기계학습

- Feature extraction (특징 추출)

정보통신공학과

Prof. Jinkyu Kang

MYONGJI UNIVERSITY

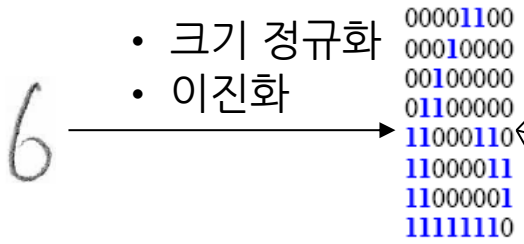
이번주 수업의 목차

- 특징 생성
- 주성분 분석
- Fisher의 선형 분별
- 실용적 관점

MYONGJI
UNIVERSITY

들어가는 말

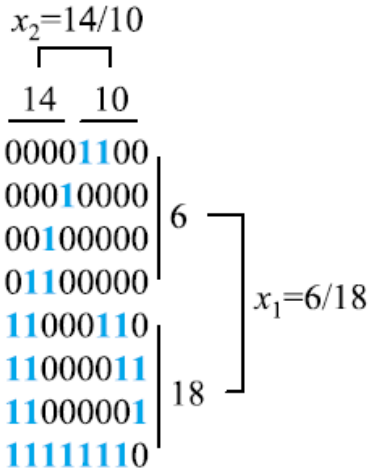
- 특징 추출의 예
  - 필기 숫자 인식



특징 추출

방법 1: 화소 각각을 특징으로 삼음
64 차원 특징 벡터
$\mathbf{x}=(0,0,0,0,1,1,0,0,0,0,0,0,1,\cdots,1,1,0)^T$

방법 2: 가로 이등분과 세로 이등분하여 검은 화소의 개수 비율을 특징으로 삼음



$\mathbf{x}=(6/18,14/10)^T$

또 다른 방법으로는?

- 특징의 우수성 기준
  - 분별력
  - 차원

MYONGJI UNIVERSITY

특징 생성 | 실제 세계의 다양성

- 특징 생성
  - 외부의 물리적 패턴을 특징 벡터라는 수학적 표현으로 변환

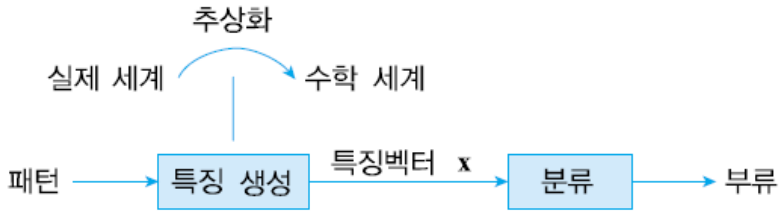


그림 8.1 패턴 인식의 일반적인 틀

- 특징 생성 과정은 매우 다양
  - 특징 추출은 외부 환경에 맞게 설계해야 하기 때문
    - 숫자와 한글은 다른 특징 필요할 수 있음
    - 한글도 통째로 인식하는 방법과 자소로 분할한 후 인식하는 방법이 다른 특징 필요할 수 있음
    - 정면 얼굴로 국한하는 하는 경우와 제약이 없는 얼굴 인식의 특징은 다를 수 있음

MYONGJI UNIVERSITY

특징 생성 | 특징 추출과 특징 선택

- 센싱으로 얻은 신호의 다양성
  - 영상
  - 시간성 신호
  - 측정 벡터
- 특징 추출과 특징 선택

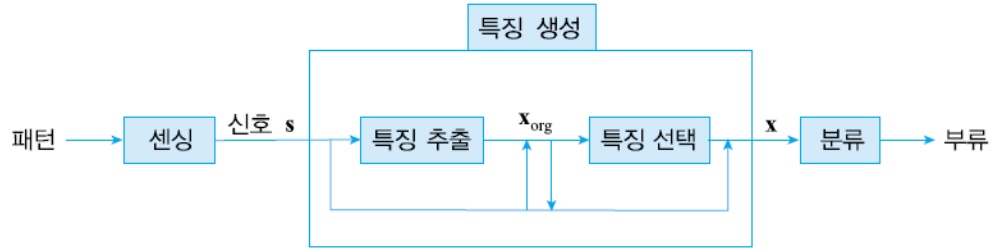


그림 8.2 특징 생성의 절차

$$\left.\begin{array}{ll} \text{특징추출} & \mathbf{x}_{org} = e(\mathbf{s}) \\ \text{특징선택} & \mathbf{x} = s(\mathbf{x}_{org}) \end{array}\right\}$$

- 다양한 상황

$$\left.\begin{array}{l} \mathbf{x} = \mathbf{s} \\ \mathbf{x} = e(\mathbf{s}) \\ \mathbf{x} = s(\mathbf{s}) \\ \mathbf{x} = s(e(\mathbf{s})) \end{array}\right\}$$

MYONGJI UNIVERSITY

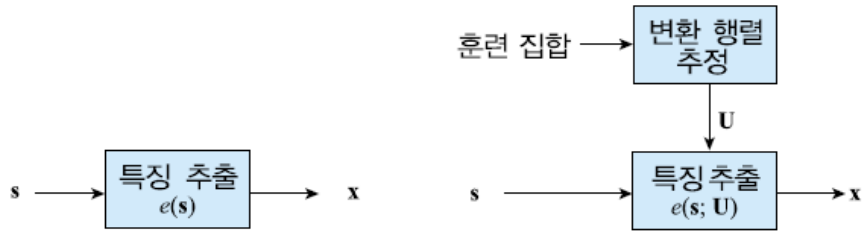
주성분 분석

- 주성분 분석principal component analysis (PCA)
  - 훈련 집합을 이용하여 매개 변수를 추정하고 그것을 이용하여 특징 추출함
  - 정보 손실을 최소화하는 조건에서 차원 축소
  - Karhunen-Loeve (KL) 변환 또는 Hotelling 변환이라고도 부름

주성분 분석 I 동기

- 주성분 분석의 동기
  - **U**는 '정보 손실을 최소화하며' 신호 **s**를 보다 낮은 차원의 특징 벡터 **x**로 변환 (신호 **s**는 D 차원 > 특징 벡터 **x**는 d 차원)
  - 변환 행렬 **U**는 $d \times D$ 행렬
  - 두가지 문제
    - 차원 축소를 어떻게 표현할 것인가?
    - 정보 손실을 어떻게 수량화할 것인가?

$$\mathbf{x} = e(\mathbf{s}; \mathbf{U}) = \mathbf{U}\mathbf{s} \qquad (8.26)$$



(a) 8.2~8.4절의 특징 추출 (b) PCA 특징 추출

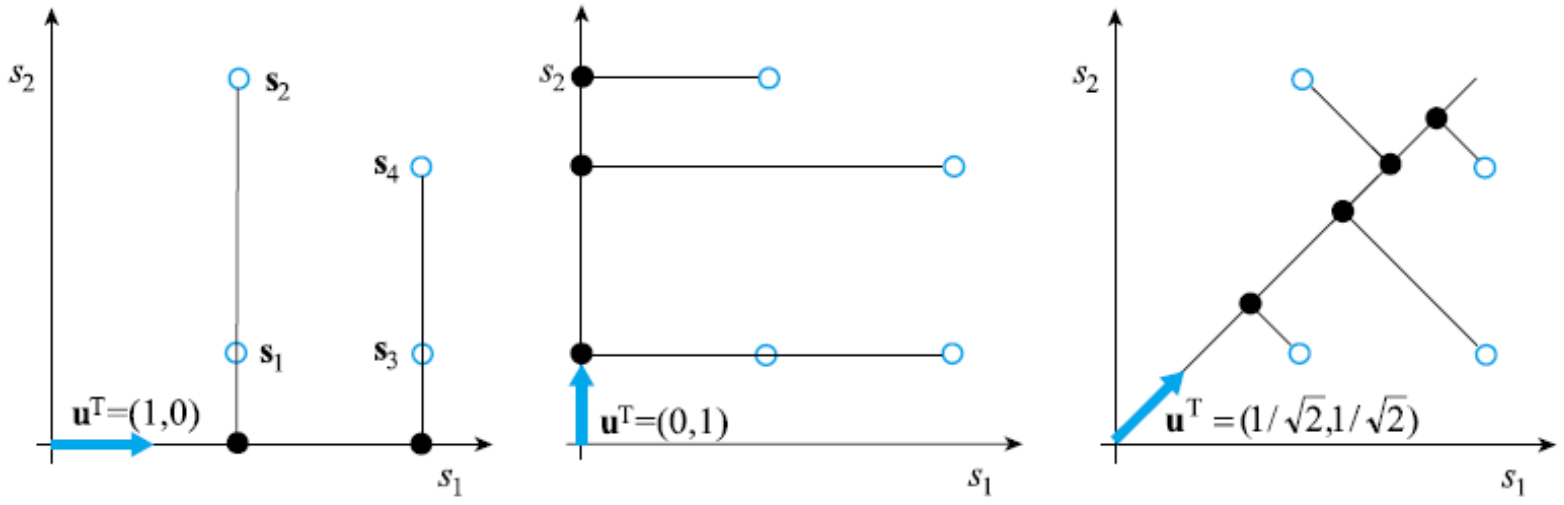
그림 8.12 특징 추출 방법의 비교

MYONGJI UNIVERSITY

주성분 분석 I 동기

- 차원 축소의 표현
  - *D* 차원 단위 벡터 **u** 축으로의 투영

$$\hat{x} = \mathbf{u}^{\mathrm{T}}\mathbf{s} \qquad (8.27)$$



(a) $\mathbf{u}^{\mathrm{T}}=(1,0)$ 축으로 투영 (b) $\mathbf{u}^{\mathrm{T}}=(0,1)$ 축으로 투영 (c) $\mathbf{u}^{\mathrm{T}}=(1/\sqrt{2},1/\sqrt{2})$ 축으로 투영

그림 8.13 투영에 의해 2 차원 공간을 1 차원 공간으로 축소

MYONGJI UNIVERSITY

주성분 분석 I 동기

- 정보 손실의 공식화
  - 원래 훈련 집합이 가진 정보란 무엇일까?
    - 샘플들 간의 거리, 그들 간의 상대적인 위치 등
  - 그림 8.13의 세 가지 축 중에 어느 것이 정보 손실이 가장 적은가?
  - PCA는 샘플들이 원래 공간에 '퍼져있는 정도를' 변환된 공간에서 얼마나 잘 유지하느냐를 척도로 삼음
    - 이 척도는 변환된 공간에서 샘플들의 분산으로 측정함

- 이러한 아이디어에 따라 문제를 공식화 하면,
  - 변환된 샘플들의 분산을 최대화하는 축 (즉 단위 벡터 $\mathbf{u}$)을 찾아라. (8.28)

주성분 분석 I 동기

- 예제) 변환 공간에서의 분산

원래 샘플

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

$\mathbf{u}^T = (1,0)$축으로 투영 변환된 샘플 분산 1.0

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

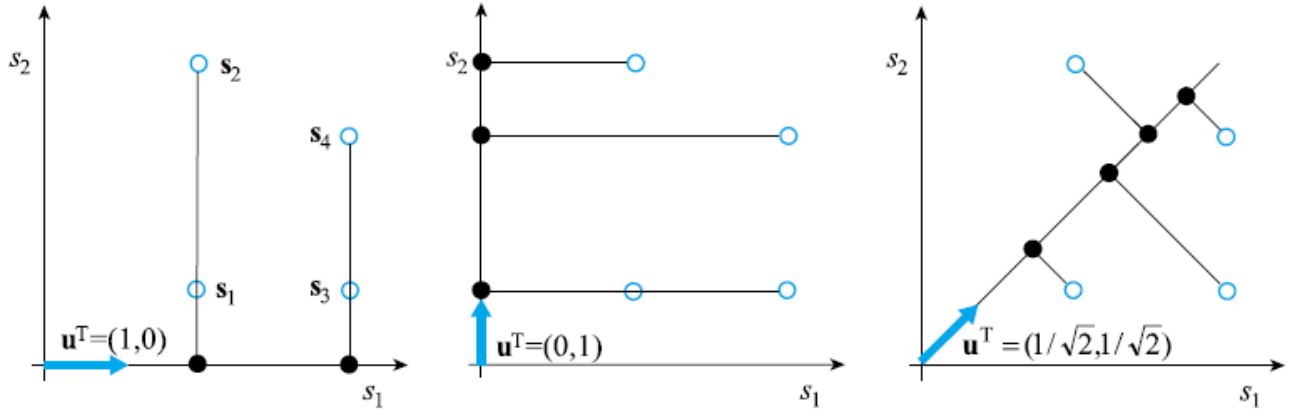
$\mathbf{u}^T = (1/\sqrt{2}, 1/\sqrt{2})$ 축으로 투영 변환된 샘플: 분산 1.0938

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

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

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

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



(a) $\mathbf{u}^T = (1,0)$ 축으로 투영 (b) $\mathbf{u}^T = (0,1)$ 축으로 투영 (c) $\mathbf{u}^T = (1/\sqrt{2}, 1/\sqrt{2})$ 축으로 투영

그림 8.13 투영에 의해 2 차원 공간을 1 차원 공간으로 축소

- 더 좋은 축이 있나?

주성분 분석 I 알고리즘과 응용

- 최적의 축을 찾아 보자.
  - 투영된 점의 평균과 분산

$\hat{x}_i, 1 \le i \le N$의 평균

$$\bar{\hat{x}} = \frac{1}{N}\sum_{i=1}^{N}\hat{x}_i = \frac{1}{N}\sum_{i=1}^{N}\mathbf{u}^{\mathrm{T}}\mathbf{s}_i = \mathbf{u}^{\mathrm{T}}\left(\frac{1}{N}\sum_{i=1}^{N}\mathbf{s}_i\right) = \mathbf{u}^{\mathrm{T}}\bar{\mathbf{s}} \quad (8.29)$$

$\hat{x}_i, 1 \le i \le N$의 분산

$$\hat{\sigma}^2 = \frac{1}{N}\sum_{i=1}^{N}(\hat{x}_i - \bar{\hat{x}})^2 = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{u}^{\mathrm{T}}\mathbf{s}_i - \mathbf{u}^{\mathrm{T}}\bar{\mathbf{s}})^2 \quad (8.30)$$

- 문제를 다시 쓰면,
  - (8.30)의 분산 $\hat{\sigma}^2$을 최대화하는 $\mathbf{u}$를 찾아라. (8.31)
  - $\mathbf{u}^{\mathrm{T}}\mathbf{u}=1$이라는 조건을 만족하는 조건부 최적화 문제로 다시 쓰면,
    - L은 라그랑제 함수, λ는 라그랑제 승수
    - $L(\mathbf{u}) = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{u}^{\mathrm{T}}\mathbf{s}_i - \mathbf{u}^{\mathrm{T}}\bar{\mathbf{s}})^2 + \lambda(1-\mathbf{u}^{\mathrm{T}}\mathbf{u})$ 를 최대화하는 $\mathbf{u}$를 찾아라. (8.32)

MYONGJI UNIVERSITY

주성분 분석 I 알고리즘과 응용

- 미분하고 수식 정리하면,

$$\partial L(\mathbf{u})/\partial \mathbf{u} = \partial \left(\frac{1}{N} \sum_{i=1}^{N} (\mathbf{u}^{\mathrm{T}}\mathbf{s}_i - \mathbf{u}^{\mathrm{T}}\bar{\mathbf{s}})^2 + \lambda(1 - \mathbf{u}^{\mathrm{T}}\mathbf{u}) \right) \Big/ \partial \mathbf{u}$$

$$= \frac{2}{N} \sum_{i=1}^{N} (\mathbf{u}^{\mathrm{T}}\mathbf{s}_i - \mathbf{u}^{\mathrm{T}}\bar{\mathbf{s}})(\mathbf{s}_i - \bar{\mathbf{s}}) - 2\lambda\mathbf{u}$$

$$= 2\mathbf{u}^{\mathrm{T}} \left(\frac{1}{N} \sum_{i=1}^{N} (\mathbf{s}_i - \bar{\mathbf{s}})(\mathbf{s}_i - \bar{\mathbf{s}}) \right) - 2\lambda\mathbf{u}$$

$$= 2\mathbf{u}^{\mathrm{T}}\boldsymbol{\Sigma} - 2\lambda\mathbf{u}$$

$$= 2\boldsymbol{\Sigma}\mathbf{u} - 2\lambda\mathbf{u}$$

- 0으로 놓고 풀면,

$$\boldsymbol{\Sigma}\mathbf{u} = \lambda\mathbf{u} \qquad (8.33)$$

- (8.33)을 해석하면,
  - 훈련 집합의 공분산 행렬 Σ를 구하고, 그것의 고유 벡터를 구하면 그것이 바로 최대 분산을 갖는 **u**가 됨

MYONGJI UNIVERSITY

주성분 분석 I 알고리즘과 응용

- 예제) 최대 분산을 갖는 축

$X = \{(2,1)^T, (2,4)^T, (4,1)^T, (4,3)^T\}$ 훈련 집합

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

$\lambda_1 = 1.7688, \quad \mathbf{u}_1^T = (-0.3092,\ 0.9510)$ 고유 벡터

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

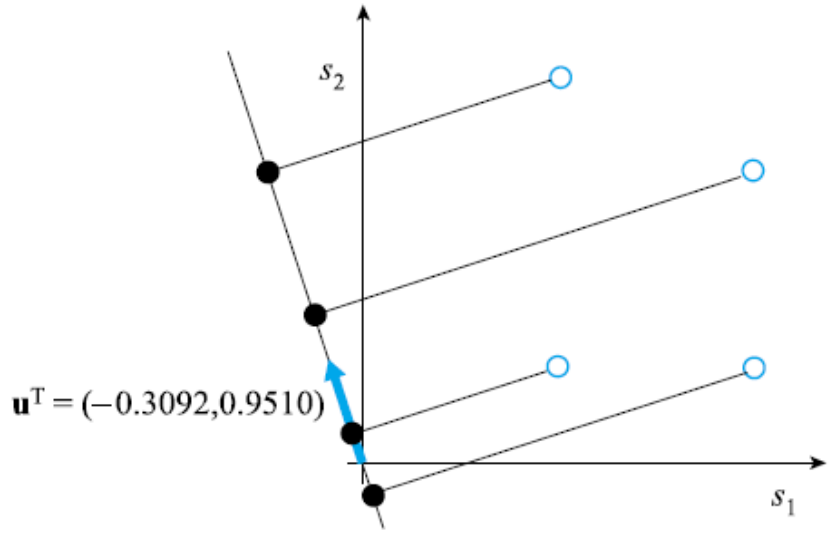


그림 8.14 PCA로 구한 최적의 축

$\mathbf{u}_1^T = (-0.3092, 0.9510)$ 축으로 투영된 특징 벡터:

$$\hat{x}_1 = \begin{pmatrix} -0.3092 & 0.9510 \end{pmatrix} \begin{pmatrix} 2 \\ 1 \end{pmatrix} = 0.3326$$

$$\hat{x}_2 = \begin{pmatrix} -0.3092 & 0.9510 \end{pmatrix} \begin{pmatrix} 2 \\ 4 \end{pmatrix} = 3.1856$$

$$\hat{x}_3 = \begin{pmatrix} -0.3092 & 0.9510 \end{pmatrix} \begin{pmatrix} 4 \\ 1 \end{pmatrix} = -0.2858$$

$$\hat{x}_4 = \begin{pmatrix} -0.3092 & 0.9510 \end{pmatrix} \begin{pmatrix} 4 \\ 3 \end{pmatrix} = 1.6162$$

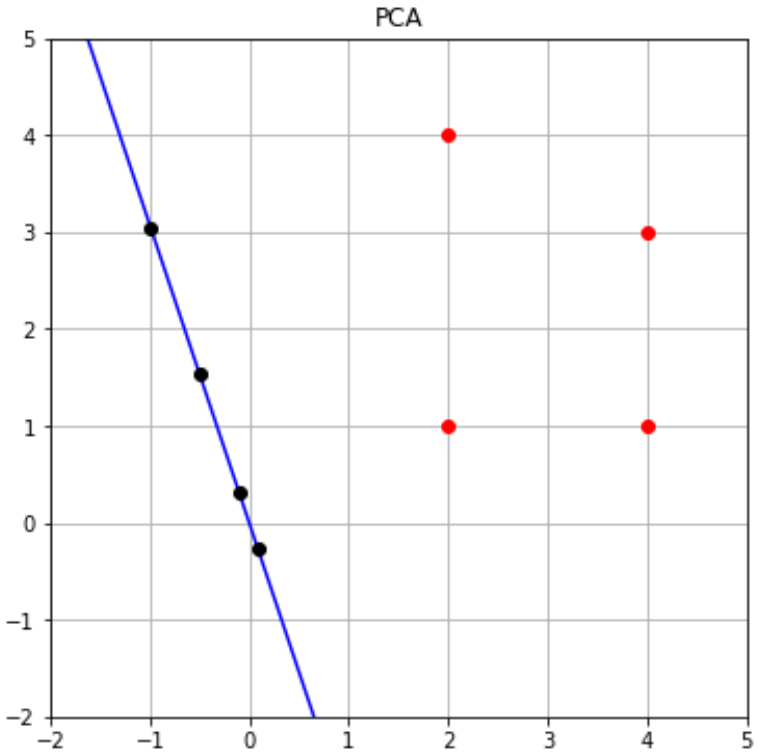
이들의 분산은 1.7688
그림 8.13과 비교해 보자.

MYONGJI UNIVERSITY

주성분 분석 | 알고리즘과 응용

- Python을 이용한 PCA
  - 최대 분산을 갖는 축

$X = \{(2,1)^T, (2,4)^T, (4,1)^T, (4,3)^T\}$ 훈련 집합



MYONGJI UNIVERSITY

주성분 분석 I 알고리즘과 응용

- 변환 행렬
  - (8.33)을 풀면 D 개의 고유 벡터. 고유값이 큰 것일수록 중요도가 큼
  - 따라서 D 차원을 d 차원으로 줄인다면 고유값이 큰 순으로 d 개의 고유 벡터를 취함. 이들을 주성분이라 부르고 $\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_d$로 표기함
  - 변환 행렬 U는,

$$\mathbf{U} = \begin{pmatrix} \mathbf{u}_1^{\mathrm{T}} \\ \mathbf{u}_2^{\mathrm{T}} \\ \vdots \\ \mathbf{u}_d^{\mathrm{T}} \end{pmatrix} \tag{8.34}$$

- 실제 변환은,

$$\mathbf{x} = \mathbf{U}\mathbf{s} \tag{8.35}$$

MYONGJI UNIVERSITY

주성분 분석 I 알고리즘과 응용

- 알고리즘

- 응용
  - 특징 추출 (예, 얼굴 인식)
  - 차원 축소
  - 데이터 압축
  - 데이터 시각화

알고리즘 [8.3] PCA에 의한 변환 행렬 구함

입력: 훈련 집합 $X = \{\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_N\}$, 원하는 차원 d

출력: 변환 행렬 $\mathbf{U}$, 평균 벡터 $\bar{\mathbf{s}}$

알고리즘:

1. $\bar{\mathbf{s}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{s}_i$ // X의 평균 벡터
2. **for** $(i = 1$ **to** $N)$ $\mathbf{s}'_i = \mathbf{s}_i - \bar{\mathbf{s}}$; // 평균 벡터를 빼줌
3. $\mathbf{s}'_i, 1 \le i \le N$ 의 공분산 행렬 $\mathbf{\Sigma}$를 구한다.
4. $\mathbf{\Sigma}$의 고유 벡터와 고유 값을 구한다.
5. 고유 값 기준으로 가장 큰 d 개의 고유 벡터를 선택한다.
   이들을 $\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_d$라 하자.
6. (8.34)로 변환 행렬 $\mathbf{U}$를 만든다.
7. **return** $\mathbf{U}$, $\bar{\mathbf{s}}$;

알고리즘 [8.4] PCA에 의한 특징 추출

입력: 변환 행렬 $\mathbf{U}$, 평균 벡터 $\bar{\mathbf{s}}$, 샘플 $\mathbf{s}$

출력: 특징 벡터 $\mathbf{x}$

알고리즘:

1. $\mathbf{s} = \mathbf{s} - \bar{\mathbf{s}}$; // 샘플에서 평균 벡터를 뺀다.
2. $\mathbf{x} = \mathbf{U}\mathbf{s}$; // (8.35)
3. **return** $\mathbf{x}$

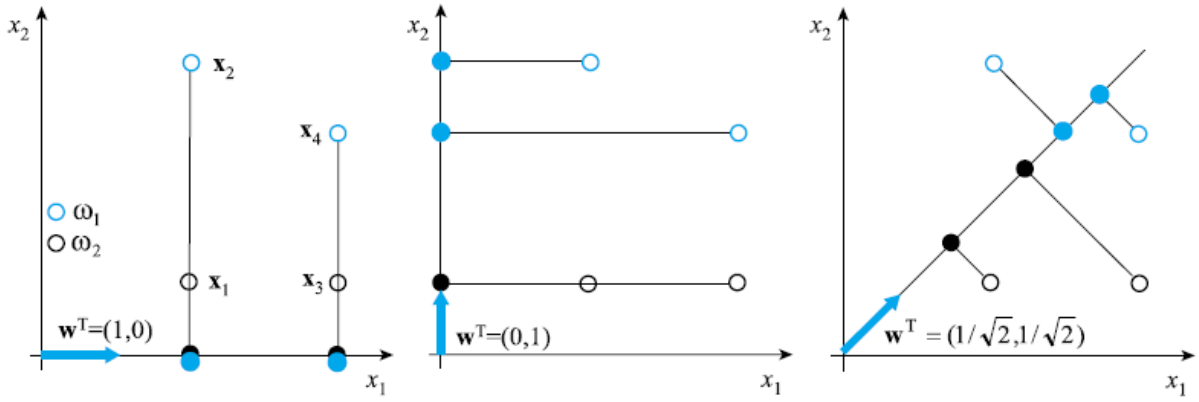
예제

- 아래 훈련집합과 이를 기반으로 PCA 기법을 수행했을때 두개의 고유 벡터 중에 고유 값이 작은 $\mathbf{u}_2$를 선택했을 때 투영된 특징 벡터들과 투영 변환된 점들의 분산을 구하고 고유 값이 큰 $\mathbf{u}_1$을 선택했을 때와 비교해보자. (Python을 이용해서 비교해보자.)
  - 훈련 집합: $X = \{(2,1)^T, (2,4)^T, (4,1)^T, (4,3)^T\}$

MYONGJI UNIVERSITY

Fisher의 선형 분별

- Fisher의 선형 분별linear discriminant (LD)
  - 특징 추출이 아니라 분류기 설계에 해당
  - 하지만 PCA와 원리가 비슷
  - PCA와 Fisher LD는 목표가 다름
    - PCA는 정보 손실 최소화 (샘플의 부류 정보 사용 안함)
    - Fisher LD는 분별력을 최대화 (샘플의 부류 정보 사용함)
- 원리
  - 축으로의 투영 $y = \mathbf{w}^T\mathbf{x}$
  - 세 개의 축 중에 어느 것이 분별력 관점에서 가장 유리한가?



(a) $\mathbf{w}^T=(1,0)$ 축으로 투영 (b) $\mathbf{w}^T=(0,1)$ 축으로 투영 (c) $\mathbf{w}^T=(1/\sqrt{2},1/\sqrt{2})$ 축으로 투영

그림 8.15 2 차원 공간을 1 차원 공간으로 투영 (Fisher의 LD)

Fisher의 선형 분별

- 문제 공식화
  - 유리한 정도를 어떻게 수식화할까?
  - 가장 유리한 축 (즉 최적의 축)을 어떻게 찾을 것인가?

- 기본 아이디어
  - “같은 부류의 샘플은 모여있고 다른 부류의 샘플은 멀리 떨어져 있을수록 유리하다.”
  - 부류간 퍼짐between-class scatter
  - 부류내 퍼짐within-class scatter

$$\mathbf{m}_i = \frac{1}{N_i}\sum_{\mathbf{x}\in\omega_i}\mathbf{x}$$

$$\left.\begin{aligned}\overline{m}_i &= \frac{1}{N_i}\sum_{y\in\omega_i} y \\ &= \frac{1}{N_i}\sum_{\mathbf{x}\in\omega_i}\mathbf{w}^{\mathrm{T}}\mathbf{x} \\ &= \mathbf{w}^{\mathrm{T}}\mathbf{m}_i\end{aligned}\right\}$$

$$\text{부류간 퍼짐} = |\overline{m}_1 - \overline{m}_2| = |\mathbf{w}^{\mathrm{T}}\mathbf{m}_1 - \mathbf{w}^{\mathrm{T}}\mathbf{m}_2| = |\mathbf{w}^{\mathrm{T}}(\mathbf{m}_1 - \mathbf{m}_2)|$$

$$\overline{s}_i^{\,2} = \sum_{y\in\omega_i}(y - \overline{m}_i)^2$$

$$\text{부류내 퍼짐} = \overline{s}_1^{\,2} + \overline{s}_2^{\,2}$$

MYONGJI UNIVERSITY

Fisher의 선형 분별

- 목적 함수 $J(\mathbf{w})$

$$J(\mathbf{w}) = \frac{\text{부류간 퍼짐}}{\text{부류내 퍼짐}} = \frac{|\overline{m}_1 - \overline{m}_2|^2}{\overline{s}_1^{\,2} + \overline{s}_2^{\,2}} \qquad (8.43)$$

  - $J(\mathbf{w})$를 최대화하는 $\mathbf{w}$를 찾아라.
- 분자와 분모를 다시 쓰면,

분모

$$\begin{aligned}
\overline{s}_1^{\,2} + \overline{s}_2^{\,2} &= \sum_{y\in\omega_1}(y-\overline{m}_1)^2 + \sum_{y\in\omega_2}(y-\overline{m}_2)^2 \\
&= \sum_{\mathbf{x}\in\omega_1}(\mathbf{w}^{\mathrm{T}}\mathbf{x}-\mathbf{w}^{\mathrm{T}}\mathbf{m}_1)^2 + \sum_{\mathbf{x}\in\omega_2}(\mathbf{w}^{\mathrm{T}}\mathbf{x}-\mathbf{w}^{\mathrm{T}}\mathbf{m}_2)^2 \\
&= \sum_{\mathbf{x}\in\omega_1}\mathbf{w}^{\mathrm{T}}(\mathbf{x}-\mathbf{m}_1)(\mathbf{x}-\mathbf{m}_1)^{\mathrm{T}}\mathbf{w} + \sum_{\mathbf{x}\in\omega_2}\mathbf{w}^{\mathrm{T}}(\mathbf{x}-\mathbf{m}_2)(\mathbf{x}-\mathbf{m}_2)^{\mathrm{T}}\mathbf{w} \\
&= \mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{W}}\mathbf{w}
\end{aligned}$$

분자

$$\begin{aligned}
|\overline{m}_1 - \overline{m}_2|^2 &= (\mathbf{w}^{\mathrm{T}}\mathbf{m}_1 - \mathbf{w}^{\mathrm{T}}\mathbf{m}_2)^2 \\
&= \mathbf{w}^{\mathrm{T}}(\mathbf{m}_1-\mathbf{m}_2)(\mathbf{m}_1-\mathbf{m}_2)^{\mathrm{T}}\mathbf{w} \\
&= \mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{B}}\mathbf{w}
\end{aligned}$$

이 때 $\mathbf{S}_{\mathrm{B}} = (\mathbf{m}_1-\mathbf{m}_2)(\mathbf{m}_1-\mathbf{m}_2)^{\mathrm{T}}$

이 때 $\mathbf{S}_{\mathrm{W}} = \mathbf{S}_1 + \mathbf{S}_2$이고, $\mathbf{S}_i = \sum_{\mathbf{x}\in\omega_i}(\mathbf{x}-\mathbf{m}_i)(\mathbf{x}-\mathbf{m}_i)^{\mathrm{T}}$

Fisher의 선형 분별

- 목적 함수를 다시 쓰면,

$$J(\mathbf{w}) = \frac{\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{B}}\mathbf{w}}{\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{W}}\mathbf{w}} \qquad (8.48)$$

- $\partial J(\mathbf{w})/\partial \mathbf{w} = 0$ 으로 두고 풀면,

$$(\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{W}}\mathbf{w})\mathbf{S}_{\mathrm{B}}\mathbf{w} = (\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{B}}\mathbf{w})\mathbf{S}_{\mathrm{W}}\mathbf{w} \qquad (8.49)$$

  - (8.49)를 정리하면,

$$(\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{B}}\mathbf{w})\mathbf{S}_{\mathrm{W}}\mathbf{w} = (\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{W}}\mathbf{w})\mathbf{S}_{\mathrm{B}}\mathbf{w}$$

$$(\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{B}}\mathbf{w})\mathbf{S}_{\mathrm{W}}\mathbf{w} = (\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{W}}\mathbf{w})\alpha_1(\mathbf{m}_1 - \mathbf{m}_2)$$

$$\mathbf{S}_{\mathrm{W}}\mathbf{w} = \frac{\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{W}}\mathbf{w}}{\mathbf{w}^{\mathrm{T}}\mathbf{S}_{\mathrm{B}}\mathbf{w}}\alpha_1(\mathbf{m}_1 - \mathbf{m}_2)$$

$$\mathbf{S}_{\mathrm{W}}\mathbf{w} = \alpha_2\alpha_1(\mathbf{m}_1 - \mathbf{m}_2)$$

$$\mathbf{w} = \alpha_2\alpha_1\mathbf{S}_{\mathrm{W}}{}^{-1}(\mathbf{m}_1 - \mathbf{m}_2)$$

  - 결국 답은 (즉 구하고자 한 최적의 축은),

$$\mathbf{w} = \alpha\mathbf{S}_{\mathrm{W}}{}^{-1}(\mathbf{m}_1 - \mathbf{m}_2) \qquad (8.50)$$

MYONGJI UNIVERSITY

Fisher의 선형 분별

- 예제 8.8 Fisher의 선형 분별

ω_1 샘플 (파랑): $\mathbf{x}_2 = (2,4)^T,\ \mathbf{x}_4 = (4,3)^T$

ω_2 샘플 (검정): $\mathbf{x}_1 = (2,1)^T,\ \mathbf{x}_3 = (4,1)^T$

$$\mathbf{m}_1 = (3,3.5)^T$$

$$\mathbf{m}_2 = (3,1)^T$$

$$\mathbf{S}_W = \mathbf{S}_1 + \mathbf{S}_2 = \left[\begin{pmatrix}-1\\0.5\end{pmatrix}\begin{pmatrix}-1 & 0.5\end{pmatrix} + \begin{pmatrix}1\\-0.5\end{pmatrix}\begin{pmatrix}1 & -0.5\end{pmatrix}\right] + \left[\begin{pmatrix}-1\\0\end{pmatrix}\begin{pmatrix}-1 & 0\end{pmatrix} + \begin{pmatrix}1\\0\end{pmatrix}\begin{pmatrix}1 & 0\end{pmatrix}\right] = \begin{pmatrix}4 & -1\\-1 & 0.5\end{pmatrix}$$

$$\mathbf{S}_W^{-1} = \begin{pmatrix}0.5 & 1\\1 & 4\end{pmatrix}$$

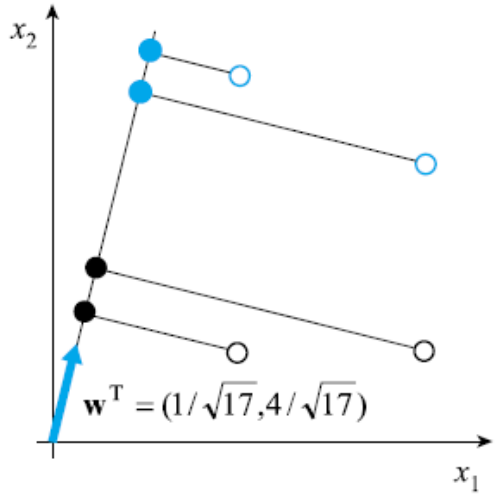


그림 8.16 Fisher의 선형 분별

$$\mathbf{w} = \alpha\begin{pmatrix}0.5 & 1\\1 & 4\end{pmatrix}\begin{pmatrix}0\\2.5\end{pmatrix} = \alpha'\begin{pmatrix}1\\4\end{pmatrix}$$

$$\mathbf{w} = \left(\frac{1}{\sqrt{17}} \quad \frac{4}{\sqrt{17}}\right)^T = (0.24254, 0.97014)^T$$

이것이 최적의 축이다.
그림 8.15와 비교해 보자.

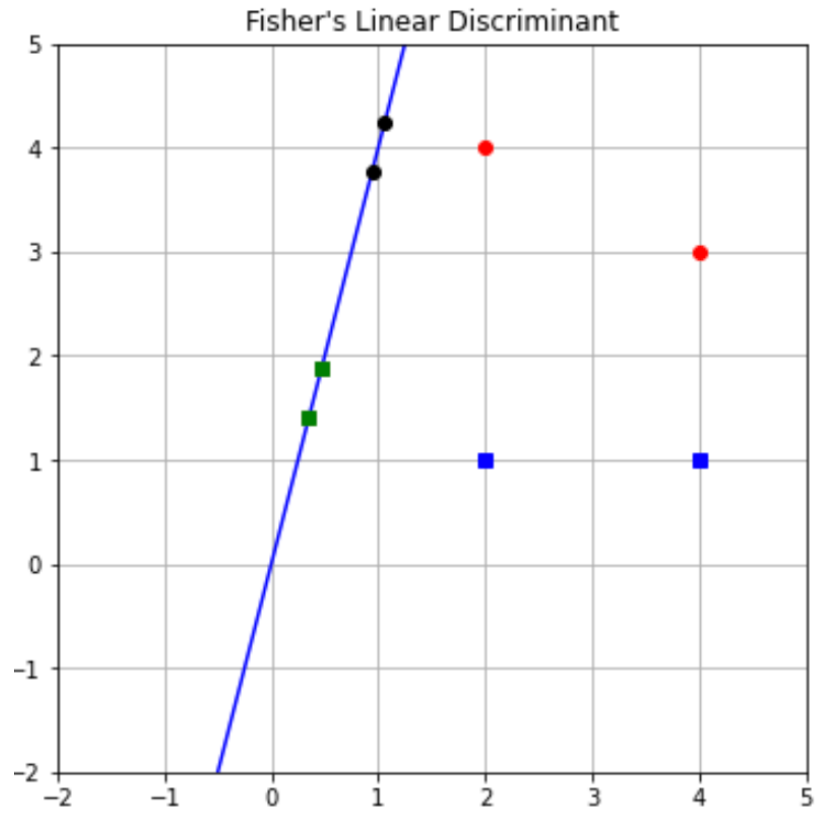
MYONGJI UNIVERSITY

Fisher의 선형 분별 | Python 예제

- 예제 8.8 Fisher의 선형 분별

ω_1 샘플 (파랑): $\mathbf{x}_2 = (2,4)^T,\ \mathbf{x}_4 = (4,3)^T$

ω_2 샘플 (검정): $\mathbf{x}_1 = (2,1)^T,\ \mathbf{x}_3 = (4,1)^T$



실용적 관점

- 특징 추출은 기계학습 과정에서 휴리스틱한 경험과 실험에 따른 시행 착오가 가장 많이 필요한 단계
  - 외부 환경에 영향을 가장 많이 받기 때문
  - 여기서 소개하는 몇가지 실용적 방법이 도움이 됨
    - 특징이 만족스러운 성능을 보이지 못하면?
      - 버리고 다른 특징을 채택
      - 또는 기존 특징에 새로운 특징을 추가하는 특징 결합
    - 특징이 거리 개념을 가지지 않으면?
      - 예) 혈액형을 나타내는 특징 $x \in \{A, B, O, AB\}$
      - 거리 개념이 없는 특징 x_i가 n 개의 값을 갖는다면 x_i를 $x_{i1}, x_{i2}, \cdots, x_{in}$으로 확장
      - $x_{i1}, x_{i2}, \cdots, x_{in}$ 중 하나만 1을 가지고 나머지는 0

MYONGJI UNIVERSITY

실용적 관점

- 특징 추출은 기계학습 과정에서 휴리스틱한 경험과 실험에 따른 시행 착오가 가장 많이 필요한 단계
  - 외부 환경에 영향을 가장 많이 받기 때문
  - 여기서 소개하는 몇가지 실용적 방법이 도움이 됨
    - 특징마다 동적 범위가 크게 다르면?
      - 특징 값의 정규화

≫ 선형 변환

$$\tilde{x}_i = low_i + \frac{high_i - low_i}{max_i - min_i}(x_i - min_i) \qquad (8.51)$$

≫ 통계에 의한 변환 (평균은 0, 표준 편차는 1을 가지도록 정규화)

$$\tilde{x}_i = \frac{x_i - \bar{x}_i}{\sigma_i} \qquad (8.52)$$

MYONGJI UNIVERSITY

실용적 관점 | 특징 전처리

- 예제) 특징 정규화
  - 사람을 키 (m 단위)와 몸무게 (kg 단위)의 두 개 특징으로 표현

$$\mathbf{a} = (1.60,\ 70.0)^{\mathrm{T}},\ \mathbf{b} = (1.65,\ 65.5)^{\mathrm{T}},\ \mathbf{c} = (1.95,\ 71.0)^{\mathrm{T}},\ \mathbf{d} = (1.68,\ 72.0)^{\mathrm{T}}$$

    - 거리 계산에 따르면 a는 b보다 c에 가깝다.
    - 몸무게의 동적 범위가 커서 거리 계산을 주도하기 때문
  - (8.51)의 정규화 식을 유도하면,

$$\tilde{x}_1 = 0 + \frac{1-0}{1.95-1.60}(x_1 - 1.60) = \frac{1}{0.35}(x_1 - 1.60)$$

$$\tilde{x}_2 = 0 + \frac{1-0}{72.0-65.5}(x_2 - 65.5) = \frac{1}{6.5}(x_2 - 65.5)$$

  - 정규화하고 거리를 계산해 보면,

$$\mathbf{a}' = (0,0.692)^{\mathrm{T}},\ \mathbf{b}' = (0.143,0)^{\mathrm{T}},\ \mathbf{c}' = (1,0.846)^{\mathrm{T}},\ \mathbf{d}' = (0.229,1)^{\mathrm{T}}$$

$$dist(\mathbf{a}',\mathbf{b}') = \sqrt{(0-0.143)^2 + (0.692-0)^2} = 0.707$$

$$dist(\mathbf{a}',\mathbf{c}') = \sqrt{(0-1)^2 + (0.692-0.846)^2} = 1.012$$

MYONGJI UNIVERSITY