

# Python 기반 기초통계 실습

2021. 8. 13

정 준 수 Ph.D

# 모집단과 샘플(표본)

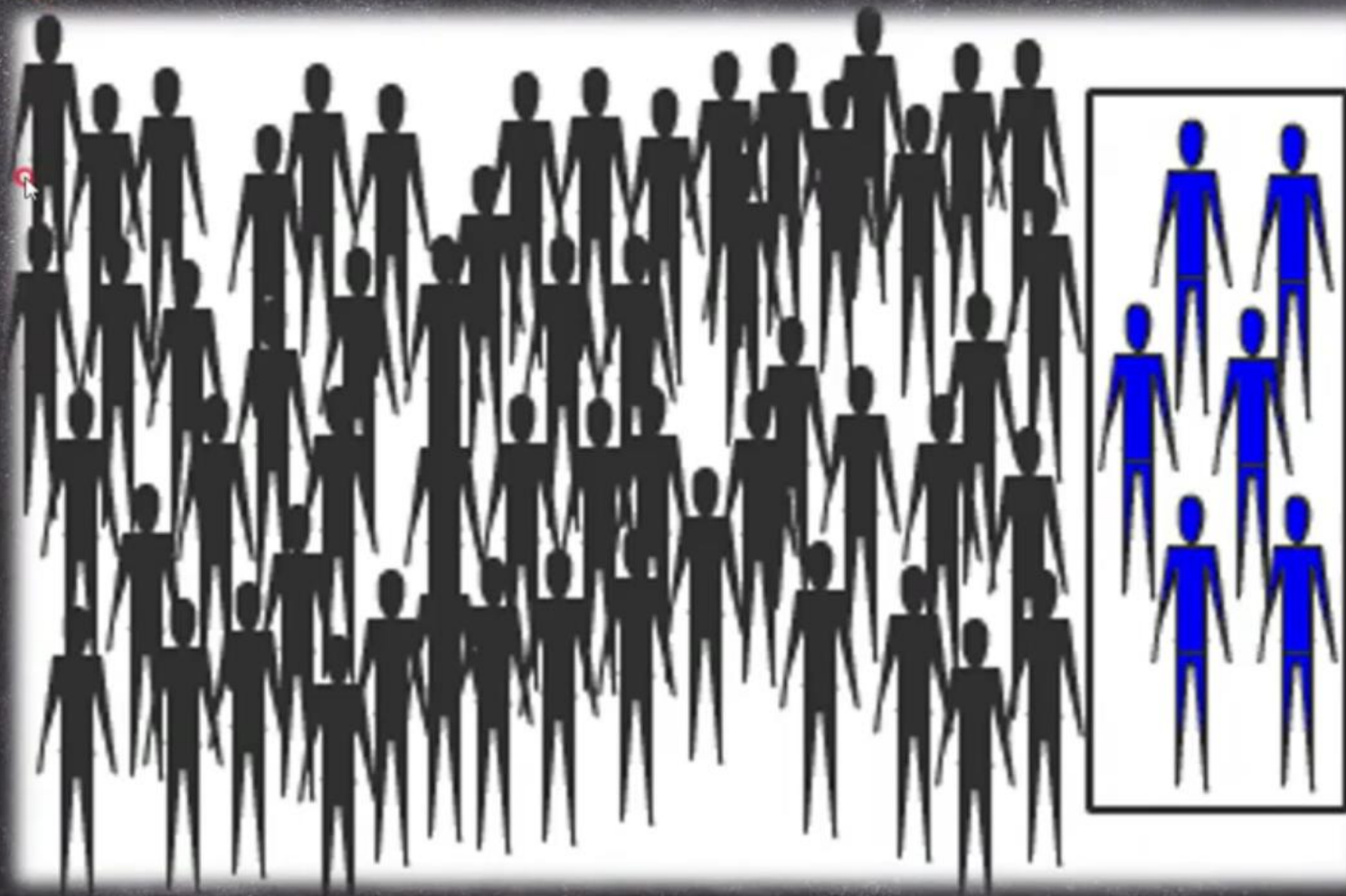
## 모집단

관 측 치 =  $N$

평 균 값 =  $\mu$

분 산 =  $\sigma^2$

표준편차 =  $\sigma$



## 표본(샘플)

관 측 치 =  $n$

평 균 값 =  $\bar{X}$

분 산 =  $s^2$

표준편차 =  $s$

## 정규분포와 표준편차

---

확률변수  $X$ 가 평균이  $\mu$ , 표준편차가  $\sigma$ 인 정규분포를 따르면 확률밀도함수는

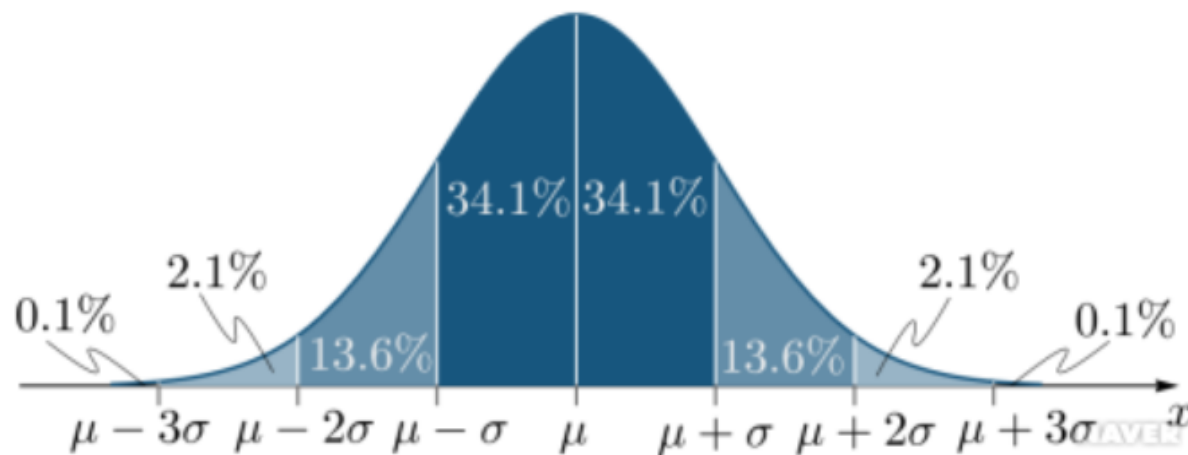
$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

이다. 모집단에서 임의로 하나의 값을 취할 때 이 값이

구간  $[\mu - \sigma, \mu + \sigma]$ 에 속할 확률은 68.27 %

구간  $[\mu - 2\sigma, \mu + 2\sigma]$ 에 속할 확률은 95.45 %

구간  $[\mu - 3\sigma, \mu + 3\sigma]$ 에 속할 확률은 99.73 %이다.



# Sampling을 통한 통계 예제

## – MLB 선수연봉 데이터

<https://github.com/JSJeong-me/SEMICON-BigData/blob/main/statistics-intro.ipynb>

# Charles Spearman

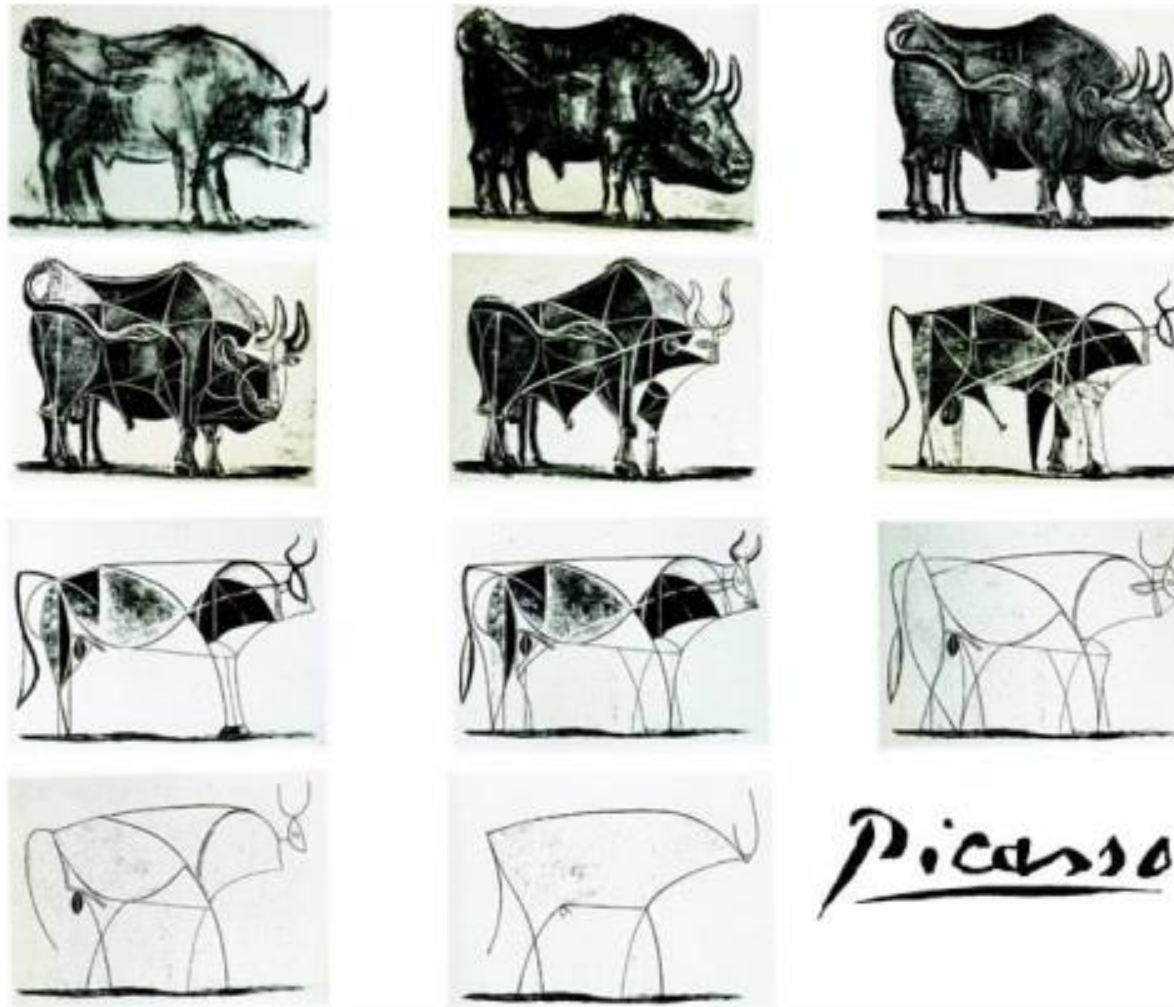


- Statistics-factor analysis
- Creator of “Spearman’s rank correlation coefficient” (-1 to +1)
- Intelligence theories
- General Intelligence or “g-factor” —positive correlations among cognitive abilities that account for most of IQ
- Much of intell. is heritable

[https://en.wikipedia.org/wiki/Spearman%27s\\_rank\\_correlation\\_coefficient](https://en.wikipedia.org/wiki/Spearman%27s_rank_correlation_coefficient)



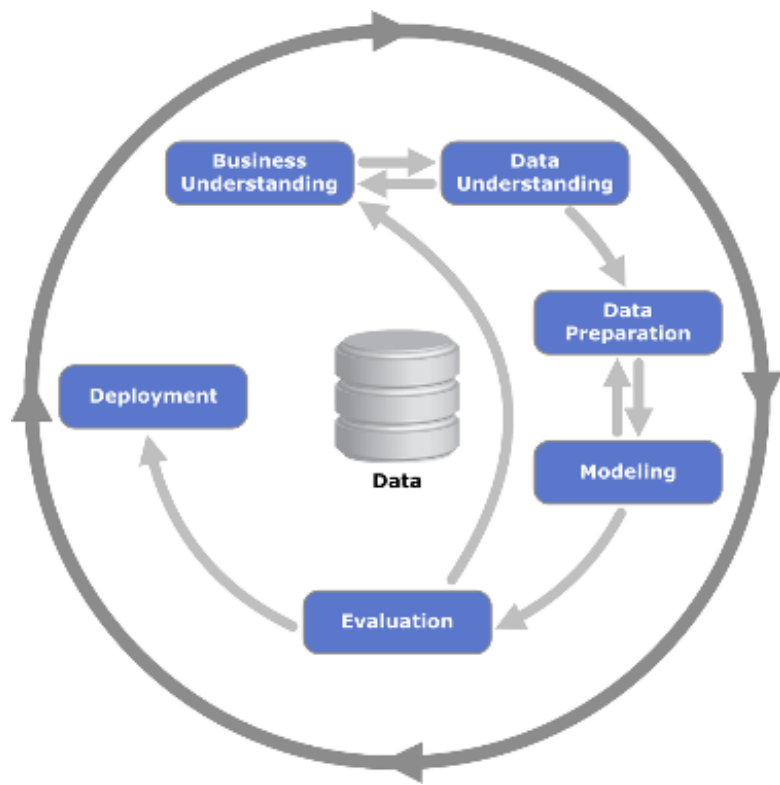
## ◆ 추상화 (Abstract)



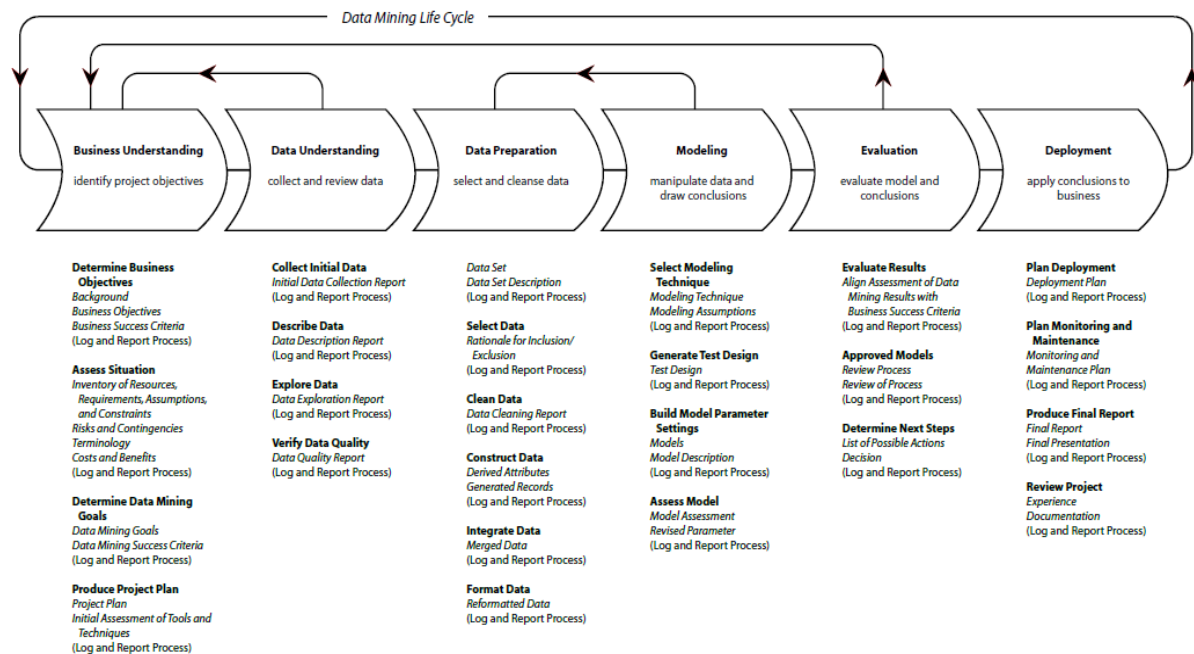
Pablo Picasso, Bull (plates I - XI) 1945

# CRISP-DM (Cross Industry Standard Process for Data Mining)

CRISP-DM(Cross Industry Standard Process for Data Mining)은 데이터 마이닝 전문가가 사용하는 일반적인 접근 방식을 설명한 가장 널리 사용되는 공개 표준 분석 모델입니다.



## Phases



**Generic Tasks**  
**Specialized Tasks**  
 (Process Instances)

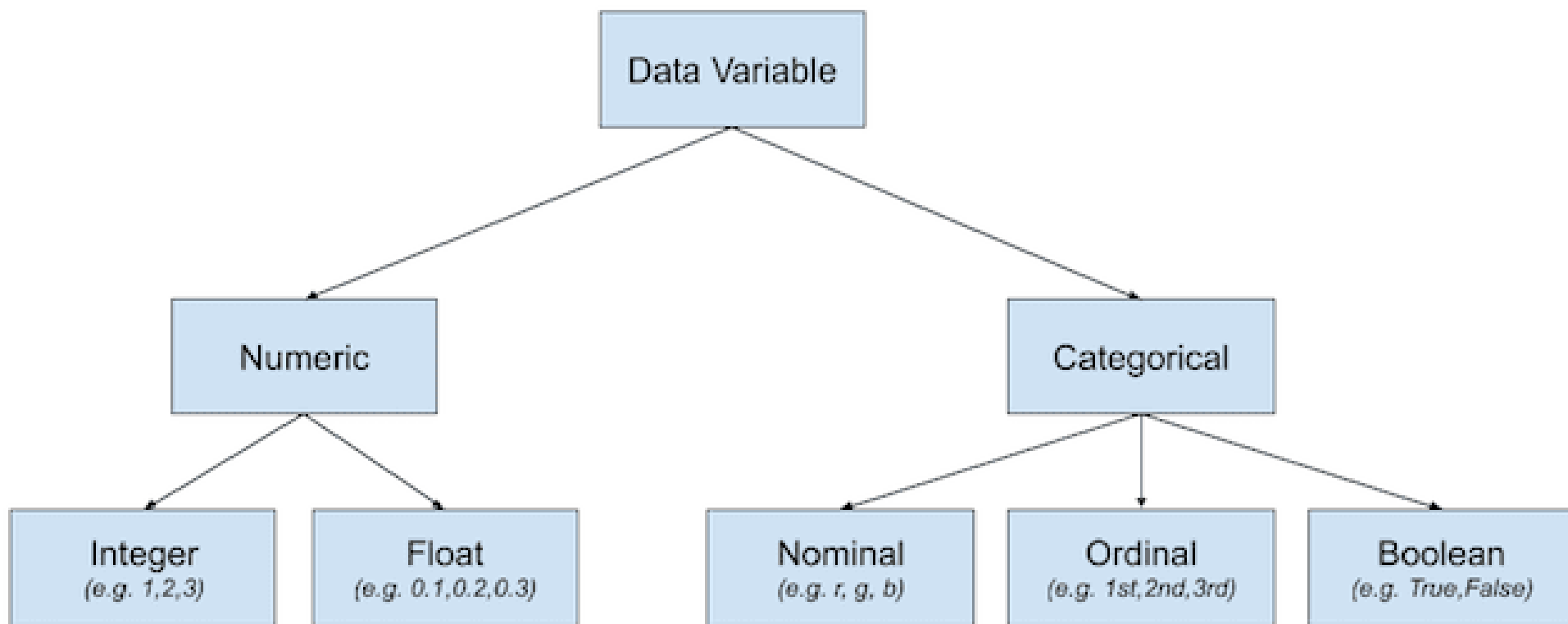
## a visual guide to CRISP-DM methodology

SOURCE CRISP-DM 1.0  
<http://www.crisp-dm.org/download.htm>  
 DESIGN Nicole Leaper  
<http://www.nicoleleaper.com>



# Data 변수 타입 분류

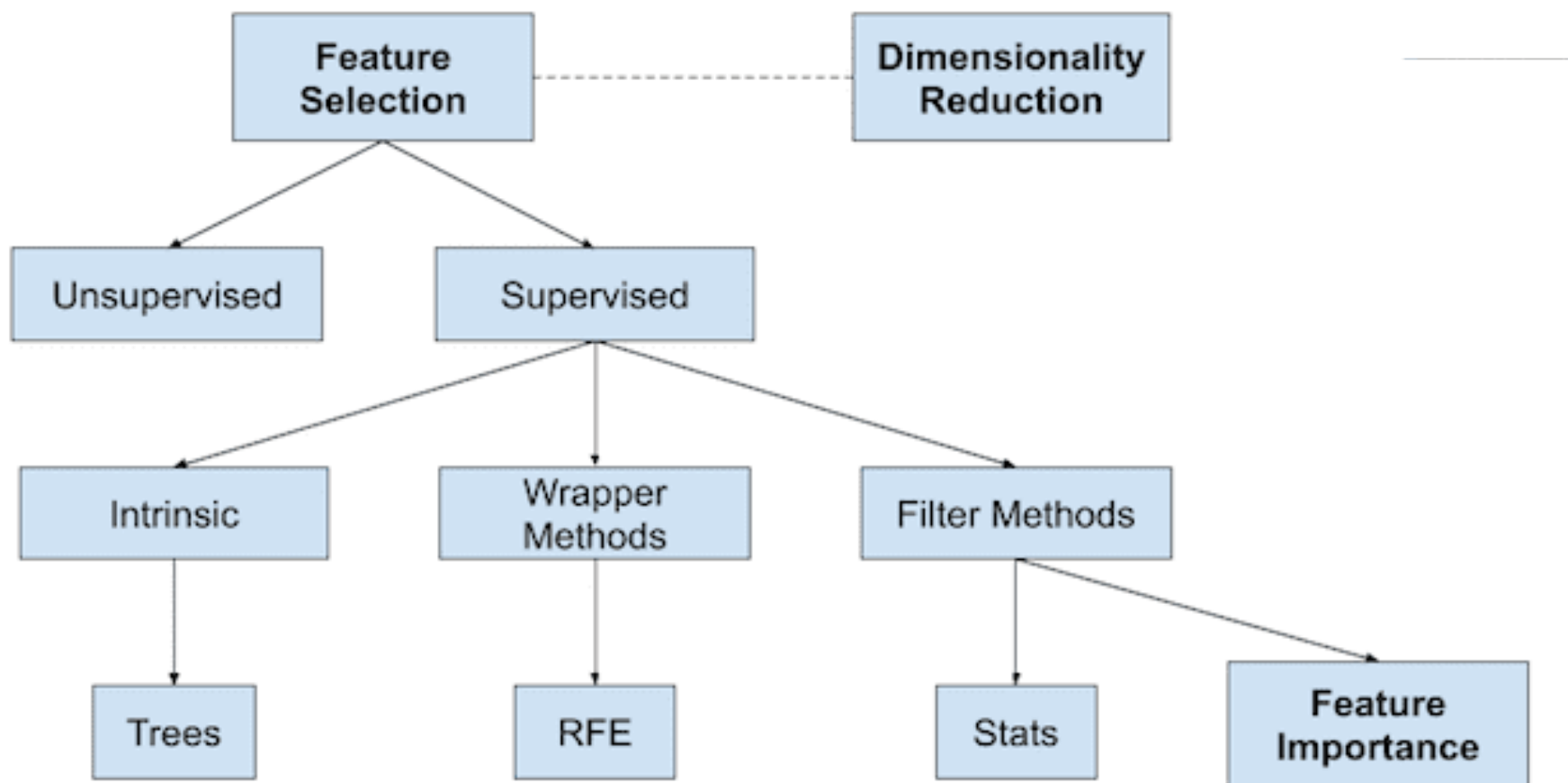
## Overview of Data Variable Types





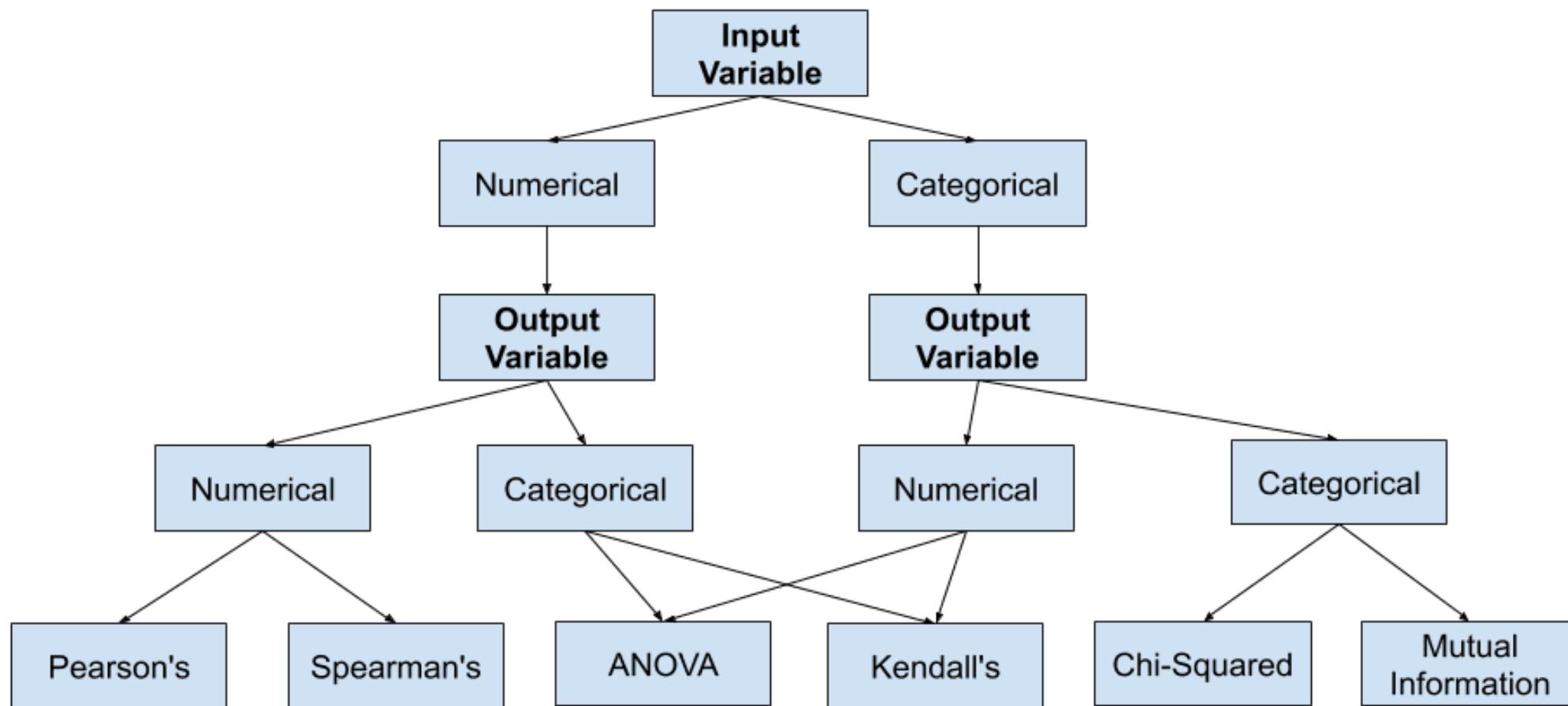
# Feature Selection 분류 과정

## Overview of Feature Selection Techniques



# Feature Selection 분류 방법

How to Choose a Feature Selection Method



## 차원의 저주란,

\*데이터 학습을 위해 **차원이 증가**하면서 학습데이터 수가 차원의 수보다 적어져 **성능이 저하되는 현상**.

\*차원이 증가할 수록 개별 차원 내 학습할 데이터 수가 **적어지는(sparse) 현상 발생**

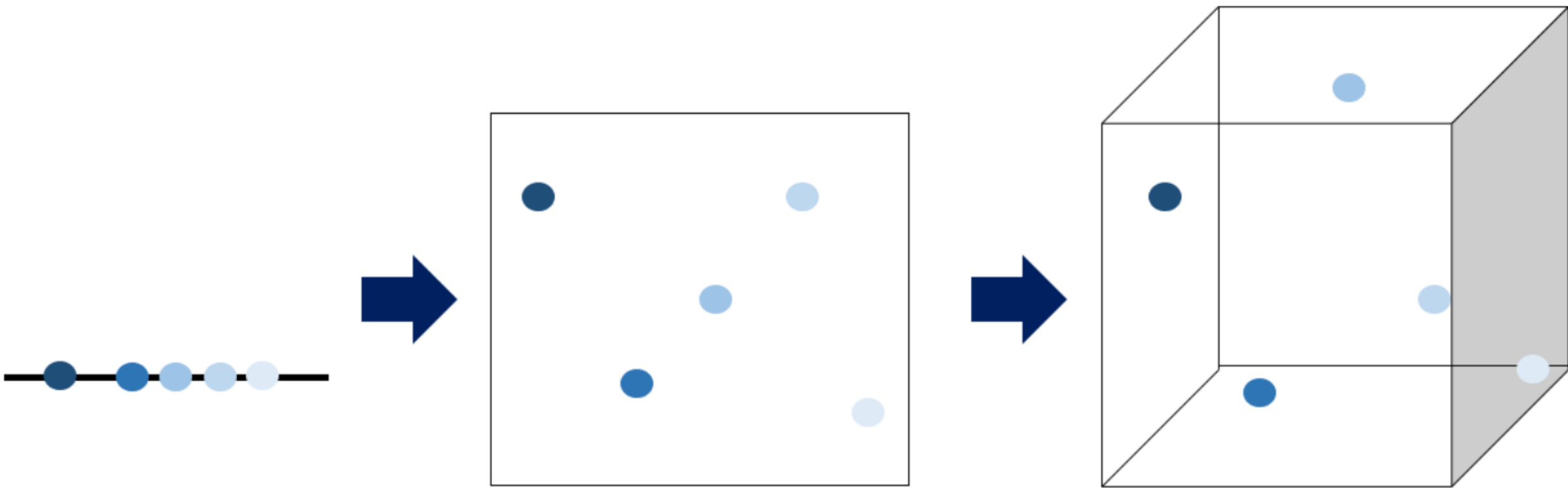
\*해결책: 차원을 줄이거나(축소시키거나) 데이터를 많이 획득

즉, 간단히 말해서

차원이 증가함에 따라(=변수의 수 증가) 모델의 성능이 안 좋아지는 현상을 의미합니다.

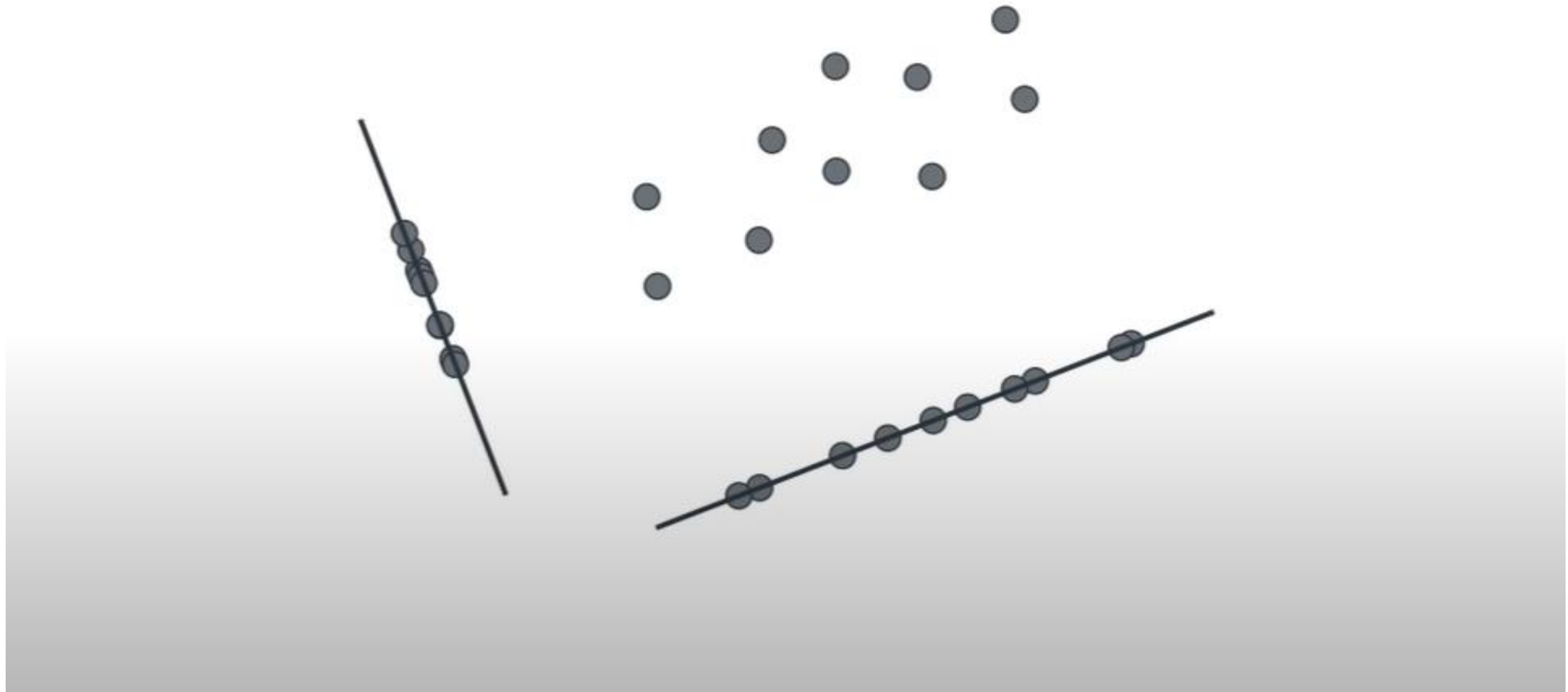
무조건 변수의 수가 증가한다고 해서 차원의 저주 문제가 있는 것이 아니라, **관측치 수보다 변수의 수가 많아지면** 발생합니다. (예를들어, 관측치 개수는 200개인데, 변수는 7000개)

왜 이런 현상이 발생할까요?

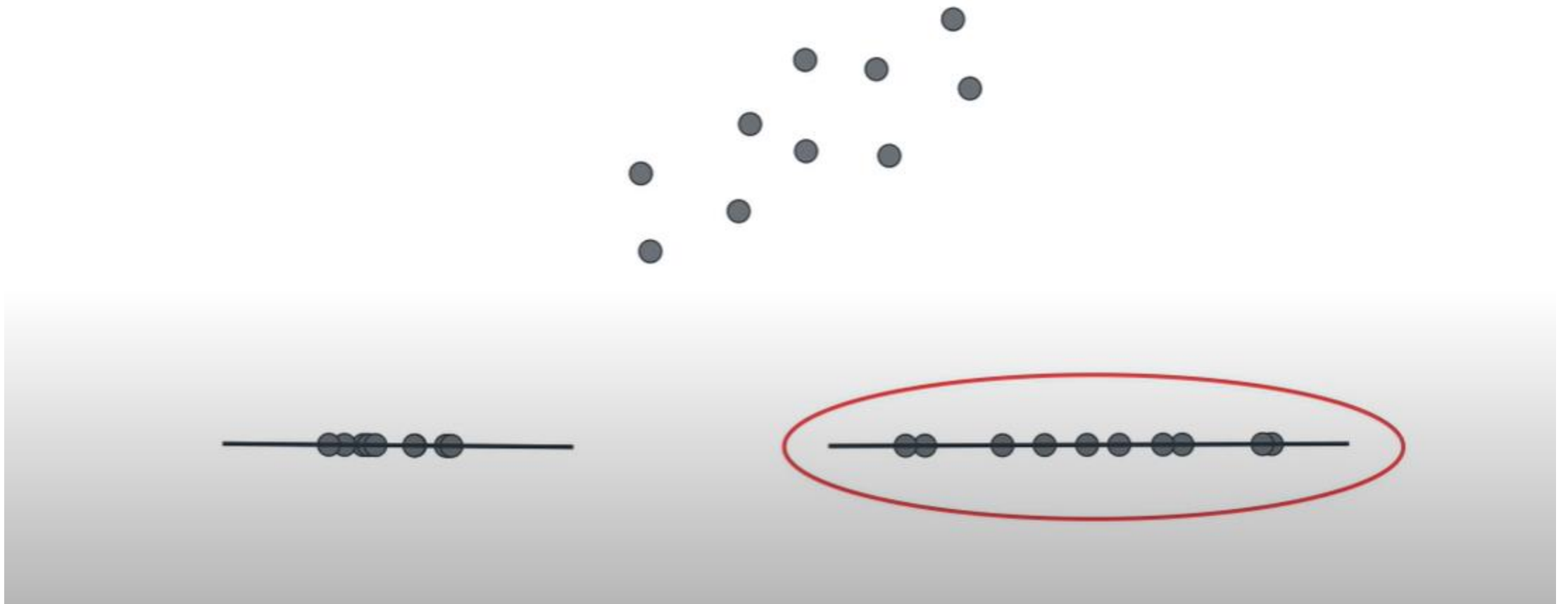


Made by: ta-daa

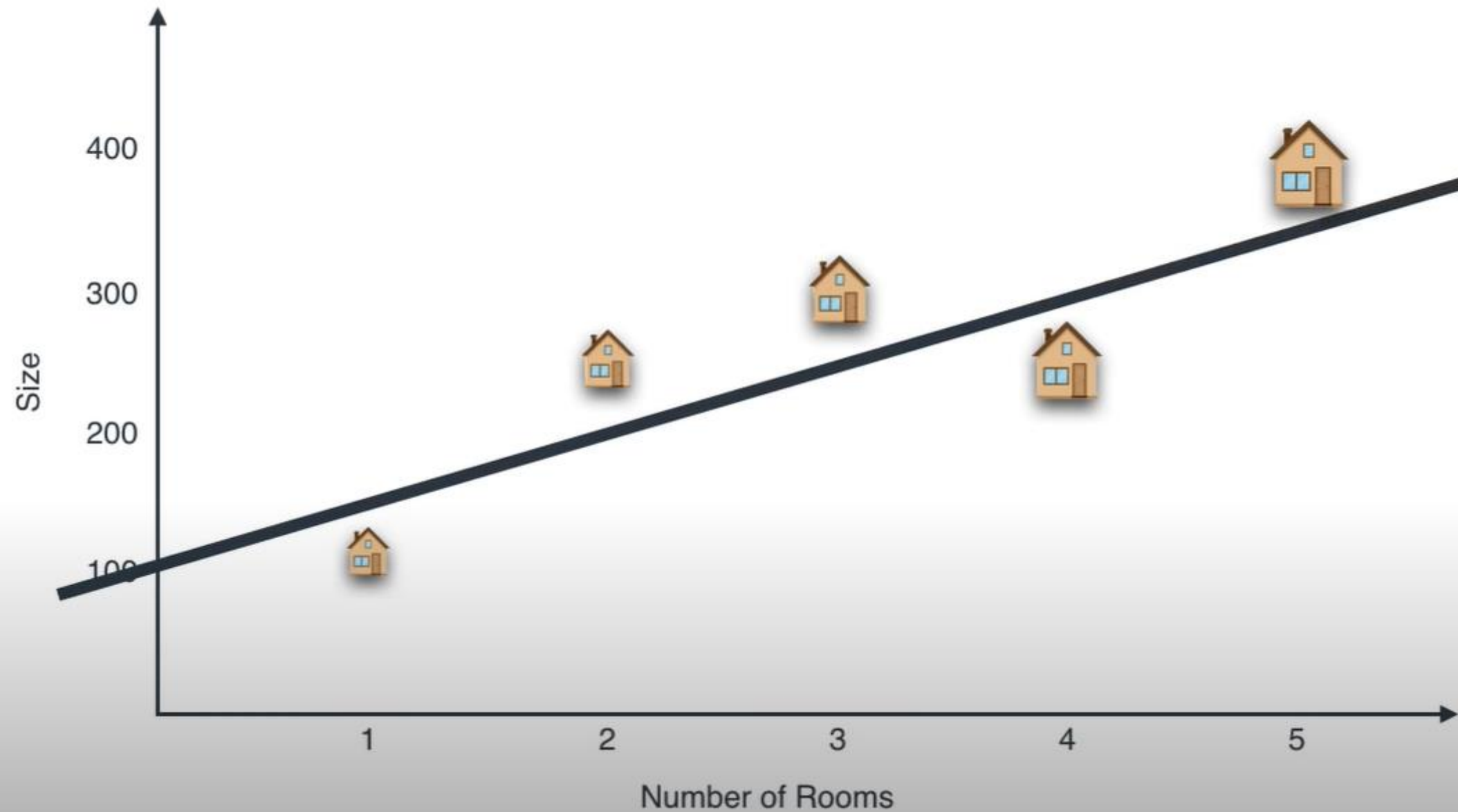
# Dimensionality Reduction



# Dimensionality Reduction





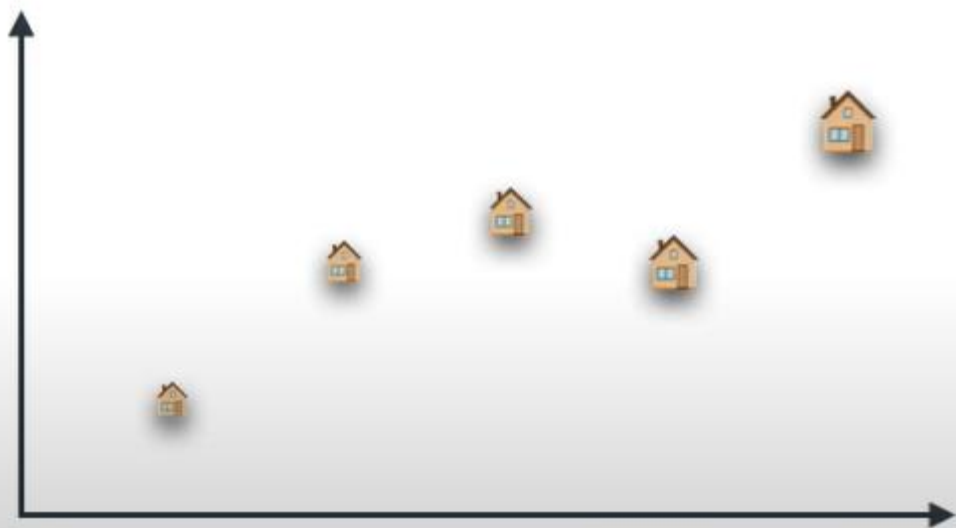




Size feature

2 dimensions

size  
number of rooms



1 dimension

size feature



# Housing Data

## 5 dimensions

Size

Number of rooms

Number of bathrooms

Schools around

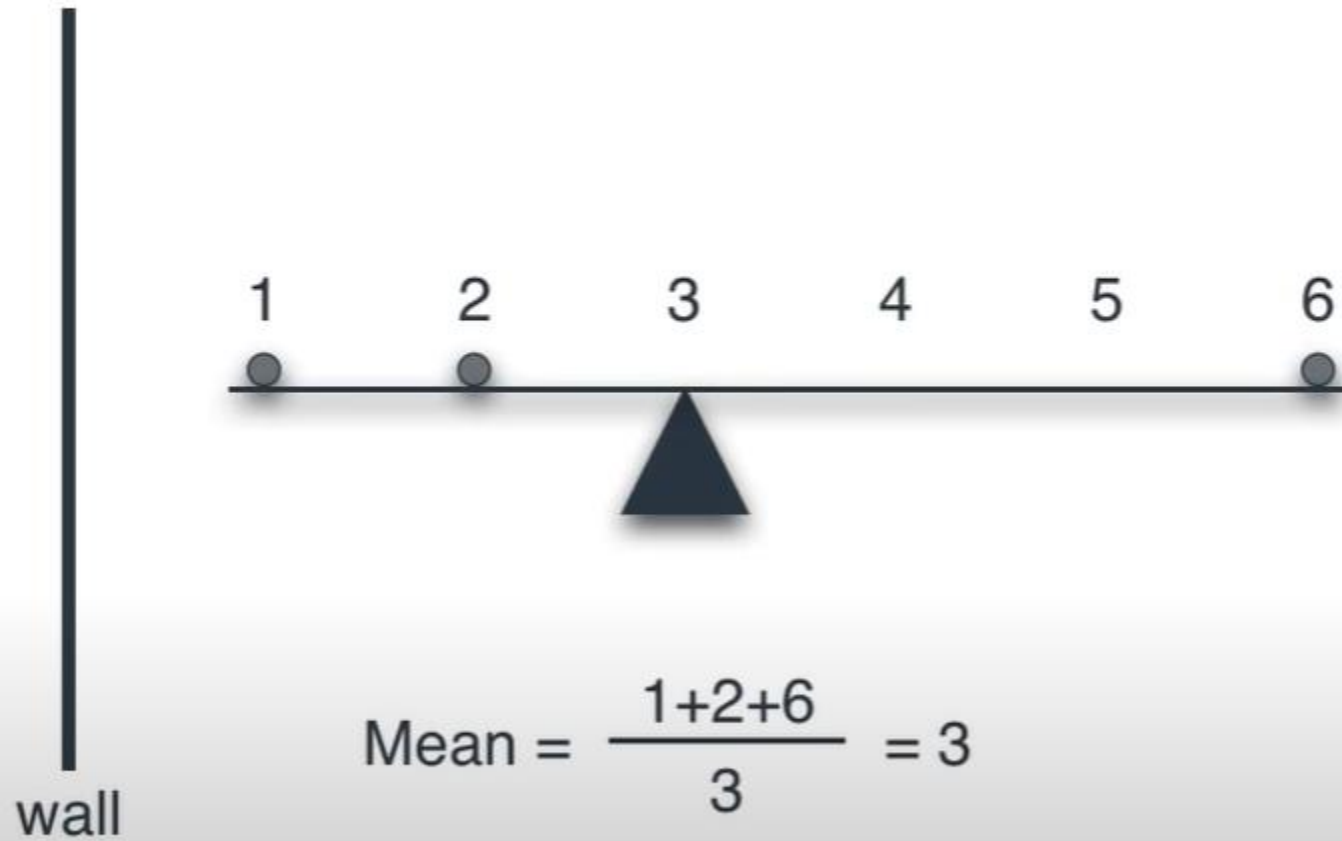
Crime rate

## 2 dimensions

Size feature

Location feature

# Mean



# Variance



$$\text{Variance} = \frac{1^2 + 0^2 + 1^2}{3} = 2/3$$



$$\text{Variance} = \frac{5^2 + 0^2 + 5^2}{3} = 50/3$$

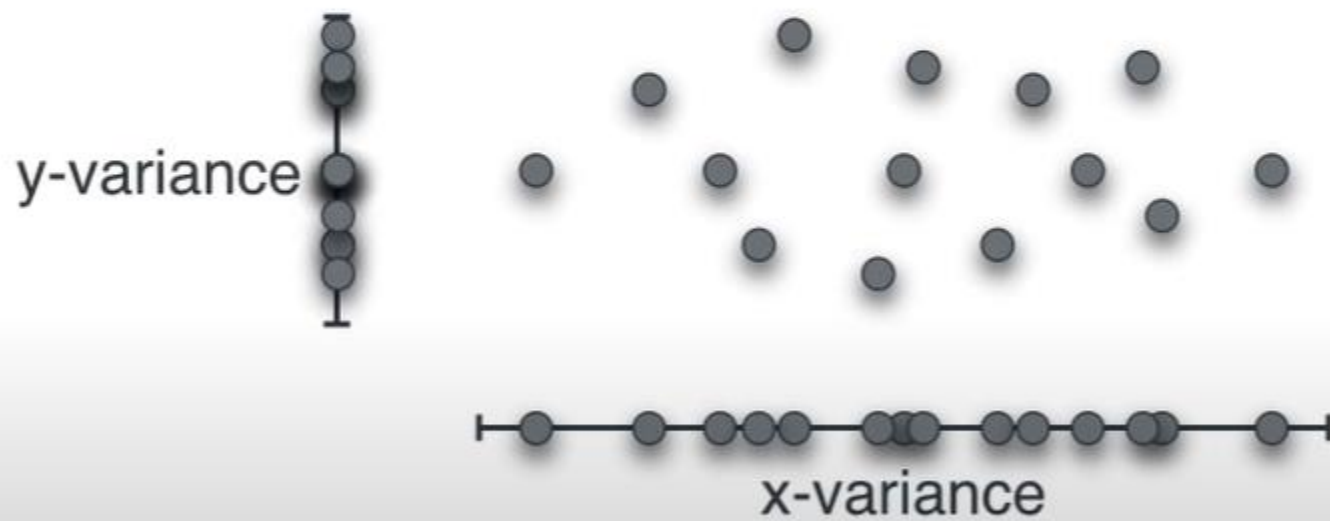


# Mean

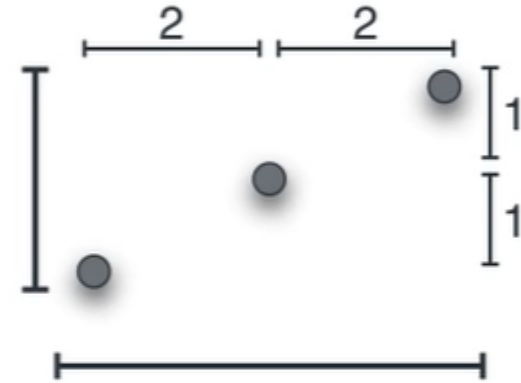
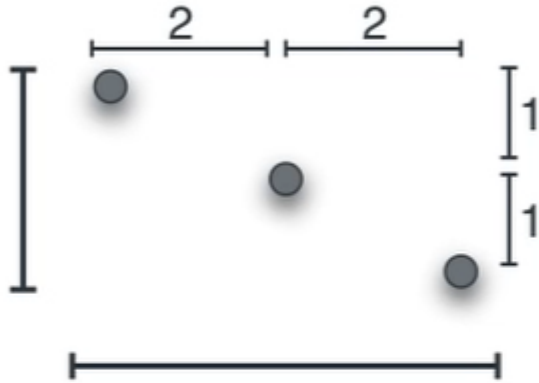


$$\text{Variance} = \frac{2^2 + 1^2 + 3^2}{3} = 14/3$$

# Variance?



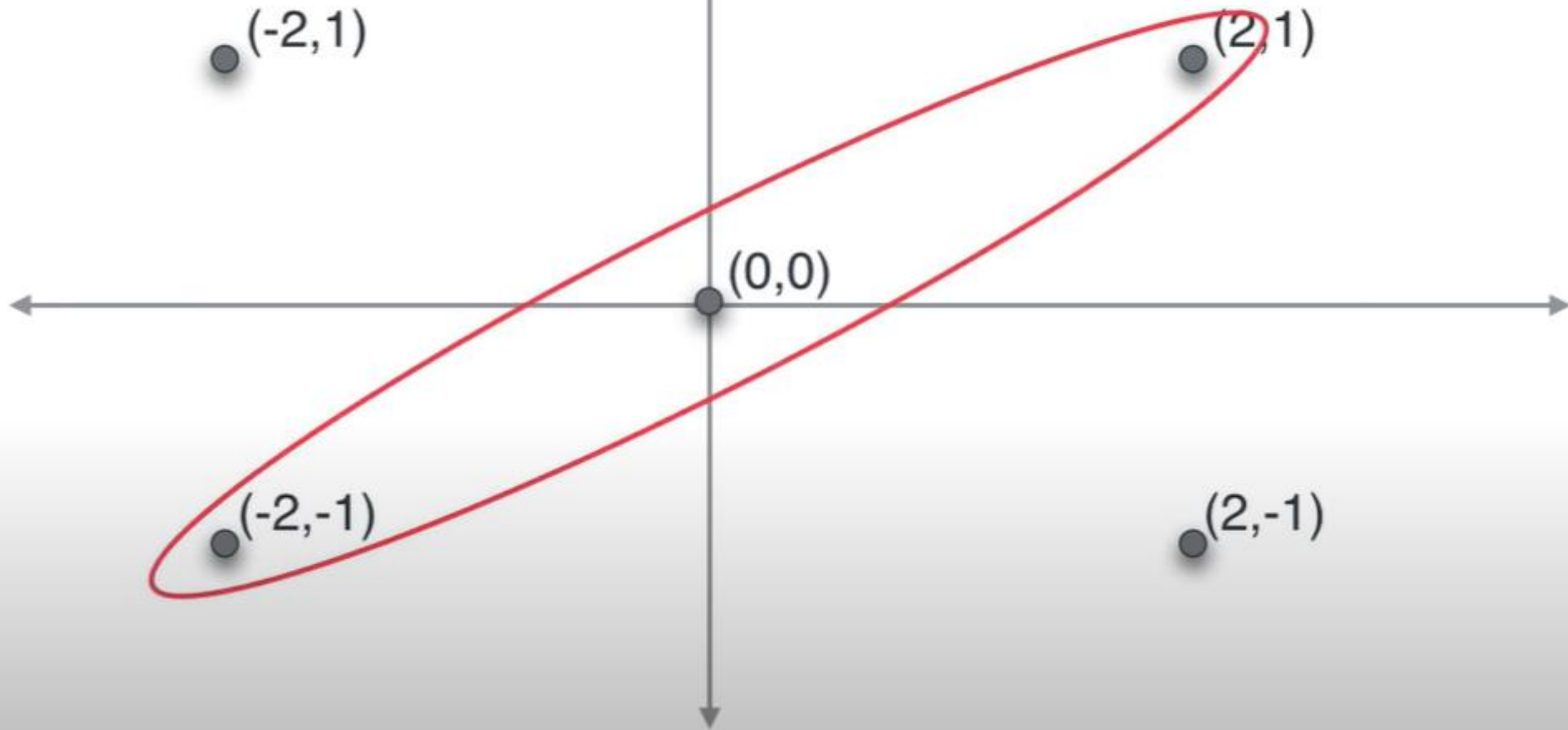
# Variance?



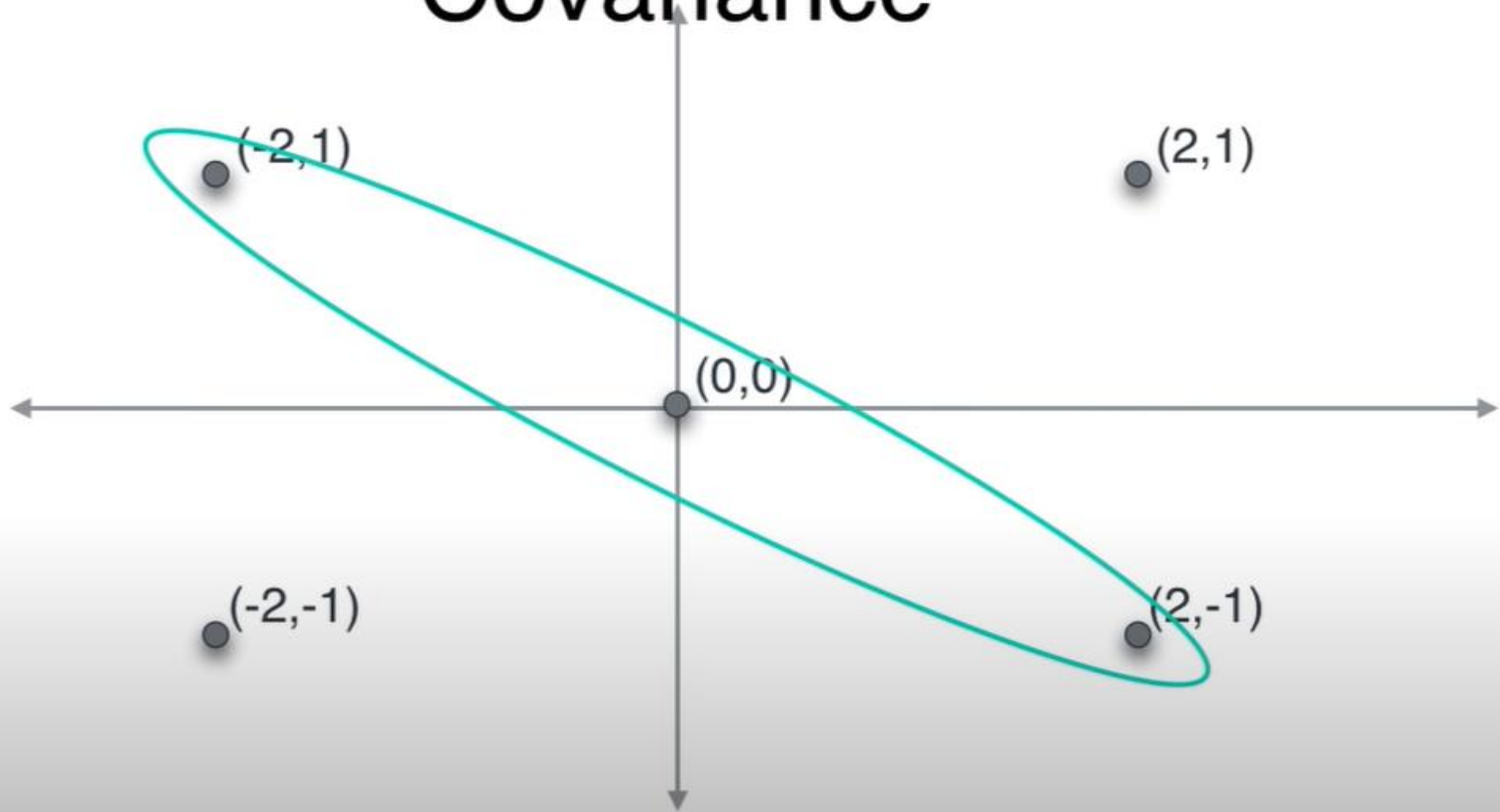
$$\text{x-variance} = \frac{2^2 + 0^2 + 2^2}{3} = 8/3$$

$$\text{y-variance} = \frac{1^2 + 0^2 + 1^2}{3} = 2/3$$

# Covariance



# Covariance



# Covariance

Product  
of  
coordinates

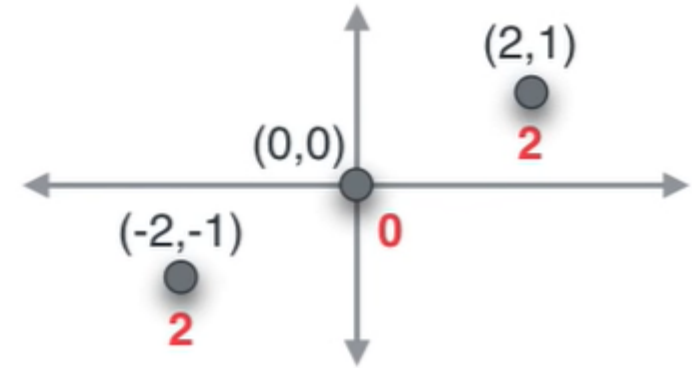




# Covariance

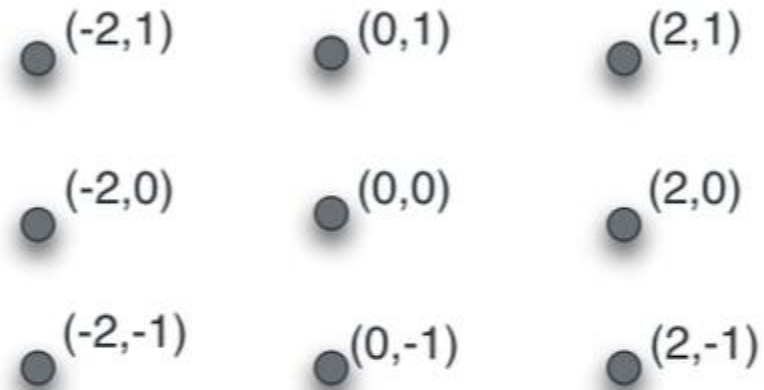


$$\text{covariance} = \frac{(-2) + 0 + (-2)}{3} = -4/3$$



$$\text{covariance} = \frac{2 + 0 + 2}{3} = 4/3$$

# Covariance



$$\text{covariance} = \frac{-2 + 0 + 2 + 0 + 0 + 0 + 2 + 0 + -2}{9} = 0$$

# Covariance



negative  
covariance



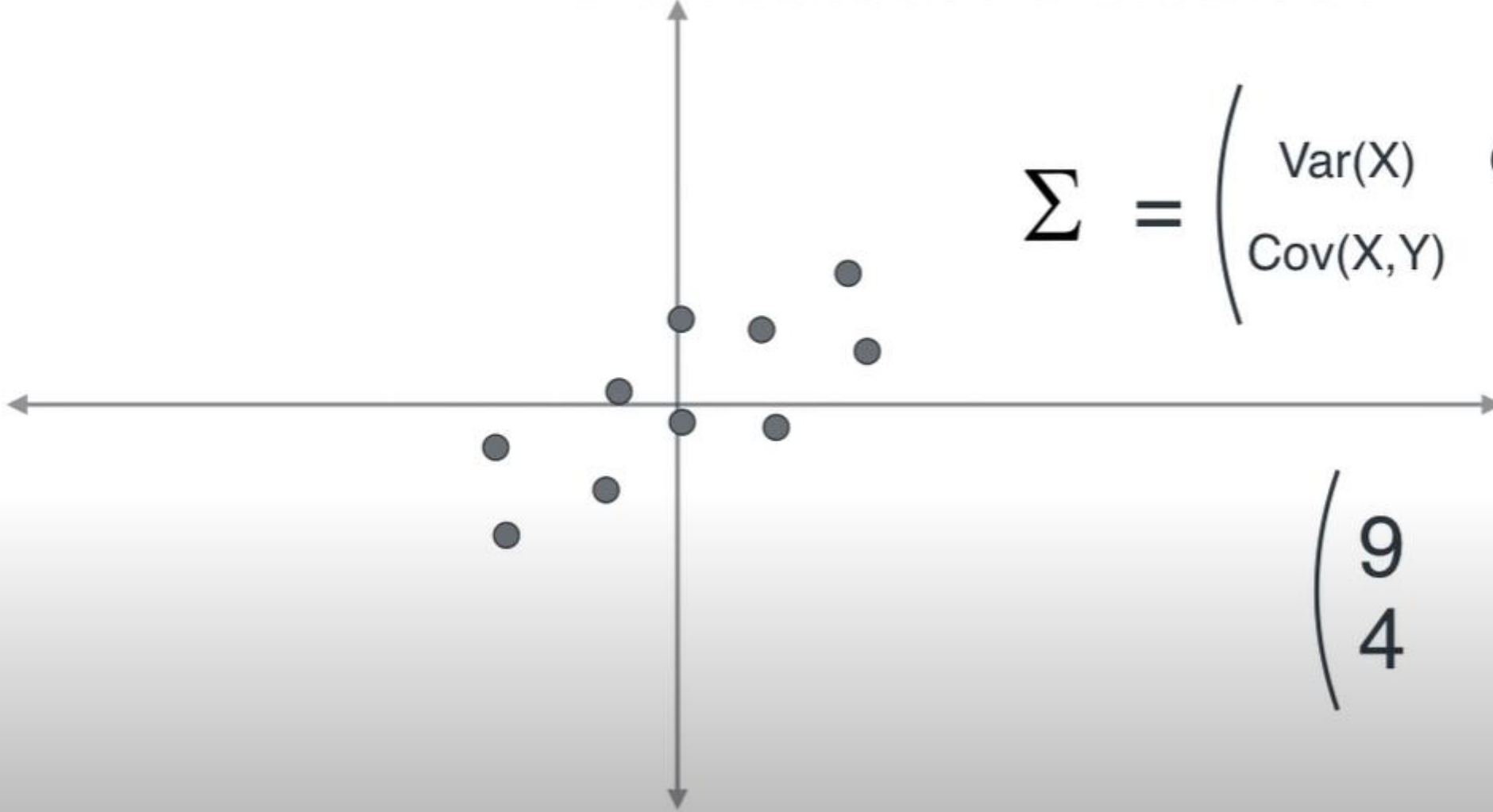
covariance zero  
(or very small)



positive  
covariance

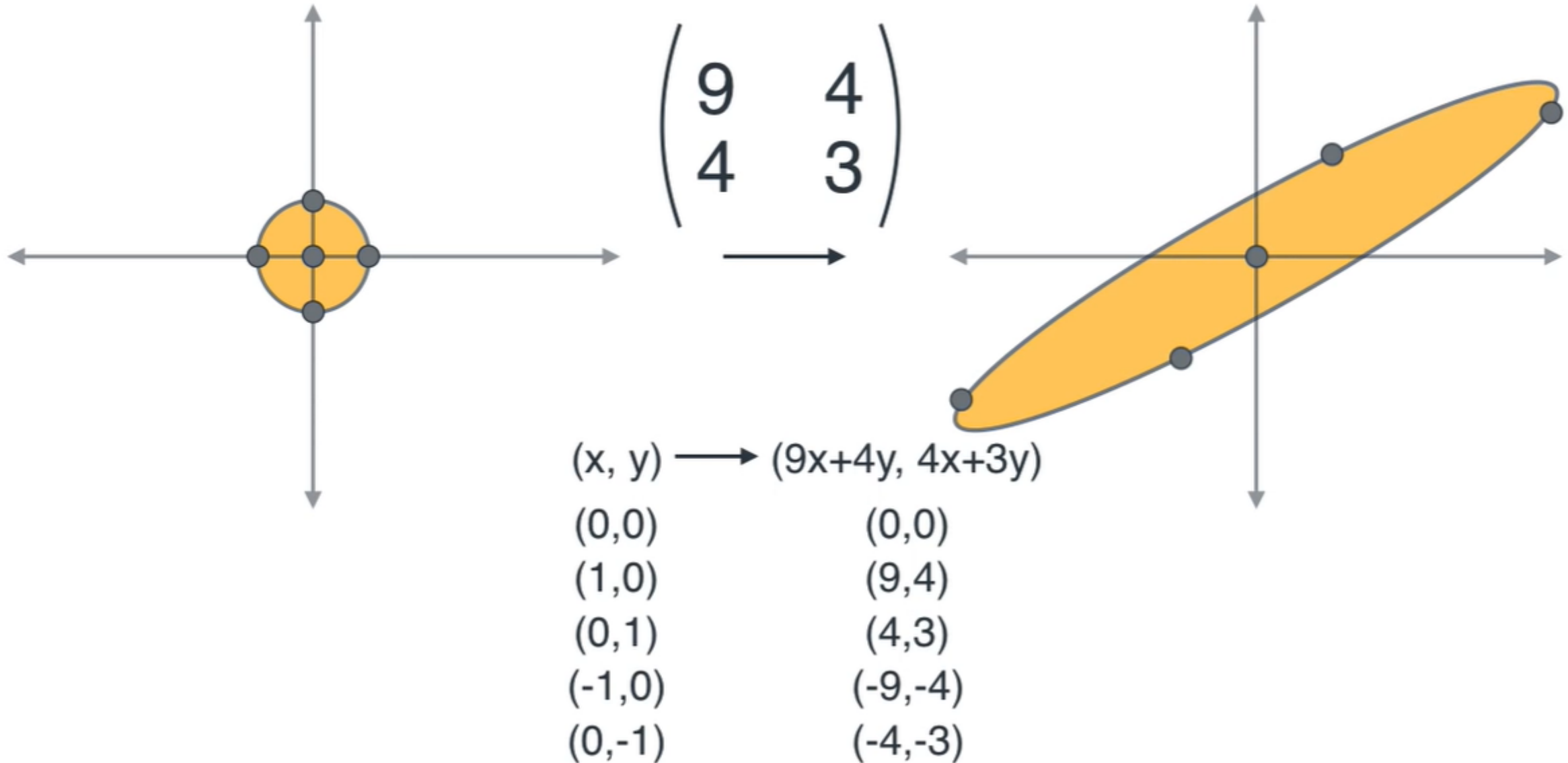
# Covariance matrix

$$\Sigma = \begin{pmatrix} \text{Var}(X) & \text{Cov}(X,Y) \\ \text{Cov}(X,Y) & \text{Var}(Y) \end{pmatrix}$$

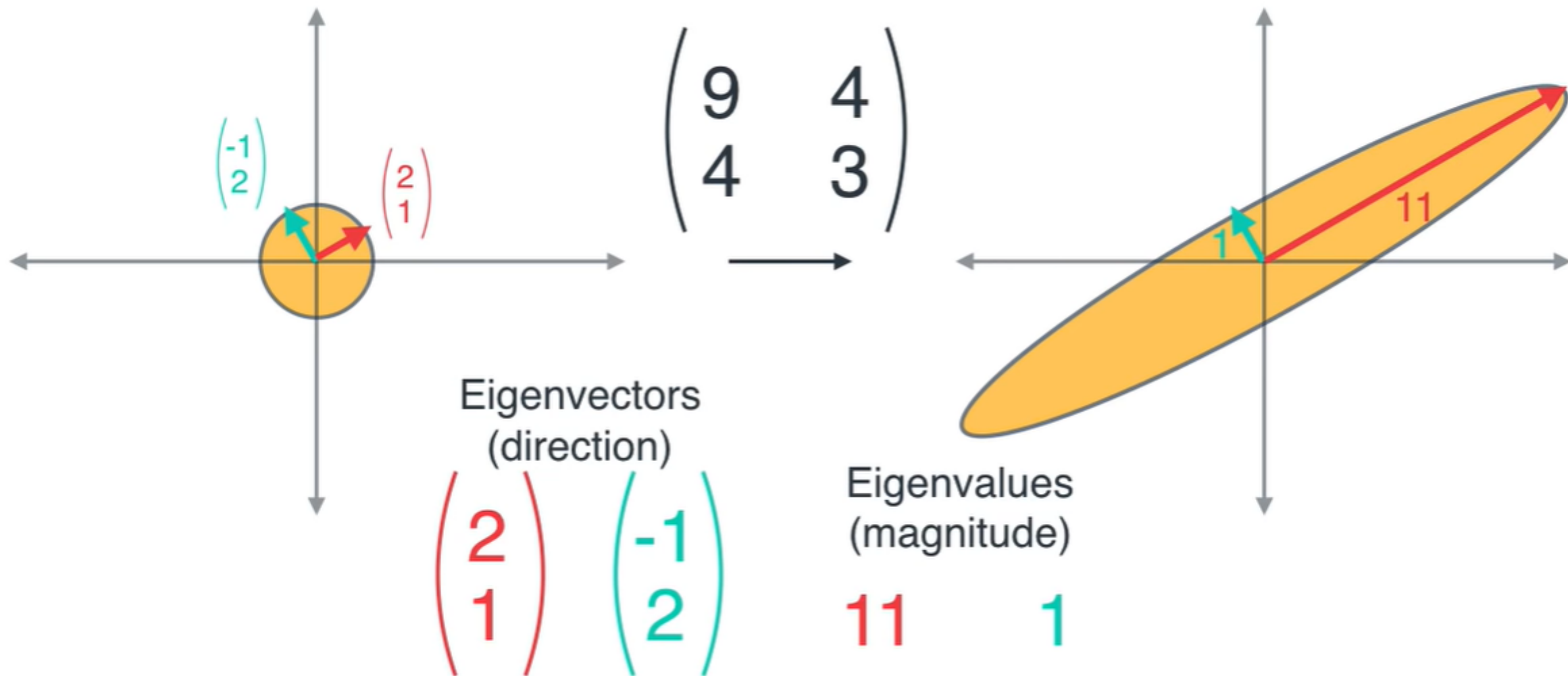


$$\begin{pmatrix} 9 & 4 \\ 4 & 3 \end{pmatrix}$$

# Linear Transformations

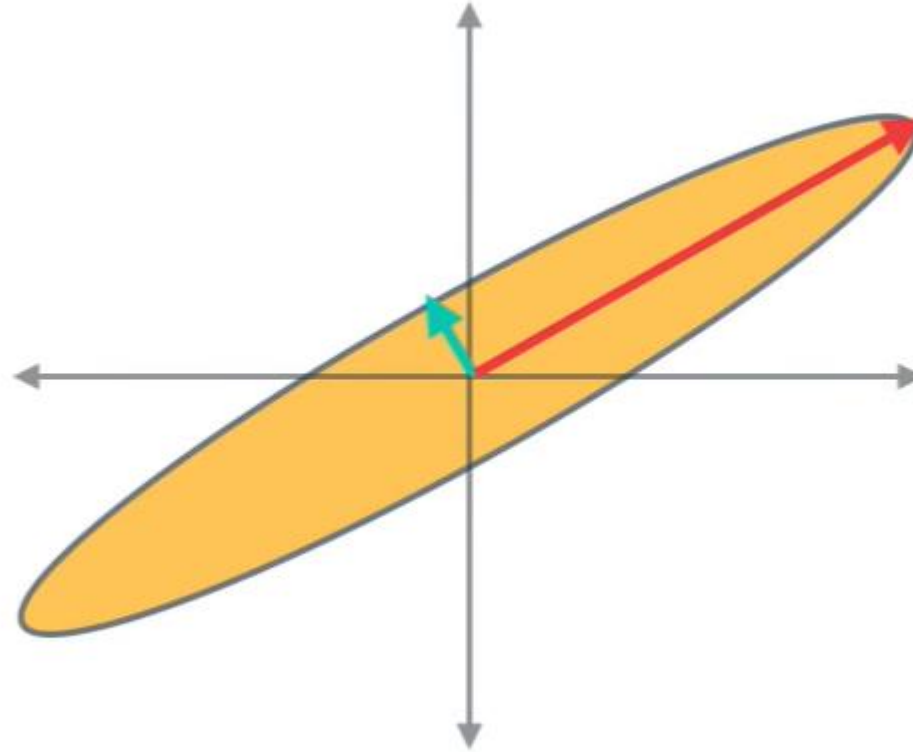


# Linear Transformations





# Linear Transformations



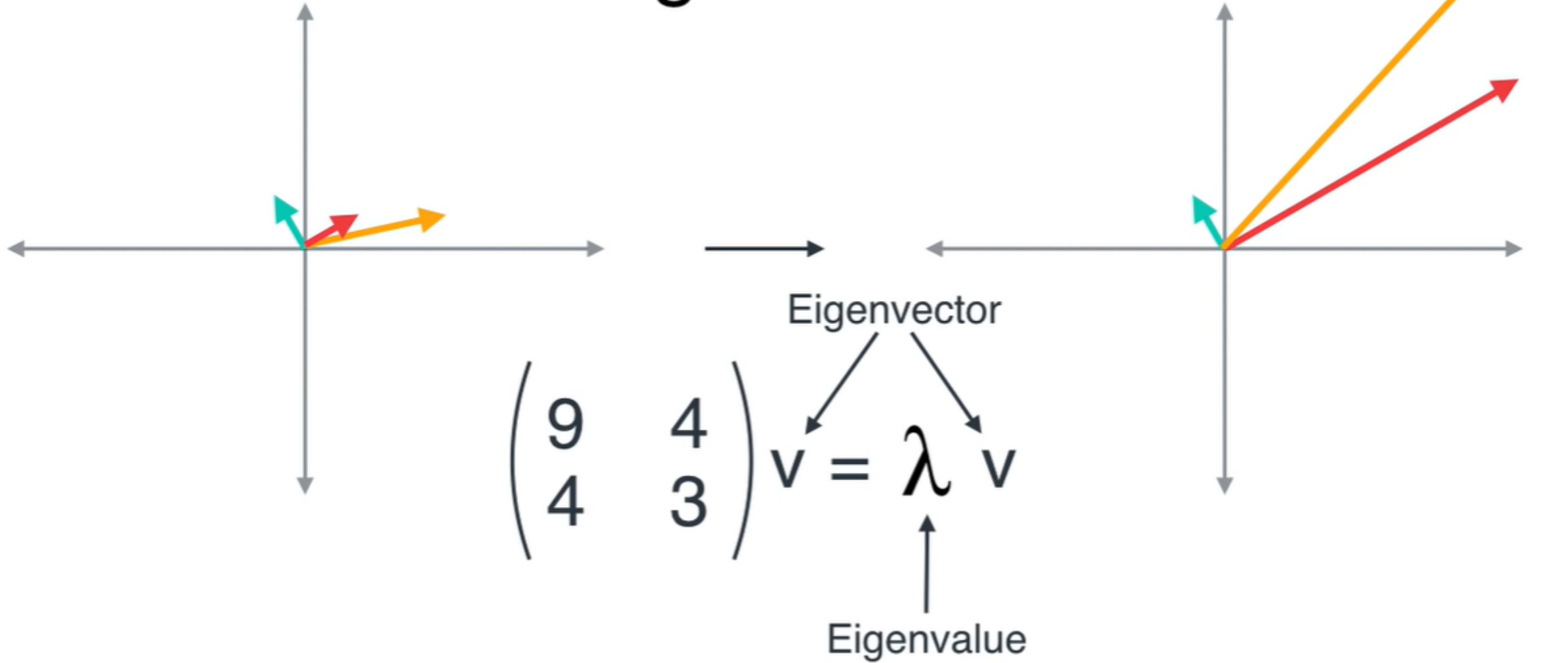
Eigenvectors  
(direction)

$$\begin{pmatrix} 2 \\ 1 \end{pmatrix} \quad \begin{pmatrix} -1 \\ 2 \end{pmatrix}$$

Eigenvalues  
(magnitude)

$$11 \quad 1$$

# Eigenstuff



# Eigenvalues

$$\begin{pmatrix} 9 & 4 \\ 4 & 3 \end{pmatrix}$$

Characteristic Polynomial

$$\begin{vmatrix} x-9 & -4 \\ -4 & x-3 \end{vmatrix} = (x-9)(x-3) - (-4)(-4) = x^2 - 12x + 11 \\ = (x-11)(x-1)$$

Eigenvalues **11** and **1**

# Eigenvalues

$$\begin{pmatrix} 9 & 4 \\ 4 & 3 \end{pmatrix}$$

Characteristic Polynomial

$$\begin{vmatrix} x-9 & -4 \\ -4 & x-3 \end{vmatrix} = (x-9)(x-3) - (-4)(-4) = x^2 - 12x + 11 \\ = (x-11)(x-1)$$

Eigenvalues **11** and **1**

# Eigenvalue

고유값

자료행렬을 요약하는 수치로서, 특성치라고도 한다. 각 고유값은 그에 대응하는 고유벡터가 있다.  $A$ 는  $m \times n$  행렬이고,  $\vec{x}$ 는  $\mathbb{R}^n$ 의 영벡터가 아닌 벡터이다. 스칼라  $\lambda$ 에 대하여  $A\vec{x}$ 가  $\vec{x}$ 의 스칼라  $\lambda$ 배, 즉  $A\vec{x} = \lambda\vec{x}$ 일 때,  $\lambda$ 를  $A$ 의 고유값(eigenvalue of  $A$ )이라 하고,  $\vec{x}$  ( $\vec{x} \neq \vec{0}$ )를  $\lambda$ 에 대응하는  $A$ 의 고유 벡터(eigenvector of  $A$ )라 한다.

예를 들어, 벡터  $\vec{x} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$ 는  $A\vec{x} = \begin{pmatrix} 3 & 0 \\ 8 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \end{pmatrix} = \begin{pmatrix} 3 \\ 6 \end{pmatrix} = 3 \begin{pmatrix} 1 \\ 2 \end{pmatrix} = 3\vec{x}$ 이므로 고유치  $\lambda = 3$ 에 대응하는 행렬  $A = \begin{pmatrix} 3 & 0 \\ 8 & -1 \end{pmatrix}$ 의 고유 벡터이다.  $A$ 가 실수의  $n \times n$  대칭행렬이면  $A$ 의 고유값은 실수이다.  $A$ 는  $n \times n$  행렬일 때  $A$ 의 고유값이  $\lambda$ 이기 위한 필요충분조건은  $\det(A - \lambda I) = 0$ 이다.  $\det(A - \lambda I)$ 는  $n$ 차 다항식이 된다. 이때  $\det(A - \lambda I) = 0$ 을 행렬  $A$ 의 특성방정식(characteristic equation of  $A$ )라고 한다.

$n \times n$  행렬  $A$ 가 서로 다른 고유값을 가지면  $A$ 는 대각화가 가능한 행렬이다. 대각행렬과 닮은 행렬을 “대각화 가능 행렬(diagonalizable matrix)”이라 한다.  $n \times n$  행렬  $A$ 가 대각행렬  $D$ 와 닮았을 때  $A$ 는 ‘대각화 가능하다(bedagonalizable)’라고 하고  $A$ 를 대각화 가능 행렬이라 한다.  $n \times n$  행렬  $A$ 가 대각화 가능 행렬이기 위한 필요충분 조건은  $A$ 가  $n$ 개의 일차독립인 고유벡터를 갖는 것이다.

행렬  $A$ 의 서로 다른 고유값  $\lambda_1, \lambda_2, \dots, \lambda_m$ 에 대응하는 고유벡터가  $\vec{x}_1, \vec{x}_2, \dots, \vec{x}_m$ 일 때  $\vec{x}_1, \vec{x}_2, \dots, \vec{x}_m$ 는 일차독립이다. SVD(Singular Value Decomposition, 특이값 분해), Pseudo-Inverse, 선형연립방정식의 풀이, PCA(Principal component analysis, 주성분분석) 등의 주요 응용이 eigenvalue, eigenvector를 그 밑바탕에 깔고 있다.

# Eigenvalues

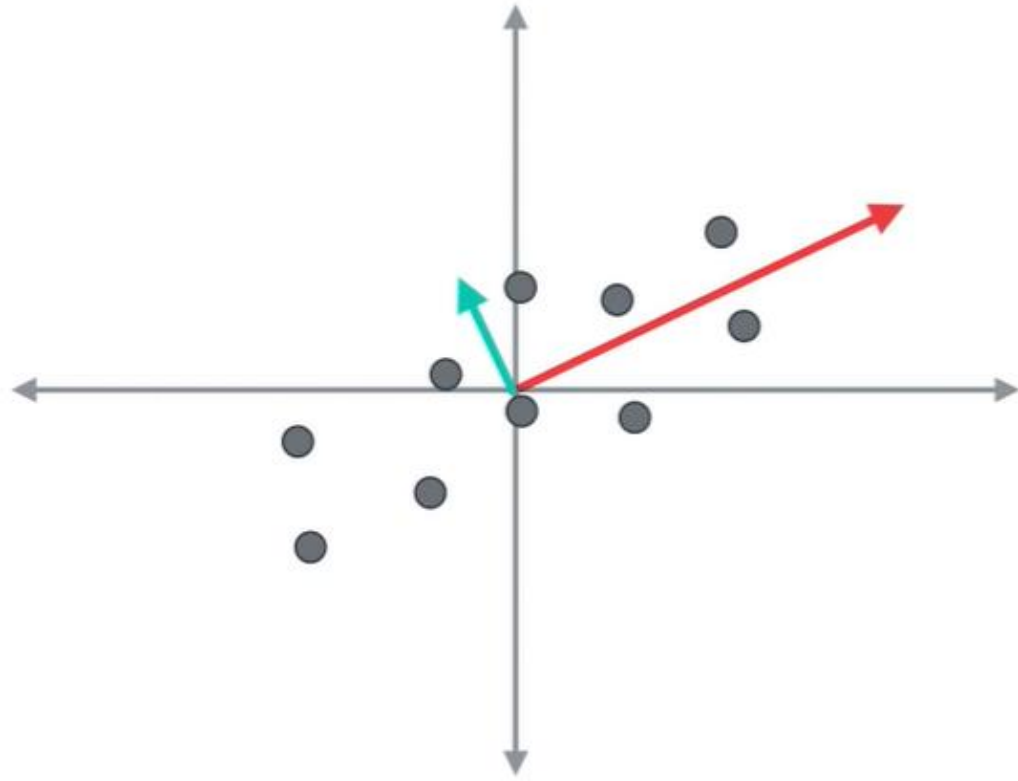
$$\begin{pmatrix} 9 & 4 \\ 4 & 3 \end{pmatrix}$$

Characteristic Polynomial

$$\begin{vmatrix} x-9 & -4 \\ -4 & x-3 \end{vmatrix} = (x-9)(x-3) - (-4)(-4) = x^2 - 12x + 11 \\ = (x-11)(x-1)$$

Eigenvalues **11** and **1**

# Principal Component Analysis (PCA)



$$\Sigma = \begin{pmatrix} 9 & 4 \\ 4 & 3 \end{pmatrix}$$

$$\begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

$$11$$

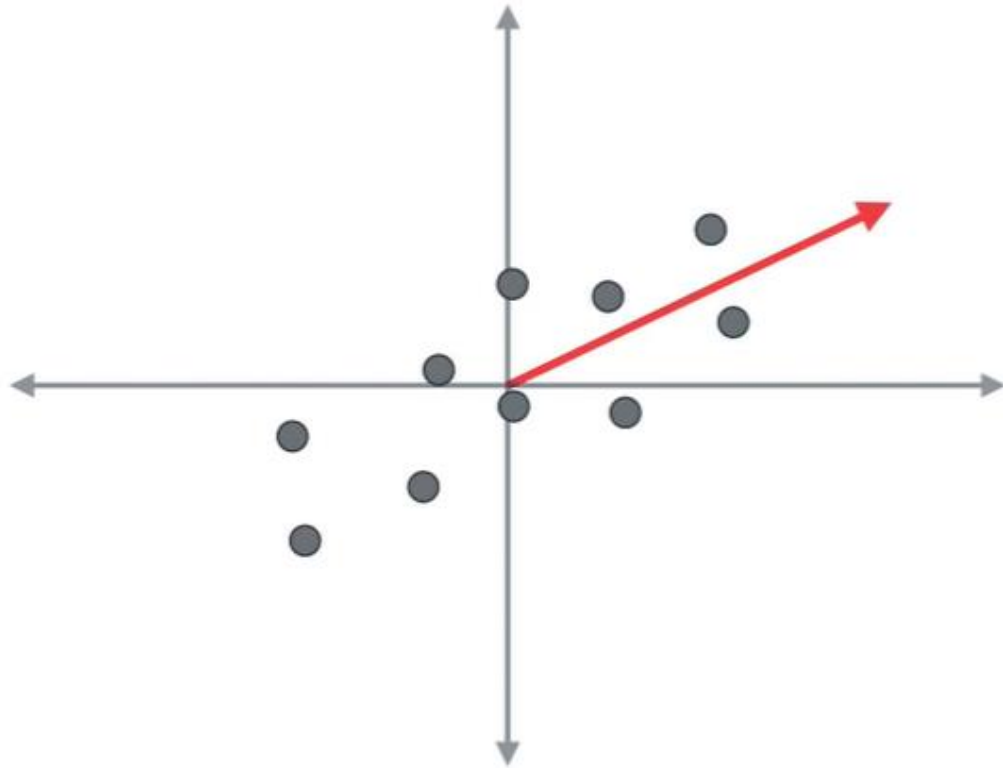
$$\begin{pmatrix} -1 \\ 2 \end{pmatrix}$$

$$1$$

Eigenvectors  
(direction)

Eigenvalues  
(magnitude)

# Principal Component Analysis (PCA)



$$\Sigma = \begin{pmatrix} 9 & 4 \\ 4 & 3 \end{pmatrix}$$

$$\begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

Eigenvectors  
(direction)

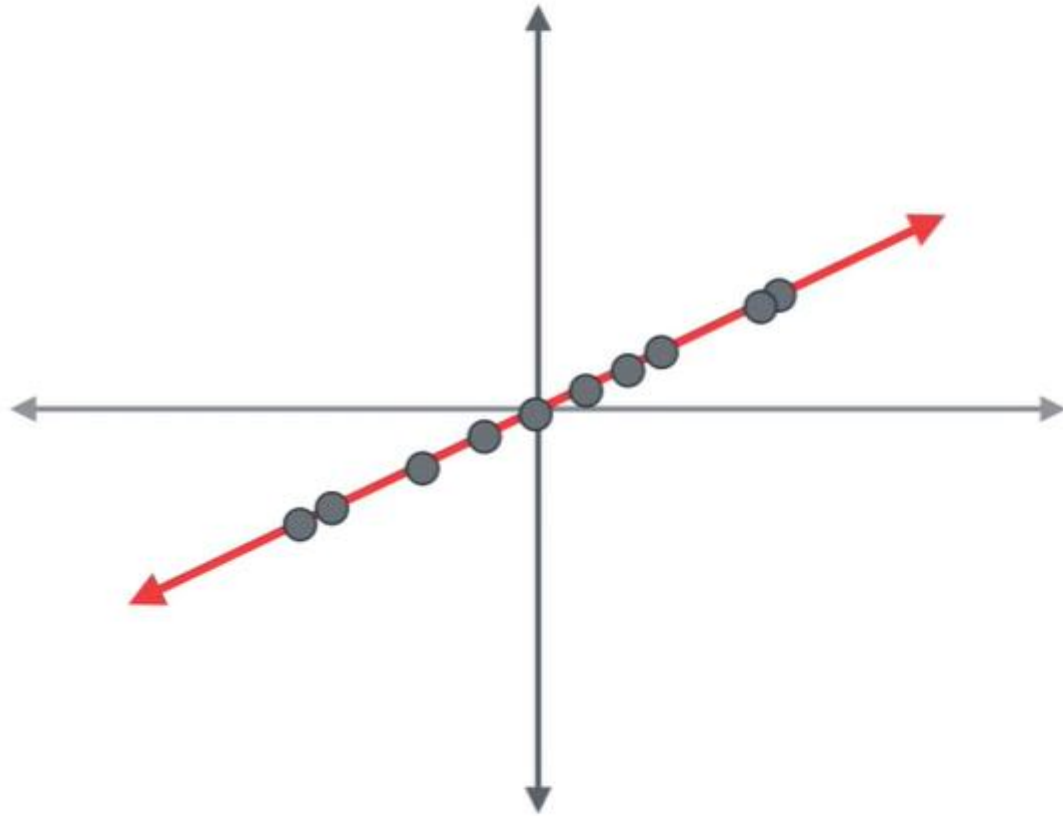
$$11$$

Eigenvalues  
(magnitude)





# Principal Component Analysis (PCA)



$$\Sigma = \begin{pmatrix} 9 & 4 \\ 4 & 3 \end{pmatrix}$$

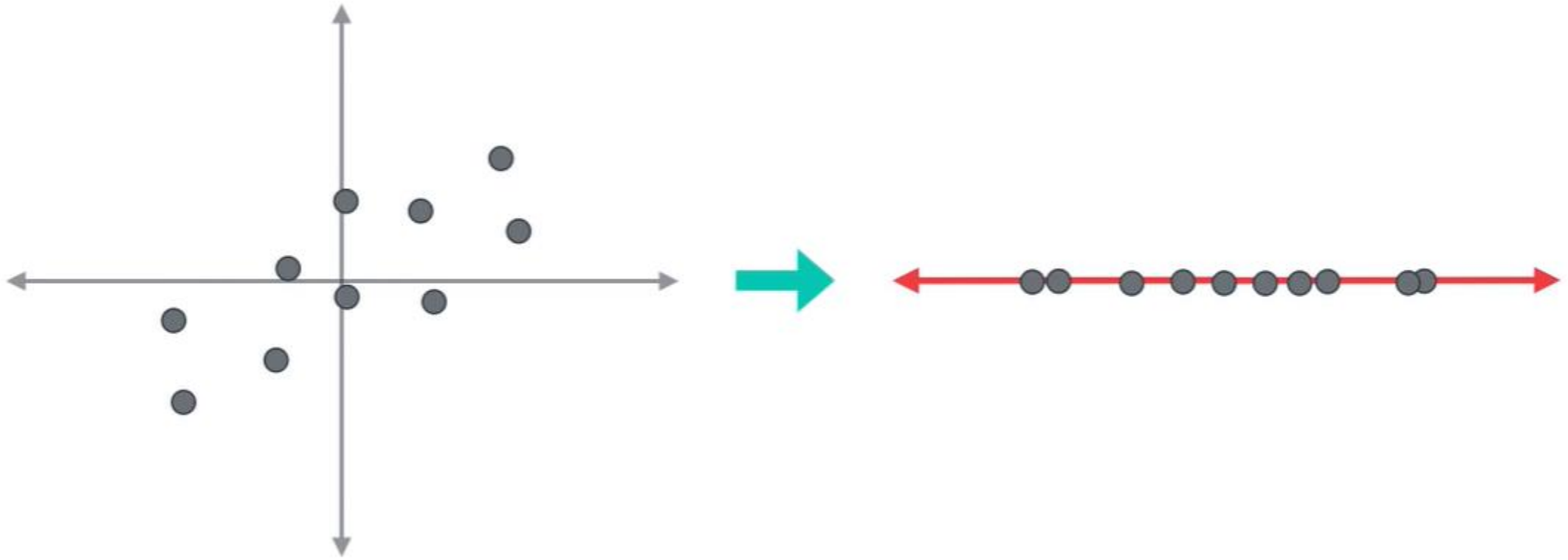
$$\begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

Eigenvectors  
(direction)

$$11$$

Eigenvalues  
(magnitude)

# Principal Component Analysis (PCA)



# PCA

Large Table

X1	X2	X3	X4	X5
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*

Covariance matrix

$$\begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{pmatrix}$$

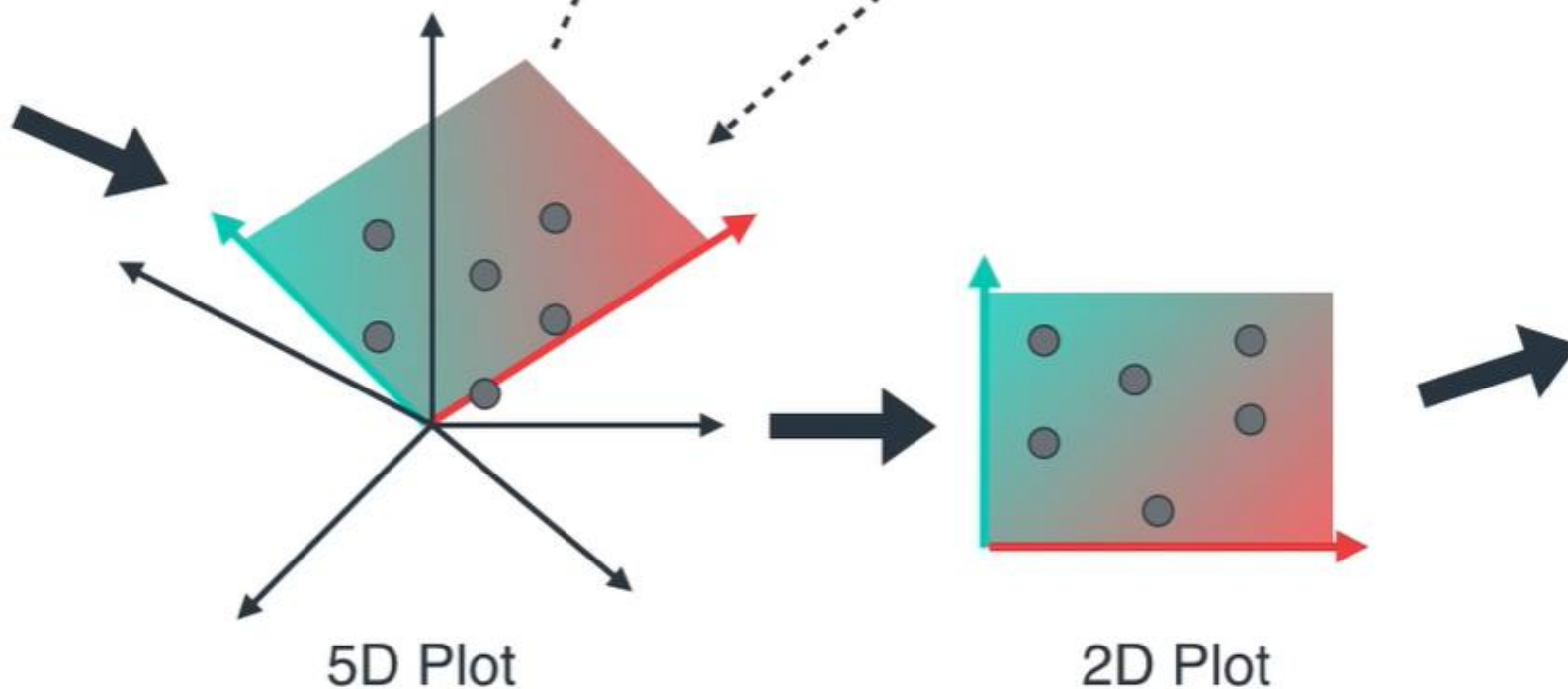
Eigenstuff

$V_1$   $\lambda_1$   
 $V_2$   $\lambda_2$

Big  
Small

Small Table

W1	W2
*	*
*	*
*	*
*	*
*	*
*	*
*	*
*	*
*	*
*	*
*	*
*	*
*	*
*	*
*	*



# 목표 변수

1. 매출

2. 팁

3. 요일별 팀 Size 수

4. Tip rate

5. 요일별 방문 예상 팀 수

## 강사 소개

정 준 수 / Ph.D ( heinem@naver.com )

- 前) 삼성전자 연구원
- 前) 삼성의료원 (삼성생명과학연구소)
- 前) 삼성SDS (정보기술연구소)
- 現) (사)한국인공지능협회, AI, 머신러닝 강의
- 現) 한국소프트웨어산업협회, AI, 머신러닝 강의
- 現) 서울디지털재단, AI 자문위원
- 現) 한성대학교 교수(겸)
- 전문분야: 시각 모델링, 머신러닝(ML), RPA
- <https://github.com/JSJeong-me/>

