

# Principal Components Analysis (part I)

Predictive Modeling & Statistical Learning

Gaston Sanchez

CC BY-SA 4.0

# Introduction

# NBA Team Stats

- ▶ NBA Team Stats: regular season (2016-17)
- ▶ Github file: `data/nba-teams-2017.csv`
- ▶ Source: **stats.nba.com**
- ▶ `http://stats.nba.com/teams/traditional/#!  
?sort=GP&dir=-1`

SEASON  
2016-17

SEASON TYPE  
Regular Season

PER MODE  
Per Game

SEASON SEGMENT  
All Games

[Advanced Filters](#)

RECENT FILTERS

GLOSSARY

SHARE

TEAM	GP	W	L	WIN%	MIN	PTS	FGM	FGA	FG%	3PM	3PA	3P%	FTM	FTA	FT%	OREB	DREB	REB	AST	TOV	STL	BLK	BLKA	PF	PFD	+/-
1 Miami Heat	82	41	41	.500	48.2	103.2	39.0	85.8	45.5	9.9	27.0	36.5	15.2	21.6	70.6	10.6	33.0	43.6	21.2	13.4	7.2	5.7	4.9	20.5	18.7	1.1
1 Atlanta Hawks	82	43	39	.524	48.5	103.2	38.1	84.4	45.1	8.9	26.1	34.1	18.1	24.9	72.8	10.3	34.1	44.3	23.6	15.8	8.2	4.8	5.2	18.2	21.6	-0.9
1 Brooklyn Nets	82	20	62	.244	48.2	105.8	37.8	85.2	44.4	10.7	31.6	33.8	19.4	24.6	78.8	8.8	35.1	43.9	21.4	16.5	7.2	4.7	5.6	21.0	20.4	-6.7
1 Charlotte Hornets	82	36	46	.439	48.4	104.9	37.7	85.4	44.2	10.0	28.6	35.1	19.4	23.8	81.5	8.8	34.8	43.6	23.1	11.5	7.0	4.8	5.5	16.6	19.9	0.2
1 Chicago Bulls	82	41	41	.500	48.2	102.9	38.6	87.1	44.4	7.6	22.3	34.0	18.0	22.5	79.8	12.2	34.1	46.3	22.6	13.6	7.8	4.8	4.6	17.7	18.8	0.4
1 Cleveland Cavaliers	82	51	31	.622	48.5	110.3	39.9	84.9	47.0	13.0	33.9	38.4	17.5	23.3	74.8	9.3	34.4	43.7	22.7	13.7	6.6	4.0	4.3	18.1	20.6	3.2
1 Dallas Mavericks	82	33	49	.402	48.2	97.9	36.2	82.3	44.0	10.7	30.2	35.5	14.8	18.5	80.1	7.9	30.7	38.6	20.8	11.9	7.5	3.7	3.4	19.1	19.4	-2.9
1 Denver Nuggets	82	40	42	.488	48.2	111.7	41.2	87.7	46.9	10.6	28.8	36.8	18.7	24.2	77.4	11.8	34.6	46.4	25.3	15.0	6.9	3.9	4.9	19.1	20.2	0.5
1 Detroit Pistons	82	37	45	.451	48.3	101.3	39.9	88.8	44.9	7.7	23.4	33.0	13.9	19.3	71.9	11.1	34.6	45.7	21.1	11.9	7.0	3.8	4.1	17.9	17.5	-1.1
1 Golden State Warriors	82	67	15	.817	48.2	115.9	43.1	87.1	49.5	12.0	31.2	38.3	17.8	22.6	78.8	9.4	35.0	44.4	30.4	14.8	9.6	6.8	3.8	19.3	19.4	11.6

```
# variables
dat <- read.csv('data/nba-teams-2017.csv')
```

```
dim(dat)
```

```
[1] 30 27
```

```
names(dat)
```

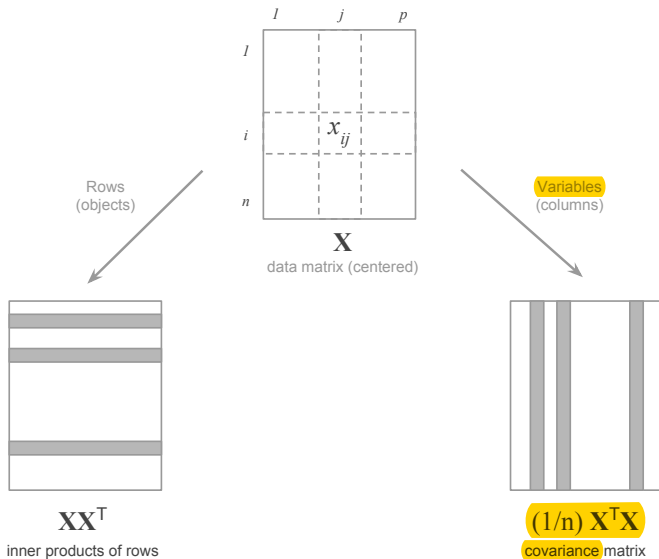
[1]	"team"	"games_played"	"wins"
[4]	"losses"	"win_prop"	"minutes"
[7]	"points"	"field_goals"	"field_goals_attempted"
[10]	"field_goals_prop"	"points3"	"points3_attempted"
[13]	"points3_prop"	"free_throws"	"free_throws_att"
[16]	"free_throws_prop"	"off_rebounds"	"def_rebounds"
[19]	"rebounds"	"assists"	"turnovers"
[22]	"steals"	"blocks"	"block_fga"
[25]	"personal_fouls"	"personal_fouls_drawn"	"plus_minus"

# Exploratory Data Analysis

For illustration purposes, let's focus on the following variables:

- ▶ wins
- ▶ losses
- ▶ points
- ▶ field\_goals
- ▶ assists
- ▶ turnovers
- ▶ steals
- ▶ blocks

# EDA: Objects and Variables Perspectives



# EDA: Objects and Variables Perspectives

## Data Perspectives

We are interested in analyzing a data set from both perspectives: **objects** and **variables**

At its simplest we are interested in 2 fundamental purposes:

- ▶ Study **resemblance** among **individuals**  
(resemblance among NBA teams)
- ▶ Study **relationship** among **variables**  
(relationship among team statistics)



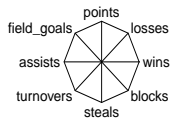
# EDA

## Exploration

Likewise, we can explore variables at different stages:

- ▶ Univariate: one variable at a time
- ▶ Bivariate: two variables simultaneously
- ▶ Multivariate: multiple variables

Let's see a shiny-app demo (see apps/ folder in github repo)



Warriors



Spurs



Rockets



Celtics



Jazz



Raptors



Cavaliers



Clippers



Wizards



Thunder



Grizzlies



Hawks



Pacers



Bucks



Bulls



Blazers



Heat



Nuggets



Pistons



Hornets



Pelicans



Mavericks



Kings



Timberwolves



Knicks



Magic



76ers



Lakers

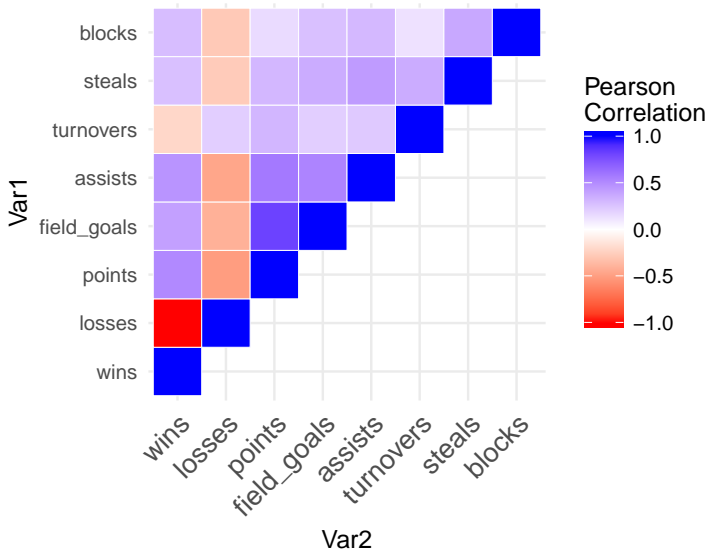


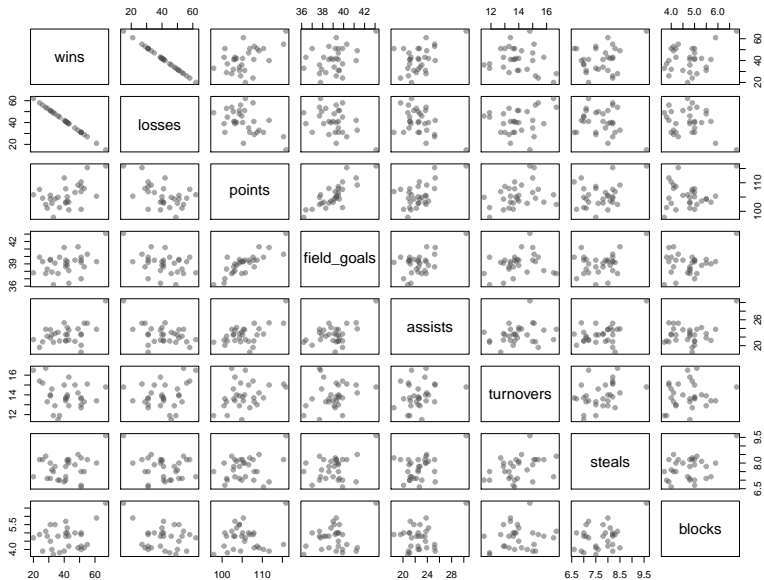
Suns



Nets

# Correlation heatmap

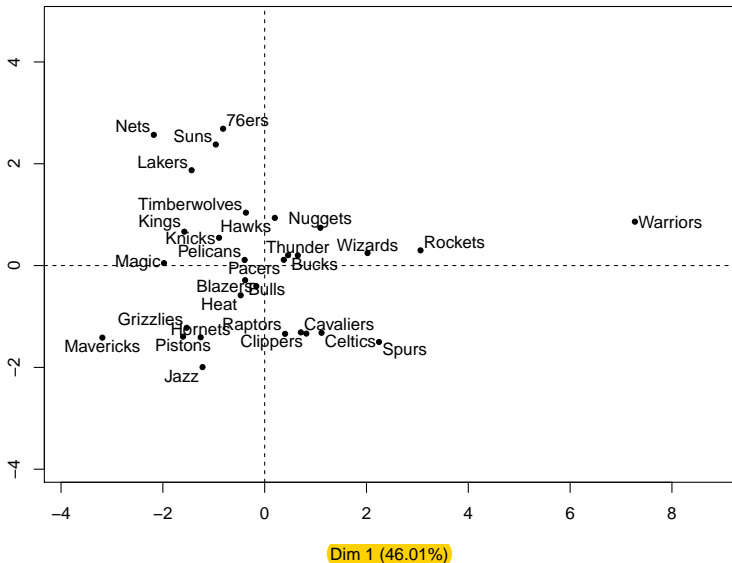


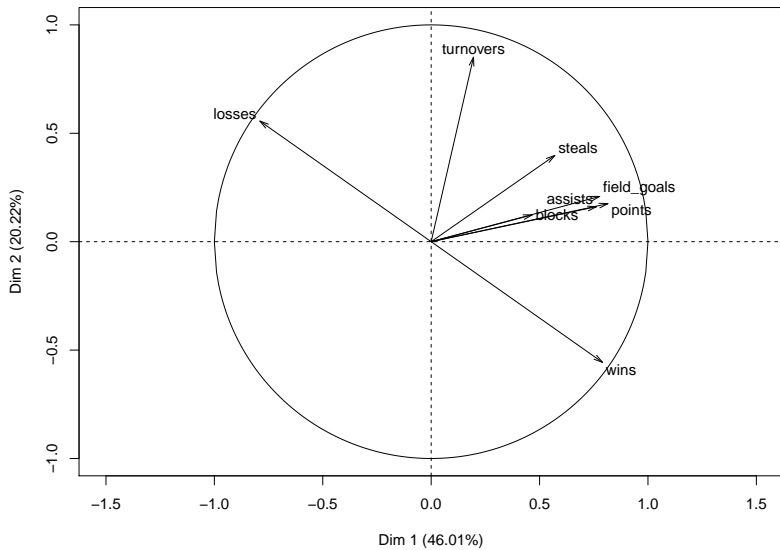


*What if we could get a better  
low-dimensional summary of the data?*



Dim 2 (20.22%)





# About PCA



# Data Structure

**Principal Components Analysis (PCA)** is a **multivariate method that** allows us **to study and explore** a set of **quantitative variables** measured on some objects.

# Landmarks

- ▶ PCA was first introduced by Karl Pearson (1904)  
*On lines and planes of closest fit to systems of points in space*
- ▶ Further developed by Harold Hotelling (1933)  
*Analysis of a complex of statistical variables into principal components*
- ▶ Singular Value Decomposition (SVD) theorem by Eckart-Young (1936)  
*The approximation of a matrix by another of a lower rank*
- ▶ Computationally implemented in the 1960s

## Core Idea

With PCA we seek to **reduce the dimensionality** (condense information in variables) of a data set while retaining as much as possible of the **variation** present in the data

# PCA: Overall Goals

- ▶ Summarize a data set with the help of a small number of **synthetic variables** (i.e. the Principal Components).
- ▶ Visualize the position (resemblance) of **individuals**.
- ▶ Visualize how **variables** are correlated.
- ▶ Interpret the synthetic variables.

# Applications

PCA can be used for

1. Dimension Reduction
2. Visualization
3. Feature Extraction
4. Data Compression
5. Smoothing of Data
6. Detection of Outliers
7. Preliminary process for further analyses

# About PCA

## Approaches:

PCA can be presented using various—different but equivalent—approaches. Each approach corresponds to a unique perspective and a way of thinking about data.

- ▶ Data dispersion from the individuals standpoint
- ▶ Data variability from the variables standpoint
- ▶ Data that follows a decomposition model

I will present PCA by mixing and connecting all of these approaches.

# Geometric Approach

# Geometric mindset

## PCA for Data Visualization

One way to present PCA is based on a data visualization approach.

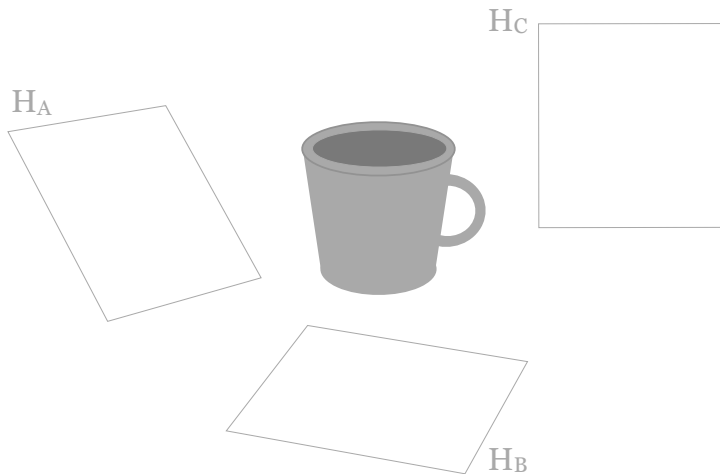
To help you understand the main idea of PCA from a geometric standpoint, I'd like to begin showing you my *mug-data* example.



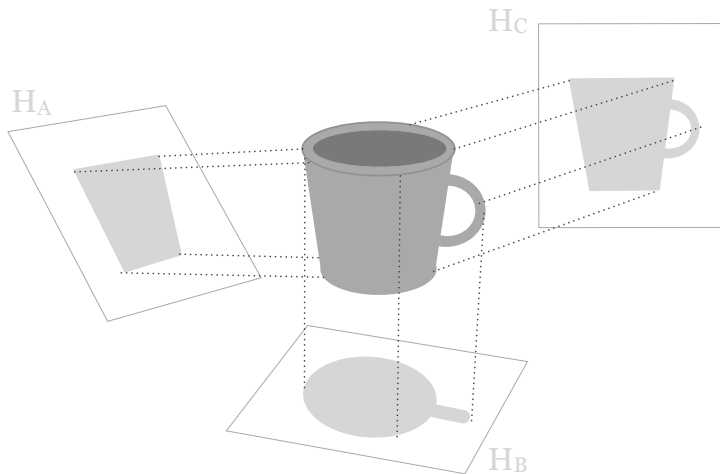
Imagine a data set in a "high-dimensional space"



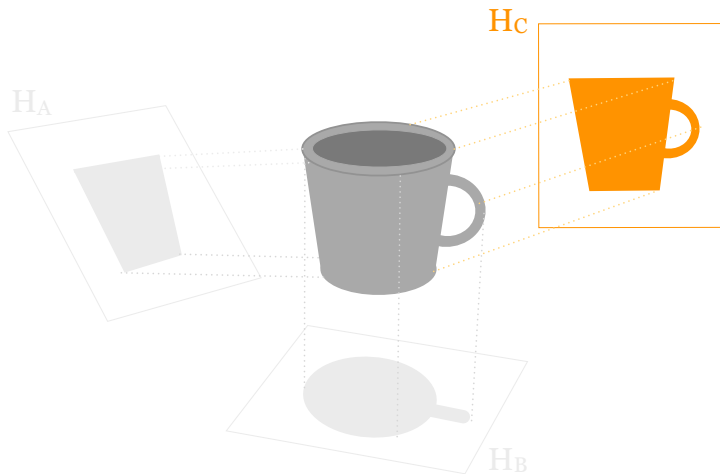
# We are looking for Candidate Subspaces



with the best low-dimensional representation



# Best low-dimensional projection



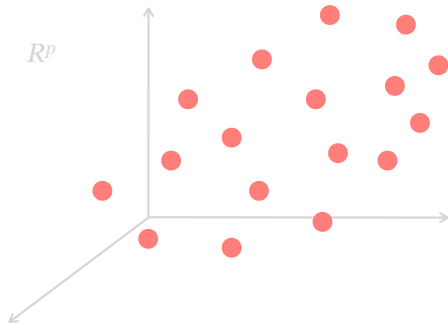
# Geometric Idea

## Looking at the cloud of points

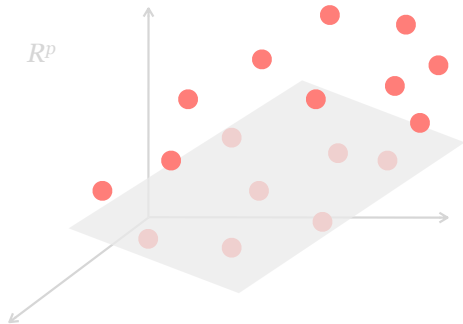
Under a purely geometric approach, PCA aims to represent the cloud of points in a space with reduced dimensionality in an “optimal” way.

We look for the “best” graphical representation that allows us to visualize the cloud of individuals in a low dimensional space (usually 2-dimensions).

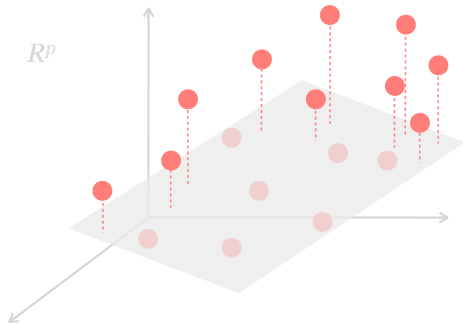
# Objects in a high-dimensional space



We look for a subspace such that

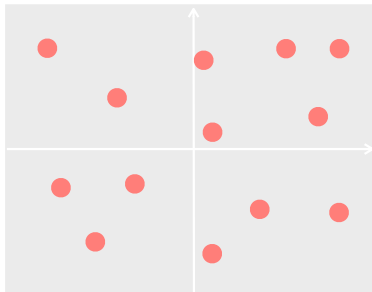


the projection of points on it





is the best low-dimensional representation



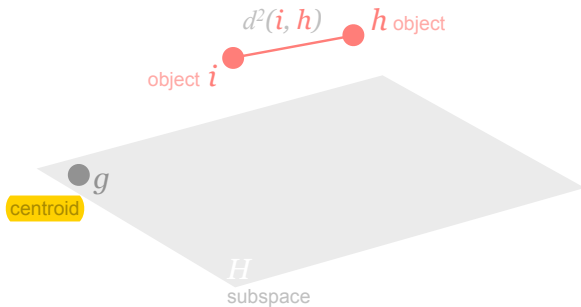
How do you find the associated axes?

# Focus on Distances

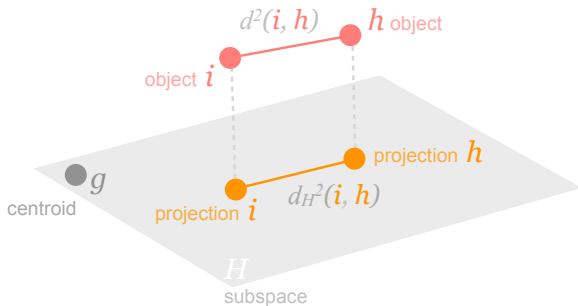
## Distances between individuals

Looking for the best low-dimensional projection means that we want to find a subspace in which the projected distances among points are as much similar as possible to the original distances.

# Focus on distances between objects



We want projected dists to preserve original dists



$d^2(i, h)$  as close as possible to  $d_{H^2}(i, h)$

# Focus on projected distances

The idea is to project the cloud of points on a plane (or a low-dim space) of  $\mathbb{R}^p$ , chosen in such a manner as to minimize distorting the distances between individuals as little as possible.

# Distances and Dispersion

## Dispersion of Data

Focusing on **distances** among all pairs of objects implicitly entails taking into account the **dispersion** or spread (i.e. variation) of the data.

## Data Configuration

The reason to pay attention to distances and dispersion is to summarize in a quantitative way the original configuration of the data points.

# How to measure dispersion?

## The concept of Inertia

# Sum of Square Distances

## Pair-wise Square distances

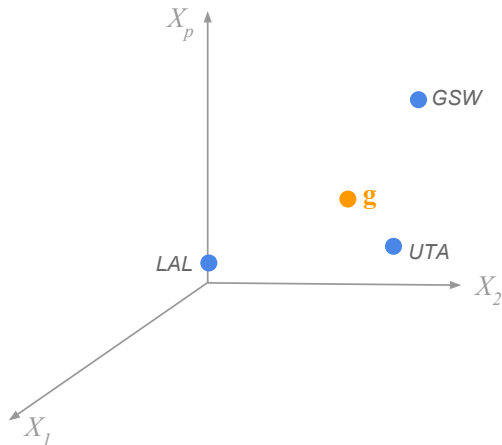
One way to consider the dispersion of data (in a mathematical form) is by adding the square distances among all pairs of points.

## Square distances from centroid

Another way to measure the dispersion of data is by considering the square distances of all points around the center of gravity (i.e. centroid)

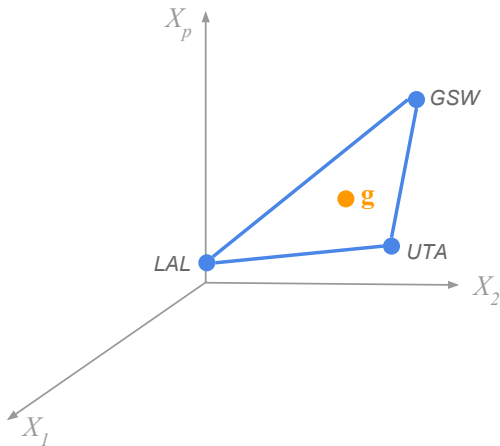


# Imagine 3 points and its centroid



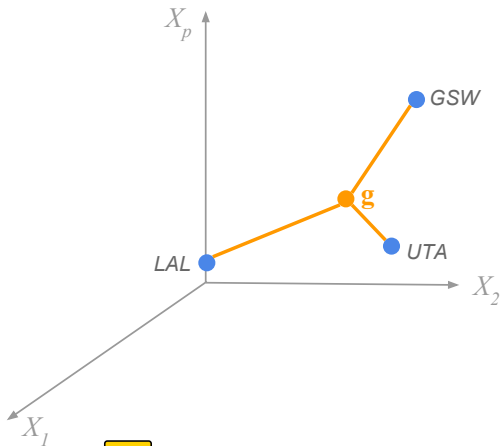
Centroid  $g$  is the “average” team.

# Dispersion: Sum of square of all dists



$$\text{SSD} = 2d^2(\text{LAL}, \text{GSW}) + 2d^2(\text{LAL}, \text{UTA}) + 2d^2(\text{GSW}, \text{UTA})$$

Sum of  $2n \times$  square dists w.r.t. centroid



$$\text{SSD} = (2 \times 3) \times \{d^2(\text{LAL}, g) + d^2(\text{GSW}, g) + d^2(\text{UTA}, g)\}$$

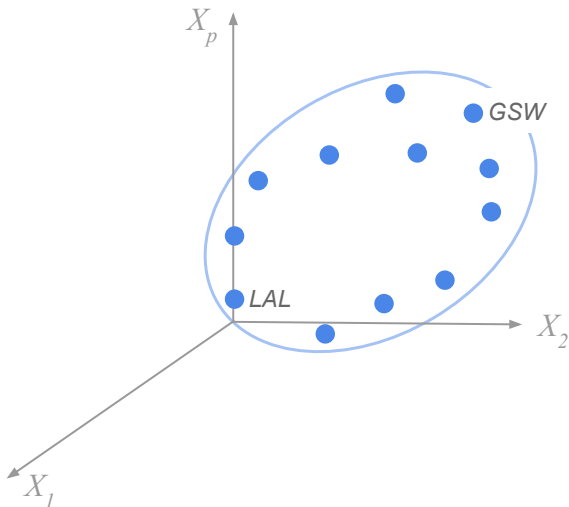
# Inertia

One way to take into account the dispersion of the data is with the concept of **Inertia**.

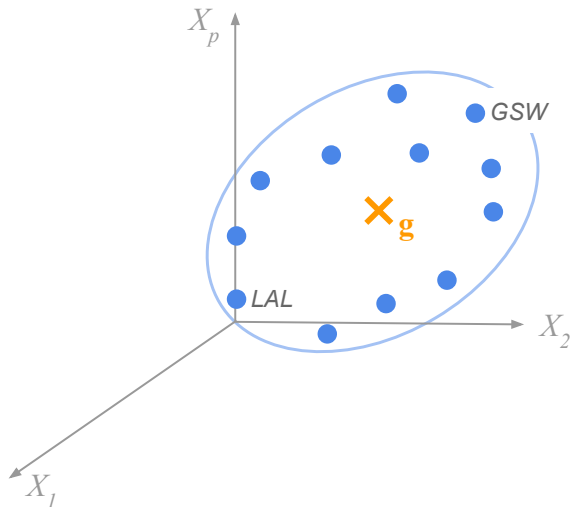
- ▶ Inertia is a term borrowed from the *moment of inertia* in mechanics (physics).
- ▶ This involves thinking about data as a rigid body (i.e. particles).
- ▶ We use the term Inertia to convey the idea of dispersion in the data.
- ▶ In multivariate methods, the term **Inertia** generalizes the notion of variance.
- ▶ Think of Inertia as a “multidimensional variance”



# Cloud of teams in p-dimensional space



# Centroid (i.e. the average team)



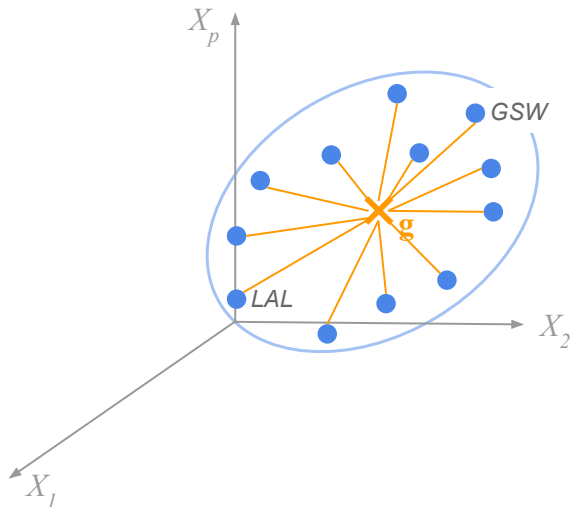
# Formula of Total Inertia

The Total Inertia,  $I$ , is a weighted sum of square distances among all pairs of objects:



$$I = \frac{1}{2n^2} \sum_{i=1}^n \sum_{h=1}^n d^2(i, h)$$

# Overall variation/spread (around centroid)





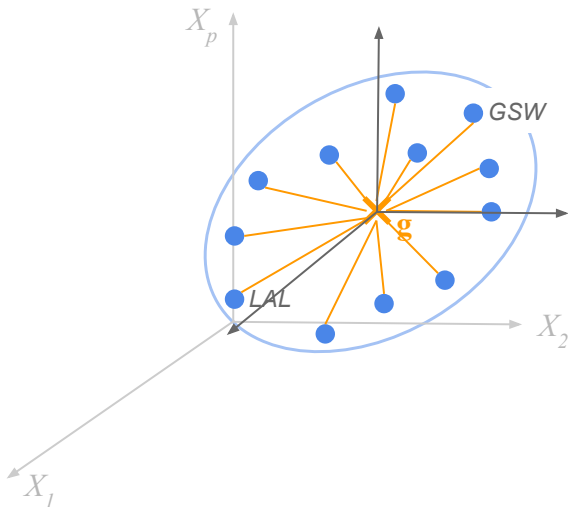
# Formula of Total Inertia

Equivalently, the **Total Inertia** can be calculated in terms of the centroid  $\mathbf{g}$ :

$$I = \frac{1}{n} \sum_{i=1}^n d^2(\mathbf{x}_i, \mathbf{g})$$

The Inertia is an average sum of square distances around the centroid  $\mathbf{g}$

Centered data: centroid is the origin



# Computing Inertia

$$\begin{aligned} \text{Inertia} &= \sum_{i=1}^n m_i d^2(\mathbf{x}_i, \mathbf{g}) \\ &= \sum_{i=1}^n \frac{1}{n} (\mathbf{x}_i - \mathbf{g})^\top (\mathbf{x}_i - \mathbf{g}) \\ &= \frac{1}{n} \text{tr}(\mathbf{X}^\top \mathbf{X}) \\ &= \frac{1}{n} \text{tr}(\mathbf{X} \mathbf{X}^\top) \end{aligned}$$



where  $m_i$  is the mass (i.e. weight) of individual  $i$ , usually  $1/n$

# Finding Principal Components

# Inertia Concept

## Inertia and PCA

In PCA we look for a low-dimensional subspace having **Projected Inertia** as close as possible to the **Original Inertia**.

## Criterion

The criterion used for dimensionality reduction implies that the **inertia** of a cloud of points in the optimal subspace is **maximum** (but **less than** the inertia in the **original space**).

# Criterion

## Maximize Projected Inertia

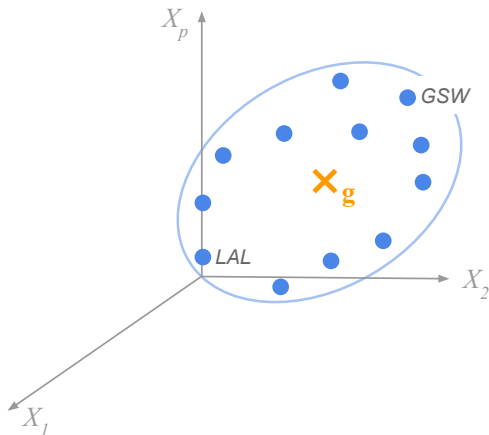
We want to maximize the Projected Inertia on subspace  $H$ :

$$\max \text{ projected } \sum_i d_H^2(\mathbf{x}_i, \mathbf{g})$$

## Axis of Inertia

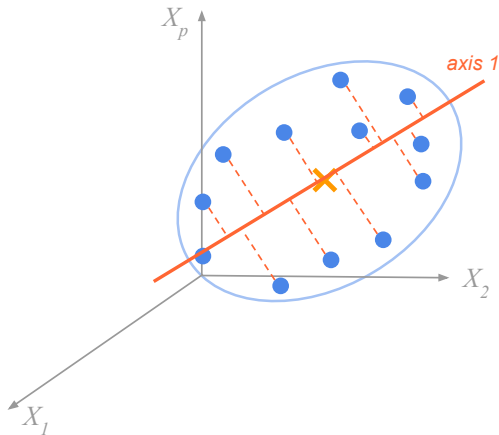
To find the subspace  $H$  we can look for each of its axes  $\Delta_1, \Delta_2, \dots, \Delta_k$  and its corresponding vectors  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k$  ( $k < p$ ).

# Looking for an axis 1



NBA teams in a  $p$ -dimensional space

# 1st axis



We want a 1st axis that retains **most of the projected inertia**



# First Axis and Principal Component

Projection of object  $i$  on axis  $\Delta_1$  generated by vector  $\mathbf{v}_1$

$$\mathbf{x}_i^T \mathbf{v}_1 = \sum_{j=1}^p x_{ij} v_{1j}$$

The 1st component  $z_1$  is the projection of all points on  $\mathbf{v}_1$

$$\mathbf{X} \mathbf{v}_1 = \mathbf{z}_1$$

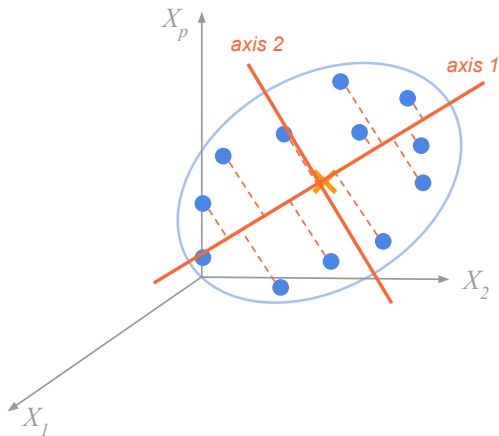
we don't really manipulate the axis  $\Delta_1$ , but its associated vector  $\mathbf{v}_1$

# First Axis and Principal Component

- ▶ The axis  $\Delta_1$  passes through the centroid  $g$  (with centered data,  $g$  is the origin)
- ▶ The axis  $\Delta_1$  is created by the unit-norm vector  $v_1$ , eigenvector of  $\frac{1}{n}X^T X$ , associated to the largest eigenvalue  $\lambda_1$
- ▶ The explained inertia by the axis  $\Delta_1$  is equal to  $\lambda_1$
- ▶ With standardized data, the proportion of explained inertia by  $\Delta_1$  is  $\lambda_1/p$



## 2nd axis



We want a 2nd axis, orthogonal to  $\Delta_1$ , that retains most of the remaining projected inertia

## Second Axis and Principal Component

Projection of object  $i$  on axis  $\Delta_2$  generated by vector  $\mathbf{v}_2$

$$\mathbf{x}_i^T \mathbf{v}_2 = \sum_{j=1}^p x_{ij} v_{2j}$$

The 2nd component  $\mathbf{z}_2$  is the projection of all points on  $\mathbf{v}_2$

$$\mathbf{X} \mathbf{v}_2 = \mathbf{z}_2$$

we don't really manipulate the axis  $\Delta_2$ , but its associated vector  $\mathbf{v}_2$

## Second Axis and Principal Component

- ▶ The axis  $\Delta_2$  passes through the centroid  $g$  and it is perpendicular to  $\Delta_1$
- ▶ The axis  $\Delta_2$  is created by the unit-norm vector  $\mathbf{v}_2$ , eigenvector of  $\frac{1}{n}\mathbf{X}^T\mathbf{X}$ , associated to the second largest eigenvalue  $\lambda_2$
- ▶ The explained inertia by the axis  $\Delta_2$  is equal to  $\lambda_2$
- ▶ With standardized data, the proportion of explained inertia by  $\Delta_2$  is  $\lambda_2/p$

# Computational note

In practice, most software routines for PCA **don't** really work with the **population covariance matrix**  $\frac{1}{n} \mathbf{X}^T \mathbf{X}$ .

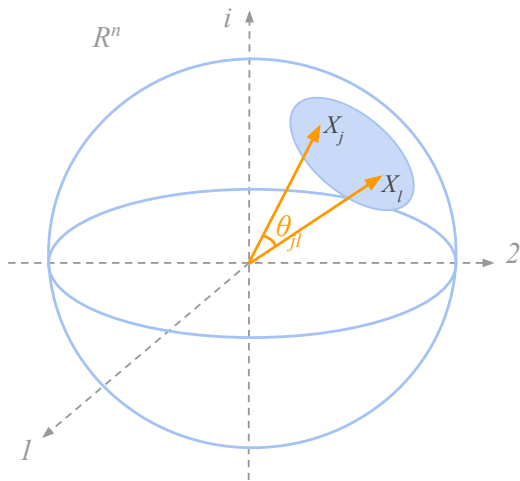
Instead, most programs work with the **sample** covariance matrix:  $\frac{1}{n-1} \mathbf{X}^T \mathbf{X}$

Notice that with **standardized data**,  $\frac{1}{n-1} \mathbf{X}^T \mathbf{X} = \mathbf{R}$ , is the (sample) **correlation** matrix.



# Looking at Variables

# Looking at the cloud of standardized variables





# Looking at the cloud of standardized variables

- ▶ With standardized data, the  $p$  variables are located within a hypersphere of radius 1 in an  $n$ -dimensional space..
- ▶ We represent them graphically as vectors.
- ▶ The scalar product between two variables  $X_j$  and  $X_l$  is:

$$\langle X_j, X_l \rangle = \sum_{i=1}^n x_{ij} x_{il} = \|\mathbf{x}_j\| \|\mathbf{x}_l\| \cos(\theta_{jl})$$

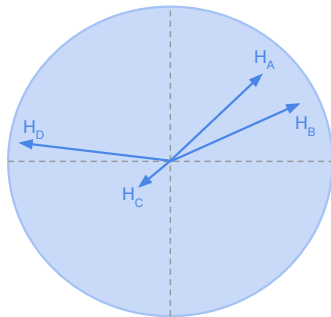
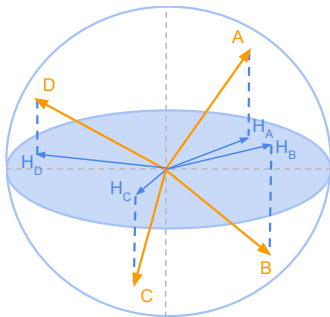
- ▶ Notice that:

$$\cos(\theta_{jl}) = \frac{\mathbf{x}_j^T \mathbf{x}_l}{\|\mathbf{x}_j\| \|\mathbf{x}_l\|} = \text{cor}(X_j, X_l)$$

# Projecting the cloud of standardized variables

- ▶ The property  $\cos(\theta_{jl}) = \text{cor}(X_j, X_l)$  is **essential** in PCA
- ▶ A representation of the cloud of variables can be used to **visualize the correlations** (through the **angles** between variables)
- ▶ The cloud of variables is also **projected onto a low dimensional space**.
- ▶ In this case, the **distance** between two variables is computed with **inner products**.

# Projection of best subspace




Projection of the scatterplot of variables on the main plane of variability

# Projecting the cloud of standardized variables

The projection of variable  $j$  onto an axis  $k$ , is equal to the cosine of the angle  $\theta_{jk}$ .

The criterion maximizes:

$$\sum_{j=1}^p \cos^2(\theta_{jk}) = \sum_{j=1}^p \text{cor}^2(\mathbf{x}_j, \mathbf{z}_k)$$


where  $\mathbf{z}_k$  is the new variable which is the most correlated with all of the original variables.

## Finding subspace for variables

Projection of variable  $j$  on axis  $H_1$  generated by vector  $\mathbf{u}_1$

$$\mathbf{x}_j^T \mathbf{u}_1 = \sum_{i=1}^n x_{ij} u_{i1}$$

The synthetic variable  $\mathbf{u}_1$  can be used to obtain a factor  $\mathbf{q}_1$

$$\mathbf{X}^T \mathbf{u}_1 = \mathbf{q}_1$$

we don't really manipulate the axis  $H_1$ , but its associated vector  $\mathbf{u}_1$

# Finding subspace for variables

Solution:  $\mathbf{u}_1$  is the first eigenvector of  $\frac{1}{n}\mathbf{X}\mathbf{X}^\top$ , the matrix of inner products between individuals

$$\frac{1}{n}\mathbf{X}\mathbf{X}^\top \mathbf{u}_1 = \lambda_1 \mathbf{u}_1$$

The subsequent dimensions are the other eigenvectors  $\mathbf{u}_2, \mathbf{u}_3, \dots$

And the corresponding variable factors are given by:

$$\mathbf{Q} = \mathbf{X}^\top \mathbf{U}$$

# Finding subspace for variables

It can be shown that the PCs can also be obtained as:

$$\mathbf{Z} = \frac{1}{\sqrt{n}} \mathbf{U} \mathbf{\Lambda}^{1/2}$$

where:

- ▶  $\mathbf{\Lambda}$  is the diagonal matrix of eigenvectors of  $\frac{1}{n} \mathbf{X} \mathbf{X}^T$
- ▶  $\mathbf{U}$  is the matrix of eigenvectors of  $\frac{1}{n} \mathbf{X} \mathbf{X}^T$

But keep in mind that PCs can be rescaled.

Note that most PCA programs work with  $n - 1$  instead of  $n$ .

# Relationship between the representations of Individuals and Variables



# Link between representations

$$\text{SVD of: } \frac{1}{\sqrt{n-1}}\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T$$

$$\mathbf{Z} = \mathbf{X}\mathbf{V} = \frac{1}{\sqrt{n-1}}\mathbf{X}\mathbf{Q}\mathbf{D}^{-1} \Rightarrow \mathbf{V} = \frac{1}{\sqrt{n-1}}\mathbf{Q}\mathbf{D}^{-1}$$

$$\mathbf{Q} = \mathbf{X}^T\mathbf{U} = \frac{1}{\sqrt{n-1}}\mathbf{X}^T\mathbf{Z}\mathbf{D}^{-1} \Rightarrow \mathbf{U} = \frac{1}{\sqrt{n-1}}\mathbf{Z}\mathbf{D}^{-1}$$

# Link between representations

Principal Components or Scores

$$z_{ik} = \frac{1}{\sqrt{n-1}} \times \frac{1}{\sqrt{\lambda_k}} \times \sum_{j=1}^p x_{ij} q_{jk}$$

Factors for Variables

$$q_{jk} = \frac{1}{\sqrt{n-1}} \times \frac{1}{\sqrt{\lambda_k}} \times \sum_{i=1}^n x_{ij} z_{ik}$$

# Principal Components?

## Meaning of *Principal*

The term **Principal**, as used in PCA, has to do with the notion of **principal axis** from geometry and linear algebra

## Principal Axis

A *principal axis* is a certain line in a Euclidean space associated to an ellipsoid or hyperboloid, generalizing the major and minor axes of an ellipse

# References

- ▶ **Exploratory Multivariate Analysis by Example Using R** by Husson, Le and Pages (2010). *Chapter 1: Principal Component Analysis (PCA)*. CRC Press.
- ▶ **An R and S-Plus Companion to Multivariate Analysis** by Brian Everitt (2004). *Chapter 3: Principal Components Analysis*. Springer.
- ▶ **Principal Component Analysis** by Ian Jolliffe (2002). Springer.
- ▶ **Data Mining and Statistics for Decision Making** by Stéphane Tuffery (2011). *Chapter 7: Factor Analysis*. Editions Technip, Paris.

# References (French Literature)

- ▶ **Statistique Exploratoire Multidimensionnelle** by Lebart et al (2004). *Chapter 3, section 3: Analyse factorielle discriminante.* Dunod, Paris.
- ▶ **Probabilites, analyse des donnees et statistique** by Gilbert Saporta (2011). *Chapter 6: Analyse en Composantes Principaux.* Editions Technip, Paris.
- ▶ **Statistique: Methodes pour decrire, expliquer et prevoir** by Michel Tenenhaus (2008). *Chapter 10: L'analyse discriminante.* Dunod, Paris.
- ▶ **Analyses factorielles simples et multiples** by Brigitte Escofier et Jerome Pages (2016, 5th edition). *Chapter 2: L'analyse discriminante.* Dunod, Paris.