Chap6 개인 요약본

Week 10 발표자료 요약본

:차원 축소 Dimensionality Reduction

6.1 차원 축소 개요

차원 축소(Dimensionality Reduction): 다수의 피처(feature)를 가진 고차원 데이터에서, 데이터의 본질적 특성을 유지하면서 **피처 수를 줄이는 과정**을 의미

고차원 데이터의 문제:

1. **데이터 희소성(Sparsity)** – 차원이 커질수록 포인트 간 거리가 멀어져 밀집도가 낮아진다.
2. **모델 신뢰도 저하** – 차원이 높을수록 데이터가 분산되어 학습이 어려워진다.
3. **다중 공선성(Multicollinearity)** – 피처 간 상관관계가 높으면 선형 모델의 안정성이 떨어진다.

→ 따라서, 차원 축소를 통해 피처를 줄이면 모델의 해석력과 일반화 성능이 높아진다.

---

* 차원 축소의 두 가지 방식

1. **피처 선택 (Feature Selection)**

   불필요하거나 종속적인 피처를 제거하고, 의미 있는 피처만 남긴다.

   → 예: 상관관계가 높은 변수 중 하나만 남기는 방식.

2. **피처 추출 (Feature Extraction)**

   기존 피처들을 결합하여 더 함축적이고 정보량이 높은 새로운 피처로 변환한다.

   → 예: 학생 평가 항목(내신, 봉사, 수상)을 종합하여 학업성취도 로 표현.

즉, 피처 선택은 기존 피처 중 일부를 고르는 것이고,

피처 추출은 새로운 축으로 데이터를 투영하여 전혀 다른 공간으로 매핑하는 것이다.

차원의 저주 (Curse of Dimensionality)

차원이 증가할수록 점들 사이의 평균 거리가 기하급수적으로 멀어진다.

예:

- 3차원 공간에서는 두 점 사이 평균 거리가 약 0.66
- 1,000,000차원 공간에서는 평균 거리가 428.35

즉, 대부분의 점들이 멀리 떨어져 있으며 새 샘플도 기존 샘플과 유사하지 않다.

이로 인해 모델의 **일반화 능력 저하**와 **과적합 위험**이 커진다.

차원 축소 접근 방식

1. **투영(Projection)**
   - 고차원 공간의 데이터가 사실상 저차원 평면 위에 존재할 경우,

     그 평면에 데이터를 수직 투영(projection)하여 차원을 줄인다.
   - 예: 3D 공간의 데이터가 거의 2D 평면에 놓여 있을 경우,

     이를 z1, z2 축으로 투영해 2차원 데이터로 표현.
2. **매니폴드 학습(Manifold Learning)**
   - 실제 데이터는 고차원이지만, 그 안에 숨겨진 저차원 구조(매니폴드)가 있다고 가정.
   - 예: '스위스 롤(swiss roll)' 형태의 데이터는 3D 공간에 있지만,

     말려 있는 2D 구조로 "펴서(flatten)" 2D로 표현할 수 있음.

   다만, 매니폴드 공간에서 분류나 회귀가 항상 더 단순해지는 것은 아니다.

6.2 PCA (Principal Component Analysis)

PCA는 데이터의 분산이 가장 큰 방향을 찾아 새로운 축(주성분)을 형성함으로써 차원을 축소하는 가장 대표적인 비지도 학습 방법

PCA의 핵심 아이디어

1. 데이터 변동성이 가장 큰 축을 찾는다.
2. 그 축에 수직인 새로운 축을 추가로 찾는다.
3. 이렇게 찾은 축들에 데이터를 투영(projection)하여 새 좌표계로 변환한다.

이 과정에서 첫 번째 축(주성분 1)이 전체 변동성의 가장 큰 비율을 설명하고,

두 번째 축이 그다음 변동성을 설명한다.

선형대수 관점

- 공분산 행렬을 고유값분해(Eigendecomposition) 하여
  고유벡터(Eigenvector)를 주성분 벡터로 사용한다.
- 고유값(Eigenvalue)은 해당 축이 설명하는 분산의 크기를 나타낸다.

즉, PCA는 입력 데이터를 **고유벡터 방향으로 선형 변환**하는 과정이다.

수식:

$C = P\Sigma P^T$

(C: 공분산 행렬, P: 직교 행렬, Σ: 고유값 대각행렬)

붓꽃(Iris) 데이터 예제

- 원래 4차원 데이터 (꽃받침·꽃잎 길이/너비) → 2차원으로 축소
- PCA로 변환 시, Setosa는 명확하게 분리되고
  Versicolor와 Virginica는 일부 중첩되지만 구분 가능

설명된 분산 비율(Explained Variance Ratio)

- 첫 번째 주성분: 72.9%
- 두 번째 주성분: 22.8%
  → 두 축만으로도 95% 이상의 변동성을 설명 가능.

신용카드 고객 데이터 예제

- 30,000개 레코드, 24개 속성 중 'BILL_AMT1~6' 간 상관도 0.9 이상
- 단 2개의 PCA 축만으로 전체 변동성의 95% 이상 설명 가능
  → PCA는 상관도가 높은 변수들을 효과적으로 압축한다.

PCA의 장단점

✅ **장점**

- 계산 속도가 빠르고, 데이터 시각화 및 압축에 탁월
- 고차원에서 패턴이나 이상치를 파악하기 용이

⚠️ **한계**

- 선형 관계만 반영, 비선형 데이터 구조 표현 불가
- 데이터 스케일에 민감 → 반드시 표준화 필요

6.3 LDA (Linear Discriminant Analysis)

LDA는 지도학습 기반의 차원 축소 방법으로,

클래스 간 분리도를 최대화하는 선형 축을 찾는다.

PCA vs LDA

구분	PCA	LDA
학습 형태	비지도 (라벨 사용 X)	지도 (클래스 라벨 사용)
목적	분산을 최대화하는 축	클래스 분리를 극대화하는 축
사용 예시	시각화, 전처리	분류(Classification)

원리

LDA는 클래스 간 분산(between-class scatter)은 크고,

클래스 내 분산(within-class scatter)은 작게 만드는 축을 찾는다.

이 비율을 최대화하는 방향으로 투영(Projection)하여 데이터를 변환한다.

붓꽃 데이터 예시

1. 데이터 스케일링 → 평균 0, 표준편차 1
2. LDA 변환 (n_components=2)
3. 클래스 라벨(y) 필요 (지도학습)

결과적으로 PCA보다 클래스 간 분리도가 높다.

특히 Setosa, Versicolor, Virginica 세 그룹이 명확히 구분됨.

6.4 SVD (Singular Value Decomposition)

SVD는 PCA와 유사하지만, **정방행렬뿐 아니라 임의 크기의 행렬에도 적용 가능**한 분해 기법이다.

핵심 수식

$A = U\Sigma V^T$

- U: 좌특이벡터 (입력 행렬의 행 방향 패턴)
- Σ: 특이값(대각행렬, 각 축의 중요도)

- V^T: 우특이벡터 (열 방향 패턴)

Truncated SVD

- Σ의 상위 p개 특이값만 사용하여 저차원 근사 행렬 생성
- 즉, 정보 손실은 있지만 압축률과 효율성 향상
- Scipy의 `svds()`로 구현 가능 (희소행렬 전용)

예제

- 6×6 행렬을 SVD로 분해 후 일부 특이값만 사용해 복원 → 근사치로 재구성
- 붓꽃 데이터에 TruncatedSVD 적용 시, PCA 결과와 거의 동일

  → 실제로 사이킷런의 PCA는 내부적으로 SVD를 이용해 구현되어 있음.

👉 따라서,

PCA는 밀집행렬(Dense)에 적합하고,

SVD는 희소행렬(Sparse)에 적합하다.

6.5 NMF (Non-Negative Matrix Factorization)

NMF는 모든 값이 양수인 행렬을 두 개의 양수 행렬로 분해하는 기법이다.

A ≈ W × H

- W: 잠재 특성(Latent Feature)의 가중치
- H: 각 특성의 조합 정도

특징

- Low-Rank Approximation 기반
- 데이터의 의미 구조를 보존하면서 압축
- 행렬 내 음수가 없기 때문에 해석이 직관적

예시: 추천 시스템

사용자-상품 평점 행렬을 NMF로 분해하여,

평가하지 않은 상품에 대한 **잠재적 선호도**를 추정 → 추천에 활용

또한 NMF는 이미지 분해, 문서 토픽 모델링, 텍스트 유사도 분석 등에도 활용된다.

6.6 기타 차원 축소 기법

1. **랜덤 투영 (Random Projection)**
    - 무작위 선형 변환으로 차원 축소
    - 계산이 빠르고 대용량 데이터에 적합
2. **MDS (Multi-Dimensional Scaling)**
    - 샘플 간의 거리(distance)를 보존하는 방식
    - 비선형 구조 시각화 가능하지만 계산량 많음
3. **Isomap**
    - 이웃 그래프를 기반으로 지오데식 거리(Geodesic Distance)를 유지
    - 스위스 롤과 같은 곡면형 데이터에 강함
4. **t-SNE**
    - 데이터의 지역적 구조(Local Structure) 보존
    - 군집 시각화에 매우 강력하지만 계산량이 큼

6.7 Kaggle 실습: Dimensionality Reduction for Beginners

📍(슬라이드: 캐글 노트북)

데이터: 유방암 진단 데이터 (악성/양성)

기법: PCA, MDS, t-SNE 비교

1. **PCA** – 분산 보존
    - 빠르고 전체 구조 파악에 유리하지만, 비선형 관계는 표현 한계
2. **MDS** – 거리 보존
    - 두 그룹이 보다 뚜렷하게 분리됨, 그러나 속도가 느림
3. **t-SNE** – 이웃 보존
    - 두 그룹을 가장 명확히 군집화함
    - 시각화 품질 높지만 계산 복잡도 큼

결론:

- PCA는 전체 구조 탐색용
- MDS는 거리 관계 분석용
- t-SNE는 군집 시각화용으로 가장 적합

기법	핵심 원리	장점	한계
PCA	분산 보존	빠르고 해석 용이	비선형 구조 표현 어려움
LDA	클래스 분리	분류 전처리에 유용	지도 학습 필요
SVD	특이값 분해	밀집·희소 데이터 모두 가능	수학적 해석 복잡
NMF	양수 행렬 분해	해석이 직관적, 추천에 유용	음수 데이터 불가
t-SNE	이웃 보존	군집 시각화 탁월	계산량 많음