머신러닝 - 차원축소

2021

1. 개요

❖ 차원 축소

- 차원이 증가할수록 데이터 포인트 간의 거리가 기하급수적으로 멀어짐
- 희소(sparse)한 구조를 가짐
- 상대적으로 저차원에서 학습된 모델보다 예측 신뢰도가 떨어짐
- 좀 더 데이터를 잘 설명할 수 있는 잠재적인 요소를 추출
- 데이터 시각화

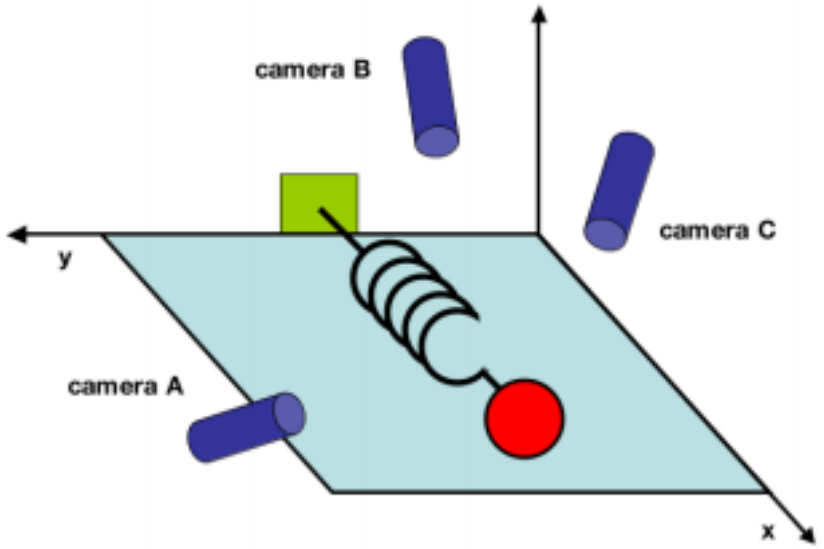
❖ 종류

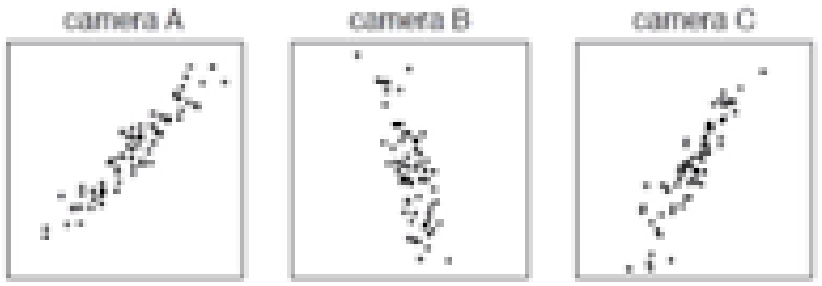
- 피처 선택(Feature Selection)
  - 특정 피처에 종속성이 강한 불필요한 피처는 제거
  - 데이터의 특징을 잘 나타내는 주요 피처만 선택
- 피처 추출(Feature Extraction)
  - 기존 피처를 저차원의 중요 피처로 압축해서 추출하는 것
  - 피처를 함축적으로 더 잘 설명할 수 있는 또 다른 공간으로 매핑

2. PCA(Principal Component Analysis)

❖ 개요

- 보는 방향에 따라 추의 움직임이 다르게 보임

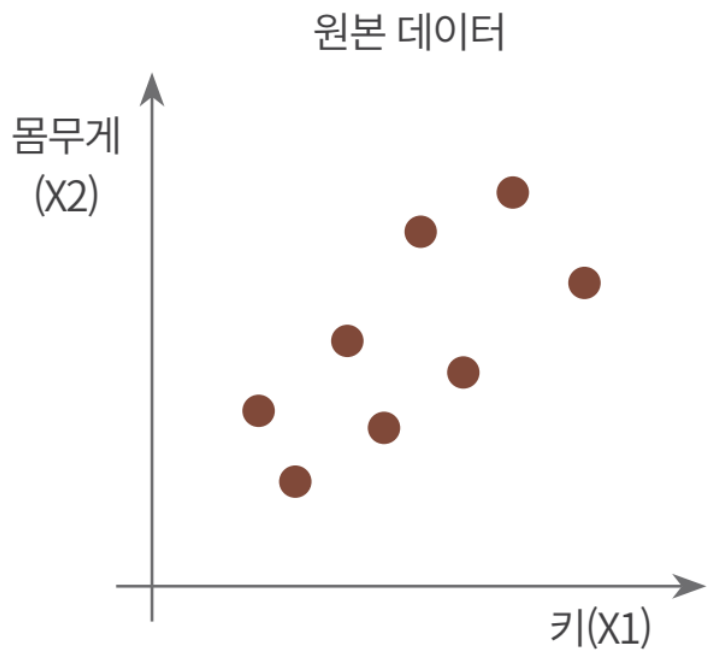




2. PCA(Principal Component Analysis)

❖ PCA

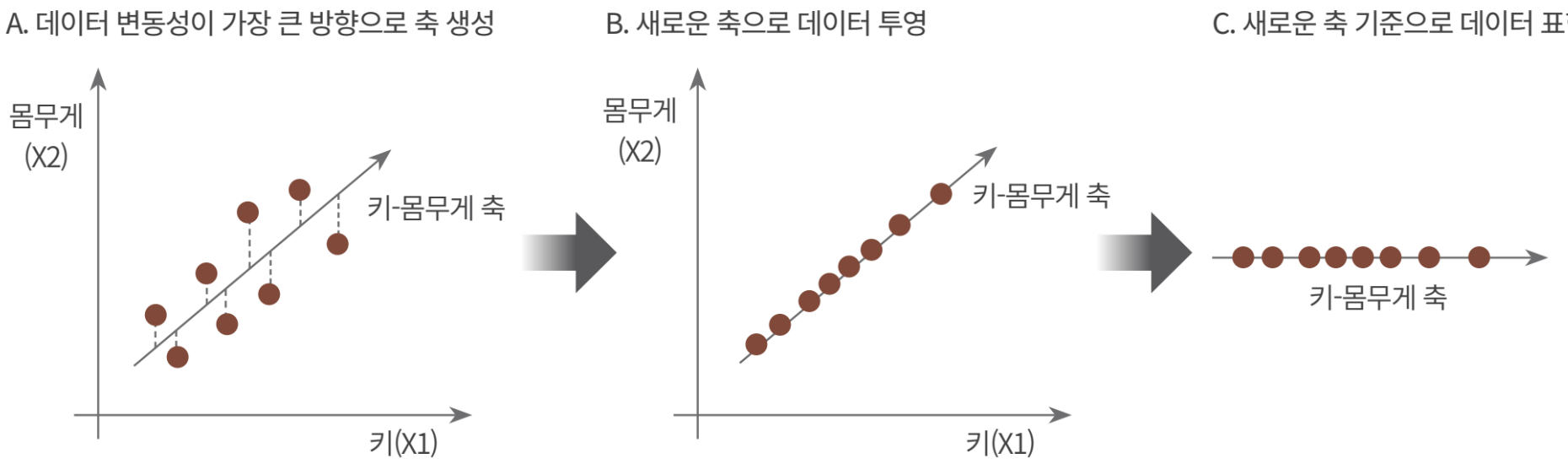
- 주성분을 추출해 차원을 축소하는 방법
- 기존 데이터의 정보 유실이 최소화되도록 함
- 가장 높은 분산을 가지는 데이터의 축을 찾아 차원을 축소



2. PCA(Principal Component Analysis)

❖ PCA

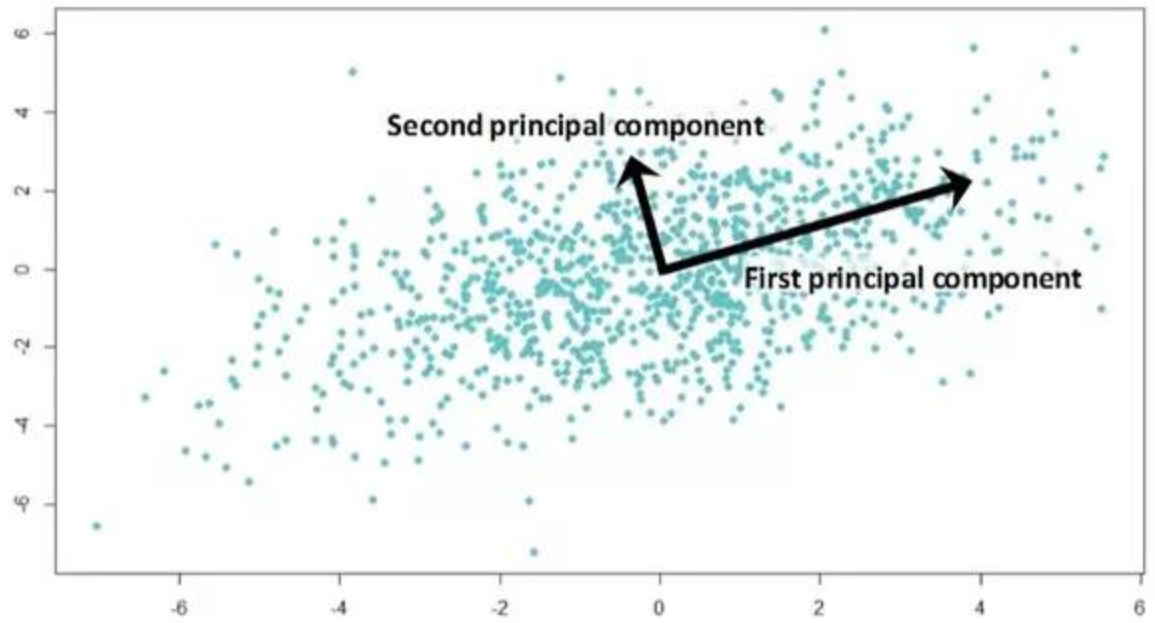
- 데이터의 변동성이 가장 큰 방향으로 축을 생성
- 새롭게 생성된 축으로 데이터를 투영하는 방식



2. PCA(Principal Component Analysis)

❖ PCA

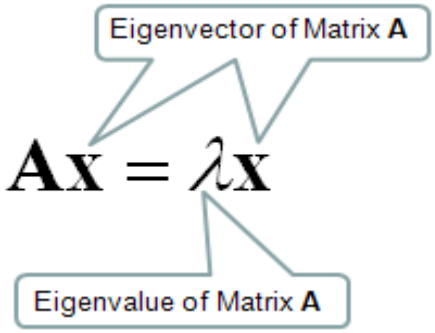
- 데이터의 변동성을 기반으로 첫번째 벡터 축을 생성
- 두번째 축은 이 벡터 축에 직각이 되는 벡터(직교 벡터)를 축으로 함



2. PCA(Principal Component Analysis)

❖ 고유값, 고유벡터

- 행렬 A를 선형 변환한 결과가 자기 자신의 상수배가 될 경우



- 고유벡터는 행렬 A를 곱하더라도 방향이 변하지 않고 그 크기만 변화
- (n, n) 행렬의 경우 n개의 고유벡터를 가짐
- 고유벡터는 서로 직교(orthogonal)함

- 고유벡터를 이용해 입력 데이터를 선형 변환하는 방식 - PCA

2. PCA(Principal Component Analysis)

❖ PCA(Principal Component Analysis)

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

2. PCA(Principal Component Analysis)

❖ 사이킷 런에서 변환

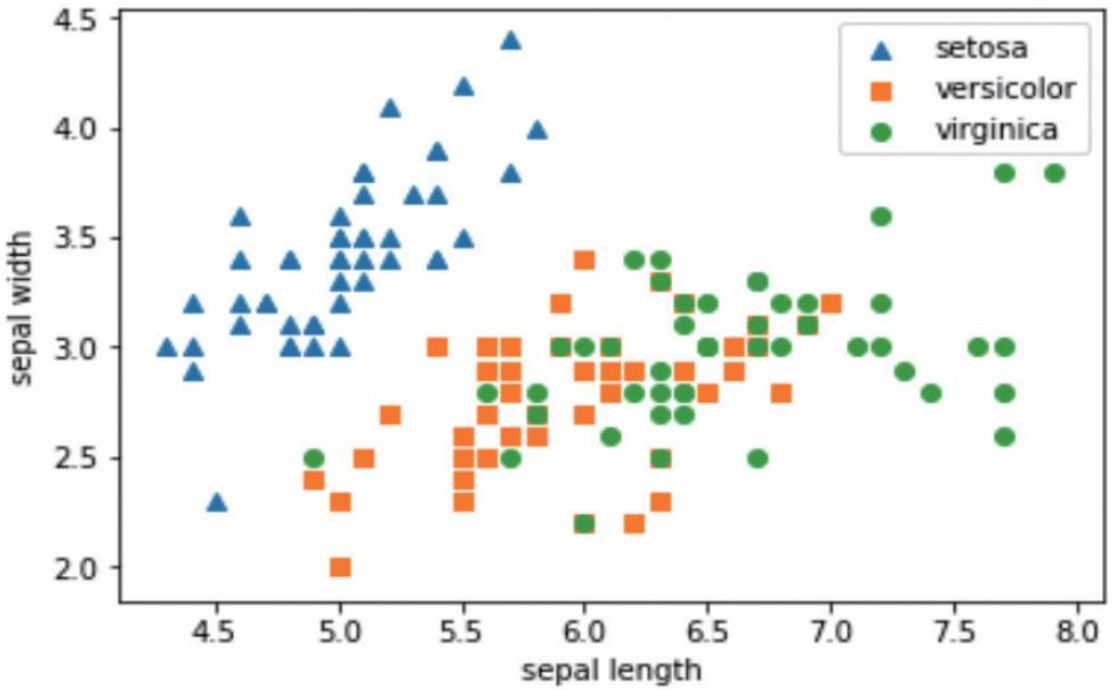
1. 정규화 전처리
2. PCA 객체 생성. 이때 변환 차수(K) 등의 옵션 세팅
3. 학습 (fit)
4. 변환 (transform)

➔ 이후에 다른 머신 러닝 알고리즘을 활용하여 분류/회귀 수행

2. PCA(Principal Component Analysis)

❖ Iris 변환 전

	sepal_length	sepal_width	petal_length	petal_width	target
0	5.1	3.5	1.4	0.2	0
1	4.9	3.0	1.4	0.2	0
2	4.7	3.2	1.3	0.2	0



2. PCA(Principal Component Analysis)

❖ Iris 변환 후

	pca_component_1	pca_component_2	target
0	-2.264703	0.480027	0
1	-2.080961	-0.674134	0
2	-2.364229	-0.341908	0