11주: 기계학습(1)

김 남 규 (ngkim@deu.ac.kr)

11주: 기계학습(1)

1 학습(learning)

김 남 규 (ngkim@deu.ac.kr)

기계 학습(Machine Learning)

- By Mitchell, 1997
  - “A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.
  - 어떤 컴퓨터 프로그램이 T라는 작업을 수행한다.
    이 프로그램의 성능을 P라는 척도로 평가했을 때
    경험 E를 통해 성능이 개선된다면 이 프로그램은 학습을 한다고 말할 수 있다

- 영상 처리 및 컴퓨터 비전에 대입
  - T = 분류(인식)
  - E = 학습 집합
  - P = 인식률

학습 과정

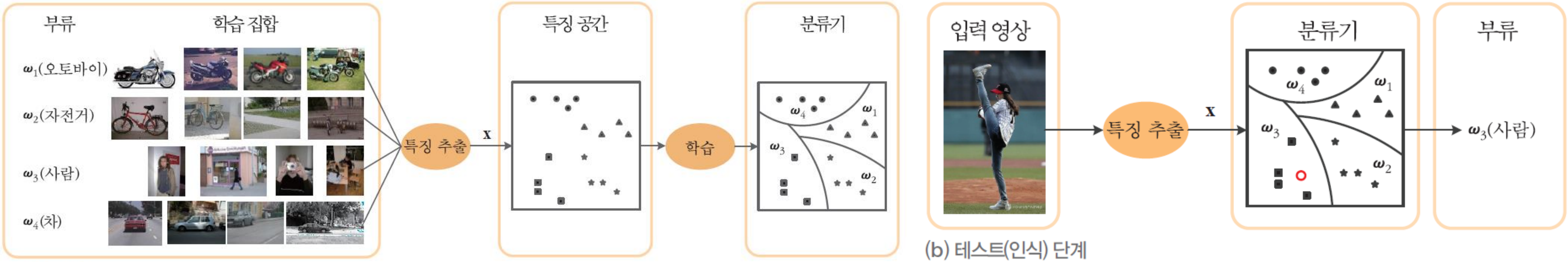
- 학습 단계: 외부 지도에 따라 배우는 과정 (학습 집합 사용)
- 테스트 단계: 현장에서 성능을 평가 하는 과정 (테스트 집합 사용)

(a) 사람의 학습 (b) 기계의 학습

- 지도(Supervised) vs 비지도(Unsupervised)학습

지도 학습

- 부류 정보를 가진 샘플 (x, t)로 구성된 학습 집합 사용
- x는 특징 벡터이고 t는 x가 속한 부류
- 입력과 출력간의 관계를 함수로 학습, 예) 필기 숫자 분류, 영상 분류



(a) 학습 단계
(b) 테스트(인식) 단계

비지도 학습

- 지도 학습에서 사용했던 (x,t) 중에 부류 정보 t가 없는 상황의 학습
- 학습 결과물의 형태를 정확히 알지 못하는 경우, 예) 클러스터링(사전 지식 없음)
- 비지도 학습 예
  - 유사한 특징 벡터들을 끼리끼리 모으는 군집화 수행 (K-means, 신경망, Mean Shift , PCA/Eigenface 등의 군집화 알고리즘)
  - 군집에서 유용한 정보 추출 (데이터 마이닝, 빅데이터, 정보 검색 등의 많은 응용)

- 준지도 학습(Semi-supervised Learning)
  - 부류 정보가 있는 샘플과 없는 샘플이 섞여 있는 상황의 학습
  - 최근 중요성 부각: 아직 부류 정보를 부여하지 못한 샘플이 시시각각 인터넷에서 발생하기 때문
  - 부류 정보가 있는 샘플로 학습한 후 부류 정보가 없는 샘플의 부류 정보를 추정
  - 추정된 정보로 반복 학습

11주: 기계학습(1)

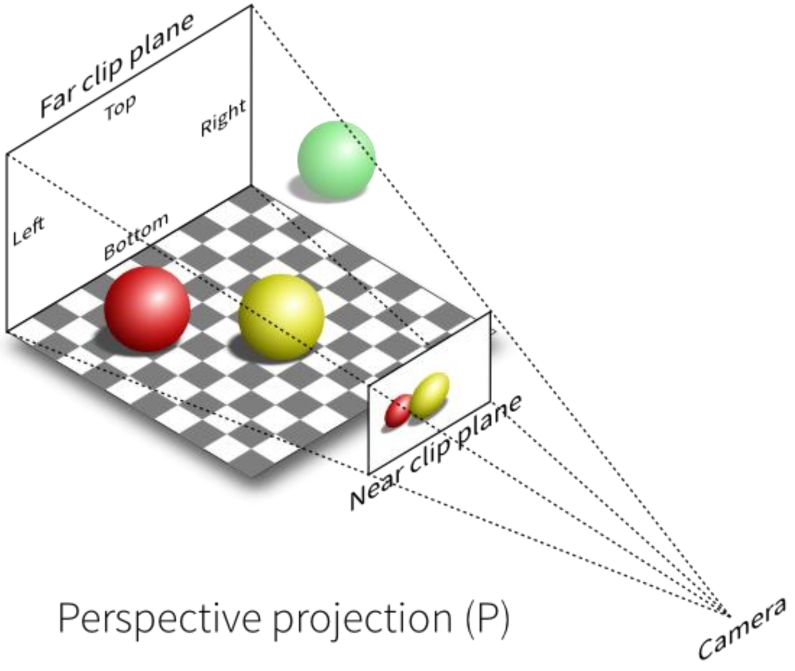
2 주성분 분석(PCA) & 고유 얼굴(Eigenface)

김 남 규 (ngkim@deu.ac.kr)

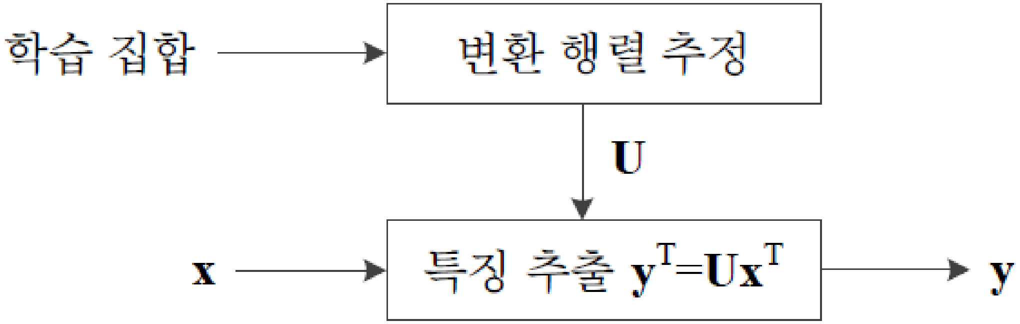
주성분 분석

- Principle Component Analysis, PCA
- 기본 개념: 고차원의 정보(벡터)를 저차원으로 축소

- 원리
  - 학습 집합 $X=\{x_1, x_2, x_3, \cdots, x_n\}$로 변환 행렬 U를 추정
  - U는 $d*D$로서 D차원의 x를 d차원의 y로 변환

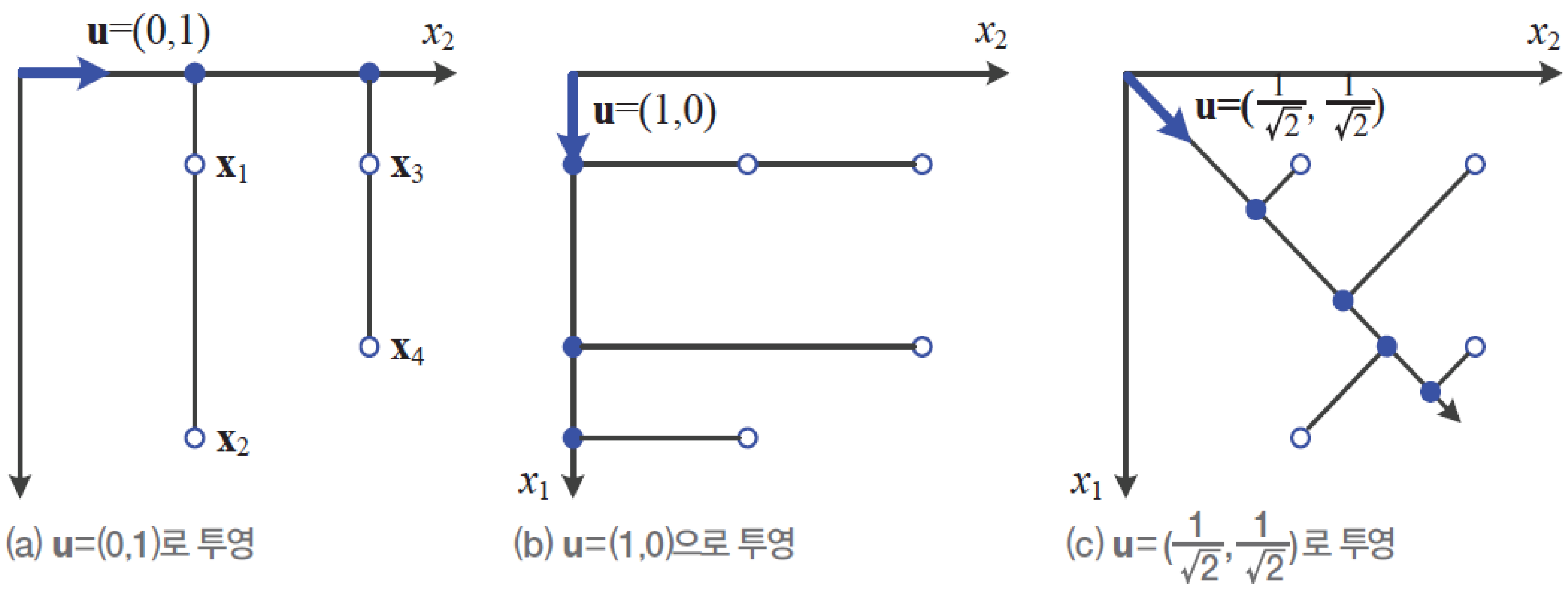


Perspective projection (P)



차원의 축소와 정보 손실

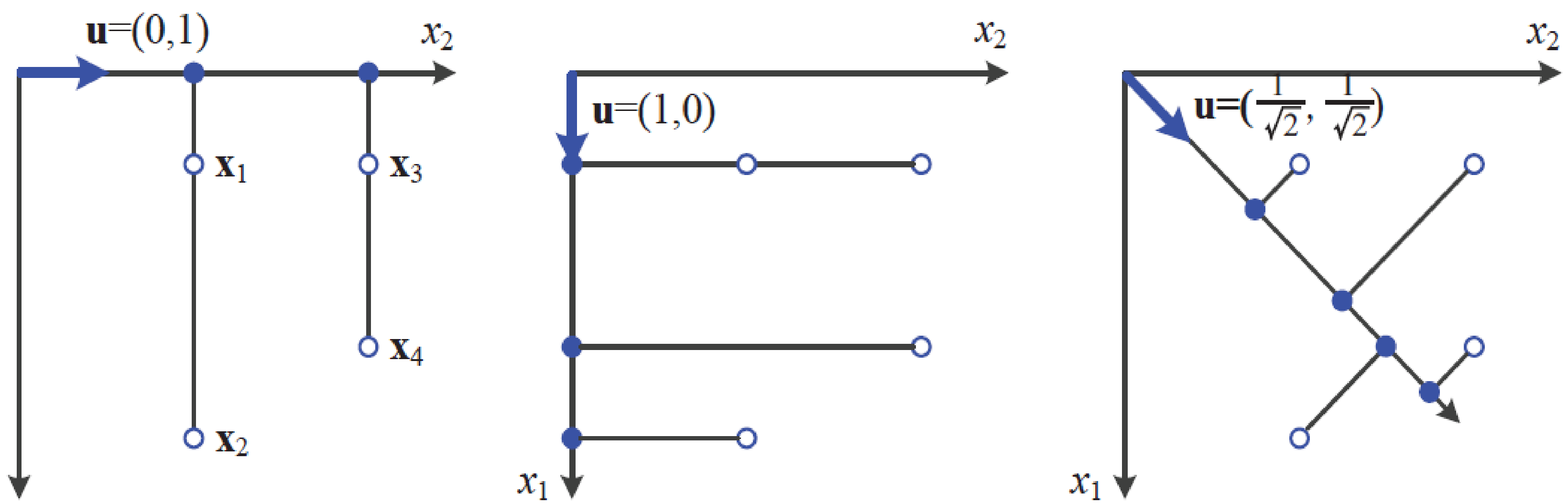
- 차원 축소의 표현: 축 u 상으로 투영으로 표현 $\hat{x} = \mathbf{u}\mathbf{x}^T$



- 정보(점 간의 거리, 상대적 위치 등)의 손실: 손실 최소화 방법?

PCA의 정보 손실 표현

- 원래 공간에 퍼져 있는 정도를 변환된 공간이 얼마나 잘 유지하는지 측정
- 이 수치를 변환된 공간에서 분산으로 측정
- 최적화 문제로 전환: 변환된 샘플들의 분산을 최대화하는 단위 벡터 **u**를 찾아라.



(a) **u**=(0,1)로 투영 (b) **u**=(1,0)으로 투영 (c) **u**=$(\frac{1}{\sqrt{2}},\frac{1}{\sqrt{2}})$로 투영

예 분석

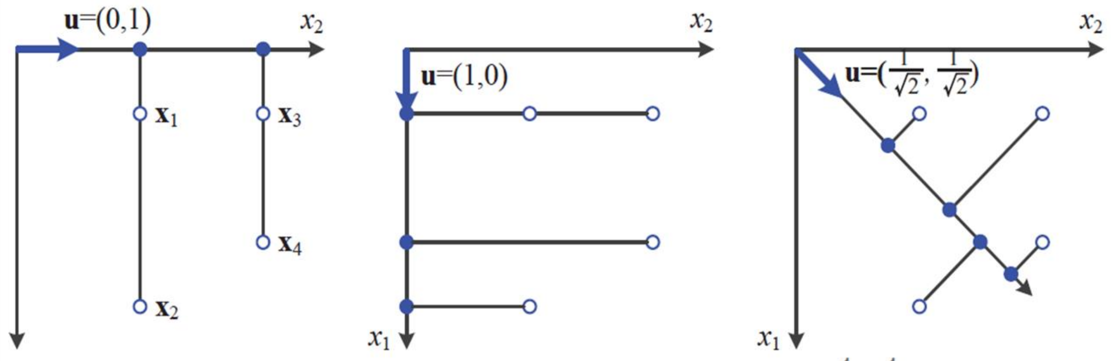
- 원래 샘플

$\mathbf{x}_1 = (1,2), \mathbf{x}_2 = (4,2), \mathbf{x}_3 = (1,4), \mathbf{x}_4 = (3,4)$

- u = (0, 1)로 투영 변환된 샘플

$$\hat{x}_1 = (0 \quad 1)\begin{pmatrix}1\\2\end{pmatrix} = 2,\ \hat{x}_2 = (0 \quad 1)\begin{pmatrix}4\\2\end{pmatrix} = 2,\ \hat{x}_3 = (0 \quad 1)\begin{pmatrix}1\\4\end{pmatrix} = 4,\ \hat{x}_4 = (0 \quad 1)\begin{pmatrix}3\\4\end{pmatrix} = 4$$

투영 변환으로 얻은 네 개의 점 2, 2, 4, 4의 평균은 3이고 분산은 1이다.



(a) u=(0,1)로 투영 (b) u=(1,0)으로 투영 (c) u=$(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$로 투영

- u = $\left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)$로 투영 변환된 샘플

$$\hat{x}_1 = \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)\begin{pmatrix}1\\2\end{pmatrix} = \frac{3}{\sqrt{2}},\quad \hat{x}_2 = \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)\begin{pmatrix}4\\2\end{pmatrix} = \frac{6}{\sqrt{2}},$$

$$\hat{x}_3 = \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)\begin{pmatrix}1\\4\end{pmatrix} = \frac{5}{\sqrt{2}},\quad \hat{x}_4 = \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)\begin{pmatrix}3\\4\end{pmatrix} = \frac{7}{\sqrt{2}}$$

이들의 평균은 3.7123이고 분산은 1.0938이다. u = $\left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)$이 u = (0,1)보다 큰 분산을 만들어 준다. PCA가 볼 때 u = $\left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)$이 더 좋은 것이다. 그렇다면 [그림 6–22]의 상황에서 u = $\left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)$보다 더 좋은 축이 있을까? 이것에 대한 답이 바로 식 (6.34) 문제의 해이다.

수학적 배경

$\hat{x}_i,\ 1 \le i \le n$의 평균 $\bar{\hat{x}} = \frac{1}{n}\sum_{i=1}^{n}\hat{x}_i = \frac{1}{n}\sum_{i=1}^{n}\mathbf{u}\mathbf{x}_i^{\mathrm{T}} = \mathbf{u}\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i^{\mathrm{T}}\right) = \mathbf{u}\bar{\mathbf{x}}^{\mathrm{T}}$

$\hat{x}_i,\ 1 \le i \le n$의 분산 $\hat{\sigma} = \frac{1}{n}\sum_{i=1}^{n}(\hat{x}_i - \bar{\hat{x}})^2 = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{u}\mathbf{x}_i^{\mathrm{T}} - \mathbf{u}\bar{\mathbf{x}}^{\mathrm{T}})^2$

- 단위 벡터 가정: $L(\mathbf{u}) = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{u}\mathbf{x}_i^{\mathrm{T}} - \mathbf{u}\bar{\mathbf{x}}^{\mathrm{T}})^2 + \lambda(1 - \mathbf{u}\mathbf{u}^{\mathrm{T}})$를 최대화하는 $\mathbf{u}$를 찾아라.

$$
\begin{aligned}
\frac{\partial L(\mathbf{u})}{\partial \mathbf{u}} &= \partial\left(\frac{1}{n}\sum_{i=1}^{n}(\mathbf{u}\mathbf{x}_i^{\mathrm{T}} - \mathbf{u}\bar{\mathbf{x}}^{\mathrm{T}})^2 + \lambda(1 - \mathbf{u}\mathbf{u}^{\mathrm{T}})\right)\Big/\partial\mathbf{u} \\
&= \frac{2}{n}\sum_{i=1}^{n}(\mathbf{u}\mathbf{x}_i^{\mathrm{T}} - \mathbf{u}\bar{\mathbf{x}}^{\mathrm{T}})(\mathbf{x}_i - \bar{\mathbf{x}}) - 2\lambda\mathbf{u} \\
&= 2\mathbf{u}\left(\frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})^{\mathrm{T}}(\mathbf{x}_i - \bar{\mathbf{x}})\right) - 2\lambda\mathbf{u} \\
&= 2\mathbf{u}\boldsymbol{\Sigma} - 2\lambda\mathbf{u} \\
&= 2\boldsymbol{\Sigma}\mathbf{u}^{\mathrm{T}} - 2\lambda\mathbf{u}
\end{aligned}
$$

$\boldsymbol{\Sigma}\mathbf{u}^{\mathrm{T}} = \lambda\mathbf{u}^{\mathrm{T}}$ $\quad$ **Σ**는 훈련 집합의 공분산 행렬

$$
\begin{array}{ccc}
 & x & y \\
x & var(x) & cov(x,y) \\
y & cov(x,y) & var(y)
\end{array}
\qquad
\begin{array}{cccc}
 & x & y & z \\
x & var(x) & cov(x,y) & cov(x,z) \\
y & cov(x,y) & var(y) & cov(y,z) \\
z & cov(x,z) & cov(y,z) & var(z)
\end{array}
$$

$$var(x) = \frac{\sum_i^n (x_i - \mu)^2}{N} \qquad cov(x,y) = \frac{\sum_i^n (x_i - \mu)\cdot(y_i - \mu)}{N}$$

PCA 기초 알고리즘

- 학습 집합의 공분산 행렬 계산 → 공분산 행렬의 고유 벡터 계산

• 원래 샘플

$\mathbf{x}_1 = (1,2), \mathbf{x}_2 = (4,2), \mathbf{x}_3 = (1,4), \mathbf{x}_4 = (3,4)$

• 공분산 행렬 : $\Sigma = \begin{pmatrix} 1.688 & -0.250 \\ -0.250 & 1.000 \end{pmatrix}$

• 고유값과 고유 벡터 :

$\lambda_1 = 1.7688, \mathbf{u}_1 = (0.9510, -0.3092)$

$\lambda_2 = 0.9187, \mathbf{u}_2 = (-0.3092, -0.9510)$

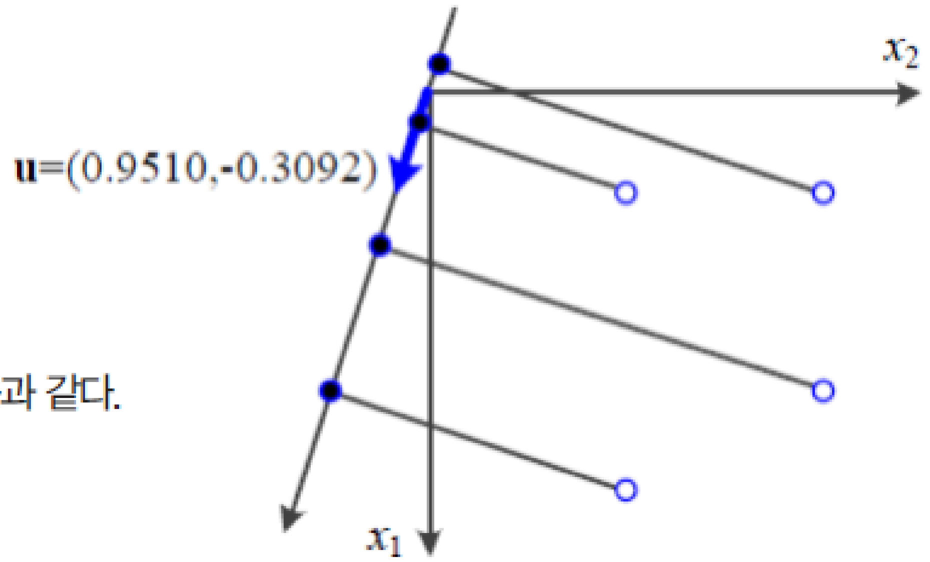
두 개의 고유 벡터 중에 고유값이 큰 $\mathbf{u}_1$을 선택한다. 이 $\mathbf{u}_1$에 네 점을 투영한 결과는 다음과 같다.

• $\mathbf{u}_1=(0.9510, -0.3092)$로 투영된 특징 벡터 :

$\hat{x}_1 = \begin{pmatrix} 0.9510 & -0.3092 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \end{pmatrix} = 0.3326, \quad \hat{x}_2 = \begin{pmatrix} 0.9510 & -0.3092 \end{pmatrix} \begin{pmatrix} 4 \\ 2 \end{pmatrix} = 3.1856$

$\hat{x}_3 = \begin{pmatrix} 0.9510 & -0.3092 \end{pmatrix} \begin{pmatrix} 1 \\ 4 \end{pmatrix} = -0.2858, \quad \hat{x}_4 = \begin{pmatrix} 0.9510 & -0.3092 \end{pmatrix} \begin{pmatrix} 3 \\ 4 \end{pmatrix} = 1.6162$

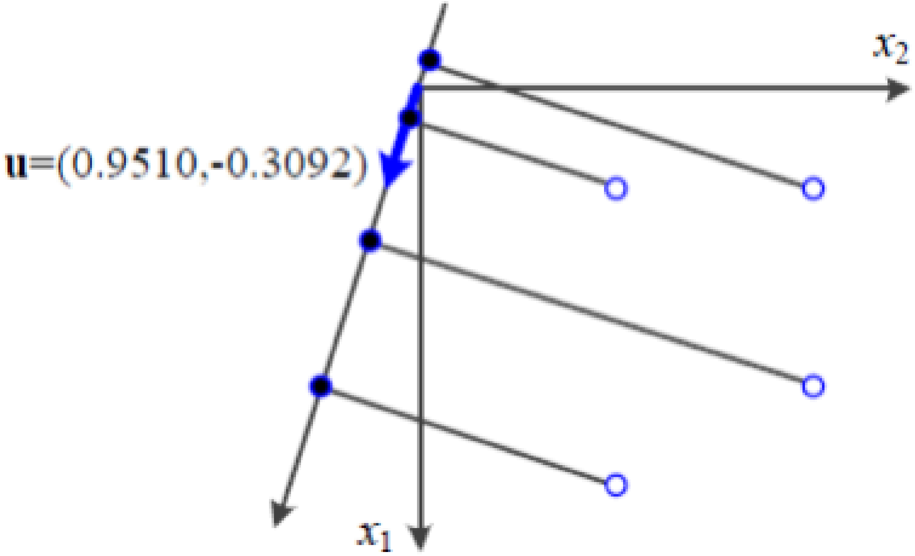
투영 변환된 점들의 평균은 1.2122이고 분산은 1.7688이다.



D 차원에 적용

- 공분산 행렬 **Σ**는 $D*D$이므로, D개의 고유 벡터가 있음
- 이들은 서로 수직인 단위 벡터, 즉 $\mathbf{u}_i\mathbf{u}_i=1$이고 $\mathbf{u}_i\mathbf{u}_j=0$, $i \neq j$
- 고유값이 큰 순서대로 상위 d개의 고유 벡터 $\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_d$를 선택, (**U**는 $d*D$)
- U를 이용한 차원 축소 $\mathbf{y}^{\mathrm{T}} = \mathbf{U}\mathbf{x}^{\mathrm{T}}$

$$\mathbf{U} = \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_d \end{pmatrix}$$



$$\lambda_1 = 1.7688, \mathbf{u}_1 = (0.9510, -0.3092)$$
$$\lambda_2 = 0.9187, \mathbf{u}_2 = (-0.3092, -0.9510)$$

PCA에 의한 특징 추출

입력 : 학습 집합 $X=\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$, 원하는 차원 d
출력 : 변환 행렬 $\mathbf{U}$, 평균 벡터 $\bar{\mathbf{x}}$

$\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i$ // X의 평균 벡터 $\bar{\mathbf{x}}$를 구한다.
for(i=1 to n) $\mathbf{x}_i' = \mathbf{x}_i - \bar{\mathbf{x}}$; // 모든 샘플을 원점 중심으로 옮긴다.
$\mathbf{x}_i', 1 \le i \le n$의 공분산 행렬 $\mathbf{\Sigma}$를 구한다.
$\mathbf{\Sigma}$의 고유 벡터와 고유값을 구한다.
고유값 기준으로 상위 d개의 고유 벡터를 선택하고, 이들을 $\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_d$라 하자.
식 $\boxed{}$에 따라 변환 행렬 $\mathbf{U}$를 만든다.

$$\mathbf{U} = \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_d \end{pmatrix}$$

입력 : 변환 행렬 $\mathbf{U}$, 평균 벡터 $\bar{\mathbf{x}}$, 벡터 $\mathbf{x}$
출력 : 변환된 벡터 $\mathbf{y}$

$\mathbf{x} = \mathbf{x} - \bar{\mathbf{x}}$; // 샘플을 원점 중심으로 옮긴다.
식 $\boxed{}$로 $\mathbf{y}$를 구한다.

$$\mathbf{y}^{\mathrm{T}} = \mathbf{U}\mathbf{x}^{\mathrm{T}}$$

고유 얼굴

- 얼굴 영상에 PCA 적용
- 영상 f_i를 벡터 형태로 변환 (벡터의 차원 $D=MN$): 행 우선으로 재배치

$$\mathbf{x}_i = (f_i(0,0), f_i(0,1), f_i(0,2), \cdots, f_i(0,N-1), f_i(1,0), f_i(1,1), \cdots, f_i(M-1,N-1))$$

- n개의 얼굴 영상으로 구성된 학습 집합 $X=\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$을 입력으로 PCA를 적용
- 고유 벡터 $\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_d$를 고유 얼굴이라 부름

$$\mathbf{x}'^{\mathrm{T}} = \mathbf{U}^{-1}\mathbf{y}^{\mathrm{T}}$$

고유값이 작아짐

고유 얼굴을 활용한 얼굴 인식

- 고유 얼굴의 활용: 얼굴 인식, Turk, 1991
  - PCA로 변환한 벡터 $\mathbf{y}_i$를 모델로 사용: $Y = \{\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_n\}$
  - 테스트 영상 f가 입력되면 PCA로 $\mathbf{y}$를 구한 후, Y에서 가장 가까운 벡터를 찾아 그 부류로 분류

- 고유 얼굴 활용 시 주의 점
  - 얼굴을 찍은 각도와 얼굴 크기, 영상 안에서의 얼굴 위치, 조명이 어느 정도 일정해야 함.
  - 영상마다 다르고 그 변화가 클수록 성능이 떨어짐
  - 조명에 변화를 준 경우 96%, 각도에 변화를 준 경우 85%, 크기에 변화를 준 경우 64%의 인식률을 얻음

디지털영상처리II (2021학년도 2학기)

11주차 : 기계학습(1)

본 강의 자료의 내용 및 그림은 아래 책으로부터 발췌 되었음

- 파이썬으로 배우는 영상처리, Sandipan Dey 지음, 정성환 , 조보호 , 배종욱 옮김, 도서출판 홍릉, 2020년
- Digital Image Processing, 4th Ed., Rafael C . Gonzalez, Richard E. Woods 지음, Pearson, 2018년
- 컴퓨터 비전(Computer Vision) 기본 개념부터 최신 모바일 응용 예까지 IT CookBook, 오일석 지음, 한빛아카데미, 2014년

김 남 규 (ngkim@deu.ac.kr)