K-nearest-neighbor

1. K-nearest neighbor 방법은 학습 과정이 존재하지 않는다.
   출력은 그냥 입력값과 훈련 데이터를 재서 가장 가까운 것을 출력한다.
   거리가 동일하다면 휴리스틱을 이용하여 출력할 값을 정함
   얼굴 인식, 음악 찾기 등에서 사용됨

2. 거리를 잰다고 했는데 어떻게?

   1) $L_2 Norm = \|\vec{x} - \vec{y}\|_2^2$

   2) $L_1 Norm = |\vec{x} - \vec{y}|$

   3) $Max\ Norm = \|\vec{x} - \vec{y}\|_\infty^2$

   4) $Scaled\ L_2\ Norm = \sigma(\|\vec{x} - \vec{y}\|_2^2)$
      규모가 다른 입력 데이터는 학습에 대한 영향력이 다르다
      → 정규화를 통해 이를 해결

   5) $Mahalanobis\ Distance = \sqrt{(x-y)^T A (x-y)}$

   6) Hamming Distance
      다른 거만 골라냄
      ex) 10100 vs 11010
      → D = 3

   7) Histogram Intersection
      히스토그램의 교집합
      $1 - \Sigma \min(x, y)$

   8) Chi Squared Histogram Distance
      $\frac{1}{2} \Sigma \frac{[x-y]^2}{x+y}$
      두 히스토그램의 유사도를 측정

   9) Earth Movers Distance
      $\min_f \sum_i^m \sum_j^n f_{ij} \cdot c_{ij}$
      두 히스토그램의 모양의 유사도
      모양은 비슷한데 이동한 것이어도
      유사도 높게 측정

⟹ 거리보단 형태가 더 비슷한 게 중요하다.

3. 얼마나 많은 데이터가 있어야 하는가?

하나의 데이터만 이웃해 있다면 아웃라이어의 경우
혹은 경계에 위치한 데이터의 경우 제대로 분류 불가

⇒ 5-NN Classification
가장 가까운 5개의 데이터로 판단

Kernal Regression
NN Regression과 비슷하나 이웃 데이터에 가중치를 부여하여 예측을 수행한다.
이때 Kernal Function을 사용한다.

4. 단점
1) 효율↓
굉장히 느리다.
2) 잡음이 많으면 잘 작동하지 않는다.
3) 차원의 저주 : 차원이 높아질수록 데이터가 희소해진다.
→ 거리가 의미가 없어진다.

5. 해결법
1) 차원의 저주? : 차원 축소, 무작위 사영
2) 잡음? : 아웃라이어 판별, Learnable 특징