Euron_ML_Week10

생성일	@November 21, 2024 10:30 AM

6. 차원 축소

01. 차원 축소(Dimension Reduction)개요

🔎 **차원축소**

: 매우 많은 피처로 구성된 다차원 데이터 세트의 차원을 축소해 새로운 차원의 데이터 세트를 생성하는 것

[차원이 증가할수록, 피처가 많을수록]

- 데이터 포인트 간의 거리가 기하급수적으로 멀어지고 희소한 구조를 가짐
- 예측 신뢰도 하락
- 개별 피처 간 상관관계가 높을 가능성이 큼
- 선형 모델에서는 입력 변수 간 상관관계가 높을 경우 다중 공선성 모델→ 예측 성능 저하

> 피처 선택(feature selection)

: 특정 피처에 종속성이 강한 불필요한 피처는 아예 제거하고 데이터의 특징을 잘 나타내는 주요 피처만 선택

> 피처 추출(feature extraction)

: 기존 피처를 저차원의 중요 피처로 압축해서 추출

: 새롭게 추출된 중요 특성은 기존의 피처가 압축된 것(기존의 피처와 완전히 다른 값)

: 단순 압축이 아닌, 피처를 함축적으로 더 잘 설명할 수 있는 또 다른 공간으로 매핑해 추출하는 것

🔎 **대표적 차원 축소 알고리즘: PCA, LDA, SVD, NMF**

- 이미지 데이터

  : 잠재된 특성을 피처로 도출해 함축적 형태의 이미지 변환과 압축을 수행

  → 원본 이미지보다 훨씬 적은 차원으로, 이미지 분류 등 분류 수행 시에 과적합 영향력이 작아져 오히려 원본 데이터로 예측하는 것보다 예측 성능을 더 끌어 올릴 수 있음

- 텍스트 문서의 숨겨진 의미 추출

  : 숨겨져 있는 시맨틱 의미나 토픽을 잠재 요소로 간주하고 이를 찾아낼 수 있음

---

02. PCA(Principal Component Analysis)

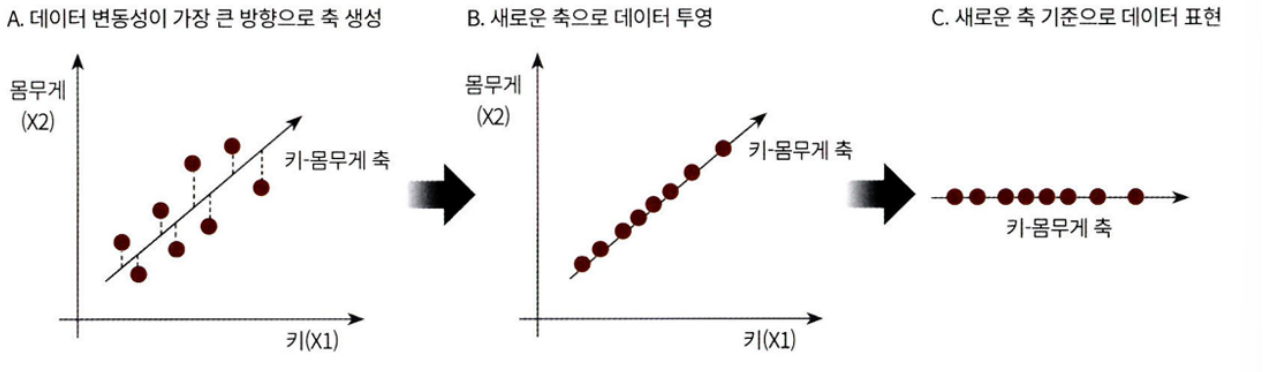
- PCA 개요

- PCA

: 가장 대표적인 차원 축소 기법

: 여러 변수 간에 존재하는 상관관계를 이용해 이를 대표하는 주성분을 추출해 차원을 축소하는 기법

: 기존 데이터의 정보 유실을 최소화하기 위해 가장 높은 분산을 가지는 데이터의 축을 찾아 차원을 축소, 이것이 PCA의 주성분



→ 데이터 변동성이 가장 큰 방향으로 축을 생성하고, 새롭게 생성된 축으로 데이터를 투영하는 방식

→ 생성된 벡터 축에 원본 데이터를 투영하면 벡터 축의 개수만큼의 차원으로 워논 데이터가 차원 축소됨

즉, PCA(주성분 분석)는 원본 데이터의 피처 개수에 비해 매우 작은 주성분으로 원본 데이터의 총 변동성을 대부분 설명할 수 있는 분석법

[선형대수 관점에서 해석]

선형 변환 : 특정 백터에 행렬 A를 곱해 새로운 벡터로 변환하는 것

정방행렬 : 같은 수의 행, 열 갖는 행렬

대칭행렬 : 대각 원소를 중심으로 원소 값이 대칭되는 행렬 ($AT=A$)

고유벡터 : 행렬 A를 곱해도 방향 변하지 않고 크기만 변하는 벡터

- 행렬이 작용하는 힘의 방향과 관계 있어 행렬 분해에 사용됨
- 정방 행렬은 최대 그 차원 수만큼의 고유벡터 가질 수 있음

고윳값: 고유벡터의 크기

공분산 : 두 변수 간의 변동

공분산 행렬 : 여러 변수와 관련된 공분산을 포함하는 정방행렬이며 대칭행렬

- 항상 고유벡터를 직교행렬, 고유값을 정방행렬로 대각화 할 수 있음

$$C = P\Sigma P^T$$

(P: nxn 직교행렬, Σ: nxn 정방행렬)

위 식은 아래와 같이 고유벡터 행렬과 고유값 행렬로 대응됨

$$C = \begin{bmatrix} e_1 & \dots & e_n \end{bmatrix} \begin{bmatrix} \lambda_1 & \dots & 0 \\ \dots & \dots & \dots \\ 0 & \dots & \lambda_n \end{bmatrix} \begin{bmatrix} e_1^t \\ \dots \\ e_n^t \end{bmatrix}$$

입력 데이터의 공분산 행렬을 고유벡터와 고유값으로 분해

→ 분해된 고유벡터로 입력 데이터를 선형 변환

① 입력 데이터 세트의 공분산 행렬 생성

② 공분산 행렬의 고유벡터, 고유값 계산

③ 고유값 가장 큰 순으로 K개(PCA 변환 차수)만큼 고유벡터 추출

④ 추출된 고유벡터로 새롭게 입력 데이터 변환

실습은 Colab에서 실행

---

03. LDA(Linear Discriminant Analysis)

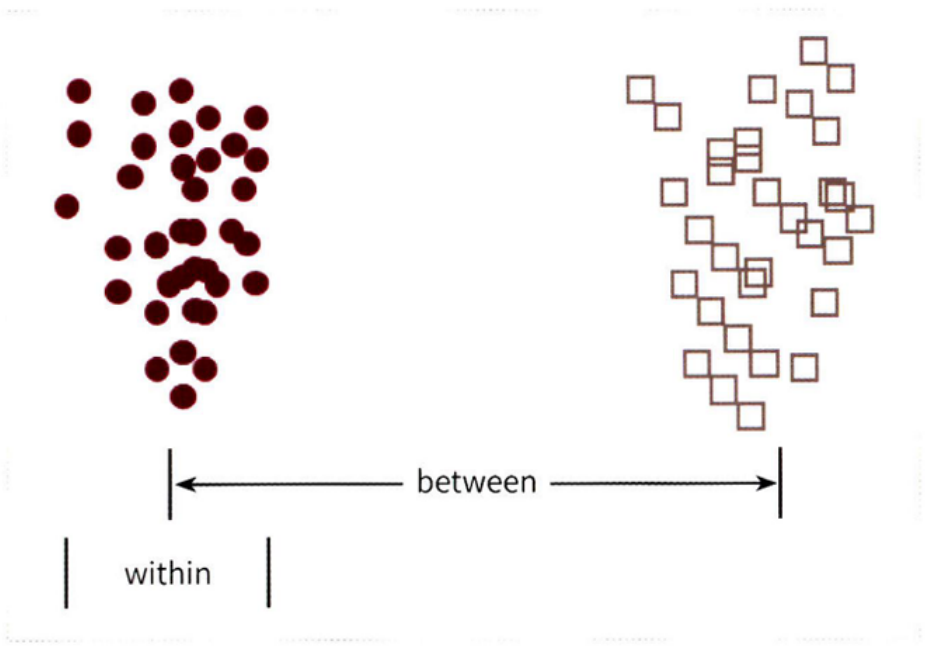
- LDA 개요

- LDA

  : 선형 판별 분석법, PCA와 유사(입력 데이터 세트를 저차원 공간에 투영해 차원 축소)

  : 지도학습의 분류에서 사용하기 쉽도록 개별 클래스를 분별할 수 있는 기준을 최대한 유지하면서 차원 축소

  : 입력 데이터의 결정 값 클래스를 최대한으로 분리할 수 있는 축을 찾음



→ 특정 공간상에서 클래스 분리를 최대화하는 축을 찾기 위해 클래스 간 분산(최대한 크게)과 클래스 내부 분산(최대한 작게)의 비율을 최대화하는 방식으로 차원 축소

1. 클래스 내부와 클래스 간 분산 행렬을 구합니다. 이 두 개의 행렬은 입력 데이터의 결정 값 클래스별로 개별 피처의 평균 벡터(mean vector)를 기반으로 구합니다.
2. 클래스 내부 분산 행렬을 S_W, 클래스 간 분산 행렬을 S_B라고 하면 다음 식으로 두 행렬을 고유벡터로 분해할 수 있습니다.

$$S_W^T S_B = \begin{bmatrix} e_1 & \cdots & e_n \end{bmatrix} \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \cdots & \cdots & \cdots \\ 0 & \cdots & \lambda_n \end{bmatrix} \begin{bmatrix} e_1^T \\ \cdots \\ e_n^T \end{bmatrix}$$

3. 고유값이 가장 큰 순으로 K개(LDA변환 차수만큼) 추출합니다.
4. 고유값이 가장 큰 순으로 추출된 고유벡터를 이용해 새롭게 입력 데이터를 변환합니다.

- 붓꽃 데이터 세트에 LDA 적용하기

Colab에서 실행

---

04. SVD(Singular Value Decomposition)

- SVD 개요

- SVD

  : PCA와 유사한 행렬 분해 기법

  - 차이: 정방행렬뿐 아니라 행, 열 크기 다른 행렬에도 적용 가능

: 특이값 분해

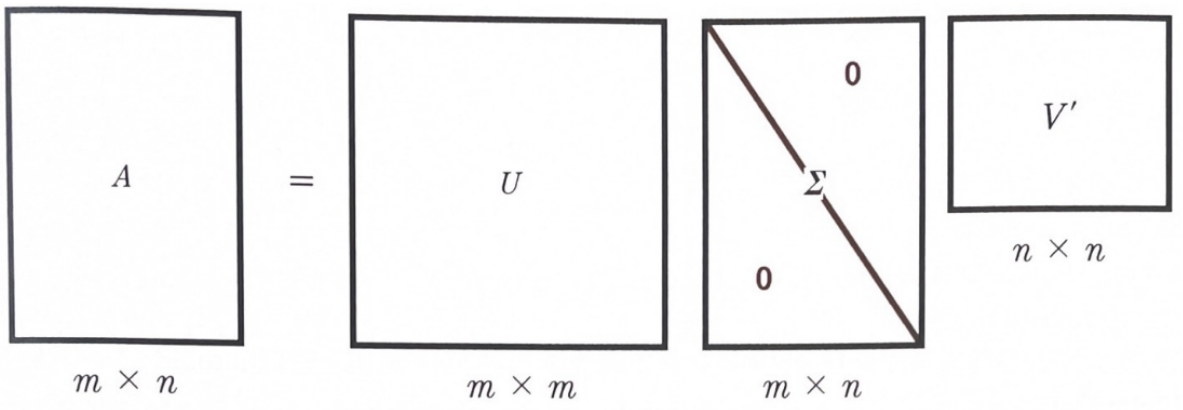
$$A = U \sum V^T$$

특이 벡터: U와 V에 속한 벡터, 서로 직교하는 성질

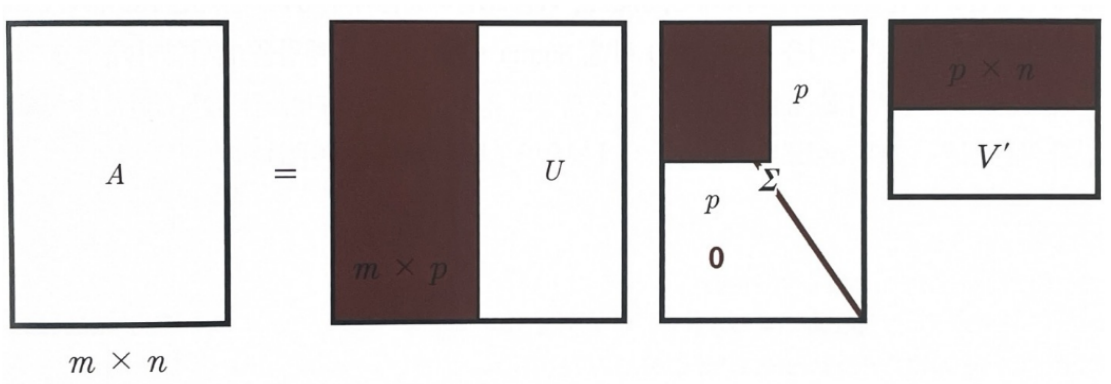
Σ 대각 행렬: 행렬의 대각에 위치한 값만 0이 아니고 나머지 위치 값은 모두 0

특이값 : Σ에 위치한 0이 아닌 값

- SVD는 A의 차원이 mxn일 때 분해 : *U* mxm, Σ mxn, *VT* nxn



- 일반적으로 Σ의 비대각인 부분, 대각 원소중 특이값 0인 부분도 제거하여 차원 줄인 형태로 SVD 적용



- 사이킷런 TruncatedSVD 클래스를 이용한 변환

Colab에서 실행

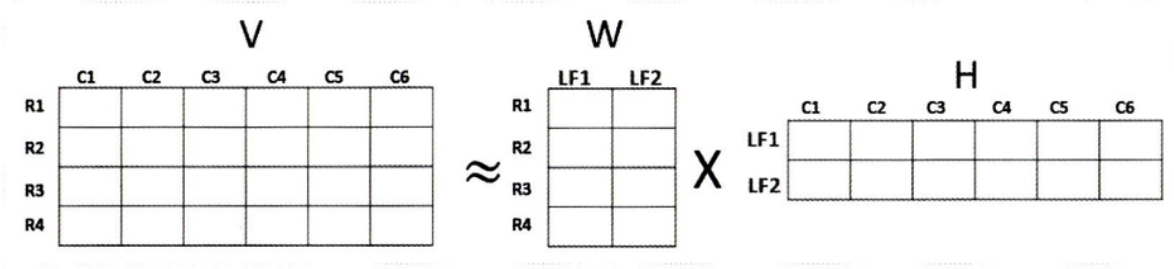
05. NMF(Non-Negative Matrix Factorization)

- NMF 개요

- NMF

  : Truncated SVD와 같이 낮은 랭크를 통한 행렬 근사 방식의 변형

  : 원본 행렬 내의 모든 원소 값이 모두 양수(0 이상)라는 게 보장되면 아래 그림과 같이 좀 더 간단하게 두 개의 기반 양수 행렬로 분해될 수 있는 기법 지칭



길고 가는 행렬 W, 작고 넓은 행렬 H로 분해

W는 원본 행에 대해 잠재 요소 값이 얼마나 되는지 대응,

H는 잠재 요소가 원본 열로 어떻게 구성됐는지 나타냄.

→ SVD와 유사하게 이미지 압축을 통한 패턴 인식, 텍스트의 토픽 모델링 기법, 문서 유사도 및 클러스터링에 잘 사용됨

→ 영화 추천과 같은 추천 영역에 활발하게 적용됨 (사용자-평가 순위 데이터 세트를 행렬 분해 기법을 통해 분해하면서 사용자가 평가하지 않은 상품에 대한 잠재적인 요소를 추출해 이를 통해 평가 순위를 예측하고 높은 순위로 예측된 상품을 추천해주는 방식)