17기 정규세션

TOBIG's 16기 박한나

KNN

K-Nearest Neighbor

CONTENTS

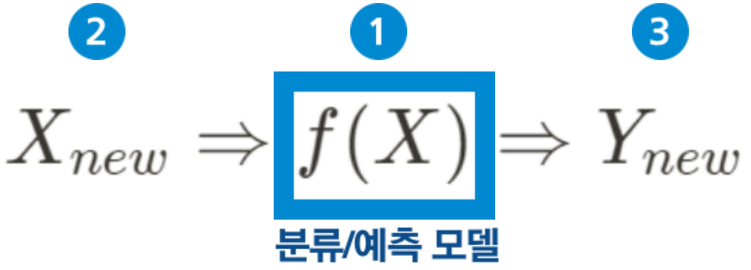
UNIT 01 | KNN

분류 및 예측을 위한 모델

Model-based vs Instance-based

1. Model-based Learning

데이터로부터 모델을 생성하여 분류 / 예측을 진행

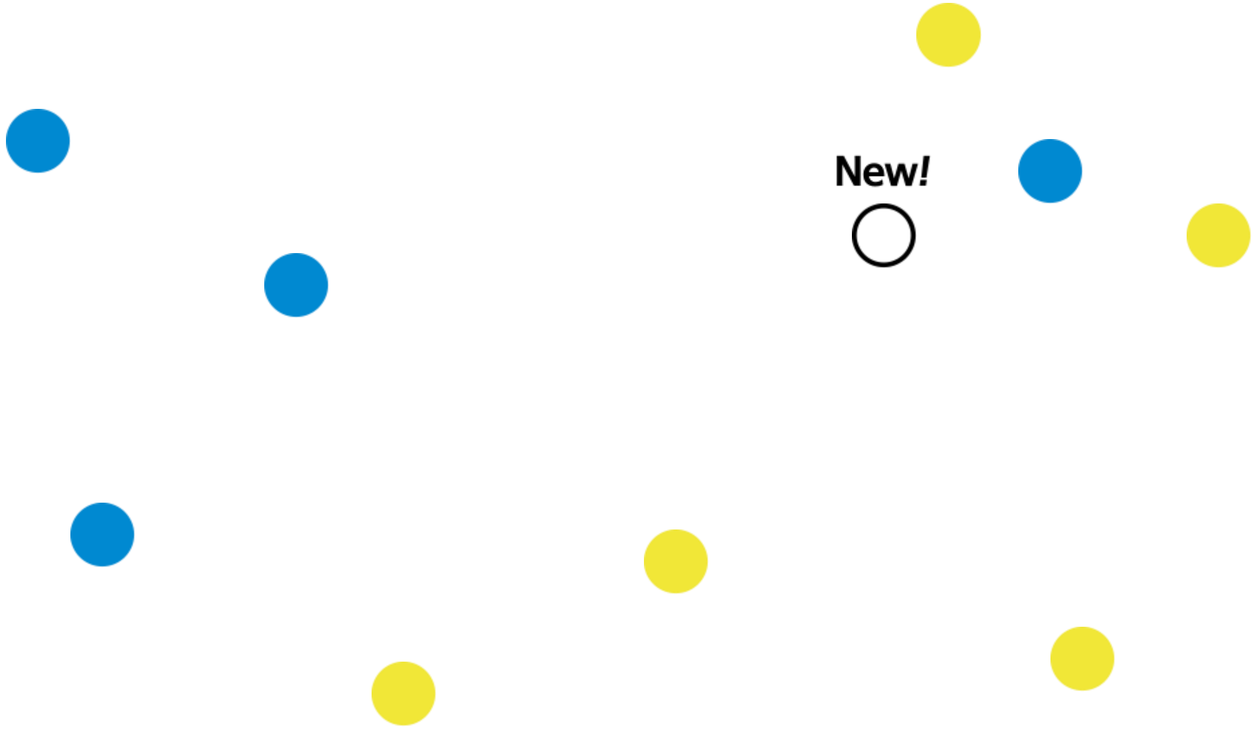


2. Instance-based Learning

데이터로부터 모델을 생성하여 분류 / 예측을 진행

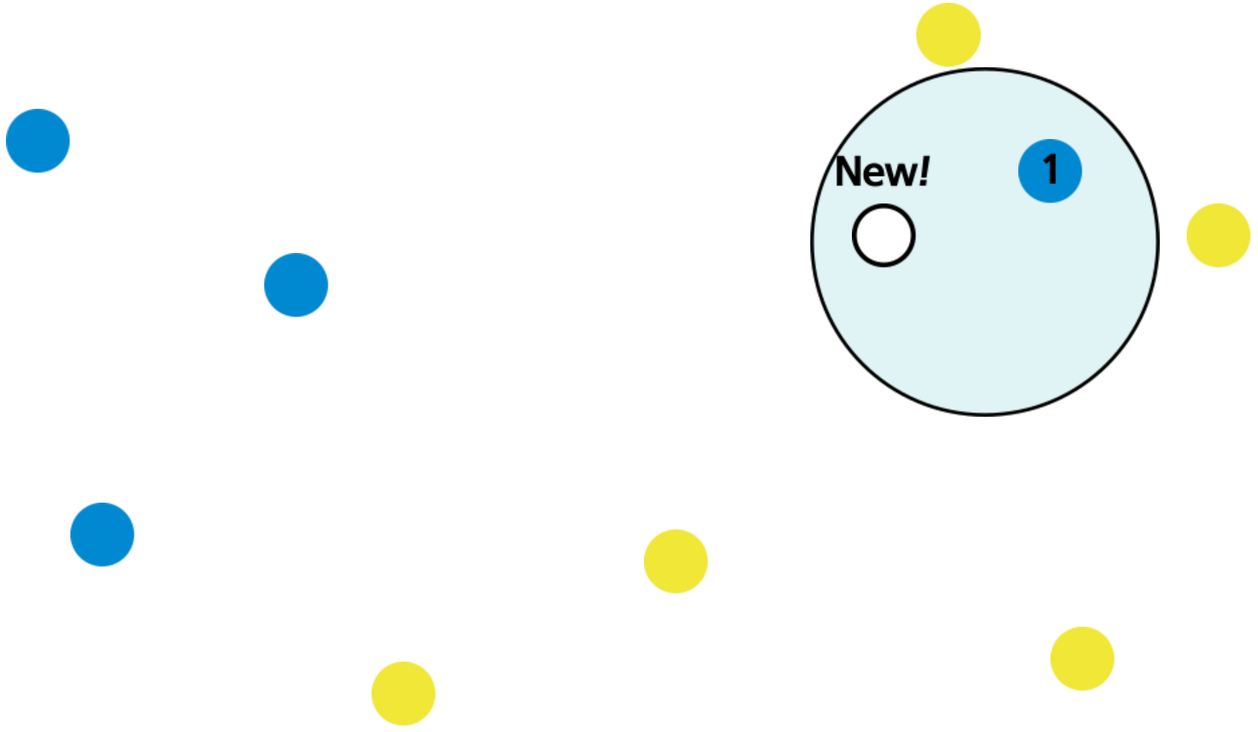
최근접 이웃
Nearest Neighbors

- n – Nearest Neighbor



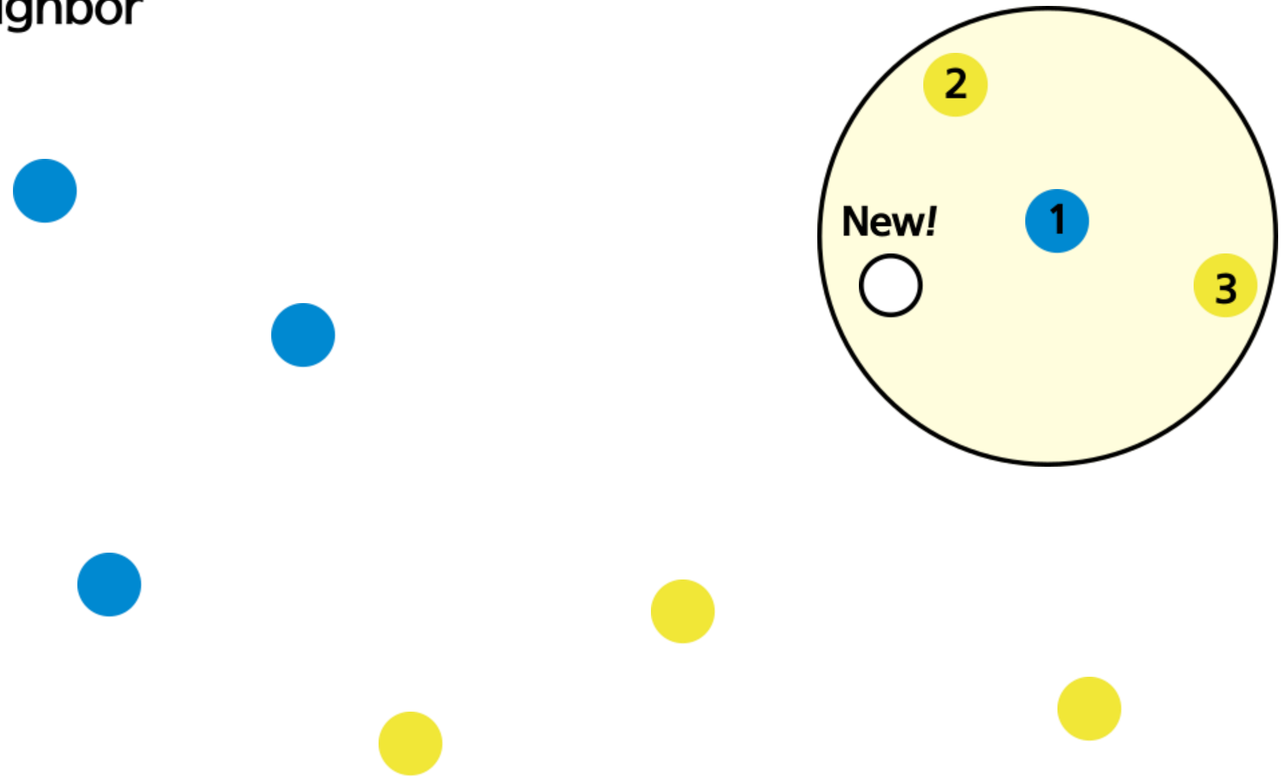
최근접 이웃
Nearest Neighbors

- 1 – Nearest Neighbor



최근접 이웃
Nearest Neighbors

- 3 – Nearest Neighbor



KNN 알고리즘의 구분 및 특징

K-Nearest Neighbor Algorithm

1 Instance-based Learning

: 각각의 관측치 (instance) 만을 이용하여 새로운 데이터에 대한 예측을 진행한다.

2 Memory-based Learning

: 모든 학습 데이터를 메모리에 저장한 후, 이를 바탕으로 예측을 시도한다.

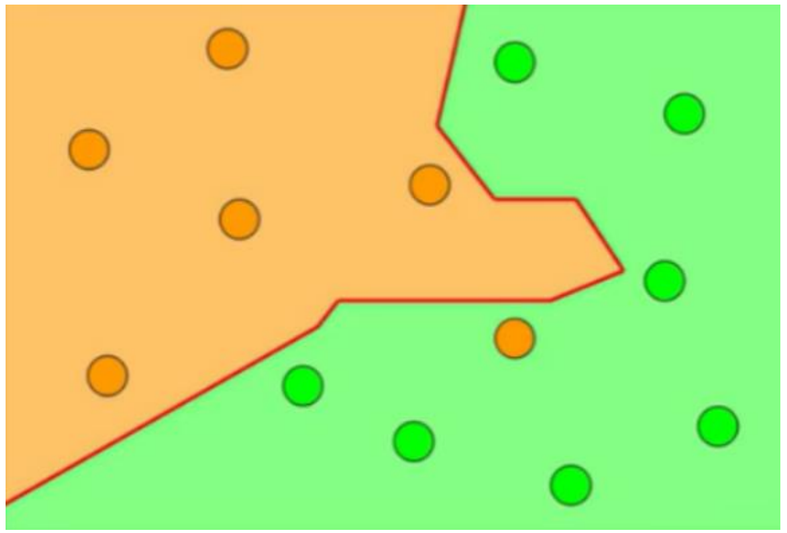
3 Lazy Learning

: 모델을 별도로 학습하지 않고, 테스팅 데이터가 들어와야 비로소 작동하는 게으른 알고리즘이다.

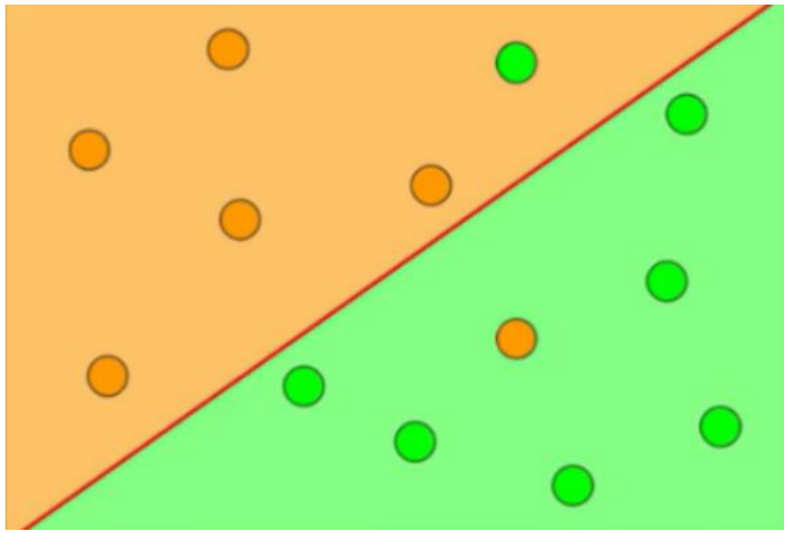
KNN과 선형모델 비교

KNN Boundary vs Linear Boundary

KNN (k=3) Boundary



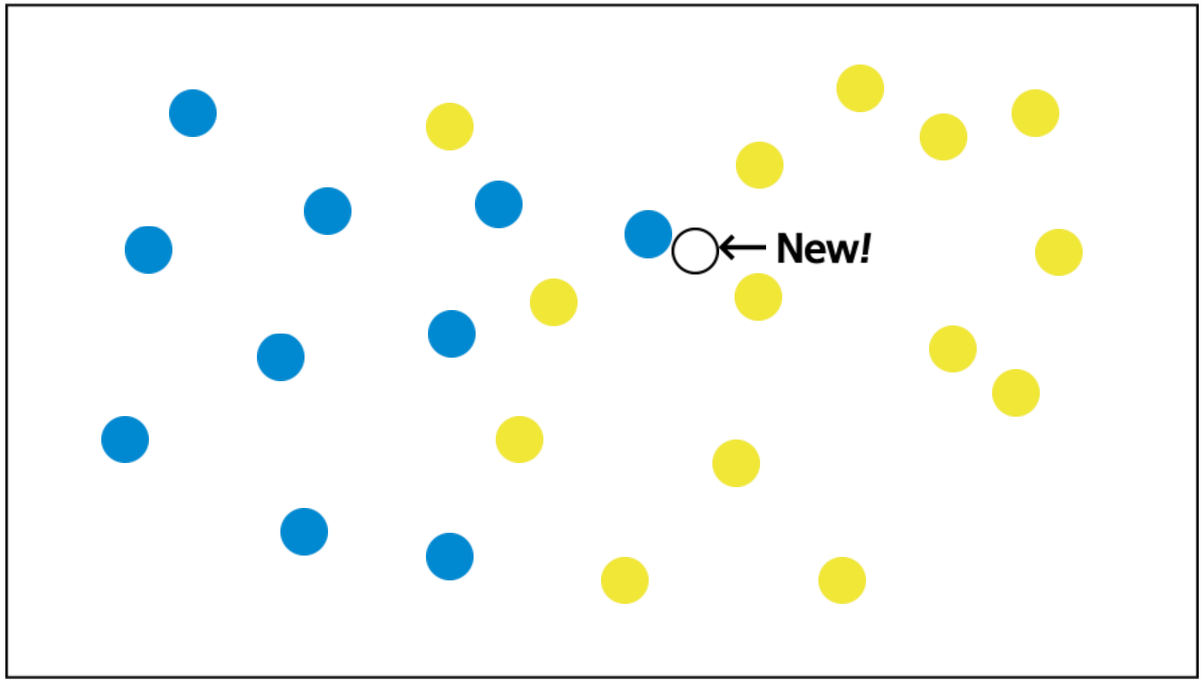
Linear Boundary



KNN 분류문제

Classification

- 인접한 K 개의 데이터로부터 majority voting을 시행한다.

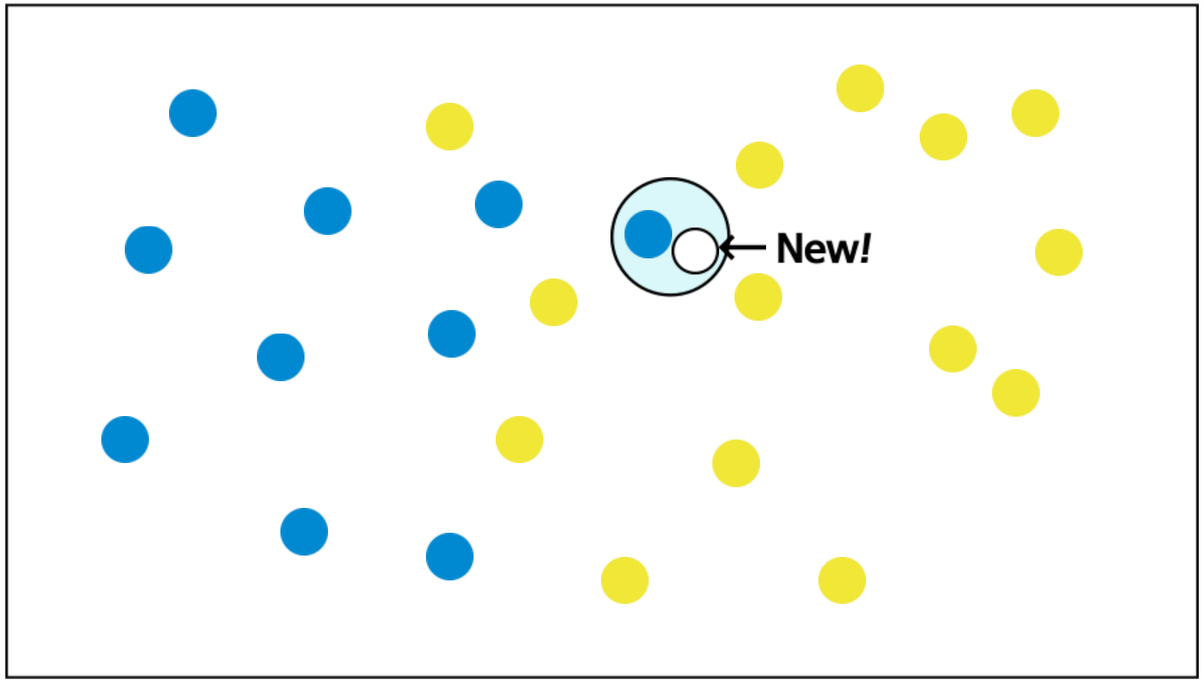


K = # of nearest neighbors

KNN 분류문제

Classification

- 인접한 K 개의 데이터로부터 majority voting을 시행한다.



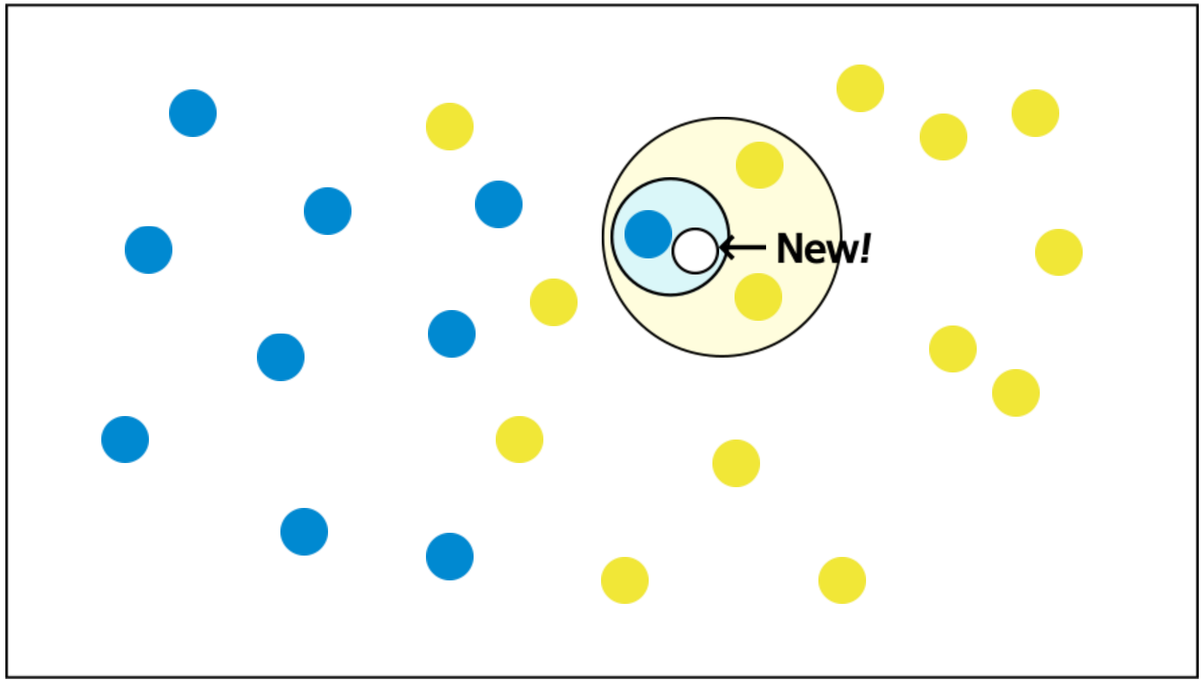
K = # of nearest neighbors

K=1: Blue

KNN 분류문제

Classification

- 인접한 K 개의 데이터로부터 majority voting을 시행한다.



K = # of nearest neighbors

K=1: Blue

K=3: Yellow

KNN 분류문제

Classification

예제

사람	유전자 A	유전자 B	유전자 C	유전자 D	질병유무
A	2.54	4.33	3.99	2.57	정상
B	3.12	3.87	3.84	3.04	정상
C	2.76	4.17	5.63	3.28	정상
D	3.87	3.56	4.25	3.65	질병
E	3.55	3.91	2.68	4.22	질병
F	4.12	2.86	3.30	3.71	질병

KNN 분류문제

Classification

예제

사람	유전자 A	유전자 B	유전자 C	유전자 D	질병유무
A	2.54	4.33	3.99	2.57	정상
B	3.12	3.87	3.84	3.04	정상
C	2.76	4.17	5.63	3.28	정상
D	3.87	3.56	4.25	3.65	질병
E	3.55	3.91	2.68	4.22	질병
F	4.12	2.86	3.30	3.71	질병
NEW G	3.24	3.68	3.82	3.77	?

KNN 분류문제

Classification

예제

	사람	유전자 A	유전자 B	유전자 C	유전자 D	질병유무	Distance from New
	A	2.54	4.33	3.99	2.57	정상	1.54
	B	3.12	3.87	3.84	3.04	정상	0.76
	C	2.76	4.17	5.63	3.28	정상	2.00
	D	3.87	3.56	4.25	3.65	질병	0.78
	E	3.55	3.91	2.68	4.22	질병	1.28
	F	4.12	2.86	3.30	3.71	질병	1.31
NEW	G	3.24	3.68	3.82	3.77	?	

KNN 분류문제

Classification

예제

사람	유전자 A	유전자 B	유전자 C	유전자 D	질병유무	Distance from New
A	2.54	4.33	3.99	2.57	정상	1.54
B	3.12	3.87	3.84	3.04	정상	0.76
C	2.76	4.17	5.63	3.28	정상	2.00
D	3.87	3.56	4.25	3.65	질병	0.78
E	3.55	3.91	2.68	4.22	질병	1.28
F	4.12	2.86	3.30	3.71	질병	1.31
NEW G	3.24	3.68	3.82	3.77	?	K = 1 : 정상

KNN 분류문제

Classification

예제

	사람	유전자 A	유전자 B	유전자 C	유전자 D	질병유무	Distance from New
	A	2.54	4.33	3.99	2.57	정상	1.54
	B	3.12	3.87	3.84	3.04	정상	0.76
	C	2.76	4.17	5.63	3.28	정상	2.00
	D	3.87	3.56	4.25	3.65	질병	0.78
	E	3.55	3.91	2.68	4.22	질병	1.28
	F	4.12	2.86	3.30	3.71	질병	1.31
NEW	G	3.24	3.68	3.82	3.77	?	

K = 1 : 정상
K = 3 : 질병

KNN 분류문제

Classification

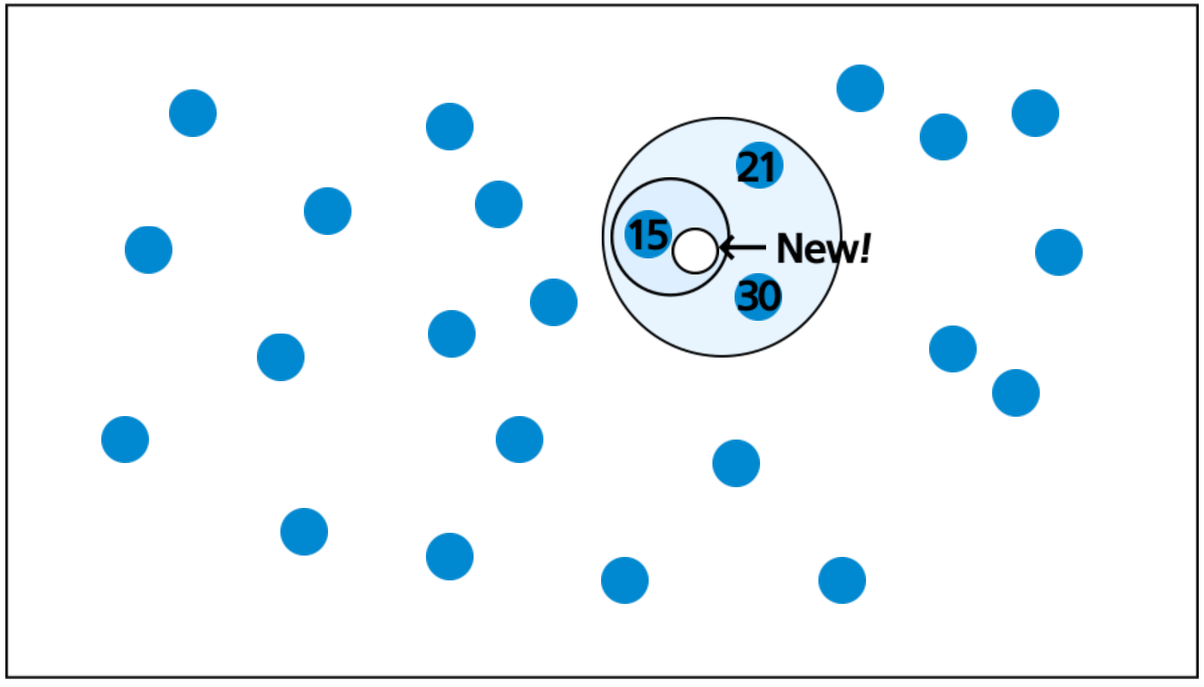
분류 알고리즘

1. 분류할 관측치 x를 선택한다.
2. x로부터 인접한 k개의 학습 데이터를 탐색한다.
3. 탐색된 k개 학습 데이터의 majority class c를 정의한다.
4. c를 x의 분류결과로 반환한다.

KNN 예측문제

Regression

- 인접한 K 개의 데이터들의 평균값으로 예측한다.



K = # of nearest neighbors

K=1: New = 15

K=3: New = (15+21+30)/3 = 22

KNN 예측문제

Regression

예제

사람	미녀와 야수	그린북	라라랜드	극한직업	명량	항거
A	7.5	7.5	7.0	9.5	8.5	5.0
B	7.5	7.0	7.5	8.0	8.0	6.0
C	8.0	7.0	8.0	8.0	8.5	8.5
D	8.5	8.0	9.5	7.5	6.0	7.0
E	10.0	9.5	9.0	7.5	7.5	10.0
F	9.0	9.0	8.0	8.0	8.0	9.0

KNN 예측문제

Regression

예제

사람	미녀와 야수	그린북	라라랜드	극한직업	명량	항거
A	7.5	7.5	7.0	9.5	8.5	5.0
B	7.5	7.0	7.5	8.0	8.0	6.0
C	8.0	7.0	8.0	8.0	8.5	8.5
D	8.5	8.0	9.5	7.5	6.0	7.0
E	10.0	9.5	9.0	7.5	7.5	10.0
F	9.0	9.0	8.0	8.0	8.0	9.0
NEW G	9.0	8.5	8.0	7.0	8.0	?

KNN 예측문제

Regression

예제

사람	미녀와 야수	그린북	라라랜드	극한직업	명량	항거	Distance from New
A	7.5	7.5	7.0	9.5	8.5	5.0	3.28
B	7.5	7.0	7.5	8.0	8.0	6.0	2.40
C	8.0	7.0	8.0	8.0	8.5	8.5	2.12
D	8.5	8.0	9.5	7.5	6.0	7.0	2.65
E	10.0	9.5	9.0	7.5	7.5	10.0	1.87
F	9.0	9.0	8.0	8.0	8.0	9.0	1.12
NEW G	9.0	8.5	8.0	7.0	8.0	?	K = 1 : 9.0

KNN 예측문제

Regression

예제

사람	미녀와 야수	그린북	라라랜드	극한직업	명량	항거	Distance from New
A	7.5	7.5	7.0	9.5	8.5	5.0	3.28
B	7.5	7.0	7.5	8.0	8.0	6.0	2.40
C	8.0	7.0	8.0	8.0	8.5	8.5	2.12
D	8.5	8.0	9.5	7.5	6.0	7.0	2.65
E	10.0	9.5	9.0	7.5	7.5	10.0	1.87
F	9.0	9.0	8.0	8.0	8.0	9.0	1.12
NEW G	9.0	8.5	8.0	7.0	8.0	?	

K = 1 : 9.0

K = 3 : (9.0+10.0+8.5)/3 = 9.17

KNN 예측문제

Regression

예측 알고리즘

1. 예측할 관측치 x를 선택한다.
2. x로부터 인접한 k개의 학습 데이터를 탐색한다.
3. 탐색된 k개 학습 데이터의 평균을 x의 예측값으로 반환한다.

UNIT 02 | KNN 하이퍼파라미터: K

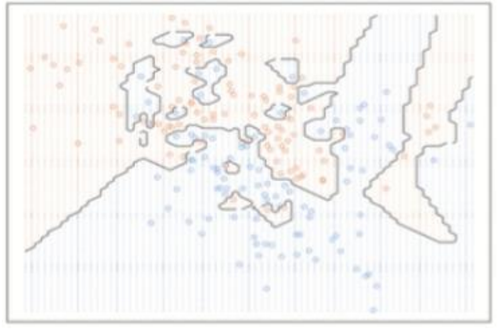
“

인접한 학습 데이터를 몇 개까지 탐색할 것인가?

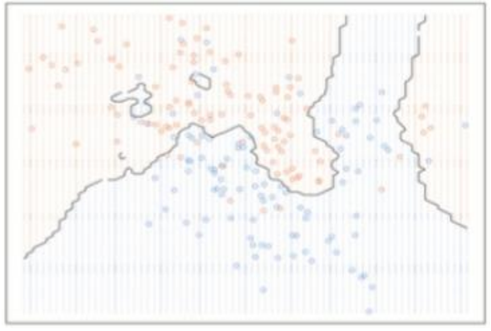
K에 따른 결과

Hyperparameter: K

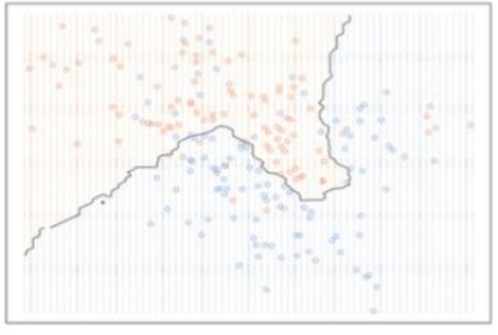
- 1 ≤ K ≤ 전체 데이터 개수



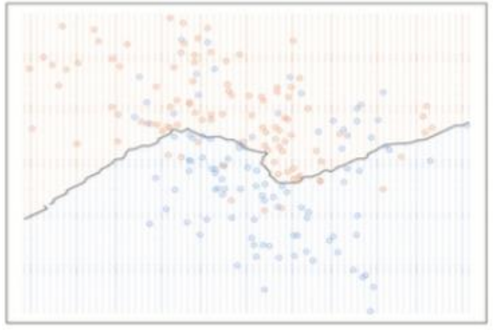
1 – Nearest Neighbor



5 – Nearest Neighbor



15 – Nearest Neighbor



50 – Nearest Neighbor

K가 매우 작을 경우

: 데이터의 지역적 특성을 지나치게 반영함 (overfitting)

K가 매우 클 경우

: 다른 범주의 개체를 너무 많이 포함하여 오분류할 위험 (underfitting)

K선택 방법

Hyperparameter: K

- 일정 범위 내로 K를 조정하여, 가장 좋은 예측 결과를 보이는 K값을 선정한다.

분류모델

$$MisclassError_k = \frac{1}{k}\sum_{i=1}^{k} I(c_i \neq \hat{c}_i) for k = 1, 2, ..., k^*$$

$$I(\cdot) : IndicatorFunction$$

예측모델

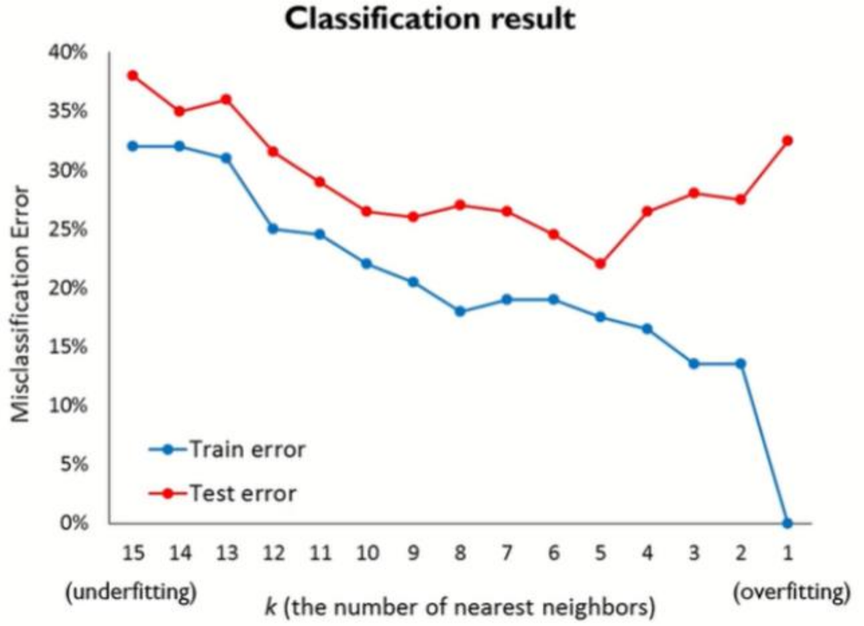
$$SSE_k = \sum_{i=1}^{k} (y_i - \hat{y}_i)^2 for k = 1, 2, ..., k^*$$

- K는 보통 홀수로 지정 (동점이 나오는 경우를 막기 위해)

K선택 방법

Hyperparameter: K

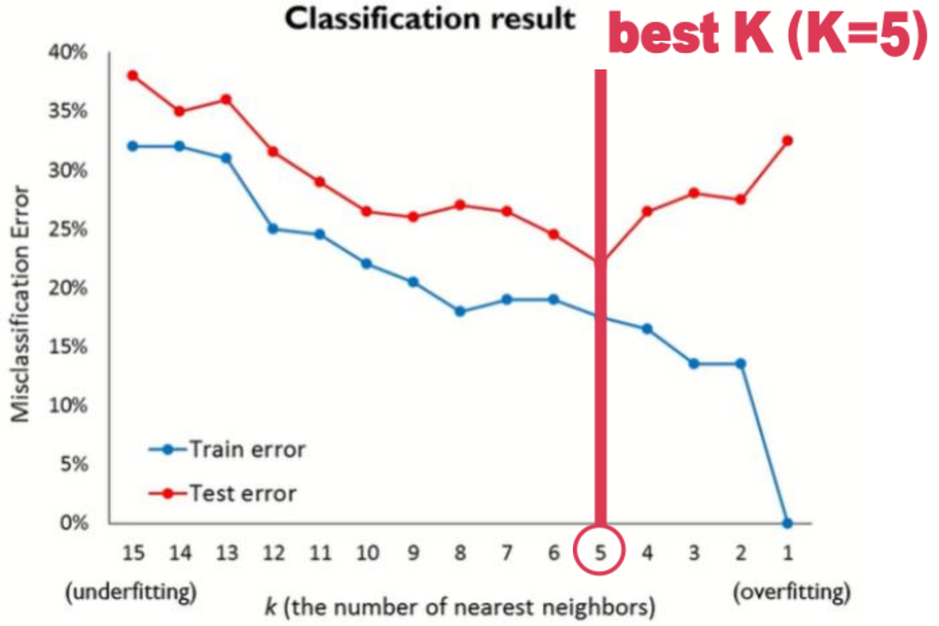
- 일정 범위 내로 K를 조정하여, 가장 좋은 예측 결과를 보이는 K값을 선정한다.



K선택 방법

Hyperparameter: K

- 일정 범위 내로 K를 조정하여, 가장 좋은 예측 결과를 보이는 K값을 선정한다.



UNIT 03 | KNN 하이퍼파라미터
: DISTANCE MEASURES

“

데이터 간 거리는 어떻게 측정할 것인가?

거리측도(1-유사도)

Hyperparameter: Distance Measures

- **다양한 거리측도 (Distance Measure) 존재한다.**

- **데이터 내 변수들이 각기 다른 데이터 범위, 분산 등을 가질 수 있으므로 데이터 정규화(혹은 표준화)를 통해 이를 맞추는 것이 중요하다.**
  - 거리를 계산할 때, 단위가 큰 특정 변수(들)가 거리를 결정하는 것 방지
  - 예) 다음과 같은 특성변수 3개가 있을 때: 키(1.5m~1.8m), 몸무게(90lb~300lb), 연봉(20,000,000원 ~ 100,000,000원)

EUCLIDEAN DISTANCE

유클리디언 거리

- 가장 흔히 사용하는 거리 측도
- 대응되는 X, Y 값 간 차이 제곱합의 제곱근으로써, 두 관측치 사이의 직선 거리를 의미함

$$d_{(X,Y)} = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

유클리디언 거리 공식

EUCLIDEAN DISTANCE

유클리디언 거리

- 가장 흔히 사용하는 거리 측도
- 대응되는 X, Y 값 간 차이 제곱합의 제곱근으로써, 두 관측치 사이의 직선 거리를 의미함

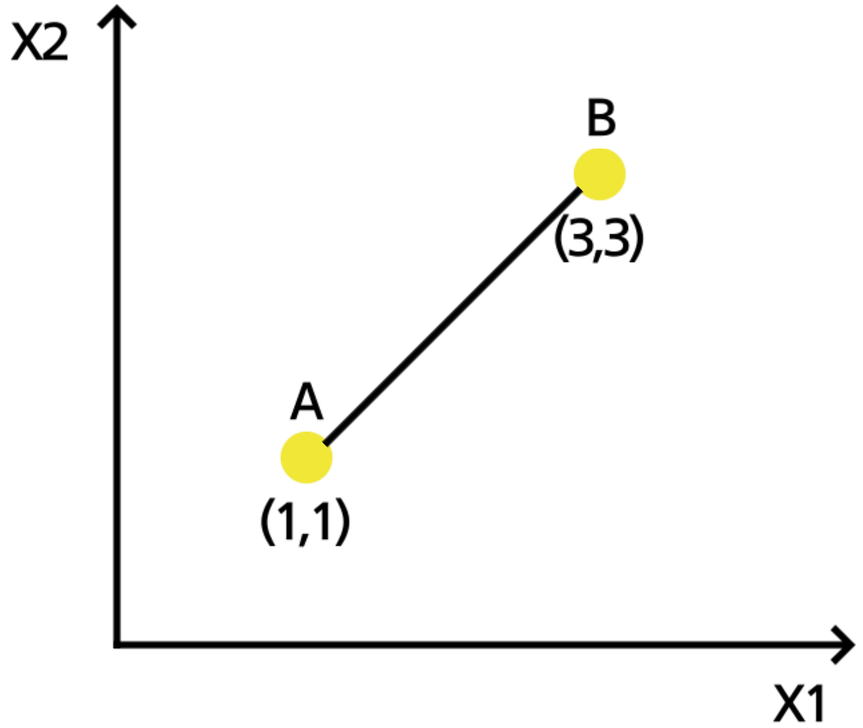
$$d_{(X,Y)} = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

$$A = (a_1, a_2, ..., a_p), B = (b_1, b_2, ..., b_p)$$

$$d_{(A,B)} = \sqrt{(a_1 - b_1)^2 + \cdots + (a_p - b_P)^2} = \sqrt{\sum_{i=1}^{p}(a_i - b_i)^2}$$

EUCLIDEAN DISTANCE

유클리디언 거리

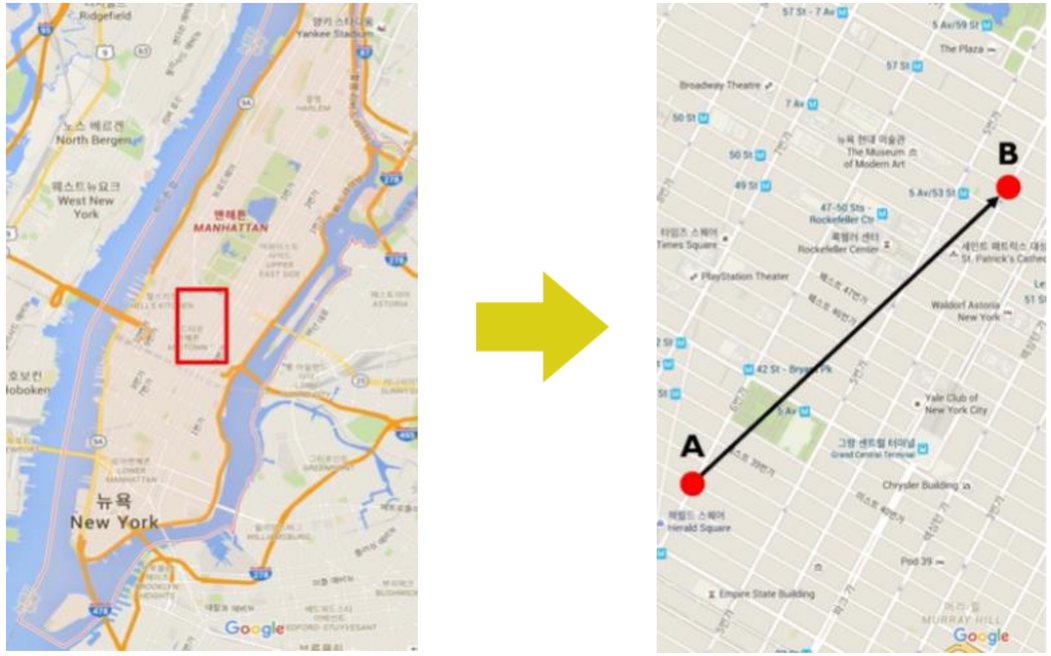


$$d_{(A,B)} = \sqrt{(3-1)^2 + (3-1)^2} = \sqrt{8}$$

MANHATTEN DISTANCE

맨해튼 거리

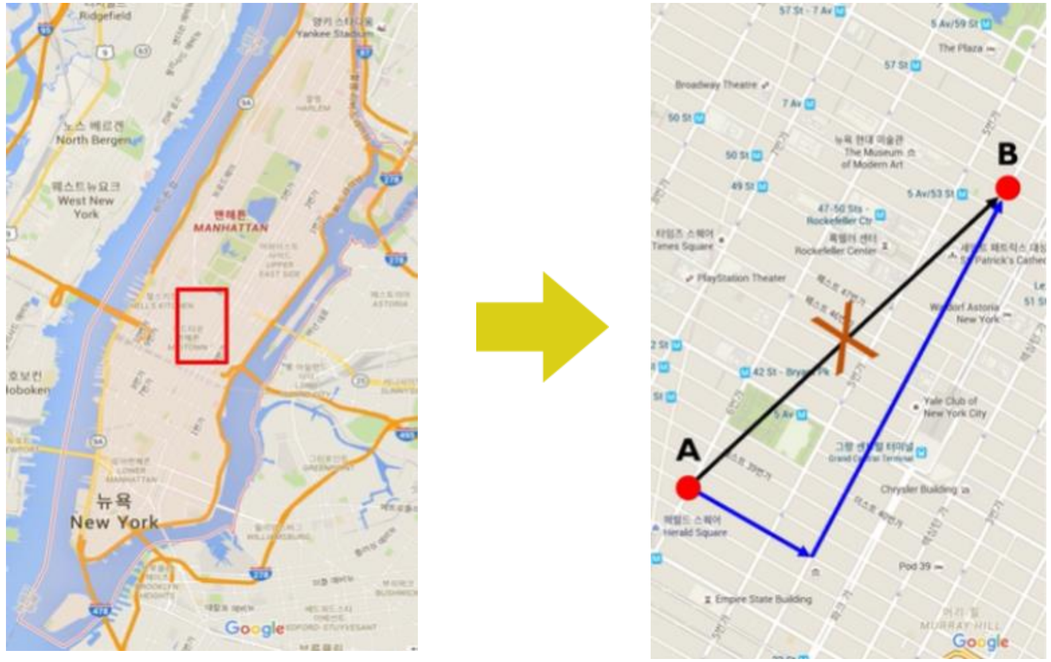
- X에서 Y로 이동 시 각 좌표축 방향으로만 이동할 경우에 계산되는 거리



MANHATTEN DISTANCE

맨해튼 거리

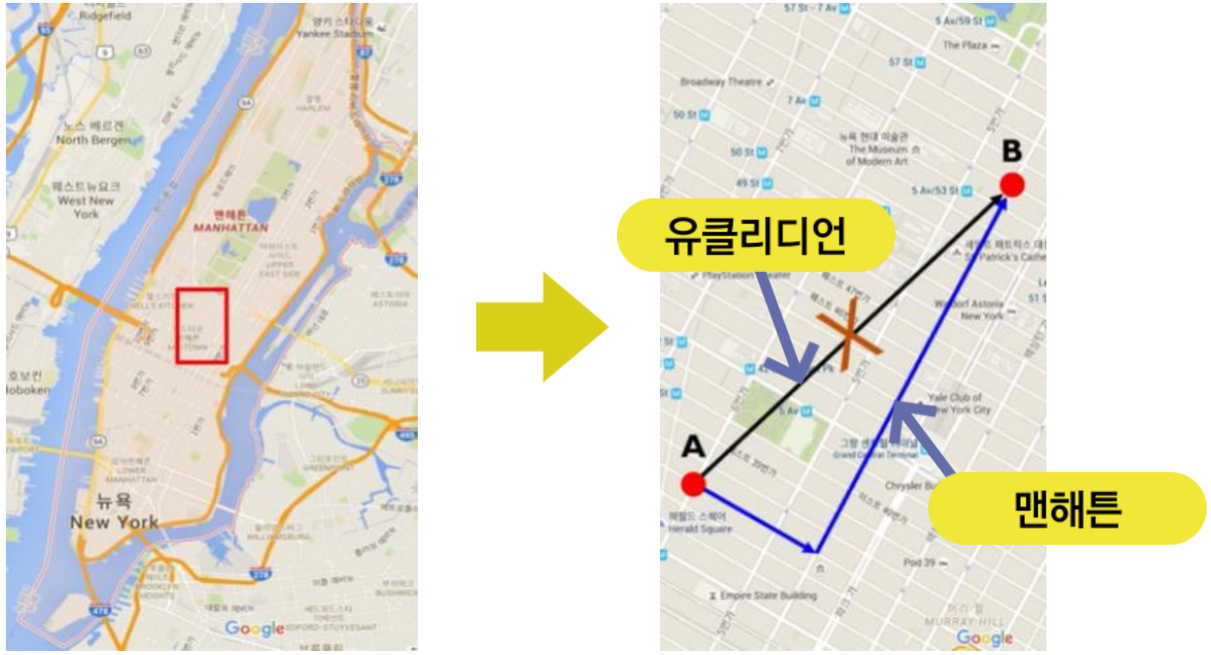
- X에서 Y로 이동 시 각 좌표축 방향으로만 이동할 경우에 계산되는 거리



MANHATTEN DISTANCE

맨해튼 거리

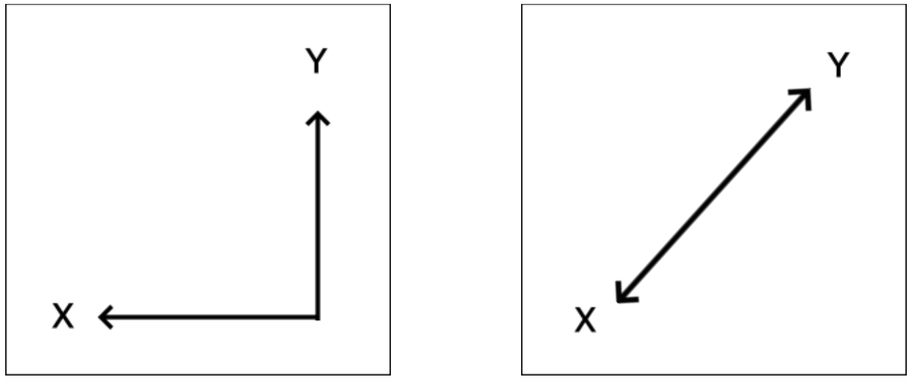
- X에서 Y로 이동 시 각 좌표축 방향으로만 이동할 경우에 계산되는 거리



MANHATTEN DISTANCE

맨해튼 거리

$$d_{Manhattan(X,Y)} = \sum_{i=1}^{n} |x_i - y_i|$$



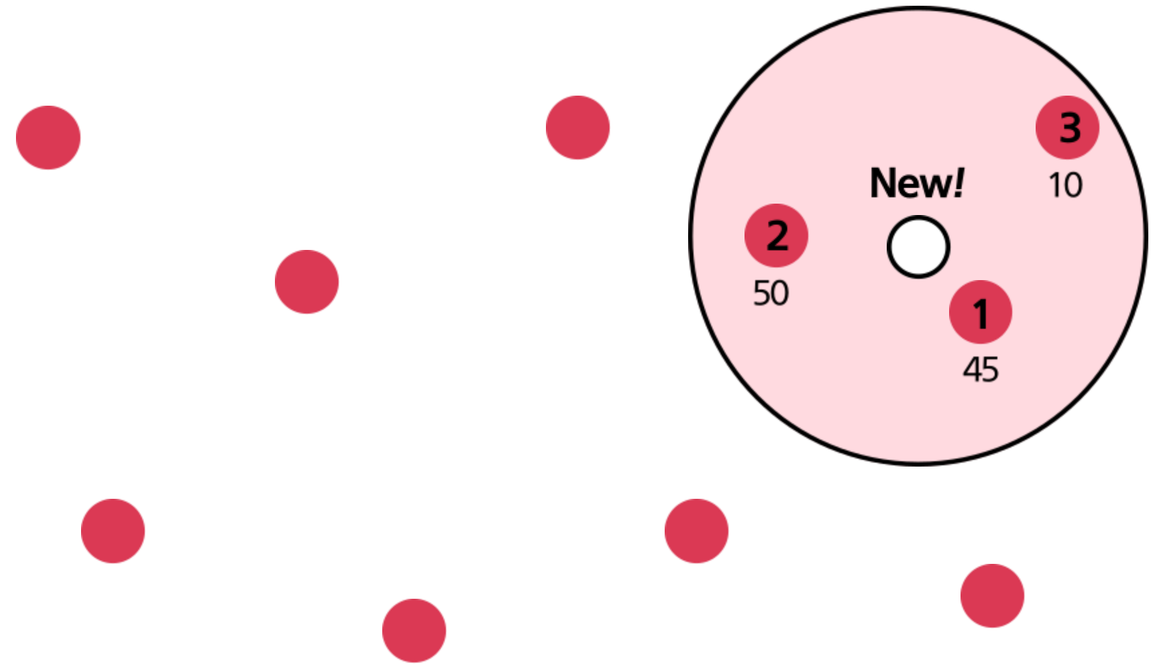
맨해튼 유클리디언

UNIT 04 | WEIGHTED KNN

3NN 예측모델

3-Nearest Neighbors Regression

- 3NN 예측모델 예제



3NN 예측모델

3-Nearest Neighbors Regression

3NN 예측모델 예제

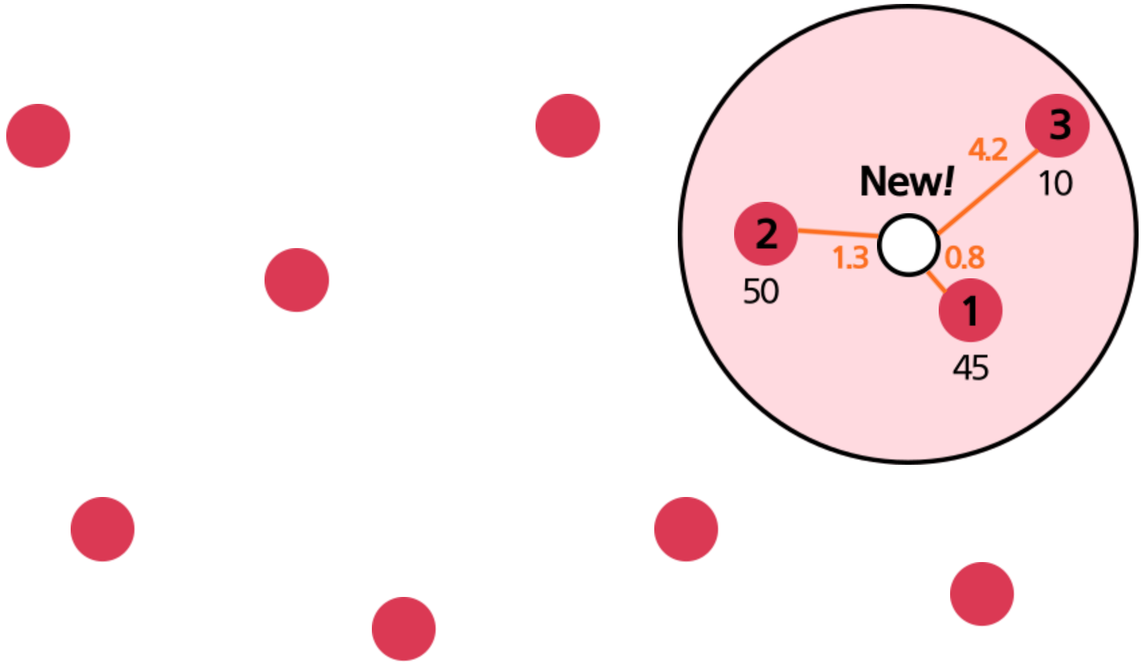
“

관측치 1, 2, 3이 전부 같은 가중치를 가져야 하는 가?

WEIGHTED 3NN 예측모델

Weighted 3-Nearest Neighbors Regression

- Weighted 3NN 예측모델 예제



New: (45+50+10) / 3 = 35

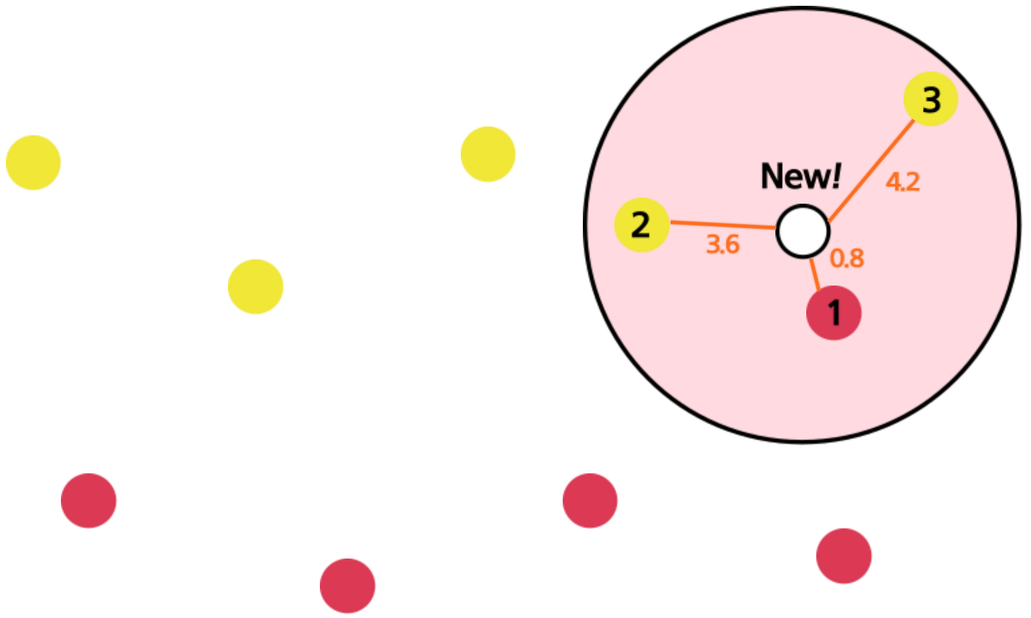
New_weighted:

$$(\frac{1}{0.8^2} \cdot 45 + \frac{1}{1.3^2} \cdot 50 + \frac{1}{4.2^2} \cdot 10)/(\frac{1}{0.8^2} + \frac{1}{1.3^2} + \frac{1}{4.2^2}) = 45.4$$

3NN 분류모델

3-Nearest Neighbors Classification

3NN 분류모델 예제



New: Yellow

New_weighted:

Yello = $\frac{1}{3.6^2} + \frac{1}{4.2^2} \cong 0.13$

Red = $\frac{1}{0.8^2} \cong 1.56$

→ Red

3NN 분류모델

3-Nearest Neighbors Classification

- 새 데이터와 기존 학습 관측치 간의 거리를 가중치로 하여 예측 결과를 도출한다.

예측모델

$$\hat{y}_{new} = \frac{\sum_{i=1}^{k} w_i y_i}{\sum_{i=1}^{k} w_i} \qquad where \;\; w_i = \frac{1}{d_{(new,x_i)}{}^2}$$

분류모델

$$\hat{c}_{new} = max_c \sum_{i=1}^{k} w_i I(w_i \in c) \qquad where \;\; w_i = \frac{1}{d_{(new,x_i)}{}^2}$$

UNIT 05 | KNN 고려사항

KNN 고려사항

거리 기반 알고리즘

Distance 기반 알고리즘

- 변수들의 단위 (Scale)에 민감 → Feature Scaling
- categorical은? → One-hot-encoding

FEATURE SCALING

데이터 전처리

Min-Max Normalization

: 데이터를 일반적으로 0 ~ 1 사이의 값으로 변환

$$x = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Standardization

: 데이터의 평균이 0, 표준편차가 1이되도록 변환

$$x = \frac{x - x_{mean}}{x_{std}}$$

⚠ **주의**

- train 데이터와 test 데이터의 scale을 따로 조정하면 안된다.
- train 데이터의 scale을 조정하고자 구한 정규화 parameter(최대최소, 평균/표준편차)를 기억하여 사용하여 test 데이터에도 변환해야 한다.

ONE-HOT ENCODING

원-핫 인코딩

- categorical 값을 feature로 만든 후 1 또는 0으로 지정하는 방법
- 1개만 Hot(1)이고 나머지는 Cold(0)
- KNN은 거리기반이므로 input에 numerical이 와야 함

color
red
green
blue
red

red	green	blue
1	0	0
0	1	0
0	0	1
1	0	0

UNIT 06 | KNN 장단점 및 요약

장점

KNN

KNN 장점

1. 데이터 내 노이즈 영향을 크게 받지 않으며, 특히 Mahalanobis distance와 같이 데이터의 분산을 고려할 경우 강건하다.

2. 학습 데이터의 수가 많을 경우 효과적이다.

한계점

KNN

KNN 한계점

1. 파라미터 K의 값을 설정해야 한다.
2. 어떤 거리 척도가 분석에 적합한지 불분명하며, 따라서 데이터의 특성에 맞는 거리 측도를 임의로 선정해야 한다.
3. 새로운 관측치와 각각의 학습 데이터 간 거리를 전부 측정해야 하므로, 계산시간이 오래 걸리는 단점이 있다.

요약

KNN

요약

- KNN은 매우 단순한 접근방식으로 새로운 관측치를 분류 및 예측할 수 있는 방법이다.
- 선형모델과 같이 학습 데이터로부터 특정 형태의 모델을 제시하는 것이 아니라, 학습 데이터 내 유사한 관측치들만을 토대로 새로운 데이터의 예측을 수행한다.
- 일부 유사한 관측치를 반응변수의 조합(ex. average, majority voting)을 통해 예상되는 반응 변수 값을 제공한다.
- weighted KNN 알고리즘으로 데이터의 가중치를 고려할 수 있으며, 이를 통해 보다 정확한 모델을 구축할 수 있다.

파이썬 실습

Python

"각자 해보기!"

❶ KNN 실습 1

- KNN 간단하게 구현해보기

❷ KNN 실습 2

- 파라미터를 바꿔가며 비교해보기
- 고려대학교 김성범 교수님 [파이썬 실습] K-nearest neighbors 알고리즘 자료
  https://www.youtube.com/watch?v=lGo7otnuVcg&list=PLpIPLT0Pf7Io8pMhxJ6vhM1chReYa8KIn&index=6
- 파이썬 및 모델 구현에 능숙하신 분들은 해당 파일을 쭉 따라서 실습을 진행해주시고, 구현이 어려우신 분들은 위 링크의 강의를 참고해주세요.

과제
Homework

KNN 구현해보기

1. Preprocessing / EDA
2. KNN & 하이퍼파라미터 튜닝
3. Evaluation

데이터: https://www.kaggle.com/llopesolivei/blackfriday

참고자료
reference

- 투빅스 15기 김현지님 강의자료
- 이 자료는 고려대학교 김성범 교수님의 핵심 머신러닝 K-nearest neighbors & Distance Measures 강의를 참고하여 제작했습니다.
  https://www.youtube.com/watch?v=W-DNu8nardo
- 투빅스 14기 김민경님 강의자료

공부자료 추천

- 이 강의자료와 같은 강의를 글로 정리해놓은 블로그
  https://ratsgo.github.io/machine%20learning/2017/04/17/KNN/

들어주셔서 감사합니다!