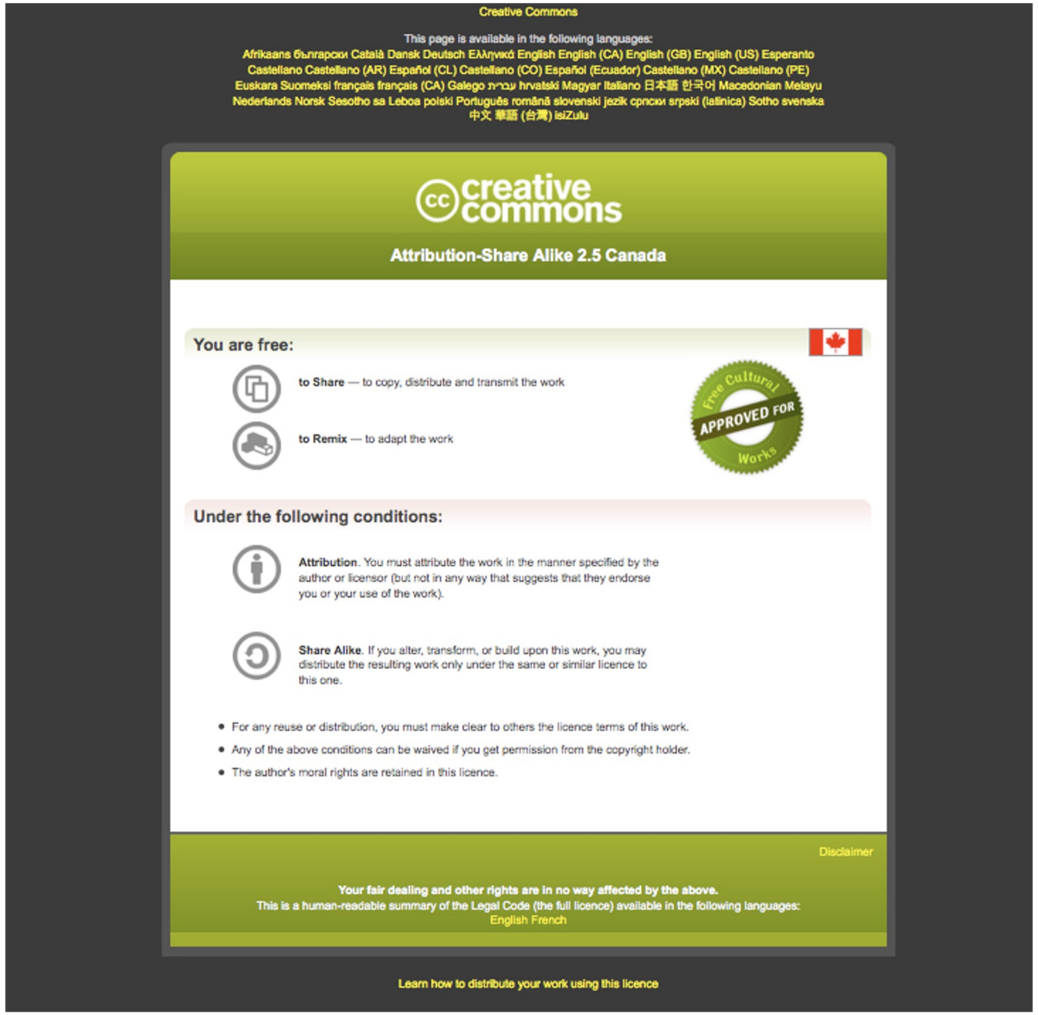
bioinformatics.ca

Canadian Bioinformatics Workshops

www.bioinformatics.ca

bioinformaticsdotca.github.io



Creative Commons

This page is available in the following languages:
Afrikaans български Català Dansk Deutsch Ελληνικά English English (CA) English (GB) English (US) Esperanto
Castellano Castellano (AR) Español (CL) Castellano (CO) Español (Ecuador) Castellano (MX) Castellano (PE)
Euskara Suomeksi français français (CA) Galego עברית hrvatski Magyar Italiano 日本語 한국어 Macedonian Melayu
Nederlands Norsk Sesotho sa Leboa polski Português română slovenski jezik српски srpski (latinica) Sotho svenska
中文 華語 (台灣) isiZulu

creative commons

Attribution-Share Alike 2.5 Canada

You are free:

- **to Share** — to copy, distribute and transmit the work
- **to Remix** — to adapt the work

Under the following conditions:

Attribution. You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

Share Alike. If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar licence to this one.

- For any reuse or distribution, you must make clear to others the licence terms of this work.
- Any of the above conditions can be waived if you get permission from the copyright holder.
- The author's moral rights are retained in this licence.

Disclaimer

Your fair dealing and other rights are in no way affected by the above.
This is a human-readable summary of the Legal Code (the full licence) available in the following languages:
English French

Learn how to distribute your work using this licence

Dimensionality reduction

Chaitra Sarathy, PhD

Analysis Using R

June, 28-29, 2023

Overview

- What is dimensionality reduction?
- Why reduce?
- A few flavors of dimensionality reduction:
  - PCA
  - tSNE
  - UMAP

Dimensionality reduction

- What is dimension?
- Define terms using data

Dimensionality reduction

Simple example data

	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	12	15	10	5	4	2
Gene 2	8	13	9	3	3	1

Dimensionality reduction

Simple example data

	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	12	15	10	5	4	2
Gene 2	8	13	9	3	3	1

2 genes

Dimensionality reduction

Simple example data

6 mice

	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	12	15	10	5	4	2
Gene 2	8	13	9	3	3	1

2 genes

Dimensionality reduction

Simple example data

6 mice → samples

	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	12	15	10	5	4	2
Gene 2	8	13	9	3	3	1

2 genes → variables

Dimensionality reduction

Simple example data

6 mice → samples

	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	12	15	10	5	4	2
Gene 2	8	13	9	3	3	1

2 genes → variables

Other biological data

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6
Protein 1	20	15	18	50	45	43
Protein 2	15	13	14	37	35	38

Dimensionality reduction

Simple example data

6 mice → samples

	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	12	15	10	5	4	2
Gene 2	8	13	9	3	3	1

2 genes → variables

Other biological data

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6
Protein 1	20	15	18	50	45	43
Protein 2	15	13	14	37	35	38

Non-omic data

	Student 1	Student 2	Student 3	Student 4	Student 5	Student 6
Math	90	85	80	65	60	68
Science	80	73	66	64	63	59

Dimensionality reduction

Simple example data

6 mice → samples

	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	12	15	10	5	4	2
Gene 2	8	13	9	3	3	1

2 genes → variables

How to visualise differences in samples?

Dimensionality reduction

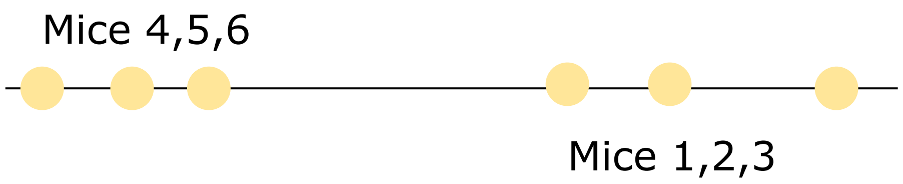
Simple example data

mice → samples

	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	12	15	10	5	4	2

genes → variables

Plot on number line



Dimensionality reduction

Simple example data

Plot on two dimensional xy axis

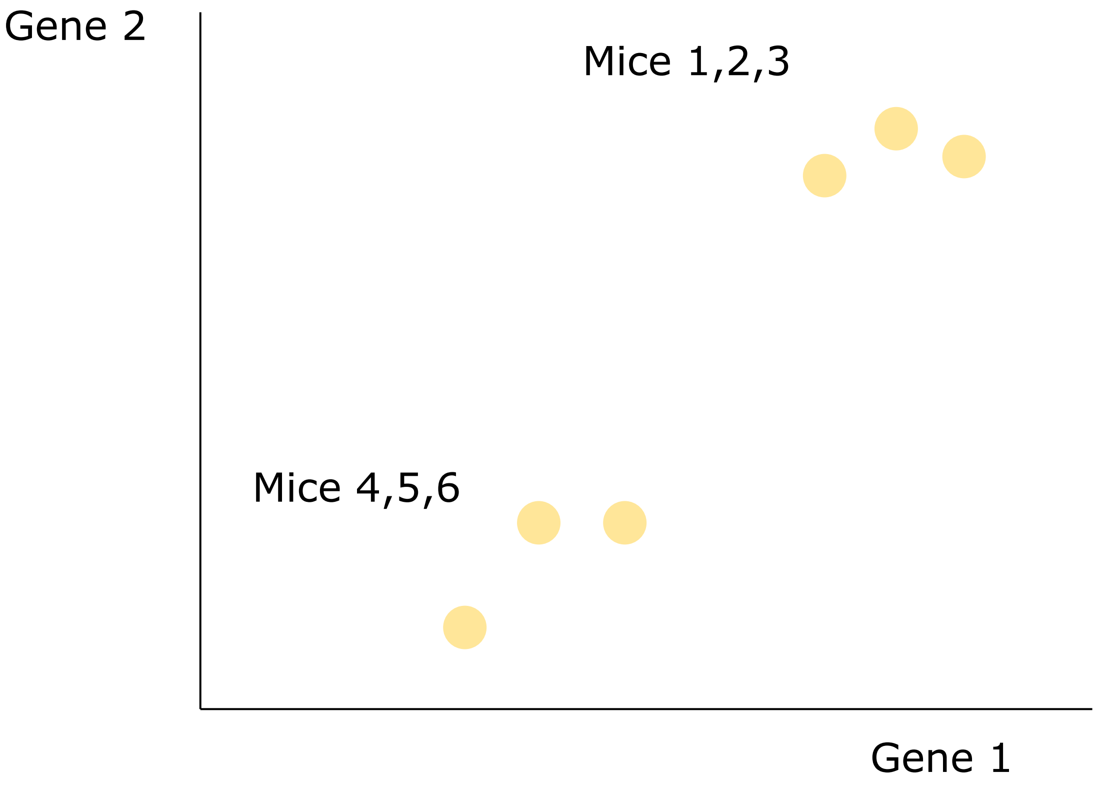
	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	12	15	10	5	4	2
Gene 2	8	13	9	3	3	1

Dimensionality reduction

Simple example data

	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	12	15	10	5	4	2
Gene 2	8	13	9	3	3	1

Plot on two dimensional xy axis

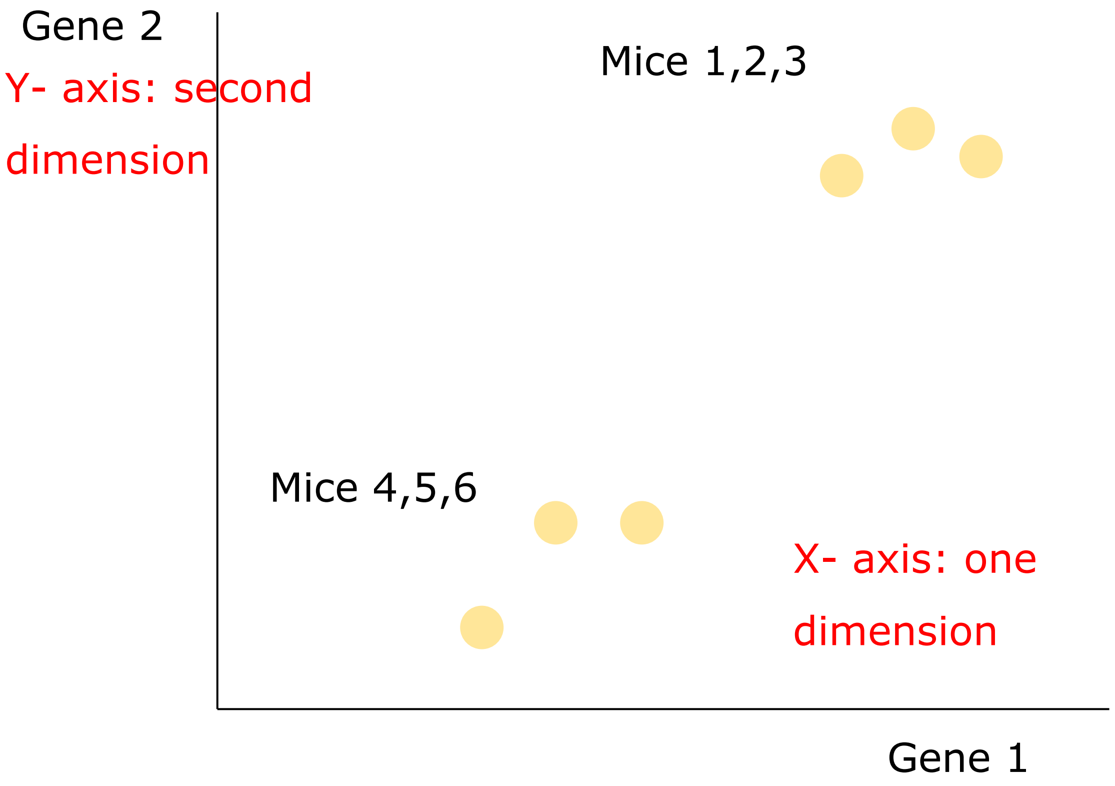


Dimensionality reduction

Simple example data

	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	12	15	10	5	4	2
Gene 2	8	13	9	3	3	1

Plot on two dimensional xy axis

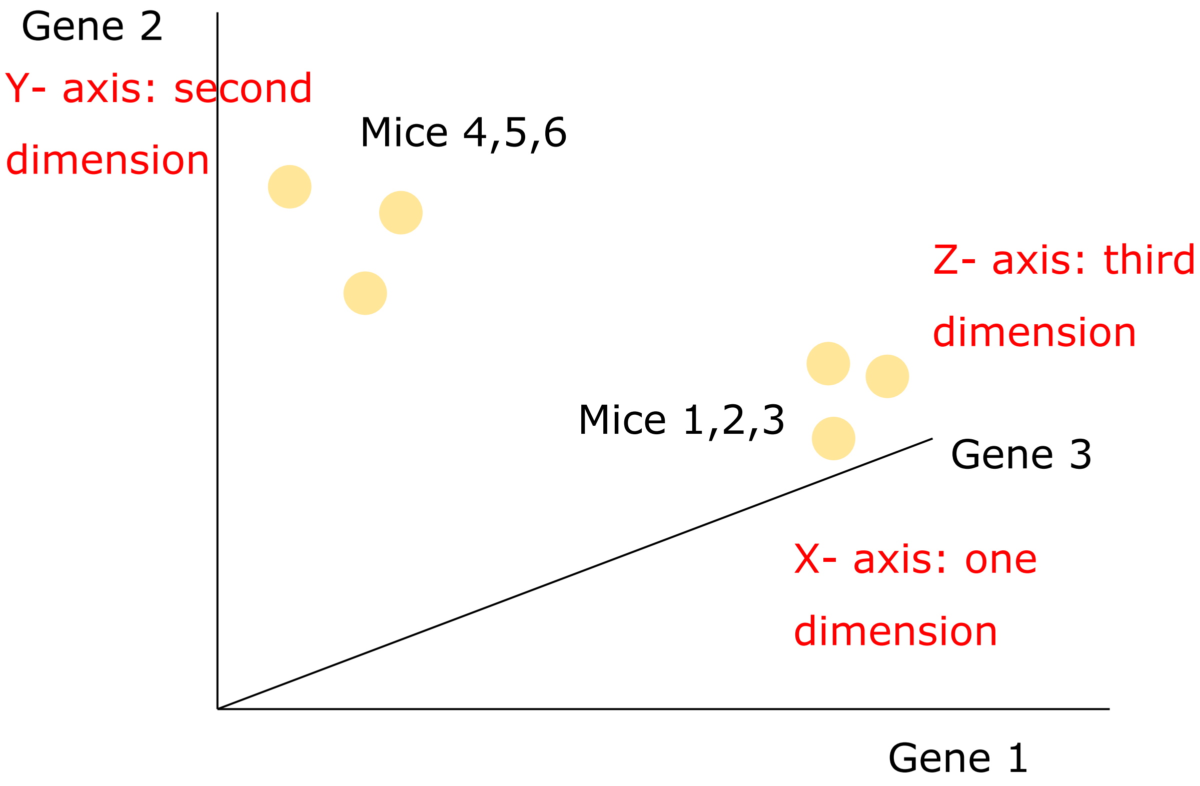


Dimensionality reduction

Simple example data

	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	12	15	10	5	4	2
Gene 2	8	13	9	3	3	1
Gene 3	5	5	6	15	18	22

Plot on three dimensional xyz axis



Dimensionality reduction

Simple example data

mice → samples

	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	12	15	10	5	4	2
Gene 2	8	13	9	3	3	1
Gene 3	5	5	6	15	18	22
Gene 4	22	25	30	30	33	23
. . .						
Gene 10000						

genes → variables

→ **Plot on four dimensions**

→ **Plot on 10000 dimensions**

Dimensionality reduction to the rescue!

Dimensionality reduction

Simple example data

mice → samples

	Mouse 1	Mouse 2	Mouse 3	Mouse 4	Mouse 5	Mouse 6
Gene 1	12	15	10	5	4	2
Gene 2	8	13	9	3	3	1
Gene 3	5	5	6	15	18	22
Gene 4	22	25	30	30	33	23
. . .						
Gene 10000						

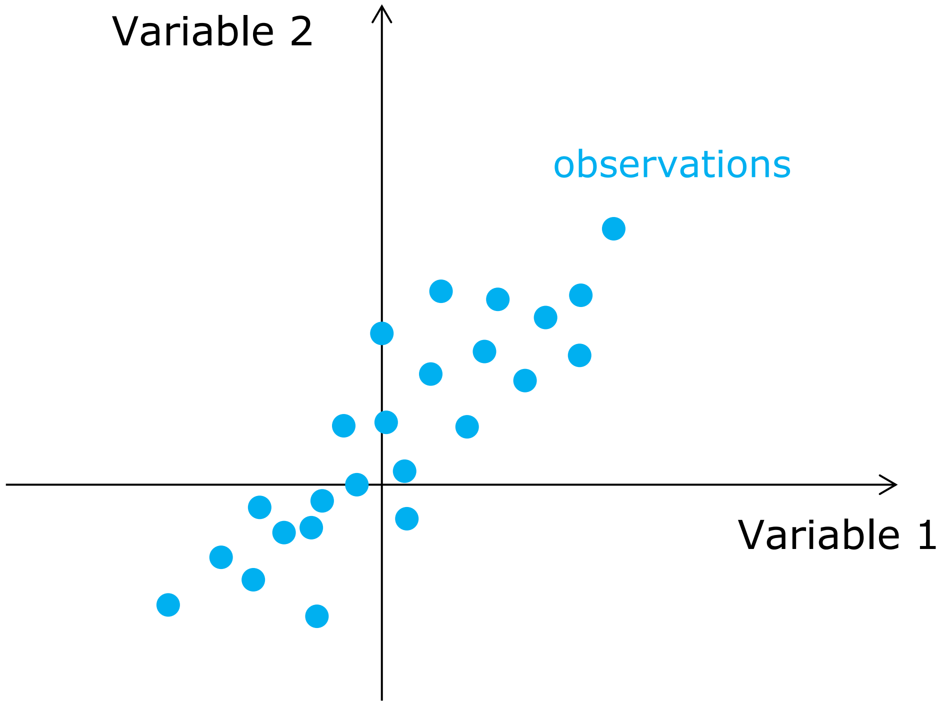
genes → variables

Dimensionality reduction: transform data to a few new variables which explain most of the differences in observations

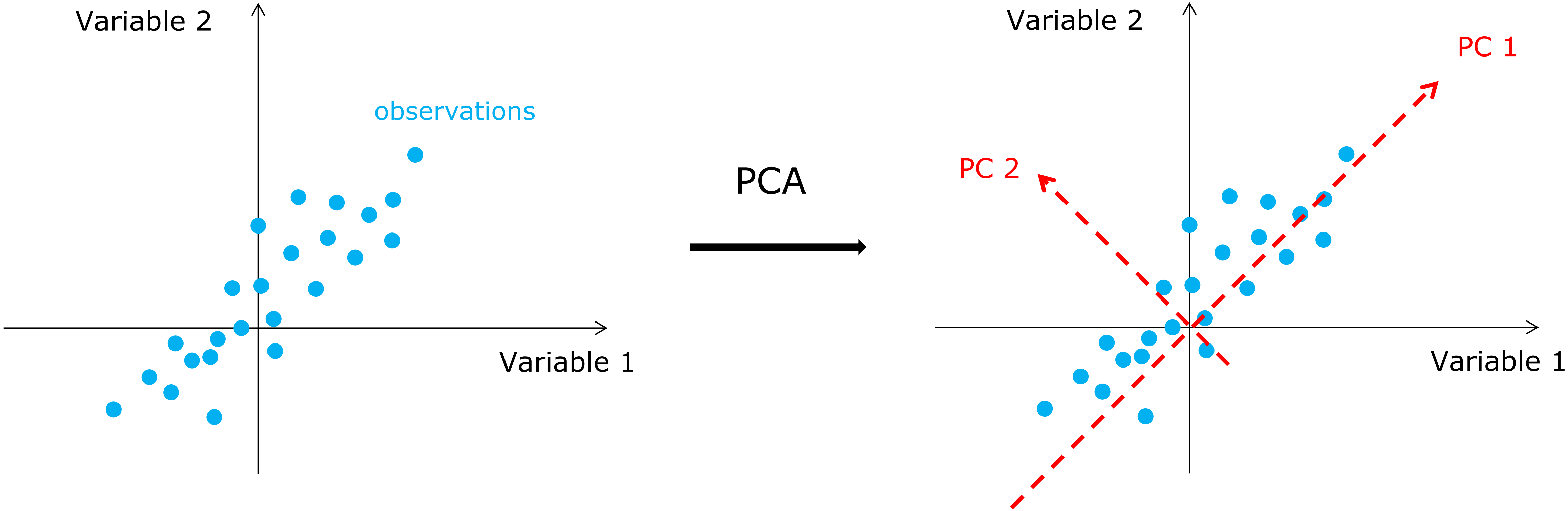
Principal Component Analysis (PCA)

- Most widely used method for dimension reduction
- One step in analysis pipeline (Refer flowchart in Module 1)

Principal Component Analysis (PCA)



Principal Component Analysis (PCA)

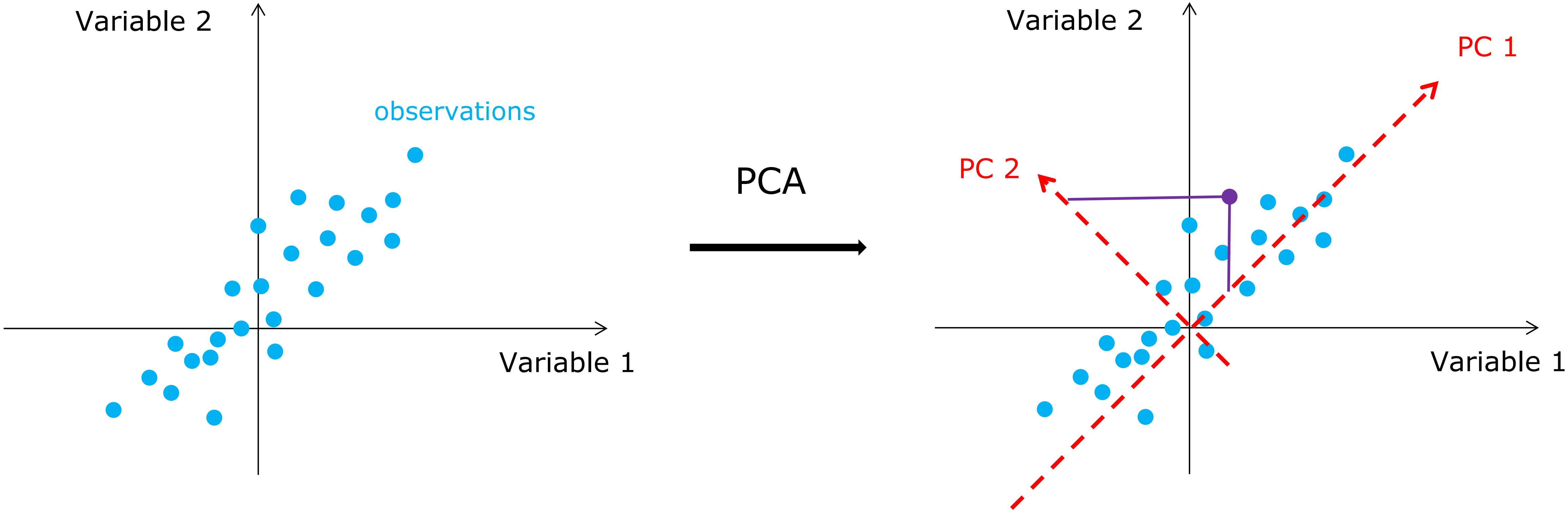


Rotate data into newer axes or dimensions – Principal Components (PC1, PC2, etc)

PC1 - First principal component – axis with maximum variance

PC2 – Second principal component – second highest variance

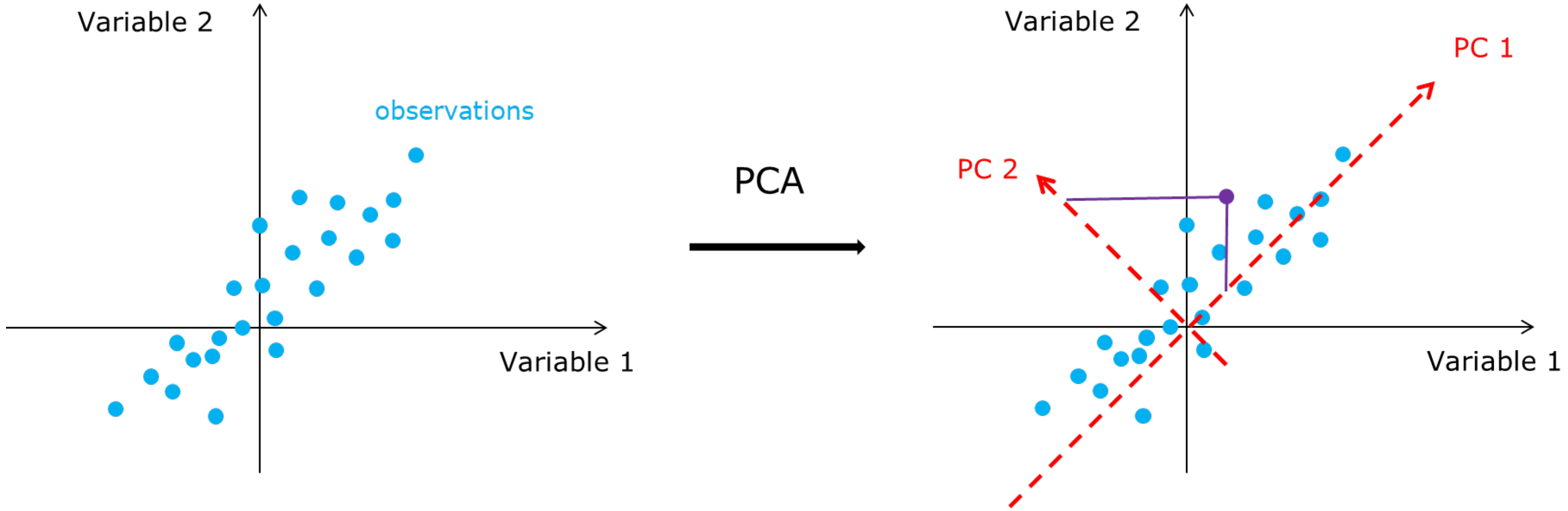
Principal Component Analysis (PCA)



Rotate data → Map data onto new axes → Projections

Amount by which data points are rotated → loading values

Principal Component Analysis (PCA)



Take away:

Rotate data into newer axes or dimensions – **Principal Components (PC1, PC2, etc)**

PC1 - First principal component – axis with maximum **variance**

PC2 – Second principal component – second highest variance

Rotate data → Map data onto new axes → **Projections**

Amount by which data points are rotated → **loading values**

Principal Component Analysis (PCA)

Applications

Applicable to both omic and non-omic datasets

Shows where the dominant structure in your data is

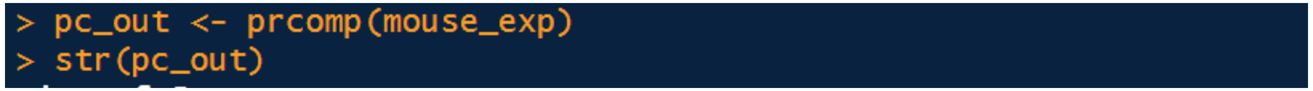
Useful for identifying batches, unmeasured variable effect, etc

Machine learning: Reducing feature set for accurate modelling

A useful PCA paper: https://www.cs.cmu.edu/~elaw/papers/pca.pdf

PCA: base r function “prcomp”

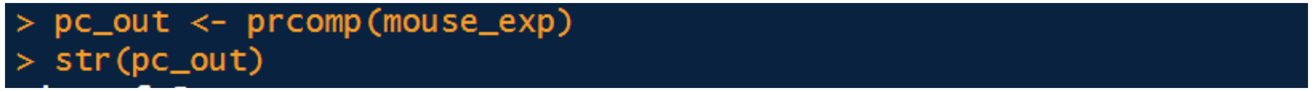
Perform PCA on your mouse gene expression data



```
> pc_out <- prcomp(mouse_exp)
> str(pc_out)
```

PCA: Results of “prcomp”

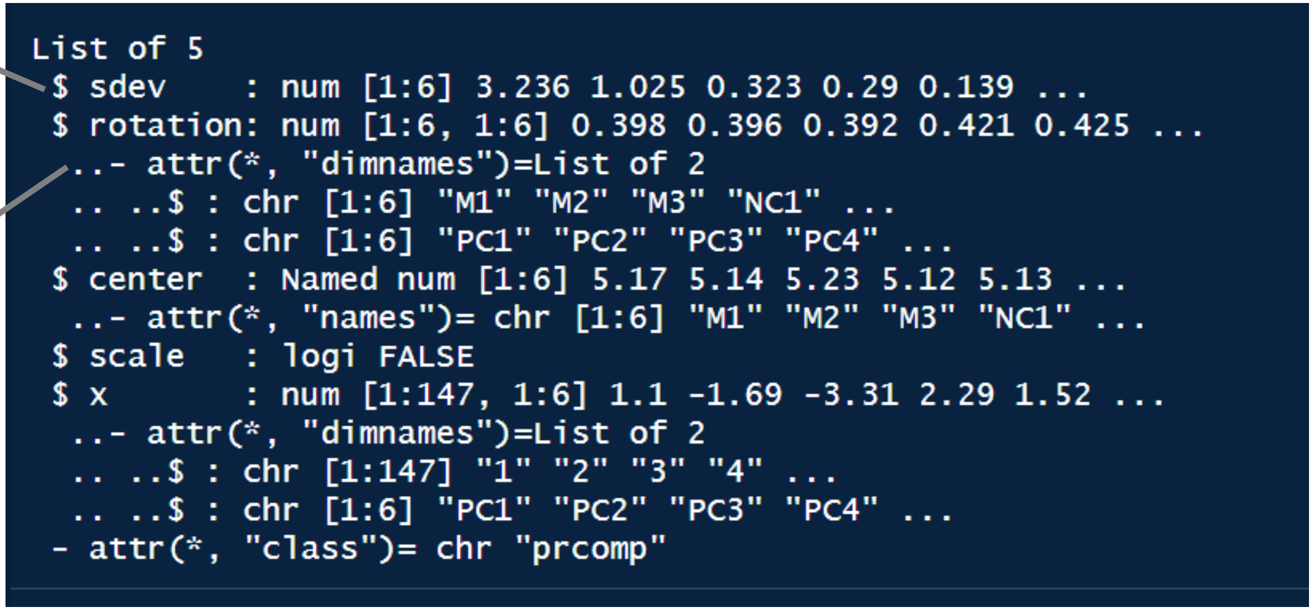
str(pc_out)



```
> pc_out <- prcomp(mouse_exp)
> str(pc_out)
```

Standard deviation

Loading values



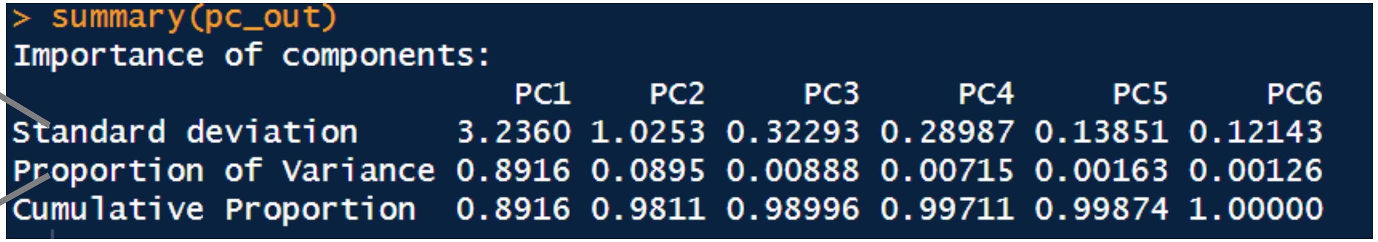
```
List of 5
 $ sdev     : num [1:6] 3.236 1.025 0.323 0.29 0.139 ...
 $ rotation: num [1:6, 1:6] 0.398 0.396 0.392 0.421 0.425 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : chr [1:6] "M1" "M2" "M3" "NC1" ...
  .. ..$ : chr [1:6] "PC1" "PC2" "PC3" "PC4" ...
 $ center   : Named num [1:6] 5.17 5.14 5.23 5.12 5.13 ...
  ..- attr(*, "names")= chr [1:6] "M1" "M2" "M3" "NC1" ...
 $ scale    : logi FALSE
 $ x        : num [1:147, 1:6] 1.1 -1.69 -3.31 2.29 1.52 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : chr [1:147] "1" "2" "3" "4" ...
  .. ..$ : chr [1:6] "PC1" "PC2" "PC3" "PC4" ...
 - attr(*, "class")= chr "prcomp"
```

PCA: Results of “prcomp”

```
summary(pc_out)
```

Standard deviation

Variance explained

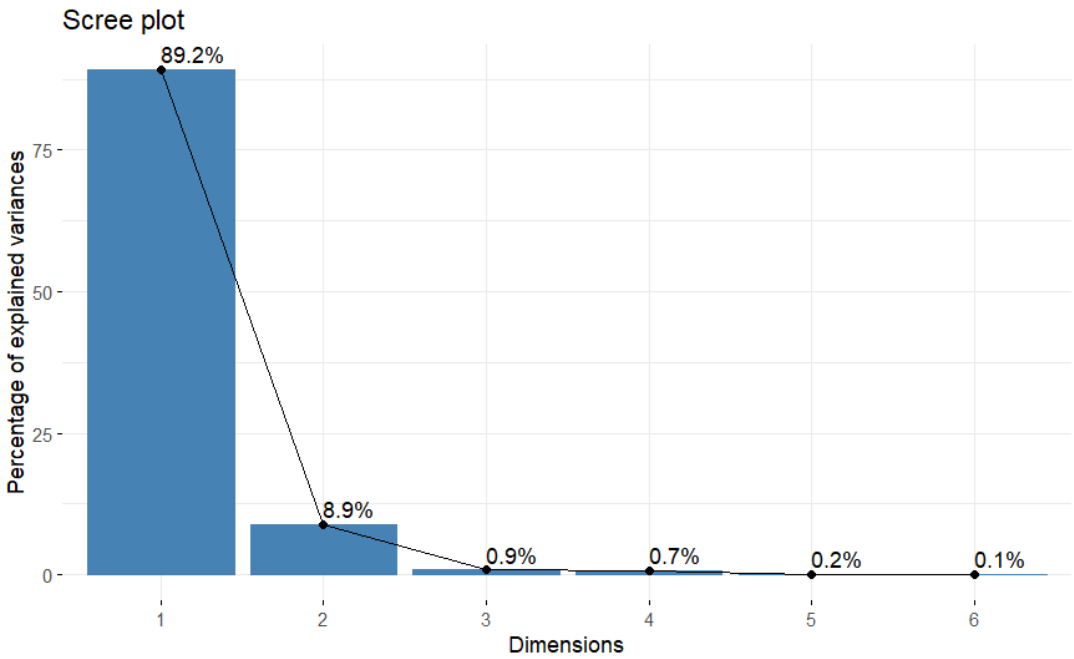


```
> summary(pc_out)
Importance of components:
                          PC1    PC2     PC3     PC4     PC5     PC6
Standard deviation     3.2360 1.0253 0.32293 0.28987 0.13851 0.12143
Proportion of Variance 0.8916 0.0895 0.00888 0.00715 0.00163 0.00126
Cumulative Proportion  0.8916 0.9811 0.98996 0.99711 0.99874 1.00000
```

- First principal component explains 89.16% of the total variance
- Second principal component explains 8.9% of the variance
- Amount of variance explained reduces further down with each component

PCA: Visualising results of “prcomp”

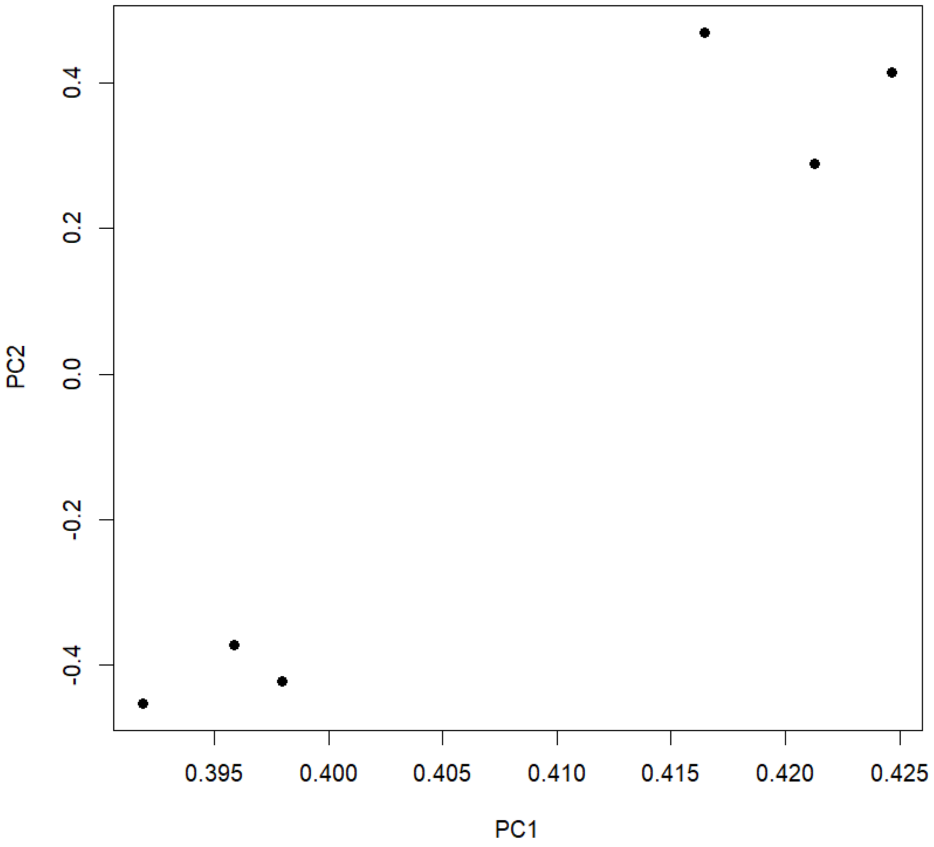
1. Scree plot



Wikipedia: Scree is a collection of broken rock fragments at the base of a cliff or other steep rocky mass that has accumulated through periodic rockfall

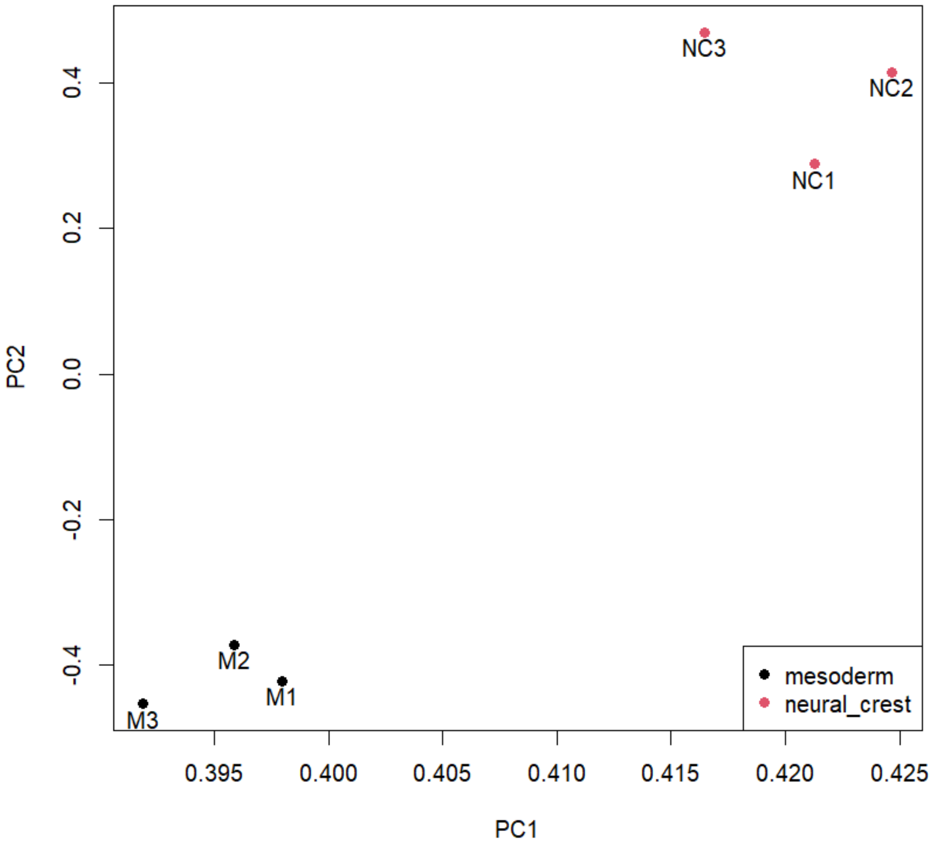
PCA: Visualising results of “prcomp”

2.Scatter plot



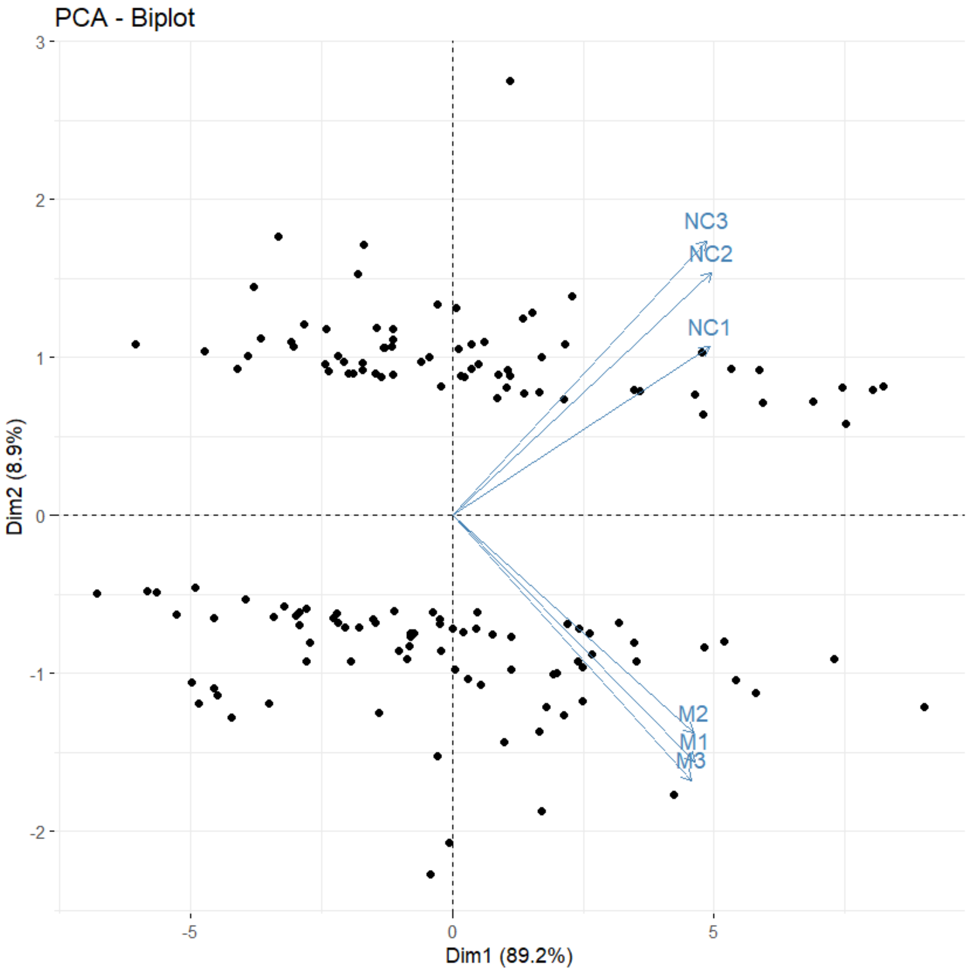
PCA: Visualising results of “prcomp”

2.Scatter plot



PCA: Visualising results of “prcomp”

3.Biplot



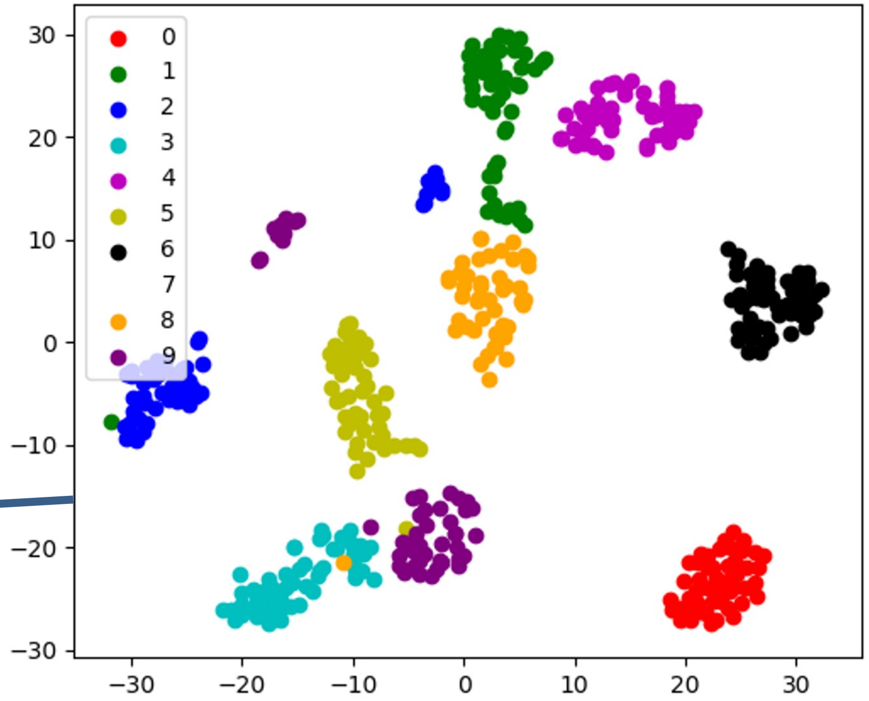
tSNE: R package “tsne”

- Stands for “t-Stochastic Neighbor Embedding”
- For data that cannot be separated by any straight line
- Finds few variables that represent many variables preserving neighborhood distances
- Great for visualizations (scRNA-seq)
- Stochastic = random (set seed to make reproducible)

- Difference from PCA
  - focus on local signal (neighborhood) vs global signal (explaining maximum variance)

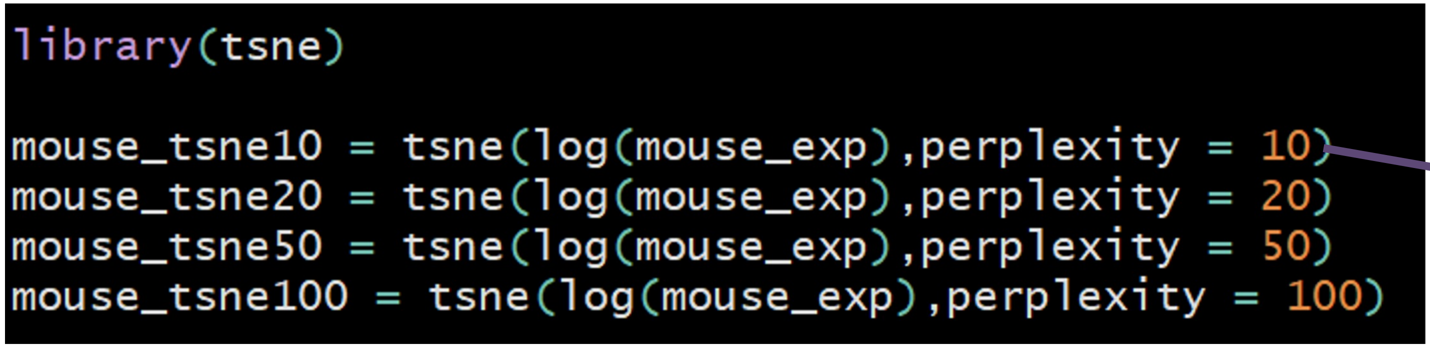
t-SNE paper: http://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf

tSNE: R package “tsne”



tSNE finds axes that maintain “neighborhoods”

tSNE: R package “tsne”

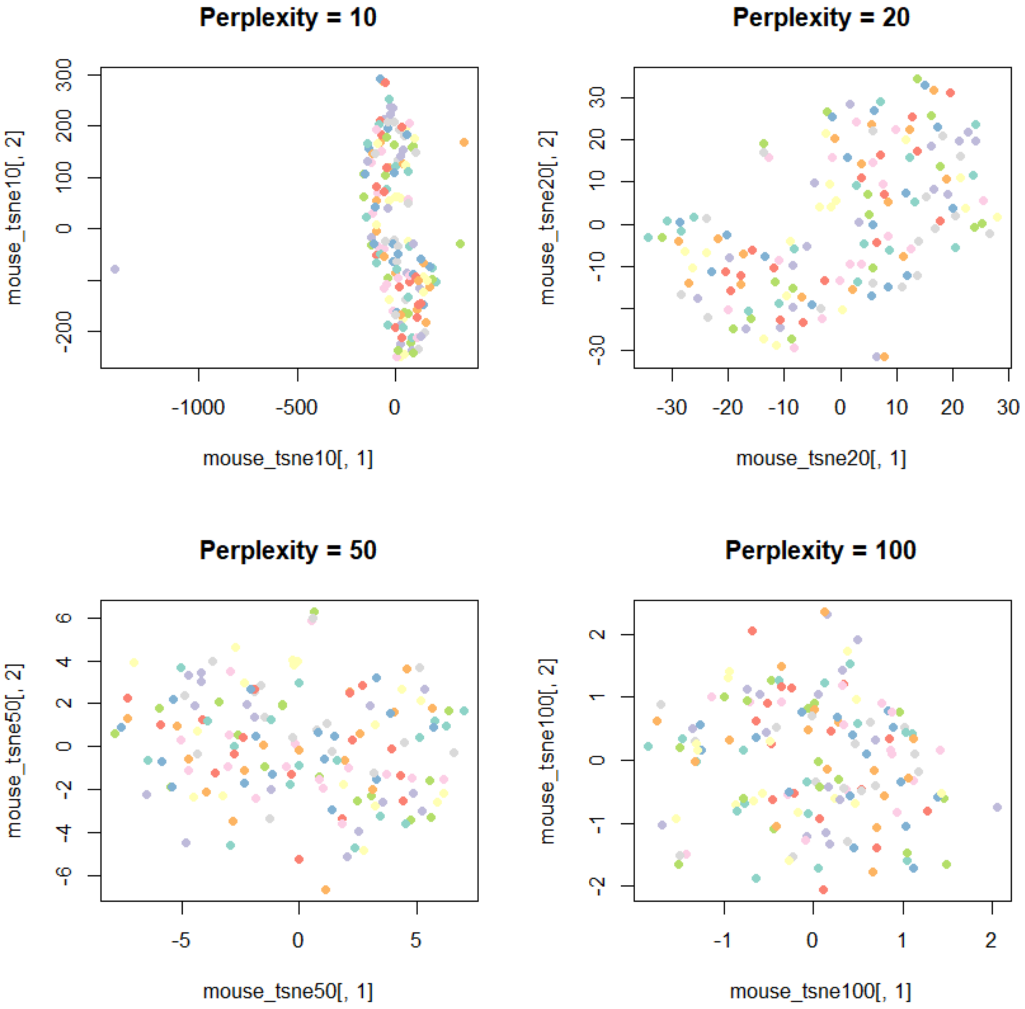


```
library(tsne)

mouse_tsne10 = tsne(log(mouse_exp),perplexity = 10)
mouse_tsne20 = tsne(log(mouse_exp),perplexity = 20)
mouse_tsne50 = tsne(log(mouse_exp),perplexity = 50)
mouse_tsne100 = tsne(log(mouse_exp),perplexity = 100)
```

Perplexity parameter determines how to balance attention to neighborhood vs global structure (smaller=more focus on the neighborhood)

Plot your tsne’s



Note that higher perplexity leads to higher spread in your data

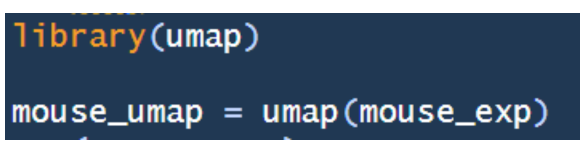
UMAP: R package “umap”

- Stands for “Uniform Manifold Approximation and Projection”
- Similar neighborhood approach as t-SNE

UMAP paper: https://arxiv.org/abs/1802.03426

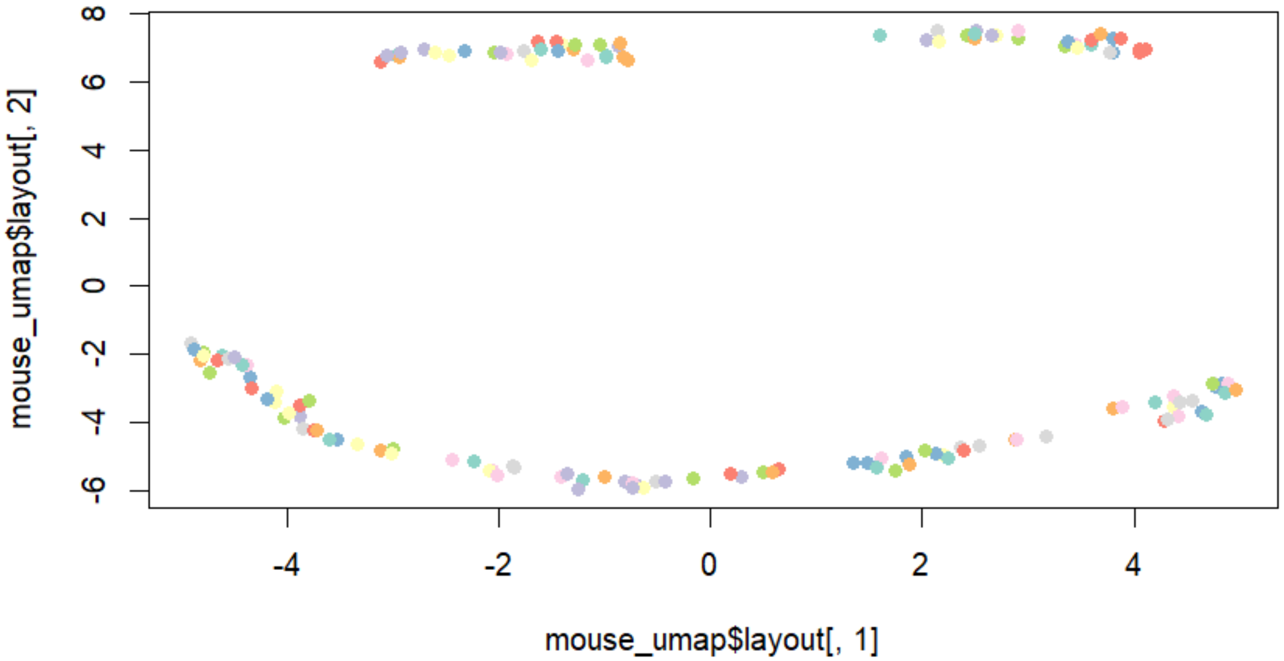
UMAP: R package “umap”

- Run umap



```
library(umap)

mouse_umap = umap(mouse_exp)
```



PCA vs tSNE vs UMAP

PCA	tSNE	UMAP
Linear combination	Non-linear	Non-linear
Lower dimensions are called Principal components	Embeddings	TBA
Data is projected onto lower-dimensional space		
Visualization, Covariates for statistical modeling	Visualization	Visualization
Concerned with preserving largest distances, to maximize variance of each PC.	Concerned with preserving nearest-neighbour distances •Tuned with “perplexity” parameter	TBA

Exercise

- Return to your crabs data
- Compute the principle components (PCs) for the numeric columns
- Plot these PCs and color them by species (“sp”) and sex
- Now compute 2 t-SNE components for these data and color by species and sex
- Finally compute 2 UMAP components for these data and color by species and sex
- Do any of these dimensionality reduction methods seem to segregate sex/species groups?

We are on a Coffee Break & Networking Session

Workshop Sponsors:

Canadian Centre for Computational Genomics

HPC4Health

OICR
Ontario Institute
for Cancer Research

GenomeCanada