밀도추정 (Density Estimation)

: 현실세계에 존재하는 수 많은 대상 중 [대상의 어떤 특정 부분을 관찰]하면 [관련된 관측값]를 얻을 수 있다.

> 데이터분석에선
> 대상 그 자체는 전체 데이터셋(DB,Dataset)으로 표현되며,
> 대상이 갖는 특징은 변수(Variable)/속성(Atrribute)/특성(Features)/칼럼(Column)이라 말하고,
> 관련 관측값은 데이터값(Value)/개체(instance)/레코드(Record)/로우(Row)/데이터포인트(Datapoint)라고 말한다.

우리는 다양하게 관측될 수 있는 변수(혹은 칼럼)이 가지고 있는 본질적인 특성에 대해서 이야기하고자 한다.

결론부터 말하면,
관측된 데이터 값들의 분포를 분석하여 그 변수에 대한 추정을 하는 것이 밀도추정(Density Estimation)이다.
: 변수는 특정 범위 안에서 무한한 다양한 데이터를 갖을 수 있다. 따라서 우리가 현재 측정한 데이터는 그 변수가 갖고 있는 일면에 불과하다.

하지만 일면에 불과한 데이터라도 자주 관측되는 것이 있고 아닌 것이 있듯이,
변수에 대한 랜덤추출의 경우 관측확률이 존재한다(예, 평균을 중심으로 퍼져나가는 확률분포)
따라서 이러한 관측확률에 대한 빈도 분석을 통해,
변수가 전체적으로 어떤 확률적인 특성을 나타내는 지 파악하여
그 확률적 특성이 결국 변수의 특성이라고 추론하는 방법을 밀도추정이라는 하는 것이다.
*통계에서 밀도란 확률분포곡선에서 구간에 대한 확률적분값을 말한다.
따라서 밀도추정은 관측값을 통해 밀도를 구하여 평균을 추정한다. 라고 생각해도 될 듯 하다.

예를 들어보자. 어떤 육교 밑을 통과하는 차량의 일일 교통량을 변수로 설정해 두었다. 이 변수의 특징은 어떻게 설명할 수 있을까?
관측하는 방법밖에 없다. 어떤 날은 400대가 지나가고 어떤 날은 300대가 지나가고, 어떤 날은 이상하게 1000대가 지나갈 수도 있다. 따라서 하루 이틀의 관측 결과만 가지고서는 이 육교의 일일 교통량이 무어라고 결론을 내리기는 힘들다.
그러나 2년을 관측했다면?
평균은 524대이고 그 분포는 어떻게 되더라라는 변수에 대한 확률적인 분포를 그릴 수 있을 것이다. 이처럼 변수에 대한 분포를 추정하여 밝히는 일련의 모든 과정을 '밀도추정'이라고 하는 것이다.

1. Parmetric Estimation(모수추정)

: pdf(probabilty density function)와 같은 정형화된 확률밀도곡선과 우리가 알고자하는 변수의 확률분포가 같다고 가정하고 평균과 분산을 통해서 그 의미를 파악하는 방법이다.
예를 들면 일일교통량이 정규분포를 따른다고 가정하면, 우리는 정규분포의 특징을 그대로 적용시킬 수 있다.

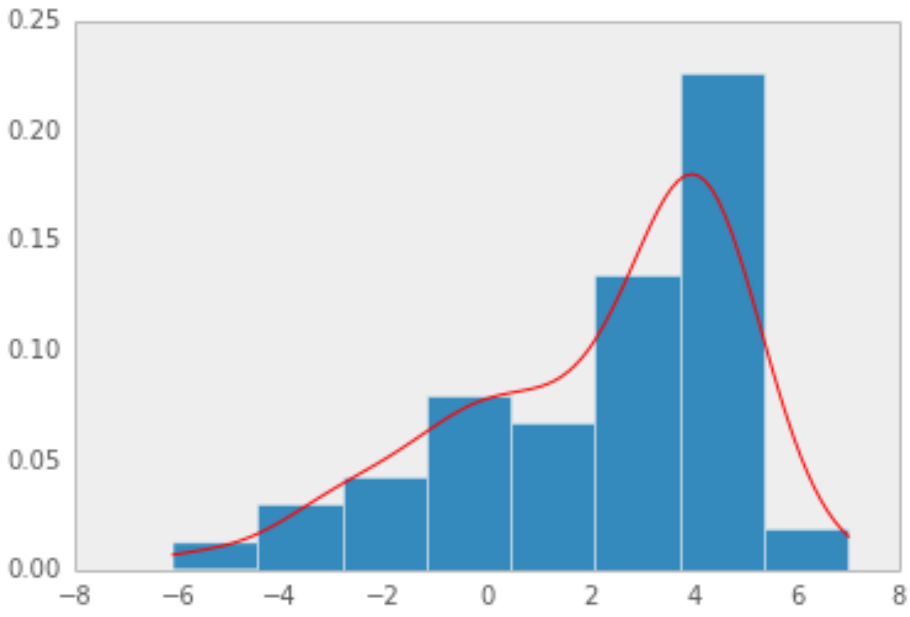
2. Non-Parmetric Estimation(비모수 추정)

비모수 추정은 변수가 이미 알고 있는 임의의 확률밀도곡선을 따른다고 가정할 수 없을 때 사용하는 방법으로
크게 히스토그램방식, 커널방식으로 나눠 볼 수 있다.

1) 히스토그램

변수가 이미 알고 있는 임의의 확률밀도곡선을 따른다고 가정할 수 없을 때, 가장 쉽게 확률밀도를 알 수 있는 방법은 히스토그램을 통해 파악해보는 것이다.
즉, 관측된 데이틀로부터 히스토그램을 구한 뒤, 그 히스토그램을 "정규화하여" 확률 밀도 곡선으로 사용하는 것이다.

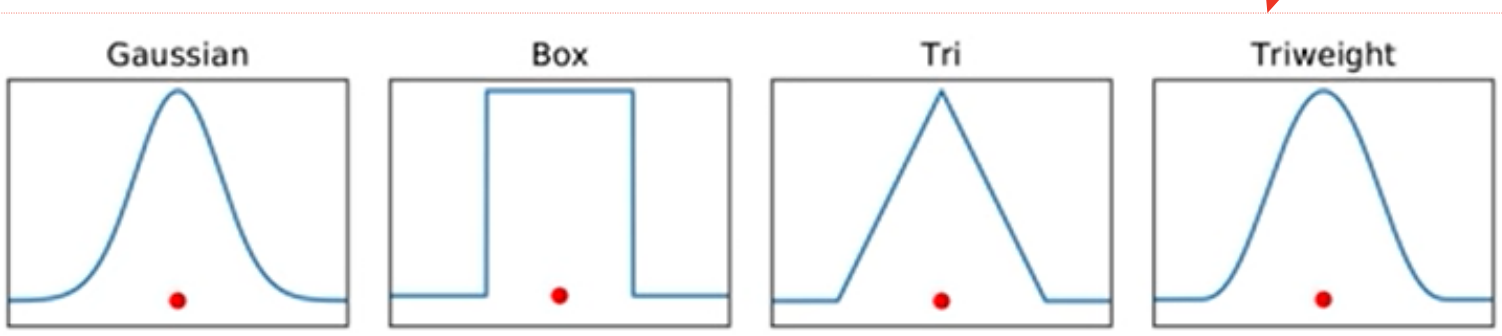
상대도수를 활용하여



2) Kernel Density Estimation (커널 밀도 추정)

앞서 비모수추정(Non-permetric Estimation)의 가장 단순한 형태가 히스토그램을 이용하는 것이라고 했다.
마찬가지로 커널밀도추정 또한 주어진 데이터에서 알 수 없는 확률을 추정하는 방법이다.
기본적인 방식은 히스토그램 방식과 비슷하다.
그러나 히스토그램 방법은 bin(구간)의 경계에서 불연속성이 나타난다는 점, bin의 크기 및 시작 위치에 따라서 히스토그램이 달라진다는 점, 고차원(high dimension) 데이터에는 메모리 문제 등으로 사용하기 힘들다는 점 등의 문제점을 갖는다.

Kernel Density Estimation (커널 밀도 추정) 방법은 non-parametric 밀도추정 방법 중 하나로서 커널함수(kernel function)를 이용하여 이러한 히스토그램 방법의 문제점을 개선한 방법이다.
히스토그램이 구간별로 네모난 박스형태를 그렸다싶이
각 데이터마다 그 데이터를 중심으로하는 특정모양의 확률곡선을 그린다고 생각하면 이해하기 쉽다.



대표적인 커널함수 종류

히스토그램의 박스 대신 그려내는 방법은 다음과 같다.
X를 변수(random variable), ← 벡터
x1, x2, ..., xn을 관측된 샘플 데이터, ← 스칼라.
K를 커널 함수라 하자.
이 때 KDE에서는 랜덤 변수 x에 대한 pdf(확률밀도함수)를 다음과 같이 추정한다.

$$\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

--- (4)

식 (4)에서 h는 커널(kernel) 함수의 bandwidth 파라미터로서 커널이 뽀족한 형태(h가 작은 값)인지 완만한 형태(h가 큰 값)인지를 조절하는 파라미터이다. 수식적으로 보면 어렵지만 이를 직관적으로 이해하면 다음과 같다.

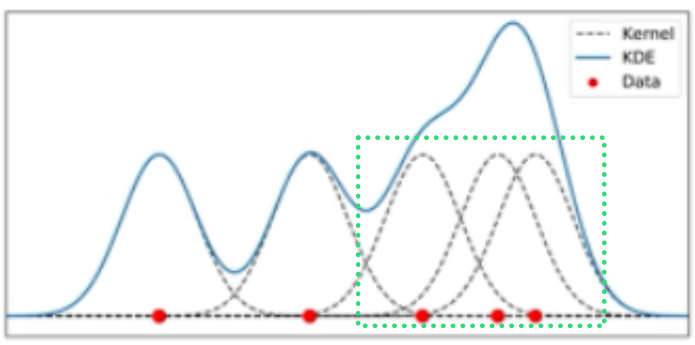
1. 관측된 데이터 각각마다 해당 데이터 값을 중심으로 하는 커널 함수를 생성한다: K(x-xi)
2. 이렇게 만들어진 커널 함수들을 모두 더한다.
*cf) Kernel 이 겹치는 구간 (데이터가 밀집된 구간) 에는 KDE 함수의 크기가 커지게 된다.

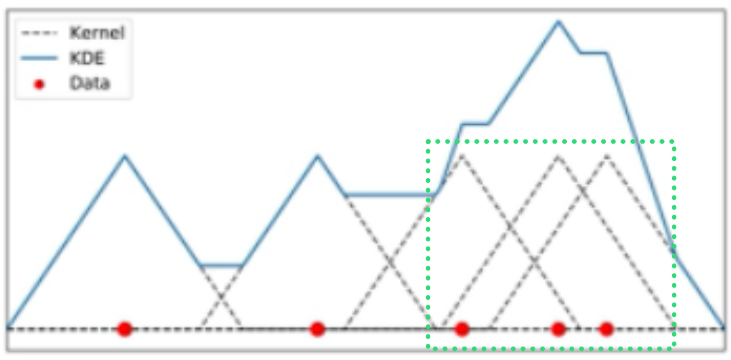
On every data point x_i, we place a kernel function K. The kernel density estimate is

$$\hat{f}(x) = \frac{1}{N}\sum_{i=1}^{N} K(x - x_i).$$

The *triangular* kernel (or *linear* kernel) is given by

$$f(x) \propto \max(1 - |x|, 0).$$





데이터 포인터별로 커널함수를 그리고 합한 모양 왼)가우시안커널 이용 오)삼각형커널 이용

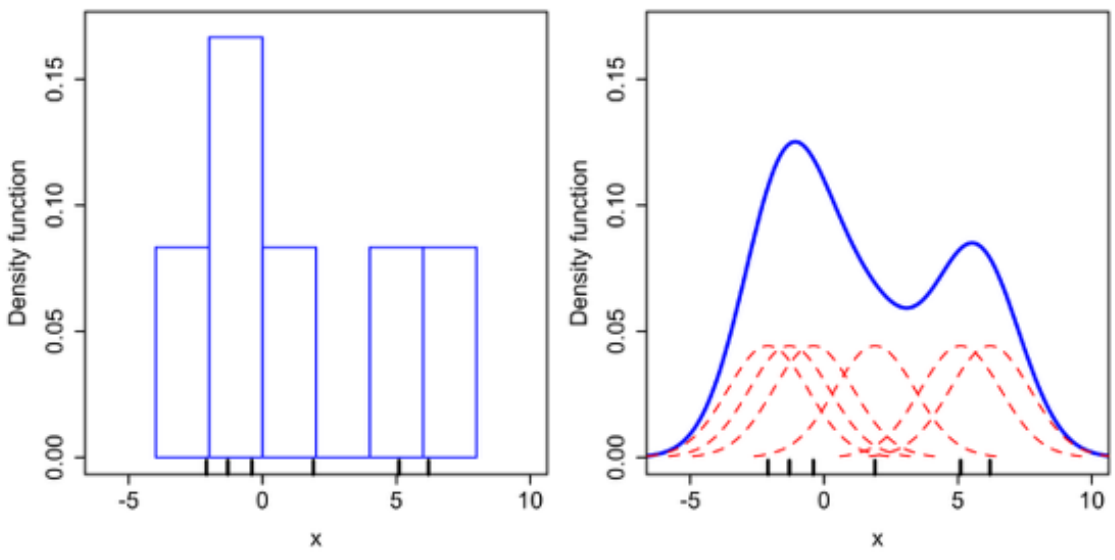
3. 전체 데이터 개수로 나눈다.
: 여기서 전체 데이터 개수로 나누는 이유는 커널함수의 특징 때문이다.
각 확률밀도 곡선이 1이 되는 값을 n개 더해주니 더해진 확률밀도값의 확률총합이 5가 되는 현상이 발생하는데, 확률밀도곡선의 적분값은 max가 1 (100%) 이므로 더한 갯수만큼 나눠줘야 하는 것이다.

*수학적으로 커널함수는
원점을 중심으로 대칭이면서 적분값이 1인 non-negative 함수로 정의되며
이것을 수식으로 표현아면 아래와 같다.

$$\int_{-\infty}^{\infty} K(u)du = 1$$

$$K(u) = K(-u), \quad K(u) \geq 0, \quad \forall u$$

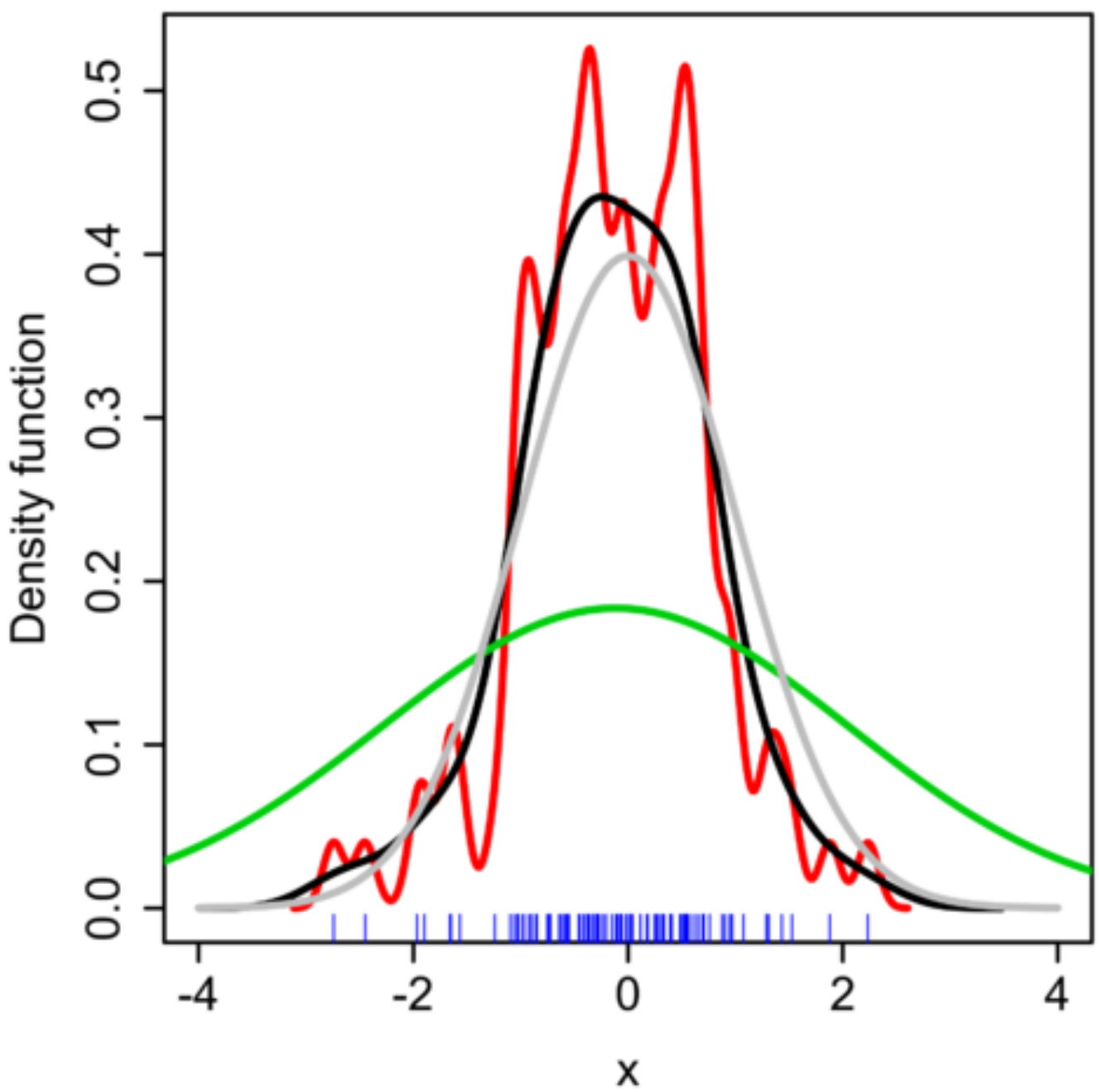
히스토그램을 이용한 밀도추정 방법과 KDE 방법을 비교해 보면,



<그림 4> "Comparison of 1D histogram and KDE" by Drleft

히스토그램 방법은 이산적(discrete)으로 각 데이터에 대응되는 bin의 값을 증가시킴으로써 불연속성이 발생하는 반면 KDE(커널밀도추정) 방법은 각 데이터를 커널 함수로 대치하여 더함으로써
그림 4 오른쪽 그래프와 같이 smooth한 확률밀도함수(pdf)를 얻을 수 있는 장점을 갖는다.

<그림 5> Kernel density estimate (KDE) with different bandwidths of a random sample of 100 points from a standard normal distribution. Grey: true density (standard normal). Red: KDE with h=0.05. Black: KDE with h=0.337. Green: KDE with h=2 (출처: 위키피디아)



실제 KDE를 사용할 때, 중요한 이슈는 어떤 커널 함수를 사용할지와 커널 함수의 bandwidth 파라미터인 h 값을 어떻게 잡을지이다. 위키피디아에 의하면 가장 최적의 커널함수는 Epanechnikov 커널이나,
계산의 편의상 Gaussian 커널함수도 많이 사용된다고 한다.
그리고 Gaussian 커널함수를 사용할 경우 최적의 bandwidth 파라미터 값은 다음과 같다고 한다.

$$h=\left(\frac{4\sigma^5}{3n}\right)^{\frac{1}{5}} \approx 1.06\sigma n^{-1/5}$$

Gaussian 커널함수 최적파라미터 h 단, n은 샘플 데이터의 개수, σ는 샘플 데이터의 표준편차.