

오늘 할 일

오늘은 '중간고사대비 연습문제' 노트북을 풉니다. 풀기 전에 몇 가지만 짚습니다.

1 시험은 어떻게 보고, 답은 어떻게 쓰나

2 척도를 구분하고 맞는 기술통계량 고르기

3 풀어 쓴 코드를 읽고 함수 하나로 바꾸기

4 문제 문장에서 분포와 인자를 고르기

5 식만 주어졌을 때 바로 계산하기 — 확률표의 빈칸, $E(aX+b) \cdot V(aX+b)$

6 다음 주 중간고사

상자그림·상관계수·추출·기댓값·분포 그래프·LaTeX 는 노트북의 설명 칸에 정리되어 있습니다. 노트북을 위에서부터 풀어 가세요.

중간고사 — 무엇을 어떻게 보나

실습본의 빈칸을 채우듯 {} 자리를 채우고 셀을 실행합니다.

항목	내용
시점	다음 주 (8주차) 수업 시간
형식	인터넷 없는 Jupyter 노트북 실기 — {} 빈칸 채우기
범위	교재 01장 ~ 06장 (1~6주차), 30점 — 07·08장 연속형은 빠진다
허용	본인이 쓴 A4 반 장 단면 요약지 한 장
제출	모든 셀을 실행한 채 저장 → 학번_이름.ipynb
1 맨 위 준비 셀을 먼저 실행한다 — 패키지와 데이터가 다 들어 있다 2 Colab 이 아닌 오프라인 Jupyter — 수업 노트북·검색·AI 를 쓸 수 없다 3 문항끼리 이어지지 않는다 — 앞 문항을 못 풀어도 다음 문항을 풀 수 있다	

답을 쓰는 법

빈칸의 모양에 따라 쓰는 방법이 다릅니다.

칸의 모양

이렇게 쓴다

$x = \{\}$

보기 문자열 중 하나를 따옴표 안에 글자 그대로 — '비율' 을 '비율척도' 로 쓰지 않는다

$x = \{ \}$

값을 내는 코드를 쓴다 — 손으로 계산한 숫자보다 코드가 안전하다

함수(... { })

빠진 인자만 채운다 — 예: , replace=False

$\$$ 와 $\$$ 사이

마크다운 셀을 더블클릭해 LaTeX 을 쓰고 Shift + Enter

1 못 푼 칸은 주석 처리하거나 실행하지 않는다 — 모두 실행(Run All)은 오류가 난 셀에서 멈춘다

2 셀을 실행해서 출력이 나온 채로 저장해야 채점된다

측정척도의 네 가지

아래 척도일수록 위 척도의 성질을 모두 포함하고 하나를 더 할 수 있습니다.

척도	가능한 비교	0 의 의미	예시
명목척도	구분	—	혈액형, 학과, 성별
서열척도	구분 + 순서	—	등수, 만족도, 학점
등간척도	구분 + 순서 + 같은 간격	임의의 0 — 0°C 는 온도가 없다는 뜻이 아니다	섭씨온도, IQ
비율척도	구분 + 순서 + 같은 간격 + 배수	절대 0 — 0kg 은 무게가 없다는 뜻이다	키, 몸무게, 나이, 소득

숫자로 적었다고 양적 척도는 아닙니다. 혈액형을 A=1, B=2, O=3, AB=4 로 적어도 명목척도입니다.

척도에 맞는 기술통계량

아래 척도는 위 척도의 통계량을 모두 쓸 수 있습니다 (+ 는 추가되는 것).

척도	쓸 수 있는 통계량	부적절한 통계량	예시
명목척도	빈도, 비율, 최빈값	평균, 중앙값, 표준편차	혈액형 코드 평균 2.3 은 어느 혈액형도 아니다
서열척도	+ 중앙값, 백분위수	평균, 표준편차는 해석에 주의	만족도 중앙값 4 = 절반이 4점 이상. 점수 간격이 같다는 보장은 없다
등간척도	+ 평균, 분산, 표준편차	배수 비교	평균 15°C 는 되지만 20°C 가 10°C 의 두 배 덥지는 않다
비율척도	+ 변동계수	척도에 따른 제한 없음	몸무게 평균 65kg · 표준편차 8kg, 80kg 은 40kg 의 두 배

리커트 한 문항은 서열척도지만 실무에서는 평균도 쓰고, 여러 문항 합산 점수는 등간척도처럼 다룹니다. 분포가 치우치거나 이상치가 있으면(소득 등) 평균보다 중앙값이 낫습니다.

풀어 쓴 코드를 함수 하나로

식을 풀어 쓴 코드가 주어지면, 안쪽부터 읽어 무엇을 구하는지 알아낸 뒤 같은 값을 내는 함수를 씁니다.

풀어 쓴 코드	안쪽부터 읽으면	함수 하나
<code>sum(x) / len(x)</code>	합 ÷ 개수	<code>np.mean(x)</code>
<code>np.mean((x - np.mean(x)) ** 2)</code>	편차 → 제곱 → 평균	<code>np.var(x)</code>
<code>np.sqrt(np.mean((x - np.mean(x)) ** 2))</code>	분산 → 제곱근	<code>np.std(x)</code>
<code>np.sort(x)[len(x) // 2]</code>	정렬 → 가운데 (개수가 홀수)	<code>np.median(x)</code>

1 `np.var` · `np.std` 는 `n` 으로 나눈다 — 풀어 쓴 코드와 값이 같다

2 `pandas` 의 `.var()` · `.std()` 는 `n-1` 로 나뉘 값이 조금 다르다 — 이 문제에는 `np` 함수를 쓴다

바꾼 뒤 셀을 실행해 위아래 값이 같은지 확인하면 틀렸는지 바로 압니다.

문장에서 분포와 인자를 고른다

문제 문장에서 상황을 읽고 어느 분포인지, 인자에 무엇을 넣을지 정합니다.

문장에 이런 말이 있으면	분포	코드
n 번 해서 정확히 k 번 성공	이항	<code>stats.binom(n, p).pmf(k)</code>
k 번째에 처음으로 성공	기하	<code>stats.geom(p).pmf(k)</code>
평균 λ 건 일어나는 곳에서 정확히 k 건	포아송	<code>stats.poisson(λ).pmf(k)</code>
한 번 해서 성공할 확률	베르누이	<code>stats.bernoulli(p).pmf(1)</code>

- 1 분포를 만들 때는 모수만, 몇 번인지(k)는 pmf 에 넣는다

2 '정확히' 는 pmf, '이하' 는 cdf
- 3 p 는 한 번 시행의 성공 확률 — 주사위의 특정 눈은 1/6, 동전 앞면은 0.5

4 λ 는 묻는 구간의 평균 — 한 시간 평균 2건인데 세 시간을 물으면 $\lambda = 6$

분포를 고르고 확률을 구하시오

각 상황의 분포를 고르고 scipy.stats 로 확률을 구하세요.

```
p1 = { }  
p2 = { }  
p3 = { }
```

1 불량률 10% 인 공정에서 제품 10개를 뽑았을 때 불량률이 정확히 1개일 확률

2 주사위를 굴릴 때 세 번째에 처음으로 1 이 나올 확률

3 한 시간에 평균 4건 전화가 오는 곳에 한 시간 동안 정확히 2건 올 확률

분포를 고르고 확률을 구하시오

```
p1 = stats.binom(10, 0.1).pmf(1)    # 이항 - 10번 중 1번  
p2 = stats.geom(1/6).pmf(3)         # 기하 - 세 번째에 처음  
p3 = stats.poisson(4).pmf(2)        # 포아송 - 평균 4건 중 2건
```

기하 — 앞의 두 번 실패, 세 번째 성공

$$p_2 = \left(\frac{5}{6}\right)^2 \cdot \frac{1}{6} = \frac{25}{216} \approx 0.116$$

$p_1 \approx 0.387, p_2 \approx 0.116, p_3 \approx 0.147$

stats.geom(1/6, 3) 처럼 몇 번째인지를 분포에 넣으면 다른 값이 나옵니다. 분포에는 성공 확률만, 몇 번째인지는 pmf 에 넣습니다.

사탕 봉지 — 표에서 확률 계산하기

한 봉지에 들어 있는 사탕 수 X 가 2, 4, 6 중 하나입니다.

x — 사탕 수(개)	2	4	6
$P(X=x)$ — 그 수가 들어 있을 확률	0.15	0.55	0.30

여러 값의 확률 — 해당하는 칸을 더한다

$$P(X \geq 4) = 0.55 + 0.30 = 0.85$$

기댓값 — 값 $\times$ 확률을 모두 더한다

$$E(X) = 2(0.15) + 4(0.55) + 6(0.30) = 4.3$$

분산 — $(\text{값} - \text{기댓값})^2 \times \text{확률}$ 을 모두 더한다

$$V(X) = (2 - 4.3)^2(0.15) + (4 - 4.3)^2(0.55) + (6 - 4.3)^2(0.30) = 1.71$$

코드로는 `mean = np.sum(x_set * prob)`, `np.sum((x_set - mean) ** 2 * prob)` 입니다. `x_set` 이 윗줄, `prob` 이 아랫줄입니다.

식만 주어졌을 때 — 확률표의 빈칸과 $aX + b$

표나 숫자만 주어지면, 분포를 코드로 정의하지 않고 바로 계산합니다.

```
prob = np.array([0.2, 1 - (0.2 + 0.3 + 0.1), 0.3, 0.1]) # 빈칸을 코드로
mean = np.sum(x_set * prob)
var  = np.sum((x_set - mean) ** 2 * prob)

E_X, V_X = 4, 3
3 * E_X - 2, 3 ** 2 * V_X # E(3X-2), V(3X-2) → (10, 27)
```

숫자만 넣으면 된다

$$E(aX + b) = a E(X) + b, \quad V(aX + b) = a^2 V(X)$$

1 확률표에 빈칸이 하나 있으면 $1 - (\text{나머지 확률의 합})$ — 확률을 다 더하면 1이다

2 빈칸을 채운 뒤에 기댓값·분산을 구한다 — 순서를 바꾸면 기댓값이 틀린다

3 $E(X)$, $V(X)$ 숫자만 주어지면 분포 없이 공식으로 바로 — b 는 분산에서 사라지고 a 는 제공된다

분산 식의 mean 자리에 `np.mean(x_set)` 을 쓰면 확률을 무시한 평균이 들어가 틀립니다.

다음 주 — 중간고사

다음 주가 중간고사입니다. 07·08장 연속형은 시험 뒤 9·10주차에 배웁니다.

1 8주차 — 중간고사: 오프라인 Jupyter 실기, 교재 01~06장, A4 반 장 요약지

2 연습: '중간고사대비 연습문제' 실습본을 풀이본을 보지 않고 채운 뒤 풀이본과 비교하세요

3 9·10주차 — 연속형 확률변수, 대표적인 연속형 확률분포 (교재 07·08장)
