

1 학습 방식의 진화

[Open in Colab](#)

1) 배경:

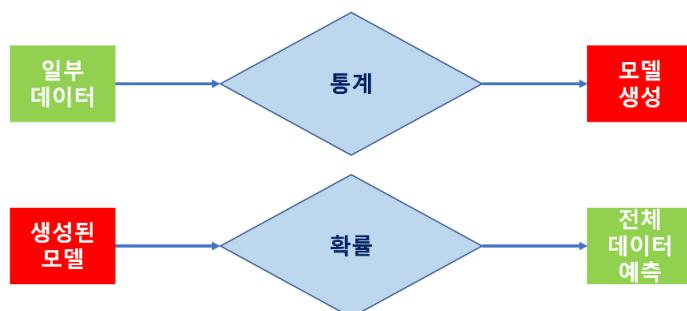
"데이터 분석에 이 울레들이 어떻게 쓰이는지 정리 해 보면.."

데이터 분석	목적	대응
데이터 시각화	데이터가 어떻게 생겼는지 알고 싶다	(1) 전체 데이터를 한눈에 확인 - 점그림, 선그림, 영역그림 - 막대그림, 등고선그림, 분포그림(히스토그램) → 통계 사용 (2) 데이터를 뿌리고 통계를 계산하기 위해 데이터 값 하나하나를 표현 → 확률/컴퓨터 사용
기술적 분석	데이터가 어떻게 생겼는지 알고 싶다	(1) 전체 데이터 특성을 몇 개의 숫자들로 확인 → 통계 사용 (2) 통계를 계산하기 위해 데이터 값 하나하나를 표현 → 확률/컴퓨터 사용
상관관계/인과관계	여러종류 데이터끼리의 관계를 알고 싶다	(1) 각 데이터를 몇 개의 숫자들로 표현 → 통계 사용 (2) 표현된 숫자들을 비교 → 확률/통계 사용
통계추론	일부 데이터로 전체를 알고 싶다	(1) 일부 데이터의 특성을 확인 → 통계 사용 (2) 기본적으로 실험 진행 및 통계치 재확인 → 컴퓨터 사용 (3) 전체 특성을 추론 → 통계 사용
알고리즘학습	전체 데이터로 미래를 알고 싶다	(1) 데이터의 관계를 수학적으로 표현 → 확률/통계/함수/컴퓨터 사용 (2) 미래를 예측한 후 정확성 확인 → 확률/통계/컴퓨터 사용
가설검정(A/B Test)	원가 진실과 가까운 의사결정을 하고 싶다	(1) 기존 데이터와 새로운 데이터 비교를 위해 숫자들로 표현 → 통계 사용 (2) 표현된 숫자들을 비교 → 확률/통계/컴퓨터 사용

2) 통계추론 vs 알고리즘학습:

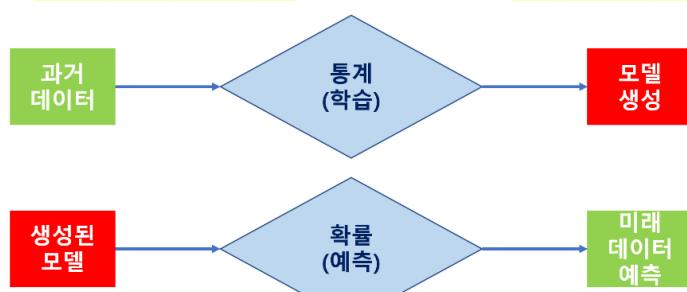
- 통계추론 기반 데이터분석 목적: 일부집단(스몰데이터) → 전체집단(빅데이터) 추론

“통계추론”은
(1) 일부 데이터로 전체특성을 확인(모델로 반영)하고
(2) 추론된 모델을 전체로 가정하여 확률적으로 전체 데이터를 예측



- 알고리즘학습 기반 데이터분석 목적: 과거의 특성으로 전체집단(빅데이터) 가정하고 일부 미래특성(스몰데이터) 추론

“알고리즘학습”은
(1) 과거 데이터를 학습하여 과거패턴을 확인(모델로 반영)하고
(2) 추론된 모델을 전체로 가정하여 확률적으로 일부 미래 데이터를 예측



3) 진화:

"수집한 데이터가 많을수록 더욱 정확하게 고객과 시장의 트렌드를 추적 할 수 있지만, 모든 데이터를 하나하나 다 들여다볼 수는 없는 노릇"

경험 및 직감 → 데이터집계(기술적분석) → 상관분석(기술적분석) → 통계추론 → 기계학습 → 딥러닝

1.1 데이터집계 및 통계량(Cross Table and Statistics)

"데이터를 분석할 때, 다양한 변수/속성/특징에 따라 데이터를 분류 하여 표형태로 정리하거나 요약(통계량)"

- 데이터집계(Cross Table): 여려개의 변수들의 관계에 따라 표형식으로 정리한 것
- 통계량(Summary Table, Statistic): 표형식의 데이터 관계성을 해석하기 위해 몇개의 숫자로 요약하여 표현하는 것

Cross tabulation Frequency Percent		What Is Your Favorite Baseball Team?			Row Totals
		Toronto Blue Jays	Boston Red Socks	New York Yankees	
Boston, MA		11	33	7	51
Row Percent		21.57%	64.71%	13.73%	34.93%
Expected Value		19.56	20.96	10.48	
Cell Chi-Square		3.75	6.92	1.16	
Computations:		Column total percent	Column total percent	Column total percent	
Expected Value		(.3836*51)	(.4110*51)	(.2055*51)	
Cell Chi-Square		(11-19.56)^2/19.56	(33-20.96)^2/20.96	(7-10.48)^2/10.48	
Montreal, Canada		23	14	9	46
Row Percent		50.00%	30.43%	19.57%	31.51%
Expected Value		17.64	18.90	9.45	
Cell Chi-Square		1.63	1.27	0.02	
In What City Do You Reside?	Computation of Cell	Column total percent	Column total percent	Column total percent	
	Expected Value	(.3836*46)	(.4110*46)	(.2055*46)	
	Cell Chi-Square	(23-17.64)^2/17.64	(14-18.90)^2/18.90	(9-9.45)^2/9.45	
Montpelier, VT		22	13	14	49
Row Percent		44.90%	26.53%	28.57%	33.56%
Expected Value		18.79	20.14	10.07	
Cell Chi-Square		0.55	2.53	1.54	
	Computation of Cell	Column total percent	Column total percent	Column total percent	
	Expected Value	(.3836*49)	(.4110*49)	(.2055*49)	
	Cell Chi-Square	(22-18.79)^2/18.79	(13-20.14)^2/20.14	(14-10.07)^2/10.07	
	Column totals	56	60	30	146
	Column Percent	38.36%	41.10%	20.55%	100.00%
	The sum of all cell Chi-Square Values = Table Chi-Square:				
	Degrees of Freedom = (# Rows -1)*(# Columns -1) = (3-1)*(3-1) = 4				
	Chi-Square Probability of Independence is not computed here, but can be looked up from a Chi-Square probability table for 19.35 and 4 df = .00067				

(<https://www.qualtrics.com/au/experience-management/research/cross-tabulation/>)

Stub: Receptionist made you feel special

	Location						
	Total	Australia	Canada	Ireland	New Zealand	UK	USA
Total Count	5512	701	523	569	399	1091	2229
Extremely satisfied	47.6%	46.4%	49.3%	47.6%	51.6%	48.1%	46.5%
Moderately satisfied	13.9%	12.8%	11.3%	15.8%	13.0%	15.1%	14.0%
Slightly satisfied	14.5%	15.5%	13.4%	13.0%	13.3%	14.0%	15.3%
Neither satisfied nor dissatisfied	14.4%	15.3%	14.9%	12.8%	13.0%	13.5%	15.0%
Slightly dissatisfied	5.5%	5.8%	7.3%	5.8%	5.0%	5.0%	5.1%
Moderately dissatisfied	2.3%	2.1%	2.1%	2.1%	2.0%	2.9%	2.1%
Extremely dissatisfied	1.9%	2.0%	1.7%	2.8%	2.0%	1.3%	2.0%

(<https://www.qualtrics.com/design-xm/cross-tabulation/>)

Age of the Respondent		Highly Dissatisfied	Dissatisfied	Partially Satisfied	Satisfied	Highly Satisfied	Total	Mean	S.D.
		25 & below	(3.5%)	(14.7%)	(18.8%)	(50.6%)	(12.4%)	(100%)	3.5353
26-35	08	9	21	90	15	143	3.6643	0.949	
36-45	05	18	16	43	05	87	3.2874	1.0444	
46-55	00	09	32	28	04	73	3.3699	0.7729	
56 & above	09	06	10	10	10	45	3.1333	1.4397	
Total		28	67	111	257	55	518		

(Source: Computed through primary data collected)

(Measurement Of Customer Satisfaction On Demographic Variables Of Banking Sector In National Capital Region - An Empirical Analysis)

- 정량적인 하위 집단의 특성을 파악하기 위해 2차원 형식으로 데이터를 요약하는 것
- 데이터집계는 다차원의 특성을 2차원으로 축약하여 해석하기 때문에 해석이 왜곡될 수 있고 자의적인 해석이 가능하도록 조작도 가능하며, 과거 특정 시점에서의 특성 스냅샷(Snapshot) 이기 때문에 사실 현재와 미래에는 맞지 않는 해석을 가능성 높음
- 최근에는 통계추론/기계학습/딥러닝을 사용하기 위한 데이터 전처리 용도 또는 보고서 작성 용도로만 사용하는 편

In [1]: # 다양한 변수/속성/특징을 갖고 있는 데이터 준비
import pandas as pd

```

df_rel = pd.read_excel(r'..\Data\ECCommerce\EC_Commerce_Dataset.xlsx', sheet_name='Raw')
target_col = ['CustomerID', 'Gender', 'MaritalStatus', 'CouponUsed', 'OrderCount']
df = df_rel[target_col].copy()
df
executed in 2.62s, finished 17:45:10 2022-07-10

```

Out[1]:

	CustomerID	Gender	MaritalStatus	CouponUsed	OrderCount
0	50001	Female	Single	1.0	1.0
1	50002	Male	Single	0.0	1.0
2	50003	Male	Single	0.0	1.0
3	50004	Male	Single	0.0	1.0
4	50005	Male	Single	1.0	1.0
...
5625	55626	Male	Married	1.0	2.0
5626	55627	Male	Married	1.0	2.0
5627	55628	Male	Married	1.0	2.0
5628	55629	Male	Married	2.0	2.0
5629	55630	Male	Married	2.0	2.0

5630 rows × 5 columns

In [2]: # 마케팅으로 쿠폰을 발행하였는데 실제 매출 증가 효과가 있을지 알고 싶음
변수들의 관계에 따라 표형식으로 정리한 교차집계 결과
pd.crosstab(index=df.CouponUsed, columns=df.OrderCount, margins=True)
executed in 44ms, finished 17:45:10 2022-07-10

Out[2]:

	OrderCount	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	9.0	10.0	11.0	12.0	13.0	14.0	15.0	16.0	All
		832	79	23	16	13	5	13	3	0	1	5	1	1	0	0	0	992
0.0	854	922	104	37	33	16	19	17	6	2	5	6	0	3	2	0	2026	
1.0	0	962	104	42	27	21	24	16	7	5	4	6	2	2	2	2	1226	
2.0	0	0	128	38	37	20	27	25	2	4	5	7	1	1	3	1	299	
3.0	0	0	0	45	31	20	30	18	10	4	5	4	5	2	1	2	177	
4.0	0	0	0	0	30	26	19	19	5	2	6	4	3	3	0	1	118	
5.0	0	0	0	0	0	19	33	22	5	3	5	3	4	6	2	0	102	
6.0	0	0	0	0	0	0	30	21	11	4	3	5	2	2	2	1	81	
7.0	0	0	0	0	0	0	0	23	7	3	0	1	1	1	2	2	40	
8.0	0	0	0	0	0	0	0	0	3	4	1	1	1	0	1	2	13	
9.0	0	0	0	0	0	0	0	0	0	0	2	2	3	1	2	1	13	
10.0	0	0	0	0	0	0	0	0	0	0	2	2	3	1	2	2	8	
11.0	0	0	0	0	0	0	0	0	0	0	2	0	1	0	2	3	8	
12.0	0	0	0	0	0	0	0	0	0	0	0	5	1	0	0	2	8	
13.0	0	0	0	0	0	0	0	0	0	0	0	0	4	1	0	1	6	
14.0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	1	4	
15.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	
16.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	
All	1686	1963	359	178	171	127	195	164	56	34	43	46	27	24	22	21	5116	

In [3]: # 통계량(비율)을 반영하여 실제 매출 증가 효과에 대한 해석하기
쿠폰을 사용하지 않는 구매자를 약 1%만 2번 이상 구매
쿠폰을 한번이라도 사용한 구매자들은 약 56%가 2번 이상 구매
쿠폰 발행은 매출 증가로 이어지는 것 같다고 추론 가능
(pd.crosstab(index=df.CouponUsed, columns=df.OrderCount, margins=True).T #
 / pd.crosstab(index=df.CouponUsed, columns=df.OrderCount).sum(axis=1)).T.iloc[:,2,:]
executed in 58ms, finished 17:45:10 2022-07-10

Out[3]:

	OrderCount	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	9.0	10.0	11.0	12.0	13.0	14.0	
		0.83871	0.079637	0.023185	0.016129	0.013105	0.005040	0.013105	0.003024	0.000000	0.001008	0.005040	0.001008	0.001008	0.000000	0.000
0.0	0.42152	0.455084	0.051333	0.018263	0.016288	0.007897	0.009378	0.008391	0.002962	0.000987	0.002468	0.002962	0.000000	0.001481	0.000	
1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	

In [4]: # 쿠폰 발행의 매출 증가 효과는 남녀간에 차이가 있을까?를 알고 싶음
세로축에 항목을 추가하여 더 상세하게 나누어서 분석 필요
(pd.crosstab(index=[df.CouponUsed, df.Gender], columns=df.OrderCount, margins=True).T #
 / pd.crosstab(index=[df.CouponUsed, df.Gender], columns=df.OrderCount, margins=True).iloc[:, :-1]).T.iloc[:, 4,:]
executed in 90ms, finished 17:45:10 2022-07-10

Out[4]:

	CouponUsed	Gender	OrderCount	1.0	2.0	3.0	4.0	
				Female	0.832447	0.069149	0.031915	0.015957
0.0				Male	0.842532	0.086039	0.017857	0.016234
				Female	0.429677	0.438710	0.045161	0.021935
1.0				Male	0.416467	0.465228	0.055156	0.015987

In [5]: # 지나치게 상세하면 각 항목별 데이터 갯수가 줄어들어 신뢰도 떨어질 수 있기 때문에 분석가 역량 중요
pd.crosstab(index=[df.CouponUsed, df.Gender, df.MaritalStatus], columns=df.OrderCount, margins=True).iloc[:, :4]

executed in 59ms, finished 17:45:10 2022-07-10

Out[5]:

		OrderCount	1.0	2.0	3.0	4.0
CouponUsed	Gender	MaritalStatus				
Female	Divorced	47	2	2	1	
	Married	162	12	9	1	
	Single	104	12	1	4	
Male	Divorced	59	6	4	1	
	Married	276	31	5	7	
	Single	184	16	2	2	

1.2 상관분석(Correlation Analysis)

"교차집계로 다양한 변수/속성은 확인이 가능하나 관계성/특징을 확인하기 어려울 경우 주로 사용하는 두 연속형 변수 사이의 상관관계를 파악하는 통계량"

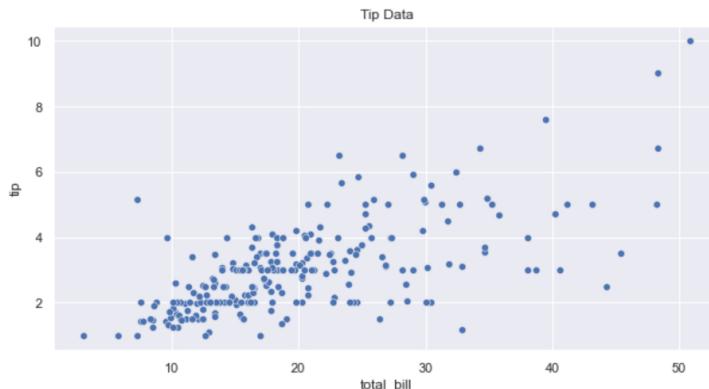
Scatter Plot: 실수 변수들의 관계성 확인

- 2차원의 데이터 표현 및 관계를 살펴보는데 많이 사용

```
sns.scatterplot(x, y, data) # X축 DataFrame 변수명, Y축 DataFrame 변수명, DataFrame
```

```
# 기본 사용
plt.figure(figsize=(10,5))
sns.scatterplot(data=tips, x='total_bill', y='tip')
plt.title('Tip Data')
plt.show()
```

executed in 134ms, finished 19:49:19 2022-04-01



• 상관계수(Correlation Coefficient):

- 통계학 관점의 선형적 상관도를 확인하여 관련성 정도를 파악하는 지표
- 간단한 분석이지만 기계학습과 딥러닝의 발전을 이해하기 위해 반드시 의미 파악 중요
- 숫자로 표현 가능한 연속형 데이터에 대해서만 분석 가능
- 1 ~ 1 사이의 크기를 가지며, 증가 방향성에 대한 것이지 인과관계 아님!

• 분석단계:

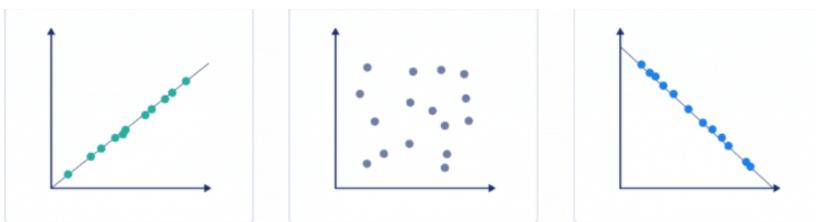
- (1) Scatter Plot으로 두 변수의 상관성을 파악
- (2) 상관계수 확인
- (3) 의사결정

Correlation coefficient value	Correlation type	Meaning
1	Perfect positive correlation	When one variable changes, the other variables change in the same direction.
0	Zero correlation	There is no relationship between the variables.
-1	Perfect negative correlation	When one variable changes, the other variables change in the opposite direction.

Perfect positive correlation

Zero correlation

Perfect negative correlation



Scribbr

```
In [6]: # 다양한 변수/속성/특징을 갖고 있는 데이터 준비
import pandas as pd
df_rel = pd.read_excel(r'..\Data\Commerce_E_Commerce_Dataset.xlsx', sheet_name='Raw')
target_col = ['CustomerID', 'Gender', 'MaritalStatus', 'CouponUsed', 'OrderCount']
df = df_rel[target_col].copy()
df
```

executed in 1.26s, finished 17:45:12 2022-07-10

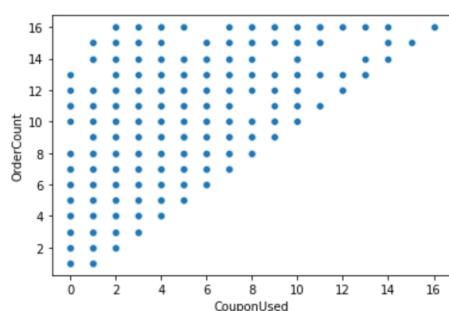
```
Out[6]:
```

	CustomerID	Gender	MaritalStatus	CouponUsed	OrderCount
0	50001	Female	Single	1.0	1.0
1	50002	Male	Single	0.0	1.0
2	50003	Male	Single	0.0	1.0
3	50004	Male	Single	0.0	1.0
4	50005	Male	Single	1.0	1.0
...
5625	55626	Male	Married	1.0	2.0
5626	55627	Male	Married	1.0	2.0
5627	55628	Male	Married	1.0	2.0
5628	55629	Male	Married	2.0	2.0
5629	55630	Male	Married	2.0	2.0

5630 rows × 5 columns

```
In [7]: # 두 변수의 상관성 정도를 그림으로 표현
import seaborn as sns
sns.scatterplot(data=df, x='CouponUsed', y='OrderCount')
executed in 1.59s, finished 17:45:13 2022-07-10
```

```
Out[7]: <AxesSubplot:xlabel='CouponUsed', ylabel='OrderCount'>
```



```
In [8]: # 두 변수의 상관성 정도를 통계량(상관계수)로 표현
df.corr()
```

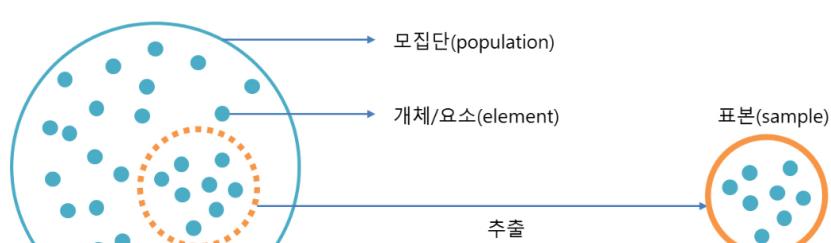
executed in 29ms, finished 17:45:13 2022-07-10

```
Out[8]:
```

	CustomerID	CouponUsed	OrderCount
CustomerID	1.000000	0.234302	0.139008
CouponUsed	0.234302	1.000000	0.745245
OrderCount	0.139008	0.745245	1.000000

1.3 통계추론(Statistical Inference)

"소수의 데이터를 통해 다수의 정보를 추정"





- 적은 비용과 시간으로 전체 시장의 변화를 파악하는데 도움 (ex. 출구조사)
- 실제 비즈니스 제도, 전략 또는 의사결정을 생성하거나 변경할 때는 통계추론으로 거시적 변화도 필요하지만 미시적 고객 변화가 더욱 중요

- 어떤 고객의 보험 가입을 승인?
- 어떤 프로모션이나 이벤트가 고객들의 반응을 높일지?
- 특정 상품이 어떤 조건에서 잘 팔리는지?
- 비용을 최소화하기 위해 재고량을 얼마나 확보?
- 제조 공정상 데이터로 불량품 검출?

In [23]:

```

import numpy as np
import pandas as pd
import scipy.stats
import matplotlib.pyplot as plt
import seaborn as sns

# 임의 데이터 생성
shape, scale = 2, 2    # mean=4, std=2*sqrt(2)
sample = np.random.gamma(shape, scale, size=100)

# 임의 데이터 시각화
plt.figure(figsize=(10,4))
sns.histplot(data=sample, kde=True)
plt.show()

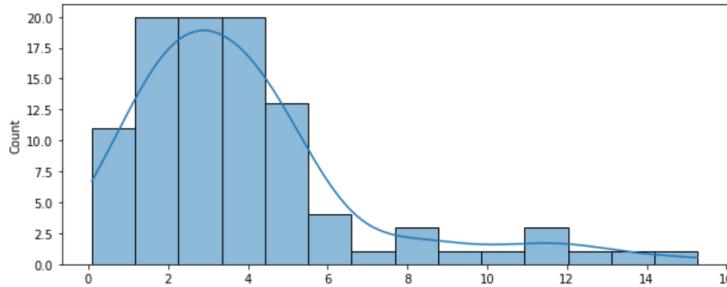
# 분포 모델 후보
dist_list = ['norm', 't', 'expon', 'chi2', 'gamma', 'beta']

# 모델과 데이터 비교 검증
result = []
for dist in dist_list:
    model = getattr(scipy.stats, dist)      # 분포의 특징 불러오기
    params = model.fit(sample)    # 데이터를 적합 및 모수 저장
    ks_stat, p_value = scipy.stats.kstest(sample, dist, params)    # 확률적 검증
    result.append([dist, p_value])    # 결과 누적 정리
result = pd.DataFrame(result, columns=['Distribution', 'p-value'])
result = result.sort_values(by='p-value', ascending=False)

# 최종 결과
display(result)
print('제일 비슷한 분포:', result.iloc[0,0])

```

executed in 414ms, finished 19:38:24 2022-07-10



C:\Users\KK\anaconda3\lib\site-packages\scipy\stats_continuous_distns.py:639: RuntimeWarning: invalid value encountered in sqrt
sk = 2*(b-a)*np.sqrt(a + b + 1) / (a + b + 2) / np.sqrt(a*b)

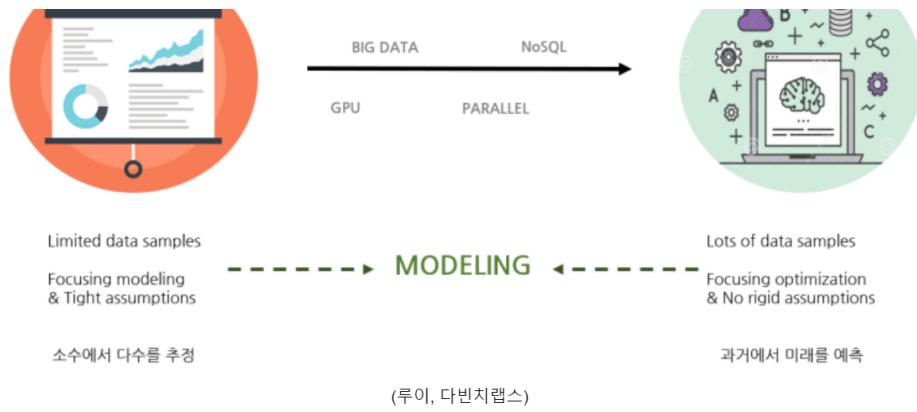
Distribution	p-value
4 gamma	0.416882
3 chi2	0.416811
5 beta	0.409182
1 t	0.404120
0 norm	0.005746
2 expon	0.002145

제일 비슷한 분포: gamma

1.4 기계학습(Machine Learning)

STATISTICS

MACHINE LEARNING



"데이터집계나 상관분석 보다 관계성을 좀 더 정량적으로 추정하기 위한 모델링"

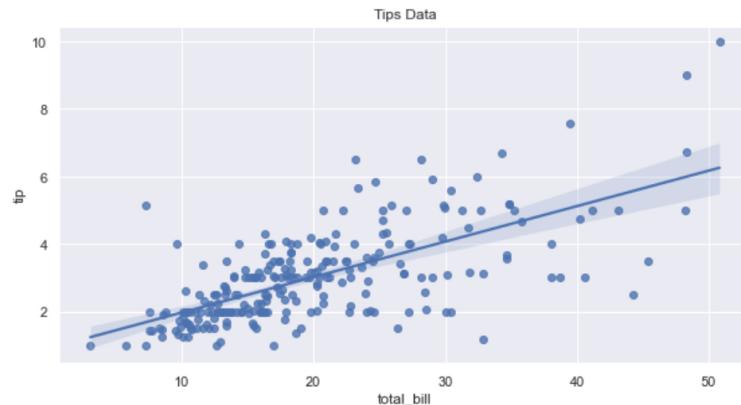
Regression Fit: 실수 변수들의 경향 확인

- scatterplot 와 lineplot 을 합쳐놓은 그래프
- lineplot 은 scatterplot 의 경향성을 예측하는 쪽으로 그어짐

```
sns.replot(x, y, data) # X는 DataFrame 변수명, Y는 DataFrame 변수명, Dataframe
```

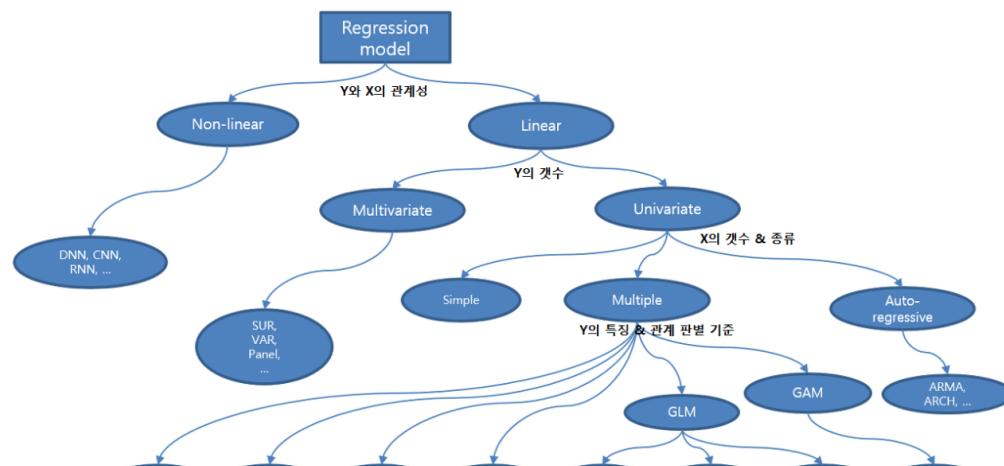
```
# 기본 사용
plt.figure(figsize=(10,5))
sns.replot(data=tips, x='total_bill', y='tip')
plt.title('Tips Data')
plt.show()
```

executed in 225ms, finished 19:49:29 2022-04-01



- 회귀분석: 변수 사이의 인과관계 알고자 시도

- 기계학습의 기본 알고리즘으로, 지도학습 예측 및 분류 문제에 사용 가능
- 예측 문제란, 기존 데이터로 생성한 모델로 새로운 미래 숫자값을 예측
- 분류 문제란, 기존 데이터로 생성한 모델로 새로운 미래 카테고리값을 예측
- 선형회귀분석: 종속변수(Y)가 연속형(숫자) 일 때
- 로지스틱회귀분석: 종속변수(Y)가 범주형(카테고리) 일 때



```
In [27]: # 다양한 변수/속성/특징을 갖고 있는 데이터 준비
import pandas as pd
df_rel = pd.read_excel(r'..\Data\EComerce\EComerce_Dataset.xlsx', sheet_name='Raw')
target_col = ['CustomerID', 'Gender', 'MaritalStatus', 'CouponUsed', 'OrderCount']
df = df_rel[target_col].copy()
df
```

executed in 1.23s, finished 19:59:04 2022-07-10

```
Out [27]:
```

	CustomerID	Gender	MaritalStatus	CouponUsed	OrderCount
0	50001	Female	Single	1.0	1.0
1	50002	Male	Single	0.0	1.0
2	50003	Male	Single	0.0	1.0
3	50004	Male	Single	0.0	1.0
4	50005	Male	Single	1.0	1.0
...
5625	55626	Male	Married	1.0	2.0
5626	55627	Male	Married	1.0	2.0
5627	55628	Male	Married	1.0	2.0
5628	55629	Male	Married	2.0	2.0
5629	55630	Male	Married	2.0	2.0

5630 rows × 5 columns

```
In [11]: # 두 변수의 상관성 정도를 통계량(상관계수)로 표현
df.corr()
```

executed in 13ms, finished 17:45:16 2022-07-10

```
Out [11]:
```

	CustomerID	CouponUsed	OrderCount
CustomerID	1.000000	0.234302	0.139008
CouponUsed	0.234302	1.000000	0.745245
OrderCount	0.139008	0.745245	1.000000

```
In [12]: # 전처리
df = df.apply(pd.to_numeric, errors = 'coerce')
df = df.fillna(method='ffill', axis=1)
```

executed in 27ms, finished 17:45:16 2022-07-10

```
In [13]: # 회귀분석 모델링
import statsmodels.api as sm
Y = df['OrderCount'].values
X = df['CouponUsed'].values
X = sm.add_constant(X)
model_lr = sm.OLS(Y, X).fit()
model_lr.summary()
```

executed in 524ms, finished 17:45:16 2022-07-10

Out [13]: OLS Regression Results

Dep. Variable:	y	R-squared:	0.019			
Model:	OLS	Adj. R-squared:	0.019			
Method:	Least Squares	F-statistic:	111.7			
Date:	Sun, 10 Jul 2022	Prob (F-statistic):	7.13e-26			
Time:	17:45:16	Log-Likelihood:	-13974.			
No. Observations:	5630	AIC:	2.795e+04			
Df Residuals:	5628	BIC:	2.796e+04			
Df Model:	1					
Covariance Type:	nonrobust					
	coef	std err	t	P> t	[0.025	0.975]
const	2.8923	0.040	73.218	0.000	2.815	2.970
x1	3.734e-05	3.53e-06	10.570	0.000	3.04e-05	4.43e-05
Omnibus:	2359.614	Durbin-Watson:	1.933			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	9065.198			
Skew:	2.123	Prob(JB):	0.00			
Kurtosis:	7.541	Cond. No.	1.14e+04			

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

[2] The condition number is large, 1.14e+04. This might indicate that there are strong multicollinearity or other numerical problems.

```
In [14]: # 예측하기
model_lr.predict(np.array([[1, 5]]))
```

Out[14]: array([2.8924745])

1.5 딥러닝(Deep Learning)

"변수들의 관계성을 현실적으로 추정하기 위한 고도화 모델링"

"어떤 문제가 나와도 당황하지 않고 시험을 잘 보기 위해서는.."

1. 작년도 기출문제(스몰데이터)가 도움이 될 수 있지만 문제의 특성과 범위가 매년 달라질 수 있으니 과거 20년치처럼 많은 기출문제(빅데이터)를 학습하는 것이 좋을 수 있습니다.

2. 많은 기출문제를 풀어서 단순히 정답을 통째로 학습(데이터집계 or 통계추론)하는 것도 좋을 수 있지만, 적은 기출문제라도 풀이 방법과 과정을 학습(기계학습)하면 새로운 문제도 잘 풀어 시험을 잘 볼 수 있습니다.

Human Learning



Machine Learning



"데이터집계나 통계추론으로 분석하기 까다로운 방대하고 복잡한 데이터를 효과적으로 분석하여, 빅데이터 속에 숨겨진 인간이 발견하기 힘든 다양한 고객과 시장의 변화 패턴 포착"

- 인간의 암기력(빅데이터)은 물론이고 학습방식(통계추론/기계학습/딥러닝)을 모방한 수리통계와 컴퓨터의 융합으로 개발된 알고리즘 기술을 사용하여 시험문제 해결

인간/엑셀 알고리즘 기술

인간/엑셀	알고리즘 기술
스몰데이터	빅데이터
느린정리	빠른정리
비정확한정답	정확한정답

- 인간이나 데이터정리 도구인 엑셀과 달리 알고리즘 기술은 고도화되어 통계추론/머신러닝 → 딥러닝 → 인공지능으로 발전중

Data Science with Machine Learning



Data Science with Deep Learning



Data Science with Artificial Intelligence



2 데이터분석에서 학습? 알고리즘이란?

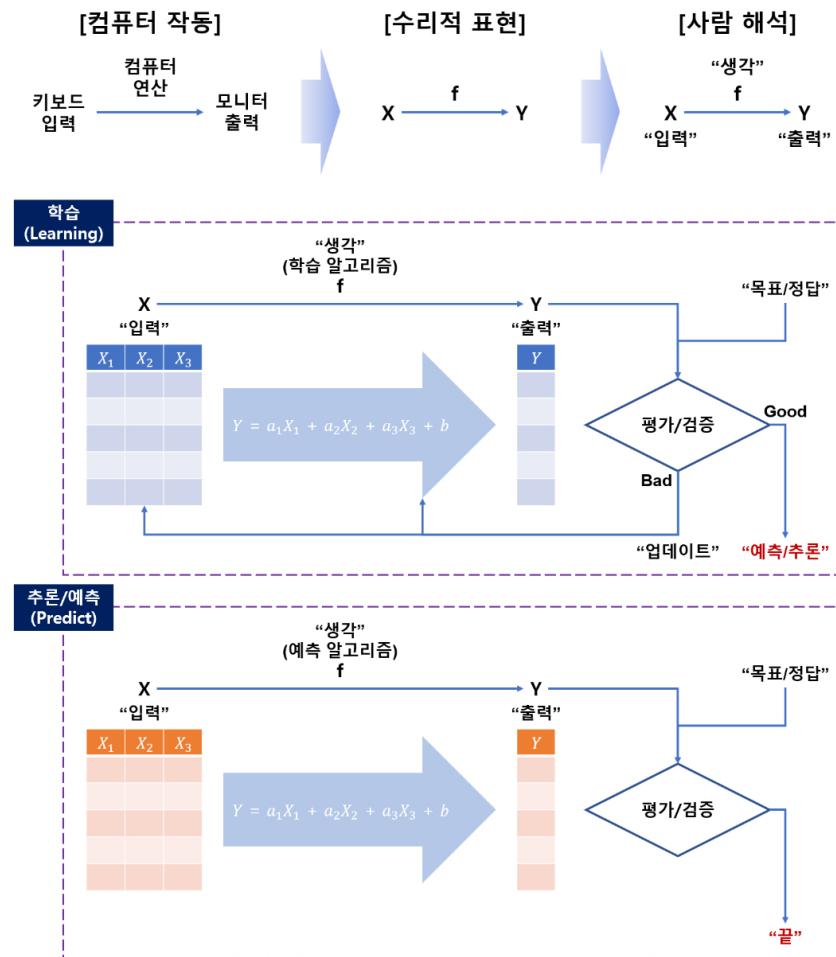
- 학습(Learning)??: 데이터분석 과정을 컴퓨터/기계 스스로가 과거와 현재를 학습하게 하고 미래를 예측하게 하는 과정

- 과거: 사람의 패턴을 모방하여 컴퓨터를 사람처럼 만들려고 노력

- 사람의 패턴이나 정답을 수학적으로 모델링(알고리즘화)
- 데이터 보다는 사람의 패턴을 수학적으로 표현하여 문제를 해결

• 현재: 사람의 학습방식을 모방하여 컴퓨터가 사람처럼 학습 하도록 노력

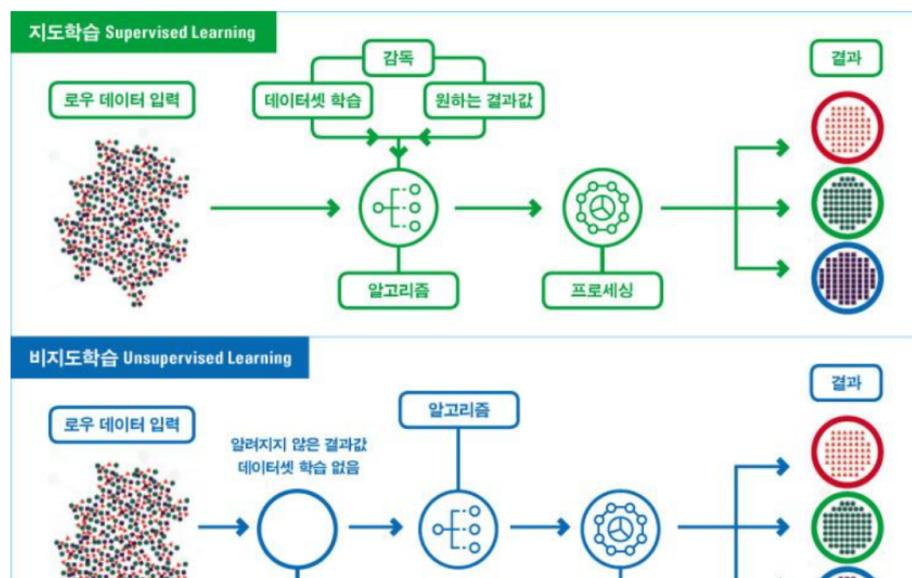
- 사람의 패턴이나 정답을 찾지 않고 뇌의 작동방식을 모델링(알고리즘화)
- 사람의 패턴은 알 필요 없이 데이터를 컴퓨터에게 지속적으로 전달(학습)하면 스스로 문제를 해결

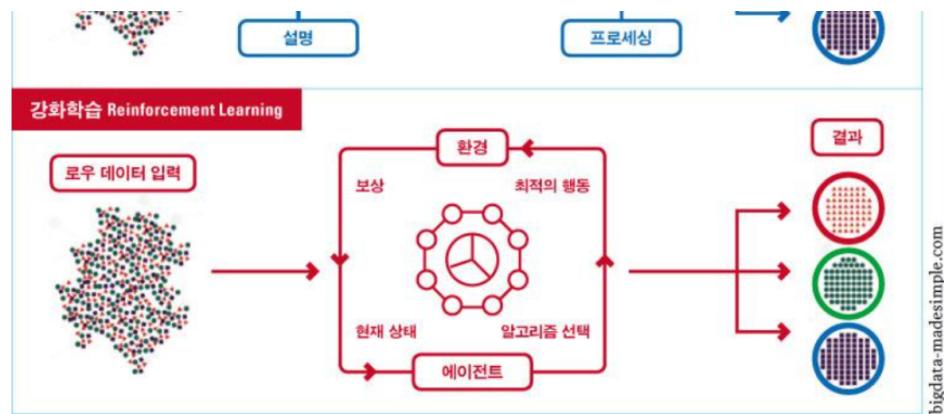


2.1 학습종류: 지도학습/비지도학습/강화학습

0) 어떤걸 풀수 있는가?

- 전체적으로 지도학습/비지도학습/강화학습이라는 3가지 종류의 알고리즘으로 해결중



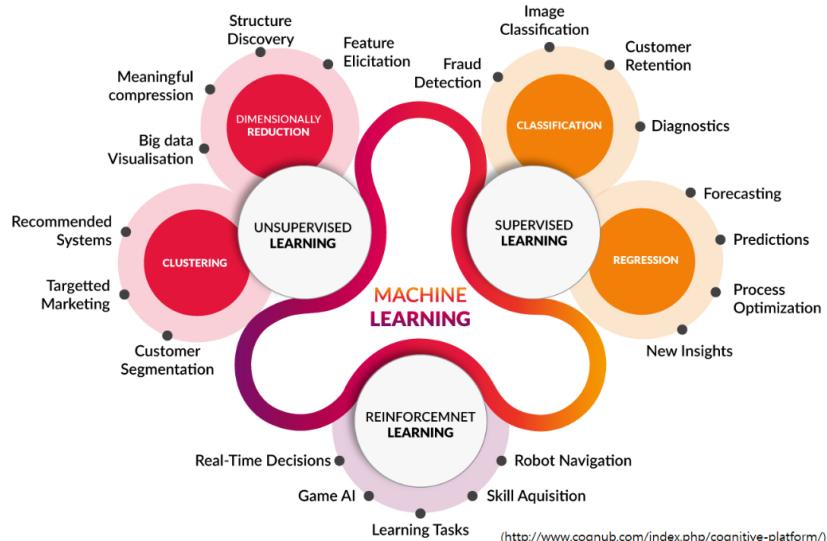


bigdata-madesimple.com

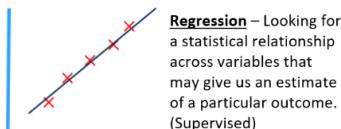
- 각 알고리즘은 인간이 해결하고 싶어하는 특정 세부문제 맞출해결 방식으로 진화중

알고리즘 종류	해결문제 종류	해결 예시
지도학습 알고리즘	예측문제	주관식 문제의 숫자형태의 정답을 해결
	분류문제	객관식 문제의 보기들 중 정답을 해결
비지도학습 알고리즘	군집문제	시험문제에 어떤 유형들이 있는지 해결
	차원변환문제	시험문제의 풀이법을 다양한 관점으로 변환
강화학습 알고리즘	-	공교육/사교육 없이 스스로 지도학습/비지도학습 문제를 해결하고 오답노트도 스스로 만들고 학습하여 성적을 계속 끌어올림

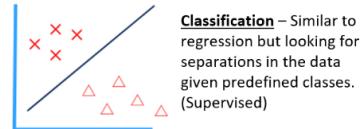
1) 예시 사례:



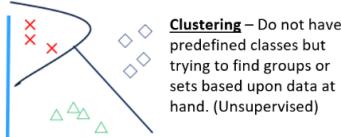
- How much is the stock of Samsung Electronics tomorrow?



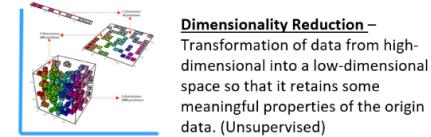
- Will Samsung Electronics' stocks rise or fall tomorrow?



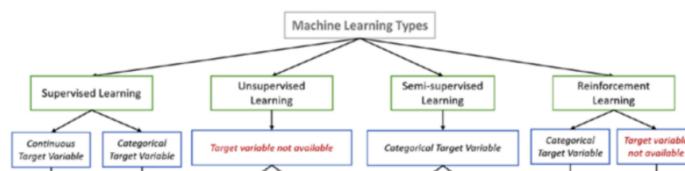
- Are Samsung Electronics and Naver similar business companies?

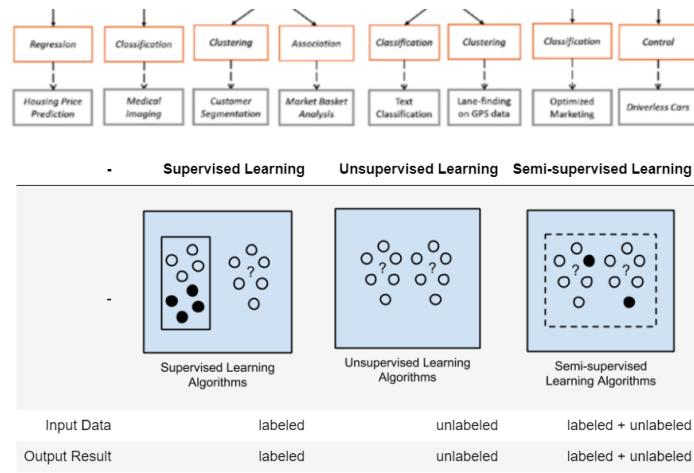


- What are the representatives among all stocks in the KOSPI?

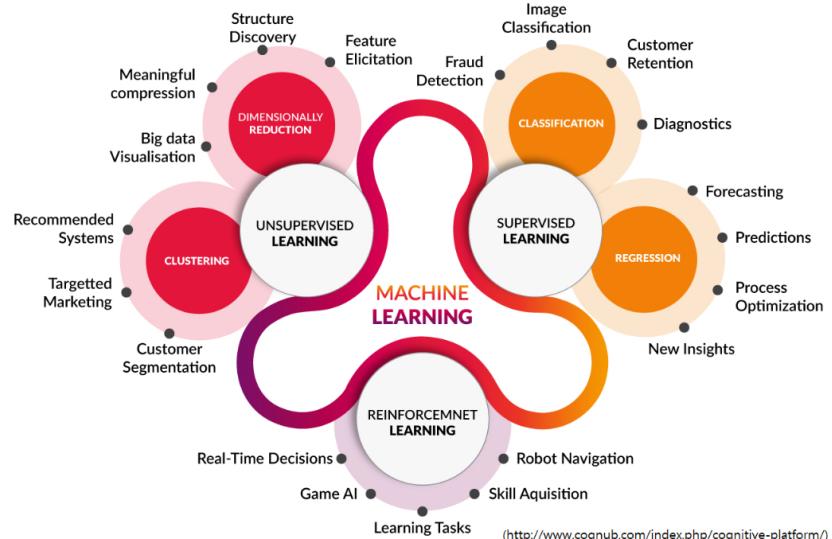


2) 기계학습의 4가지 분류:





3) 정리:



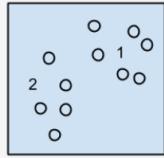
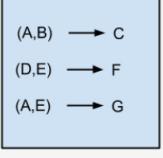
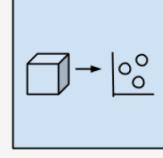
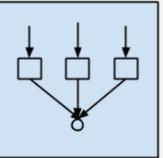
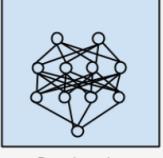
알고리즘들은 크게 3-4가지 문제들을 해결하기 위해 만들어짐
내가 풀어야 할 문제가 무엇인지 알면 분석설정과 해결은 수동적인 작업(단순)

- (1) 문제가 어디에 속하는지 → 문제정의 및 분석기획(가설설정) 가능
- (2) 알고리즘의 입력은 무엇인지 → 분석기획(데이터준비 + 전처리) 가능
- (3) 알고리즘의 출력은 무엇인지 → 분석기획(가설검정) 및 성능검증 가능

2.2 지도학습(Supervised Learning) 알고리즘

Regression Algorithms	Instance-based Algorithms	Regularization Algorithms	Decision Tree Algorithms	Bayesian Algorithms	Artificial Neural Network Algorithms
Ordinary Least Squares Regression (OLSR)	k-Nearest Neighbor (kNN)	Ridge Regression	Classification and Regression Tree (CART)	Naive Bayes	Perceptron
Linear Regression	Learning Vector Quantization (LVQ)	Least Absolute Shrinkage and Selection Operator (LASSO)	Iterative Dichotomiser 3 (ID3)	Gaussian Naive Bayes	Back-Propagation
Logistic Regression	Self-Organizing Map (SOM)	Elastic Net	C4.5 and C5.0 (different versions of a powerful approach)	Multinomial Naive Bayes	Hopfield Network
Stepwise Regression	Locally Weighted Learning (LWL)	Least-Angle Regression (LARS)	Chi-squared Automatic Interaction Detection (CHAID)	Averaged One-Dependence Estimators (AODE)	Radial Basis Function Network (RBFN)
Multivariate Adaptive Regression Splines (MARS)	-	-	Decision Stump	Bayesian Belief Network (BBN)	-
Locally Estimated Scatterplot Smoothing (LOESS)	-	-	M5	Bayesian Network (BN)	-
			Conditional Decision Trees		

2.3 비지도학습(Unsupervised Learning) 알고리즘

Clustering Algorithms	Association Rule Learning Algorithms	Dimensionality Reduction Algorithms	Ensemble Algorithms	Deep Learning Algorithms
				
Clustering Algorithms	Association Rule Learning Algorithms	Dimensional Reduction Algorithms	Ensemble Algorithms	Deep Learning Algorithms
k-Means	Apriori algorithm	Principal Component Analysis (PCA)	Boosting	Deep Boltzmann Machine (DBM)
k-Medians	Eclat algorithm	Principal Component Regression (PCR)	Bootstrapped Aggregation (Bagging)	Deep Belief Networks (DBN)
Expectation Maximisation (EM)	-	Partial Least Squares Regression (PLSR)	AdaBoost	Convolutional Neural Network (CNN)
Hierarchical Clustering	-	Sammon Mapping	Stacked Generalization (blending)	Stacked Auto-Encoders
-	-	Multidimensional Scaling (MDS)	Gradient Boosting Machines (GBM)	-
-	-	Projection Pursuit	Gradient Boosted Regression Trees (GBRT)	-
-	-	Linear Discriminant Analysis (LDA)	Random Forest	-
-	-	Mixture Discriminant Analysis (MDA)	-	-
-	-	Quadratic Discriminant Analysis (QDA)	-	-
-	-	Flexible Discriminant Analysis (FDA)	-	-