데이터과학과 AI를 위한 파이썬

07강. 고윳값과 고유벡터

세종사이버대학교

김명배 교수

학습내용

- 고윳값 분해
- 특이값 분해
- 주성분 분석

학습목표

- 고윳값과 고유벡터, 대각화를 설명하고 파이썬으로 고유값과 고유벡터를 계산할 수 있다.
- 특이값 분해에 대해 설명할 수 있고 파이썬으로 특이값과 특이벡터를 산출 할 수 있다.
- 주성분 분석의 개념을 설명할 수 있고, 공분산행렬의 고윳값과 고유벡터가 가지는 의미를 설명할 수 있다.

1. 고윳값 분해(eigen-value decomposition)

1) 고윳값과 고유벡터

- 고유벡터(eigen vector)는 선형 변환을 취했을 때 방향은 변하지 않고 크기만 변하는 벡터, 이 때 변한 크기가 고유 값(eigen value)임
- 고윳값과 고유벡터를 찾는 작업을 고윳값분해 또는 고유분해라고 함
- 정방행렬 A에 대해 $A\vec{x} = \lambda\vec{x}$($\lambda$는 상수)가 성립하는 0이 아닌 벡터 $\vec{x}$가 존재할 때, 다음을 만족

$$\boldsymbol{A\vec{x}} = \underbrace{\boldsymbol{\lambda}}_{\text{고윳값}}\underbrace{\boldsymbol{\vec{x}}}_{\text{고유벡터}}(\boldsymbol{\lambda}\text{는 상수})$$

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \lambda \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

1. 고윳값 분해

1) 고윳값과 고유벡터

[예시1] 정방행렬 $A = \begin{bmatrix} 2 & 4 \\ 5 & 3 \end{bmatrix}$, 고유 값 $\lambda = 7$, 고유 벡터 $\vec{x} = \begin{bmatrix} 4 \\ 5 \end{bmatrix}$일 때,

$A\vec{x} = \lambda\,\vec{x}$ 를 만족하는가?

[풀이]

$$A\vec{x} = \begin{bmatrix} 2 & 4 \\ 5 & 3 \end{bmatrix}\begin{bmatrix} 4 \\ 5 \end{bmatrix} = \begin{bmatrix} (2 \times 4) + (4 \times 5) \\ (5 \times 4) + (3 \times 5) \end{bmatrix} = \begin{bmatrix} 28 \\ 35 \end{bmatrix} = 7\begin{bmatrix} 4 \\ 5 \end{bmatrix}$$

$$\lambda\vec{x} = 7\begin{bmatrix} 4 \\ 5 \end{bmatrix}$$

1. 고윳값 분해

1) 고윳값과 고유벡터

[예시2] 정방행렬 $A = \begin{bmatrix} 2 & 4 \\ 5 & 3 \end{bmatrix}$, 고유 값 $\lambda = -2$, 고유 벡터 $\vec{x} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$일 때,

$A\vec{x} = \lambda\,\vec{x}$ 를 만족하는가?

[풀이]

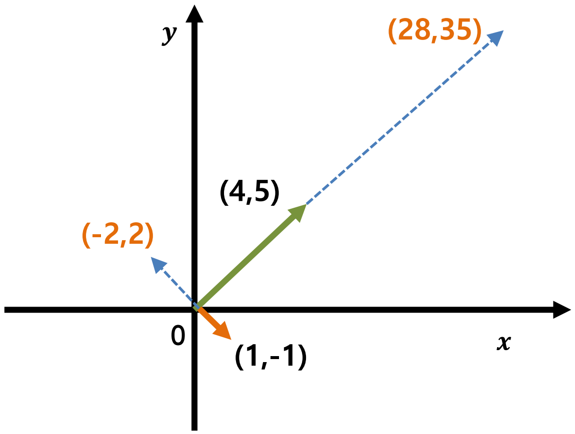
$$A\vec{x} = \begin{bmatrix} 2 & 4 \\ 5 & 3 \end{bmatrix}\begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} (2 \times 1) + (4 \times (-1)) \\ (5 \times 1) + (3 \times (-1)) \end{bmatrix} = \begin{bmatrix} -2 \\ 2 \end{bmatrix} = -2\begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

$$\lambda\vec{x} = -2\begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

1. 고윳값 분해

2) 고윳값과 고유벡터의 의미

행렬 곱의 결과가 원래 벡터와 '<u>방향</u>'은 같고, <u>비율</u>만 **λ**만큼 비례하는 변환



1. 고윳값 분해

3) 고유벡터와 고윳값구하는 방법

- 고윳값을 구하고 가우스 소거법을 이용하여 고유벡터를 구함

가) 고윳값 구하는 공식

$$A\vec{x} = \lambda\vec{x} \Leftrightarrow (A-\lambda I)\vec{x} = 0 \Leftrightarrow \overbrace{\underbrace{\det(\underbrace{A-\lambda I}_{\text{특성행렬}})}_{\text{특성행렬식}} = 0}^{\text{특성방정식}}$$

[예시] $A-\lambda I = \begin{bmatrix} 2 & 4 \\ 5 & 3 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} = \begin{bmatrix} 2-\lambda & 4 \\ 5 & 3-\lambda \end{bmatrix}$

$\det(A-\lambda I) = \det\begin{bmatrix} 2-\lambda & 4 \\ 5 & 3-\lambda \end{bmatrix} = 0$

$(2-\lambda)(3-\lambda) - 5 \times 4 = 0$

$\Leftrightarrow \lambda^2 - 5\lambda - 14 = 0$

$\Leftrightarrow (\lambda-7)(\lambda+2) = 0$

$\therefore \lambda = 7, -2$

1. 고윳값 분해

3) 고유벡터와 고윳값구하는 방법

나) 고유 벡터 구하는 공식

- 고유 값에 대입하고 이에 대응하는 고유 벡터($\vec{x}$)를 구함

[예시] λ = 7에 대응하는 고유 벡터

$$\begin{bmatrix} 2 & 4 \\ 5 & 3 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} - \begin{bmatrix} 7 & 0 \\ 0 & 7 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$\begin{bmatrix} 2-7 & 4 \\ 5 & 3-7 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$5x_1 - 4x_2 = 0$$

$$\therefore x_1 = 4,\ x_2 = 5$$

[파이썬] numpy라이브러리의 linalg.eig() 함수 사용

1. 고윳값 분해

4) 대각화(diagonalization)

- N차원 정방행렬 A가 복소수 고윳값 N개와 이에 대응하는 고유벡터를 가짐

행렬 A의 고윳값 : $\lambda_1, \lambda_2, \cdots, \lambda_N$

행렬 A의 고유벡터 : $v_1, v_2, \cdots, v_N$

고유벡터 행렬 : $V = [v_1, v_2, \cdots, v_N], \quad V \in R^{N \times N}$

고윳값행렬 : $\Lambda = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_N \end{bmatrix}, \quad \Lambda \in R^{N \times N}$

1. 고윳값 분해

4) 대각화(diagonalization)

- 행렬과 고유벡터행렬의 곱은 고유벡터행렬과 고윳값행렬의 곱과 같음

$$
\begin{aligned}
\mathrm{AV} &= \mathrm{A}[v_1, v_2, \cdots, v_N] \\
&= [Av_1 \cdots Av_n] \\
&= [\lambda_1 v_1 \cdots \lambda_N v_N] \\
&= [v_1, v_2, \cdots, v_N] \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_N \end{bmatrix} \\
&= V\Lambda
\end{aligned}
$$

1. 고윳값 분해

4) 대각화(diagonalization)

- 고유벡터행렬 V의 역행렬이 존재한다면, 행렬을 고유벡터행렬과 고윳값행렬의 곱으로 표현할 수 있음

$$A = V\Lambda V^{-1}$$ ➔ 대각화(diagonalization)

- 행렬을 대각화할 수 있으면 대각화가능한 행렬이라고 함
- 행렬이 대각화가능하려면 고유벡터가 선형독립이어야 함(Full rank)
- 어떤 행렬을 고유값행렬과 고유벡터 행렬로 표현할 수 있다는 것은 대각화를 통해 행렬 A의 특성을 알 수 있다는 뜻
- 미분방정식을 풀거나 반복적인 행렬의 거듭제곱 계산 등을 간소화 시킴

1. 고윳값 분해

5) 고유분해의 성질

N차원 정방행렬 A에 대해 $X^T X$인 정방행렬이라고 할 때

① 행렬 A는 N개의 고윳값-고유벡터를 가진다.
(복소수인 경우와 중복인 경우를 포함)

② 행렬의 대각합은 모든 고윳값의 합과 같다.

$$tr(A) = \sum_{i=1}^{N} \lambda_i$$

③ 행렬의 행렬식은 모든 고윳값의 곱과 같다.

$$\det(A) = \prod_{i=1}^{N} \lambda_i$$

1. 고윳값 분해

5) 고유분해의 성질

N차원 정방행렬 A에 대해

④ 행렬 A가 대칭행렬이면 실수 고윳값 N개를 가지며 고유벡터들이 서로 직교이다.

⑤ 행렬 A가 대칭행렬이고 고윳값이 모두 양수이면, 양의 정부호이고 역행렬이 존재한다. 역도 성립한다.

⑥ 행렬 A가 어떤 행렬 X의 $X^T X$이면 0또는 양의 고윳값을 가진다.

⑦ 행렬 X가 풀랭크이면 $X^T X$은 역행렬이 존재한다.

1. 고윳값 분해

6) 고유공간

- 특정 고유 값에 대응되는 무수히 많은 고유 벡터가 이루는 공간

[고유공간의 성질]

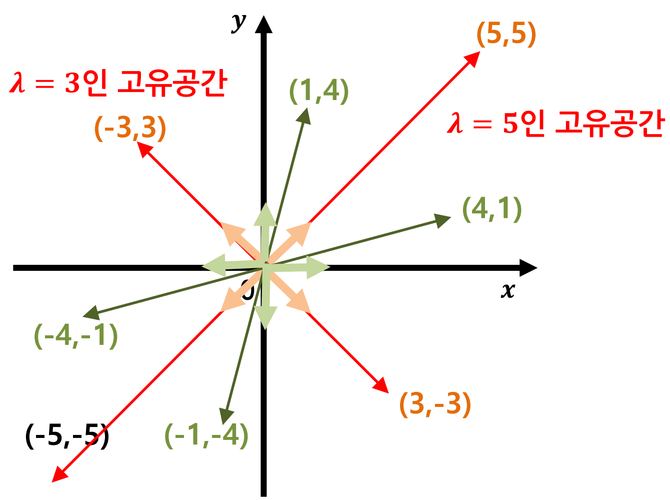
- 고유 값 λ에 대응하는 모든 고유 벡터에 영 벡터를 추가하여 구성된 집합
- 각각의 고유 값 λ에 대응하는 행렬 A의 고유 공간이 있음

1. 고윳값 분해

6) 고유공간

(예시)

(x_1, x_2)	결과	λ값
(1,0)	(4,1)	
(0,1)	(1,4)	
(1,1)	(5,5)	5
(-1,0)	(-4,-1)	
(0,-1)	(-1,-4)	
(-1,1)	(-3,3)	3
(-1,-1)	(-5,-5)	5
(1,-1)	(3,-3)	3



2. 특이값 분해

1) 특이값과 특이벡터

- 고윳값분해는 정방행렬에서만 가능
- 정방행렬이 아닌 행렬에 대해 다음과 같은 3개의 행렬의 곱으로 나타내는 것을 특잇값분해(singular value decomposition) 또는 특이분해라고 함

$$A = U\Sigma V^T$$

- 행렬 Σ의 대각성분 : 특이값(singular value)
- U의 열벡터 : 왼쪽 특이벡터(singular vector)
- V의 행벡터 : 오른쪽 특이벡터(singular vector)

2. 특이값 분해

1) 특이값과 특이벡터

$A = U\Sigma V^T$에서 $U,\ \Sigma,\ V^T$는 다음을 만족한다.

- Σ 는 대각성분이 양수인 대각행렬이고 큰 수부터 작은 수 순서로 배열한다. $\Sigma \in R^{N\times M}$
- U는 N차원 정방행렬로 모든 열벡터가 단위벡터이고 서로 직교해야 한다. $U \in R^{N\times N}$
- V는 M차원 정방행렬로 모든 열벡터가 단위벡터이고 서로 직교해야 한다. $V \in R^{M\times M}$

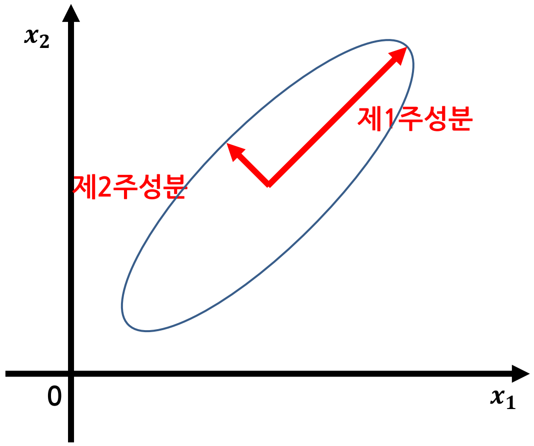
※ 특잇값의 개수는 행렬의 열과 행 개수 중 작은 값과 같음

[파이썬] numpy.linalg 또는 scipy.linalg 서브패키지의 svd()함수 사용

3. 주성분 분석

1) 주성분 분석(PCA; Principal Component Analysis)의 개념

- 다차원 공간의 현상을 저차원 공간으로 축소하여 데이터의 특성을 해석할 수 있는 다변량 자료분석기법 중에 하나
- 주성분분석은 분포된 데이터들의 주성분을 찾아주는 기법



주성분 점수란

- 주성분 축에 정사영 시킨 좌표
- x_1과 x_2의 선형결합으로 표현됨

3. 주성분 분석

2) 공분산 행렬

- 공분산 행렬은 변수 쌍들 간에 갖는 공통분산을 행렬로 표현한 것
- n개의 개체와 m개의 변수의 데이터를 생각하면 $n \times m$ 행렬로 생각할 수 있음

$n \times m$행렬 X에 대해 공분산 행렬 Σ는

$$\Sigma = \frac{1}{n} X^T X$$

3. 주성분 분석

2) 공분산 행렬

[예시]

행렬 $X = \begin{bmatrix} 3 & 4 \\ 5 & 9 \\ 7 & 12 \end{bmatrix}$ 의 공분산행렬은

$$\Sigma = \frac{1}{n} X^T X = \frac{1}{3} \begin{bmatrix} 3 & 5 & 7 \\ 4 & 9 & 10 \end{bmatrix} \begin{bmatrix} 3 & 4 \\ 5 & 9 \\ 7 & 10 \end{bmatrix}$$

$$= \frac{1}{3} \begin{bmatrix} 9+25+49 & 12+45+70 \\ 12+45+70 & 16+81+100 \end{bmatrix}$$

$$= \frac{1}{3} \begin{bmatrix} 83 & 127 \\ 127 & 197 \end{bmatrix} \cong \begin{bmatrix} 27.7 & 42.3 \\ 42.3 & 65.7 \end{bmatrix}$$

3. 주성분 분석

3) 공분산 행렬의 고유값과 고유벡터

- 공분산 행렬의 고유 벡터는 데이터가 어떤 방향으로 많이 분산되어 있는지를 나타냄
- 공분산 행렬의 고윳값은 점을 고유벡터에 정사영 시켰을 때의 분산임
- 고윳값이 큰 순서대로 고유 벡터를 정렬하면 중요한 순서대로 주성분 축이 됨
- 변수 2개로 PCA를 실행하면 주성분 2개
  변수 3개로 PCA를 실행하면 주성분 3개
  …
- 표준화를 통해 고유값의 합은 변수 개수와 같고,
  제 1 주성분의 고유값이 가장 크고 제 n 주성분으로 갈 수록 작아짐
- 주성분들 간에는 직교를 이룸

3. 주성분 분석

4) 주성분 분석의 응용

- m개 변수들의 가중평균을 구하고자 할 때
- 2 or 3차원 점들의 1차원 근사 직선을 찾을 때
- 너무 많고 상관성이 높은 변수들을 서로 독립인 몇 개의 변수로 축소할 때
- 이미지의 노이즈를 제거할 때

출처 : https://t1.daumcdn.net/cfile/tistory/274E8336527C9F5409

거할

정리하기

1. 고윳값 분해

- 고유벡터(eigen vector)는 선형 변환을 취했을 때 방향은 변하지 않고 크기만 변하는 벡터, 이 때 변한 크기가 고유 값(eigen value)임
- 정방행렬 A에 대해 $A\vec{x} = \lambda\vec{x}$($\lambda$는 상수)가 성립하는 0이 아닌 벡터 $\vec{x}$가 존재할 때, 다음을 만족

$$\boldsymbol{A\vec{x}} = \underbrace{\boldsymbol{\lambda}}_{\text{고윳값}}\underbrace{\boldsymbol{\vec{x}}}_{\text{고유벡터}}(\boldsymbol{\lambda}\text{는 상수})$$

- 행렬 곱의 결과가 원래 벡터와 '방향'은 같고, 비율만 $\boldsymbol{\lambda}$만큼 비례하는 변환
- 고윳값 구하는 방법 : $A\vec{x} = \lambda\vec{x} \iff (A - \lambda I)\vec{x} = 0 \iff \det(A - \lambda I) = 0$
- $(A - \lambda I)\vec{x} = 0$ 을 특성방정식이라고 함
- 고유벡터 구하는 방법 : 고윳값에 대응하는 벡터를 구함
- [파이썬] numpy라이브러리의 linalg.eig() 함수 사용
- 대각화 : 고유벡터행렬 V의 역행렬이 존재한다면, 행렬을 고유벡터행렬과 고윳값행렬의 곱으로 표현하는 것
- 고유분해의 성질 7가지가 있음

정리하기

2. 특이값 분해

- 정방행렬이 아닌 행렬에 대해 다음과 같은 3개의 행렬의 곱으로 나타내는

$$A = U\Sigma V^T$$

- 특잇값의 개수는 행렬의 열과 행 개수 중 작은 값과 같음
- [파이썬] numpy.linalg 또는 scipy.linalg 서브패키지의 svd()함수 사용

3. 주성분 분석

- 다차원 공간의 현상을 저차원 공간으로 축소하는 다변량 자료분석기법 중에 하나
- 데이터의 특성을 해석 할 수 있음(분포된 데이터들의 주성분을 찾아주는 기법)
- 공분산 행렬은 변수 쌍들 간에 갖는 공통분산을 행렬로 표현한 것
- 공분산 행렬과 고윳값 분해를 통해 주성분 축들을 찾아냄
- 고윳값은 점을 고유벡터에 정사영 시켰을 때의 분산임
- 변수 개수만큼 주성분 축이 생성됨
- 표준화를 통해 고유값의 합은 변수 개수와 같고, 주성분들 간에는 직교를 이룸
- [파이썬] sklearn.decomposition 서브패키지의 PCA