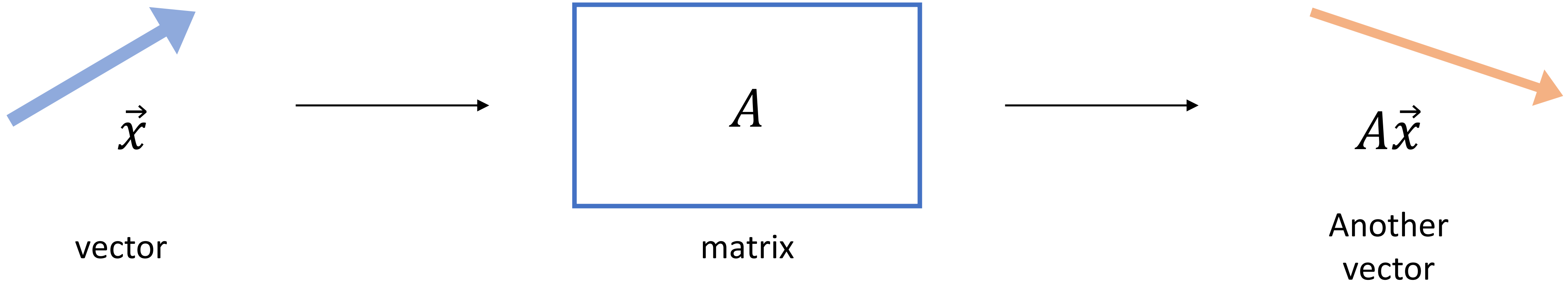
공분산 행렬의 의미와 주성분 분석(PCA)

공돌이의 수학정리노트

잠깐 복습

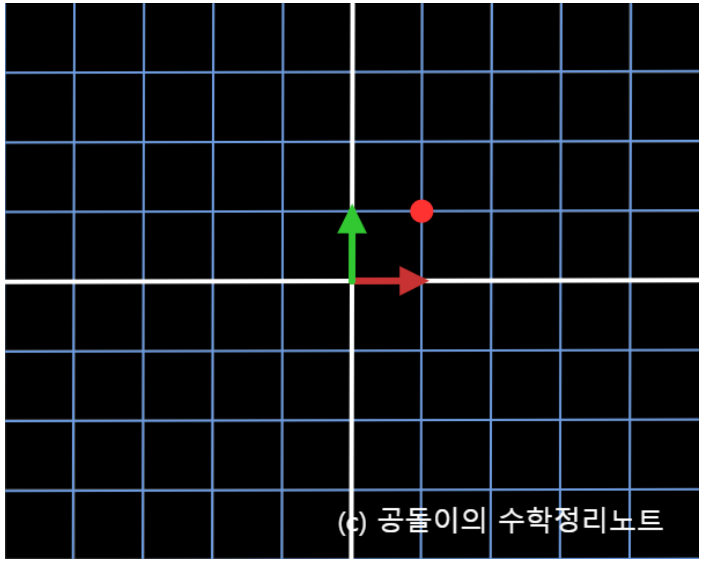
- 행렬은 선형변환이다

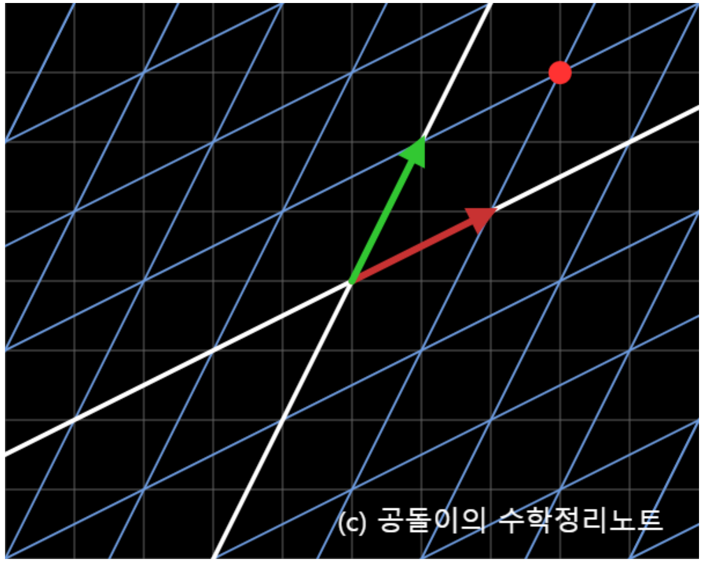


여러 선형 변환의 시각적 예시

- Shearing

$$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

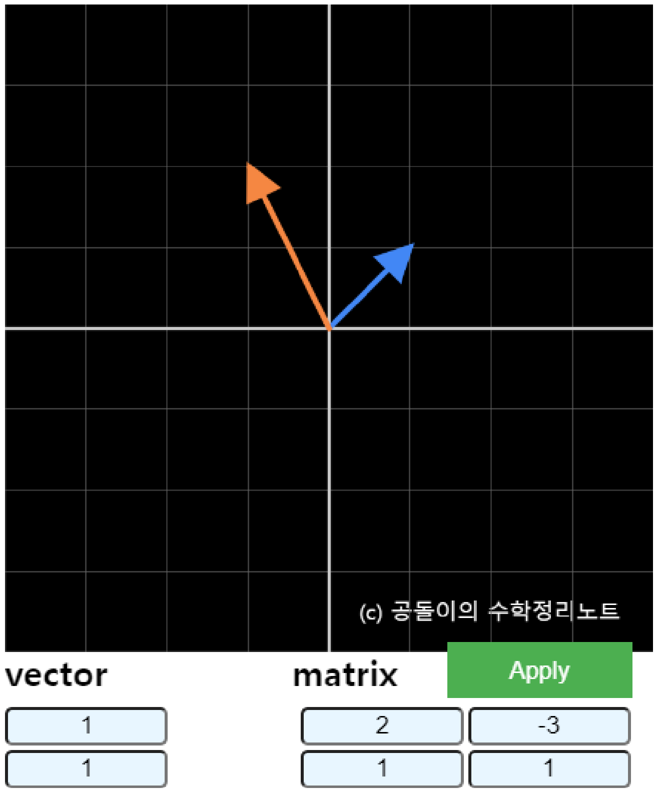




https://angeloyeo.github.io/p5/Matrix_as_a_linear_transformation/shear/

임의의 벡터에 대한 선형 변환

- JS Applet 시연

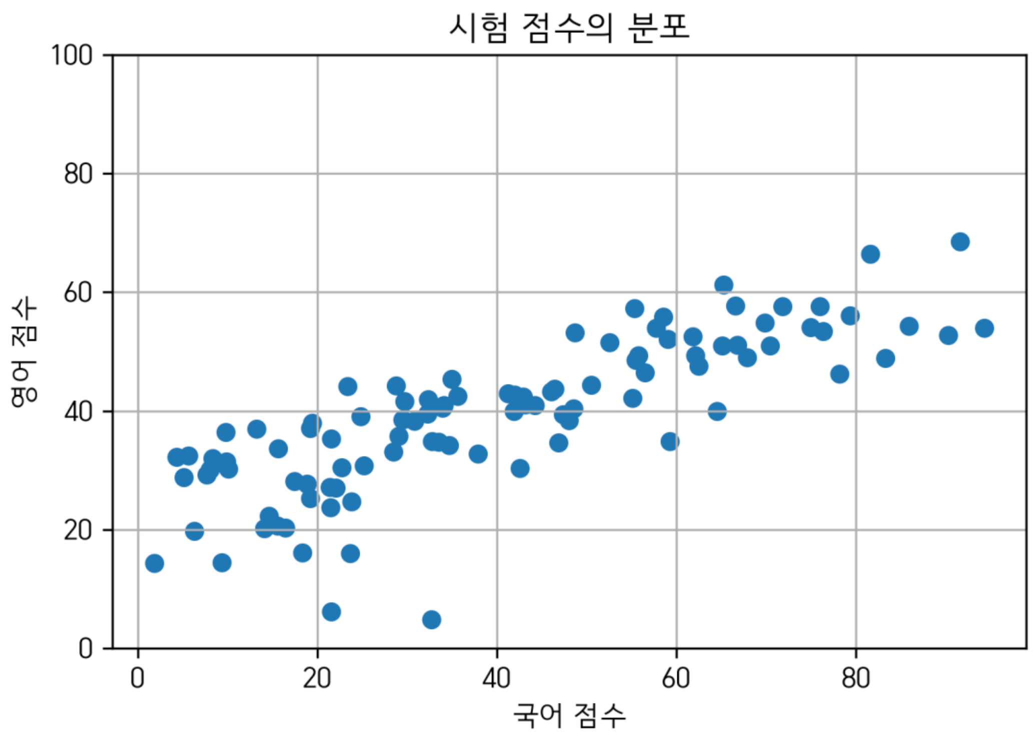


목차

- PCA의 원시적 의미
- 공분산 행렬의 의미
  - 기하학적 의미: 선형 변환(shearing)
  - 데이터 구조적 의미: 각 feature의 변동이 얼마나 닮았나?
- PCA
  - 공분산 행렬의 Eigenvector & eigenvalue
  - PCA = Projecting data onto eigenvectors of 공분산 행렬

PCA의 원시적 의미

- PCA는 **종합점수**를 ‘잘’ 계산하는 방법
- 100명 학생들의 국어 시험 성적과 영어 시험 성적



국어 점수	영어 점수
100	83
70	50
30	25
45	30
⋮	⋮
80	60

PCA의 원시적 의미

- 이 두 시험 점수에 대한 종합점수를 잘 내려면?
  1. 두 점수의 평균값
  2. 국어:영어=6:4 비중으로 한 종합 점수
     (영어가 좀 더 어려웠으니까…)

국어 점수	영어 점수
100	83
70	50
30	25
45	30
⋮	⋮
80	60

PCA의 원시적 의미

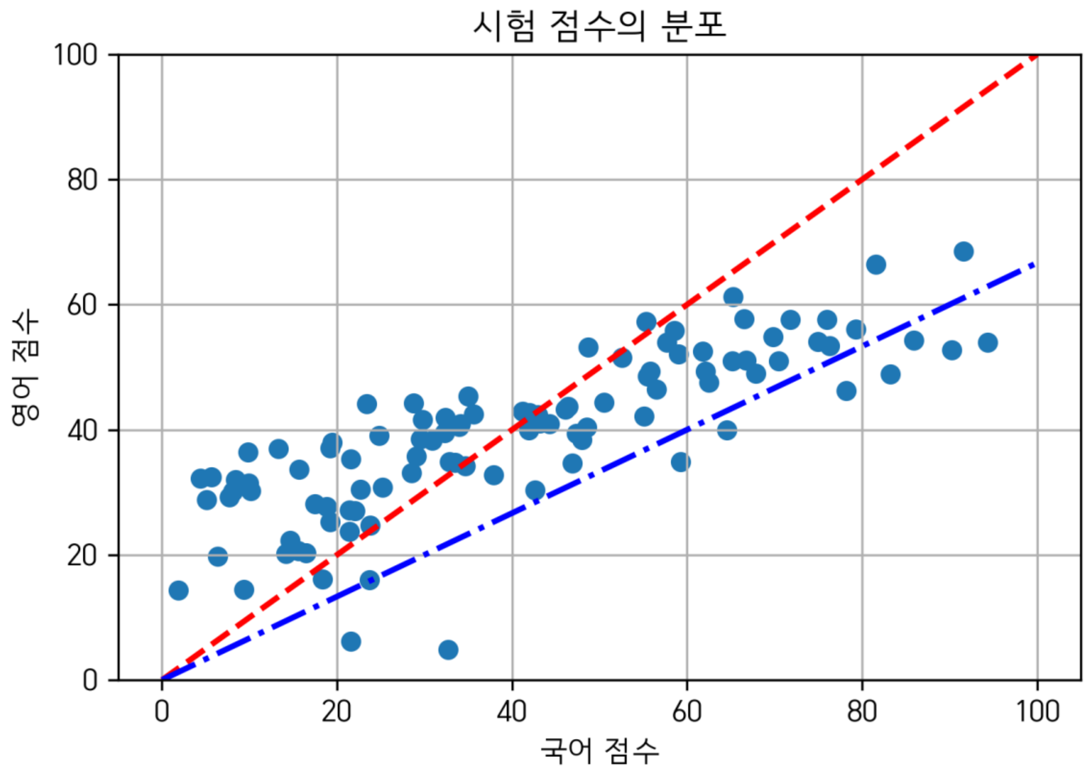
- 수학적으로 표현하면,
- 예를 들어 1번 학생에 대해서,
  1. 평균값 내기: $100 \times 0.5 + 83 \times 0.5$
  2. 6:4 비중으로: $100 \times 0.6 + 83 \times 0.4$

- 다른 말로 하면, 두 벡터의 내적으로 볼 수 있음.
  - $[100, 83]^T \cdot [0.5, 0.5]^T$
  - $[100, 83]^T \cdot [0.6\ 0.4]^T$

국어 점수	영어 점수
100	83
70	50
30	25
45	30
⋮	⋮
80	60

PCA의 원시적 의미

- 그림으로 표현하면 아래와 같이 [0.5, 0.5] 벡터 혹은 [0.6, 0.4] 벡터에 내적을 취하는 것과 같다.



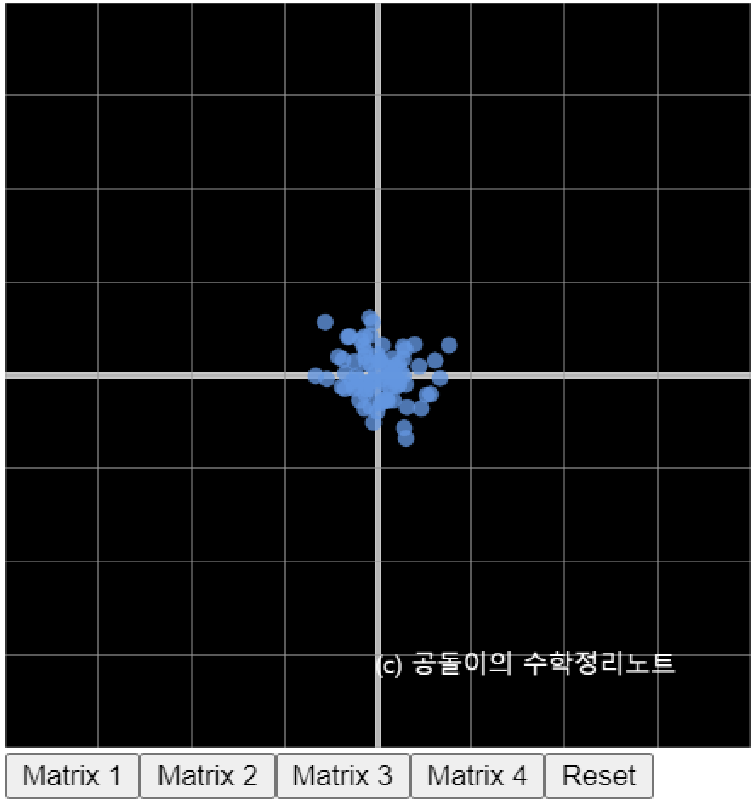
PCA의 원시적 의미

- 그렇다면, 우리가 고민해야 할 문제는?
  - 데이터 벡터를 어떤 벡터에 내적(혹은 정사영)하는 것이 최적의 결과를 얻어주는가?
- 또, 부차적으로는,
  - 기왕 정사영 할 벡터(혹은 축)를 찾는데, 데이터 분포의 중심을 중심축으로 하는 벡터를 찾는게 좋지 않을까?

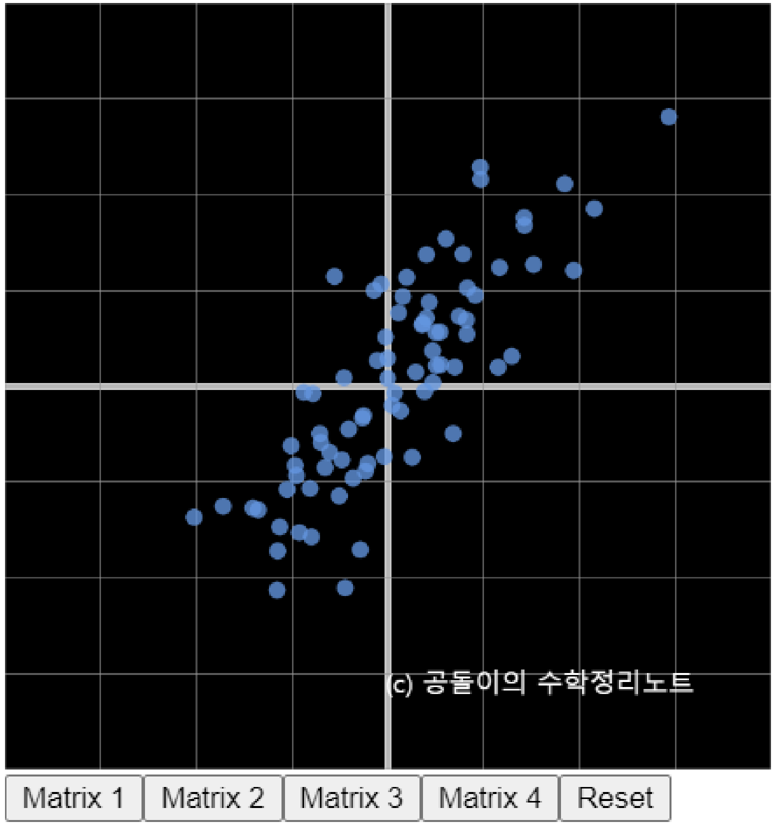
➔ '데이터의 구조'를 기술하는 '공분산 행렬'로 부터 해답을 얻자.

공분산 행렬의 기하학적 의미

- 행렬=선형변환, 벡터 공간을 다른 벡터 공간으로 mapping.



$$\begin{bmatrix} 3 & 2 \\ 2 & 4 \end{bmatrix}$$



공분산 행렬의 기하학적 의미

x축 방향으로 퍼진 정도

x, y축 방향으로
함께 퍼진 정도

$$\begin{bmatrix} 3 & 2 \\ 2 & 4 \end{bmatrix}$$

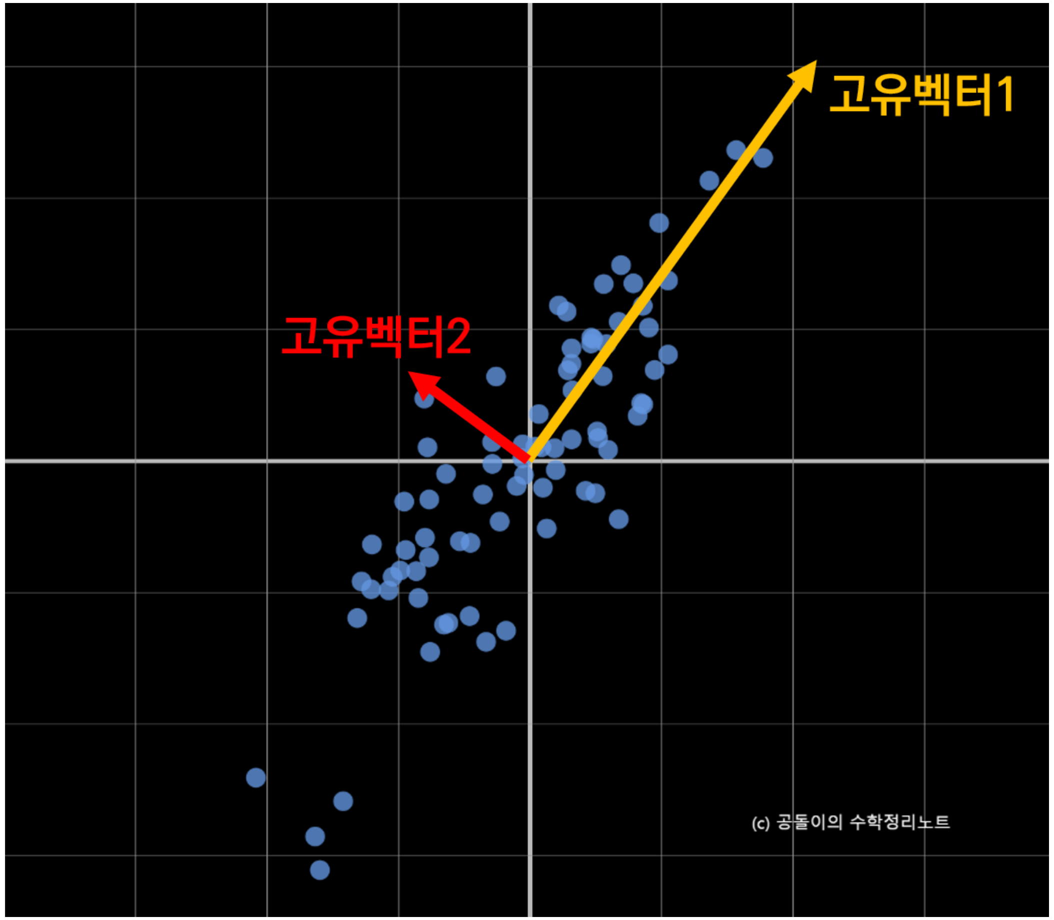
x, y축 방향으로
함께 퍼진 정도

y축 방향으로 퍼진 정도

공분산 행렬과 고유벡터

- 고유벡터의 의미
  - 그 행렬이 벡터의 변화에 작용하는 주축(principal axis)의 방향을 나타내줌.
  - 즉, 공분산 행렬의 고유벡터는 데이터가 어떤 방향으로 분산되어 있는지를 찾아줌.
- 고윳값의 의미
  - 고유벡터 방향으로 얼마만큼의 크기로 벡터공간이 늘려지는지를 의미.
  - 즉, 고윳값이 큰 순서대로 고유벡터를 정렬하면 결과적으로 중요한 순서대로 주성분을 구성하게 됨.

공분산 행렬과 고윳값, 고유벡터



공분산 행렬의 수식적 의미

- 데이터 구조적 의미: 각 feature의 변동이 얼마나 닮았나?

- 다음과 같은 data matrix를 생각해보자.
- 이 data matrix에서 행은 samples, 열은 features를 의미한다.
- 또, 열(features)들의 평균 값은 0이라고 하자 (중요).
  (즉, feature들의 평균값은 빼준 상태)

$$X = \begin{pmatrix} | & | & | & & | \\ X_1 & X_2 & X_3 & \cdots & X_d \\ | & | & | & & | \end{pmatrix} \in \mathbb{R}^{n \times d}$$

공분산 행렬의 수식적 의미
- 데이터 구조적 의미: 각 feature의 변동이 얼마나 닮았나?

- 데이터 행렬 X를 이용해 공분산 행렬을 구해보자.
- 닮은 정도를 보기 위해선 내적이 필요함.
- 그 과정은 다음과 같이 행렬의 곱으로 표현할 수 있음.

$$X^T X = \begin{pmatrix} \text{——} & X_1 & \text{——} \\ \text{——} & X_2 & \text{——} \\ & \cdots & \\ \text{——} & X_d & \text{——} \end{pmatrix} \begin{pmatrix} | & | & & | \\ X_1 & X_2 & \cdots & X_d \\ | & | & & | \end{pmatrix}$$

공분산 행렬의 수식적 의미

– 데이터 구조적 의미: 각 feature의 변동이 얼마나 닮았나?

$$X^T X = \begin{pmatrix} \text{———} & X_1 & \text{———} \\ \text{———} & X_2 & \text{———} \\ & \cdots & \\ \text{———} & X_d & \text{———} \end{pmatrix} \begin{pmatrix} | & | & & | \\ X_1 & X_2 & \cdots & X_d \\ | & | & & | \end{pmatrix}$$

$$= \begin{pmatrix} dot(X_1, X_1) & dot(X_1, X_2) & \cdots & dot(X_1, X_d) \\ dot(X_2, X_1) & dot(X_2, X_2) & \cdots & dot(X_2, X_d) \\ \vdots & \vdots & \ddots & \vdots \\ dot(X_d, X_1) & dot(X_d, X_2) & \cdots & dot(X_d, X_d) \end{pmatrix}$$

공분산 행렬의 수식적 의미

– 데이터 구조적 의미: 각 feature의 변동이 얼마나 닮았나?

$$\frac{X^T X}{\boxed{n}} = \frac{1}{n}\begin{pmatrix} dot(X_1, X_1) & dot(X_1, X_2) & \cdots & dot(X_1, X_d) \\ dot(X_2, X_1) & dot(X_2, X_2) & \cdots & dot(X_2, X_d) \\ \vdots & \vdots & \ddots & \vdots \\ dot(X_d, X_1) & dot(X_d, X_2) & \cdots & dot(X_d, X_d) \end{pmatrix}$$

Sample Covariance를 계산하기 위해선 $n-1$로 나누기도 함.

공분산 행렬의 수식적 의미
– 데이터 구조적 의미: 각 feature의 변동이 얼마나 닮았나?

- $(X^TX)_{ij}$는 i번째 feature와 j번째 feature의 변동이 닮은 정도를 말해주고 있음.
- 그런데 문제점은, sample 수(n)가 많아질 수록 그 값이 커진다는 것임.
- 따라서, 이 문제점을 방지해주기 위해 값을 n으로 나눠주면,
  아래와 같은 결과를 얻을 수 있음.

$$\Sigma = cov(X) \equiv \frac{X^TX}{n}$$

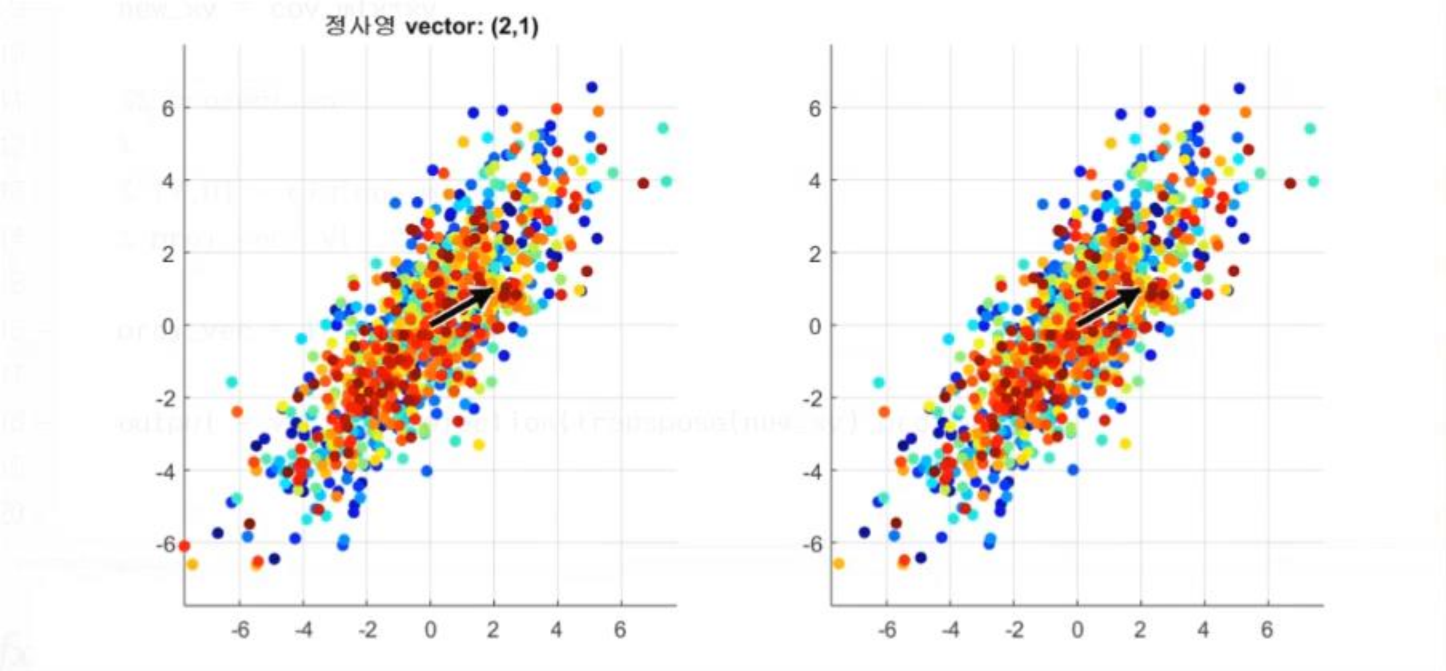
- 즉, covariance matrix의 i 번째 행과 j 번째 행은
  i번 feature와 j번 feature가 서로 함께 변하는 정도를 의미하고 있음.

※ feature들의 평균을 0으로 만들었다는 사실을 잊으면 안됨.

Principal Component Analysis

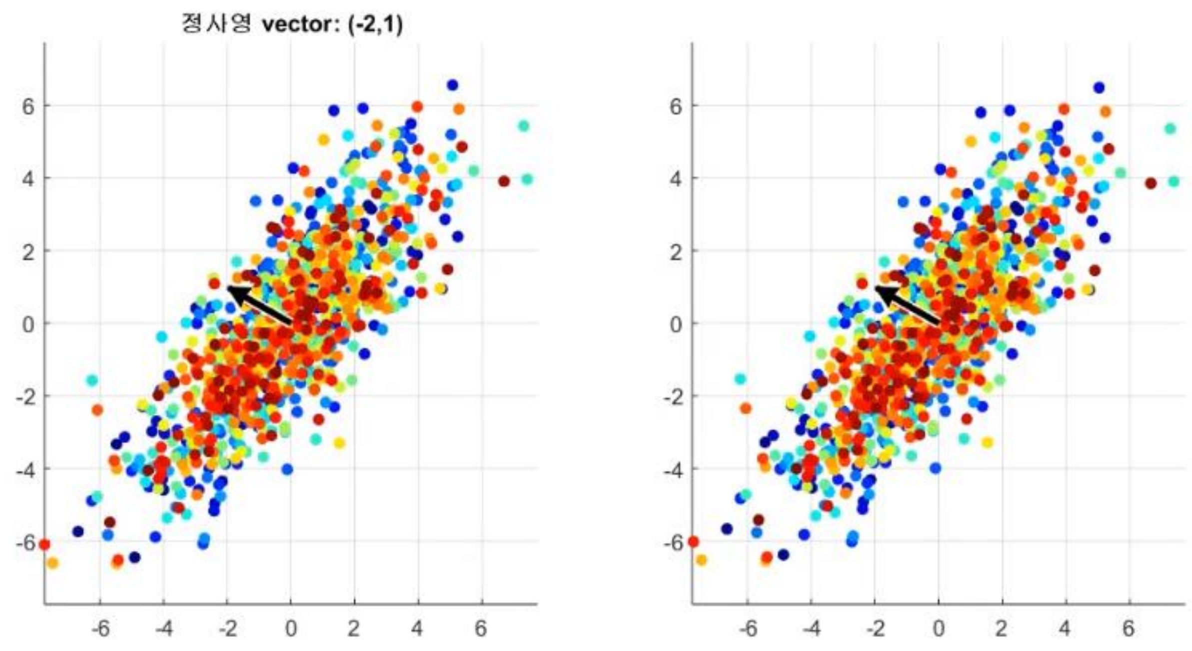
–공분산 행렬의 eigenvector

- 임의로 (2,1) 이라는 벡터에 정사영 해보자.



Principal Component Analysis
-공분산 행렬의 eigenvector

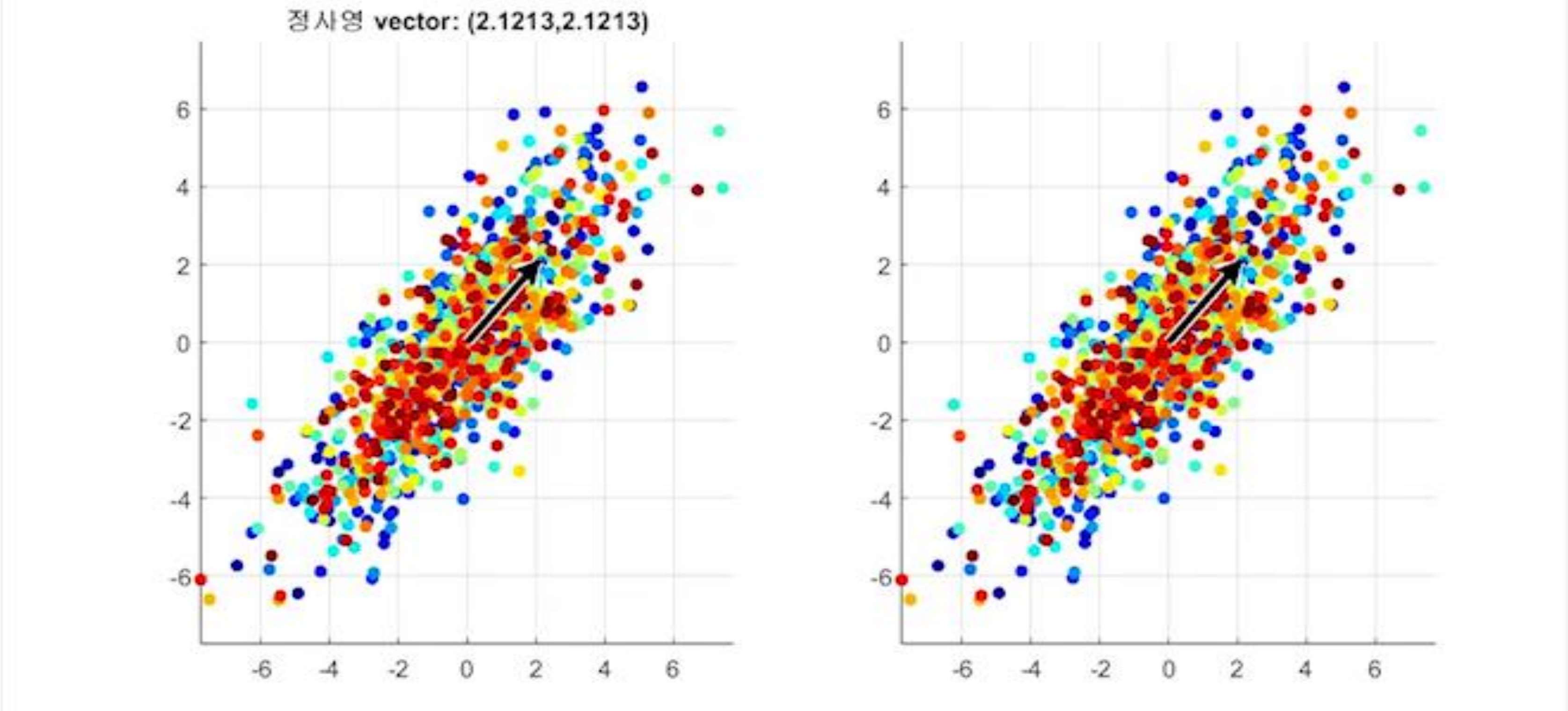
- 임의로 (-2,1) 이라는 벡터에 정사영 해보자.



Principal Component Analysis

–PCA = Projecting data onto eigenvectors of 공분산 행렬

- eigenvector(혹은 주축)에 정사영한 결과



Principal Component Analysis

-PCA = Projecting data onto eigenvectors of 공분산 행렬

- 3차원 데이터의 경우에는?
  - 3개의 eigenvector가 나옴.
  - 이 중 eigenvalue가 큰 두 개의 eigenvector가 이루는 평면에 데이터를 정사영할 수 있음. 이를 통해 2차원으로 차원 감소된 데이터를 획득할 수 있음.
  - MATLAB 시연
- N차원 데이터의 경우에는?
  - N개의 eigenvector가 나옴.
  - 만약 k차원까지 데이터를 차원 감소 하고자 한다면, eigenvalue가 큰 k개의 eigenvector가 이루는 평면에 데이터를 정사영하여 k차원으로 감소된 데이터를 획득할 수 있음.