STATISTICS AND BIG DATA '25-'26

PCA & CA in a nutshell

+ R Markdown bonus

Dr. Niccolò Salvini

Adjunct Professor @UCSC campus Rome,
Sr. Data Scientist

R markdown, what's that?

1 **R Markdown** intro

2 **R Markdown** examples

3 Markdown syntax

high level PCA concepts

PCA in R

4 minimal R code!

live coding session!

Section 1

R Markdown few concepts

UNIVERSITÀ
CATTOLICA
del Sacro Cuore

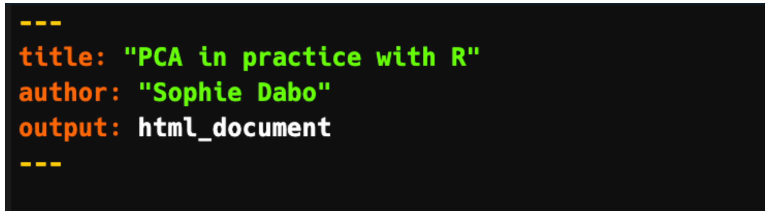
What is R Markdown? 🤌

- **R Markdown** is a tool that combines R code with narrative text to create dynam**ic documents, presentations, and reports.**
- It uses **Markdown syntax** for text formatting and allows the insertion of R code, which is executed when the document is compiled.

Why Use R Markdown? 🤌🤌

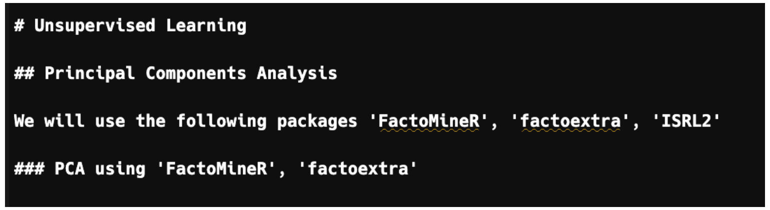
- **Reproducibility:** Code and results are integrated into the document, making it easier to share and reproduce the analysis.
- **Flexibility:** Supports various output formats like HTML, PDF, Word, and presentations.
- **Efficiency:** Automates the data analysis and reporting process.

3 Components of an R Markdown File 🦿



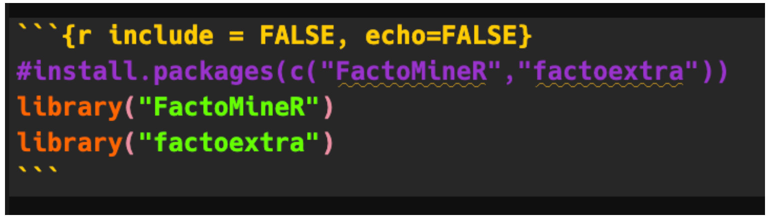
```
---
title: "PCA in practice with R"
author: "Sophie Dabo"
output: html_document
---
```

YAML Header: Specifies the title, author, date, and output format.



```
# Unsupervised Learning

## Principal Components Analysis

We will use the following packages 'FactoMineR', 'factoextra', 'ISRL2'

### PCA using 'FactoMineR', 'factoextra'
```

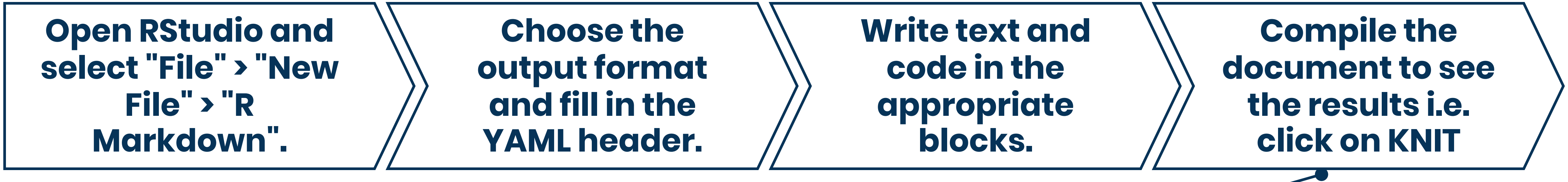
Narrative Text: Written in Markdown to describe the analysis.



~~~
```{r include = FALSE, echo=FALSE}
#install.packages(c("FactoMineR","factoextra"))
library("FactoMineR")
library("factoextra")
```
~~~

R Code Chunks: Called 'chunks', they contain executable R code.

Creating a Document 💡



Open RStudio and select "File" › "New File" › "R Markdown".

Choose the output format and fill in the YAML header.

Write text and code in the appropriate blocks.

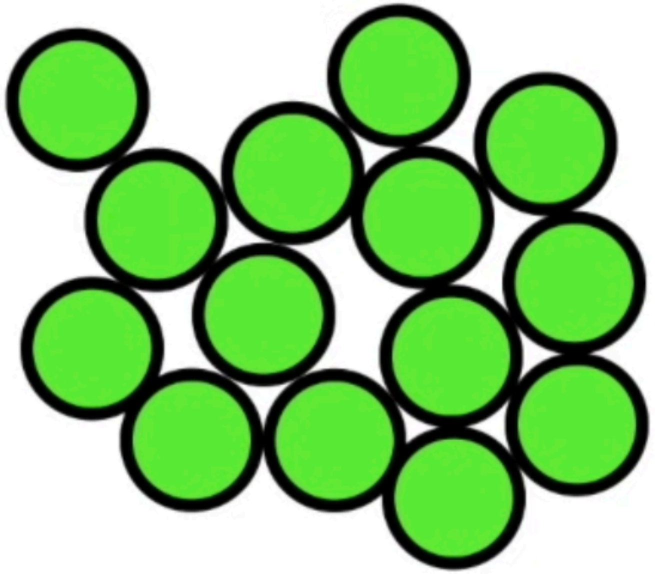
Compile the document to see the results i.e. click on KNIT

PSSS 🗣 I have also written a very brief doc on BB for the last step

Section 2

PCA, veeeery brief

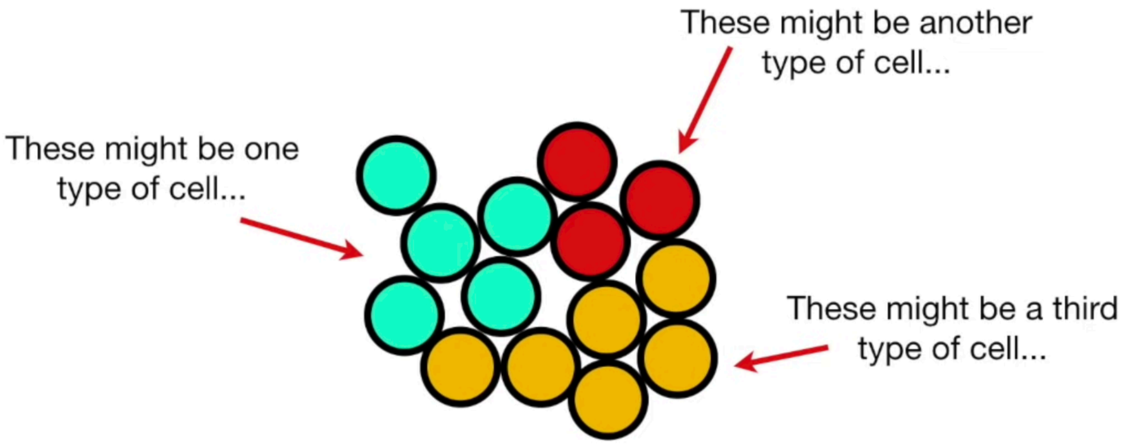
PCA!



Let's say we had some normal cells...

let's throw some data about cells! 🔮

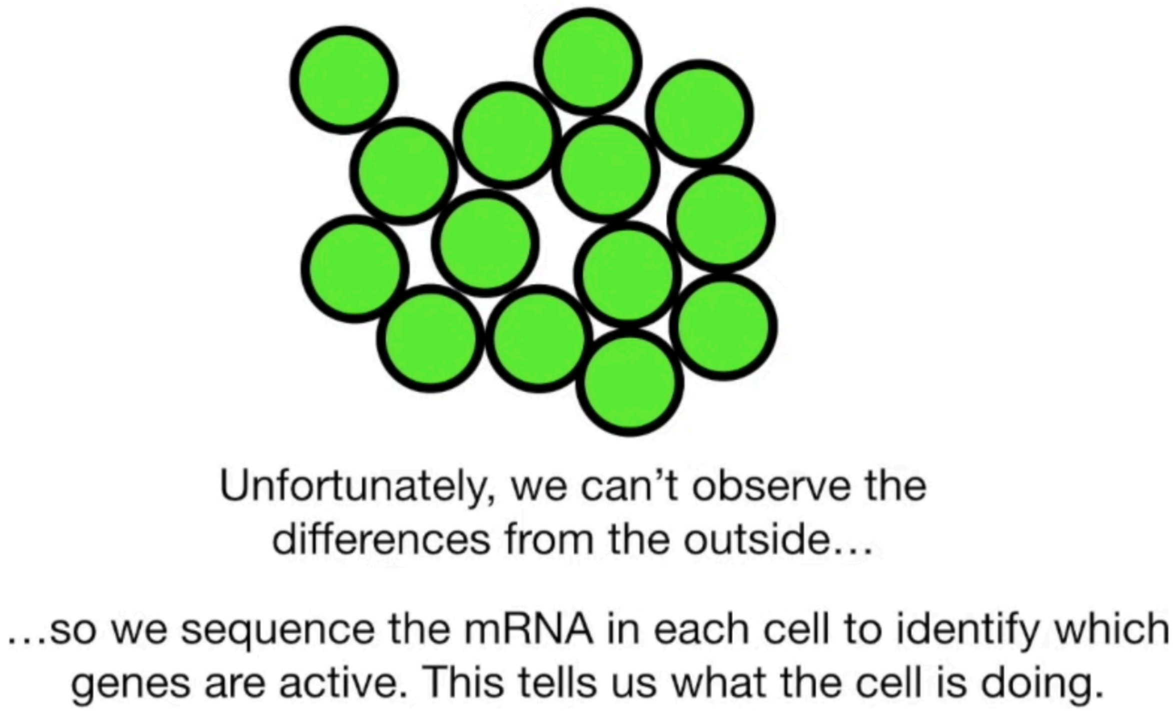
PCA!


These might be another type of cell...
These might be one type of cell...
These might be a third type of cell...

are there any differences?

mmmh they seems all clustered together

PCA!



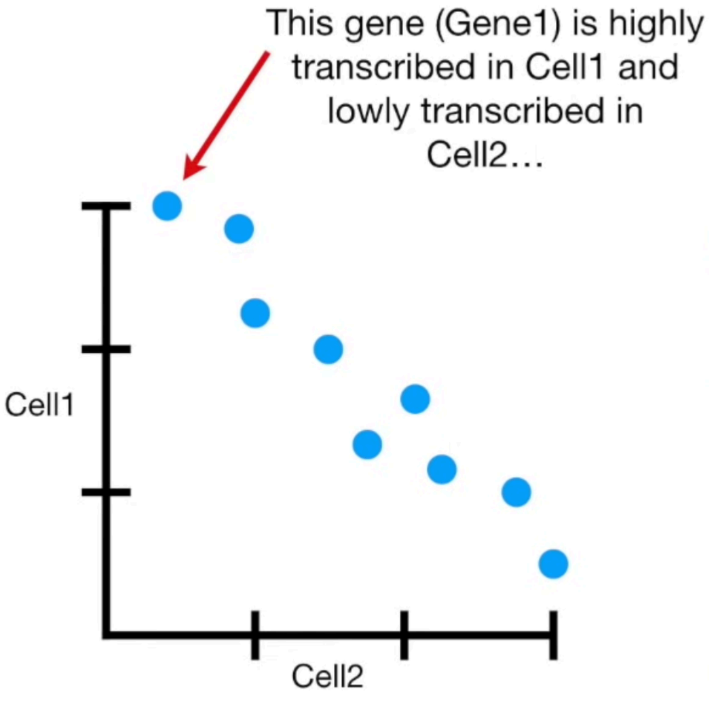
Unfortunately, we can’t observe the differences from the outside…

…so we sequence the mRNA in each cell to identify which genes are active. This tells us what the cell is doing.

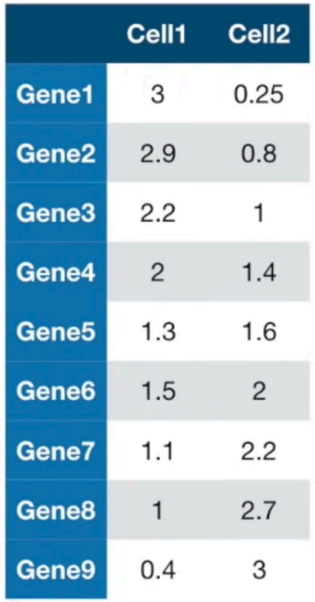
Why not observing MRNA sequence

pssss. this is the technolgy behind COVID19 vaccine

PCA!



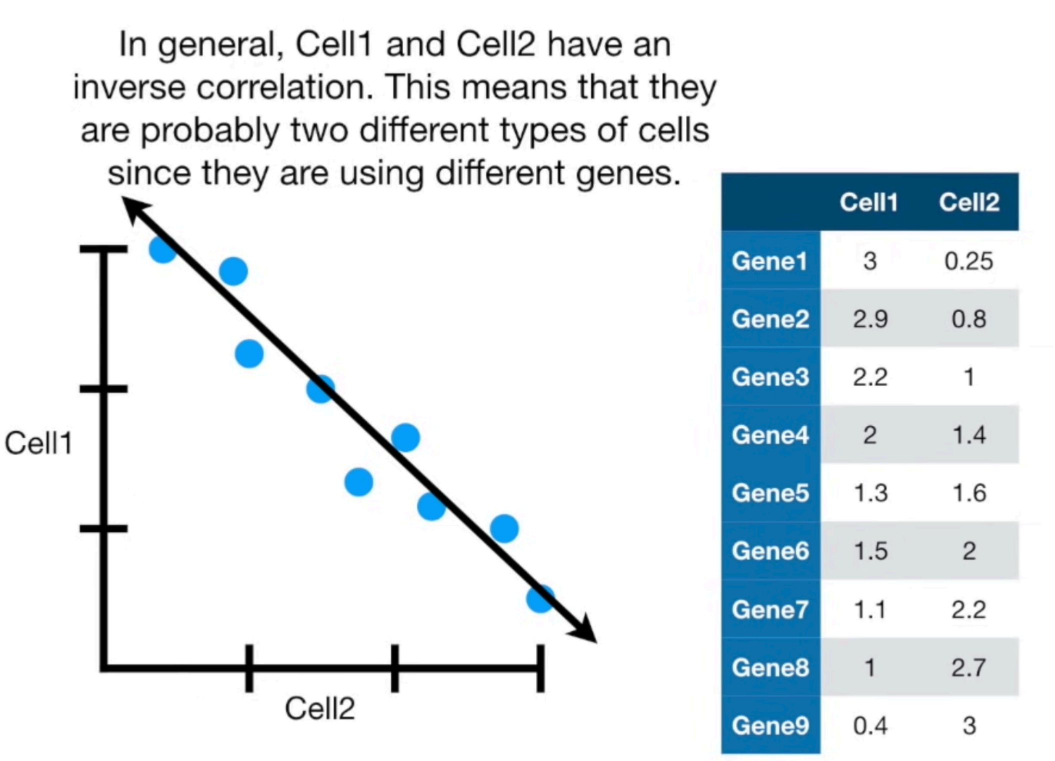
This gene (Gene1) is highly transcribed in Cell1 and lowly transcribed in Cell2…



	Cell1	Cell2
Gene1	3	0.25
Gene2	2.9	0.8
Gene3	2.2	1
Gene4	2	1.4
Gene5	1.3	1.6
Gene6	1.5	2
Gene7	1.1	2.2
Gene8	1	2.7
Gene9	0.4	3

plot cell1 vs cell2

PCA!



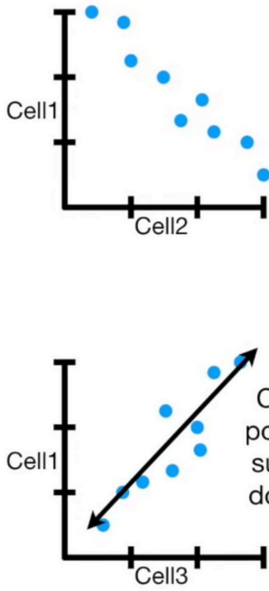
In general, Cell1 and Cell2 have an inverse correlation. This means that they are probably two different types of cells since they are using different genes.

	Cell1	Cell2
Gene1	3	0.25
Gene2	2.9	0.8
Gene3	2.2	1
Gene4	2	1.4
Gene5	1.3	1.6
Gene6	1.5	2
Gene7	1.1	2.2
Gene8	1	2.7
Gene9	0.4	3

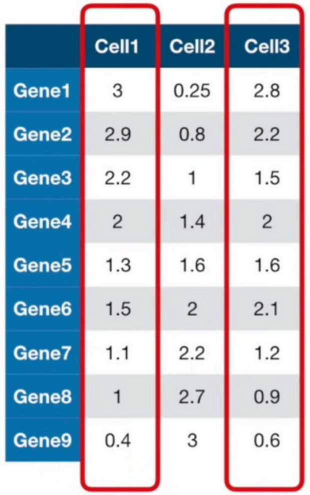
How cell1 is related to cell2?

neg related

PCA!



Cell1 and Cell3 are positively correlated, suggesting they are doing similar things.

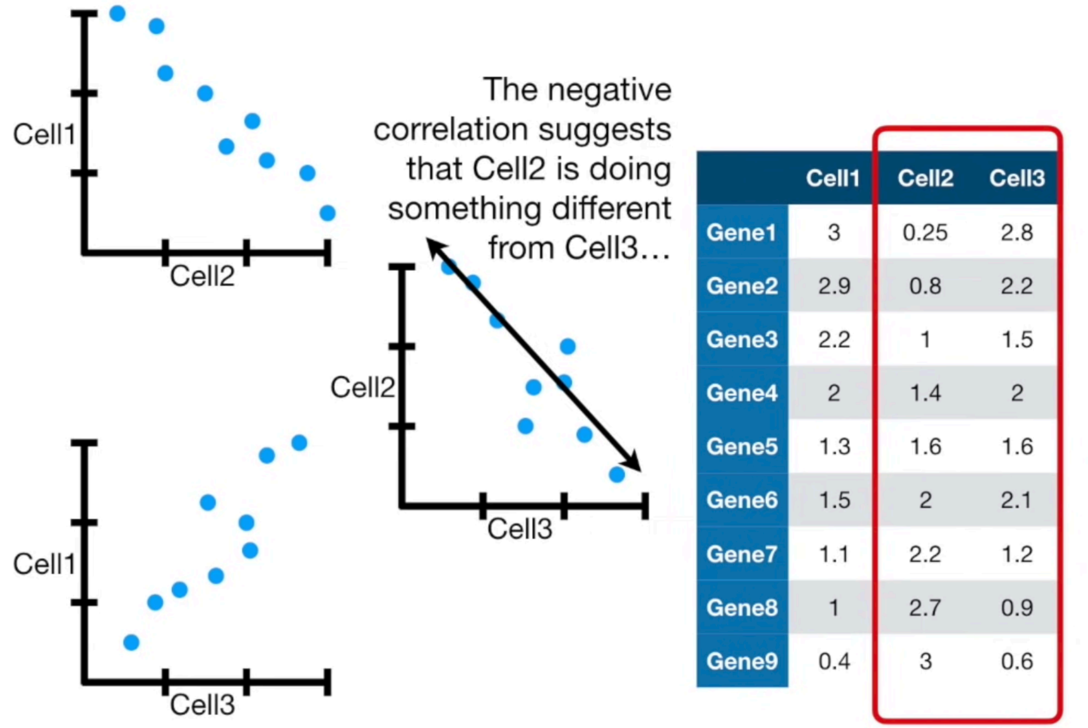


	Cell1	Cell2	Cell3
Gene1	3	0.25	2.8
Gene2	2.9	0.8	2.2
Gene3	2.2	1	1.5
Gene4	2	1.4	2
Gene5	1.3	1.6	1.6
Gene6	1.5	2	2.1
Gene7	1.1	2.2	1.2
Gene8	1	2.7	0.9
Gene9	0.4	3	0.6

cell1 vs cell3

pos related

PCA!



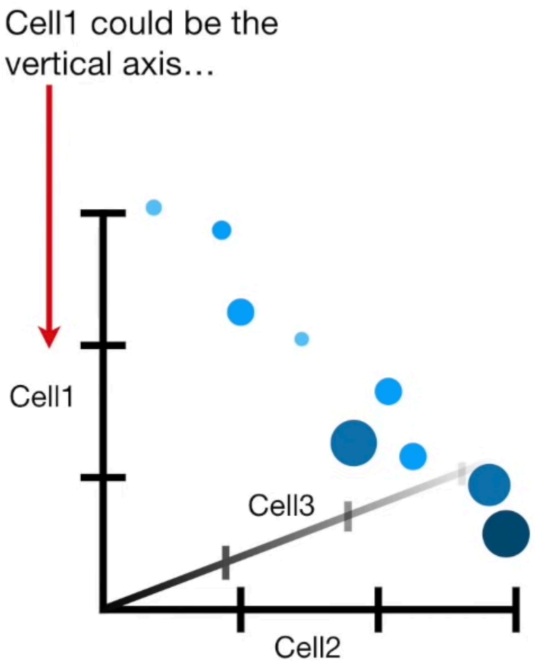
The negative correlation suggests that Cell2 is doing something different from Cell3…

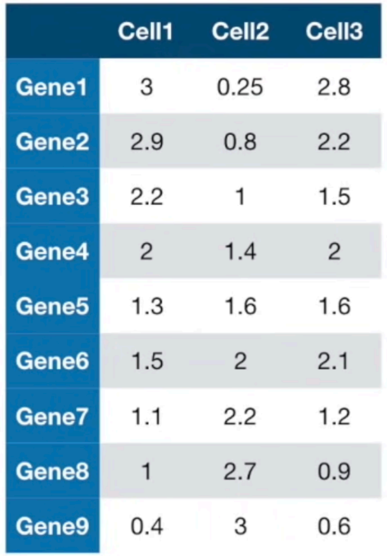
	Cell1	Cell2	Cell3
Gene1	3	0.25	2.8
Gene2	2.9	0.8	2.2
Gene3	2.2	1	1.5
Gene4	2	1.4	2
Gene5	1.3	1.6	1.6
Gene6	1.5	2	2.1
Gene7	1.1	2.2	1.2
Gene8	1	2.7	0.9
Gene9	0.4	3	0.6

cell2 vs cell3

neg related

PCA!

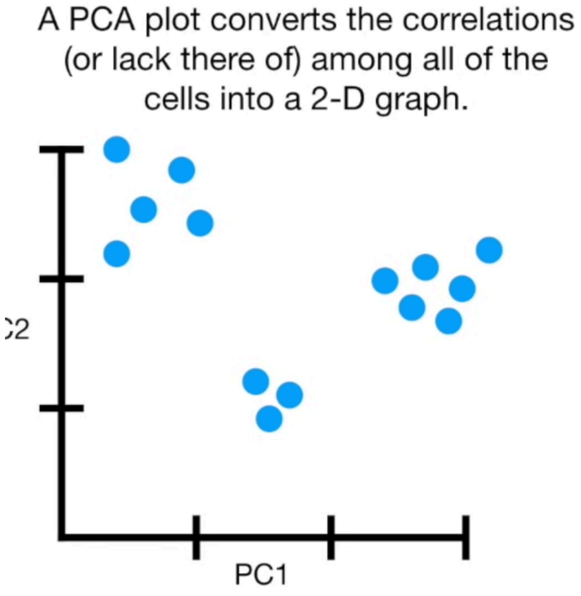




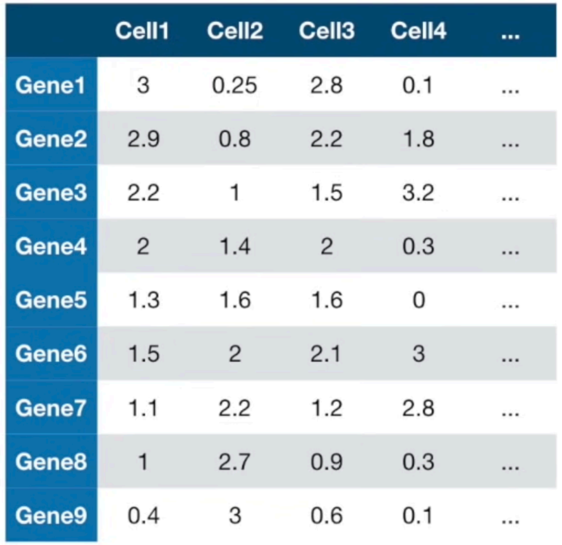
	Cell1	Cell2	Cell3
Gene1	3	0.25	2.8
Gene2	2.9	0.8	2.2
Gene3	2.2	1	1.5
Gene4	2	1.4	2
Gene5	1.3	1.6	1.6
Gene6	1.5	2	2.1
Gene7	1.1	2.2	1.2
Gene8	1	2.7	0.9
Gene9	0.4	3	0.6

3D plot with 3 cells

PCA!



A PCA plot converts the correlations (or lack there of) among all of the cells into a 2-D graph.

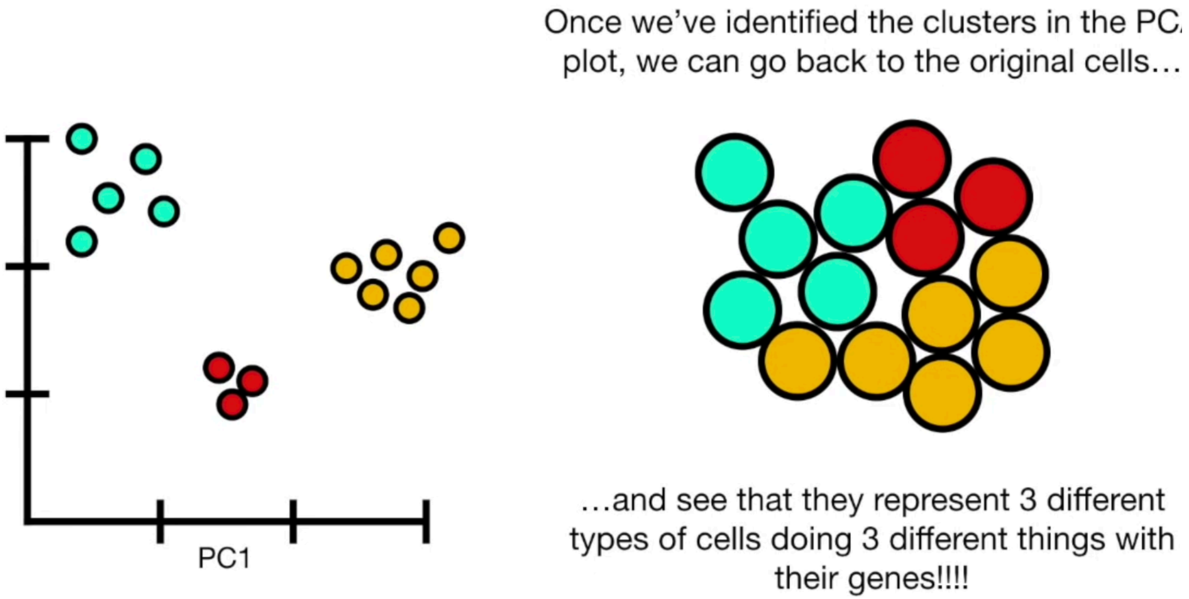


	Cell1	Cell2	Cell3	Cell4	...
Gene1	3	0.25	2.8	0.1	...
Gene2	2.9	0.8	2.2	1.8	...
Gene3	2.2	1	1.5	3.2	...
Gene4	2	1.4	2	0.3	...
Gene5	1.3	1.6	1.6	0	...
Gene6	1.5	2	2.1	3	...
Gene7	1.1	2.2	1.2	2.8	...
Gene8	1	2.7	0.9	0.3	...
Gene9	0.4	3	0.6	0.1	...

PCA converts correlation into a 2D graph

(for 2 components)

PCA!

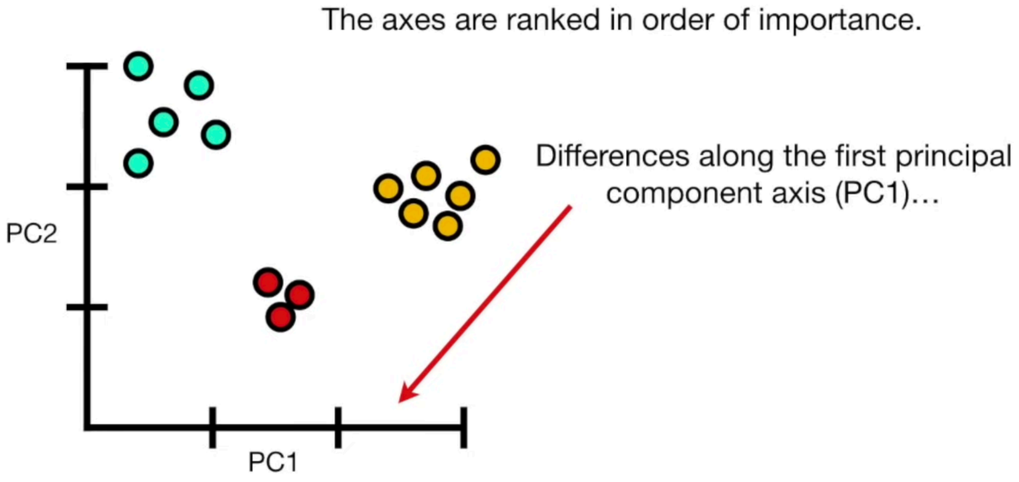


Once we've identified the clusters in the PC
plot, we can go back to the original cells…

…and see that they represent 3 different
types of cells doing 3 different things with
their genes!!!!

now back to initial cells plot

PCA!

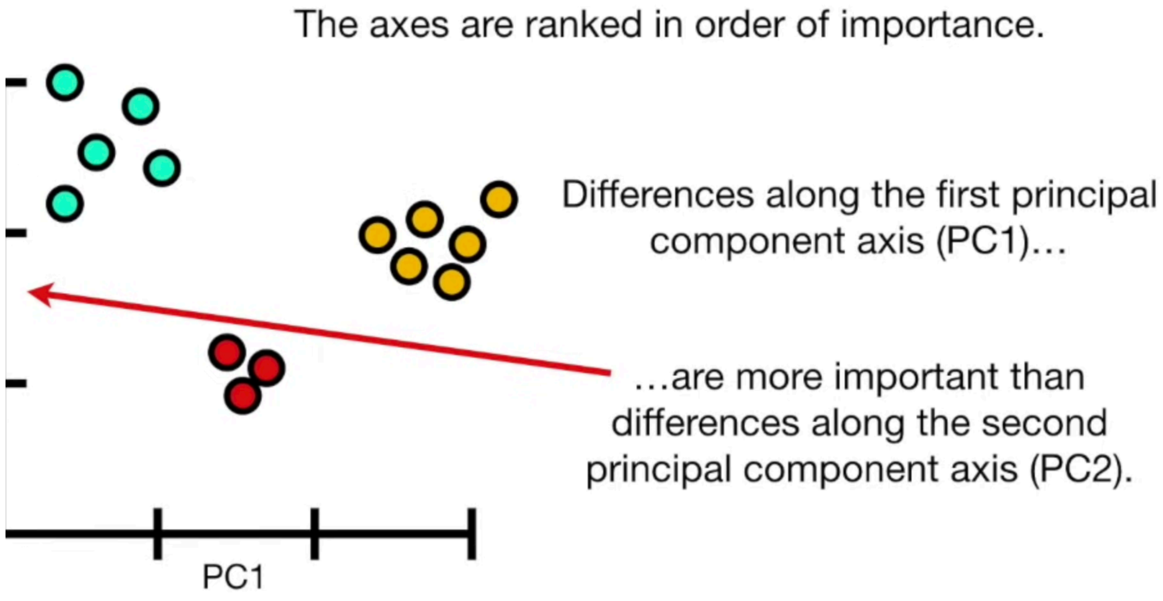


The axes are ranked in order of importance.

Differences along the first principal component axis (PC1)…

now look at x (PC1) axis

PCA!



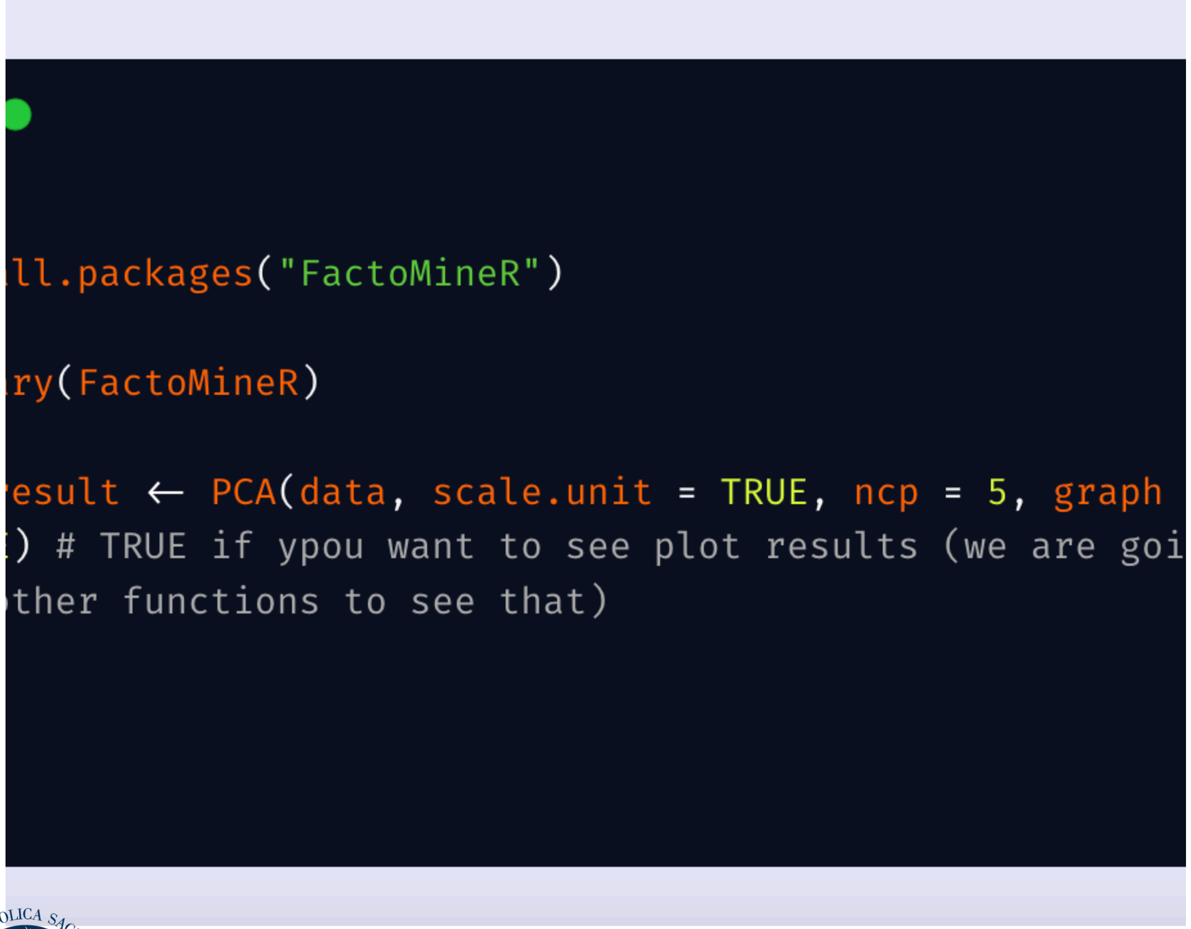
The axes are ranked in order of importance.
Differences along the first principal component axis (PC1)…
…are more important than differences along the second principal component axis (PC2).
PC1

now look at y (PC2) axis

Section 3

PCA in R

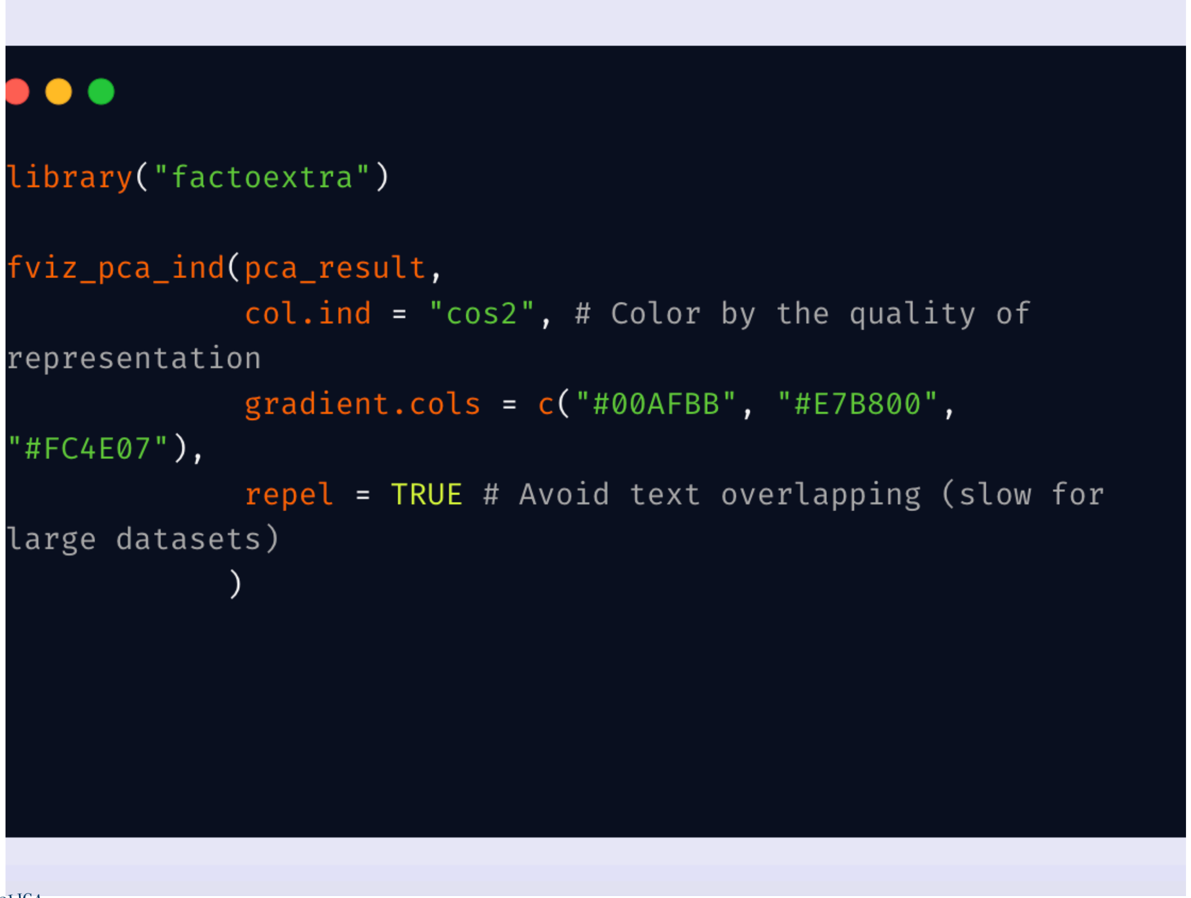
R code...



```
ll.packages("FactoMineR")

ry(FactoMineR)

esult ← PCA(data, scale.unit = TRUE, ncp = 5, graph
) # TRUE if ypou want to see plot results (we are goi
ther functions to see that)
```

PCA results

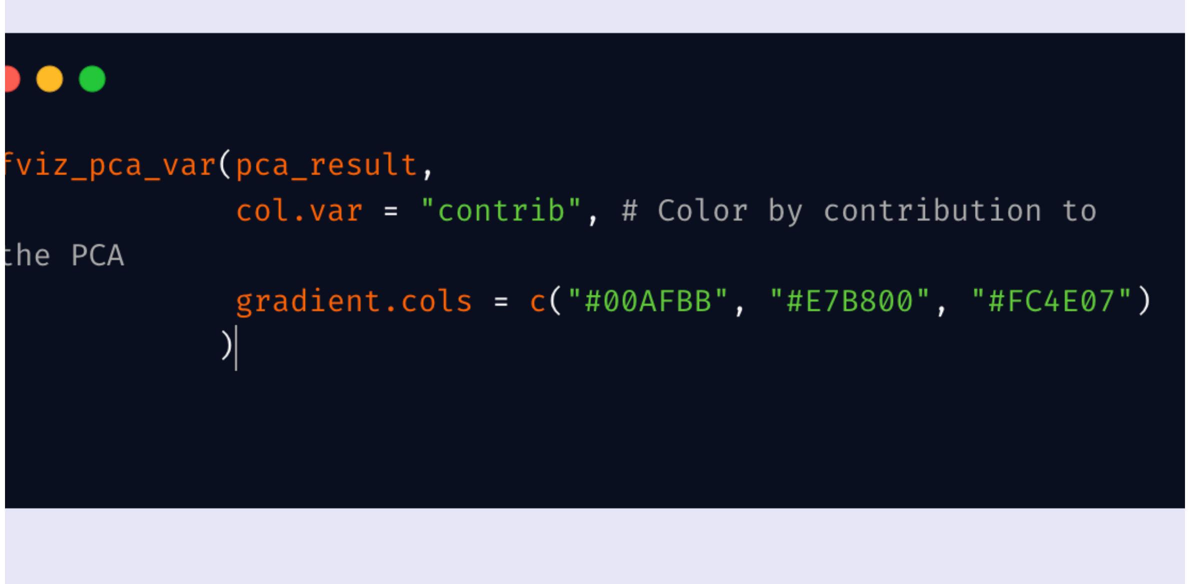
R code...



```
library("factoextra")

fviz_pca_ind(pca_result,
             col.ind = "cos2", # Color by the quality of
representation
             gradient.cols = c("#00AFBB", "#E7B800",
"#FC4E07"),
             repel = TRUE # Avoid text overlapping (slow for
large datasets)
            )
```

Visualizing Individuals (Observations)

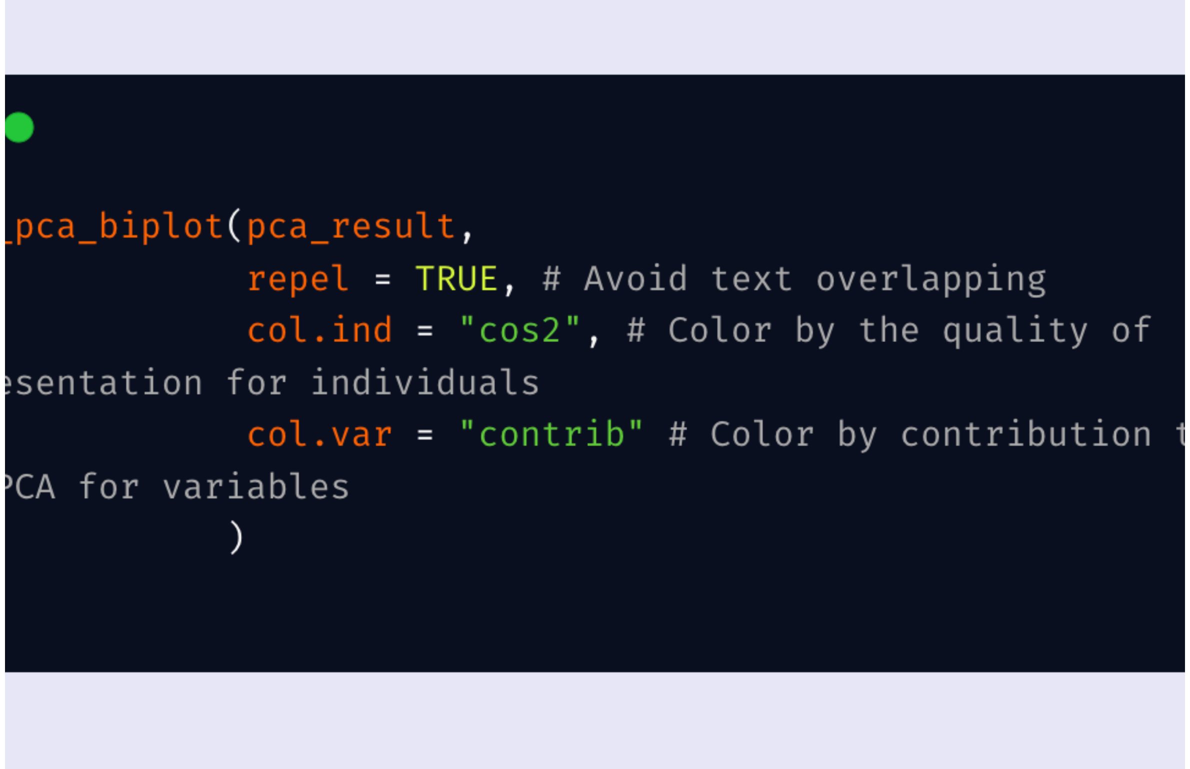
R code...



```
fviz_pca_var(pca_result,
             col.var = "contrib", # Color by contribution to
the PCA
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07")
            )
```

visualising variables

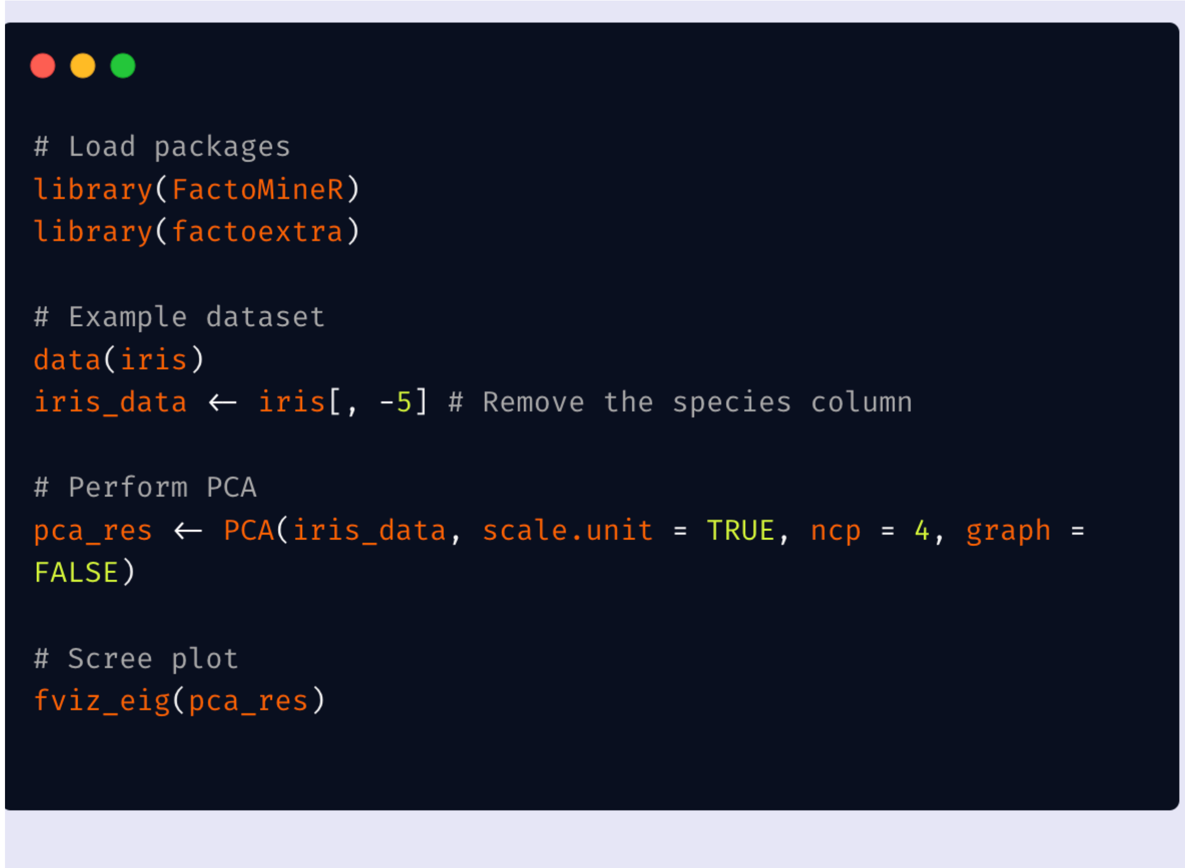
R code...



```
_pca_biplot(pca_result,
            repel = TRUE, # Avoid text overlapping
            col.ind = "cos2", # Color by the quality of
esentation for individuals
            col.var = "contrib" # Color by contribution t
PCA for variables
           )
```

Creating a Biplot

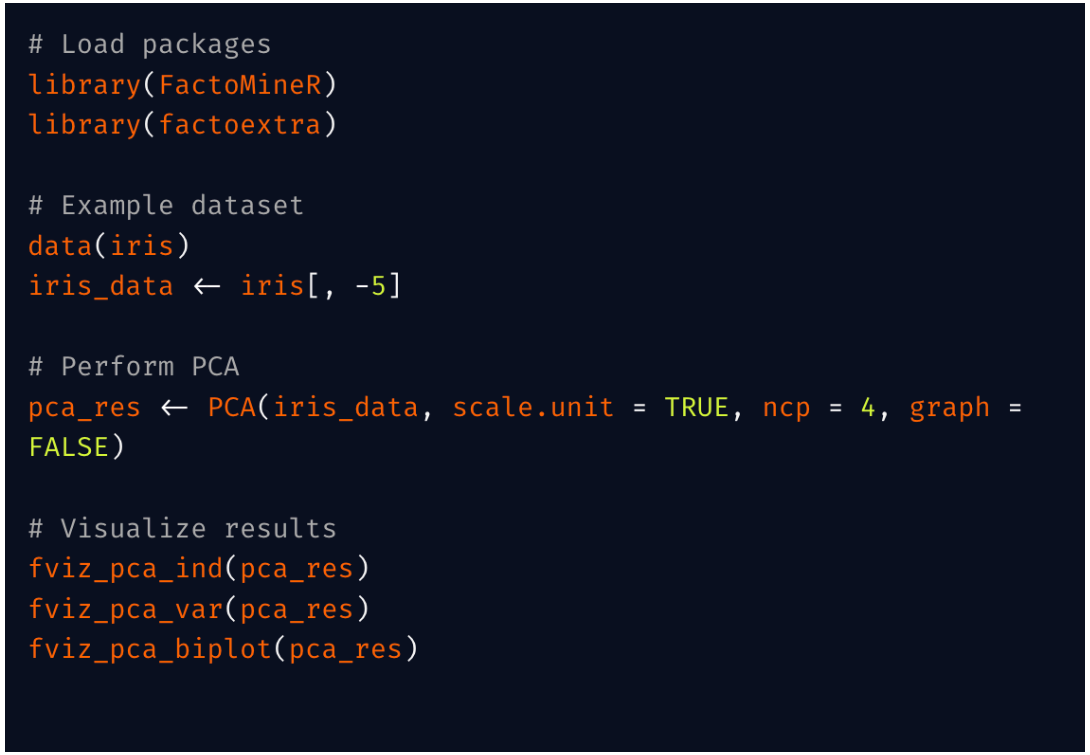
R code…



```
# Load packages
library(FactoMineR)
library(factoextra)

# Example dataset
data(iris)
iris_data ← iris[, -5] # Remove the species column

# Perform PCA
pca_res ← PCA(iris_data, scale.unit = TRUE, ncp = 4, graph =
FALSE)

# Scree plot
fviz_eig(pca_res)
```

scree plot

R code...



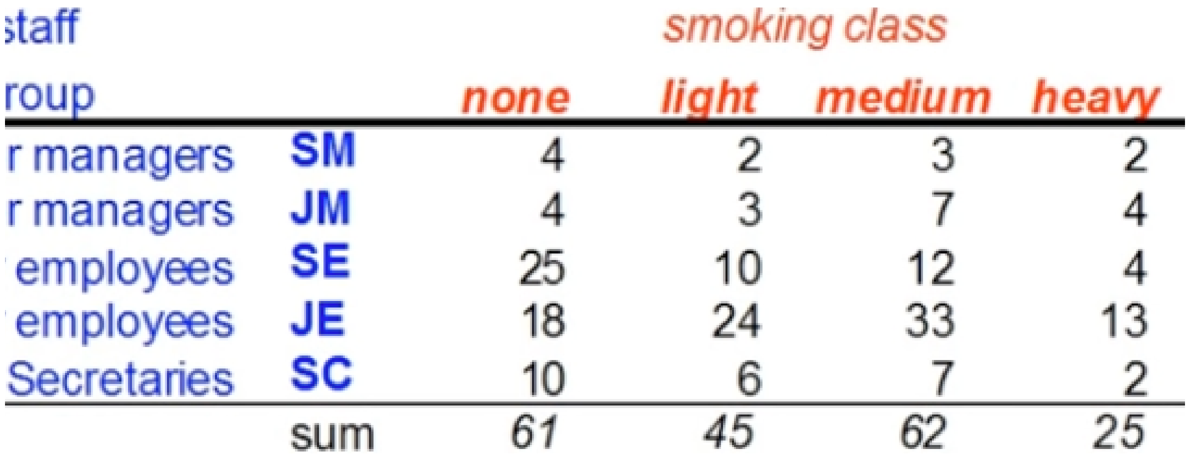
```
# Load packages
library(FactoMineR)
library(factoextra)

# Example dataset
data(iris)
iris_data ← iris[, -5]

# Perform PCA
pca_res ← PCA(iris_data, scale.unit = TRUE, ncp = 4, graph =
FALSE)

# Visualize results
fviz_pca_ind(pca_res)
fviz_pca_var(pca_res)
fviz_pca_biplot(pca_res)
```

working example

Section 1

Correspondant Analysis **(CA)**

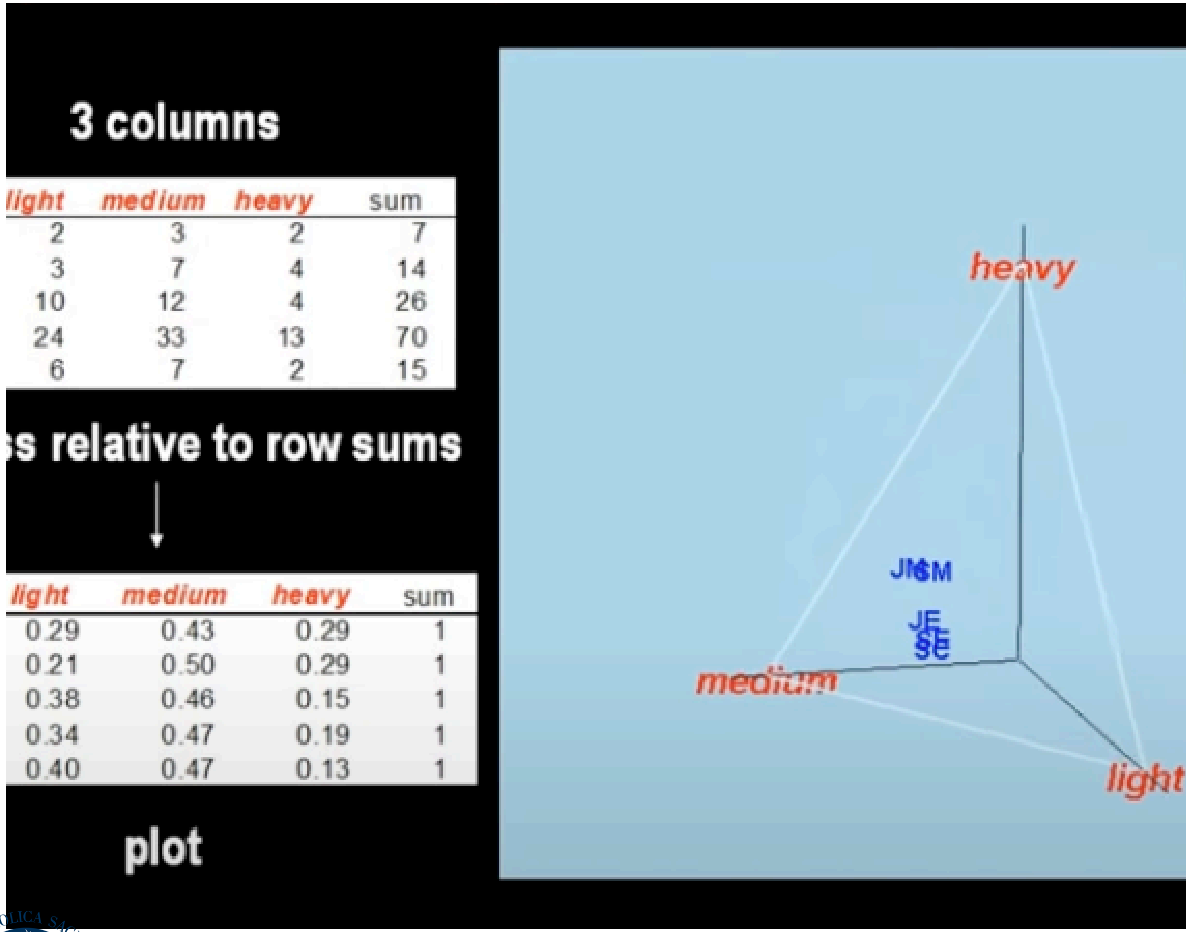
CA background



staff		smoking class			
roup		*none*	*light*	*medium*	*heavy*
r managers	**SM**	4	2	3	2
r managers	**JM**	4	3	7	4
employees	**SE**	25	10	12	4
employees	**JE**	18	24	33	13
Secretaries	**SC**	10	6	7	2
	sum	*61*	*45*	*62*	*25*

Idea? compare thin

CA background



3 columns

light	medium	heavy	sum
2	3	2	7
3	7	4	14
10	12	4	26
24	33	13	70
6	7	2	15

s relative to row sums

↓

light	medium	heavy	sum
0.29	0.43	0.29	1
0.21	0.50	0.29	1
0.38	0.46	0.15	1
0.34	0.47	0.19	1
0.40	0.47	0.13	1

plot

PCA results

Section 4

Live coding session!

JUMP TO RSTUDIO!