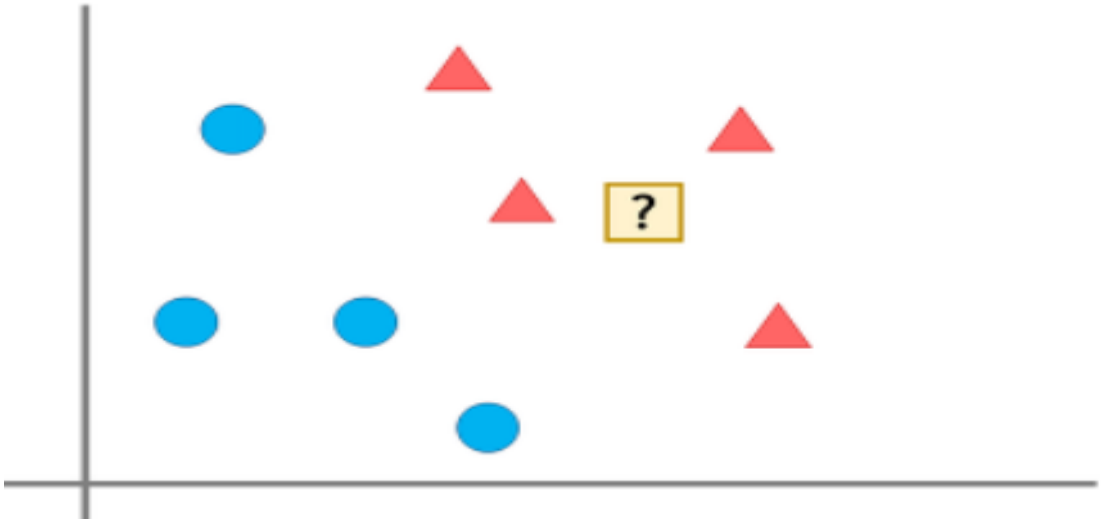
KNN (K - Nearest Neighbors)

어떤 한 record를 분류하기 위해, 주변(Near)의 record를 이용하는 분류기법(지도학습).

NN

=>한 record에서 가장 가까운 record가 속한 class 로 분류한다.



과일과 야채 분류 예시, y축 당도, x축 아삭함

"?" 가 속할 class는?

"가장 가까움"을 측정 하는 방법

➔유클리드 거리법(Euclidean Distance) [점과 점 사이의 거리]

$$\sqrt{(x_1-u_1)^2+(x_2-u_2)^2+\cdots+(x_p-u_p)^2}$$

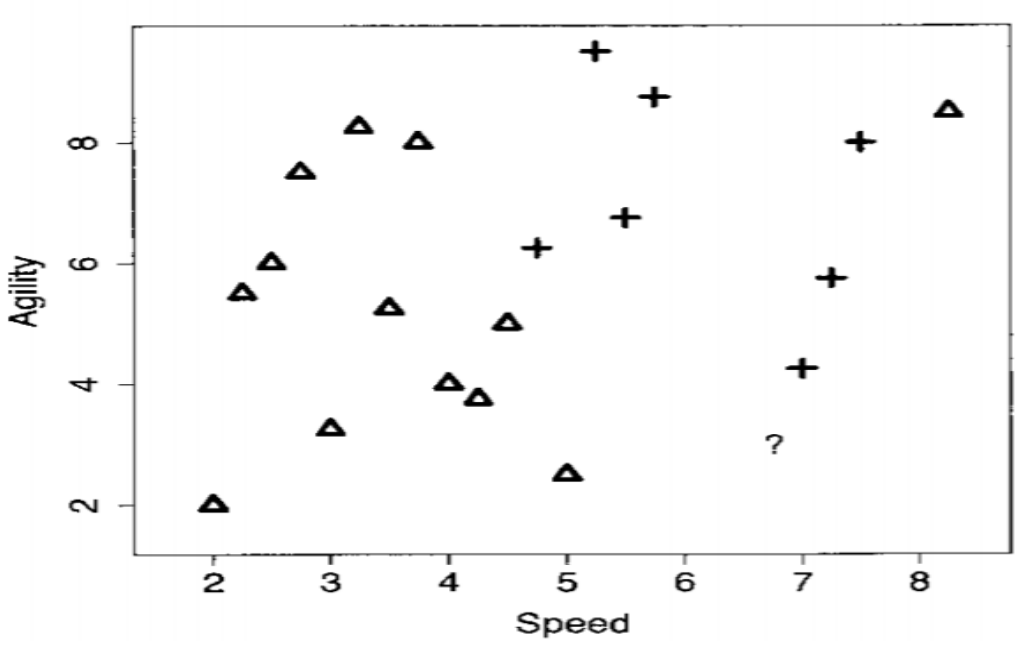
Normalization의 필요성

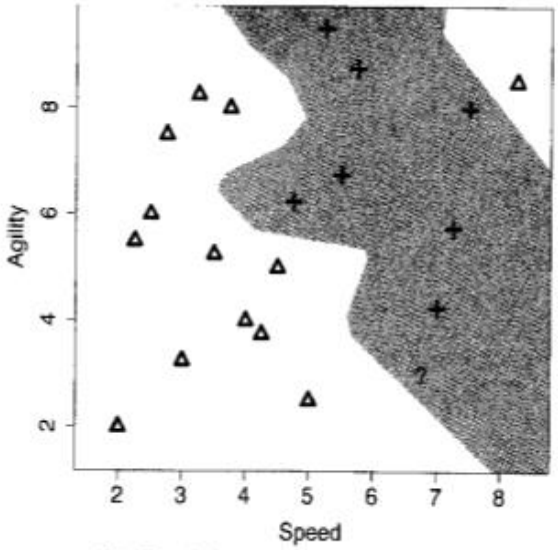
	연식	주행거리(km)	class
A	1.5	5000	GOOD
B	20	4800	BAD
X	1	100	?

단 하나의 최근접 이웃으로 분류를 하는 것이 옳은가?

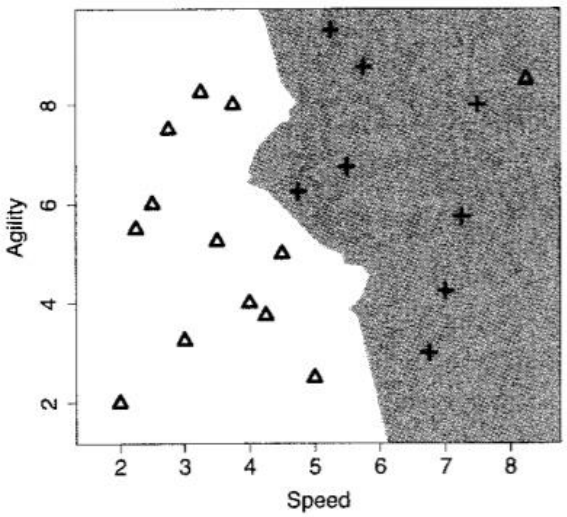
The SPEED and AGILITY ratings for 20 college athletes and whether they were drafted by a professional team.

ID	SPEED	AGILITY	DRAFT	ID	SPEED	AGILITY	DRAFT
1	2.50	6.00	no	11	2.00	2.00	no
2	3.75	8.00	no	12	5.00	2.50	no
3	2.25	5.50	no	13	8.25	8.50	no
4	3.25	8.25	no	14	5.75	8.75	yes
5	2.75	7.50	no	15	4.75	6.25	yes
6	4.50	5.00	no	16	5.50	6.75	yes
7	3.50	5.25	no	17	5.25	9.50	yes
8	3.00	3.25	no	18	7.00	4.25	yes
9	4.00	4.00	no	19	7.50	8.00	yes
10	4.25	3.75	no	20	7.25	5.75	yes

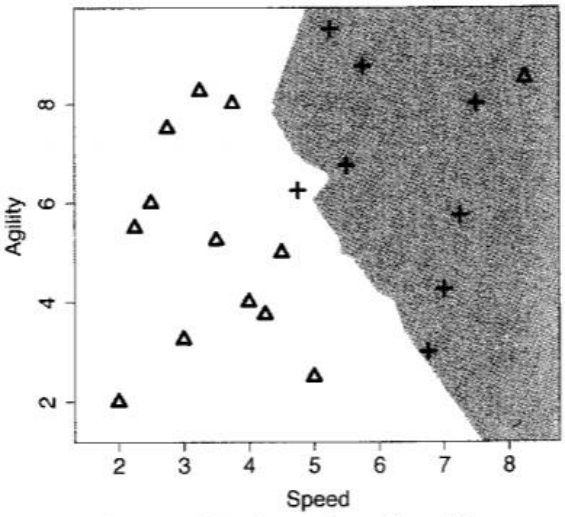




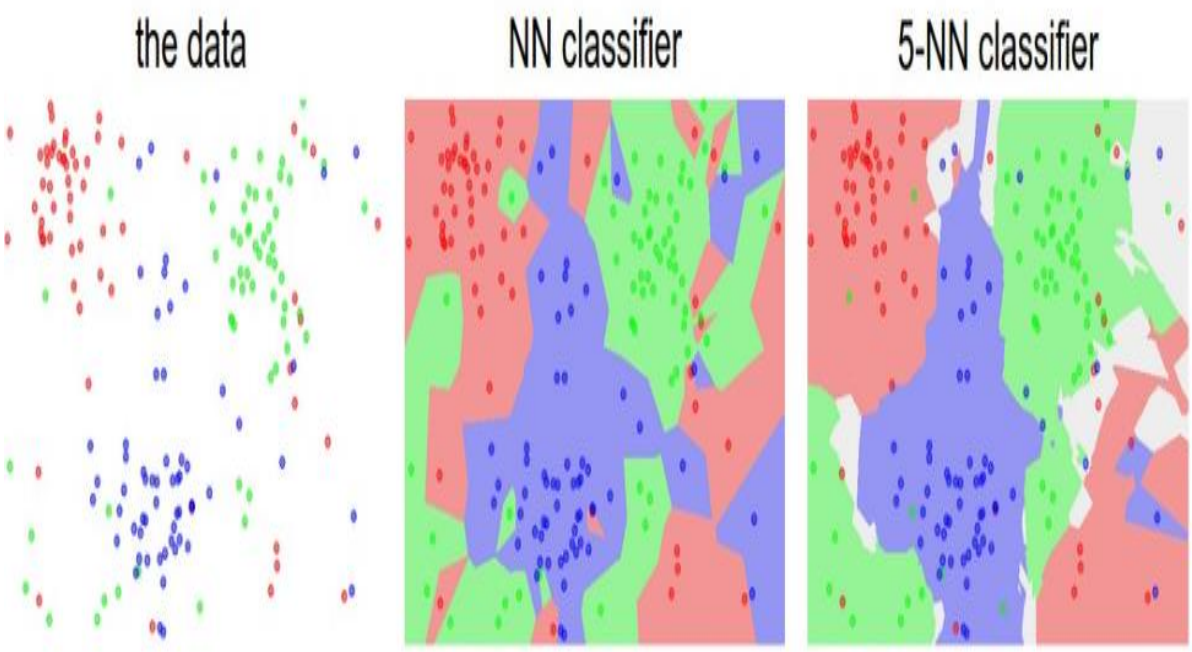
(b) Decision boundary ($k = 1$)



(a) Decision boundary ($k = 3$)



(b) Decision boundary ($k = 5$)

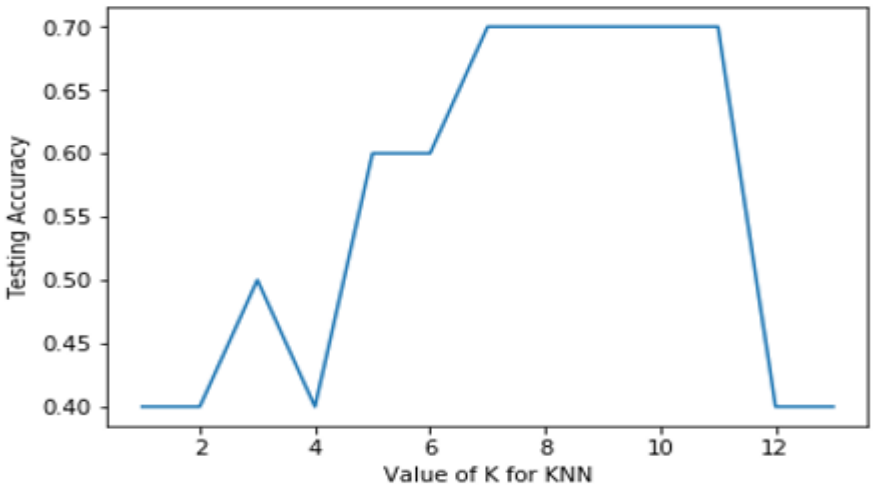


K 값의 결정

K값은 보통 3이상의 홀수로 한다.

-다수결로 이루어 지는 "분류"라는 의사결정을 위해

여러 K(3<= K <<n)값에 대해서 학습한 후 validation set으로 검증.



*검증 후

Accuracy 가 비슷한 수준일 때 가장 작은 K값 선택

K 값이 너무 작으면 → outlier(이상치)에 민감

K 값이 너무 크면 → over fitting

Q : If K==n ???

분류에 대한 평가에 대한 척도

		실제 정답	
		True	False
분류 결과	True	True Positive	False Positive
	False	False Negative	True Negative

정확도(Accuracy): (TP+TN)/ALL

-한계가 있음

ex) 암환자판단의 경우 중요한 것은 암환자를 정상이라고 판단하는 비율 낮게 하는 것 이다.

정밀도(precision): True라고 분류한 것들 중 정말 True인 비율

재현율(recall) or (sensitivity): 정답이 True인데 분류를 True라고 한 것의 비율

Precision 과 recall은 trade off 관계이다.

-둘 중에 하나를 올리면 어느 하나는 떨어질 수 있는 관계

-둘 다를 증가시킬 수는 없다.

Ex) 4월의 30일 중에 20일은 맑은 날인 데이터가 존재

모델이 판단하기에 가장 확실해 보이는 2일만 맑았다고 판단하고 (맑았다고 판단하는 기준을 높인다.) 나머지는 틀렸다고 판단한다면

TP = 2 FP = 0 ➔precision =1

TF =18 TN=10

Recall

➔0.1

Recall 을 올리기 위해 맑았다고 판단하는 기준을 조금 낮추면

FP(실제로는 어두웠는데 맑다고 판단)가 증가 ➔ precision down

TF는 작아짐 ➔ recall up

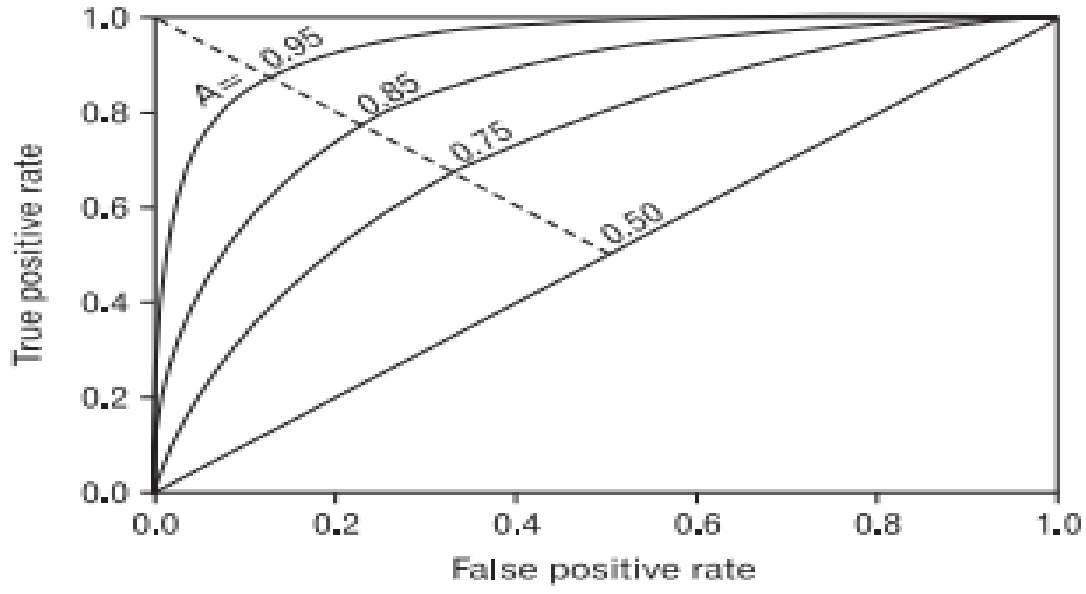
		실제 정답	
		True	False
분류 결과	True	True Positive	False Positive
	False	False Negative	True Negative

ROC – curve (수신자 판단 특성 곡선)

위 그림으로부터

TPR(왼쪽 빨강이 / 왼쪽 노랑이) 을 Y축,

FPR(오른쪽 빨강이/ 오른쪽노랑이) 을 X축 으로 해서 그린 곡선



TPR 은 성능의 좋음에 대한 지표 이고 FPR 은 성능의 좋지 않음에 대한 지표이다

위에서 언급 했던 내용을 상기해 보자

To make TPR(recall) up ➔ FN 감소시켜야함.

그러나, FN 감소하면

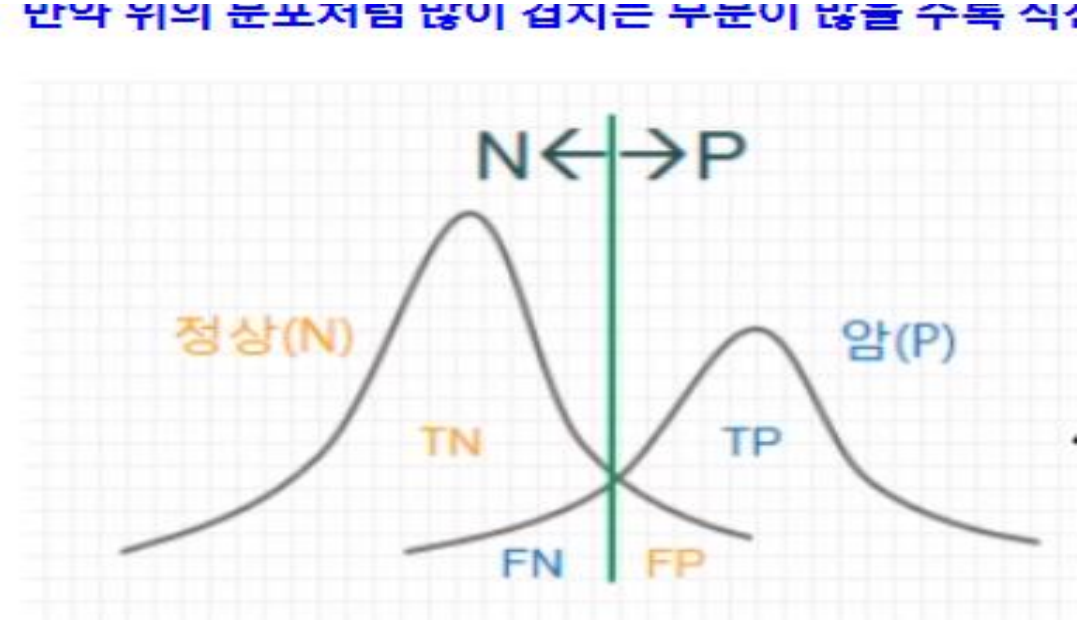
Trade off 효과로 FP 증가➔ FPR 또한 증가

분류 모델에 대한 평가를 위해서는 분류기준을 바꿔가면서 TPR 과 FPR의 변화 양상을 보며 판단해야 하며 그것을 위해 ROC-curve 이용

ROC- curve의 면적을 AUC(Area Under Curve)라 하며

AUC의 크기를 이용해 분류모델의 성능을 평가한다.

0.5~1사이의 값을 가지며 1에서의 성능이 제일 좋다.



주어진 데이터(곡선)에서 최적의 분류 기준(초록선) 일때 AUC 값이 가장 크게 되며 좌,우로 이동할 경우 AUC값이 작아진다.

머신러닝 , 딥러닝에서 ROC-curve를 모델판단의 척도로 사용

정리:

K-NN

주변에 존재 하는 인접한 K개의 record로부터 분류 해 낸다.

변수들의 scale이 다르므로 정규화를 해주어야 한다.

적당한 K값을 찾아 내야 한다.(작으면 이상치에 민감, 크면 overfitting)

장점:

1. 간단하다.
2. 모델을 설정할 필요없이 데이터를 바로 이용하면 된다.

단점:

1. 변수의 수가 늘어날수록 필요한 training set의 크기가 커진다.

   -변수(차원)이 늘어남에 따라 계산 record간의 거리가 커지게 되므로 적은 dataset으로는 유의미하게 가까운 record를 찾는 것이 힘들다.

2. Training set이 커짐에 따라 필요한 계산 횟수도 커진다.
3. 모델 설정을 하지 않으므로 실행할 때 마다 반복적으로 계산을 처음부터 다시 해야함
4. 분류 카테고리가 3가지 이상일 때 동점일 확률이 있다.

(이경우 랜덤하게 둘 중에서 선택한다.)