본 자료는 학습용으로 제공되는 것입니다.

다른 곳에 공유 시 출처를 밝혀 주시기 바랍니다. *^^*

재생목록 : https://www.youtube.com/watch?v=Y5IujU4XKwE&list=PLnp1rUgG4UVaHL5KKWkJxpT02X7Fh6ggv

youtube 채널 : eduatoz
작성자 : 윤소영


R의 array, matrix

R에서의 array, vector, matrix

- array
  - 동일한 데이터 타입을 갖는 데이터의 다차원 구조 (1차원 이상일 수 있음)
  - 동일한 데이터 타입이란? integer, float, logical, character 등의 같은 데이터 타입으로 된 것
  - 1차원 array 대신 vector 를 주로 사용
  - 2차원 array 대신 matrix 을 주로 사용

- vector
  - 동일한 데이터 타입을 갖는 데이터의 1차원 구조 (각 데이터 접근을 위치 번호로 함)

- matrix
  - 동일한 데이터 타입을 갖는 데이터의 2차원 구조 (각 데이터 접근을 행, 열번호로 함)

R에서의 1차원 array 구조

- 1차원 array
  - 1개의 행 또는 열로 볼 수 있지만, 그 보다는 여러 개의 element를 갖는 1차원 구조로 이해하는 것이 좋음
  - 1개 행 또는 1개 열을 갖는 2차원과 혼란이 있을 수 있기 때문에 조심해야 함

1차원 array
(4개 element)


```
a1 <- array(1:4, c(4))
a1[3]  # 3
```

array 요소 개수

접근할 요소의 위치 번호

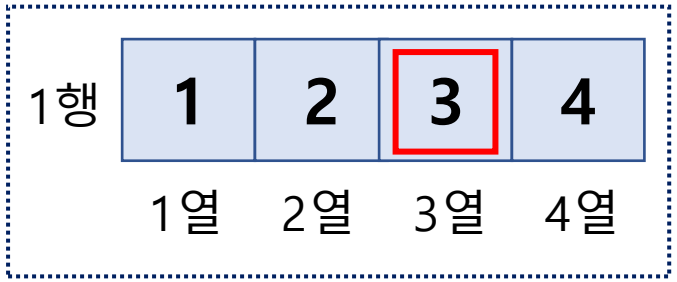
vector

```
v <- 1:4
v[3]  # 3
```

접근할 요소의 위치 번호

R에서의 2차원 array 구조

- 2차원 array
  - 1이상의 행, 열을 가지고 있으며 구조는 항상 (행, 열)의 순서로 표기함

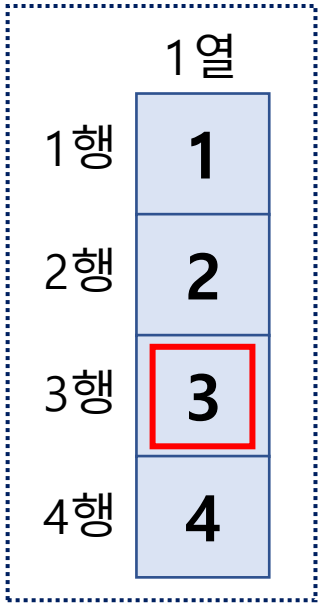


2차원 array
(1행 4열)

```
a21r <- array(1:4, c(1, 4))
a21r[1, 3]
```

array 행,열 개수

접근할 요소의 행,열 번호



2차원 array
(4행 1열)

```
a21c <- array(1:4, c(4, 1))
a21c[3, 1]
```

array 행,열 개수

접근할 요소의 행,열 번호

2차원 array
(2행 4열)

```
a2 <- array(1:8, c(2, 4))
a2[1, 3]
```

array 행,열 개수

접근할 요소의 행,열 번호

R에서의 matrix 구조

- matrix(data, nrow=1, ncol=1, byrow=FALSE)
  - 동일한 데이터 타입을 갖는 데이터의 2차원 구조로 기본은 열 우선 채우기이지만 행 우선 채우기 가능
  - nrow, ncol을 생략가능 (생략 시 자동계산, 모두 생략 시 nrow=데이터 개수, ncol=1 의 2차원 구조)

	1열	2열	3열	4열
1행	1	3	5	7
2행	2	4	6	8

```
m2a<- matrix(1:8, nrow=2)
m2a[1, 3]  # 5
```

nrow : matrix에서 행의 수 (열의 수 자동계산)

```
m2b <- matrix(1:8, ncol=4)
m2b[1, 3]  # 5
```

ncol : matrix에서 열의 수 (행의 수 자동계산)

접근할 데이터의 [행, 열] 번호

	1열	2열	3열	4열
1행	1	2	3	4
2행	5	6	7	8

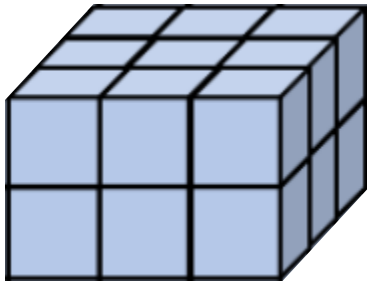
```
m2c <- matrix(1:8, nrow=2, byrow=TRUE)
m2c[1, 3]  # 3
```

matrix 채우기를 row 우선으로 함

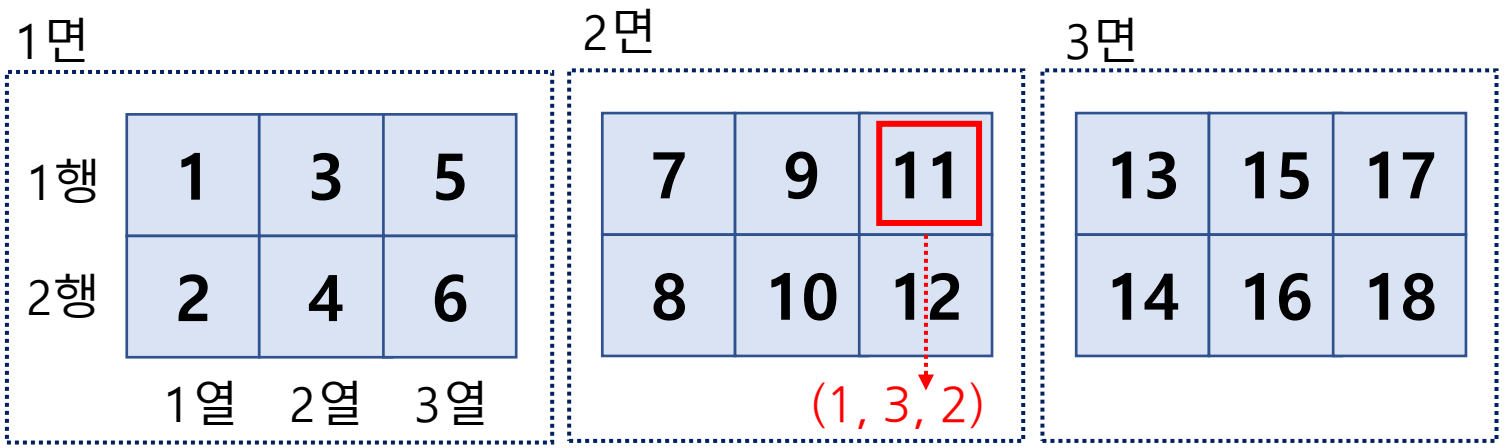
접근할 데이터의 [행, 열] 번호

R에서의 3차원 array 구조

- 3차원 array
  - 1이상의 행, 열, 면을 가지고 있으며 구조는 항상 (행, 열, 면)의 순서로 표기함



3차원 array
2행 3열 3면




```
a3 <- array(1:18, c(2,3,3))
a3[1, 3, 2]  # 11
```

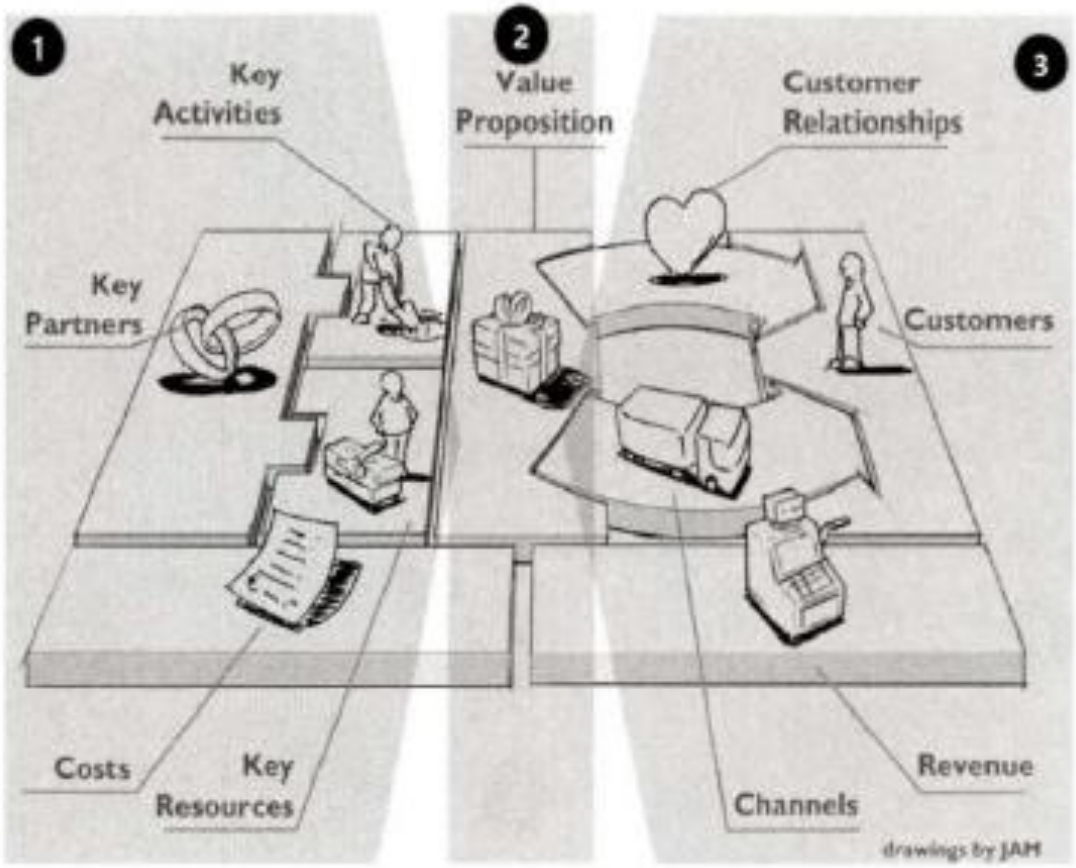
array에서 (행, 열, 면)의 수

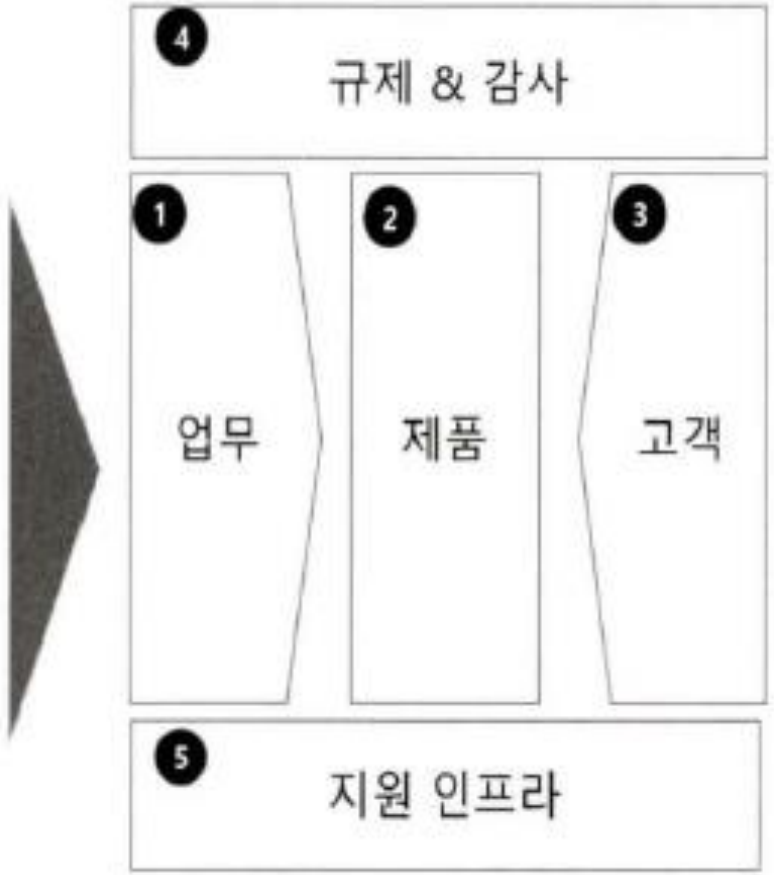
접근할 데이터의 [행, 열, 면] 번호

비즈니스 모델 캠퍼스
데이터 단위, 데이터 유형

비즈니스 모델 캔버스

- 28회 빈칸 채우기

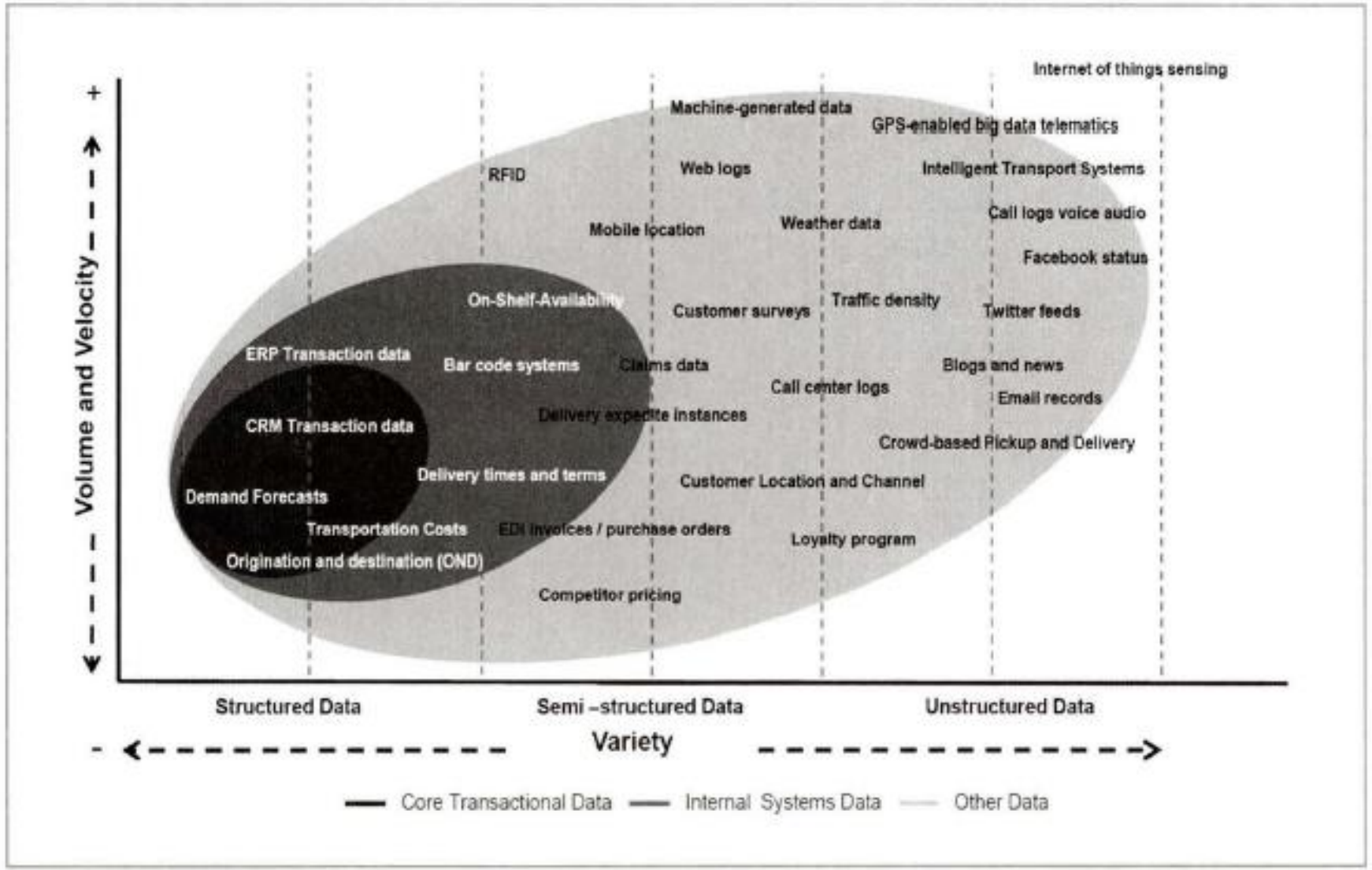




데이터 단위

단위	접두어	10^n	2^n	
KB	Kilo	10^3	2^{10}	
MB	Mega	10^6	2^{20}	
GB	Giga	10^9	2^{30}	
TB	Tera	10^{12}	2^{40}	
PB	Peta	10^{15}	2^{50}	
EB	Exa	10^{18}	2^{60}	six(6)를 뜻하는 그리스어 'ηεξα(이엑사)'에서 왔다
ZB	Zetta	10^{21}	2^{70}	7을 의미하는 이탈리아어 setta 에서 왔다
YB	Yotta	10^{24}	2^{80}	8을 의미하는 이탈리아어 otto 에서 왔다

데이터 유형



출처 : https://blog.naver.com/PostView.naver?blogId=jdhpuppy&logNo=221343689820

데이터의 분류, 머신러닝의 분류

데이터 특성에 따른 분류

범주형 categorical

- 고유한 값이나 범주 수가 제한된 데이터 (문자, 숫자 등으로 표현)
- 순서가 없는 명목형, 순서가 존재하는 순위형이 있음

명목형 nominal

- 단순히 대상의 특성을 분류하는 것, 숫자로 바꾸어도 값의 크고 작음 없음
- 성별(F, M), 혈액형(A, B, O, AB), 부서(기획, 마케팅,...) 등

순위형 ordinal

- 항목들 간에 서열이나 순위가 존재하는 데이터 (양적 비교 불가)
- 학점(A, B, C, D, F), 메달(금, 은, 동), 만족도(1, 2, 3, 4, 5) 등

수치형 numeric

- 숫자를 사용하여 표현된 데이터
- 정수로 표현되는 이산형과 실수로 표현되는 연속형이 있음

이산형 discrete

- 셀 수 있고 특정 값과 값 사이에 다른 값이 존재하지 못하는 정수 데이터
- 주사위 눈의 수, 자녀 수, 사고 횟수, 제품의 개수 등

연속형 continuous

- 데이터 값과 값 사이에 무수히 많은 다른 값들이 존재 하는 실수 데이터
- 키, 몸무게, 기온 등 (등간, 비율 척도 포함)

기계학습(Machine Learning)의 분류

지도학습 Supervised

- X를 사용해 Y를 예측할 때, 학습 데이터에 X, Y 데이터가 모두 존재하는 학습
- X를 독립변수, Y를 종속변수라고 하며, Y에는 실제 값, 예측 값이 존재함
- 회귀(Regression), 분류(Classification) 모델이 있음

회귀 Regression

- 예측 값이 실제 값보다 크거나 작거나 사이 값일 수 있음 (연속형 결과)
- 부모 키를 사용해 딸의 키 예측, 판매량 예측, 집값 예측

분류 Classification

- 예측 값이 실제 값에서 주어진 데이터 범주(종류)로 제한됨 (범주형 결과)
- 화물의 정시 도착 여부 예측, 생존 여부 예측, 품종 예측, 이미지 숫자 예측

비지도학습 Unsupervised

- 학습 데이터에 X에 대한 데이터만 존재하는 학습
- 군집(Clustering), 연관(Association) 모델이 있음

군집 Clustering

- 데이터를 특성에 따라 구분되는 몇 개의 그룹으로 나누는 학습
- 고객을 3개 그룹으로 나눔 (그룹내 서로 유사한 특성, 범주형 결과)

연관 Association

- 항목들 간의 '조건-결과' 식으로 표현되는 유용한 패턴을 발견하는 것
- 삼겹살→상추, 빵→우유 (지지도, 신뢰도, 향상도 등으로 연속형 결과)

기계학습(Machine Learning)에서의 데이터

- 예측 또는 그룹나누기, 패턴 규칙을 발견하는 방법을 수식화 하여 모델 또는 알고리즘이라고 부름
- 회귀, 분류, 군집, 연관분석 각각에 여러 가지 모델들이 존재함
- 모델에 데이터를 넣어 학습(=분석)을 진행하는 것을 모델링이라고 함
- 모델링 과정에서 데이터들은 모델의 종류에 따라 다양한 수학적 연산을 통해 결과를 도출함
- 따라서, Machine Learning에 사용되는 모든 데이터는 수치형(=연산가능한) 이어야 함

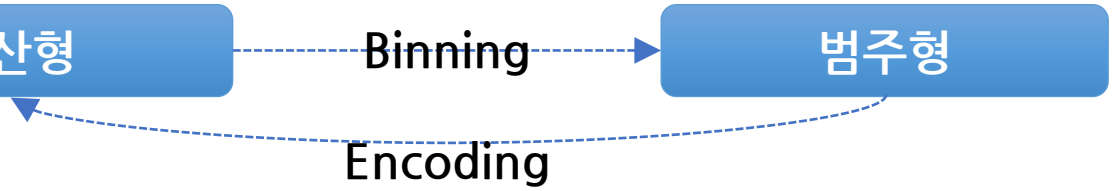
인코딩 Encoding	▪ 사용자가 입력한 문자나 기호들을 컴퓨터가 이용할 수 있는 것으로 바꾸는 것 ▪ 범주형 데이터가 문자/문자열로 표현된 경우, 이산형 수치로 바꿔 사용
구간화 Binning	▪ 데이터의 종류가 많은 경우, 구간으로 나누어 묶어 사용 (학습에 도움이 될 수 있음) ▪ 예) 날짜시간 ➔ 월별, 일별, 요일별, 시간별, 나이 ➔ 연령별 등 ▪ 임의의 구간을 만들 수 있음 ➔ 성수기/비수기

데이터 분류 어떻게 하지?

- 한 가지 데이터를 하나의 형으로 한정하지 않기!

나이는 어떤 데이터로 분류하면 될까요?

나이는 매일 매시간 늘어가고 있음	1, 2, 3, 4... 이렇게 태어난 년도를 1살로 년도가 바뀔 때마다 1씩 증가함	'10대', '20대', '30대' .. 와 같은 연령대
연속형	이산형	범주형



이산형과 범주형을 같은 개념으로 이야기 하는 경우도 있음