이 자료에서는 확률과 관련된 기본 개념을 알아보겠습니다. 기존에 배웠던 개념과 용어를 다시 한번 명확하게 복습한다고 생각하세요.

확률은 컴퓨터 공학에서 SW/HW 시스템의 동작을 분석하고 예측하기 위해 사용됩니다. 예를 들어 작성한 코드의 각 조건문을 만족할 확률이 어느 정도인지 분석해서 평균 수행 속도를 계산할 수 있습니다. 또한 기계 학습에서도 많이 활용됩니다. 다양한 문자의 형태를 학습한 후, 임의의 입력 이미지가 어떤 문자를 나타내는지 파악하기 위해 가장 예측 확률이 높은 문자를 선정하는 것이 그러한 예입니다.

- Die(주사위, 단수): '주사위'라고 하면 보통 정육면체를 가리키며, 6개의 면 각각에 1~6개의 점이 있어 1~6까지의 서로 다른 수를 나타냅니다. 주사위를 한 번 던지는 것을 한 번의 **roll** 혹은 **toss**라 합니다. **게임에서는 난수를 발생**시키기 위해 주사위를 던져 각 player의 움직임을 결정하는 경우가 자주 있으며, 주사위를 사용하지 않는 경우라도 난수를 사용했다면 주사위를 사용한 것으로 보고 같은 방법으로 게임을 분석할 수 있습니다.

- N-sided die: N개의 면을 가진 주사위를 말합니다. 왼쪽 그림은 8-sided die를, 오른쪽 그림은 3-sided die를 보여줍니다. 가장 많이 사용되는 것은 6-sided die입니다. 동전은 2개의 면이 있으므로 2-sided die로 볼 수 있으며, 앞면을 head, 뒷면을 tail이라 합니다.

- Dice(주사위, 복수): 둘 이상의 주사위를 나타내는 복수형입니다.

- Fair die: **각 면이 나올 확률이 같은** 주사위를 'fair die'라 합니다. 6개의 면을 가진 fair die라면 각 면이 나올 확률이 1/6입니다. 만약 하나 이상의 면이 나올 확률이 다른 면과 다르다면 '**un**fair die'라 합니다. 이번 시간에 볼 주사위는 fair die라고 가정합니다.

- Double: 두 개의 주사위를 던져 두 주사위 모두에서 같은 면이 나왔을 때를 double이라 합니다. 왼쪽 그림은 double six인 경우를 보여줍니다. Double은 발생할 확률이 높지 않으므로 게임 중 가끔 일어나는 이벤트에 대응된다고 볼 수 있습니다. 예를 들어 1/6의 확률로 어떤 아이템을 습득할 수 있다면 주사위를 던져 double이 나왔을 때 아이템을 습득한다고 보고 게임을 분석할 수 있습니다.

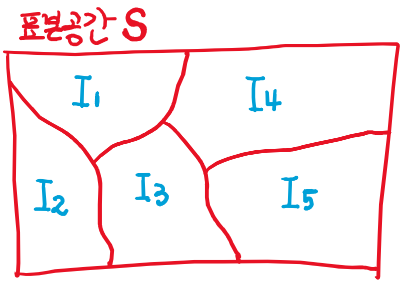
앞으로는 6-sided fair die를 두 번 던졌을 때 나올 수 있는 경우를 예로 들어 여러 가지 확률 관련 개념을 알아보겠습니다. 혹은 6-sided fair dice 2개를 던졌다고 볼 수도 있습니다. 아래 표 1과 같이 36가지의 서로 다른 경우가 나올 수 있습니다.

<표 1. Fair 6-sided die를 두 번 던졌을 때 나올 수 있는 경우>

| |
|---|
| 1&1, 1&2, 1&3, 1&4, 1&5, 1&6, 2&1, 2&2, 2&3, 2&4, 2&5, 2&6, |
| 3&1, 3&2, 3&3, 3&4, 3&5, 3&6, 4&1, 4&2, 4&3, 4&4, 4&5, 4&6, |
| 5&1, 5&2, 5&3, 5&4, 5&5, 5&6, 6&1, 6&2, 6&3, 6&4, 6&5, 6&6 |

- 표본 공간(Sample space) $S$: 어떤 시행에서 일어날 수 있는 **모든 가능한 사건들의 모임**입니다. 6-sided die를 두 번 던졌을 때는 표 1의 36가지 경우가 표본 공간 $S$를 구성합니다.

- $I$가 표본 공간 $S$에 속한 사건 $e$들의 부분집합이라면, **$I$가 발생할 확률(probability)** $P(I) = \frac{|e \in I|}{|e \in S|}$ 입니다. 예를 들어 표 1에서 첫 번째로 나온 숫자가 1일 확률을 계산해 보겠습니다. 첫 번째 숫자가 1인 경우는 다음과 같이 6가지입니다: $I$ = {1&1, 1&2, 1&3, 1&4, 1&5, 1&6}. 따라서 6-sided die를 두 번 던졌을 때 첫 번째 숫자가 1일 확률 $P(I) = \frac{6}{36} = \frac{1}{6}$ 입니다. 단 확률을 구하는 식 $P(I) = \frac{|e \in I|}{|e \in S|}$는 **표본 공간 $S$를 구성하는 각 사건이 발생할 확률이 모두 같은 경우에 유효**합니다. 만약 각 사건의 발생 확률이 다르다면 사건에 따라 다른 가중치(weight)를 부여해 확률을 계산해야 할 것입니다.




- 확률의 정의에 따라 표본 공간 $S$에 속한 **모든 사건이 발생할 확률의 합은 1**이어야 합니다. 즉 $P(S)$ = 1이어야 합니다. 표 1에서도 $P(S) = \frac{36}{36} = 1$ 임을 볼 수 있습니다.

이 사실을 기억하면 확률 계산 시 오류가 없는지 검증하는 데 활용할 수 있습니다. 예를 들어 위 그림과 같이 표본 공간 $S$를 서로 배타적인(exclusive, 겹치지 않는) 사건 $I_1$ ~ $I_5$로 나눈 후 이들 각각의 확률 $P(I_i)$를 계산했다면 계산한 확률의 합 $\sum_{i=1}^{5} P(I_i)$ = 1 임을 확인하는 것도 $P(I_i)$ 각각의 계산이 올바른지 검증하는 하나의 방법입니다.

[Q] 2-sided dice 2개를 던졌을 때 두 주사위 모두 1이 나올 확률은?

1/36
1/4
1/9
1/2
1/16

※ 이 자료에서 [Q]로 제시된 문제는 학습 후 풀이하는 온라인 퀴즈에 그대로 나오니 학습하면서 그때그때 문제를 풀어 두세요. 온라인 퀴즈에서 보기의 순서는 바뀔 수 있으니 유의하세요. (예: 보기 1이 보기 2가 되고, 보기 2가 보기 1이 될 수 있음)

- 두 사건 **$A$ 또는 $B$가 일어날 확률** $P(A \cup B) = P(A) + P(B) - P(A \cap B)$ 입니다. 예를 들어 **표** 1에서 사건 $A$는 첫 번째 숫자가 1인 경우를 나타내고, 사건 $B$는 두 번째 숫자가 1인 경우를 나타낸다고 가정하겠습니다. $P(A) = \frac{6}{36}$ 입니다. 사건 $A$는 6가지 경우 {1&1, 1&2, 1&3, 1&4, 1&5, 1&6}으로 구성되기 때문입니다. 마찬가지로 $P(B) = \frac{6}{36}$ 입니다. 사건 $B$도 6가지 경우 {1&1, 2&1, 3&1, 4&1, 5&1, 6&1}로 구성되기 때문입니다. $P(A \cap B) = \frac{1}{36}$ 입니다. $A \cap B$ 에는 한 가지 경우 {1&1}만 해당하기 때문입니다. 따라서 $P(A \cup B) = P(A) + P(B) - P(A \cap B) = \frac{6}{36} + \frac{6}{36} - \frac{1}{36} = \frac{11}{36}$ 입니다. 이 값은 $A \cup B$에 속하는 서로 다른 경우를 세어 봄으로써 검증할 수 있습니다. $A \cup B$ = {1&1, 1&2, 1&3, 1&4, 1&5, 1&6, 2&1, 3&1, 4&1, 5&1, 6&1}, 총 11가지이므로 위 방식으로 계산한 확률이 맞음을 알 수 있습니다.

- 두 사건 $A$와 $B$가 서로 배타적(exclusive, 겹치는 경우 없음)이라면 $P(A \cap B) = 0$입니다. 이럴 때 두 사건 $A$ 또는 $B$가 일어날 확률 $P(A \cup B) = P(A) + P(B) - P(A \cap B) = P(A) + P(B) - 0 = P(A) + P(B)$ 입니다. 즉 **서로 배타적인 두 사건이 함께 일어날 확률을 구할 때는 두 사건의 확률을 더하면** 됩니다. 예를 들어 **표** 1에서 사건 $A$는 첫 번째 숫자가 1인 6가지 경우 {1&1, 1&2, 1&3, 1&4, 1&5, 1&6}를 나타내고 사건 $C$는 첫 번째 숫자가 2인 6가지 경우 {2&1, 2&2, 2&3, 2&4, 2&5, 2&6}를 나타낸다고 하겠습니다. 첫 번째 숫자가 1이면서 동시에 2일 수는 없으므로 두 사건은 서로 배타적입니다. 따라서 $P(A \cup C) = P(A) + P(C) = \frac{6}{36} + \frac{6}{36} = \frac{12}{36}$ 입니다.

[Q] 6-sided dice를 두 번 던졌다. 사건 A는 3이 나오고 이어서 짝수가 나온 경우를 말하며, 사건 B는 5가 나오고 이어서 짝수가 나온 경우를 말한다. P(A) = 3/36 = 1/12, P(B) = 3/36 = 1/12이다. A 또는 B가 발생할 확률 P(A∪B)는 무엇인가?

1/2
1/3
1/6
1/12
5/12

- 조건부 확률(conditional probability) $P(B|A) = \frac{P(A \cap B)}{P(A)}$ 는 사건 $A$가 이미 일어난 상황에서(즉, 사건 $A$가 발생했다는 조건을 만족할 때) 사건 $B$가 일어날 확률입니다. 예를 들어 **표** 1에서 사건 $A$는 첫 번째 숫자가 1인 6가지 경우 {1&1, 1&2, 1&3, 1&4, 1&5, 1&6}를 나타내고 사건 $D$는 두 번째 숫자가 ≥5인 12가지 경우 {1&5, 1&6, 2&5, 2&6, 3&5, 3&6, 4&5, 4&6, 5&5, 5&6, 6&5, 6&6}를 나타낸다고 하겠습니다. $A \cap D$ 가 2가지 경우 {1&5, 1&6}를 포함하므로, 조건부 확률 $P(D|A) = \frac{P(A \cap D)}{P(A)} = \frac{\frac{2}{36}}{\frac{6}{36}} = \frac{2}{6}$ 입니다. 즉 첫 번째 숫자가 1인 6가지 경우 중 두 번째 숫자가 ≥5인 경우는 2가지 {1&5, 1&6}이라는 의미입니다.

- 앞에서 본 조건부 확률 *P(D|A)*는 조건 없이 사건 D가 일어날 확률인 *P(D)*와는 다름에 유의하세요. *P(D)* = $\frac{12}{36} = \frac{1}{3}$입니다. *P(D)*는 조건부 확률로도 표현할 수 있는데, **표본 공간 *S* 내 모든 경우 중 *D*가 일어날 확률**을 나타내므로 *P(D|S)*라고 볼 수 있습니다. 이 값을 계산해 보아도 *P(D|S)* = $\frac{P(S \cap D)}{P(S)} = \frac{\frac{12}{36}}{\frac{36}{36}} = \frac{12}{36} = P(D)$ 임을 알 수 있습니다. 이로부터 **조건부 확률 *P(D|A)*는 표본 공간을 *A*로 축소했을 때(*A*라는 더 작은 세상에서) *D*가 일어날 확률**로 볼 수도 있습니다. 즉 앞의 예에서 *P(D|A)*는 표본 공간이 *A*={1&1, 1&2, 1&3, 1&4, 1&5, 1&6}으로 축소되었을 때 두 번째 수가 ≥5일 확률이 되는 것입니다.

- 조건부 확률의 정의로부터 *P(A∩B)* = *P(A)* × *P(B|A)* 입니다. 이를 활용하면 *P(A)*와 *P(B|A)*가 주어졌을 때 둘을 곱하여 *P(A∩B)*를 구할 수 있습니다. 예를 들어 **표** 1에서 사건 *A, D, E*가 다음과 같다고 가정하겠습니다.

*A*: 첫 번째 숫자가 1인 6가지 경우 {1&1, 1&2, 1&3, 1&4, 1&5, 1&6}. 따라서 P(A) = 6/36 = 1/6임
*D*: 두 번째 숫자가 ≥ 5인 12가지 경우 {1&5, 1&6, 2&5, 2&6, 3&5, 3&6, 4&5, 4&6, 5&5, 5&6, 6&5, 6&6}
*E*: 두 번째 숫자가 < 5인 경우로, 36 – 12 = 24가지 경우임

첫 번째 숫자가 1인 조건을 만족하는 경우 중 두 번째 숫자가 ≥ 5일 확률인 *P(D|A)*는 2/6입니다. *A*에 속하는 6가지 경우 중 2가지 경우 {1&5, 1&6}에 대해서만 두 번째 숫자가 ≥ 5이기 때문입니다. 따라서 첫 번째 숫자가 1이면서 동시에 두 번째 숫자가 ≥5일 확률 *P(A∩D)* = *P(A)* × *P(D|A)* = 1/6 × 2/6 = 2/36입니다. **즉 *A*라는 작은 세상에서는 *D*에 해당하는 사건이 발생할 확률 *P(D|A)*가 2/6로 큰 편이지만, *S*라는 더 큰 세상에서 보면 같은 사건이 일어날 확률 *P(A∩D|S)* = *P(A∩D)*는 2/36으로 작아지는 것입니다.**

첫 번째 숫자가 1인 조건을 만족하는 경우 중 두 번째 숫자가 < 5일 확률인 *P(E|A)*는 4/6입니다. *A*에 속하는 6가지 경우 중 4가지 경우 {1&1, 1&2, 1&3, 1&4}에 대해서 두 번째 숫자가 < 5 이기 때문입니다. 따라서 첫 번째 숫자가 1이면서 동시에 두 번째 숫자가 < 5일 확률 *P(A∩E)* = *P(A)* × *P(E|A)* = 1/6 × 4/6 = 4/36입니다. 즉 *A*라는 작은 세상에서는 *E*에 해당하는 사건이 발생할 확률 *P(E|A)*가 4/6로 큰 편이지만, *S*라는 더 큰 세상에서 보면 같은 사건이 일어날 확률 *P(A∩E|S)*=*P(A∩E)*는 4/36으로 작아지는 것입니다.

[Q] A와 B가 게임에서 아래와 같은 사건이라고 가정하자.

A: player가 방 A에 들어감
B: player가 아이템 B를 습득함

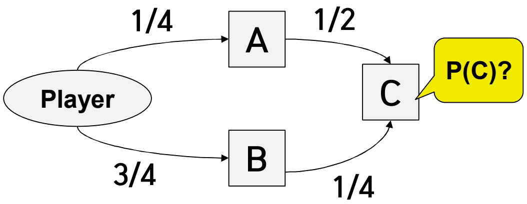
Player가 이미 방 A에 들어갔다고 하자. 이 player가 아이템 B를 습득할 확률은 무엇인가? 즉 A라는 작은 세상에서 B를 습득할 확률은 무엇인가?

- P(A|B)

- $P(B|A)$
- $P(A)$
- $P(B)$
- $P(A \cap B)$

[Q] A, B, C가 게임에서 아래와 같은 사건이라고 가정하자.

A: player가 방 A에 들어감
B: player가 방 B에 들어감
C: player가 방 C에 들어감




게임의 목표는 최종적으로 방 C에 들어가는 것인데, 방 C는 방 A 혹은 방 B를 통해서만 들어갈 수 있다. 게임이 시작하면 먼저 Player는 방 A와 B 중 하나에만 들어갈 수 있으며 둘 다에 들어갈 수는 없다. 방 A에 들어갈 확률은 1/4, 방 B에 들어갈 확률은 3/4이다. 이제 player는 방 A 혹은 B로부터 방 C에 들어갈 수 있는데, A로부터 C에 들어갈 확률은 1/2, B로부터 C에 들어갈 확률은 1/4이다. Player가 방 C에 들어갈 확률 P(C)는 무엇인가?

1/8
3/16
5/16
1

<Python 언어에서 이번 시간에 필요한 기능 이해>

리스트에 저장된 원소를 정렬하고자 할 때는 sorted() 함수를 사용합니다. 이 함수의 인자로 정렬하고자 하는 리스트를 전달하면 정렬 결과를 새로운 리스트에 담아 반환하며, 원본 리스트는 변경하지 않습니다. 아래 예제를 참조하세요.

```
>>> a = [3, 1, 2]
>>> b = sorted(a)
>>> a
[3, 1, 2]
>>> b
[1, 2, 3]
```

원본을 정렬하고자 한다면 리스트 객체에 대해 sort() 함수를 호출하면 됩니다. 아래 예제를 참조하세요.

```
>>> a = [3, 1, 2]
>>> a.sort()
>>> a
[1, 2, 3]
```

Tuple로 이루어진 리스트를 sorted() 혹은 sort() 함수로 정렬하면 tuple의 첫 원소의 오름차순으로 정렬하되, 첫 원소가 같다면 다음 원소를 차례대로 비교해서 정렬합니다. 아래 예제를 참조하세요. 3개의 2-tuple을 담은 리스트를 정렬하는데, tuple의 첫 원소가 0인 tuple을 앞에, 1인 tuple을 뒤에 둡니다. 이 중 첫 원소가 0으로 같은 tuple이 둘 있으므로, 두 번째 원소가 ‘a’인 tuple을 앞에, ‘b’인 tuple을 뒤에 둡니다.

```
>>> c = [(1, 'a'), (0, 'b'), (0, 'a')]
>>> sorted(c)
[(0, 'a'), (0, 'b'), (1, 'a')]
```

만약 위와 같은 기본 정렬 순서와 다르게 리스트를 정렬하고 싶다면 정렬 함수의 인자 key를 통해 새로운 정렬 기준을 전달하면 됩니다. 인자 key에는 함수를 전달하는데, 리스트의 원소 하나를 입력으로 받아 정렬에 사용할 값을 출력으로 내는 함수입니다. 이 값이 증가하는 순서로 정렬하게 됩니다. 아래 예제를 참조하세요. “lambda x: x[1]”는 원소 x를 입력으로 받아 x[1]을 출력으로 내는 함수를 의미합니다. x가 tuple이라면 x[1]은 tuple의 두 번째 값을 의미하므로, tuple의 두 번째 값의 오름차순으로 정렬하게 됩니다.

```
>>> d = [(1, 'c'), (1, 'a'), (2, 'b')]
>>> sorted(d, key=lambda x: x[1])
[(1, 'a'), (2, 'b'), (1, 'c')]
```

만약 tuple의 두 번째 값이 같은 경우에는 이어서 첫 번째 값을 비교하도록 하고 싶다면 “lambda x: (x[1], x[0])”을 인자 key로 전달하면 됩니다. 두 번째 값(x[1])을 먼저 비교하되, 이어서 첫 번째 값(x[0])을 비교하라는 의미입니다. 아래 예제를 참조하세요. 리스트 e에 포함된 4개의 tuple을 두 번째 값 차례로 정

렬하되, 이 중 3개 tuple은 두 번째 값이 ‘b’로 같으므로 첫 번째 값 순서로 정렬하였습니다.

```
>>> e = [(2, 'b'), (1, 'b'), (0, 'c'), (4, 'b')]
>>> sorted(e, key=lambda x: (x[1], x[0]))
[(1, 'b'), (2, 'b'), (4, 'b'), (0, 'c')]
```

[Q] 아래는 리스트 f를 정렬하는 코드와 실행 결과를 보여준다. 어떤 기준에 따라 정렬하였는가?

```
>>> f = [('a', 0.3), ('b', 0.2), ('c', 0.5)]
>>> sorted(f, key=lambda x: -x[1])
[('c', 0.5), ('a', 0.3), ('b', 0.2)]
```

Tuple의 첫 번째 값의 오름차순으로 정렬
Tuple의 첫 번째 값의 내림차순으로 정렬
Tuple의 두 번째 값의 오름차순으로 정렬
Tuple의 두 번째 값의 내림차순으로 정렬
Tuple의 두 번째 값의 오름차순으로 정렬하되, 이 값이 같다면 첫 번째 값의 오름차순으로 정렬