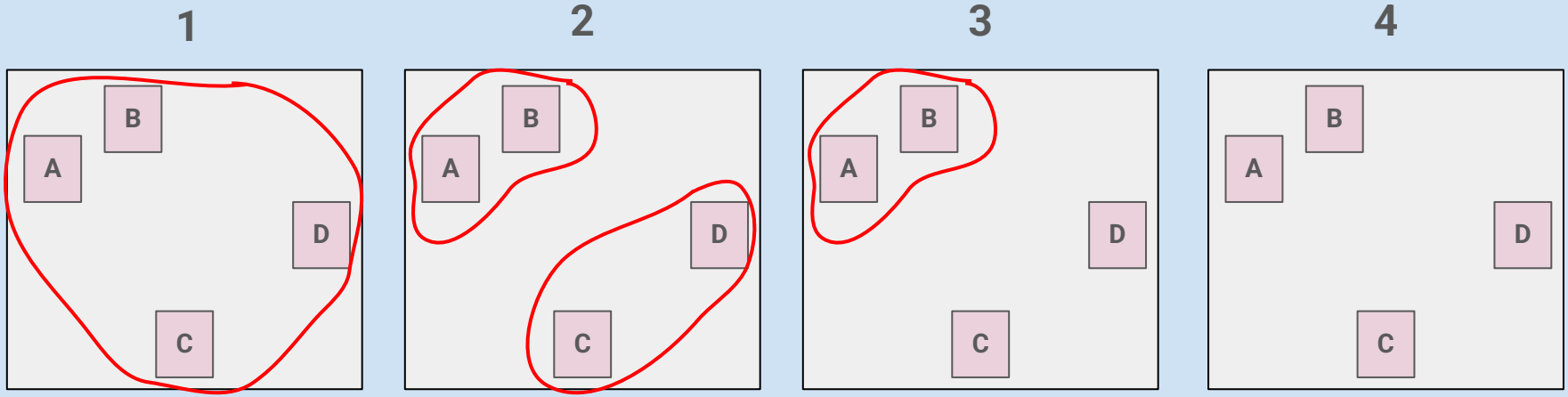
Divisive Hierarchical Clustering

Divisive = start with all points in one cluster, finish with each point as its own cluster



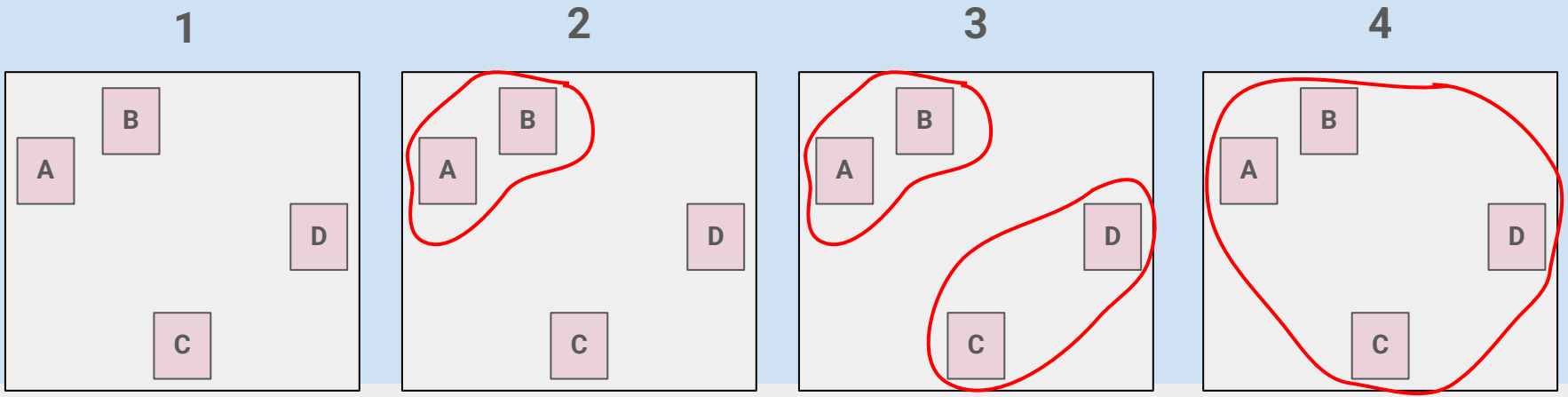
Divisive Hierarchical Clustering

Divisive = start with all points in one cluster, finish with each point as its own cluster



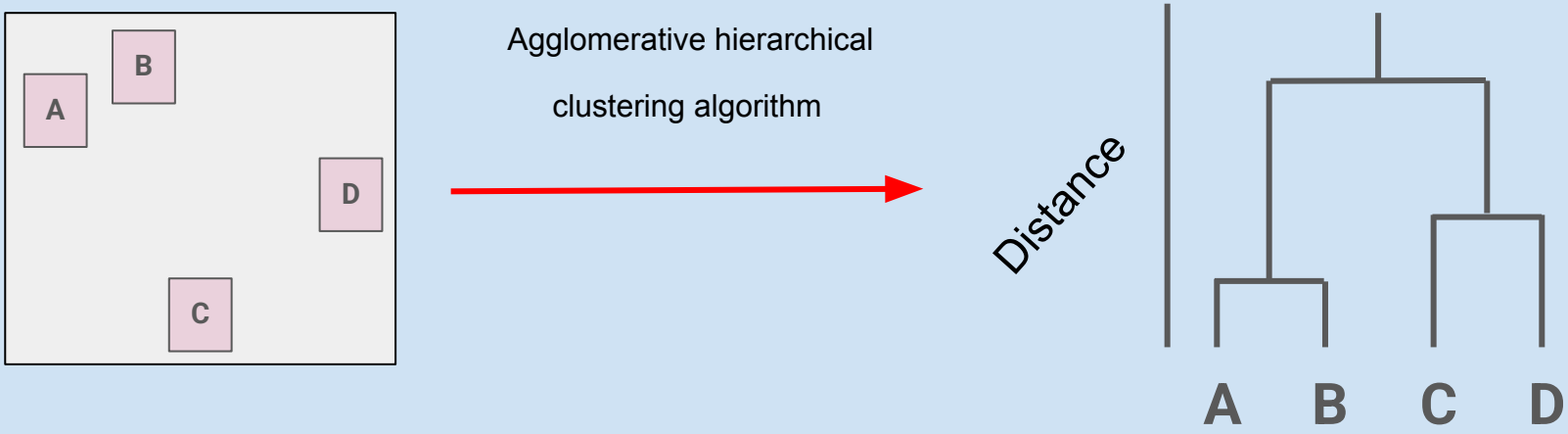
From the algorithm to clusters

- Hierarchical clustering algorithms produce a **series of steps each defining a different set of clusters**
- We must choose at which point to “stop” and interpret the clusters
- How could we decide this?



Dendrogram

- **Visual representation of the iterations of a hierarchical clustering algorithm**



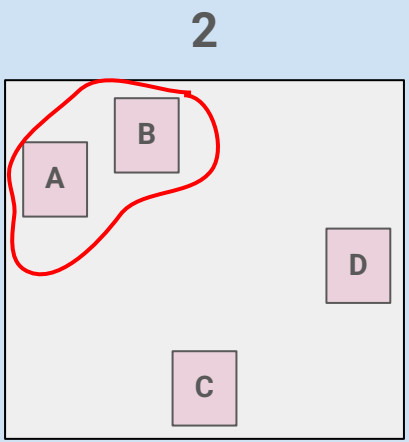
How to construct a dendrogram?

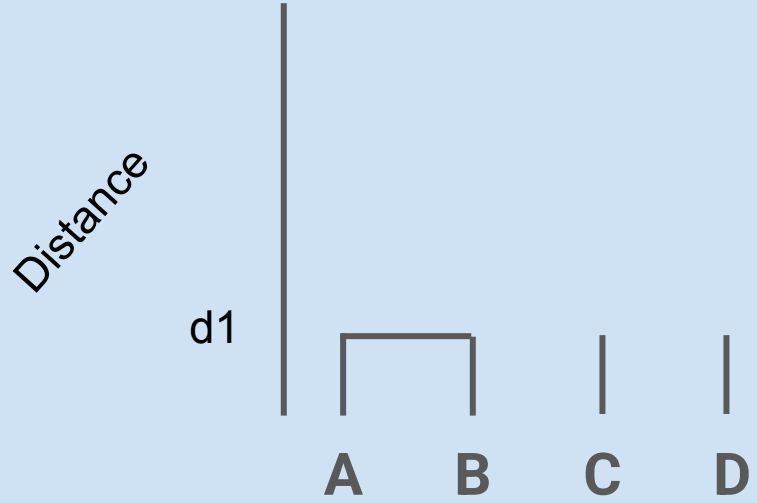
Stage 1: no clusters



How to construct a dendrogram?

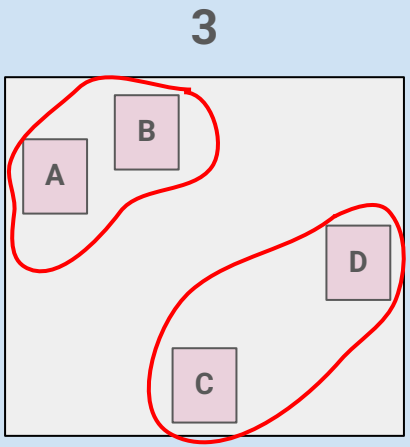
Stage 2: A and B join with some distance d1

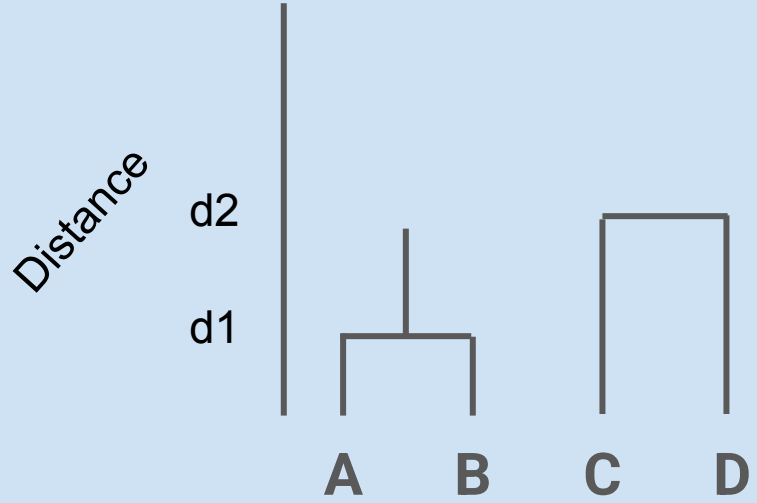




How to construct a dendrogram?

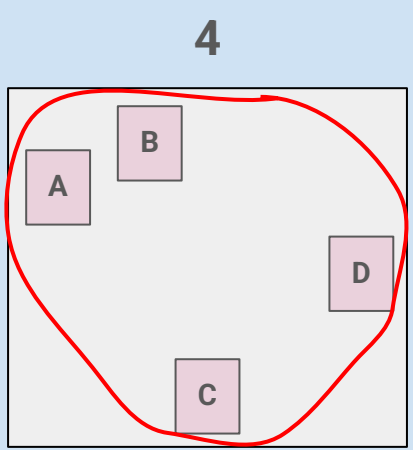
Stage 3: C and D join with some distance d2

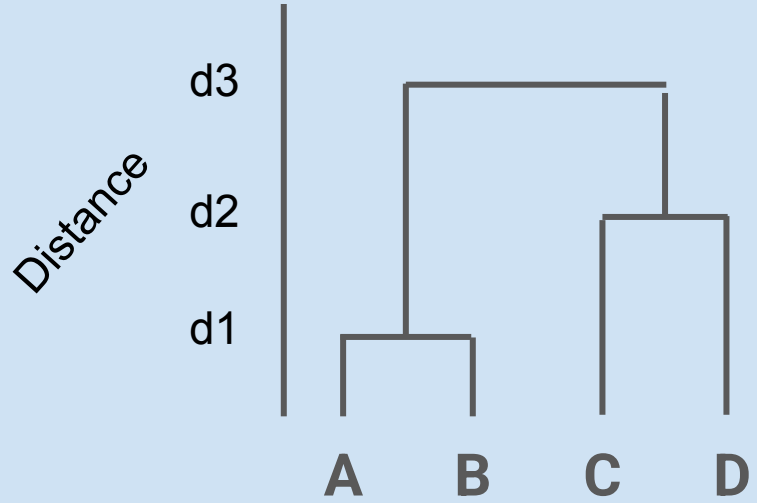




How to construct a dendrogram?

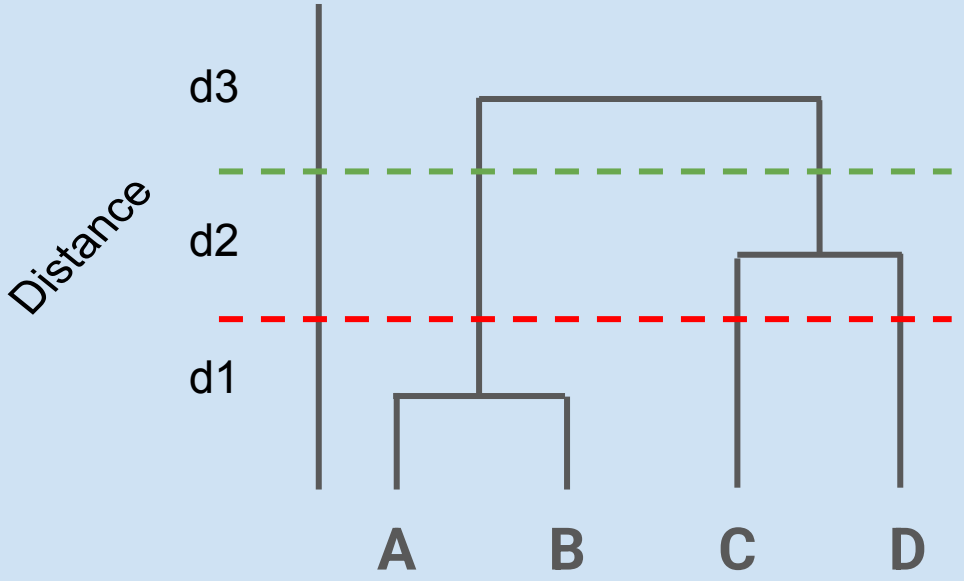
Stage 4: the cluster containing A and B joins with the cluster containing C and D with some distance d3





How to interpret dendrograms?

- Vertical axis gives **distance** at which clusters were **merged**
- ***Cutting* a dendrogram** at a given point therefore **defines a set of clusters**



Hierarchical Clustering summary

Hierarchical clustering inputs

- **Distance metric** : a measure of similarity between points
- **Linkage function :** a way to measure similarity between groups

HC can be interpreted using dendrograms

- **True clusters are unknown**, hierarchical clustering is an **exploratory, unsupervised method**

Practical work

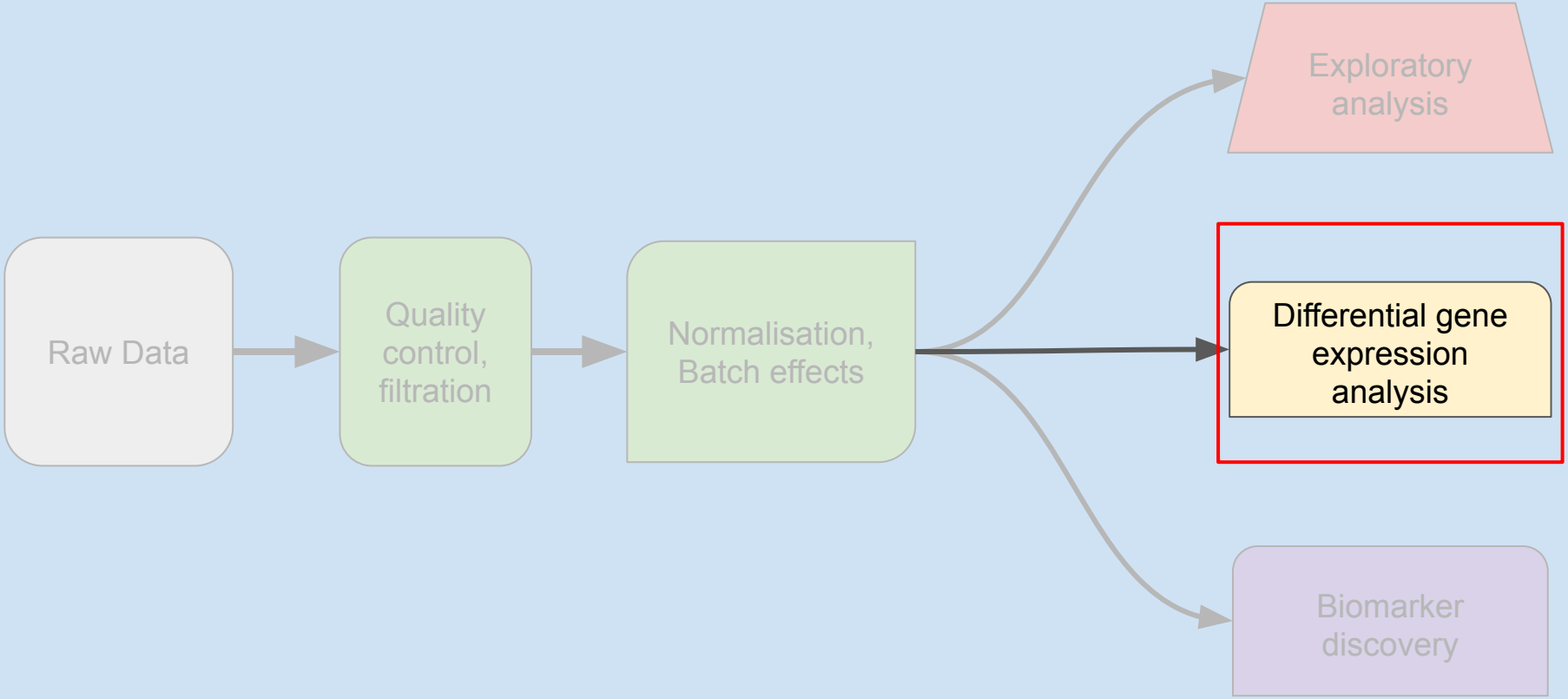
We are going to re-analyse a dataset containing transcriptomic samples from healthy patients and patients infected with mycobacterium tuberculosis, which causes tuberculosis disease. In this first practical, we will focus on

- Normalisation and batch effects
- Exploratory analysis using PCA and hierarchical clustering

Complete parts 0,1,2, and 3 from the R markdown file *PHDS_omics_GMA_2025_questions.Rmd*

Differential Gene Expression Analysis

How can we model count data?


Raw Data
Quality control, filtration
Normalisation, Batch effects
Exploratory analysis
Differential gene expression analysis
Biomarker discovery

Why do we do statistical modelling?

Model: simplified description of how data are generated

Why do we do statistical modelling?

Model: simplified description of how data are generated

1. Generalise beyond the observed data
2. Quantify uncertainty
3. Summarise, interpret, predict

Why do we do statistical modelling?

Model: simplified description of how data are generated

1. Generalise beyond the observed data
2. Quantify uncertainty
3. Summarise, interpret, predict

Parametric model: assumption that the data follow a specified distribution controlled by a small number of *parameters*

- These parameters can be estimated from the data
- However, conclusions rely on the correctness of the assumptions

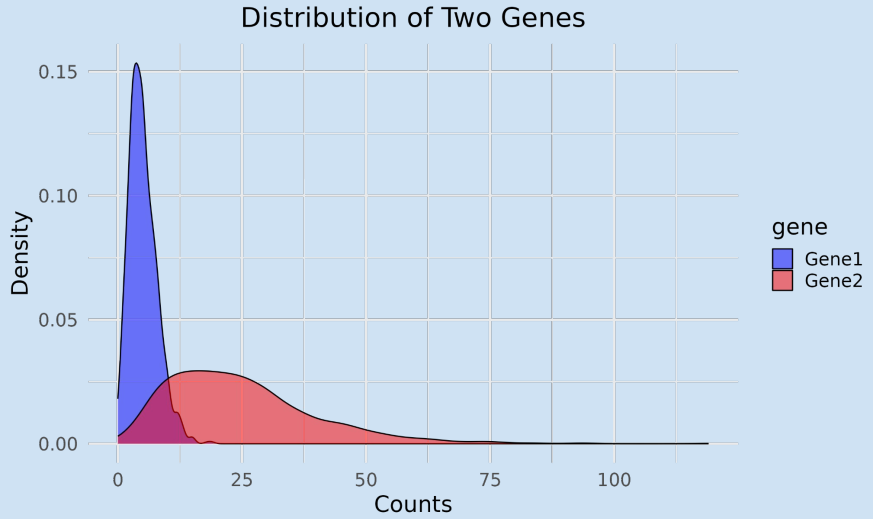
What is differential expression?

A gene is ***differentially expressed*** if there is a **statistically significant change in expression** between experimental conditions

- i.e. the change is greater than what would be expected due to random variation
- We need statistical modelling to assess this

Reminder on RNA-seq data challenges

- Non-negative, discrete (integer) values
- Non-symmetric distribution
- Large dynamic range
- Heteroskedastic (mean depends on variance)
- Often small sample sizes
- **Noise - random, technical and biological**



How can we model count data?

- In RNA-seq, we randomly sample r reads (mRNA fragments) from a *sequencing library*
- These events *can be thought of as* independent
- If a library contains n_1 fragments corresponding to gene 1, n_2 to gene 2, etc., the library size is $n = n_1 + n_2 + ...$
- The probability of sampling a read corresponding to gene g is therefore $p_g = n_g/n$
- The mean number of counts for a gene is $\lambda_g = r \times p_g$

Poisson - a distribution for count data

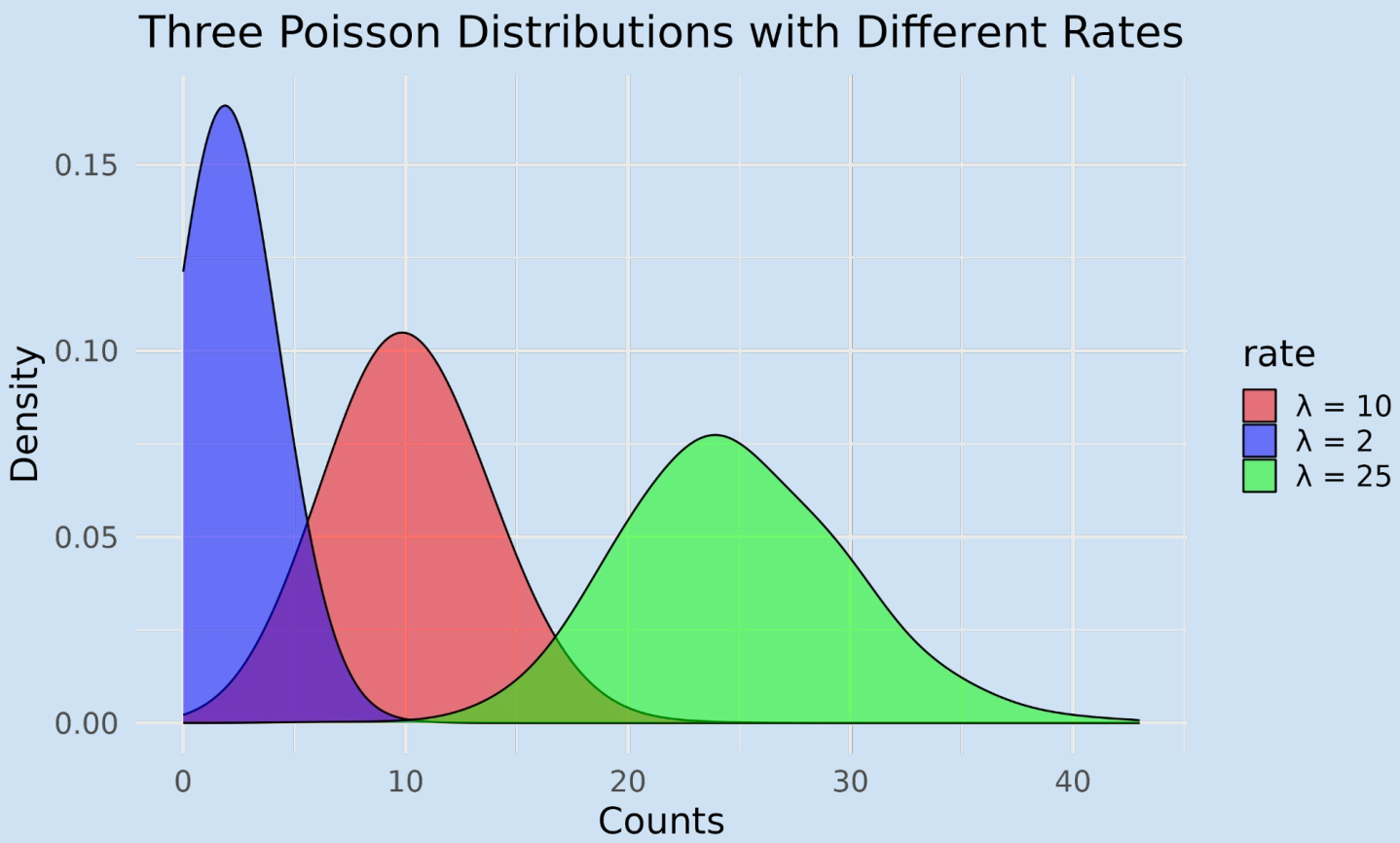
- Poisson distribution = number of events in a given period of time
- A random variable X follows a Poisson distribution with *rate parameter* $\lambda > 0$ if

$$\mathbb{P}(X = k) = \begin{cases} \dfrac{e^{-\lambda}\lambda^k}{k!}, & k = 0, 1, 2, \ldots, \\ 0, & \text{otherwise}, \end{cases}$$

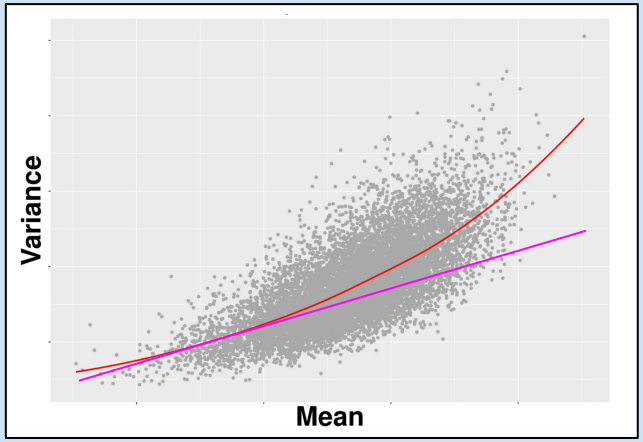
  Then $X \sim \text{Poisson}(\lambda)$.
- The expectation and variance of X are

$$\mathbb{E}[X] = \lambda, \quad \text{Var}(X) = \lambda.$$

- Could we model the counts for gene g, sample j as $y_{gj} \sim \text{Poisson}(\lambda_g)$?


Three Poisson Distributions with Different Rates
Density
0.15
0.10
0.05
0.00
0
10
20
30
40
Counts
rate
λ = 10
λ = 2
λ = 25

Reminder : RNA-seq is *overdispersed*



Mean vs variance for genes (simulated data)

- Pink line : mean = variance
- Red line : estimated mean-variance relationship
- **Mean-variance relationship is typically increasing and non-linear**

Negative Binomial Distribution

- Negative binomial = generalisation of Poisson with *overdispersion*
- A random variable X follows a Negative Binomial distribution with *mean* $\mu > 0$ and *dispersion* parameter $\theta > 0$, if

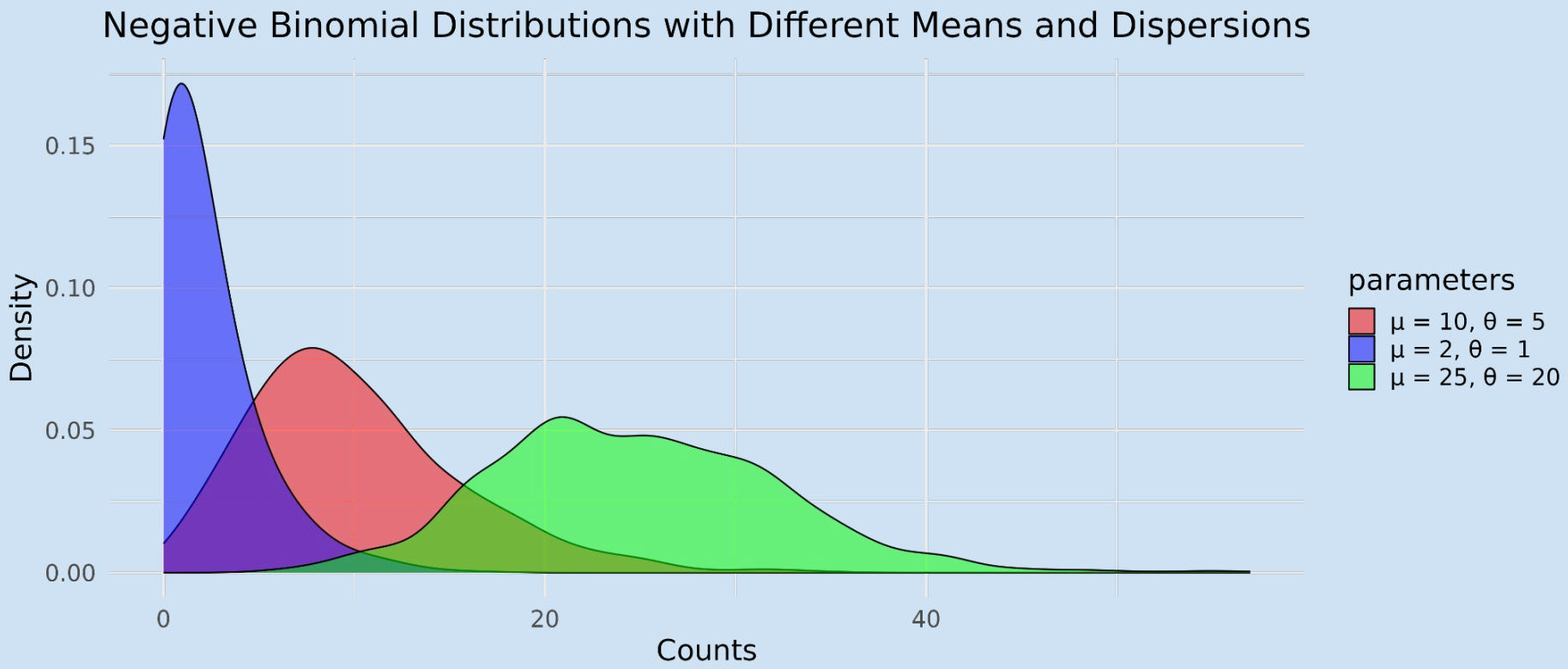
$$\mathbb{P}(X = k) = \begin{cases} \dfrac{\Gamma(k+\theta)}{k!\,\Gamma(\theta)} \left(\dfrac{\theta}{\theta+\mu}\right)^{\theta} \left(\dfrac{\mu}{\theta+\mu}\right)^{k}, & k = 0, 1, 2, \ldots, \\ 0, & \text{otherwise}, \end{cases}$$

  Then $X \sim \mathrm{NB}(\mu, \theta)$.

- The expectation and variance of X are

$$\mathbb{E}[X] = \mu, \quad \mathrm{Var}(X) = \mu + \frac{\mu^2}{\theta}.$$

- Could we model the counts for gene g, sample j as $y_{gj} \sim \mathrm{NB}(\lambda_g, \theta_g)$?


Negative Binomial Distributions with Different Means and Dispersions
Density
0.15
0.10
0.05
0.00
0
20
40
Counts
parameters
μ = 10, θ = 5
μ = 2, θ = 1
μ = 25, θ = 20

How to estimate dispersion parameters?

- Small sample size, large number of variables : difficult to estimate overdispersion for each gene

Popular packages using negative binomial modelling

- **DESeq2** : estimate mean-variance relationship across genes with regression
- **EdgeR** : borrow information across genes

Differential Gene Expression Analysis

DESeq2 and EdgeR Packages

DESeq2/EdgeR - estimating the mean

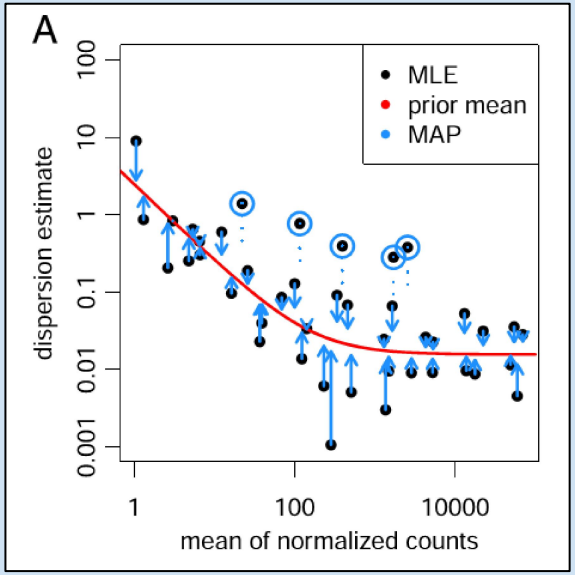
Estimating the mean

- Let x_j = experimental condition for sample j (i.e. 1 if treated, 0 if control)
- counts for gene g, sample j : $y_{gj} \sim NB(\mu_{gj}, \alpha_g)$
- mean $\mu_{gj} = s_j \times q_{gj}$ depends on
  - Sample-specific *normalisation factor* s_j
  - q_{gj} abundance of g in sample j
- Estimate $log(q_{gj}) = \beta_0 + \beta_1 x_j$

DESeq2/EdgeR - estimating the dispersion

Estimating the dispersion parameters

1. Estimate per-gene dispersion parameters with MLE
2. Estimate *mean-variance relationship* with local linear regression
3. *Shrink* per-gene estimates towards regression line (excluding large outliers)



DESeq2/EdgeR - testing

Hypothesis Testing for each gene

- Null hypothesis for gene g: $\beta_1 = 0$ (no effect of condition)
- Fit Negative Binomial GLM:

$$y_{gj} \sim NB(\mu_{gj}, \alpha_g), \quad \log(\mu_{gj}) = \log(s_j) + \beta_0 + \beta_1 x_j$$

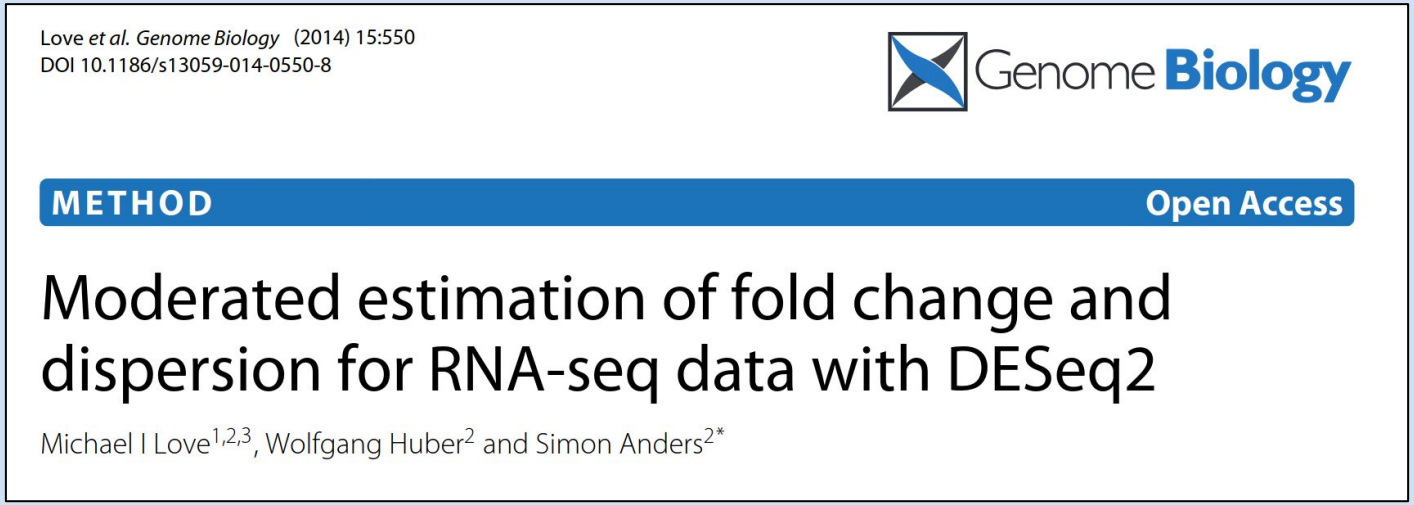
- **DESeq2 :** Compute Wald statistic:

$$W_g = \frac{\hat{\beta}_1}{\mathrm{SE}(\hat{\beta}_1)}$$

  - Approximate p-value: $p_g = 2\Pr(|Z| > |W_g|),\ Z \sim N(0,1)$

- **EdgeR :** Quasi-likelihood F-test (detail omitted)
- Adjust p-values for multiple testing and reject null if $p < 0.05$

For more information



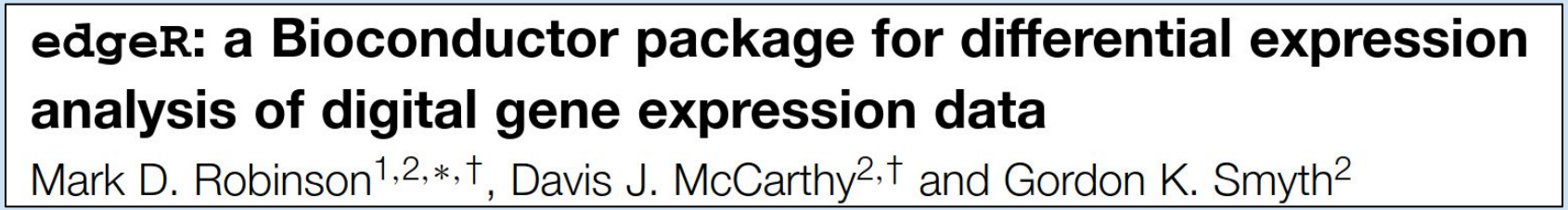
Love *et al. Genome Biology* (2014) 15:550
DOI 10.1186/s13059-014-0550-8

Genome Biology

METHOD Open Access

Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2

Michael I Love[1,2,3], Wolfgang Huber[2] and Simon Anders[2*]



edgeR: a Bioconductor package for differential expression analysis of digital gene expression data

Mark D. Robinson[1,2,*,†], Davis J. McCarthy[2,†] and Gordon K. Smyth[2]

Some problems with these packages

- Rely on the assumption that the raw counts are negative binomial
- Do not allow for complex experimental designs (longitudinal data, covariate correction)
- A linear model framework could fix these problems

Packages using linear model frameworks

- **Voom-limma**
- **dearseq**

Differential Gene Expression Analysis

Voom-Limma package

Voom-limma

Normalization and transformation

- Let x_j = experimental condition for sample j (e.g. 1 if treated, 0 if control)
- counts for gene g, sample j: y_{gj}
- Normalise counts for library size (e.g. TMM) to obtain scaling factors s_j
- Transform counts to log2-counts-per-million (logCPM):

$$\text{logCPM}_{gj} = \log_2\left(\frac{y_{gj}}{s_j} \cdot 10^6 + 0.5\right)$$

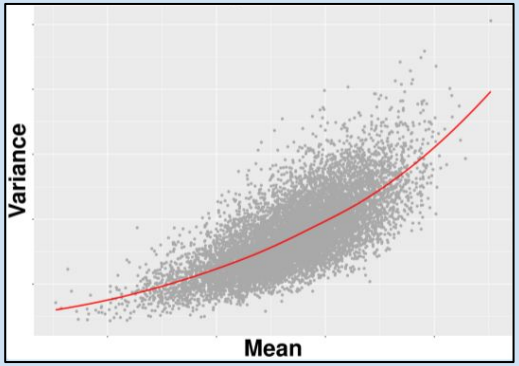
Voom-Limma

Estimating precision weights

- Compute the mean and variance of logCPM for each gene
- Estimate smooth mean–variance trend across genes
- Assign precision weights to each observation:

$$w_{gj} = \frac{1}{\hat{\text{Var}}(\text{logCPM}_{gj})}$$

- The higher the variance, the less the weight (i.e. noisy genes have less influence)



Voom-limma

Linear modeling and empirical Bayes

- Fit weighted linear model for each gene:

$$\text{logCPM}_{gj} \sim \beta_0 + \beta_1 x_j$$

- Use weights w_{gj} in weighted least squares
- *Shrink* estimates towards global mean

Voom-limma

Hypothesis testing

- Null hypothesis for gene g: $\beta_1 = 0$ (no effect of condition)
- Compute moderated t-statistics for $\hat{\beta}_1$
- Obtain p-values and adjust for multiple testing

Differential Gene Expression Analysis

dearseq Package

Dearseq package

Framework

- Let x_i denote covariates for sample i (e.g. condition, batch)
- Let y_{gi} be the (normalized) expression for gene g in sample i
- Working linear model for each gene:

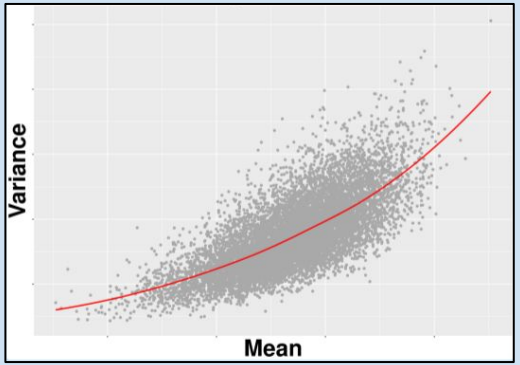
$$y_{gi} = \alpha_{g0} + X_i \alpha_g + \Phi_i \beta_g + \varepsilon_{gi},$$

where ε_{gi} has mean 0 and variance σ^2_{gi}; Φ_i contains the variable(s) of interest and β_g is the target parameter (gene is DE if $\beta_g \neq 0$).

Dearseq package

Estimating the mean–variance relationship

- Compute normalised expression
- Estimate mean-variance relationship across all genes and use as precision weights



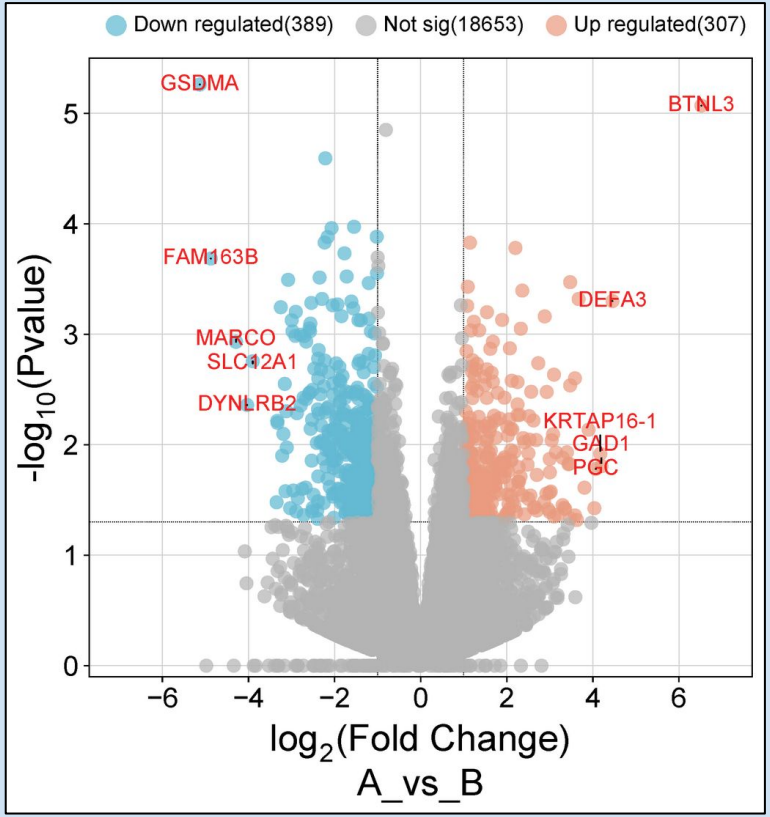
Dearseq package

Variance-component score testing

- Null hypothesis for gene g: $\beta_g = 0$
- Test with *score test*
- If n large enough : asymptotic test
- If n is small : permutation test
- Obtain p-values and adjust for multiple testing

Process	DESeq2	EdgeR	voom-limma	dearseq
Normalisation	Size factor (median of ratios)	TMM	TMM	User choice
Dispersion Estimation	**Negative binomial**, empirical Bayes shrinkage	**Negative binomial**, empirical Bayes shrinkage	Model mean-variance relationship with local linear smoothing	Model mean-variance relationship with local linear smoothing
Statistical Test	Wald Test or Likelihood ratio test	Likelihood Ratio Test or Quasi-likelihood F-test	Moderated t-test	Score test
Experimental Design	Simple	Can incorporate more complex experimental conditions	Can incorporate more complex experimental conditions + covariate correction	Can incorporate more complex experimental conditions + covariate correction + longitudinal data

Volcano plot



Conclusion on differential gene expression analysis

- Lots of different tools with the same goal
- Each tool has its own assumptions
- **Advice : try multiple methods and compare the results**

Recap

Recap

- Omics data allow us to study underlying molecular processes behind exposures
- **Normalisation**: pre-processing of the data to make fair comparisons
  - TMM-CPM normalisation a good default
  - Pay close attention to batch effects
- **Exploratory analysis**: PCA and hierarchical clustering to visualise data
  - These should be used to ensure good normalisation!
- **Differential gene analysis**: statistical frameworks for identifying genes whose expression changes under experimental conditions

Practical work

We will now perform an analysis to investigate differentially expressed gene signatures of active tuberculosis infection

Complete part 4 from the R markdown file
PHDS_omics_GMA_2025_questions.Rmd