



University
of Manitoba

Principal component analysis

MATH 2740 – Mathematics of Data Science – Unit U05

Julien Arino

`julien.arino@umanitoba.ca`

Department of Mathematics @ University of Manitoba

Fall 2026

The University of Manitoba campuses are located on original lands of Anishinaabeg, Ininew, Anisininew, Dakota and Dene peoples, and on the National Homeland of the Red River Métis. We respect the Treaties that were made on these territories, we acknowledge the harms and mistakes of the past, and we dedicate ourselves to move forward in partnership with Indigenous communities in a spirit of Reconciliation and collaboration.

Outline

A running example: hockey players

Change of basis

A crash course on probability

Back to PCA

Hockey players

Dimensionality reduction

In the previous unit, we saw that SVD could be used to reduce the dimension of data (matrices)

Now we look at another dimensionality reduction technique: principal component analysis (PCA)

PCA is often used as a blackbox technique, here we take a look at the math behind it

What is PCA?

Linear algebraic technique

Helps reduce a complex dataset to a lower dimensional one

Non-parametric method: does not assume anything about data distribution
(distribution from the statistical point of view)

A running example: hockey players

Change of basis

A crash course on probability

Back to PCA

Hockey players

A 2D example

Dataset (link) of height and weight of some hockey players

```
# From https://figshare.com/ndownloader/files/5303173  
data = read.csv("https://github.com/julien-arino/math-of-data-science/raw/re  
dim(data)  
  
## [1] 6292 13
```

In case you are wondering, this is a database of ice hockey players at IIHF world championships, 2001-2016, assembled by the dataset's author

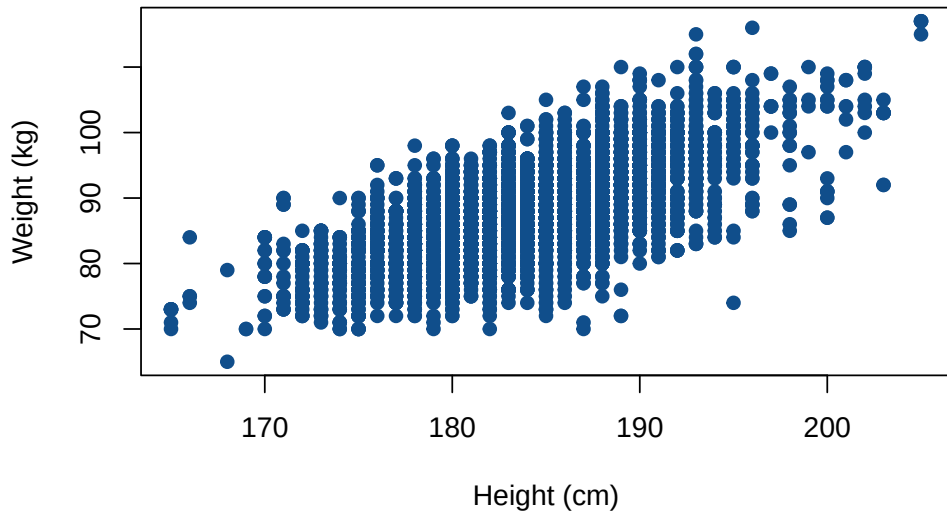
See some comments [here](#)

```
head(data, n=3)
```

```
##      year country no      name position side height weight      bi
## 1 2001      RUS 10   tverdovsky oleg      D      L    185     84 1976-05
## 2 2001      RUS  2   vichnevsky vitali      D      L    188     86 1980-03
## 3 2001      RUS 26 petrochinin evgeni      D      L    182     95 1976-02
##
##              club      age cohort      bmi
## 1   anaheim mighty ducks  24.95277   1976 24.54346
## 2   anaheim mighty ducks  21.11978   1980 24.33228
## 3 severstal cherepovetal  25.22930   1976 28.68011
```

As usual, it is a good idea to plot this to get a sense of the lay of the land

IIHF players 2001–2016 (unprocessed)



The author of the study is interested in the evolution of weights, so it is likely that the same person will be in the dataset several times

Let us check this: first check below will be FALSE if the number of unique names does not match the number of rows in the dataset

```
length(unique(data$name)) == dim(data)[1]
```

```
## [1] FALSE
```

```
length(unique(data$name))
```

```
## [1] 3278
```

```
dim(data)[1]
```

```
## [1] 6292
```

We are not interested in the evolution of weights, so we simplify: if more than one record for someone, take average of recorded weights and heights

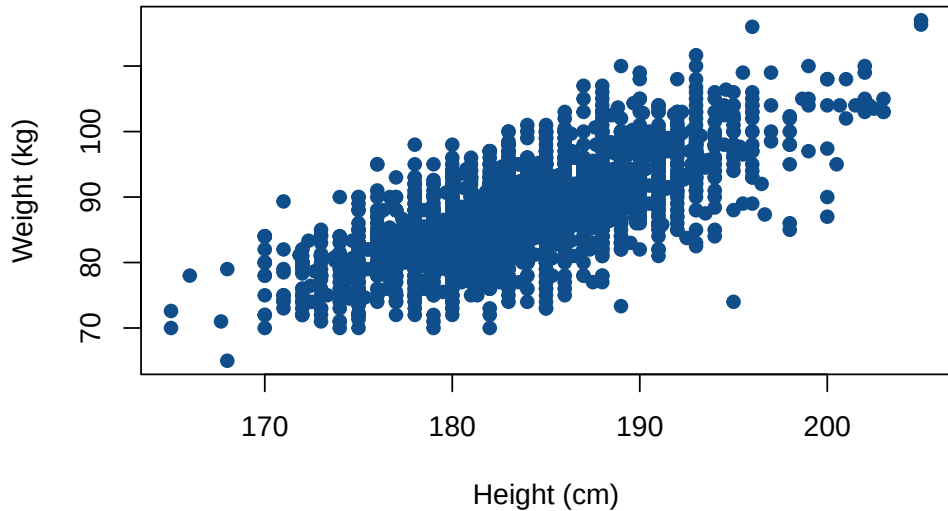
To be extra careful, could check as well that there are no major variations on player height (homonymies?)

```
data_simplified = data.frame(name = unique(data$name))
w = c()
h = c()
for (n in data_simplified$name) {
  tmp = data[which(data$name == n),]
  h = c(h, mean(tmp$height))
  w = c(w, mean(tmp$weight))
}
data_simplified$weight = w
data_simplified$height = h
```

```
data = data_simplified
head(data_simplified, n = 6)
```

##		name	weight	height
## 1		tverdovsky oleg	84.0	185.0
## 2		vichnevsky vitali	86.0	188.0
## 3		petrochinin evgeni	95.0	182.0
## 4		zhdan alexander	85.5	178.5
## 5		orekhovsky oleg	88.0	175.0
## 6		zhukov sergei	92.5	193.0

IIHF players 2001–2016 (uniqued)



Centre the data

```
mean(data$weight)
```

```
## [1] 87.71555
```

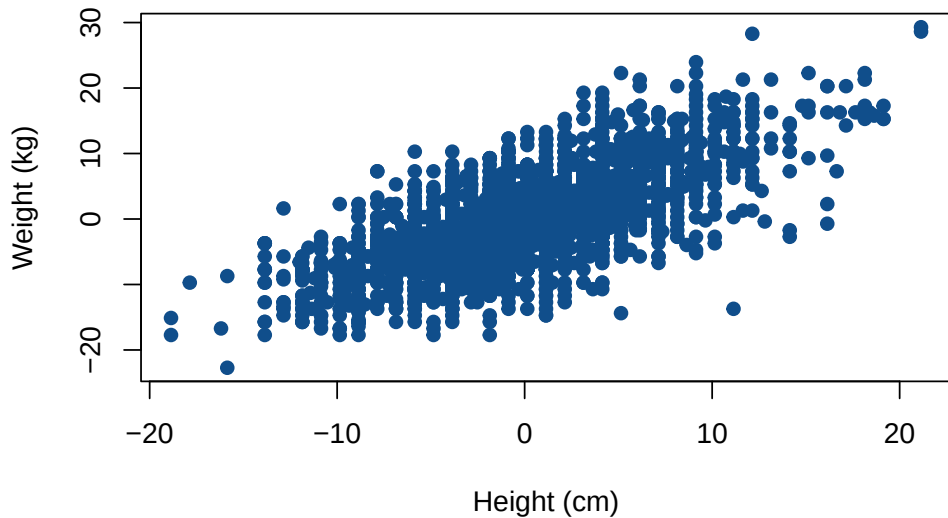
```
mean(data$height)
```

```
## [1] 183.8596
```

```
data$weight.c = data$weight - mean(data$weight)
```

```
data$height.c = data$height - mean(data$height)
```

IIHF players 2001–2016 (centred)



Setting things up

Each player is a row in the data matrix (an *observation*), each feature or variable (*height* and *weight*) is a column

After deduplication, we have an 3278×2 matrix (actually, 3278×4 if we consider the uncentred and centred features, but we will use one or the other, not both uncentred and centred)

We want to find what carries the most information

For this, we are going to project the information in a new basis in which the first “dimension” will carry most information (in a sense we’ll define later), the second dimension will carry a little less, etc.

In order to do so, we need to learn how to change bases

A running example: hockey players

Change of basis

A crash course on probability

Back to PCA

Hockey players

Change of basis

Theory

A simple example

Basis for a trajectory in $\mathbb{R}^2$

In the following,

$$[\mathbf{x}]_{\mathcal{B}}$$

denotes the coordinates of $\mathbf{x}$ in the basis $\mathcal{B} = \{\mathbf{u}_1, \dots, \mathbf{u}_n\}$, i.e., $[\mathbf{x}]_{\mathcal{B}} = (b_1, \dots, b_n)^T$ such that $\mathbf{x} = b_1 \mathbf{u}_1 + \dots + b_n \mathbf{u}_n$

The aim of a change of basis is to express vectors in another coordinate system (another basis) $\mathcal{C} = \{\mathbf{v}_1, \dots, \mathbf{v}_n\}$

We do so by finding a matrix allowing to move from one basis to another

Change of basis

Definition 80 (Change of basis matrix)

Let $\mathcal{B} = \{\mathbf{u}_1, \dots, \mathbf{u}_n\}$ and $\mathcal{C} = \{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ be bases of vector space V

The **change of basis matrix** $P_{\mathcal{C} \leftarrow \mathcal{B}} \in \mathcal{M}_n$ from basis $\mathcal{B}$ to $\mathcal{C}$ is

$$P_{\mathcal{C} \leftarrow \mathcal{B}} = [[\mathbf{u}_1]_{\mathcal{C}} \cdots [\mathbf{u}_n]_{\mathcal{C}}]$$

i.e., its columns are the coordinate vectors $[\mathbf{u}_1]_{\mathcal{C}}, \dots, [\mathbf{u}_n]_{\mathcal{C}}$ of vectors in $\mathcal{B}$ with respect to $\mathcal{C}$

Example – $\mathcal{B}$ is the canonical basis of $\mathbb{R}^3$

Let $\mathcal{B} = \{\mathbf{i}, \mathbf{j}, \mathbf{k}\}$ be the canonical basis of $\mathbb{R}^3$ and $\mathcal{C} = \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\}$, with

$$\mathbf{v}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{v}_2 = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \quad \mathbf{v}_3 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

To find $P_{\mathcal{C} \leftarrow \mathcal{B}} = [[\mathbf{i}]_{\mathcal{C}} \ [\mathbf{j}]_{\mathcal{C}} \ [\mathbf{k}]_{\mathcal{C}}]$, express each canonical vector in terms of $\mathcal{C}$:

$$\mathbf{i} = \mathbf{v}_1, \quad \mathbf{j} = \mathbf{v}_2 - \mathbf{v}_1, \quad \mathbf{k} = \mathbf{v}_3 - \mathbf{v}_2$$

$$\Rightarrow [\mathbf{i}]_{\mathcal{C}} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad [\mathbf{j}]_{\mathcal{C}} = \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix}, \quad [\mathbf{k}]_{\mathcal{C}} = \begin{pmatrix} 0 \\ -1 \\ 1 \end{pmatrix}$$

The change of basis matrix from $\mathcal{B}$ to $\mathcal{C}$ is

$$P_{\mathcal{C} \leftarrow \mathcal{B}} = [[\mathbf{i}]_{\mathcal{C}} \ [\mathbf{j}]_{\mathcal{C}} \ [\mathbf{k}]_{\mathcal{C}}] = \begin{pmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix}$$

Example – $\mathcal{C}$ is the canonical basis of $\mathbb{R}^3$

Now consider the reverse: let $\mathcal{B} = \{\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3\}$ with

$$\mathbf{u}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{u}_2 = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \quad \mathbf{u}_3 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

and let $\mathcal{C} = \{\mathbf{i}, \mathbf{j}, \mathbf{k}\}$ be the canonical basis of $\mathbb{R}^3$

Since $\mathcal{C}$ is the canonical basis, the coordinates in $\mathcal{C}$ are simply the vector itself:

$$[\mathbf{u}_1]_{\mathcal{C}} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad [\mathbf{u}_2]_{\mathcal{C}} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \quad [\mathbf{u}_3]_{\mathcal{C}} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

Thus, the change of basis matrix from $\mathcal{B}$ to the canonical basis $\mathcal{C}$ is

$$P_{\mathcal{C} \leftarrow \mathcal{B}} = [[\mathbf{u}_1]_{\mathcal{C}} \quad [\mathbf{u}_2]_{\mathcal{C}} \quad [\mathbf{u}_3]_{\mathcal{C}}] = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

Theorem 81

$\mathcal{B} = \{\mathbf{u}_1, \dots, \mathbf{u}_n\}$ and $\mathcal{C} = \{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ bases of vector space V and $P_{\mathcal{C} \leftarrow \mathcal{B}}$ a change of basis matrix from $\mathcal{B}$ to $\mathcal{C}$

1. $\forall \mathbf{x} \in V, P_{\mathcal{C} \leftarrow \mathcal{B}}[\mathbf{x}]_{\mathcal{B}} = [\mathbf{x}]_{\mathcal{C}}$
2. $P_{\mathcal{C} \leftarrow \mathcal{B}}$ s.t. $\forall \mathbf{x} \in V, P_{\mathcal{C} \leftarrow \mathcal{B}}[\mathbf{x}]_{\mathcal{B}} = [\mathbf{x}]_{\mathcal{C}}$ is **unique**
3. $P_{\mathcal{C} \leftarrow \mathcal{B}}$ invertible and $P_{\mathcal{C} \leftarrow \mathcal{B}}^{-1} = P_{\mathcal{B} \leftarrow \mathcal{C}}$

Example – Linking the two directions

Let $\mathcal{B} = \{\mathbf{i}, \mathbf{j}, \mathbf{k}\}$ be the canonical basis and $\mathcal{C} = \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\}$ with

$$\mathbf{v}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad \mathbf{v}_2 = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \quad \mathbf{v}_3 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

From our two previous examples:

$$P_{\mathcal{C} \leftarrow \mathcal{B}} = \begin{pmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad P_{\mathcal{B} \leftarrow \mathcal{C}} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

By item 3 in Theorem 81, $P_{\mathcal{C} \leftarrow \mathcal{B}}^{-1} = P_{\mathcal{B} \leftarrow \mathcal{C}}$. Indeed:

$$P_{\mathcal{C} \leftarrow \mathcal{B}} P_{\mathcal{B} \leftarrow \mathcal{C}} = \begin{pmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \mathbb{I}$$

Since $P_{\mathcal{B} \leftarrow \mathcal{C}} = [\mathbf{v}_1 \ \mathbf{v}_2 \ \mathbf{v}_3]$ is immediate, we have $P_{\mathcal{C} \leftarrow \mathcal{B}} = [\mathbf{v}_1 \ \mathbf{v}_2 \ \mathbf{v}_3]^{-1}$

Row-reduction method for changing bases

Theorem 82

$\mathcal{B} = \{\mathbf{u}_1, \dots, \mathbf{u}_n\}$ and $\mathcal{C} = \{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ bases of vector space V . Let $\mathcal{E}$ be any basis for V ,

$$B = [[\mathbf{u}_1]_{\mathcal{E}}, \dots, [\mathbf{u}_n]_{\mathcal{E}}] \text{ and } C = [[\mathbf{v}_1]_{\mathcal{E}}, \dots, [\mathbf{v}_n]_{\mathcal{E}}]$$

and let $[C|B]$ be the augmented matrix constructed using C and B . Then

$$\text{RREF}([C|B]) = [\mathbb{I} | P_{\mathcal{C} \leftarrow \mathcal{B}}]$$

If working in $\mathbb{R}^n$, this is quite useful with $\mathcal{E}$ the standard basis of $\mathbb{R}^n$ (it does not matter if $\mathcal{B} = \mathcal{E}$)

So the question now becomes

How do we find what new basis to look at our data in?

(Changing the basis does not change the data, just the view you have of it)

(Think of what happens when you do a headstand.. your up becomes down, your right and left switch, but the world does not change, just your view of it)

(Changes of bases are *fundamental* operations in Science)

Change of basis

Theory

A simple example

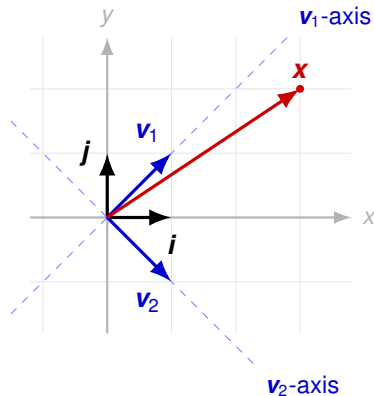
Basis for a trajectory in $\mathbb{R}^2$

Example: change of basis

Problem: Find the change of basis matrix from basis $\mathcal{B}$ to basis $\mathcal{C}$ in $\mathbb{R}^2$, where

$$\mathcal{B} = \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\} \quad \text{and} \quad \mathcal{C} = \left\{ \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ -1 \end{pmatrix} \right\}$$

Then use this matrix to find the coordinates of $\mathbf{x} = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$ in basis $\mathcal{C}$



Step 1 – Set up the matrices

- ▶ $\mathcal{B} = \{\mathbf{i}, \mathbf{j}\}$ is the standard basis of $\mathbb{R}^2$, so using $\mathcal{E} = \mathcal{B}$, we have

$$[\mathbf{i}]_{\mathcal{E}} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad [\mathbf{j}]_{\mathcal{E}} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

- ▶ For $\mathcal{C}$:

$$[\mathbf{v}_1]_{\mathcal{E}} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad [\mathbf{v}_2]_{\mathcal{E}} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

Step 2 – Row reduce

Using Theorem 82, we form the augmented matrix $[C|B]$:

$$[C|B] = \left[\begin{array}{cc|cc} 1 & 1 & 1 & 0 \\ 1 & -1 & 0 & 1 \end{array} \right]$$

Row reduce to RREF:

$$\begin{aligned} \left[\begin{array}{cc|cc} 1 & 1 & 1 & 0 \\ 1 & -1 & 0 & 1 \end{array} \right] &\xrightarrow{R_2 \leftarrow R_2 - R_1} \left[\begin{array}{cc|cc} 1 & 1 & 1 & 0 \\ 0 & -2 & -1 & 1 \end{array} \right] \\ &\xrightarrow{R_2 \leftarrow -\frac{1}{2}R_2} \left[\begin{array}{cc|cc} 1 & 1 & 1 & 0 \\ 0 & 1 & \frac{1}{2} & -\frac{1}{2} \end{array} \right] \\ &\xrightarrow{R_1 \leftarrow R_1 - R_2} \left[\begin{array}{cc|cc} 1 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 1 & \frac{1}{2} & -\frac{1}{2} \end{array} \right] \end{aligned}$$

Step 3 – Extract the change of basis matrix

From the RREF form $[\mathbb{I} | P_{\mathcal{C} \leftarrow \mathcal{B}}]$, we get:

$$P_{\mathcal{C} \leftarrow \mathcal{B}} = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{pmatrix}$$

Verification: Let's check that this matrix works correctly.

- ▶ $[i]_{\mathcal{C}} = P_{\mathcal{C} \leftarrow \mathcal{B}}[i]_{\mathcal{B}} = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}$
- ▶ Check: $\frac{1}{2} \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \frac{1}{2} \begin{pmatrix} 1 \\ -1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$

Step 4 – Coordinates of $\mathbf{x}$ in basis $\mathcal{C}$

Now we find $[\mathbf{x}]_{\mathcal{C}}$ for $\mathbf{x} = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$:

$$[\mathbf{x}]_{\mathcal{C}} = P_{\mathcal{C} \leftarrow \mathcal{B}} [\mathbf{x}]_{\mathcal{B}} = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{pmatrix} \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} \frac{5}{2} \\ \frac{1}{2} \end{pmatrix}$$

Verification: Check that this gives us back the original vector:

$$\frac{5}{2} \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \frac{1}{2} \begin{pmatrix} 1 \\ -1 \end{pmatrix} = \begin{pmatrix} \frac{5}{2} + \frac{1}{2} \\ \frac{5}{2} - \frac{1}{2} \end{pmatrix} = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$$

Answer: The coordinates of $\mathbf{x} = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$ in basis $\mathcal{C}$ are $[\mathbf{x}]_{\mathcal{C}} = \begin{pmatrix} \frac{5}{2} \\ \frac{1}{2} \end{pmatrix}$

Alternative method – Direct calculation

We could also solve this directly by setting up the system:

$$\mathbf{x} = c_1 \mathbf{v}_1 + c_2 \mathbf{v}_2$$

$$\begin{pmatrix} 3 \\ 2 \end{pmatrix} = c_1 \begin{pmatrix} 1 \\ 1 \end{pmatrix} + c_2 \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

This gives us the system:

$$c_1 + c_2 = 3$$

$$c_1 - c_2 = 2$$

Solving: $c_1 = \frac{5}{2}$, $c_2 = \frac{1}{2}$

This confirms our result: $[\mathbf{x}]_{\mathcal{C}} = \begin{pmatrix} \frac{5}{2} \\ \frac{1}{2} \end{pmatrix}$

Computation in R – Finding $P_{C \leftarrow B}$

Set up $C = [\mathbf{v}_1, \mathbf{v}_2]$, $B = [\mathbf{i}, \mathbf{j}]$, form the augmented matrix $[C|B]$ and row reduce:

```
C = matrix(c(1, 1, 1, -1), nrow = 2)
```

```
B = diag(2)
```

```
CB = cbind(C, B)
```

```
(R = pracma::rref(CB))
```

```
##          [,1] [,2] [,3] [,4]
```

```
## [1,]      1     0  0.5  0.5
```

```
## [2,]      0     1  0.5 -0.5
```

```
(P = R[, 3:4])
```

```
##          [,1] [,2]
```

```
## [1,]    0.5  0.5
```

```
## [2,]    0.5 -0.5
```

Computation in R – Coordinates of $\mathbf{x}$

Now compute $[\mathbf{x}]_C = P_{C \leftarrow B}[\mathbf{x}]_B$ and verify that $C[\mathbf{x}]_C = \mathbf{x}$:

```
x = c(3, 2)
(x_C = P %*% x)

##      [,1]
## [1,]  2.5
## [2,]  0.5

# Verification: reconstruct x in the canonical basis
C %*% x_C

##      [,1]
## [1,]    3
## [2,]    2
```

Change of basis

Theory

A simple example

Basis for a trajectory in $\mathbb{R}^2$

Example – A trajectory in $\mathbb{R}^2$

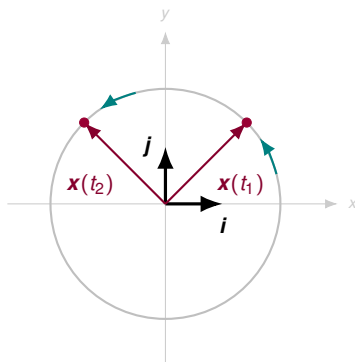
Let $\mathcal{B} = \{\mathbf{i}, \mathbf{j}\}$ be the canonical basis of $\mathbb{R}^2$.

Consider a trajectory along the unit circle:

$$\mathbf{x}(t) = \begin{pmatrix} \cos t \\ \sin t \end{pmatrix}, \quad t \in \mathbb{R}$$

This is a function $\mathbf{x} : \mathbb{R} \rightarrow \mathbb{R}^2$ that we call a **parametric equation**:

- ▶ t is the parameter (often interpreted as time)
- ▶ For each t , $\mathbf{x}(t)$ is a point in $\mathbb{R}^2$; as t varies, $\mathbf{x}(t)$ traces out the trajectory



A moving basis along the trajectory

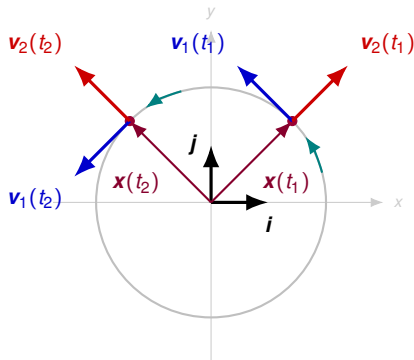
At each point $\mathbf{x}(t)$ on the trajectory, define a moving basis $\mathcal{C}(t) = \{\mathbf{v}_1(t), \mathbf{v}_2(t)\}$:

► $\mathbf{v}_1(t) = \mathbf{x}'(t) = \begin{pmatrix} -\sin t \\ \cos t \end{pmatrix}$ (unit tangent)

► $\mathbf{v}_2(t) = \begin{pmatrix} \cos t \\ \sin t \end{pmatrix}$ (outward unit normal)

Since $\mathbf{v}_1(t) \bullet \mathbf{v}_2(t) = 0$ and $\|\mathbf{v}_1\| = \|\mathbf{v}_2\| = 1$, $\mathcal{C}(t)$ is an orthonormal basis for all t

As t changes, the basis $\{\mathbf{v}_1(t), \mathbf{v}_2(t)\}$ moves and rotates along with $\mathbf{x}(t)$



Coordinates in the moving basis

The change of basis matrix from $\mathcal{C}(t)$ to the canonical basis $\mathcal{B}$ is

$$P_{\mathcal{B} \leftarrow \mathcal{C}(t)} = [\mathbf{v}_1(t) \ \mathbf{v}_2(t)] = \begin{pmatrix} -\sin t & \cos t \\ \cos t & \sin t \end{pmatrix}$$

Since $P_{\mathcal{B} \leftarrow \mathcal{C}(t)}$ is symmetric and orthogonal, the change of basis matrix to $\mathcal{C}(t)$ is:

$$P_{\mathcal{C}(t) \leftarrow \mathcal{B}} = P_{\mathcal{B} \leftarrow \mathcal{C}(t)}^{-1} = P_{\mathcal{B} \leftarrow \mathcal{C}(t)}^T = \begin{pmatrix} -\sin t & \cos t \\ \cos t & \sin t \end{pmatrix}$$

In the moving basis $\mathcal{C}(t)$, the coordinates of position and velocity are:

$$[\mathbf{x}(t)]_{\mathcal{C}(t)} = P_{\mathcal{C}(t) \leftarrow \mathcal{B}} \begin{pmatrix} \cos t \\ \sin t \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad [\mathbf{x}'(t)]_{\mathcal{C}(t)} = P_{\mathcal{C}(t) \leftarrow \mathcal{B}} \begin{pmatrix} -\sin t \\ \cos t \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

In the local frame, the particle is always at distance 1 along the normal axis with velocity 1 along the tangent

Choosing the inward normal $-\mathbf{v}_2(t)$ simply gives $[\mathbf{x}(t)]_{\mathcal{C}(t)} = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$

A running example: hockey players

Change of basis

A crash course on probability

Back to PCA

Hockey players

Why probability?

We said earlier that we would look for a basis in which the first dimension carries most information

But how do we define *information*?

We use concepts from probability and statistics to do so

A good measure of information is *variance* (how much data varies around the mean)

Brief “review” of some probability concepts

Proper definition of *probability* requires to use *measure theory*.. will not get into details here

A **random variable** X is a *measurable* function $X : \Omega \rightarrow E$, where Ω is a set of outcomes (*sample space*) and E is a measurable space

$$\mathbb{P}(X \in S \subseteq E) = \mathbb{P}(\omega \in \Omega | X(\omega) \in S)$$

Distribution function of a r.v., $F(x) = \mathbb{P}(X \leq x)$, describes the distribution of a r.v.

R.v. can be discrete or continuous or .. other things.

Variance

Definition 83 (Variance)

Let X be a random variable. The **variance** of X is given by

$$\text{Var } X = E \left[(X - E(X))^2 \right]$$

where E is the expected value

Covariance

Definition 84 (Covariance)

Let X, Y be jointly distributed random variables. The **covariance** of X and Y is given by

$$\text{cov}(X, Y) = E[(X - E(X))(Y - E(Y))]$$

Note that $\text{cov}(X, X) = E[(X - E(X))^2] = \text{Var } X$

In practice: “true law” versus “observation”

In statistics: we reason on the *true law* of distributions, but we usually have only access to a sample

We then use **estimators** to .. estimate the value of a parameter, e.g., the mean, variance and covariance

Definition 85 (Unbiased estimators of the mean and variance)

Let $x_1, \dots, x_n$ be data points (the *sample*) and

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

be the **mean** of the data. An unbiased estimator of the variance of the sample is

$$\sigma^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

Unbiased: the expected value of the estimator equals the true parameter value ($E(\bar{x}) = E(X)$ and $E(\sigma^2) = \text{Var } X$): on average, it does not systematically overestimate or underestimate the parameter

Variance of hockey players' height and weight

We compute the sample variance in R using the function `var`:

```
var(data$height)
## [1] 29.66176
var(data$weight)
## [1] 47.81112
```

The variance is about 29.7 cm² for height and 47.8 kg² for weight

Normalising does not change the variance

Centring (normalising) shifts data by a constant, leaving spread unchanged:

$$\text{Var}(X - c) = \text{Var } X$$

Indeed, computing the variance using the centred values:

```
var(data$height.c)
## [1] 29.66176
var(data$weight.c)
## [1] 47.81112
```

The variance is identical to that of the uncentred data

Definition 86 (Unbiased estimator of the covariance - 2D case)

Let $(x_1, y_1), \dots, (x_n, y_n)$ be data points,

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \text{ and } \bar{y} = \frac{1}{n} \sum_{i=1}^n y_i$$

be the means of the data. An unbiased estimator of the covariance of the sample is

$$\text{cov}(x, y) = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})$$

Computation of covariance in R – “By hand” and with a function

We compute the means and sum directly from the definition:

```
n = length(data$height)
x_bar = sum(data$height) / n
y_bar = sum(data$weight) / n
sum((data$height - x_bar) * (data$weight - y_bar)) / (n - 1)

## [1] 26.63506

cov(data$height, data$weight)

## [1] 26.63506
```

The covariance is about 26.6 cm · kg, matching R's built-in `cov`

What does covariance do?

Variance explains how data disperses around the mean, in a 1-D case

Covariance measures the relationship between two dimensions. E.g., height and weight

More than the exact value, the sign is important:

- ▶ $\text{cov}(X, Y) > 0$: both dimensions change in the same “direction”; e.g., larger height usually means higher weight
- ▶ $\text{cov}(X, Y) < 0$: both dimensions change in reverse directions; e.g., time spent on social media and performance in this class
- ▶ $\text{cov}(X, Y) = 0$: the dimensions are independent from one another; e.g., sex/gender and “intelligence”

The covariance matrix

Definition 87 (Covariance matrix)

Suppose p random variables $X_1, \dots, X_p$. Then the covariance matrix is the symmetric matrix

$$\begin{pmatrix} \text{Var } X_1 & \text{cov}(X_1, X_2) & \cdots & \text{cov}(X_1, X_p) \\ \text{cov}(X_1, X_2) & \text{Var } X_2 & \cdots & \text{cov}(X_2, X_p) \\ \vdots & \vdots & & \vdots \\ \text{cov}(X_1, X_p) & \text{cov}(X_2, X_p) & \cdots & \text{Var } X_p \end{pmatrix}$$

The actual form of the matrix is below but the form above stems from the properties of covariance

$$\begin{pmatrix} \text{cov}(X_1, X_1) & \text{cov}(X_1, X_2) & \cdots & \text{cov}(X_1, X_p) \\ \text{cov}(X_2, X_1) & \text{cov}(X_2, X_2) & \cdots & \text{cov}(X_2, X_p) \\ \vdots & \vdots & & \vdots \\ \text{cov}(X_n, X_1) & \text{cov}(X_n, X_2) & \cdots & \text{cov}(X_n, X_n) \end{pmatrix}$$

Sample covariance matrix

Let $X \in \mathcal{M}_{np}$ be the data matrix (n observations in rows, p features or variables in columns) and $\tilde{X} \in \mathcal{M}_{np}$ be the **centred data matrix**. Then the (sample) covariance matrix $S_X \in \mathcal{M}_p$ takes the form

$$S_X = \frac{1}{n-1} \tilde{X}^T \tilde{X}$$

where the data is centred!

Sometimes you will see $S_X = 1/(n-1) \tilde{X} \tilde{X}^T$. This is for matrices with observations in columns and features (variables) in rows. Just remember that you want the covariance matrix to have size the number of features (p), not observations (n), this will give you the order in which to take the product

Why is $S_X = \frac{1}{n-1} \tilde{X}^T \tilde{X}$? – Data matrix setup

Let us prove why the sample covariance matrix takes this matrix form

Suppose we have n observations of p features (variables) $X_1, \dots, X_p$

Organise the centred data into the **centred data matrix** $\tilde{X} \in \mathcal{M}_{np}$ where each row is an observation and each column is a centred feature:

$$\tilde{X} = \begin{pmatrix} \tilde{x}_{11} & \tilde{x}_{12} & \cdots & \tilde{x}_{1p} \\ \tilde{x}_{21} & \tilde{x}_{22} & \cdots & \tilde{x}_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{x}_{n1} & \tilde{x}_{n2} & \cdots & \tilde{x}_{np} \end{pmatrix} = (\tilde{\mathbf{c}}_1 \quad \tilde{\mathbf{c}}_2 \quad \cdots \quad \tilde{\mathbf{c}}_p)$$

where $\tilde{x}_{ij} = x_{ij} - \bar{x}_j$ and column $\tilde{\mathbf{c}}_j = (\tilde{x}_{1j}, \tilde{x}_{2j}, \dots, \tilde{x}_{nj})^T \in \mathbb{R}^n$ contains all n centred measurements of feature X_j

Because the data is **centred**, $\frac{1}{n} \sum_{i=1}^n \tilde{x}_{ij} = 0$ for all $j = 1, \dots, p$

Why is $S_X = \frac{1}{n-1} \tilde{X}^T \tilde{X}$? – Sample covariances

Recall the definition of the sample covariance between features X_j and X_k :

$$\text{cov}(X_j, X_k) = \frac{1}{n-1} \sum_{i=1}^n (x_{ij} - \bar{x}_j)(x_{ik} - \bar{x}_k)$$

Writing $\tilde{x}_{ij} = x_{ij} - \bar{x}_j$ and $\tilde{x}_{ik} = x_{ik} - \bar{x}_k$, this simplifies to:

$$\text{cov}(X_j, X_k) = \frac{1}{n-1} \sum_{i=1}^n \tilde{x}_{ij} \tilde{x}_{ik}$$

In particular, when $j = k$, we obtain the sample variance of feature X_j :

$$\text{Var } X_j = \text{cov}(X_j, X_j) = \frac{1}{n-1} \sum_{i=1}^n \tilde{x}_{ij}^2$$

Notice: $\sum_{i=1}^n \tilde{x}_{ij} \tilde{x}_{ik} = \tilde{\mathbf{c}}_j^T \tilde{\mathbf{c}}_k$, the dot product of column $\tilde{\mathbf{c}}_j$ and column $\tilde{\mathbf{c}}_k$!

Why is $S_X = \frac{1}{n-1} \tilde{X}^T \tilde{X}$? – Matrix product $\tilde{X}^T \tilde{X}$

Now consider the matrix product $\tilde{X}^T \tilde{X}$:

$\tilde{X}^T \in \mathcal{M}_{pn}$ has the columns of $\tilde{X}$ as its rows:

$$\tilde{X}^T = \begin{pmatrix} \tilde{\mathbf{c}}_1^T \\ \tilde{\mathbf{c}}_2^T \\ \vdots \\ \tilde{\mathbf{c}}_p^T \end{pmatrix} = \begin{pmatrix} \tilde{X}_{11} & \tilde{X}_{21} & \cdots & \tilde{X}_{n1} \\ \tilde{X}_{12} & \tilde{X}_{22} & \cdots & \tilde{X}_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{X}_{1p} & \tilde{X}_{2p} & \cdots & \tilde{X}_{np} \end{pmatrix}$$

Since $\tilde{X}^T \in \mathcal{M}_{pn}$ and $\tilde{X} \in \mathcal{M}_{np}$, the product $\tilde{X}^T \tilde{X}$ is a $p \times p$ matrix

Its (j, k) -entry is the product of row j of $\tilde{X}^T$ and column k of $\tilde{X}$:

$$(\tilde{X}^T \tilde{X})_{jk} = \tilde{\mathbf{c}}_j^T \tilde{\mathbf{c}}_k = \sum_{i=1}^n \tilde{X}_{ij} \tilde{X}_{ik}$$

Why is $S_X = \frac{1}{n-1} \tilde{X}^T \tilde{X}$? – Conclusion

Dividing by $n - 1$, each entry of $\frac{1}{n-1} \tilde{X}^T \tilde{X}$ is:

$$\left(\frac{1}{n-1} \tilde{X}^T \tilde{X} \right)_{jk} = \frac{1}{n-1} \sum_{i=1}^n \tilde{x}_{ij} \tilde{x}_{ik} = \begin{cases} \text{Var } X_j & \text{if } j = k \\ \text{cov}(X_j, X_k) & \text{if } j \neq k \end{cases}$$

Thus, writing out the matrix:

$$\frac{1}{n-1} \tilde{X}^T \tilde{X} = \begin{pmatrix} \text{Var } X_1 & \text{cov}(X_1, X_2) & \cdots & \text{cov}(X_1, X_p) \\ \text{cov}(X_2, X_1) & \text{Var } X_2 & \cdots & \text{cov}(X_2, X_p) \\ \vdots & \vdots & \ddots & \vdots \\ \text{cov}(X_p, X_1) & \text{cov}(X_p, X_2) & \cdots & \text{Var } X_p \end{pmatrix} = S_X$$

This proves that $S_X = \frac{1}{n-1} \tilde{X}^T \tilde{X}$ for the centred data matrix $\tilde{X}$

What if observations are in columns?

Suppose the centred data is stored with observations in columns and features (variables) in rows:

$$\tilde{Y} = \tilde{X}^T \in \mathcal{M}_{pn}$$

where row j of $\tilde{Y}$ is $\tilde{\mathbf{c}}_j^T = (\tilde{x}_{1j}, \dots, \tilde{x}_{nj}) \in \mathbb{R}^{1 \times n}$ (feature X_j)

Then $\tilde{Y}\tilde{Y}^T \in \mathcal{M}_p$, and its (j, k) -entry is:

$$(\tilde{Y}\tilde{Y}^T)_{jk} = \tilde{\mathbf{c}}_j^T \tilde{\mathbf{c}}_k = \sum_{i=1}^n \tilde{x}_{ij} \tilde{x}_{ik}$$

Dividing by $n - 1$:

$$S_Y = \frac{1}{n-1} \tilde{Y}\tilde{Y}^T = S_X$$

The resulting sample covariance matrix is identical

What if the data is not centred beforehand?

Suppose $X \in \mathcal{M}_{np}$ is the **uncentred** data matrix

Let $\mathbf{1} = (1, 1, \dots, 1)^T \in \mathbb{R}^n$. The vector of column means is $\bar{\mathbf{x}}^T = \frac{1}{n} \mathbf{1}^T X \in \mathbb{R}^{1 \times p}$

Subtracting the means from each row of X gives the **centred data matrix** $\tilde{X} \in \mathcal{M}_{np}$:

$$\tilde{X} = X - \mathbf{1} \bar{\mathbf{x}}^T = X - \frac{1}{n} \mathbf{1} \mathbf{1}^T X = \left(\mathbb{I}_n - \frac{1}{n} \mathbf{1} \mathbf{1}^T \right) X = C_n X$$

where $C_n = \mathbb{I}_n - \frac{1}{n} \mathbf{1} \mathbf{1}^T \in \mathcal{M}_n$ is the **centring matrix**

C_n is symmetric ($C_n^T = C_n$) and idempotent ($C_n^2 = C_n$), so:

$$S_X = \frac{1}{n-1} \tilde{X}^T \tilde{X} = \frac{1}{n-1} X^T C_n^T C_n X = \frac{1}{n-1} X^T C_n X$$

Covariance matrix of hockey players in R

When passed a data frame or matrix, `cov` computes the covariance matrix:

```
cov(data[, c("height", "weight")])  
  
##           height    weight  
## height 29.66176 26.63506  
## weight 26.63506 47.81112
```

The diagonal entries are the variances ($\text{Var}(\text{height}) \approx 29.7$, $\text{Var}(\text{weight}) \approx 47.8$) and the off-diagonal entries are $\text{cov}(\text{height}, \text{weight}) \approx 26.6$

A running example: hockey players

Change of basis

A crash course on probability

Back to PCA

Hockey players

Setting things up

I will use notation (mostly) as in Joliffe's *Principal Component Analysis* (PDF of older version available for free from UofM Libraries)

$\mathbf{x} = (x_1, \dots, x_p)$ vector of p random variables

We seek a linear function $\alpha_1^T \mathbf{x}$ with maximum variance, where $\alpha_1 = (\alpha_{11}, \dots, \alpha_{1p})$, i.e.,

$$\alpha_1^T \mathbf{x} = \sum_{j=1}^p \alpha_{1j} x_j$$

Then we seek a linear function $\alpha_2^T \mathbf{x}$ with maximum variance, uncorrelated to $\alpha_1^T \mathbf{x}$

And we continue...

At k th stage, we find a linear function $\alpha_k^T \mathbf{x}$ with maximum variance, uncorrelated to $\alpha_1^T \mathbf{x}, \dots, \alpha_{k-1}^T \mathbf{x}$

$\alpha_i^T \mathbf{x}$ is the i th **principal component** (PC)

Case of known covariance matrix

Suppose we know Σ , covariance matrix of $\mathbf{x}$ (i.e., typically: we know $\mathbf{x}$)

Then the k th PC is

$$z_k = \alpha_k^T \mathbf{x}$$

where α_k is an eigenvector of Σ corresponding to the k th largest eigenvalue λ_k

If, additionally, $\|\alpha_k\| = \alpha_k^T \alpha_k = 1$, then $\lambda_k = \text{Var } z_k$

Why is that?

Let us start with

$$\alpha_1^T \mathbf{x}$$

We want maximum variance, where $\alpha_1 = (\alpha_{11}, \dots, \alpha_{1p})$, i.e.,

$$\alpha_1^T \mathbf{x} = \sum_{j=1}^p \alpha_{1j} x_j$$

with the constraint that $\|\alpha_1\| = 1$

We have

$$\text{Var } \alpha_1^T \mathbf{x} = \alpha_1^T \Sigma \alpha_1$$

Computation of $\text{Var}(\alpha_1^T \mathbf{x})$

$$\begin{aligned}\text{Var}(\alpha_1^T \mathbf{x}) &= E \left[\left(\alpha_1^T \mathbf{x} - E(\alpha_1^T \mathbf{x}) \right)^2 \right] \\&= E \left[\left(\alpha_1^T \mathbf{x} - \alpha_1^T E(\mathbf{x}) \right)^2 \right] \\&= E \left[\left(\alpha_1^T (\mathbf{x} - E(\mathbf{x})) \right)^2 \right] \\&= E \left[\left(\alpha_1^T (\mathbf{x} - E(\mathbf{x})) \right) \left(\alpha_1^T (\mathbf{x} - E(\mathbf{x})) \right)^T \right] \\&= E \left[\alpha_1^T (\mathbf{x} - E(\mathbf{x})) (\mathbf{x} - E(\mathbf{x}))^T \alpha_1 \right] \\&= \alpha_1^T E \left[(\mathbf{x} - E(\mathbf{x})) (\mathbf{x} - E(\mathbf{x}))^T \right] \alpha_1 \\&= \alpha_1^T \Sigma \alpha_1\end{aligned}$$

Objective

We want to maximise $\text{Var } \alpha_1^T \mathbf{x}$, i.e.,

$$\alpha_1^T \Sigma \alpha_1$$

under the constraint that $\|\alpha_1\| = 1$

$\implies$ use **Lagrange multipliers**

Maximisation using Lagrange multipliers

(Recall from the mathematical preliminaries slides)

As presented in Unit 02 (*Mathematical preliminaries*), the method of **Lagrange multipliers** establishes that to find the extremum of a function $f(x_1, \dots, x_n)$ under the constraint $g(x_1, \dots, x_n) = k$:

1. Solve

$$\begin{aligned}\nabla f(x_1, \dots, x_n) &= \lambda \nabla g(x_1, \dots, x_n) \\ g(x_1, \dots, x_n) &= k\end{aligned}$$

where $\nabla = \left(\frac{\partial}{\partial x_1}, \dots, \frac{\partial}{\partial x_n} \right)$ is the **gradient operator**

2. Plug all solutions into $f(x_1, \dots, x_n)$ and find maximum values (provided values exist and $\nabla g \neq \mathbf{0}$ there)

$\lambda \in \mathbb{R}$ is the **Lagrange multiplier**

Geometrically, this corresponds to the tangency condition: gradient vectors are **parallel** ($\nabla f = \lambda \nabla g$)

Bear with me..

(You may experience a brief period of discomfort)

$\alpha_1^T \Sigma \alpha_1$ and $\|\alpha_1\|^2 = \alpha_1^T \alpha_1$ are functions of $\alpha_1 = (\alpha_{11}, \dots, \alpha_{1p})$

In the notation of the previous slide, we want the max of

$$f(\alpha_{11}, \dots, \alpha_{1p}) := \alpha_1^T \Sigma \alpha_1$$

under the constraint that

$$g(\alpha_{11}, \dots, \alpha_{1p}) := \alpha_1^T \alpha_1 = 1$$

and with gradient operator

$$\nabla = \left(\frac{\partial}{\partial \alpha_{11}}, \dots, \frac{\partial}{\partial \alpha_{1p}} \right)$$

Effect of ∇ on g

g is easiest to see:

$$\begin{aligned}\nabla g(\alpha_{11}, \dots, \alpha_{1p}) &= \left(\frac{\partial}{\partial \alpha_{11}}, \dots, \frac{\partial}{\partial \alpha_{1p}} \right) (\alpha_{11}, \dots, \alpha_{1p}) \begin{pmatrix} \alpha_{11} \\ \vdots \\ \alpha_{1p} \end{pmatrix} \\ &= \left(\frac{\partial}{\partial \alpha_{11}}, \dots, \frac{\partial}{\partial \alpha_{1p}} \right) (\alpha_{11}^2 + \dots + \alpha_{1p}^2) \\ &= (2\alpha_{11}, \dots, 2\alpha_{1p}) \\ &= 2\alpha_1\end{aligned}$$

(And that's a general result: $\nabla \|\mathbf{x}\|_2^2 = 2\mathbf{x}$ with $\|\cdot\|_2$ the Euclidean norm)

Effect of ∇ on f

Expand (write $\Sigma = [s_{ij}]$ and do not exploit symmetry)

$$\begin{aligned}\alpha_1^T \Sigma \alpha_1 &= (\alpha_{11}, \dots, \alpha_{1p}) \begin{pmatrix} s_{11} & s_{12} & \cdots & s_{1p} \\ s_{21} & s_{22} & \cdots & s_{2p} \\ \vdots & \vdots & & \vdots \\ s_{p1} & s_{p2} & & s_{pp} \end{pmatrix} \begin{pmatrix} \alpha_{11} \\ \alpha_{12} \\ \vdots \\ \alpha_{1p} \end{pmatrix} \\&= (\alpha_{11}, \dots, \alpha_{1p}) \begin{pmatrix} s_{11}\alpha_{11} + s_{12}\alpha_{12} + \cdots + s_{1p}\alpha_{1p} \\ s_{21}\alpha_{11} + s_{22}\alpha_{12} + \cdots + s_{2p}\alpha_{1p} \\ \vdots \\ s_{p1}\alpha_{11} + s_{p2}\alpha_{12} + \cdots + s_{pp}\alpha_{1p} \end{pmatrix} \\&= (s_{11}\alpha_{11} + s_{12}\alpha_{12} + \cdots + s_{1p}\alpha_{1p})\alpha_{11} \\&\quad + (s_{21}\alpha_{11} + s_{22}\alpha_{12} + \cdots + s_{2p}\alpha_{1p})\alpha_{12} \\&\quad \vdots \\&\quad + (s_{p1}\alpha_{11} + s_{p2}\alpha_{12} + \cdots + s_{pp}\alpha_{1p})\alpha_{1p}\end{aligned}$$

We have

$$\begin{aligned}\alpha_1^T \Sigma \alpha_1 &= (\mathbf{s}_{11}\alpha_{11} + \mathbf{s}_{12}\alpha_{12} + \cdots + \mathbf{s}_{1p}\alpha_{1p})\alpha_{11} \\ &\quad + (\mathbf{s}_{21}\alpha_{11} + \mathbf{s}_{22}\alpha_{12} + \cdots + \mathbf{s}_{2p}\alpha_{1p})\alpha_{12} \\ &\quad \vdots \\ &\quad + (\mathbf{s}_{p1}\alpha_{11} + \mathbf{s}_{p2}\alpha_{12} + \cdots + \mathbf{s}_{pp}\alpha_{1p})\alpha_{1p}\end{aligned}$$

$$\begin{aligned}\Rightarrow \frac{\partial}{\partial \alpha_{11}} \alpha_1^T \Sigma \alpha_1 &= (\mathbf{s}_{11}\alpha_{11} + \mathbf{s}_{12}\alpha_{12} + \cdots + \mathbf{s}_{1p}\alpha_{1p}) + \mathbf{s}_{11}\alpha_{11} \\ &\quad + \mathbf{s}_{21}\alpha_{12} + \cdots + \mathbf{s}_{p1}\alpha_{1p} \\ &= \mathbf{s}_{11}\alpha_{11} + \mathbf{s}_{12}\alpha_{12} + \cdots + \mathbf{s}_{1p}\alpha_{1p} \\ &\quad + \mathbf{s}_{11}\alpha_{11} + \mathbf{s}_{21}\alpha_{12} + \cdots + \mathbf{s}_{p1}\alpha_{1p} \\ &= 2(\mathbf{s}_{11}\alpha_{11} + \mathbf{s}_{12}\alpha_{12} + \cdots + \mathbf{s}_{1p}\alpha_{1p})\end{aligned}$$

(last equality stems from symmetry of Σ)

In general, for $i = 1, \dots, p$,

$$\begin{aligned}\frac{\partial}{\partial \alpha_{1i}} \alpha_1^T \Sigma \alpha_1 &= s_{i1} \alpha_{11} + s_{i2} \alpha_{12} + \dots + s_{ip} \alpha_{1p} \\ &\quad + s_{i1} \alpha_{11} + s_{i2} \alpha_{12} + \dots + s_{ip} \alpha_{1p} \\ &= 2(s_{i1} \alpha_{11} + s_{i2} \alpha_{12} + \dots + s_{ip} \alpha_{1p})\end{aligned}$$

(because of symmetry of Σ)

As a consequence,

$$\nabla \alpha_1^T \Sigma \alpha_1 = 2 \Sigma \alpha_1$$

So solving

$$\nabla f(x_1, \dots, x_n) = \lambda \nabla g(x_1, \dots, x_n)$$

means solving

$$2\Sigma\alpha_1 = \lambda 2\alpha_1$$

i.e.,

$$\Sigma\alpha_1 = \lambda\alpha_1$$

$\implies (\lambda, \alpha_1)$ eigenpair of Σ , with α_1 having unit length

Picking the right eigenvalue

(λ, α_1) eigenpair of Σ , with α_1 having unit length

But which λ to choose?

Recall that we want $\text{Var } \alpha_1^T \mathbf{x} = \alpha_1^T \Sigma \alpha_1$ maximal

We have

$$\text{Var } \alpha_1^T \mathbf{x} = \alpha_1^T \Sigma \alpha_1 = \alpha_1^T (\Sigma \alpha_1) = \alpha_1^T (\lambda \alpha_1) = \lambda (\alpha_1^T \alpha_1) = \lambda$$

$\implies$ we pick $\lambda = \lambda_1$, the largest eigenvalue (covariance matrix symmetric so eigenvalues real)

What we have this far..

The first principal component is $\alpha_1^T \mathbf{x}$ and has variance λ_1 , where λ_1 the largest eigenvalue of Σ and α_1 an associated eigenvector with $\|\alpha_1\| = 1$

We want the second principal component to be *uncorrelated* with $\alpha_1^T \mathbf{x}$ and to have maximum variance $\text{Var } \alpha_2^T \mathbf{x} = \alpha_2^T \Sigma \alpha_2$, under the constraint that $\|\alpha_2\| = 1$

$\alpha_2^T \mathbf{x}$ uncorrelated to $\alpha_1^T \mathbf{x}$ if $\text{cov}(\alpha_1^T \mathbf{x}, \alpha_2^T \mathbf{x}) = 0$

We have

$$\begin{aligned}\text{cov}(\alpha_1^T \mathbf{x}, \alpha_2^T \mathbf{x}) &= \alpha_1^T \Sigma \alpha_2 \\ &= \alpha_2^T \Sigma^T \alpha_1 \\ &= \alpha_2^T \Sigma \alpha_1 \quad [\Sigma \text{ symmetric}] \\ &= \alpha_2^T (\lambda_1 \alpha_1) \\ &= \lambda \alpha_2^T \alpha_1\end{aligned}$$

So $\alpha_2^T \mathbf{x}$ uncorrelated to $\alpha_1^T \mathbf{x}$ if $\alpha_1 \perp \alpha_2$

This is beginning to sound a lot like Gram-Schmidt, no?

In short

Take whatever covariance matrix is available to you (known Σ or sample S_X) – assume sample from now on for simplicity

For $i = 1, \dots, p$, the i th principal component is

$$z_i = \mathbf{v}_i^T \mathbf{x}$$

where $\mathbf{v}_i$ eigenvector of S_X associated to the i th largest eigenvalue λ_i

If $\mathbf{v}_i$ is normalised, then $\lambda_i = \text{Var } z_i$

A 3D example – Sample covariance matrix

Suppose we have a dataset with $p = 3$ variables, $\mathbf{x} = (x_1, x_2, x_3)^T$, centred so that sample means are zero

After computing all sample variances and covariances, we obtain the sample covariance matrix:

$$S_X = \begin{pmatrix} 15 & 10 & 4 \\ 10 & 14 & 6 \\ 4 & 6 & 7 \end{pmatrix}$$

- ▶ Sample variances (diagonal): $\text{Var}(x_1) = 15$, $\text{Var}(x_2) = 14$, $\text{Var}(x_3) = 7$
- ▶ Sample covariances (off-diagonal): all variables are positively correlated ($\text{cov}(x_1, x_2) = 10$, $\text{cov}(x_1, x_3) = 4$, $\text{cov}(x_2, x_3) = 6$)
- ▶ Total sample variance in the dataset is the trace:

$$\text{tr}(S_X) = 15 + 14 + 7 = 36$$

A 3D example – Eigenvalues

To compute the principal components, we find the eigenvalues of S_X by solving $\det(S_X - \lambda I) = 0$:

$$\det \begin{pmatrix} 15 - \lambda & 10 & 4 \\ 10 & 14 - \lambda & 6 \\ 4 & 6 & 7 - \lambda \end{pmatrix} = 0$$

This yields three distinct positive eigenvalues:

$$\lambda_1 = 27, \quad \lambda_2 = 6, \quad \lambda_3 = 3$$

Note that:

$$\lambda_1 + \lambda_2 + \lambda_3 = 27 + 6 + 3 = 36 = \text{tr}(S_X)$$

As expected from theory, the total variance is preserved!

A 3D example – Variance explained

The variance of each principal component $z_i = \mathbf{v}_i^T \mathbf{x}$ is given by $\text{Var } z_i = \lambda_i$ (for normalized $\mathbf{v}_i$)

Proportion of total variance explained by each principal component:

► **1st PC (z_1):**

$$\frac{\lambda_1}{\text{tr}(\mathbf{S}_X)} = \frac{27}{36} = \frac{3}{4} = 75\%$$

► **2nd PC (z_2):**

$$\frac{\lambda_2}{\text{tr}(\mathbf{S}_X)} = \frac{6}{36} = \frac{1}{6} \approx 16.67\%$$

► **3rd PC (z_3):**

$$\frac{\lambda_3}{\text{tr}(\mathbf{S}_X)} = \frac{3}{36} = \frac{1}{12} \approx 8.33\%$$

The first two principal components together account for $\frac{27+6}{36} = \frac{33}{36} \approx 91.67\%$ of the total variance

A 3D example – Eigenvectors

Next, we find the associated normalized eigenvectors by solving $(S_X - \lambda_i I)\mathbf{v}_i = \mathbf{0}$ subject to $\|\mathbf{v}_i\| = 1$:

► For $\lambda_1 = 27$:

$$\mathbf{v}_1 = \frac{1}{3} \begin{pmatrix} 2 \\ 2 \\ 1 \end{pmatrix} = \begin{pmatrix} 2/3 \\ 2/3 \\ 1/3 \end{pmatrix}, \quad \|\mathbf{v}_1\|^2 = \frac{4 + 4 + 1}{9} = 1$$

► For $\lambda_2 = 6$:

$$\mathbf{v}_2 = \frac{1}{3} \begin{pmatrix} -2 \\ 1 \\ 2 \end{pmatrix} = \begin{pmatrix} -2/3 \\ 1/3 \\ 2/3 \end{pmatrix}, \quad \|\mathbf{v}_2\|^2 = \frac{4 + 1 + 4}{9} = 1$$

► For $\lambda_3 = 3$:

$$\mathbf{v}_3 = \frac{1}{3} \begin{pmatrix} 1 \\ -2 \\ 2 \end{pmatrix} = \begin{pmatrix} 1/3 \\ -2/3 \\ 2/3 \end{pmatrix}, \quad \|\mathbf{v}_3\|^2 = \frac{1 + 4 + 4}{9} = 1$$

A 3D example – Orthonormality check

Since S_X is symmetric with distinct eigenvalues, the eigenvectors are mutually orthogonal:

$$\mathbf{v}_1^T \mathbf{v}_2 = \frac{1}{9}(2(-2) + 2(1) + 1(2)) = \frac{-4 + 2 + 2}{9} = 0$$

$$\mathbf{v}_1^T \mathbf{v}_3 = \frac{1}{9}(2(1) + 2(-2) + 1(2)) = \frac{2 - 4 + 2}{9} = 0$$

$$\mathbf{v}_2^T \mathbf{v}_3 = \frac{1}{9}(-2(1) + 1(-2) + 2(2)) = \frac{-2 - 2 + 4}{9} = 0$$

Hence, $V = (\mathbf{v}_1 \quad \mathbf{v}_2 \quad \mathbf{v}_3)$ is an orthogonal matrix ($V^T V = I$), forming an orthonormal basis of $\mathbb{R}^3$

A 3D example – The principal components

The principal components are the linear combinations $z_i = \mathbf{v}_i^T \mathbf{x}$:

$$z_1 = \mathbf{v}_1^T \mathbf{x} = \frac{2}{3}x_1 + \frac{2}{3}x_2 + \frac{1}{3}x_3$$

$$z_2 = \mathbf{v}_2^T \mathbf{x} = -\frac{2}{3}x_1 + \frac{1}{3}x_2 + \frac{2}{3}x_3$$

$$z_3 = \mathbf{v}_3^T \mathbf{x} = \frac{1}{3}x_1 - \frac{2}{3}x_2 + \frac{2}{3}x_3$$

Properties of these new variables:

- ▶ $\text{Var } z_1 = \lambda_1 = 27$, $\text{Var } z_2 = \lambda_2 = 6$, $\text{Var } z_3 = \lambda_3 = 3$
- ▶ $\text{cov}(z_i, z_j) = 0$ for all $i \neq j$ (uncorrelated)
- ▶ In the new basis, the covariance matrix is diagonal:

$$S_Z = V^T S_X V = \text{diag}(27, 6, 3)$$

A 3D example – Dimensionality reduction

Suppose we want to reduce the dimension of the data from 3 to 2:

- ▶ Keep only the first two components, z_1 and z_2
- ▶ z_1 and z_2 together preserve 91.67% of the total variance
- ▶ Each 3D data point $\mathbf{x} = (x_1, x_2, x_3)^T$ is mapped to its 2D coordinates $(z_1, z_2)^T$
- ▶ Approximation of $\mathbf{x}$ using only 2 components:

$$\mathbf{x} \approx z_1 \mathbf{v}_1 + z_2 \mathbf{v}_2$$

- ▶ Discarding z_3 loses only 8.33% of the variance

A 3D example – Verifying in R

We can confirm our calculations directly in R:

```
S = matrix(c(15, 10, 4, 10, 14, 6, 4, 6, 7), nrow = 3, byrow = TRUE)
ev = eigen(S)
ev$values

## [1] 27 6 3

# Proportion and cumulative proportion of variance explained
ev$values / sum(ev$values)

## [1] 0.7500000 0.1666667 0.08333333

cumsum(ev$values) / sum(ev$values)

## [1] 0.7500000 0.9166667 1.0000000
```



A running example: hockey players

Change of basis

A crash course on probability

Back to PCA

Hockey players

Covariance of height and weight

The function `cov` returns the sample covariance of two variables

Recall that `cov` yields the same result whether the data is centred or uncentred:

```
cov(data$height, data$weight)

## [1] 26.63506

cov(data$height.c, data$weight.c)

## [1] 26.63506
```

As seen in the scatter plot, there is a positive linear relationship between height and weight

Sample covariance matrix

Let us form the $n \times 2$ centred data matrix $\tilde{X}$ and compute the sample covariance matrix $S = \frac{1}{n-1} \tilde{X}^T \tilde{X}$

```
X = as.matrix(data[, c("height.c", "weight.c")])
S = 1 / (dim(X)[1] - 1) * t(X) %*% X
S

##           height.c weight.c
## height.c 29.66176 26.63506
## weight.c 26.63506 47.81112
```

The off-diagonal entries match $\text{cov}(X_1, X_2)$

Sample covariance matrix – Variances

Let us check that the diagonal entries indeed match the individual sample variances:

```
var(X[, 1])  
## [1] 29.66176  
  
var(X[, 2])  
## [1] 47.81112
```

Both match perfectly

Computing the principal components

We compute the eigenvalues and eigenvectors of S using `eigen`

```
ev = eigen(S)
ev

## eigen() decomposition
## $values
## [1] 66.87496 10.59793
##
## $vectors
##           [,1]      [,2]
## [1,] 0.5820222 -0.8131729
## [2,] 0.8131729  0.5820222
```

`eigen` returns eigenvalues in decreasing order and orthonormal eigenvectors

Variance explained

The total variance is

$$\text{tr}(S) = \lambda_1 + \lambda_2 = 66.87 + 10.6 = 77.47$$

The first principal component explains

$$\frac{\lambda_1}{\lambda_1 + \lambda_2} = \frac{66.87}{77.47} \approx 86.3\%$$

of the total variance in the height and weight data

The second principal component accounts for the remaining 13.7%

Plotting the first principal component

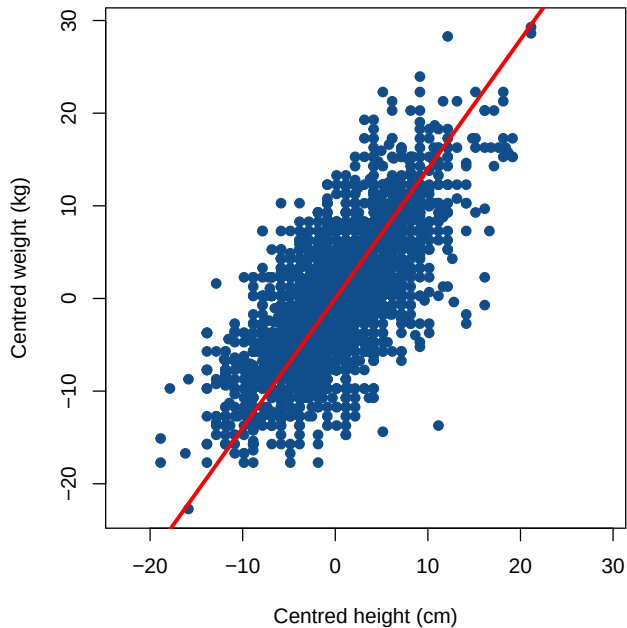
We superimpose the line spanned by the first eigenvector $\mathbf{v}_1$ on the centred data

The line passes through (0, 0) and has slope $b = v_{21}/v_{11}$

To avoid visual distortion of angles, we use a square plot (pty = "s", asp = 1) with equal ranges on both axes

```
par(pty = "s")
lims = range(c(data$height.c, data$weight.c))
plot(data$height.c, data$weight.c,
      xlim = lims, ylim = lims, asp = 1,
      pch = 19, col = "dodgerblue4",
      main = "IIHF players 2001-2016 (with first component)",
      xlab = "Centred height (cm)", ylab = "Centred weight (kg)")
abline(a = 0, b = ev$eigenvectors[2, 1] / ev$eigenvectors[1, 1],
       col = "red", lwd = 3)
```

IHF players 2001–2016 (with first component)



First principal component vs. least squares

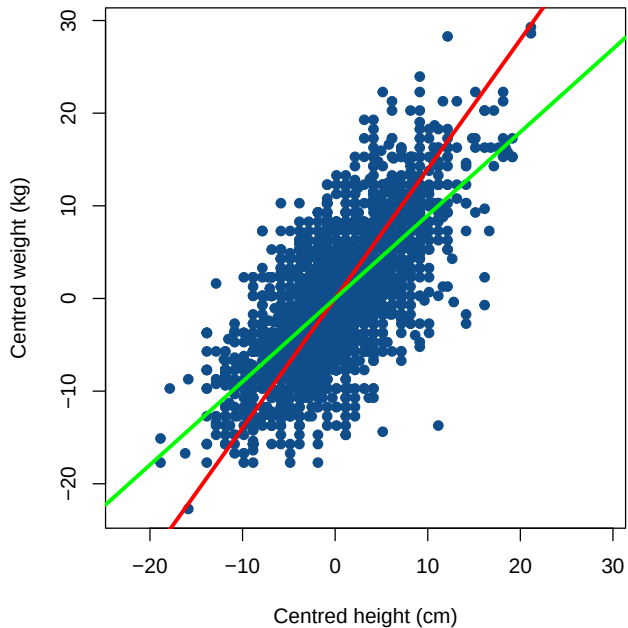
How does the first principal component line compare with ordinary least squares (OLS) regression?

Fit a linear model with zero intercept (since data is centred):

```
z = lm(weight.c ~ 0 + height.c, data = data)
coef(z)

## height.c
## 0.8979593
```

First PC (red) vs. least squares fit (green)



Why are the two lines different?

Ordinary least squares (OLS):

- ▶ Treats `height.c` as an exact predictor
- ▶ Minimises the sum of squared **vertical** residuals (errors parallel to the y -axis)

Principal component analysis (PCA):

- ▶ Treats both variables **symmetrically**
- ▶ Maximises the variance of the orthogonal projection
- ▶ Equivalently, minimises the sum of squared **orthogonal (perpendicular)** distances from points to the line

Rotating the data

Let us rotate the data so that the first principal component becomes the horizontal axis

A counter-clockwise rotation by an angle ϕ in $\mathbb{R}^2$ is represented by

$$R_\phi = \begin{pmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{pmatrix}$$

The first eigenvector makes an angle $\theta > 0$ with the positive x -axis, $\tan \theta = v_{21}/v_{11}$

To align it with the horizontal axis, we rotate **clockwise** by θ , i.e., by $\phi = -\theta$

Computing the rotation matrix

```
theta = atan(ev$vector[2, 1] / ev$vector[1, 1])
theta

## [1] 0.949583

R_minus_theta = matrix(c(cos(-theta), -sin(-theta),
                          sin(-theta),  cos(-theta)),
                        nr = 2, byrow = TRUE)

R_minus_theta

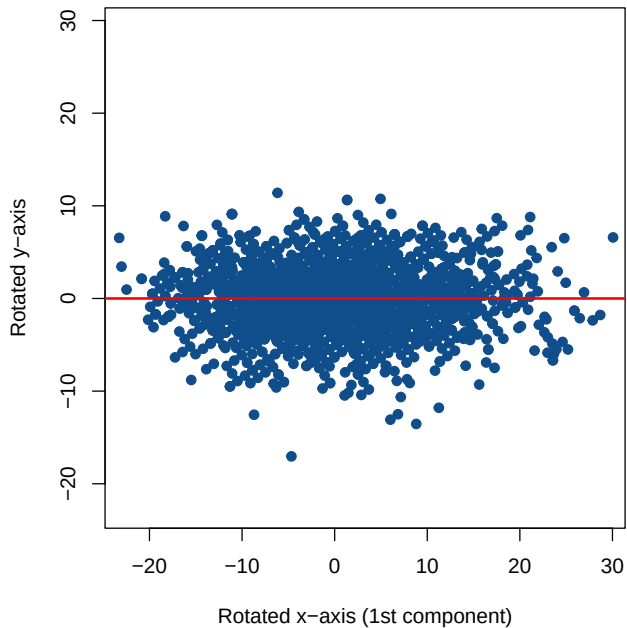
##           [,1]      [,2]
## [1,]  0.5820222  0.8131729
## [2,] -0.8131729  0.5820222
```

Rotating the data

Apply the rotation matrix to each observation:

```
rot_pts = X %*% t(R_minus_theta)
data$height.c_r = rot_pts[, 1]
data$weight.c_r = rot_pts[, 2]
plot(data$height.c_r, data$weight.c_r,
      xlim = lims, ylim = lims, asp = 1,
      pch = 19, col = "dodgerblue4",
      xlab = "Rotated x-axis (1st component)",
      ylab = "Rotated y-axis",
      main = "IIHF players 2001-2016 (rotated to first component)")
abline(h = 0, col = "red", lwd = 2)
```

IIHF players 2001–2016 (rotated to first component)

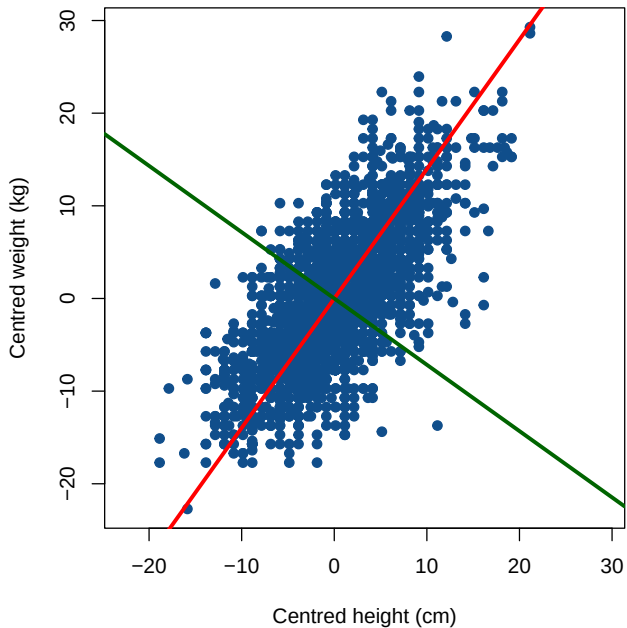


First and second principal components

Plot both eigenvectors on the centred data. Since S is symmetric, the eigenvectors are orthogonal (90° angle):

```
plot(data$height.c, data$weight.c,  
      xlim = lims, ylim = lims, asp = 1,  
      pch = 19, col = "dodgerblue4",  
      main = "IIHF players 2001-2016 (1st and 2nd eigenvectors)",  
      xlab = "Centred height (cm)", ylab = "Centred weight (kg)")  
abline(a = 0, b = ev$vectors[2, 1] / ev$vectors[1, 1],  
       col = "red", lwd = 3)  
abline(a = 0, b = ev$vectors[2, 2] / ev$vectors[1, 2],  
       col = "darkgreen", lwd = 3)
```

IHF players 2001–2016 (1st and 2nd eigenvectors)



Representing data in the new coordinate system

Rather than 2D trigonometry, express data in the orthonormal eigenbasis using our general change-of-basis machinery

Enforce orthonormality with `pracma::gramSchmidt` and compute the change-of-basis matrix using `pracma::rref`:

```
GS = pracma::gramSchmidt(A = ev$vectors, tol = 1e-10)
A = cbind(GS$Q, diag(2))
pracma::rref(A)
```

```
##      [,1] [,2]      [,3]      [,4]
## [1,]    1    0  0.5820222  0.8131729
## [2,]    0    1 -0.8131729  0.5820222
```

$P = Q^{-1} = Q^T$ is the change-of-basis matrix (right-hand block)

Projecting into the new basis

Multiplying $\tilde{X}$ by P^T gives coordinates X_{new} in the new PC coordinate system:

```
P = pracma::rref(A)[, c(3, 4)]
```

```
P
```

```
##           [,1]      [,2]
```

```
## [1,]  0.5820222 0.8131729
```

```
## [2,] -0.8131729 0.5820222
```

```
X.new = X %*% t(P)
```

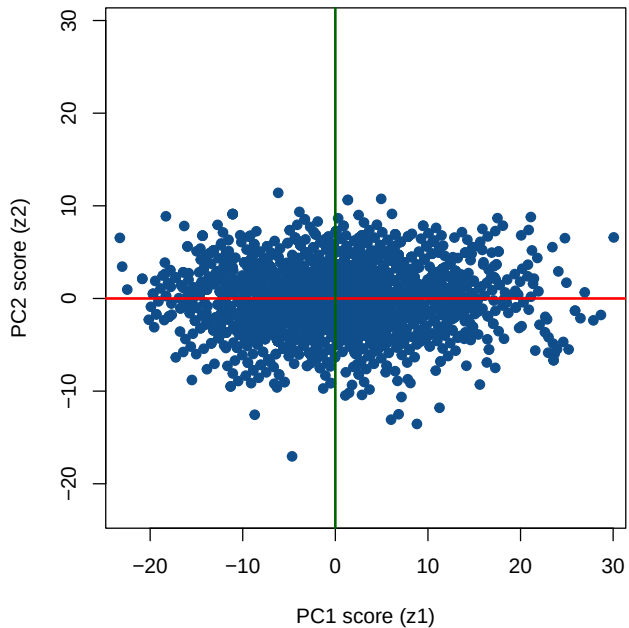
```
cov(X.new)
```

```
##           [,1]      [,2]
```

```
## [1,] 6.687496e+01 -2.328994e-13
```

```
## [2,] -2.328994e-13 1.059793e+01
```

IHF players in principal component coordinates



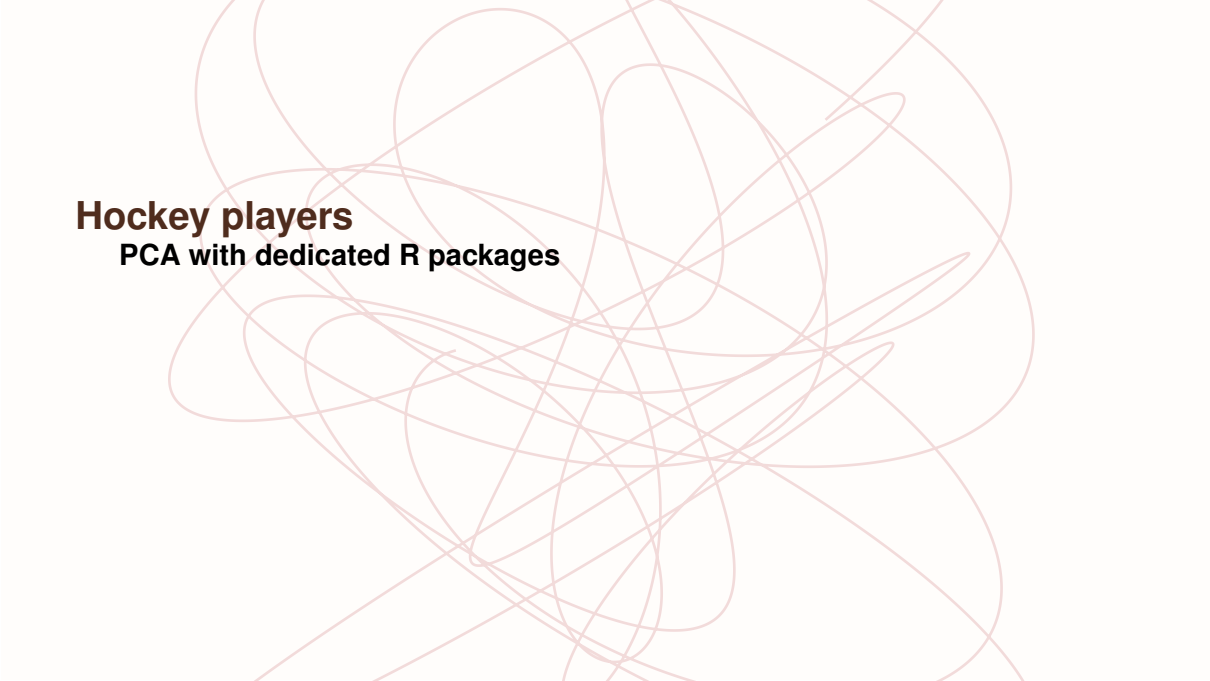
Diagonal covariance in the new basis

As confirmed by `cov(X.new)`:

```
cov(X.new)

##           [,1]           [,2]
## [1,]  6.687496e+01 -2.328994e-13
## [2,] -2.328994e-13  1.059793e+01
```

- ▶ The sample covariance matrix in the new coordinate system is **diagonal**
- ▶ Diagonal entries equal $\lambda_1 = 66.87$ and $\lambda_2 = 10.6$
- ▶ Off-diagonal entry is zero: the new coordinates are completely **uncorrelated**

The background of the slide is white, featuring a series of overlapping, hand-drawn red ellipses of various sizes and orientations. These lines create a complex, abstract pattern that covers most of the slide area.

Hockey players

PCA with dedicated R packages

Multivariate hockey dataset

In practice, we use dedicated packages such as `prcomp`, `FactoMineR` and `ggbiplot`, especially with > 2 variables

We retain 4 numerical attributes (`height`, `weight`, `age`, `bmi`) and player position (`Defence`, `Forward`, `Goalkeeper`):

```
agg_num = aggregate(cbind(height, weight, age, bmi) ~ name,  
                    data = raw_data, FUN = mean)  
agg_pos = aggregate(position ~ name, data = raw_data,  
                    FUN = function(x) x[1])  
hockey_multi = merge(agg_pos, agg_num, by = "name")  
hockey_multi$position = factor(  
  hockey_multi$position,  
  levels = c("D", "F", "G"),  
  labels = c("Defence", "Forward", "Goalkeeper")  
)
```

Multivariate hockey dataset

The aggregated multivariate dataset:

```
head(hockey_multi, n = 4)
```

##		name	position	height	weight	age	bmi
## 1		aagaard mikkel	Forward	180	81	20.53662	25.00000
## 2		aalto antti	Forward	187	96	27.15948	27.45289
## 3		aaltonen juhamatti	Forward	184	87	27.15606	25.69707
## 4		abdelkader justin	Forward	185	97	26.17933	28.34186

PCA using prcomp

Because variables are in different physical units (cm, kg, years, kg/m²), we standardise to unit variance (`scale. = TRUE`):

```
hockey.pca = prcomp(  
  hockey_multi[, c("height", "weight", "age", "bmi")],  
  center = TRUE, scale. = TRUE  
)  
summary(hockey.pca)
```

Importance of components:

##	PC1	PC2	PC3	PC4
## Standard deviation	1.3933	1.1044	0.9154	0.03081
## Proportion of Variance	0.4853	0.3049	0.2095	0.00024
## Cumulative Proportion	0.4853	0.7903	0.9998	1.00000

Importance of components

- ▶ PC1 (48.5%) and PC2 (30.5%) explain 79.0% of the total variance across the four standardised variables
- ▶ The first three components capture 99.98% of the total variance
- ▶ PC4 accounts for almost zero variance (0.02%):

$$\text{BMI} = \frac{\text{weight}}{\text{height}^2}$$

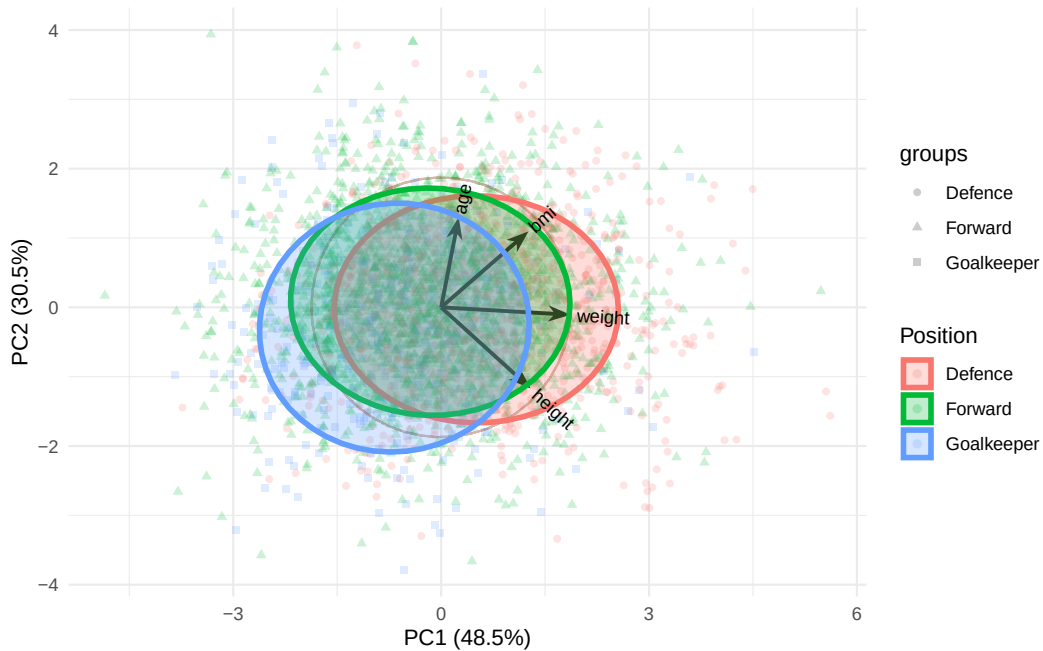
Around the sample mean, BMI is nearly a linear combination of `height` and `weight`, so the 4D point cloud is practically three-dimensional!

PCA biplot with ggbiplot

The ggbiplot package overlays projected observations and variable loadings (arrows) with confidence ellipses:

```
library(ggbiplot)
p_biplot = ggbiplot(hockey.pca, obs.scale = 1, var.scale = 1,
                    groups = hockey_multi$position, ellipse = TRUE,
                    circle = TRUE, alpha = 0.2) +
  theme_minimal() +
  labs(title = "PCA biplot of IIHF players by position",
       color = "Position", fill = "Position")
print(p_biplot)
```

PCA biplot of IIHF players by position



Interpreting the biplot

Variable loadings (arrows):

- ▶ `weight` points along positive PC1 (overall body size)
- ▶ PC2 contrasts `age` and `bmi` (pointing upward) against `height` (pointing downward)

Group ellipses (positions):

- ▶ Defencemen tend to have higher PC1 scores (taller and heavier on average)
- ▶ Forwards and goalkeepers cluster at lower PC1 scores

Multivariate analysis with FactoMineR

FactoMineR::PCA supports supplementary qualitative variables (`quali.sup = 1` for position):

```
library(FactoMineR)
res.pca = PCA(hockey_multi[, c("position", "height", "weight",
                               "age", "bmi")],
              quali.sup = 1, scale.unit = TRUE, graph = FALSE)
```

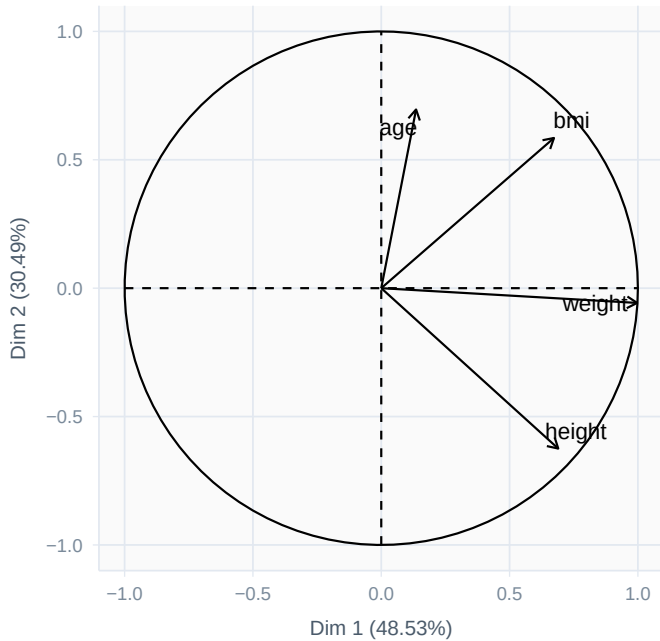
Provides two key diagnostic visualisations:

1. **Variable correlation circle:** correlations with PC1 and PC2 inside the unit circle
2. **Factor map with group ellipses:** observations and barycentres by position

Variable correlation circle (FactoMineR)

```
plot.PCA(res.pca, axes = c(1, 2), choix = "var",  
          title = "Variable correlation circle (FactoMineR)")
```

Variable correlation circle (FactoMineR)



Position ellipses and barycentres (FactoMineR)

```
plotellipses(res.pca, keepvar = 1, label = "quali",  
             title = "Individuals and position barycentres")
```

Individuals and position barycentres

