**군집화**

**K-평균**

K-평균은 군집 중심점(centroid)이라는 특정한 임의의 지점을 선택해 해당 중심에 가장 가까운 포인트들을 선택하는 군집화 기법

군집 중심점은 선택된 포인트 들의 평균 지점으로 이동하고 이동된 중심점에서 다시 가까운 포인트를 선택, 다시 평균 지점으로 이동하는 프로세스를 반복적으로 수행하고 중심점 이동했는데 데이터들의 중심점 소속 변경이 없으면 군집화 완료

* 장점: 쉽고 간결, 일반적인 군집화에서 가장 많이 활용되는 알고리즘
* 단점: 거리 기반으로 속성 개수 매우 많으면 군집화 정확도 떨어짐, 반복 횟수 많으면 수행 시간 느려짐, 몇 개의 군집을 선택해야 할지 가이드 어려움

사이킷런은 K-평균 구현하기 위해 KMeans 클래스 제공

중요 파라미터

* n\_clusters: 군집화할 개수, 군집 중심점의 개수
* init: 초기에 군집 중심점의 좌표를 설정할 방식, 보통 k-means++ 방식으로 최초 설정
* max\_iter: 최대 반복 횟수, 이 횟수 이전에 모든 데이터의 중심점 이동 없으면 종료

비지도 학습 클래스와 마찬가지로 fit(), fit\_transform() 메소드 이용해 수행

주요 속성

* labels\_: 각 데이터 포인트가 속한 군집 중심점 레이블
* cluster\_centers\_: 각 군집 중심점 좌표(shape는 [군집 개수, 피처 개수])

**군집화 알고리즘 테스트를 위한 데이터 생성**

둘 다 여러 개의 클래스에 해당하는 데이터 세트 만드는데, 하나의 클래스에 여러 개의 군집이 분포될 수 있게 데이터 생성 🡪 분류 용도로도 테스트 데이터 생성 가능

* make\_blobs(): 개별 군집의 중심점과 표준 편차 제어 기능
* make\_classification(): 노이즈를 포함한 데이터를 만드는 데 유용
* make\_circle(), make\_moon()은 중심 기반의 군집화로 해결하지 어려운 데이터 세트 만듦

make\_blobs()의 호출 파라미터

* n\_samples: 생성할 총 데이터 개수, 디폴트 100
* n\_features: 데이터의 피처 개수
* centers: int 값, 3으로 설정하면 군집의 개수, ndarray로 표현하면 개별 군집 중심점의 좌표 의미
* cluster\_id: 생성될 군집 데이터의 표준 편차, 군집별로 다른 표준편차 설정할 때 사용

**군집 평가(Cluster Evaluation)**

군집화는 데이터 내에 숨어 있는 별도의 그룹을 찾아서 의미를 부여하거나 동일한 분류 값에 속하더라도 더 세부화된 군집화를 추구하거나 서로 다른 분류 값의 데이터도 더 넓은 군집화 레벨화 등의 영역을 가지고 있음 🡪 분류와 많이 다름

군집화의 성능 평가하는 대표적 방법 🡪 **실루엣 분석(silhouette analysis)**

실루엣 분석: 각 군집 간의 거리가 얼마나 효율적으로 분리되어 있는지 나타냄

개별 데이터가 가지는 군집화 지표인 **실루엣 계수**를 기반으로 함

* 같은 군집 내의 데이터와는 얼마나 가깝게 군집화 되어 있고, 다른 군집 데이터와는 얼마나 멀리 분리되어 있는지 나타내는 지표
* a(i): 해당 데이터와 같은 군집 내의 다른 데이터들과 평균 거리, b(i): 가장 가까운 군집의 데이터들과 평균 거리 🡪 s(i) = (b(i) – a(i)) / max(a(i), b(i))
* -1에서 1 사이의 값, 1로 가까워질수록 근처 군집과 멀리 떨어짐, 0에 가까울수록 근처 군집과 가까워짐, - 값은 아예 다른 군집에 데이터가 할당되었음을 의미

Sklearn.metrics.silhouette\_samples(): 인자로 X features와 y label 입력해주면 각 데이터 포인트의 실루엣 계수 반환

Sklearn.metrics.silhouette\_score(): 인자로 X features와 y label 입력해주면 전체 데이터의 실루엣 계수 평균해 반환

좋은 군집화가 되려면

* 전체 실루엣 계수의 평균 값은 0~1 사이를 가지며 1에 가까울수록 좋음
* 하지만 전체 실루엣 평균값과 더불어 개별 군집의 평균값의 편차가 작아야 함, 개별 군집의 실루엣 계수 평균값이 전체 평균값에서 크게 벗어나지 않아야 함

실루엣 계수를 통한 K-평균 군집 평가 방법은 직관적으로 이해하기는 쉽지만 각 데이터 별로 다른 데이터와의 거리를 반복적으로 계산해야 하므로 데이터 양 늘어나면 수행시간 크게 증가

**평균 이동(Mean Shift)**

군집의 중심을 데이터가 모여 있는 밀도가 가장 높은 곳으로 이동 🡪 확률 밀도 함수 찾기 위해 KDE 이용

개별 데이터의 특정 반경 내에 주변 데이터를 포함한 데이터 분포도를 KDE 기반의 Mean Shift 알고리즘으로 계산하여 데이터 분포도가 높은 방향으로 데이터 이동, 개별 데이터들이 군집 중심점으로 모임 🡪 지정된 반복 횟수만큼 데이터를 이동시키면서 군집화 수행 🡪 개별 데이터들이 모인 중심점을 군집 중심점으로 설정

KDE: 커널 함수를 통해 어떤 변수의 확률 밀도 함수 추정하는 대표적 방법

* 개별 관측 데이터에 커널 함수를 적용한 뒤, 이 적용 값을 모두 더한 후 개별 관측 데이터의 건수로 나눠 확률 밀도 함수 추정, 대표적인 커널 함수는 가우시안 분포 함수
* h: 대역폭, 부드러운 또는 뾰족한 형태로 평활화하는 데 적용
* 작은 h 값은 좁고 뾰족한 KDE를 가지게 되며 변동성이 큰 방식으로 PDF 추정하여 과적합하기 쉬움, 큰 h 값은 과도하게 평활화된 KDE로 인해 단순화된 방식으로 PDF 추정하여 과소적합 하기 쉬움 🡪 적절한 KDE의 대역폭 h 계산!

평균 이동 군집화는 대역폭이 클수록 적은 수의 군집 중심점 가지며 대역폭이 적을수록 많은 군집 중심점 가짐 🡪 군집 개수 지정하지 않으며 **대역폭의 크기에 따라 군집화 수행**

사이킷런은 MeanShift 클래스 제공, 가장 중요한 초기화 파라미터는 bandwidth, 최적의 대역폭 계산을 위해 estimate\_bandwidth() 메소드 제공

데이터 세트의 형태를 특정 형태로 가정한다든가, 특정 분포도 기반의 모델로 가정하지 않아서 좀 더 유연한 군집화 가능, 이상치의 영향력 크지 않고 군집 개수 정할 필요 없음 but, 수행시간 오래 걸리고 bandwidth 크기에 따른 군집화 영향도가 매우 큼 🡪 컴퓨터 비전 영역에서 사용

**GMM (Gaussian Mixture Model)**

전체 데이터 세트가 여러 개의 가우시안 분포가 섞인 것으로 간주하고 군집화 수행

섞인 데이터 분포에서 개별 유형의 가우시안 분포 추출, 개별 데이터가 그 중 어떤 정규 분포에 속하는지 결정 🡪 **모수추정**

**모수추정**: 개별 정규 분포의 평균과 분산, 각 데이터가 어떤 정규 분포에 해당되는 지의 확률, EM 방법 적용

사이킷런은 EM 방식을 통한 모수 추정 군집화 지원하기 위해 GaussianMixture 지원

* GaussianMixture 객체의 가장 중요한 초기화 파라미터는 n\_components, Gaussian Mixture 모델의 총 개수, k-평균의 n\_clusters 같은 역할

K-평균은 평균 거리 중심으로 중심을 이동하면서 군집화 수행하므로 데이터 세트가 원형의 범위를 가질수록 군집화 효율 더 높아짐

cluster\_std를 작게 설정하면 데이터가 원형 형태로 분산될 수 있음 --> KMeans는 원형의 범위에서 군집화 수행하므로 효과적으로 군집화됨

**GMM은 KMeans보다 유연하게 다양한 데이터 세트에 잘 적용될 수 있지만 수행시간이 오래걸림**

**DBSCAN (Density Based Spatial Clustering of Applications with Noise)**

특정 공간 내에 데이터 밀도 차이를 기반 알고리즘으로 하여서 복잡한 기하학적 분포도를 가진 데이터 세트에 대해서도 군집화 잘 수행 가능

* 입실론 주변 영역(epsilon): 개별 데이터를 중심으로 입실론 반경을 가지는 원형의 영역
* 최소 데이터 개수(min points): 개별 데이터의 입실론 주변 영역에 포함되는 타 데이터 개수

입실론 주변 영역 내에 포함되는 최소 데이터 개수를 충족시키는가 아닌가에 따라 데이터 포인터를 다음과 같이 정의

1. 핵심 포인트, Core Point: 주변 영역 내에 최소 데이터 개수 이상의 타 데이터를 가진 경우 해당 포인트
2. 이웃 포인트, Neighbor Point: 주변 영역 내에 위치한 타 데이터
3. 경계 포인트, Border Point: 주변 영역 내에 최소 데이터 개수 이상의 이웃 포인트를 가지고 있지 않지만 핵심 포인트를 이웃 포인트로 가지고 있는 데이터
4. 잡음 포인트, Noise Point: 최소 데이터 개수 이상의 이웃 포인트를 가지고 있지 않으며, 핵심 포인트도 이웃 포인트로 가지고 있지 않는 데이터

핵심 포인트의 이웃 포인트 역시 핵심 포인트일 경우 두 포인트를 연결해 직접 접근 가능 🡪 이 포인트들을 서로 연결하면서 군집화 구성, 점차적으로 군집 영역 확장, 경계 포인트는 군집의 외곽 형성

사이킷런은 DBSCAN 클래스를 통해 지원

* Eps: 입실론 주변 영역의 반경
* Min\_samples: 핵심 포인트가 되기 위해 입실론 주변 영역 내에 포함되어야 할 데이터의 최소 개수, 자신의 데이터를 포함, min points + 1

군집 레이블이 -1인 것은 노이즈에 속하는 군집, DBSCAN을 적용할 때는 특정 군집 개수 강제하지 않는 것이 좋으며 eps, min\_samples 파라미터 조정하여 최적의 군집 찾는 게 중요

🡪 eps 크게하면 반경이 커져 노이즈 데이터 개수 작아짐, min\_samples 크게하면 주어진 반경 내에 더 많은 데이터 포함시켜야 하므로 노이즈 데이터 개수 커짐

**정리**

군집화의 기능적 의미는 숨어있는 새로운 집단 발견하는 것 --> 새로운 군집 내의 데이터 값 분석하고 이해함으로써 이 집단에 새로운 의미 부여 가능

1. K-Means: 거리 기반으로 군집 중심점 이동하면서 군집화 수행

매우 쉽고 직관적이지만 복잡한 구조를 가지는 데이터 세트에 적용하기에는 한계가 있으며 군집 개수 최적화하기 어려움 🡪 평가를 위해 실루엣 계수 이용

1. Mean Shift: K-평균과 유사하지만 거리 중심이 아니라 데이터가 모여 있는 밀도가 가장 높은 쪽으로 군집 중심점 이동하면서 군집화 수행

컴퓨터 비전 영역에서 이미지나 영상 데이터에서 특정 개체 구분하거나 움직임 추적하는 데 뛰어난 역할 수행

1. GMM: 데이터가 여러 가우시안 분포를 섞어서 생성된 모델로 가정해 수행, 전체 데이터 세트에서 서로 다른 정규 분포 형태 추출해 다른 정규 분포 가진 데이터 세트를 각각 군집화

K-Means보다 유연하게 다양한 데이터 세트에 적용할 수 있지만 수행 시간이 오래 걸림

1. DBSCAN: 밀도 기반 군집화의 대표적 알고리즘, 입실론 주변 영역 내에 포함되는 최소 데이터 개수의 충족 여부에 따라 데이터 포인트를 핵심 포인트, 이웃 포인트, 경계 포인트, 잡음 포인트로 구분하고 특정 핵심 포인트에서 직접 접근 가능한 다른 핵심 포인트를 서로 연결하면서 군집화

간단하고 직관적이며 기하학적으로 복잡한 데이터 세트에도 효과적인 군집화 가능