Ch 6. 차원 축소

1. 차원 축소 개요

차원 축소 : 다차원 데이터 세트의 차원을 축소해 새로운 차원의 데이터 세트를 생성

다차원의 피처를 차원 축소해 피처 수를 줄이면 더 직관적으로 데이터를 해석할 수 있음

차원 축소는 일반적으로 피처 선택과 피처 추출로 나뉨

피처 선택 : 특정 피처에 종속성이 강한 불필요한 피처는 아예 제거, 주요 피처만 선택

피처 추출 : 기존 피처를 저차원의 중요 피처로 압축해서 추출. 기존의 피처와는 완전히 다른 값이 됨 -> 피처를 함축적으로 더 잘 설명할 수 있는 또 다른 공간으로 매핑해 추출. 이는 기존 피처가 전혀 인지하기 어려웠던 잠재적인 요소를 추출하는 것을 의미

PCA, SVD, NMF는 잠재적인 요소를 찾는 대표적인 차원 축소 알고리즘

p.378

2. PCA (Principal Component Analysis)

PCA 개요

PCA : 주성분 분석. 가장 대표적인 차원 축소 기법. 여러 변수 간 상관관계를 이용해 이를 대표하는 주성분을 추출해 차원을 축소. 가장 높은 분산을 가지는 데이터의 축을 찾아 이 축으로 차원을 축소, 이것이 주성분이 됨. p.379

가장 큰 데이터 변동성을 기반으로 첫번째 벡터 축을 생성. 두 번째 축은 이 벡터 축에 직각이 되는 벡터(직교 벡터)를 축으로 함. 세번째 축은 다시 두 번째 축과 직각이 되는 벡터를 설정하는 방식으로 축을 생성. 이렇게 생성된 벡터 축에 원본 데이터를 투영하면 벡터 축의 개수만큼의 차원으로 차원 축소됨.

선형 변환 : 특정 벡터에 행렬 A를 곱해 새로운 벡터로 변환하는 것. 특정 벡터를 하나의 공간에서 다른 공간으로 투영하는 것

공분산 행렬 : 여러 변수와 관련된 공분산을 포함하는 정방형 행렬. 정방행렬이며 대칭행렬.

고유벡터 : 행렬 A를 곱하더라도 방향이 변하지 않고 크기만 변하는 벡터. 여러 개가 존재. 정방

행렬은 최대 그 차원의 수만큼의 고유벡터를 가질 수 있음. 행렬이 작용하는 힘의 방향과 관계가 있어서 행렬을 분해하는 데 사용됨

⇨ 입력 데이터의 공분산 행렬이 고유벡터와 고유값으로 분해될 수 있으며, 이렇게 분해된 고유벡터를 이용해 입력 데이터를 선형 변환하는 방식이 PCA라는 것!

PCA 과정

1. 입력 데이터의 공분산 행렬을 생성
2. 공분산 행렬의 고유벡터와 고유값을 계산
3. 고유값이 가장 큰 순으로 K개(PCA 변환 차수만큼) 고유벡터를 추출
4. 고유값이 가장 큰 순으로 추출된 고유벡터를 이용해 새롭게 입력 데이터를 변환

PCA는 많은 속성으로 구성된 원본 데이터를 그 핵심을 구성하는 데이터로 압축한 것

PCA는 여러 속성의 값을 연산해야 하므로 속성 스케일에 영향을 받는다

따라서 여러 속성을 PCA로 압축하기 전에 각 속성값을 동일한 스케일로 변환하는 것이 필요

PCA : 사이킷런에서 PCA 변환을 위해 제공하는 클래스. 생성 파라미터로 n_components를 입력받음. fit, transform으로 수행

PCA 변환을 수행한 PCA 객체의 explained_variance_ratio_ 속성으로 전체 변동성에서 개별 PCA 컴포넌트 별로 차지하는 변동성 비율을 확인

read_excel() : 엑셀 파일명과 엑셀 시트명을 입력하면 엑셀 파일 로드

3. LDA (Linear Discriminant Analysis

LDA 개요 p.394

LDA : 선형 판별 분석법. PCA처럼 입력 데이터 세트를 저차원 공간에 투영해 차원을 축소하는 기법. but 지도학습 분류에서 사용하기 쉽도록 개별 클래스를 분별할 수 있는 기준을 최대한 유지하면서 차원을 축소.

(PCA는 비지도 학습, LDA는 지도학습)

입력 데이터의 결정 값 클래스를 최대한으로 분리할 수 있는 축을 찾음. 클래스 간 분산은 최대화, 클래스 내부분산은 최소화

->클래스 간 분산과 클래스 내부 분산 행렬을 생성한 뒤, 이 행렬에 기반해 고유벡터를 구하고 입력 데이터를 투영

4. SVD (Singular Value Decomposition)

SVD 개요

SVD : 특이값 분해. PCA와 유사한 행렬 분해 기법 이용. PCA는 정방행렬만 분해 가능하지만 SVD는 정방행렬뿐만 아니라 행과 열의 크기가 다른 행렬에도 적용 가능.

Trancated SVD : Sum의 대각원소 중에 상위 몇 개만 추출해서 여기에 대응하는 U와 V의 원소도 함께 제거해 더욱 차원을 줄인 형태로 분해하는 것. 특이값 중 상위 일부 데이터만 추출해 분해. 넘파이가 아닌 사이파이에서만 지원.

사이킷런 TruncatedSVD 클래스를 이용한 변환

TruncatedSVD : 사이킷런에서 Truncated SVD 수행하는 클래스. 사이파이의 svds와 같이 U, Sigma, Vt를 반환하지는 않음. fit()과 transform() 통해 원본 데이터를 몇 개의 주요 컴포넌트로 차원 축소해 변환. 원본 데이터를 Truncated SVD 방식으로 분해된 U*Sigma 행렬에 선형 변환해 생성

5. NMF (Non-Negative Matrix Factorization)

NMF 개요 p.405

NMF : Truncated SVD와 같이 낮은 랭크를 통한 행렬 근사 방식의 변형. 원본 행렬 내의 모든 원소 값이 모두 양수라는 게 보장되면 두 개의 기반 양수 행렬로 분해될 수 있는 기법.

분해 행렬 W : 원본 행에 대해서 이 잠재 요소의 값이 얼마나 되는지에 대응

분해 행렬 H : 잠재 요소가 원본 열(속성)에 대해서 어떻게 구성됐는지.

사이킷런 NMF 클래스를 통해서 지원

행렬 분해는 일반적으로 SVD와 같은 행렬 분해 기법을 통칭