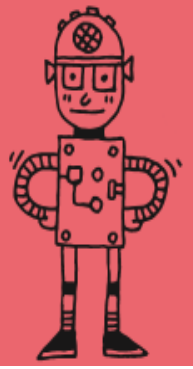
kNN

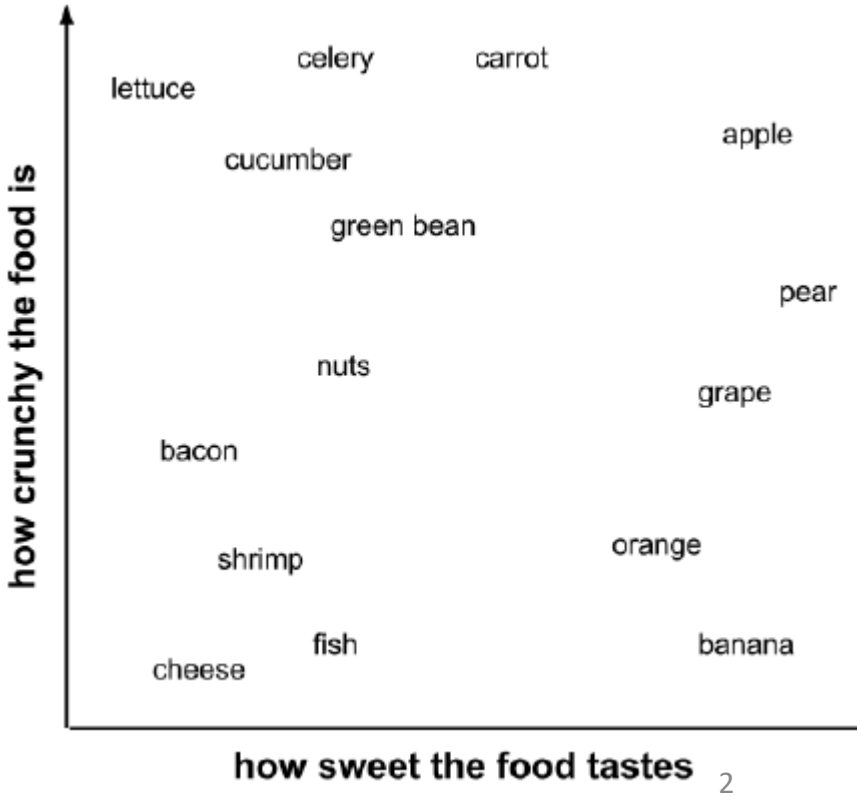
인하대학교
데이터사이언스 학과
김 승 환
swkim4610@inha.ac.kr



1 kNN(k Nearest Neighbor)

- 최근접 이웃 알고리즘(k nearest neighbor)은 거리를 이용해 분류를 수행하는 알고리즘이다.
- 토마토는 채소인가? 과일인가? 를 kNN을 통해 해결해보자.
- 채소와 과일의 특징을 구분할 수 있는
  Feature로 sweetness, crunchiness 를 선정
- Feature Selection: 집단을 구분할 수 있는 변수를 선정하는 것
  예: 남녀 구분(머리카락 길이, 키, 몸무게 등)

ingredient	sweetness	crunchiness	food type
apple	10	9	fruit
bacon	1	4	protein
banana	10	1	fruit
carrot	7	10	vegetable
celery	3	10	vegetable
cheese	1	1	protein

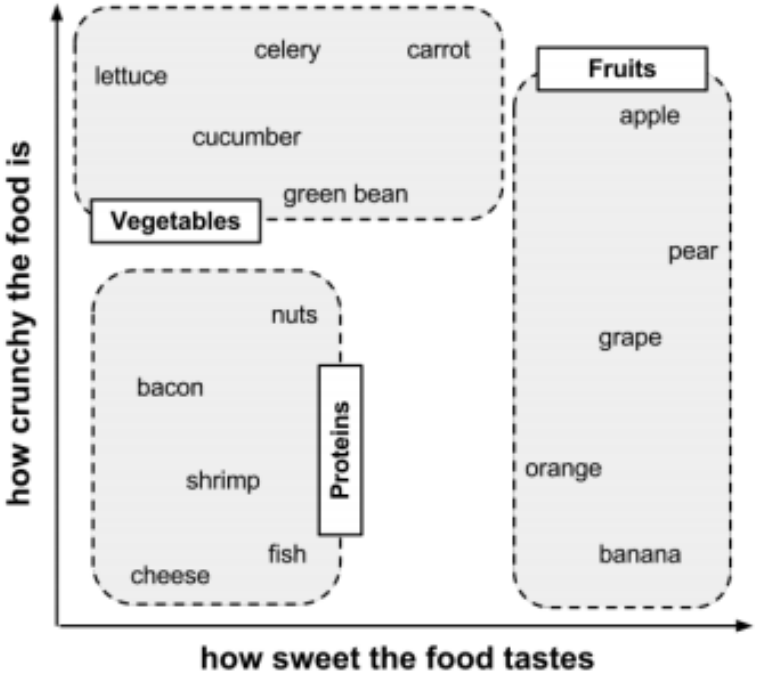


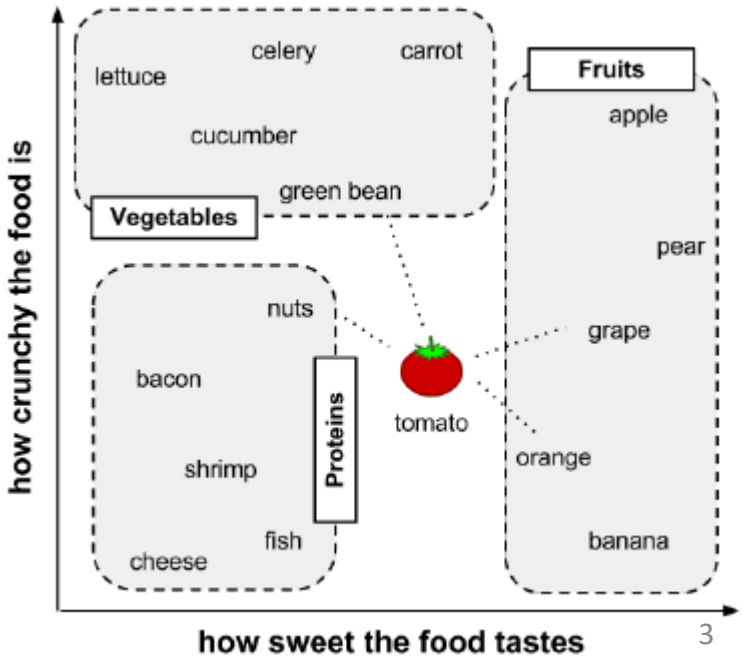
1 kNN(k Nearest Neighbor)

"유유상종(類類相從), Birds of a feather flock together."

좌측 그림처럼 같은 종류끼리 비슷한 위치에 모여 있음을 알 수 있다.

토마토가 과일인지 채소인지 알아보기 위해 토마토에서 가장 가까운 k=4개(4-Nearest Neighbor)의 종류를 보자.

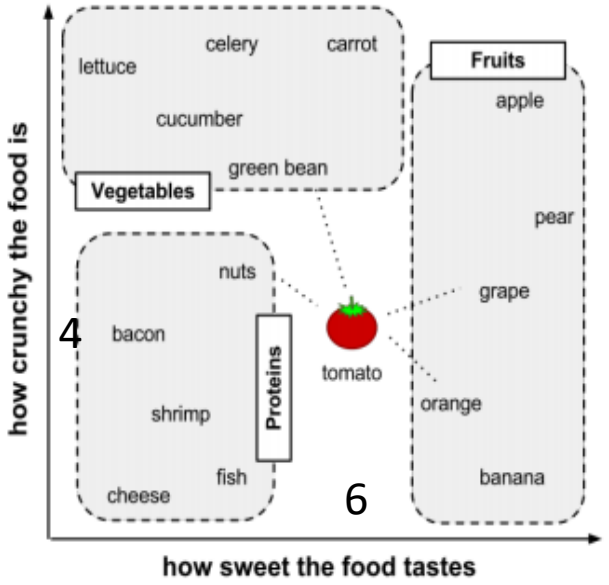




1 kNN(k Nearest Neighbor)

- 어떻게 토마토와 각 개체의 거리를 계산할 수 있을까?

sweetness of tomato: 6,
crunchiness of tomato: 4



n 차원 공간에서 두 점 (p, q) 거리는 유클리디안(Euclidean) 거리를 이용하여 구할 수 있다.

$$\text{dist}(p,q)=\sqrt{(p_1-q_1)^2+(p_2-q_2)^2+\ldots+(p_n-q_n)^2}$$

1 kNN(k Nearest Neighbor)

ingredient	sweetness	crunchiness	food type	distance to the tomato
grape	8	5	fruit	*sqrt((6 - 8)^2 + (4 - 5)^2) = 2.2*
green bean	3	7	vegetable	*sqrt((6 - 3)^2 + (4 - 7)^2) = 4.2*
nuts	3	6	protein	*sqrt((6 - 3)^2 + (4 - 6)^2) = 3.6*
orange	7	3	fruit	*sqrt((6 - 7)^2 + (4 - 3)^2) = 1.4*

- 1-NN: 토마토는 과일인 오렌지에 가깝다. 그러므로, 토마토는 과일로 판정한다.
- 3-NN: 토마토와 가장 가까운 (오렌지, 포도, 땅콩) 3개를 찾고 이들이 과일2종류, 단백질 1종류이므로 다수결에 의해 과일로 판정한다.
- Optimal k: 일반적으로 k는 3~10개 사이로 정하지만 절대적 기준은 아니다.
  학습 데이터에서 성과를 측정해 오분류가 적은 k를 결정하고 이를 테스트 셋에서 검증하여 결정하는 것이 합리적이다.
- k가 작으면 몇 개의 잘못된 데이터들이 판단에 영향을 미칠 수 있고, k가 크면 다수의 데이터가 판단에 참여해 안정적인 판단을 할 수는 있으나 가까이 있는 것과 멀리 있는 것이 같은 정도로 의사결정에 참여하는 단점을 가지고 있다.

1 kNN(k Nearest Neighbor)

- kNN은 거리에 의해 유사도를 측정하기 때문에 모든 입력변수는 양적변수이고 Scale이 같아야 한다.
- 자료 중 범주형 자료가 있다면 one-hot-encoding 변환을 해야 한다.
- 아래는 변수의 스케일을 통일하는 Standardization과 one-hot-encoding에 대한 설명이다.

Z Score Standardization

$$z_{ij} = \frac{x_{ij} - \bar{x}}{\bar{s}_i}$$

Minimax Standardization

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

Label Encoding

Food Name	Categorical #	Calories
Apple	1	95
Chicken	2	231
Broccoli	3	50

→

One Hot Encoding

Apple	Chicken	Broccoli	Calories
1	0	0	95
0	1	0	231
0	0	1	50

1 kNN(k Nearest Neighbor)

- 임의의 k에 대해 테스트 데이터셋에서 아래와 같은 정오표를 만들면
  분류 정확도, 오분류율, Precision, Recall과 같은 정확도를 구할 수 있다.
- k는 정확도와 Precision, Recall이 동시에 높은 값을 선택하는 것이 좋다.
- 실무적으로는 $F_1 = \frac{2(Precision*Recall)}{Precision+Recall}$ 의 값이 높은 모형을 선택하는 기준을 많이 사용한다.

		Predicted	
Actual		yes	no
	yes	✓true positive	false negative
	no	false positive	✓true negative

false negative: mis-judge to negative
제 2종 오류

false positive: mis-judge to positive
제 1종 오류

$$\text{Accuracy} = \frac{tp + tn}{tp + tn + fp + fn}$$

$$\text{Precision} = \frac{tp}{tp + fp}$$

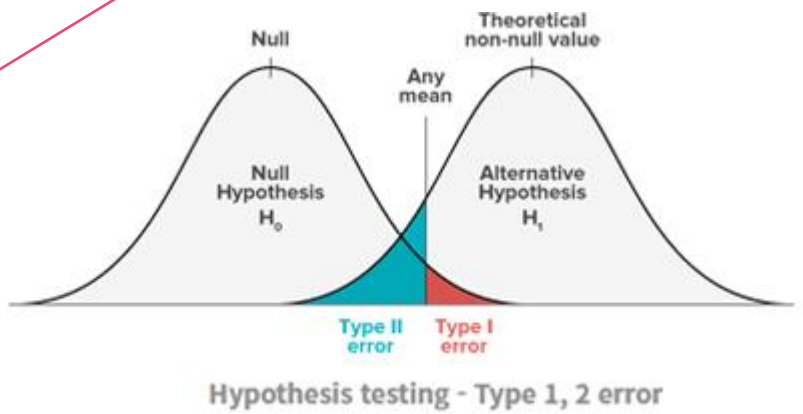
$$\text{Recall} = \frac{tp}{tp + fn}$$

1 kNN(k Nearest Neighbor)

- 참고: Precision과 Recall의 Trade off 관계

false negative: mis-judge to negative
제 2종 오류

		Predicted	
Actual		yes	no
	yes	✓true positive	false negative
	no	false positive	✓true negative

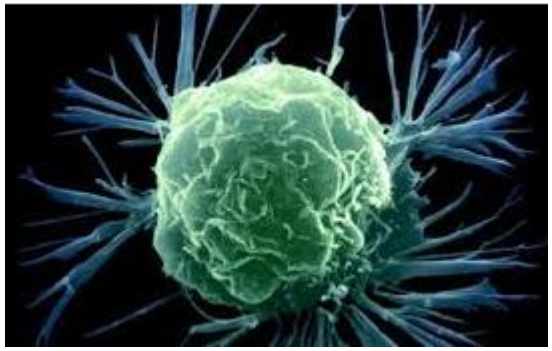


Hypothesis testing - Type 1, 2 error

false positive: mis-judge to positive
제 1종 오류

2 Example: Diagnosing breast cancer

- 아래는 환자의 세포 영상을 이용해 악성종양 여부를 구분하는 문제다.



diagnosis B: Benign, M: Malignant

Benign	Malignant	total
357	212	569

id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness mean	compactness mean	concavity_mean
87139402	B	12.32	12.39	78.85	464.1	0.1028	0.06981	0.03987
8910251	B	10.6	18.95	69.28	346.4	0.09688	0.1147	0.06387
905520	B	11.04	16.83	70.92	373.2	0.1077	0.07804	0.03046
868871	B	11.28	13.39	73	384.8	0.1164	0.1136	0.04635
9012568	B	15.19	13.21	97.65	711.8	0.07963	0.06934	0.03393
906539	B	11.57	19.04	74.2	409.7	0.08546	0.07722	0.05485
925291	B	11.51	23.93	74.52	403.5	0.09261	0.1021	0.1112
87880	M	13.81	23.75	91.56	597.8	0.1323	0.1768	0.1558
new patient	**?**	**15**	**27**	**93**	**725**	**0.1843**	**0.1677**	**0.1233**

새로운 환자가 악성종양을 가졌는지 kNN을 이용하여 알아보자.

2 Example: Diagnosing breast cancer

- t-Test를 통해 smoothness_mean Feature가 악성과 양성을 잘 구분하는지 알아 보자.
- 아래의 결과로 부터 smoothness_mean Feature는 악성여부를 잘 구별하는 특징이라는 것을 알 수 있다.

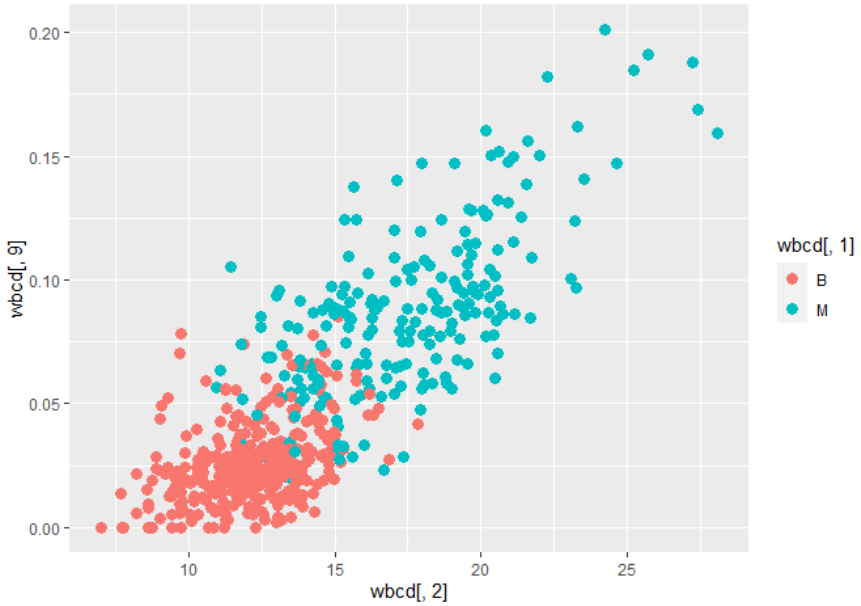
```
from scipy import stats
smoothness_B = data[data['diagnosis'] == 'B']['smoothness_mean']
smoothness_M = data[data['diagnosis'] == 'M']['smoothness_mean']
stats.ttest_ind(smoothness_B, smoothness_M)

Ttest_indResult
(statistic=-25.435821610057058, pvalue=8.465940572262422e-96)
```

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{S_1^2/n_1 + S_2^2/n_2}}$$

2 Example: Diagnosing breast cancer

- 두 개의 Feature(radius, points)를 대상으로 양성과 악성을 얼마나 구분할 수 있는지 산점도를 그린 결과다. 악성은 양성에 비해 radius와 points 값이 큰 경향이 있다.



2 Example: Diagnosing breast cancer

- 3-NN을 수행하는 R 코드다.

```
# StandardScaler
from sklearn.preprocessing import scale, minmax_scale
Z = minmax_scale(X)

from sklearn.model_selection import train_test_split
Z_train, Z_test, y_train, y_test = train_test_split(Z, y,
test_size=0.3)


knn = neighbors.KNeighborsClassifier(n_neighbors=3)
knn.fit(Z_train, y_train)
pred = knn.predict(Z_test)
```

2 Example: Diagnosing breast cancer

wbcd_test_labels	wbcd_test_pred Benign	Malignant	Row Total
Benign	61	0	61
	1.000	0.000	0.610
	0.968	0.000	
	0.610	0.000	
Malignant	2	37	39
	0.051	0.949	0.390
	0.032	1.000	
	0.020	0.370	
Column Total	63	37	100
	0.630	0.370	

정확도는 98%, Precision = 100%, Recall = 37 / 39 = 95% 로 높은데 과연 좋은 모형일까?

이 모형은 악성환자를 양성으로 오판하는 문제가 존재한다.

2 Example: Diagnosing breast cancer

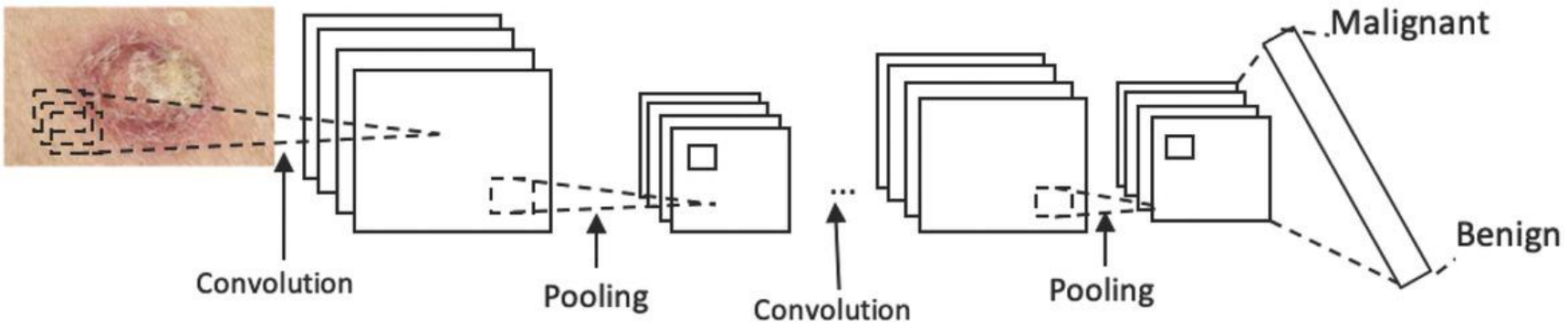
여러 가지 k 값으로 계산한 결과 …

k value	# false negatives	# false positives	Percent classified Incorrectly
1	1	3	4 percent
5	2	0	2 percent
11	3	0	3 percent
15	3	0	3 percent
21	2	0	2 percent
27	4	0	4 percent

- k=5, 21이 적합하다.
- FN을 낮추기 위해 새로운 Feature 개발을 고민할 필요가 있다.
  육안으로 FN 케이스를 왜 오분류하는지 확인하는 작업과 이를 통해 새로운 Feature가 오분류를 해결할 수 있는지를 고민해야 한다.
- 빅데이터에서는 거리계산이 너무 많아 속도가 매우 느린 단점이 존재한다.(Lazy Learning)
- 암영상 정보가 충분히 확보 가능하다면 Feature를 자동으로 추출하는 CNN 알고리즘을 고려할 수 있다.

3 Convolutional Neural Network

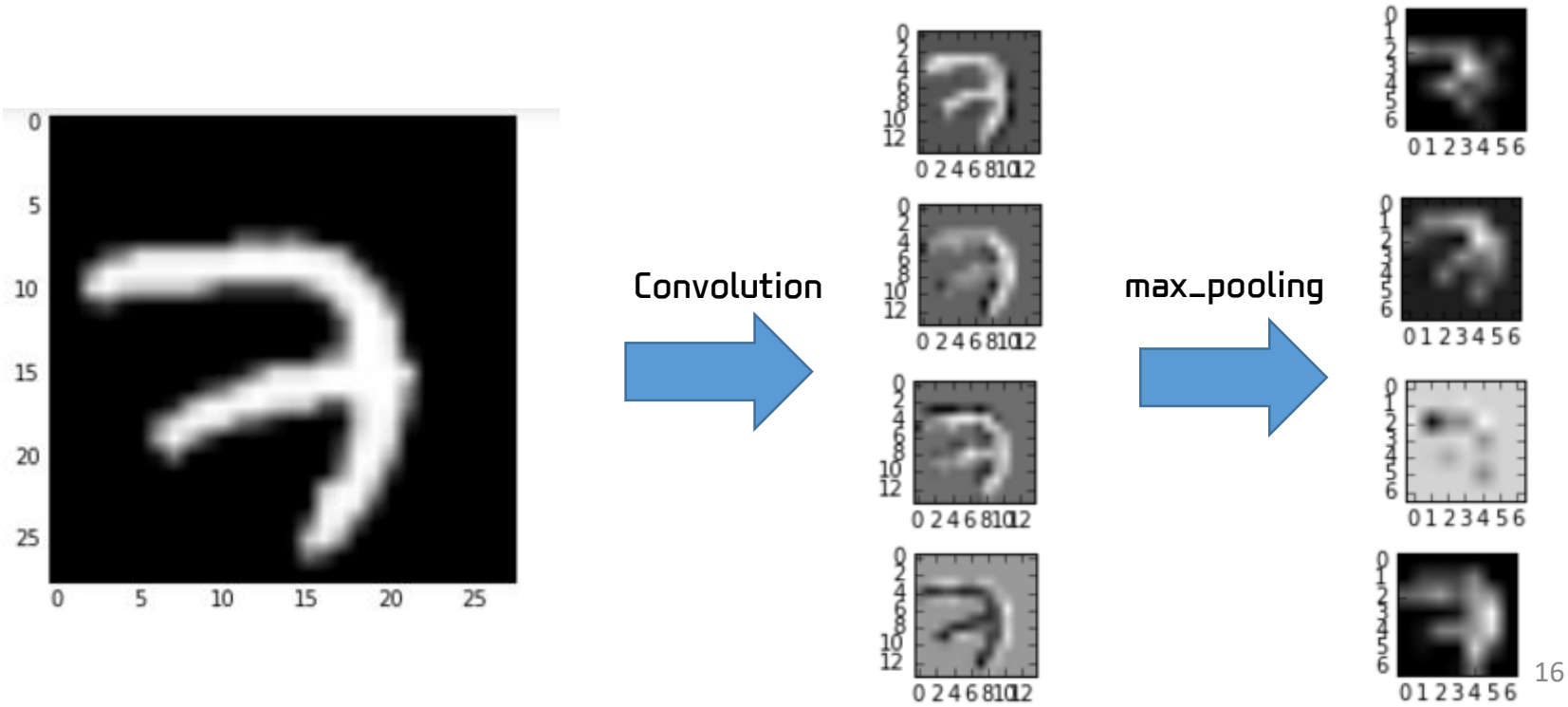
- CNN은 딥러닝의 방법론 중 하나로 CNN을 통해 Feature Extraction 하고 이를 다시 입력으로 보고 신경망 모형을 학습하는 방법론임
- 암영상 이미지 pixel을 입력으로 받아 Target인 Malignant 여부를 잘 맞출 수 있는 Feature 변수 값을 Gradient Descent Algorithm으로 찾아가는 알고리즘을 사용한다.



3 Convolutional Neural Network

- Convolution Layer, Pooling Layer를 통과하면서 이미지는 간단한 shape 정보로 축약된다. 이렇게 축약된 이미지는 원본 이미지보다 인식률이 더 좋은 결과를 줄 수 있다는 것이 CNN의 핵심이다.
- CNN으로 kNN 보다 더 좋은 결과를 얻으려면 충분히 많은 데이터가 필요하다.(빅데이터의 필요성)

숫자 7에 대한 특징 추출 예



감사합니다