

## 6.1 집합

확률을 수학적으로 정의하기 위해서는 집합에 대한 지식이 필요하다. 따라서 확률론 공부에 들어가기 전에 집합론의 기초를 간단하게 복습한다. 파이썬에서 집합을 사용하기 위해 필요한 `set` 자료형과 `frozenset` 자료형의 사용법도 같이 살펴본다.

### 집합과 원소

구별 가능한 객체의 모임을 **집합(set)**이라고 하고 집합에 포함된 구별 가능한 객체를 그 집합의 **원소(element)**라고 한다. 집합은 보통 알파벳 대문자를 사용하여 표시하고 원소는 알파벳 소문자로 표시한다.

원소  $x$ 와 그 원소를 포함하는 집합  $A$ 의 관계는 다음처럼 표시한다.

$$x \in A \quad (6.1.1)$$

만약 원소  $x$ 가 집합  $A$ 에 포함되지 않는다면 다음처럼 표시한다.

$$x \notin A \quad (6.1.2)$$

따라서 만약  $A = \{1, 2, 3\}$ 이면,

$$1 \in A, \quad 4 \notin A \quad (6.1.3)$$

이다.

집합을 이루는 객체가 반드시 숫자일 필요는 없다. 다음처럼 어떠한 원소도 포함할 수 있다.

$$B = \{H, T\} \quad (6.1.4)$$

$$C = \{\spadesuit, \heartsuit, \diamondsuit, \clubsuit\} \quad (6.1.5)$$

파이썬에서는 `set` 과 `frozenset` 자료형으로 집합을 나타낸다. `set` 은 내용을 변경할 수 있는 **mutable**(mutable)자료형이고 `frozenset` 은 내용을 변경할 수 없는 **immutable**(immutable)자료형이다. (리스트 자료형과 튜플 자료형의 관계와 같다.) **mutable** 자료형은 **딕셔너리(dictionary)** 자료형의 **키(key)**나 혹은 `set` 자료형의 원소가 될 수 없다. **immutable** 자료형만 **딕셔너리** 자료형의 **키**나 `set` 자료형의 원소가 될 수 있다.

In [1]:

```
A = set([1, 2, 3, 3, 2]) # 중복된 자료는 없어진다.  
A
```

Out[1]:

```
{1, 2, 3}
```

In [2]:

```
B = frozenset(['H', 'T'])  
B
```

Out[2]:

```
frozenset({'H', 'T'})
```

`set` 자료형은 `{ }` 기호를 사용하여 만들 수도 있다.

In [3]:

```
C = {"Wu2660", "Wu2661", "Wu2662", "Wu2663"}
C
```

Out[3]:

```
{'♠', '♥', '♦', '♣'}
```

In [4]:

```
type(C)
```

Out[4]:

```
set
```

## 집합의 크기

집합의 **크기(cardinality)**는 집합이 가지는 원소의 갯수를 말한다.  $|A|$  기호나 `card` 기호를 사용하여 나타낸다. 만약  $A = \{1, 2, 3\}$  이면,

$$|A| = \text{card}(A) = 3 \quad (6.1.6)$$

이다.

파이썬에서는 `len` 명령을 사용하여 집합의 원소의 갯수를 구한다.

In [5]:

```
len(A), len(B), len(C)
```

Out[5]:

```
(3, 2, 4)
```

두 실수 사이에는 항상 다른 실수가 존재하므로 다음과 같은 실수 구간 집합은 무한개의 원소를 가진 집합이다. 예를 들어 다음 집합은 0보다 크고 1보다 같거나 작은 모든 실수로 이루어진 집합이며 원소의 갯수는 무한히 많다.

$$D = \{x : 0 < x \leq 1\} \quad (6.1.7)$$

이러한 집합은 파이썬의 `set` 또는 `frozenset` 자료형으로 표현할 수 없다.

## 합집합과 교집합

두 집합의 **합집합(union)**은 각 집합의 원소를 모두 포함하는 집합을 말하고  $\cup$  기호를 사용하여 표시한다.

$$A \cup B \quad (6.1.8)$$

두 집합의 **교집합(intersection)**은 두 사건 모두에 속하는 원소로만 이루어진 집합을 말하고  $\cap$  기호를 사용하여 표시한다.

$$A \cap B \quad (6.1.9)$$

파이썬에서는 `union`, `intersection` 메서드나 `|`, `&` 연산자를 사용하여 합집합과 교집합을 구할 수 있다.

In [6]:

```
A1 = set([1, 2, 3, 4])
A2 = set([2, 4, 6])
A3 = set([1, 2, 3])
A4 = set([2, 3, 4, 5, 6])
```

In [7]:

```
A1.union(A2)
```

Out[7]:

```
{1, 2, 3, 4, 6}
```

In [8]:

```
A2 | A1
```

Out[8]:

```
{1, 2, 3, 4, 6}
```

In [9]:

```
A3.intersection(A4)
```

Out[9]:

```
{2, 3}
```

In [10]:

```
A4 & A3
```

Out[10]:

```
{2, 3}
```

## 전체집합, 부분집합, 여집합

어떤 집합의 원소 중 일부만을 포함하는 집합을 **부분집합(subset)**이라고 하고 원래의 집합을 **전체집합**이라고 한다. 집합  $A$ 가 집합  $\Omega$ 의 부분집합이면 다음처럼 표시한다.

$$A \subset \Omega \quad (6.1.10)$$

모든 집합은 자기 자신의 부분집합이다.

$$A \subset A, \text{ for all } A \quad (6.1.11)$$

원소의 크기가 더 작은 부분집합을 **진부분집합(proper subset)**이라고 한다.

파이썬에서는 두 집합이 부분집합인지를 알아보는 `issubset` 메서드가 있다. 객체가 인수의 부분집합이면 `True`를 반환한다. 등호를 포함하는 부등식 연산자로도 같은 결과를 구할 수 있다. 더 작은 쪽이 부분집합이다. 등호가 없는 부등식 연산자는 진부분집합 관계를 구한다.

In [11]:

```
A3.issubset(A1)
```

Out[11]:

True

In [12]:

```
A3 <= A1
```

Out[12]:

True

In [13]:

```
A3.issubset(A2)
```

Out[13]:

False

In [14]:

```
A3 <= A2
```

Out[14]:

False

In [15]:

```
A3 <= A3 # 모든 집합은 자기 자신의 부분집합이다.
```

Out[15]:

True

In [16]:

```
A3 < A3 # 모든 집합은 자기 자신의 진부분집합이 아니다.
```

Out[16]:

False

## 차집합과 여집합

어떤 집합  $A$ 에 속하면서 다른 집합  $B$ 에는 속하지 않는 원소로 이루어진  $A$ 의 부분집합을  $A$ 에서  $B$ 를 뺀 **차집합(difference)**이라고 하며

$$A - B \quad (6.1.12)$$

로 나타낸다.

전체집합  $\Omega$  중에서 부분집합  $A$ 에 속하지 않은 원소로만 이루어진 부분집합을 **여집합(complement)**이라고 하고 윗첨자  $C$ 를 사용하여

$$A^C \quad (6.1.13)$$

로 표시한다. 여집합  $A^C$ 은 전체집합에서 집합  $A$ 를 뺀 차집합과 같다.

$$A^C = \Omega - A \quad (6.1.14)$$

파이썬에서는 difference 메서드 혹은 - 연산자로 차집합을 구한다.

In [17]:

```
A1.difference(A2)
```

Out[17]:

```
{1, 3}
```

In [18]:

```
A1 - A2
```

Out[18]:

```
{1, 3}
```

## 공집합

아무런 원소도 포함하지 않는 집합을 **공집합(null set)**이라고 하며  $\emptyset$  기호로 나타낸다.

공집합은 모든 집합의 부분집합이 된다.

$$\emptyset \subset A, \text{ for all } A \quad (6.1.15)$$

또한 임의의 집합과 공집합의 교집합은 공집합이, 임의의 집합과 공집합의 합집합은 그 집합 자신이 된다.

$$A \cap \emptyset = \emptyset \quad (6.1.16)$$

$$A \cup \emptyset = A \quad (6.1.17)$$

여집합과 원래의 집합의 교집합은 공집합이다.

$$A \cap A^C = \emptyset \quad (6.1.18)$$

In [19]:

```
empty_set = set([])
empty_set
```

Out[19]:

```
set()
```

In [20]:

```
empty_set < A1
```

Out[20]:

True

In [21]:

```
empty_set.intersection(A1)
```

Out[21]:

set()

In [22]:

```
empty_set.union(A1)
```

Out[22]:

{1, 2, 3, 4}

## 부분집합의 수

집합  $A = \{1, 2\}$ 는 다음과 같은 4개의 부분집합을 가진다. 공집합과 자기 자신인 집합도 부분집합이라는 점에 주의한다.

$$A_1 = \emptyset \quad (6.1.19)$$

$$A_2 = \{1\} \quad (6.1.20)$$

$$A_3 = \{2\} \quad (6.1.21)$$

$$A_4 = \{1, 2\} \quad (6.1.22)$$

부분집합을 만들때 우리가 결정할 수 있는 것은 각각의 원소가 우리가 만들고자 하는 부분집합에 포함되느냐 포함되지 않느냐이다.  $N$ 개의 원소 모두에 대해 이러한 의사결정을 하면 부분집합의 갯수는 우리가 할 수 있는 의사결정의 가짓수인  $2^N$ 개가 된다.

**[정리]** 원소의 갯수가  $N$ 개인 집합은  $2^N$ 개의 부분집합을 가진다.

### 연습 문제 6.1.1

다음 집합의 부분집합을 생각한다.

$$\Omega = \{HH, HT, TH, TT\} \quad (6.1.23)$$

(1) 이 집합의 부분집합의 갯수는?

(2) 이 집합의 모든 부분집합을 `frozenset` 자료형 객체로 만들고 이 부분집합들을 원소로 가지는 하나의 `set` 객체를 만든다. 이 집합은 일종의 "부분집합의 집합"이 된다.

## 합집합과 교집합의 분배 법칙

곱셈과 덧셈의 분배 법칙은 다음과 같다.

$$a \times (b + c) = a \times b + a \times c \quad (6.1.24)$$

곱셈과 덧셈의 분배 법칙처럼 교집합과 합집합도 괄호를 풀어내는 분배법칙이 성립한다.

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C) \quad (6.1.25)$$

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C) \quad (6.1.26)$$

### 연습 문제 6.1.2

다음 세 집합  $A, B, C$ 에 대해 위에서 말한 두가지 분배법칙이 성립하는지 파이썬 코드로 증명하라.

$$A = \{1, 3, 5\} \quad (6.1.27)$$

$$B = \{1, 2, 3\} \quad (6.1.28)$$

$$C = \{2, 4, 6\} \quad (6.1.29)$$