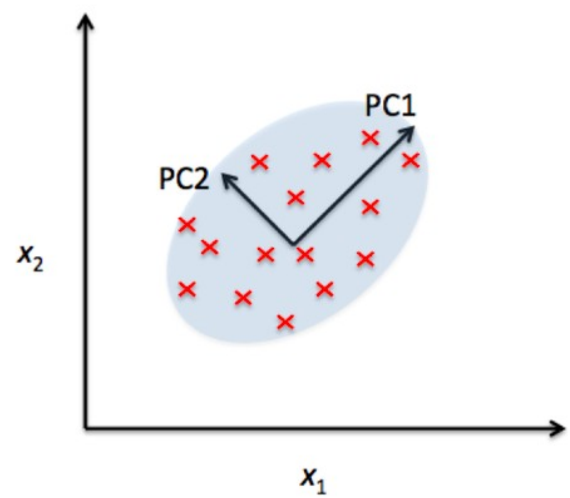
R-course:
Machine Learning using R

Multidimensional data and dimensionality reduction



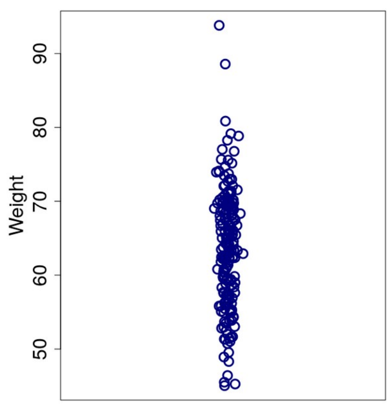
Yannick Rothacher

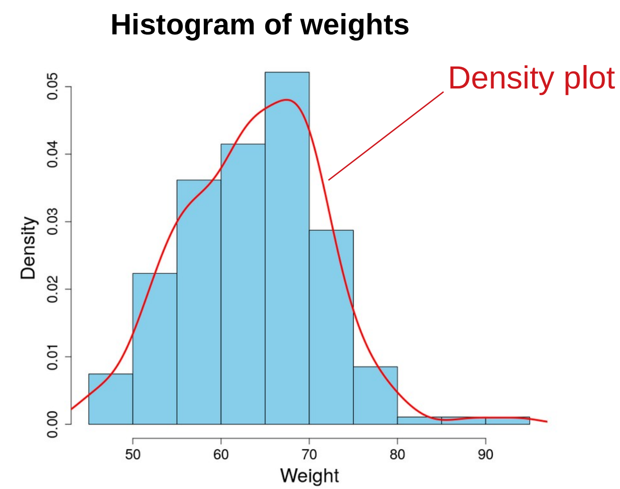
Zürich, 2021

Univariate data set

- This is the most simple data set possible

Participant	Weight (kg)
S1	64
S2	80
S3	55
S4	84
...	...

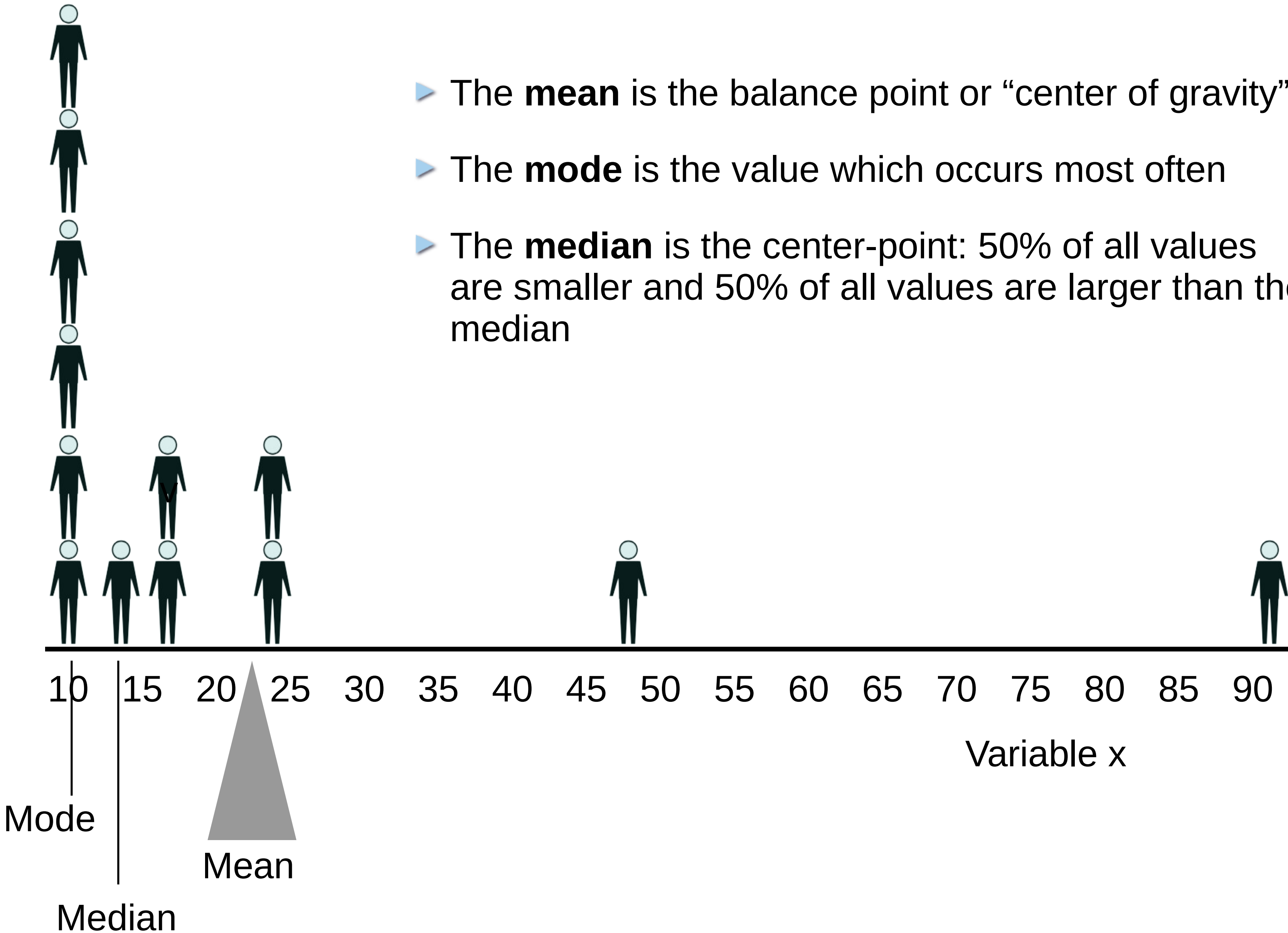




Strictly speaking this is already a multivariate data set because Participant-ID is also a variable.

Recap: Descriptive statistics

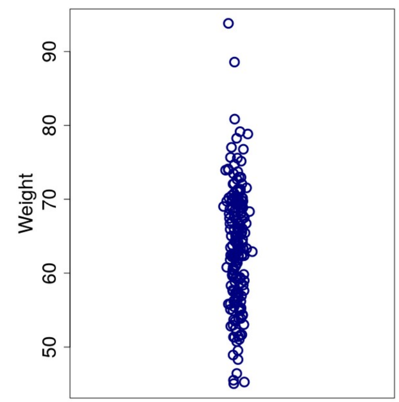
- The **mean** is the balance point or “center of gravity”
- The **mode** is the value which occurs most often
- The **median** is the center-point: 50% of all values are smaller and 50% of all values are larger than the median

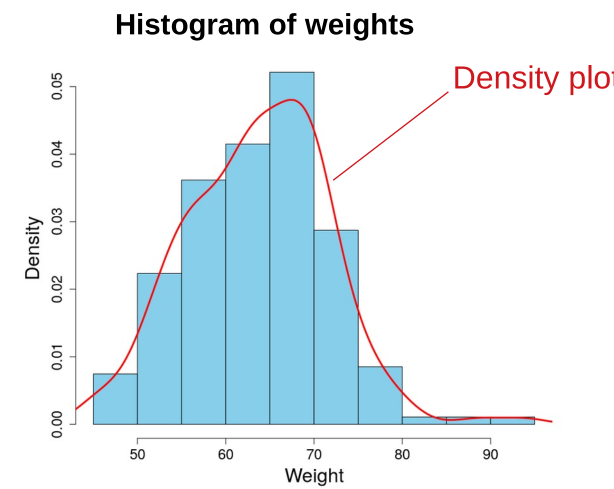


Univariate data set

- This is the most simple data set possible

Participant	Weight (kg)
S1	64
S2	80
S3	55
S4	84
...	...

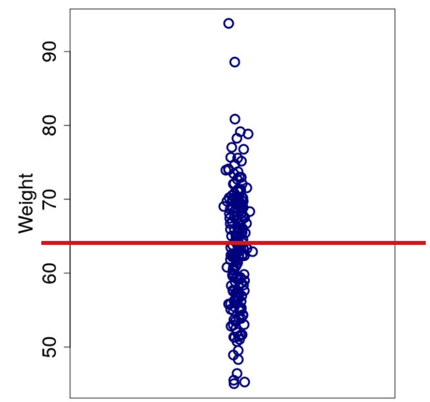


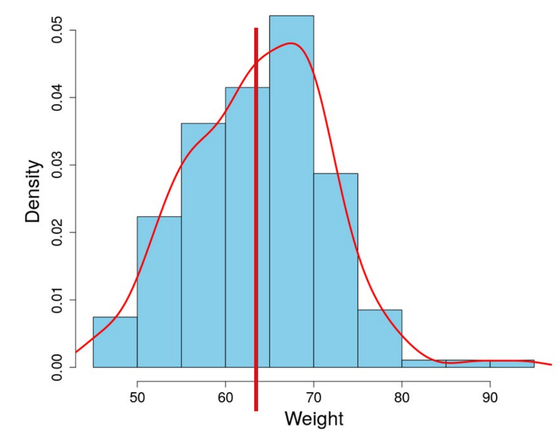


Univariate data set

- Descriptive statistics

Participant	Weight (kg)
S1	64
S2	80
S3	55
S4	84
...	...



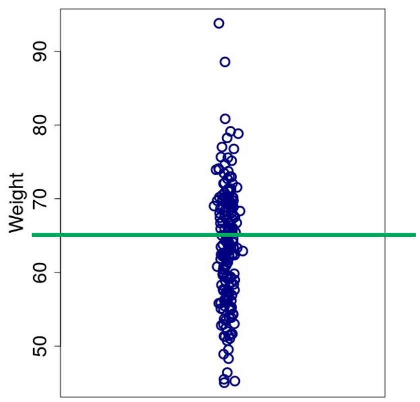


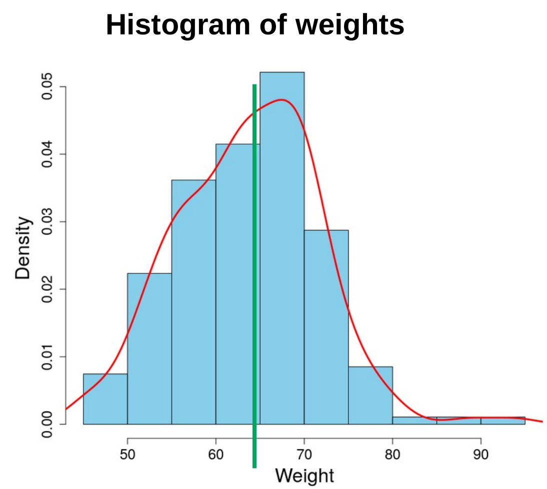
Mean: *"Average of a distribution"* $\bar{x} = \frac{\sum x_i}{n}$ 63.7 kg

Univariate data set

- Descriptive statistics

Participant	Weight (kg)
S1	64
S2	80
S3	55
S4	84
...	...





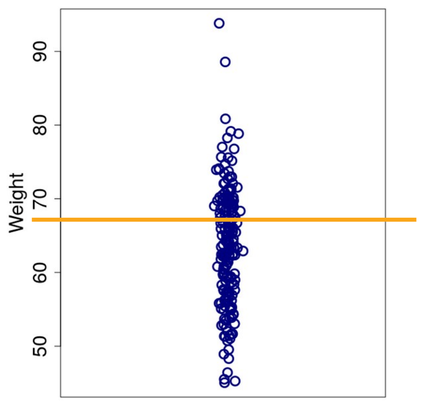
Mean: *"Average of a distribution"* $\bar{x} = \frac{\sum x_i}{n}$ 63.7 kg

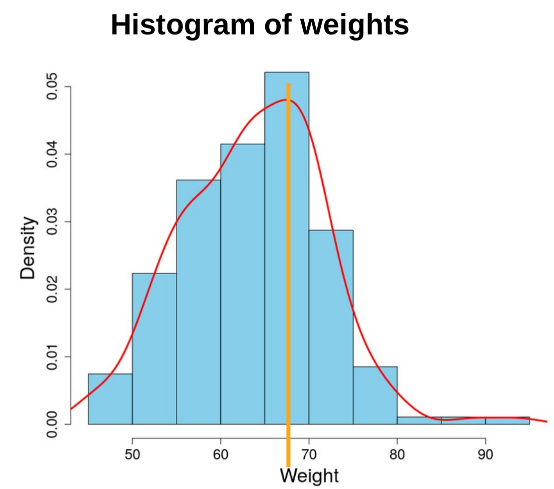
Median: *"Midpoint of a distribution"* 63.9 kg

Univariate data set

- Descriptive statistics

Participant	Weight (kg)
S1	64
S2	80
S3	55
S4	84
...	...





Mean: *"Average of a distribution"* $\bar{x} = \frac{\sum x_i}{n}$ 63.7 kg

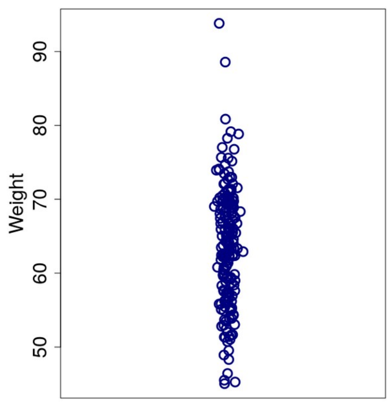
Median: *"Midpoint of a distribution"* 63.9 kg

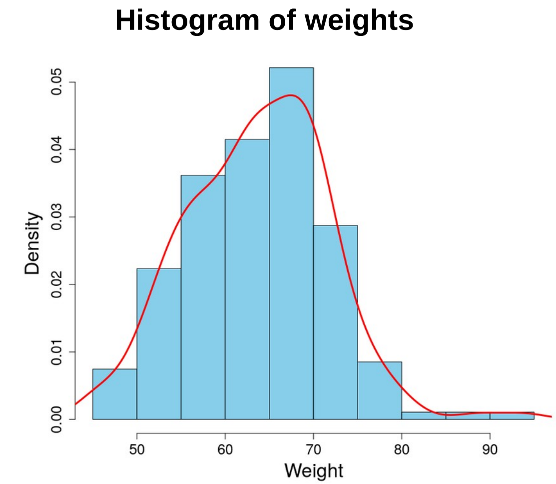
Mode: *"The peak(s) of a distribution"* 67.4 kg

Univariate data set

- Descriptive statistics

Participant	Weight (kg)
S1	64
S2	80
S3	55
S4	84
...	...





Mean: *"Average of a distribution"* $\bar{x} = \frac{\sum x_i}{n}$ 63.7 kg

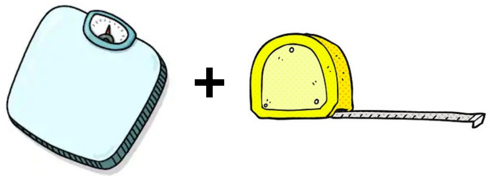
Median: *"Midpoint of a distribution"* 63.9 kg

Mode: *"The peak(s) of a distribution"* 67.4 kg

Standard deviation: *"Spread of distribution"* 7.9 kg

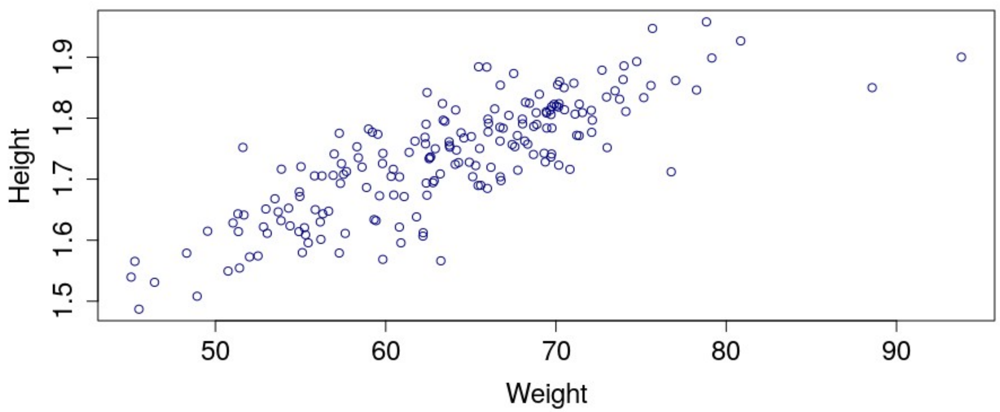
$$s = \sqrt{\frac{\sum (x_i - \bar{x})^2}{n-1}}$$ (sample standard deviation)

Multivariate data set

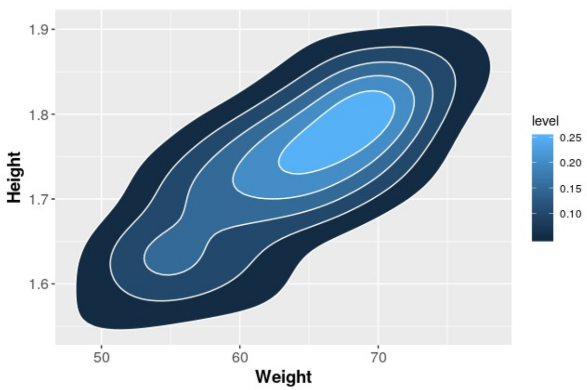


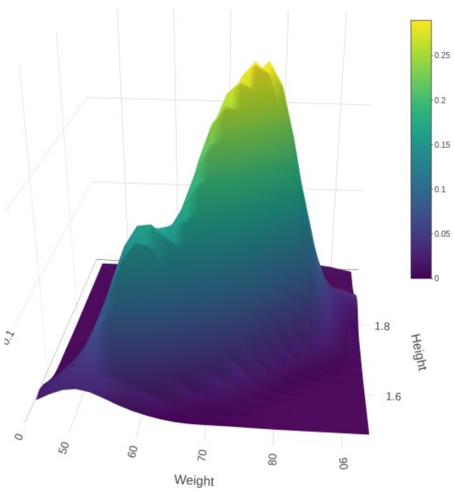
- Two variables

Participant	Weight (kg)	Height (m)
S1	64	1.71
S2	80	1.82
S3	55	1.65
S4	84	1.84
...	...	...

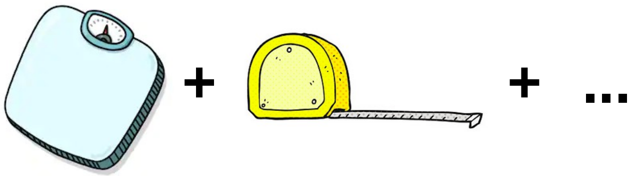


- We can still draw a density plot!





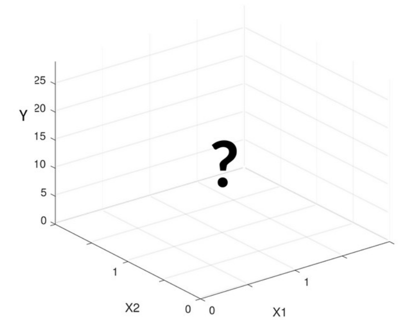
Multivariate data set



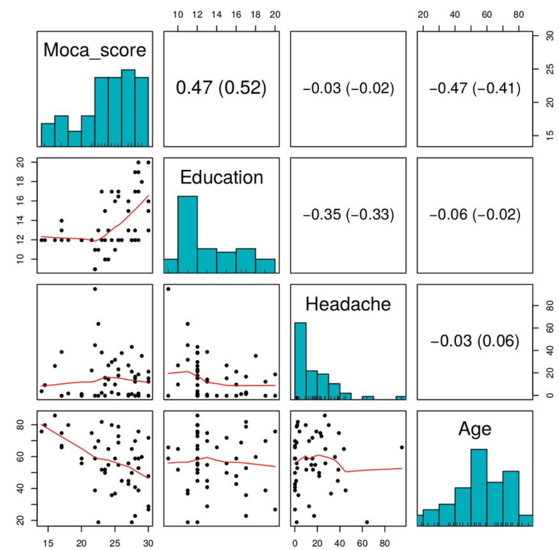
- More than two variables

Participant	Weight (kg)	Height (m)	...
S1	64	1.71	...
S2	80	1.82	...
S3	55	1.65	...
S4	84	1.84	...
...	...	...	...

- Data cannot be visualized anymore...

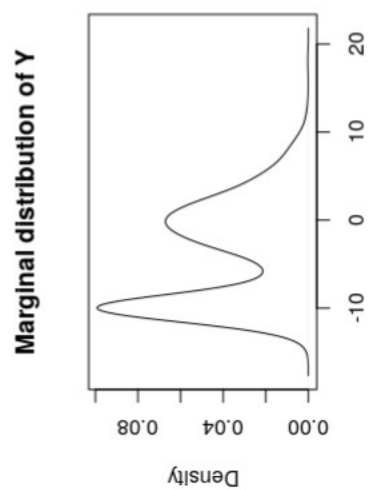


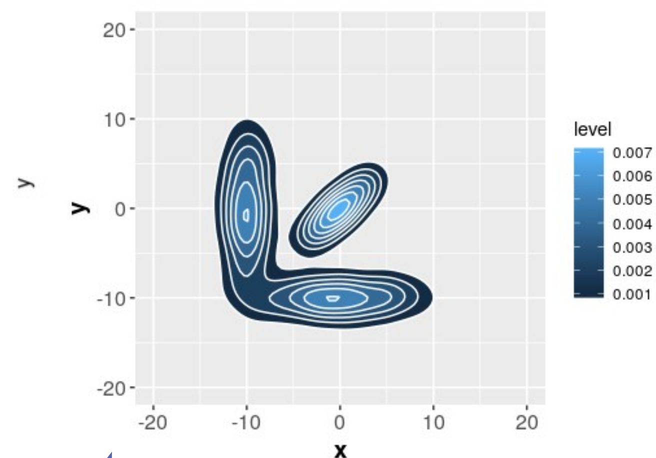
- **Pairs-plot** can help to get an overview of data:

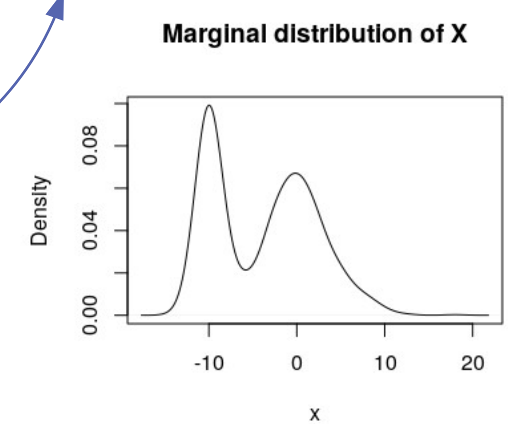


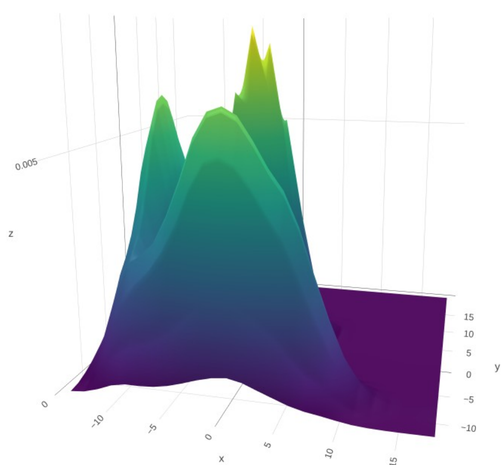
Multivariate data – descriptive statistics

- Does a multivariate distribution have a mode?
  - Yes, the mode is the **most probable realization** of the multidimensional distribution
- The multidimensional mode is **not necessarily composed** of the marginal distributions' modes!



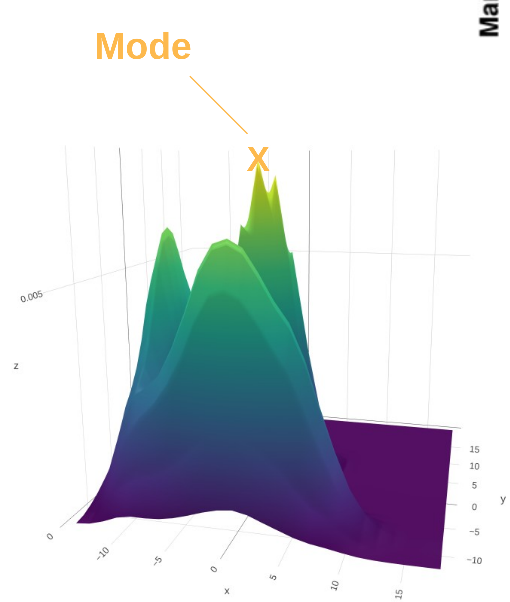


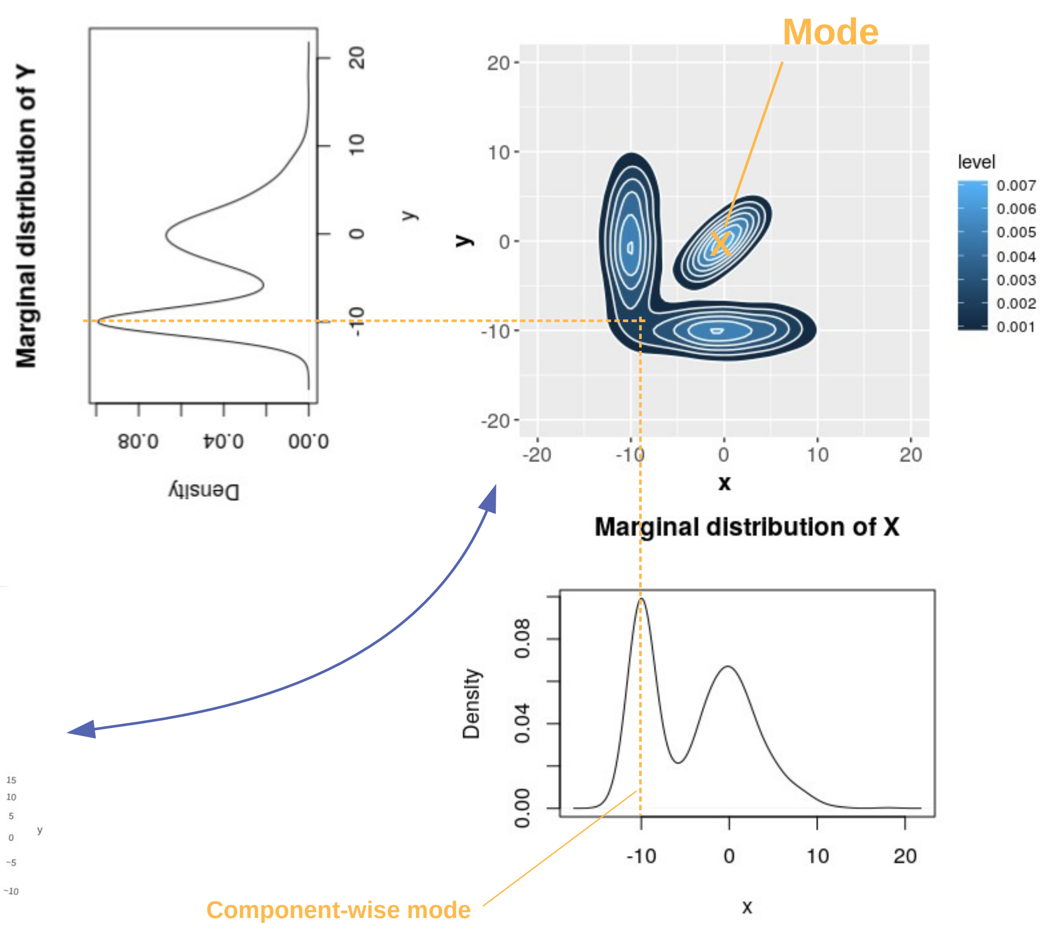




Multivariate data – descriptive statistics

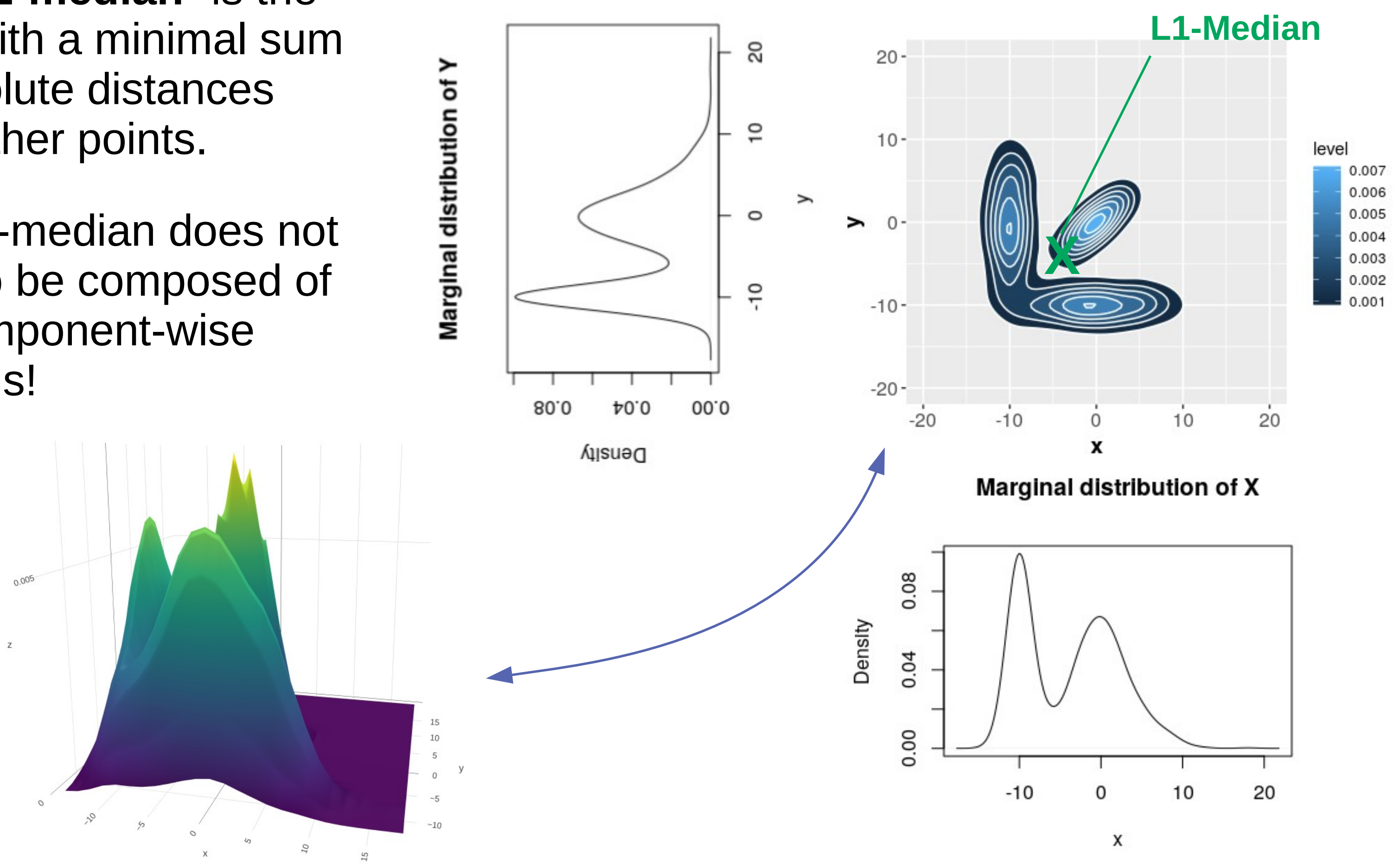
- Does a multivariate distribution have a **mode**?
  - Yes, the mode is the **most probable realization** of the multidimensional distribution
- The multidimensional mode is **not necessarily composed** of the marginal distributions' modes!





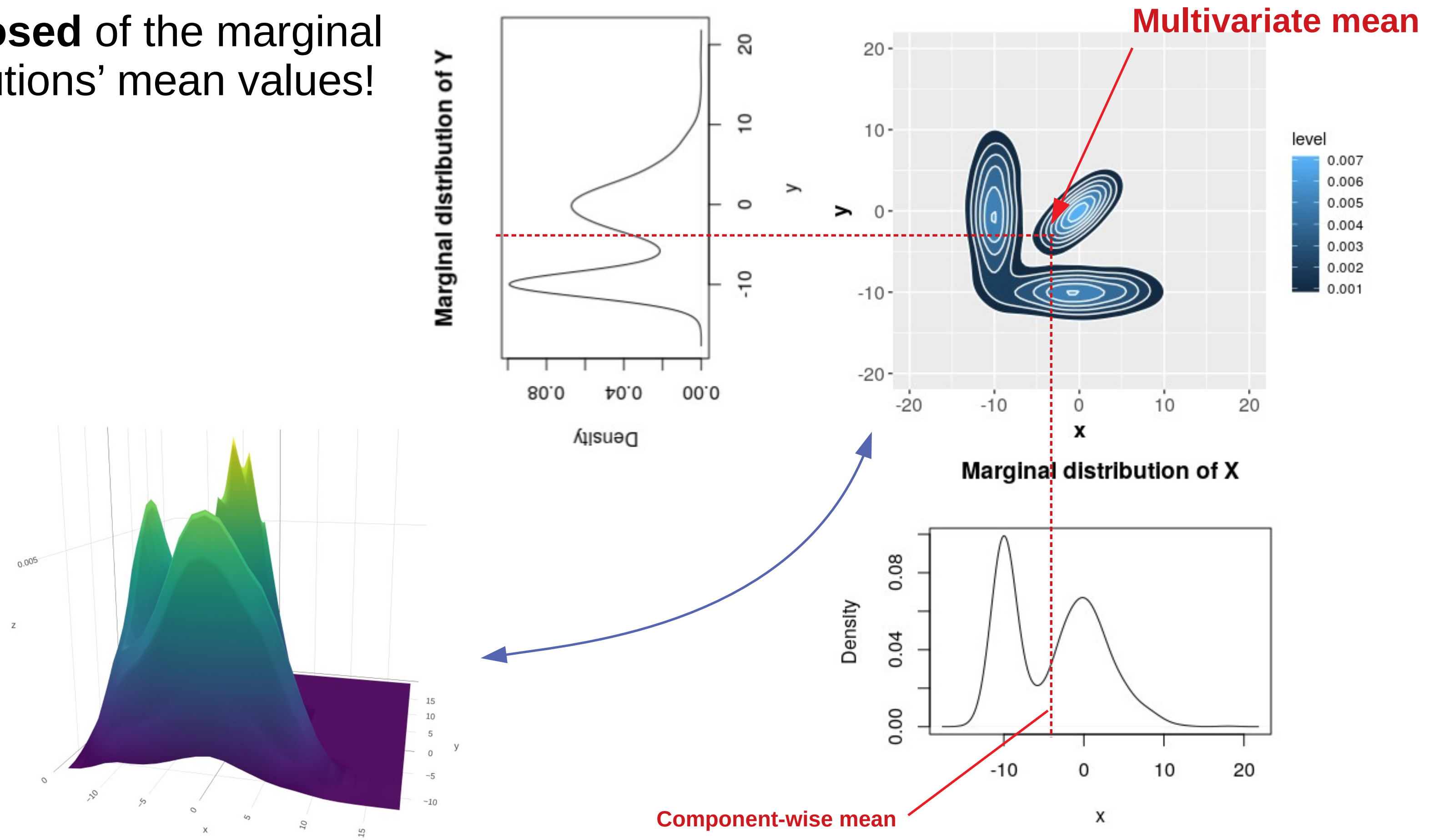
Multivariate data – descriptive statistics

- Does a multivariate distribution have a **median**?
  - A universally accepted definition of a multivariate median does not exist!
- The “**L1-median**” is the point with a minimal sum of absolute distances to all other points.
- The L1-median does not have to be composed of the component-wise medians!



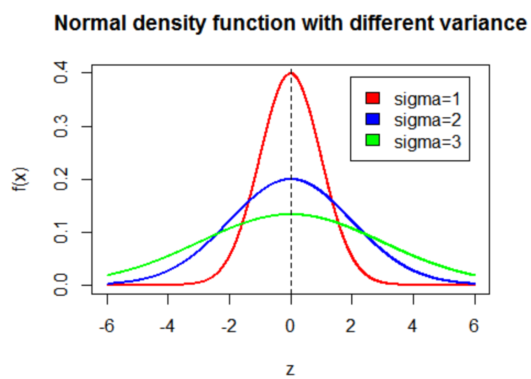
Multivariate data – descriptive statistics

- Does a multivariate distribution have a **mean**?
  - The multivariate mean is the "balance point" of the distribution
- The multivariate mean **is composed** of the marginal distributions' mean values!



Variance of multivariate distribution

- Variance is a measure of the amount of “spread” in a univariate distribution



$$s=\sqrt{\frac{\sum(x_i-\bar{x})^2}{n-1}}$$ (sample standard deviation)

The variance is the squared standard deviation:

Var = s^2

- In the case of multivariate distributions the variance is represented in a **variance-covariance** matrix

$$\mathbf{S}_{pxp}=\begin{array}{c|c} & \begin{matrix} X_1 & X_2 & & X_p \end{matrix} \\ \begin{matrix} X_1 \\ X_2 \\ \\ X_p \end{matrix} & \begin{pmatrix} s_1^2 & s_{12} & \cdots & s_{1p} \\ s_{21} & s_2^2 & \cdots & s_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ s_{p1} & s_{p2} & \cdots & s_p^2 \end{pmatrix} \end{array}$$

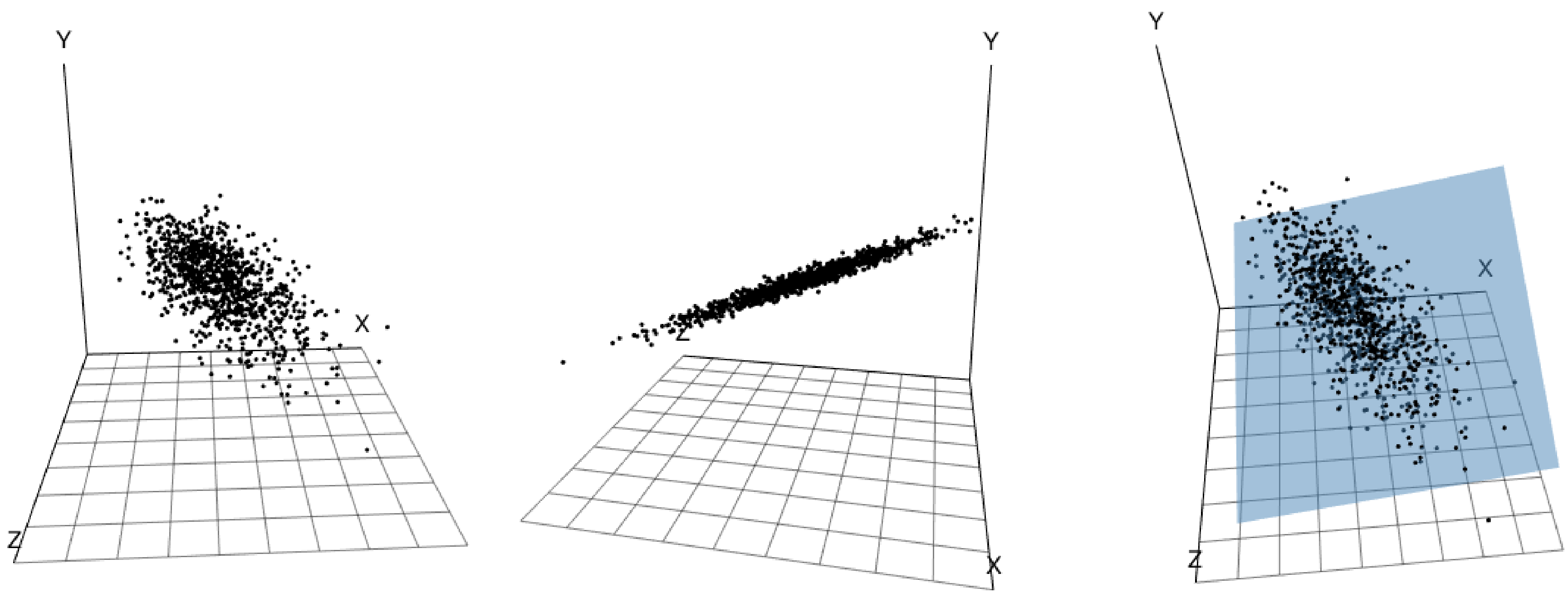
The diagonal elements hold the variances of variables

The off-diagonal elements hold the covariance between the respective variables

$$Cov(x,y)=\frac{\sum(x_i-\bar{x})^2(y_i-\bar{y})^2}{n-1}$$ (sample covariance)

Dimensionality reduction

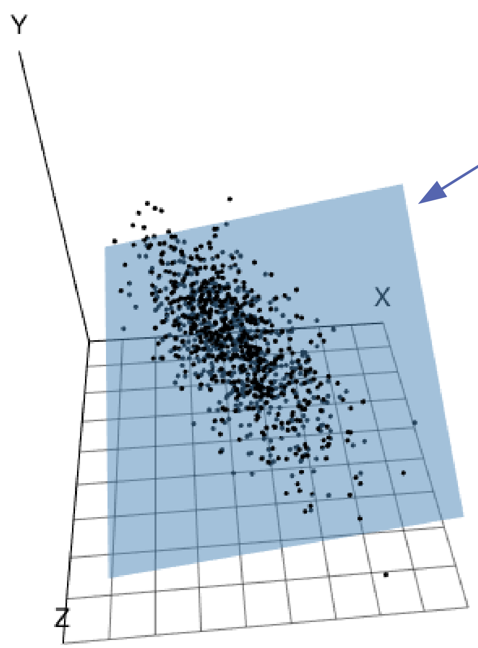
- Is the high number of dimensions really necessary to represent the data?
- Example: The data below is almost lying on a plane
  - The data could also be represented in a two-dimensional coordinate system, without losing a lot of information

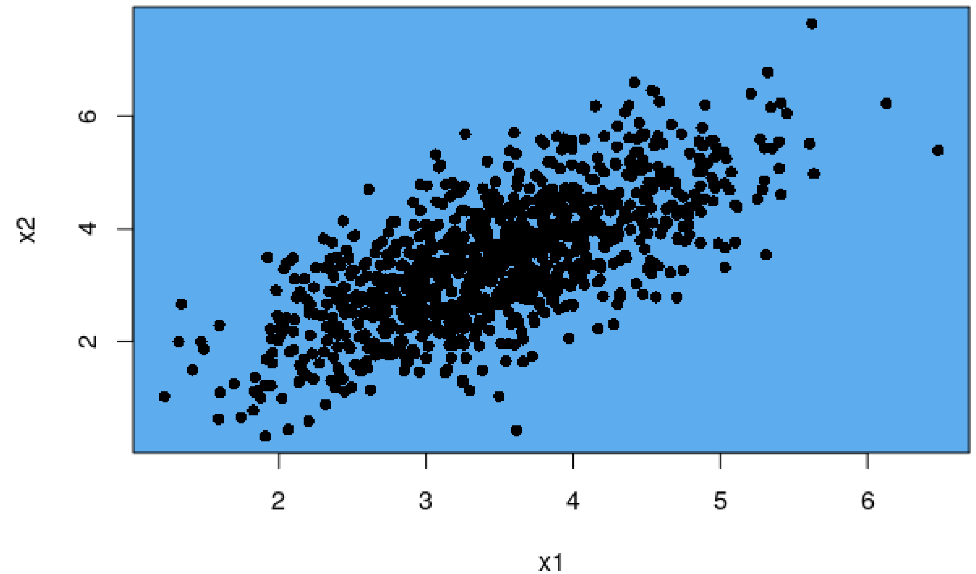


Dimensionality reduction

- Is the high number of dimensions really necessary to represent the data?
- Example: The data below is almost lying on a plane
  - The data could also be represented in a two-dimensional coordinate system, without losing a lot of information

Went from **three** dimensions down to a representation in **two** dimensions by taking the blue plane as the new coordinate system

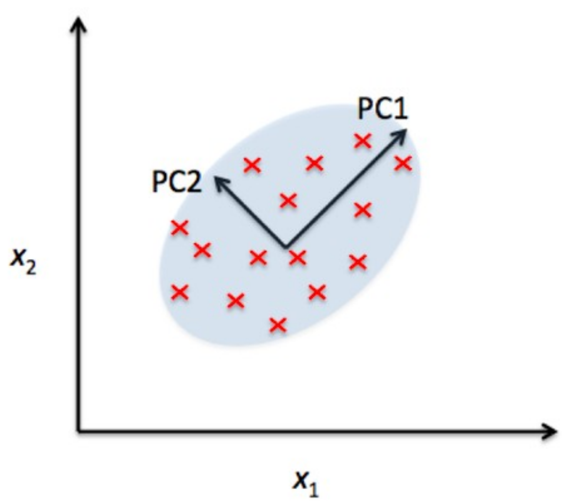




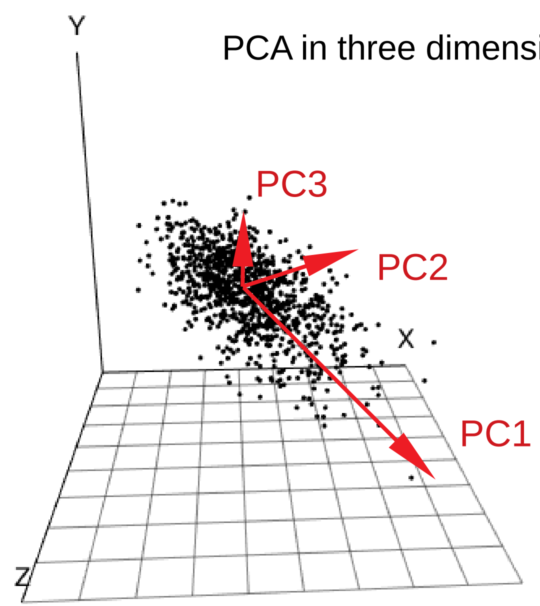
Principal Component Analysis

- **PCA** (**Principal Component Analysis**) is a common method used for dimensionality reduction
- The idea behind PCA is a rotation of the coordinate system (new axes are principal-component 1, principal-compoinent2, …)
  - **Principal-component 1** is the direction in which the data shows the **highest variance**
  - **Principal-component 2** lies **orthogonal** to PC1 and is the direction of the second-highest variance
  - ...

PCA in two dimensions

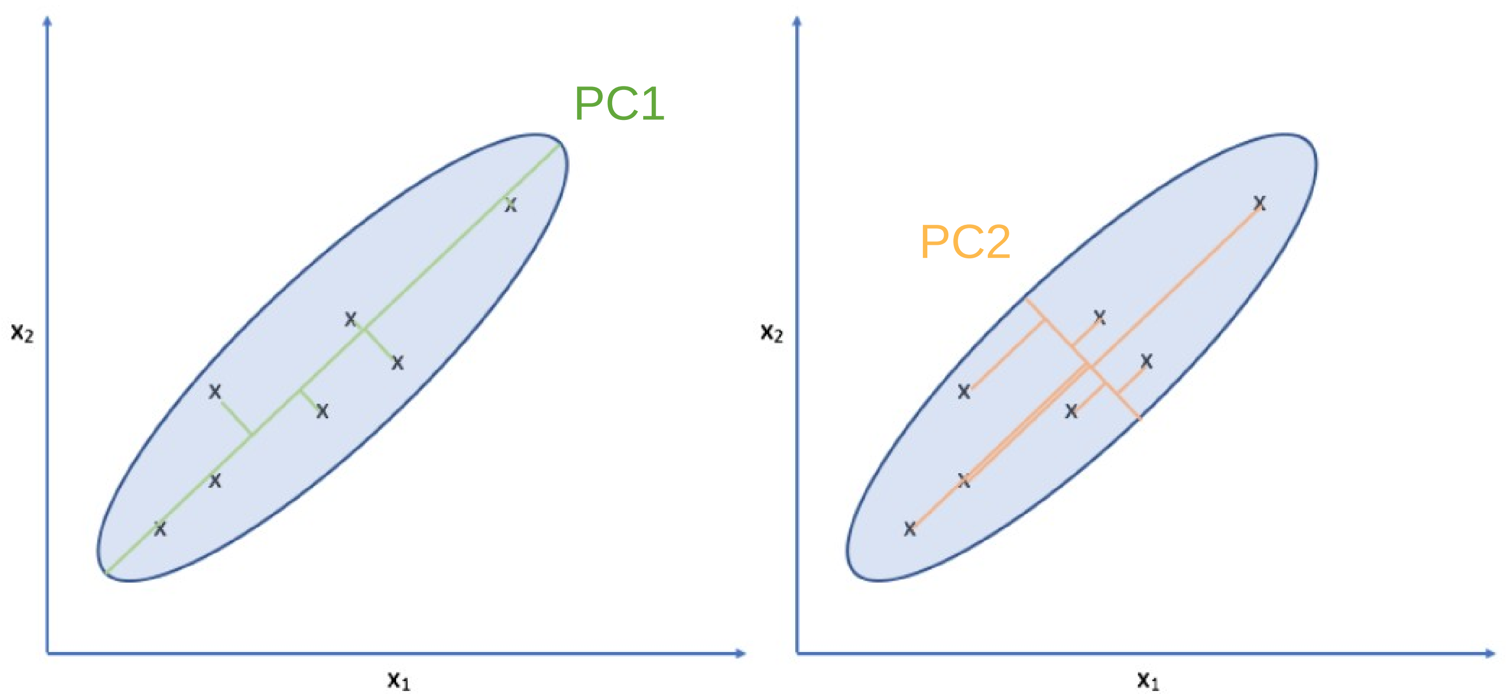


PCA in three dimensions



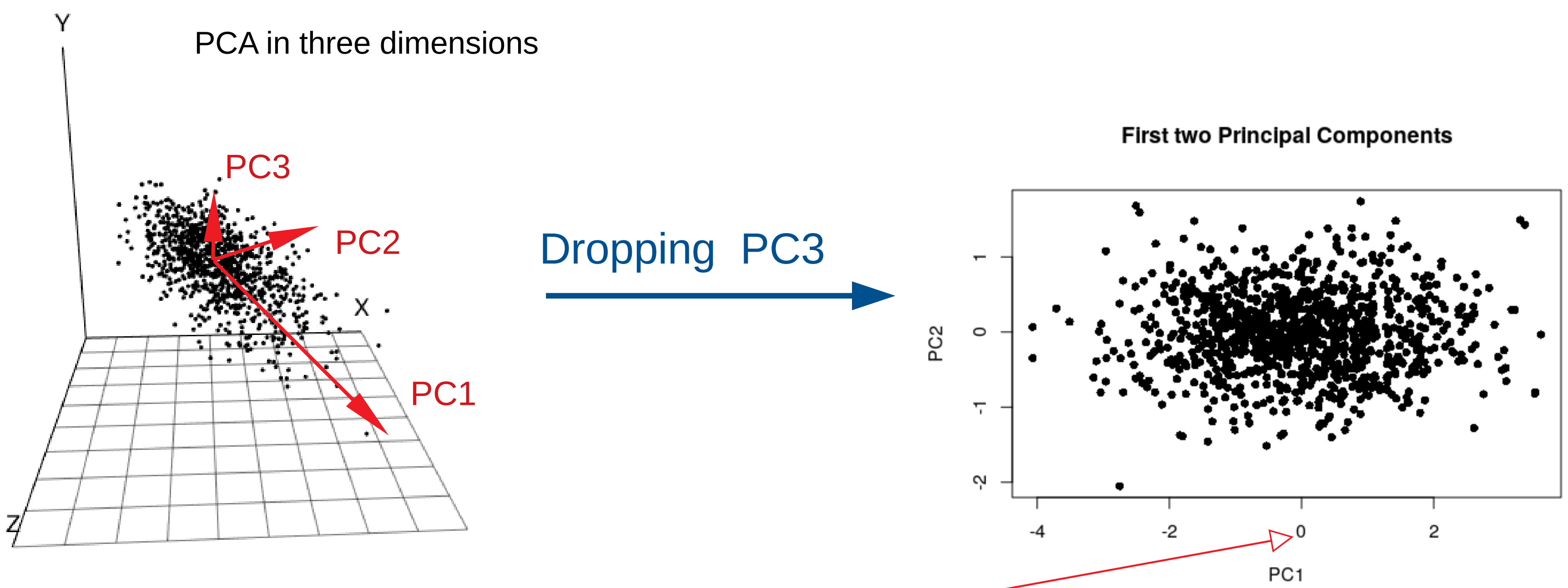
Principal Component Analysis

- One can maximally obtain as many principal component axes as there are dimensions in the data
- Alternative interpretation of principal components:
  - PC1 is the projection line with the minimal sum of squared **orthogonal** distances



Principal Component Analysis

- In practice, the goal is often to visualize the high-dimensional data in a 2D-plot, by dropping all principal components except the first two:



- The data is centered (subtracting the mean value from each variable, respectively) when PCA is applied
  - Often the data is also scaled (see later slides)

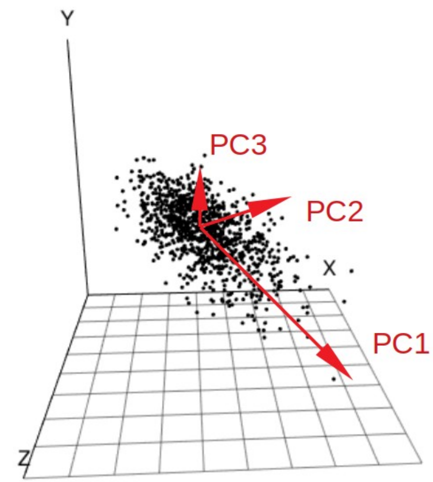
How do we find the PCs?

- After the data has been centered, PCA is just a rotation of the coordinate system (no information is lost)
  - PCA is always possible, one just has to find the right rotation matrix
- It can be shown with linear algebra, that finding the rotation matrix is easier than expected
- The rotation matrix is formed by the **eigen-vectors** of the **variance-covariance-matrix** of the centered data
  - In the case of scaled data, the principal components are the eigen-vectors of the correlation matrix

PCA in R

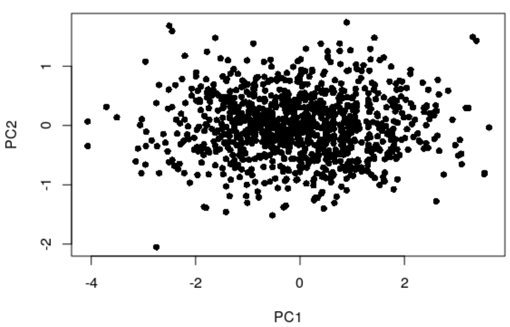
- PCA can be performed in R using the **prcomp()** function

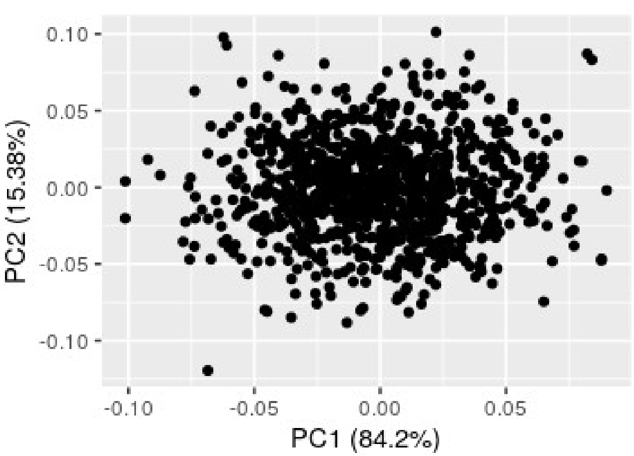
```
> pc <- prcomp(x = myData)  # perform PCA
> summary(pc)
Importance of components:
                          PC1    PC2     PC3
Standard deviation     1.270 0.5430 0.08999
Proportion of Variance 0.842 0.1538 0.00423
Cumulative Proportion  0.842 0.9958 1.00000
```



- Plot the first two PCs:

```
> plot(PC2~PC1, data=pc$x)
# gg-based plot:
> library(ggplot2)
> library(ggfortify)
> autoplot(pc)
```





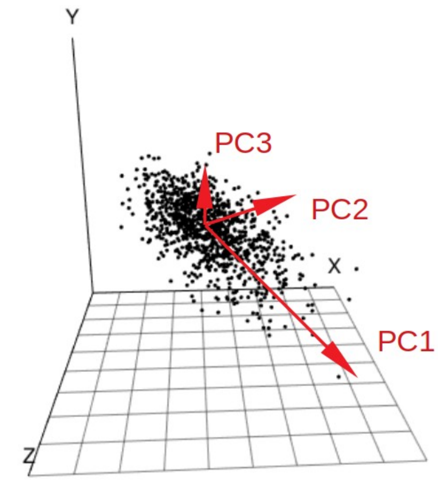
PCA in R

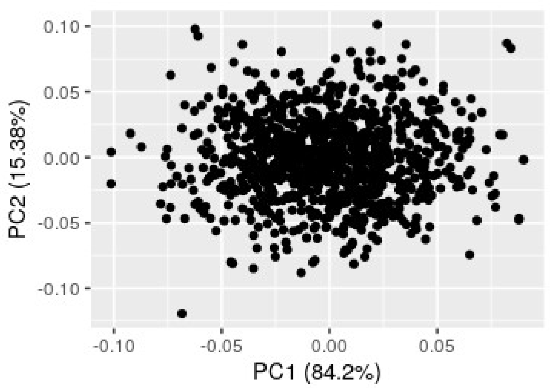
- PCA can be performed in R using the **prcomp()** function

```
> pc <- prcomp(x = myData)  # perform PCA
> pc
Standard deviations (1, .., p=3):
[1] 1.27035807 0.54295340 0.08998992

Rotation (n x k) = (3 x 3):
         PC1        PC2        PC3
x  0.4918003  0.8525767  0.1767639
y -0.2408321  0.3282877 -0.9133603
z  0.8367391 -0.4066205 -0.3667798
```

shows the contribution of the individual variables to the PCs
(also referred to as **variable loadings**)



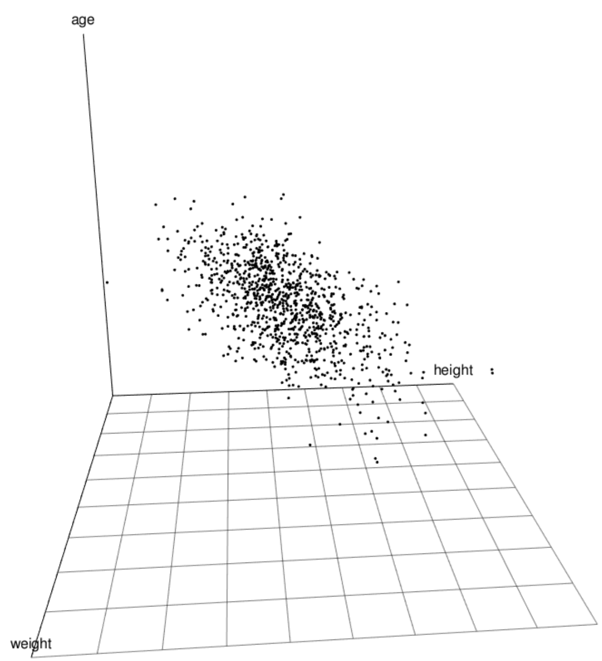


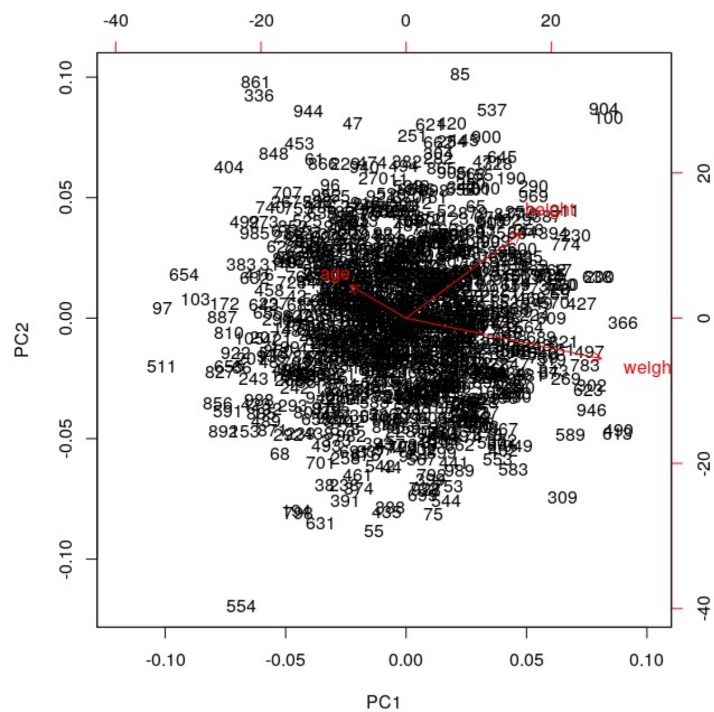
- The first principal component is composed in the following way:

PC1 = 0.4918003***x** – 0.2408321***y** + 0.8367391***z**

Biplot – projection of variables

- In Biplots the original variables are projected into the PC-coordinate system
  - Gives an impression of the contribution of each variable to the PCs



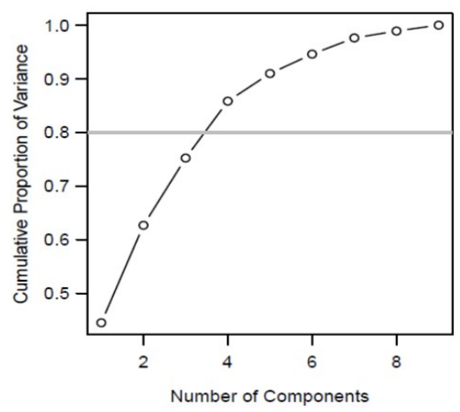


- In R:

```
> pc <- prcomp(x = myData)  # perform PCA
> biplot(pc)
```

How many PCs are needed?

- Take a look at the explained variance by the PCs
- Rule of thumb is that **~80% of total variance** should be explained by the first k PCs

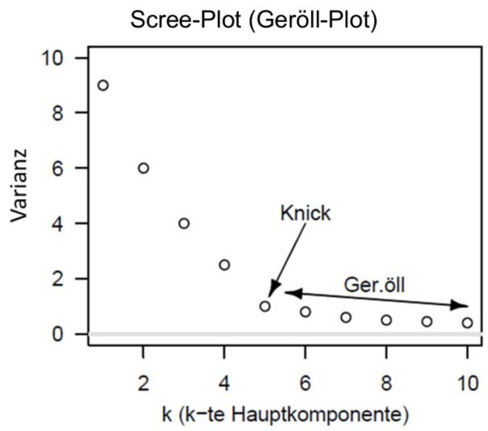


proportion of variance explained
by the first k principle components:

$$P_k = \frac{\sum_{j=1}^{k} \mathrm{var}(Y_j)}{V_{total}} \in [0,1]$$

→ Following this rule, we need the **first four PCs** to explain data

- Other method: Check for a “bend” in the Scree-Plot (Geröll-Plot)




```
# In R:
> screeplot(pca_object, npcs=10, type=”l”)
```

→ In this case, we need the **first four PCs**, afterwards not much more information is won

When to scale the data?

- Often the data is scaled before performing the PCA
- Scaling means that the values of each variable are divided by the variable's standard deviation
  - Now, every variable has unit-variance (variance = 1)
  - **Scaling does affect the results of the PCA**! (e.g. variable measured in ms vs variable in hours, see covariance-table below)
- In R scaling can be set in the options of the **prcomp()** command
  - Default is not to scale: prcomp(mydata, scale.=FALSE)
- Scaling is generally **advisable** when the variables have **different scales** (e.g. cm, m, kg, …)
- Scaling is **not advised** when the variables have the **same scale** and are comparable with respect to their variability

→ Assault will be a large component of PC1 because of its large variance

	Murder	Assault	UrbanPop	Rape
Murder	18.970465	291.0624	4.386204	22.99141
Assault	291.062367	6945.1657	312.275102	519.26906
UrbanPop	4.386204	312.2751	209.518776	55.76808
Rape	22.991412	519.2691	55.768082	87.72916