* **Mean Shift 군집화**

**KDE**(Kernel Density Esitimation)을 이용하여 데이터 포인트들이 데이터 분포가 높은 곳으로 이동하며 군집화 수행

Mean Shift는 **데이터 분포도에 기반**하여 **군집화 개수를 정함**

**KDE(Kernel Density Esitimation)** :

커널(kernel)함수를 통해 어떤 변수의 **확률밀도 함수를 추정**하는 방식.

관측된 데이터 각각에 커널 함수를 적용한 값을 모두 더한 뒤 데이터 건수로 나누어서 확률 밀도 함수를 추정함

**확률밀도함수 (PDF:Probability Density Function)** : 확률 변수의 분포를 나타내는 함수

ex) 정규 분포, 감마 분포, t-분포 등

* 확률밀도함수를 알게 되면 **특정 변수가 어떤 값을 갖게 될지의 확률**을 알게 됨.

즉, **확률밀도 함수를 통해 변수의 특성** (예를 들어 정규 분포의 경우 평균, 분산), 확률 분포 등 변수의 많은 요소를 알 수 있게 됨

* **확률 밀도 추정 방법**

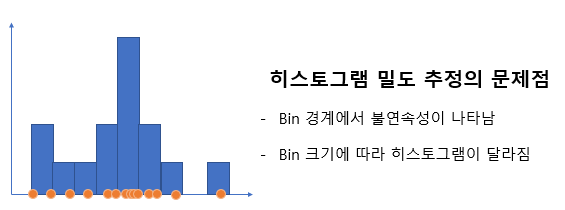
**모수적(Parametric) 추정** : 데이터가 특정 **데이터분포(ex:가우시안 분포)를 따른다는 가정하**에

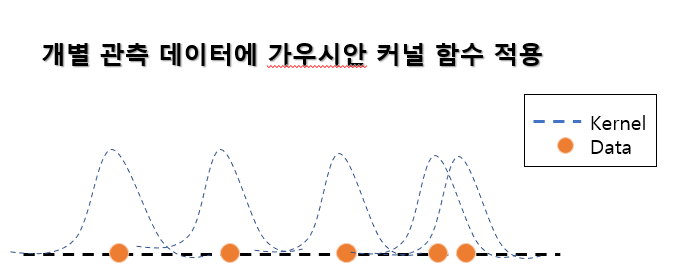
데이터 분포를 찾는 방법 Gaussian Mixture 등..

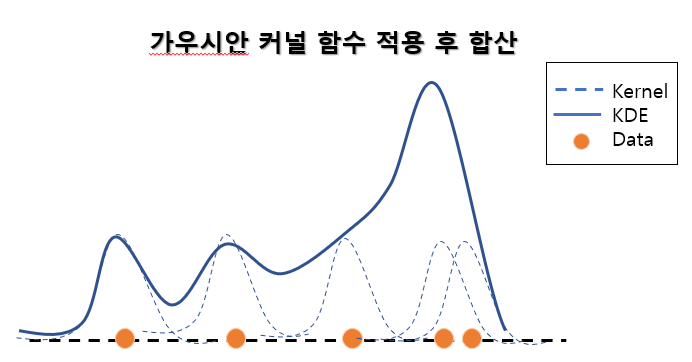
**비모수적(Non-Parametric) 추정** : 데이터가 **특정 분포를 따르지 않는다는 가정 하**에 밀도를 추정

**관측된 데이터만**으로 확률 밀도를 찾는 방법 ex ) KDE

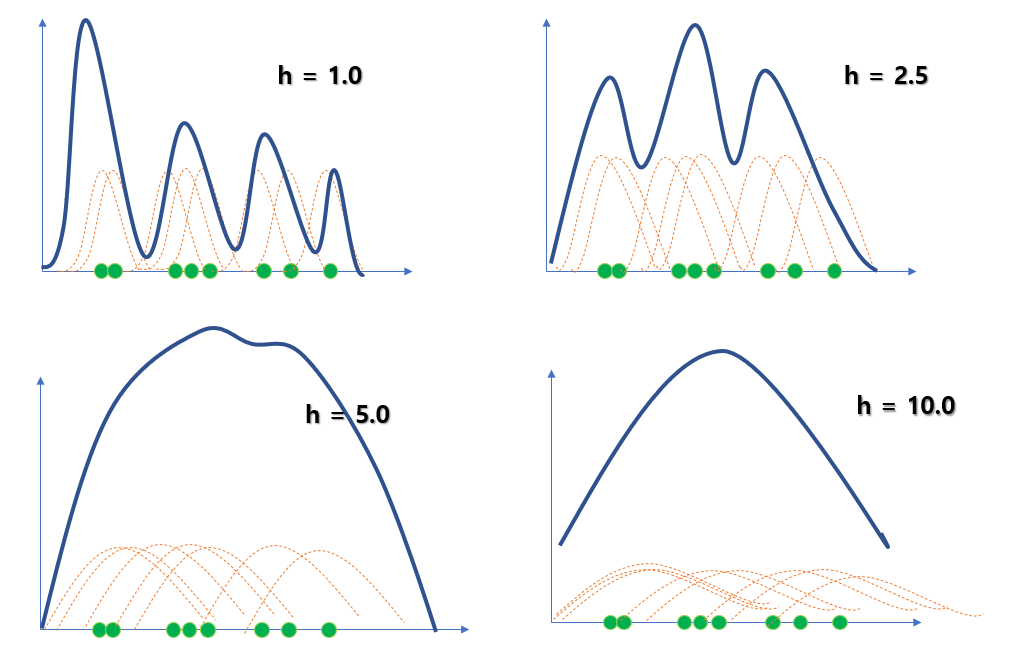
* **비모수적 밀도 추정 – 히스토그램(Histogram)**





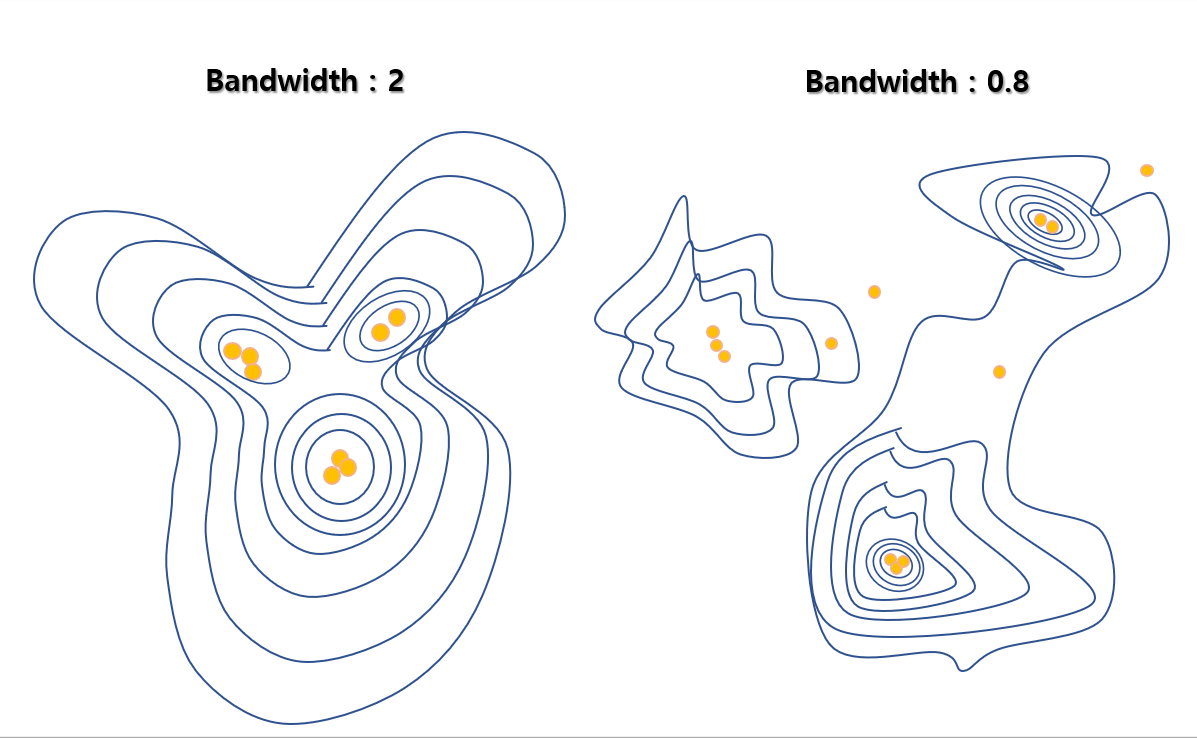


* **Bandwidth에 따른 KDE의 변화**



**작은 h값**은 좁고 Spike한 KDE로 **변동성이 큰 확률밀도함수**를 추정 (오버피팅)

**큰 h값**은 과도하게 Smoothing 된 KDE로 **단순화된 학률밀도함수**를 추정 (언더피팅)



MeanShift는 **Bandwidth가 클수록 적은 수의 클러스터링 중심점**을, **Bandwidth가 작을수록 많은 수의 클러스터링 중심점**을 가지게 됨

또한 MeanShift는 군집의 개수를 정하지 않으며, **오직 Bandwidth의 크기에 따라 군집화 수행**

* **사이킷런 Mean Shift**

MeanShift 클래스의 **가장 중요한 초기화 파라미터**는 **bandwidth**

**:** 밀도중심으로 이동 할 때 사용되는 커널 함수의 bandwidth 🡪 이에 따라 군집화 성능이 달라짐.

최적의 bandwidth 계산을 위해 **estimate\_bandwidth() 함수** 이용