-Model-based Learning (모델 만든 후 learning 하는 것)
  -선형/비선형 모델
  -Neural network
  -의사 결정 나무
  -Support vector machine
=> 데이터로부터 모델을 생성하여 분류/예측 진행
-예측 과정)
  traning data가 주어졌을 때 분류/예측 모델 구축 -> 새로운 데이터를 분류 모델에 넣음 -> 새로운 데이터의 상태 분류/예측

-Instance-based Learning (모델 생성 X)
  -K-nearest neighbor
  -Locally weighted regression
=> 별도의 모델 생성 없이 인접 데이터를 분류/예측에 사용
-예측 과정)
  새로운데이터 -> traning data 패턴보고 -> 데이터의 상태 분류/예측

-knn 분류 알고리즘
  -Instance-based Learning
  -Memory-based Learning
  -Lazy Learning :모델을 별도로 학습하지 않고 테스팅 데이터가 들어와야 비로소 작동함
예측 과정)
  -분류할 관측치 x선택
  -x로부터 인접한 k개의 학습 데이터 탐색
  -탐색된 k개 학습 데이터의 majority class c를 정의
  -c를 x의 분류결과로 반환

-knn 예측 알고리즘
예측 과정)
  -예측할 관측치 x를 선택
  -x로부터 인접한 k개의 학습 데이터 탐색
  -탐색된 k개 학습 데이터의 평균을 x의 예측값으로 반환

-knn 하이퍼파라미터 (유저가 결정해야하는 것 - 하이퍼파라미터)
  -k (몇 개의 neighbor 사용할거임?) (1<=k<=전체데이터 수)
    -작을 경우 데이터의 지역적 특성을 지나치게 반영함(overfitting)
    -클 경우 다른 범주의 개체를 너무 많이 포함하여 오분류할 위험이 커짐 (underfitting)
  -distance measures (데이터 간 distance 어떻게 측정할거임?)

k의 선택 방법

-일정 범위 내로 k를 조정하여 가장 좋은 예측 결과를 보이는 k 값 선정
-k가 작아지면 training data의 오류는 적어짐, but test error는 작아지다가 커짐

거리측도
-다양한 거리측도 존재
-데이터 내 변수들이 각각 다른 범위, 분산 가질 수 있으므로 데이터 표준화를 시키고 해야함
-Euclidean Distance : x,y값 간 차이 제곱합의 제곱근, 두 관측치 사이 직선거리 의미(최단거리)
-Manhattan Distance : 격자거리, 차이의 절댓값의 합
-Mahalanobis Distance : 유클리드에 공분산의 역행렬 곱하기, 변수 내 분산, 변수 간 공분산을 모두 반영하여 x,y 간 거리를 계산하는 방식, 데이터의 공분산 행렬이 단위행렬인 경우는 유클리드 거리랑 동일, 마한론비스 거리의 제곱은 타원이 됨
-Correlation Distance : -1<=c<=1, 0<=코릴레이션 거리<=2, signal data 사이의 유사성 판별할 때 사용, 전반적인 패턴의 차이를 보고 싶을 때 사용
-Spearman Rank Correlation Distance : -1<=로우<=1

knn 장점
-데이터 내 노이즈에 영향을 크게 받지 않음, 마한론비스 거리와 같이데이터의 분산을 고려할 경우 강건함
-학습 데이터 수가 많을 경우 효과적
knn 고려해야할 점
-하이퍼파라미터 값을 선정해야함
-어떤 거리 척도가 분석에 적합한지 불분명함
-새로운 관측치와 각각의 학습 데이터 간 거리를 전부 측정해야함 -> 계산 시간 오래걸림
-고차원의 데이터에서는 knn이 잘 작동하지 않음

가중치)
예측모델 : 거리 제곱의 반비례