7. 군집화

01. K-평균 알고리즘 이해

K-평균: 군집 중심점(centroid)이라는 특정한 임의의 지점을 선택해 해당 중심에 가장 가까운 포인트들을 선택하는 군집 화기법. 군집화(Clustering)에서 가장 일반적으로 사용되는 알고리즘

군집 중심점: 선택된 포인트의 평균 지점으로 이동하고 이동된 중심점에서 다시 가까운 포인트를 선 택, 다시 중심점을 평균 지점으로 이동하는 프로세스를 반복적으로 수행. 모든 데이터 포인트에 서 더 이상 중심점의 이동이 없을 경우에 반복을 멈추고 해당 중심점에 속하는 데이터 포인트들을 군집 화하는 기법

<K-평균 동작>

1. 먼저 군집화의 기준이 되는 중심을 구성하려는 군집화 개수만큼 임의의 위치에 가져다 놓습니다. 전체 데이터를 2개로 군집화하려면 2개의 중심을 임의의 위치에 가져다 놓는 것입니다.
2. 각 데이터는 가장 가까운 곳에 위치한 중심점에 소속됩니다. 위 그림에서는 A, B 데이터가 같은 중심점에 소속되며, C, E, F 데이터가 같은 중심점에 소속됩니다.
3. 이렇게 소속이 결정되면 군집 중심점을 소속된 데이터의 평균 중심으로 이동합니다. 위 그림에서는 A, B 데이터 포 인트의 평균 위치로 중심점이 이동했고, 다른 중심점 역시 C, E, F 데이터 포인트의 평균 위치로 이동했습니다.
4. 중심점이 이동했기 때문에 각 데이터는 기존에 속한 중심점보다 더 가까운 중심점이 있다면 해당 중심점으로 다시 소속을 변경합니다. 위 그림에서는 C 데이터가 기존의 중심점보다 더 가까운 중심점으로 변경됐습니다.
5. 다시 중심을 소속된 데이터의 평균 중심으로 이동합니다. 위 그림에서는 데이터 C가 중심 소속이 변경되면서 두 개 의 중심이 모두 이동합니다.
6. 중심점을 이동했는데 데이터의 중심점 소속 변경이 없으면 군집화를 종료합니다. 그렇지 않다면 다시 4번 과정을 거쳐서 소속을 변경하고 이 과정을 반복합니다.

<K-평균의 장점>

- 일반적인 군집화에서 가장 많이 활용되는 알고리즘입니다.
- 알고리즘이 쉽고 간결합니다.

<K-평균의 단점>

- 거리 기반 알고리즘으로 속성의 개수가 매우 많을 경우 군집화 정확도가 떨어집니다
- 반복을 수행하는데, 반복 횟수가 많을 경우 수행 시간이 매우 느려집니다.
- 몇 개의 군집(cluster)을 선택해야 할지 가이드하기가 어렵습니다.

사이킷런 KMeans 클래스 소개

사이킷런 패키지는 K-평균을 구현하기 위해 KMeans 클래스를 제공

<중요 파라미터>

- KMeans 초기화 파라미터 중 가장 중요한 파라미터는 n_clusters이며, 이는 군집화할 개수, 즉 군집 중심점의 개수를 의 미합니다.
- init는 초기에 군집 중심점의 좌표를 설정할 방식을 말하며 보통은 임의로 중심을 설정하지 않고 일반적으로 k=eans++ 방식으로 최초 설정합니다.

- max_iter는 최대 반복 횟수이며, 이 횟수 이전에 모든 데이터의 중심점 이동이 없으면 종료합니다.

KMeans 객체 수행: fit(데이터 세트) 또는 fit_transform(데이터 세트) 메서드를 이용해 수행

KMeans 객체의 속성:

- labels_: 각 데이터 포인트가 속한 군집 중심점 레이블
- cluster_centers_ : 각 군집 중심점 좌표(Shape는 [군집 개수, 피처 개수]). 이를 이용하면 군집 중심점 좌표가 어디인지 시각화할 수 있습니다.

군집화 알고리즘 테스트를 위한 데이터 생성

사이킷런은 다양한 유형의 군집화 알고리즘을 테스트해 보기 위한 간단한 데이터 생성기를 제공
-> make_blobs( )와 make classification( ): 하나의 클래스에 여러 개의 군집이 분포될 수 있게 데이터를 생성할 수 있음

< make_blobs() 호출 파라미터 >

- n_samples : 생성할 총 데이터의 개수입니다. 디폴트는 100개입니다.
- n_features : 데이터의 피처 개수입니다. 시각화를 목표로 할 경우 2개로 설정해 보통 첫 번째 피처는 x 좌표, 두 번째 피처는 y 좌표상에 표현합니다.
- centers : int 값 예를 들어 3으로 설정하면 군집의 개수를 나타냅니다. 그렇지 않고 ndarray 형태로 표현할 경우 개별 군집 중심점의 좌표를 의미합니다.
- cluster_std : 생성될 군집 데이터의 표준 편차를 의미합니다. 만일 float 값 0.8과 같은 형태로 지정하면 군집 내에서 데이터가 표준편차 0.8을 가진 값으로 만들어집니다. [0.8,1,2, 0.6]과 같은 형태로 표현되면 3개의 군집에서 첫 번째 군집 내 데이터의 표준편차는 0.8. 두 번째 군집 내 데이터의 표준 편차는 1.2, 세 번째 군집 내 데이터의 표준편차는 0.6으로 만듭니다. 군집별로 서로 다른 표준 편차를 가진 데이터 세트를 만들 때 사용합니다.

02. 군집 평가(Cluster Evaluation)

실루엣 분석의 개요

실루엣 분석(silhouette analysis): 군집화 평가 방법. 실루엣 분석은 각 군집 간의 거리가 얼마나 효율적으로 분리돼 있는지를 나타냄.
-> 효율적으로 잘 분리됐다는 것은 다른 군집과의 거리는 떨어져 있고 동일 군집끼리의 데이터는 서로 가깝게 잘 뭉쳐 있다는 의미
-> 실루엣 계수를 기반으로 함

<실루엣 계수>

- 실루엣 계수: 개별 데이터가 가지는 실루엣 계수는 해당 데이터가 같은 군집 내의 데이터와 얼마나 가깝게 군집화돼 있고, 다른 군집에 있는 데이터와는 얼마나 멀리 분리돼 있는지를 나타내는 지표.

- i번째 데이터 포인트의 실루엣 계수 값 s(i)
  = (b(i)-a(i)) / (max(a(i),b(i))

- -1에서 1 사이의 값을 가지며, 1로 가까워질수록 근처의 군집과 더 멀리 떨어져 있다는 것이고 0에 가까울수록 근처의 군집과 가까워진다는 것입니다. - 값은 아예 다른 군집에 데이터 포인트가 할당됐음을 뜻함

< 좋은 군집화의 기준 조건>

1. 전체 실루엣 계수의 평균값, 즉 사이킷런의 silhouette_score( ) 값은 0 ~ 1 사이의 값을 가지며, 1에 가까울수록 좋습니다.
2. 하지만 전체 실루엣 계수의 평균값과 더불어 개별 군집의 평균값의 편차가 크지 않아야 합니다. 즉, 개별 군집의 실루엣 계수 평균값이 전체 실루엣 계수의 평균값에서 크게 벗어나지 않는 것이 중요합니다. 만약 전체 실루엣 계수의 평균값은 높지만, 특정 군집의 실루엣 계수 평균값만 유난히 높고 다른 군집들의 실루엣 계수 평균값은 낮으면 좋은 군집화 조건이 아닙니다.

군집별 평균 실루엣 계수의 시각화를 통한 군집 개수 최적화 방법

http://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.htm

<평균 실루엣 계수로 군집 개수를 최적화하는 방법>

i) 주어진 데이터에 대해서 군집의 개수 2개를 정했을 때

평균 실루엣 계수, 즉 silhouette_score는 약 0.704로 매우 높음
1번 군집의 모든 데이터는 평균 실루엣 계수 값 이상이지만, 2번 군집의 경우는 평균보다 적은 데이터 값이 매우 많음

ii) 군집 개수가 3개
전체 데이터의 평균 실루엣 계수 값은 약 0.588
1번, 2번 군집의 경우 평균보다 높은 실루엣 계수 값을 가지고 있지만, 0번의 경우 모두 평균보다 낮음

iii) 군집 개수가 4개
평균 실루엣 계수 값은 약 0.65
개별 군집의 평균 실루엣 계수 값이 비교적 균일하게 위치

-> 군집이 2개인 경우보다는 평균 실루엣 계수 값이 작지만 4개인 경우가 가장 이상적인 군집화 개수로 판단

03 평균이동

평균 이동(Mean Shift)의 개요

평균 이동(Mean Shift): K-평균과 유사하게 중심을 군집의 중심으로 지속적으로 움직이면서 군집화를 수행. *K-평균이 중심에 소속된 데이터의 평균 거리 중심으로 이동하는 데 반해, 평균 이동은 중심을 데이터가 모여 있는 밀도가 가장 높은 곳으로 이동*

평균 이동 군집화: 데이터의 분포도를 이용해 군집 중심점을 찾음. 특정 데이터를 반경 내의 데이터 분포 확률 밀도가 가장 높은 곳으로 이동하기 위해 주변 데이터와의 거리 값을 KDE 함수 값으로 입력한 뒤 그 반환 값을 현재 위치에서 업데이트하면서 이동하는 방식

군집 중심점: 데이터 포인트가 모여있는 곳이라는 생각에서 착안한 것이며 이를 위해 확률 밀도 함수(probability density function)를 이용합니다.
가장 집중적으로 데이터가 모여있어 확률 밀도 함수가 피크인 점을 군집 중심점으로 선정하며 일반적으로 주어진 모델의 확률 밀도 함수를 찾기 위해서 **KDE(Kernel Density Estimation)** 이용합니다.

KDE(Kernel Density Estimation): 커널(Kern이) 함수(대표적인 커널 함수: 가우시안 분포 함수)를 통해 어떤 변수의 확률 밀도 함수를 추정하는 대표적인 방법. 관측된 데이터 각각에 커널 함수를 적용한 값을 모두 더한 뒤 데이터 건수로 나눠 확률 밀도 함수를 추정

KDE = 1/nh * Σ_(i=1 to n) {K((x-xi) / h)}

-> h: 대역폭 오는 KDE 형태를 부드러운(또는 뾰족한) 형태로 평활화(Smoothing)하는 데 적용되며, 이 h를 어떻게 설정하느냐에 따라 확률 밀도 추정 성능을 크게 좌우. 대역폭이 클수록 평활화된 KDE로 인해 적은 수의 군집 중심점을 가지며 대역폭이 적을수록 많은 수의 군집 중심점을 가짐

사이킷런은 최적의 대역폭 계산을 위해 estimate_bandwidth() 함수를 제공

04.GMM(Gaussian Mixture Model)

GMM(Gaussian Mixture Model) 소개

GMM 군집화: 군집화를 적용하고자 하는 데이터가 여러 개의 가우시안 분포(GaussianDistri bution)를 가진 데이터 집합들이 섞여서 생성된 것이라는 가정하에 군집화를 수행하는 방식

정규 분포: 평균 mu를 중심으로 높은 데이터 분포도를 가지고 있으며, 좌우 표준편차 1 에 전체 데이터의 68.27%, 좌우 표준편차 2에 전체 데이터의 95.45%를 가지고 있습니다. 평균이 0이고, 표준편차가 1인 정규 분포를 표준 정규 분포

GMM에서의 *모수 추정*: 데이터 세트를 구성하는 여러 개의 정규 분포 곡선을 추출하고, 개별 데이터가 이 중 어떤 정규 분포에 속하는지 결정하는 방식.

- 개별정규 분포의평균과 분산
- 각 데이터가 어떤 정규 분포에 해당되는지의 확률
  추정

모수 추정을 위해 GMM은 EM(Expectation and Maximization) 방법을 적용

GMM과 K—평균의 비교

<KMeans 군집화>

- 원형의 범위에서 군집화를 수행. 데이터 세트가 원형의 범위를 가질수록 KMeans의 군집화 효율은 더욱 높아짐

<GMM 군집화>

- 데이터가 분포된 방향에 따라 정확하게 군집화됨
- 이와 달리 KMM은 길쭉한 방향으로 데이터가 밀접해 있을 경우 최적의 군집화 어려움

05.DBSCAN

DBSCAN 개요

DBSCAN(Density Based Spatial Clustering of Applications with Noise): 밀도 기반 군집화의 대표적인 알고리즘

- 간단하고 직관적인 알고리즘으로 돼있음에도 데이터의 분포가 기하학적으로 복잡한 데이터 세트에도 효과적인 군집화가 가능

- 내부의 원 모양과 외부의 원 모양 형태의 분포를 가진 데이터 세트를 군집화한다고 가정할 때 K 평균, 평균 이동, GMM으로는 효과적인 군집화를 수행하기가 어려움

- 파라미터 - 입실론 주변 영역(epsilon): 개별 데이터를 중심으로 입실론 반경을 가지는 원형의 영역입니다

- 파라미터 - 최소 데이터 개수(min points): 개별 데이터의 입실론 주변 영역에 포함되는 타 데이터의 개수입니다.

- 데이터 포인트 정의:
  ㄴ핵심 포인트(Core Point): 주변 영역 내에 최소 데이터 개수 이상의 타 데이터를 가지고 있을 경우 해당 데이터를 핵심 포 인트라고 합니다.
  ㄴ 이웃 포인트(Neighbor Point) : 주변 영역 내에 위치한 타 데이터를 이웃 포인트라고 합니다. ㄴ 경계 포인트(Border Point) : 주변 영역 내에 최소 데이터 개수 이상의 이웃 포인트를 가지고 있지 않지만 핵심 포인트를 이웃 포인트로 가지고 있는 데이터를 경계 포인트라고 합니다.
  ㄴ 잡음 포인트(Noise Point) : 최소 데이터 개수 이상의 이웃 포인트를 가지고 있지 않으며, 핵심 포인트도 이웃 포인트로 가지고 있지 않는 데이터를 잡음 포인트라고 합니다.

< 12개 데이터 세트에 대해서 DBSCAN 군집화 적용>

1. 특정 입실론 반경 내에 포함될 최소 데이터 세트를 6개로(자기 자신의 데이터를 포함) 가정하겠습니다.
2. P1 데이터를 기준으로 입실론 반경 내에 포함된 데이터가 7개 (자신은 P1, 이웃 데이터 P2, P6, P7, P8, P9, P11) 로 최소 데이터 5개 이상을 만족하므로 P1 데이터는 핵심 포인트 (Core Point) 입니다.
3. 다음으로 P2 데이터 포인트를 살펴보겠습니다. P2 역시 반경 내에 6개의 데이터 (자신은 P2, 이웃 데이터 P1, P3, P4, P9, P10) 를 가지고 있으므로 핵심 포인트입니다.
4. 핵심 포인트 P1 의 이웃 데이터 포인트 P2 역시 핵심 포인트일 경우 P1 에서 P2로 연결해 직접 접근이 가능합니다.
5. 특정 핵심 포인트에서 직접 접근이 가능한 다른 핵심 포인트를 서로 연결하면서 군집화를 구성합니다. 이러한 방식으로 점차적으로 군집(Cluster) 영역을 확장해 나가는 것이 DBSCAN 군집화 방식입니다
6. P3 데이터의 경우 반경 내에 포함되는 이웃 데이터는 P2, P4로 2개이므로 군집으로 구분할 수 있는 핵심 포인트가 될 수 없습니다. 하지만 이웃 데이터 중에 핵심 포인트인 P2를 가지고 있습니다. 이처럼 자신은 핵심 포인트가 아니지만, 이웃 데이터로 핵심 포인트를 가지고 있는 데이터를 경계 포인트(Border Point)라고 합니다. 경계 포인트는 군집의 외곽을 형성합니다.
7. 다음 그림의 P5와 같이 반경 내에 최소 데이터를 가지고 있지도 않고, 핵심 포인트 또한 이웃 데이터로 가지고 있지 않는 데이터를 잡음 포인트(Noise Point)라고 합니다.

사이킷런은 DBSCAN 클래스를 통해 DBSCAN 알고리즘을 지원 주요 초기화 파라미터

- eps: 입실론 주변 영역의 반경을 의미합니다.
- min_samples: 핵심 포인트가 되기 위해 입실론 주변 영역 내에 포함돼야 할 데이터의 최소 개수를 의미합니다 (자신의 데 이터를 포함합니다. 위에서 설명한 min points + 1)

DBSCAN 적용하기 - make_circles() 데이터 세트

복잡한 기하학적 분포를 가지는 데이터 세트에서 DBSCAN과 타 알고리즘을 비교
-> 거리 기반 군집화로는 위와 같이 데이터가 특정한 형태로 지속해서 이어지는 부분을 찾아내기 어려움
GMM도 내부와 외부의 원형으로 구성된 더 복잡한 형태의 데이터 세트에서는 군집화가 원하는 방향으로 되지 않음
DBSCAN으로 군집화를 적용해 원하는 방향으로 정확히 군집화가 됨

06. 군집화 실습 - 고객 세그먼테이션

고객 세그먼테이션의 정의와 기법

고객 세그먼테이션(Customer Segmentation): 다양한 기준으로 고객을 분류하는 기법
-> 주요 목표 : 타깃 마케팅(고객을 여러 특성에 맞게 세분화해서 그 유형에 따라 맞춤형 마케팅이나 서비스를

제공하는 것)

-> 고객의 어떤 요소를 기반으로 군집화할 것인가를 결정하는 것이 중요
실습에서는 기본적인 고객 분석 요소인 *RFM 기법*(RECENCY (R) : 가장 최근 상품 구입 일에서 오늘까지의 기간, FREQUENCY (F) : 상품 구매 횟수, MONETARY VALUE(M) : 총 구매금액)을 이용