이 장의 목적

외적의 소개:

- 표준적 정의와 계산법
- 기하학적 의미
- 행렬식과의 연결

두 장으로 분리:

- Chapter 10: 표준적 이해 (이 장)
- Chapter 11: 선형변환 관점과 쌍대성 (다음 장)

왜 중요한가?

- 3D 물리학과 그래픽스의 핵심
- 방향성 있는 넓이/부피
- 회전과 각운동량

외적의 차원

중요한 차이:

```
내적: nD에서 정의 (2D, 3D, 4D, ...)
외적: 3D에서만 정의 (특별한 경우)
```

왜 3D만?

- 기하학적 특수성
- 차원의 우연한 일치
- 다음 장에서 자세히

이 장의 범위:

- 2D 버전 (준비 단계)
- 3D 표준 외적

2D 외적 (준비)

두 벡터 v, w:

```
v = [v₁, v₂]
w = [w₁, w₂]
```

평행사변형:

```
      w
     ↗
    /
```

```
  /
 /______ v
```

v와 w가 만드는 평행사변형

2D 외적:

```
v × w = 평행사변형의 넓이 (부호 포함)
```

주의:

- 2D 외적 = 스칼라 (숫자)
- 3D 외적 = 벡터

2D 외적의 부호

방향성:

양수 (v × w > 0):

```
    w
   ↗
  /
 /
└──→ v

v가 w의 오른쪽
(v에서 w로 반시계방향)
```

음수 (v × w < 0):

```
  w
 ↙
  \
   \
    └──→ v

v가 w의 왼쪽
(v에서 w로 시계방향)
```

기억법:

```
v × w > 0: v를 w로 반시계 회전
v × w < 0: v를 w로 시계 회전
```

순서 중요:

```
v × w = -(w × v)
```

2D 외적의 계산

행렬식 이용:

설정:

$$v = [v_1, v_2]$$
$$w = [w_1, w_2]$$

변환 행렬:

$$\begin{bmatrix} v_1 & w_1 \\ v_2 & w_2 \end{bmatrix}$$

행렬식:

$$\det\begin{bmatrix} v_1 & w_1 \\ v_2 & w_2 \end{bmatrix} = v_1w_2 - v_2w_1$$

외적:

$$v \times w = v_1w_2 - v_2w_2$$

왜 행렬식인가?

행렬식의 의미:

det = 넓이 스케일링 계수

단위 정사각형:

$\hat{\imath} = [1, 0]$, $\hat{\jmath} = [0, 1]$

넓이 = 1

변환 후:

$\hat{\imath} \to v = [v_1, v_2]$
$\hat{\jmath} \to w = [w_1, w_2]$

정사각형 → 평행사변형

변환 행렬:

$$\begin{bmatrix} v_1 & w_1 \\ v_2 & w_2 \end{bmatrix}$$

행렬식 = 넓이 변화:

```
det = 평행사변형 넓이 / 단위 정사각형 넓이
    = 평행사변형 넓이 / 1
    = 평행사변형 넓이
```

부호:

```
det > 0: 방향 보존 (반시계)
det < 0: 방향 뒤집힘 (시계)
```

2D 외적 예제

예 1:

```
v = [3, 1]
w = [1, 2]

v × w = 3×2 - 1×1 = 6 - 1 = 5

넓이 = 5, 반시계 방향
```

예 2:

```
v = [2, 3]
w = [4, 1]

v × w = 2×1 - 3×4 = 2 - 12 = -10

넓이 = 10, 시계 방향
```

예 3 (평행):

```
v = [2, 4]
w = [1, 2]  (w = v/2)

v × w = 2×2 - 4×1 = 4 - 4 = 0

평행 → 넓이 = 0
```

3D 외적으로

2D와 3D의 차이:

2D 외적:

- 입력: 두 벡터
- 출력: 스칼라 (넓이)

3D 외적:

- 입력: 두 벡터

- 출력: **벡터**

왜 벡터?

- 3D 평행사변형은 평면
- 평면의 "방향" 필요
- → 수직 벡터로 표현

3D 외적의 정의

두 벡터 v, w (3D):

```
v = [v₁, v₂, v₃]
w = [w₁, w₂, w₃]
```

외적 v × w:

- **방향:** v와 w 모두에 수직
- **크기:** 평행사변형 넓이
- **부호:** 오른손 법칙

오른손 법칙

손가락 사용:

```
1. 검지: v 방향
2. 중지: w 방향
3. 엄지: v × w 방향
```

시각화:

```
      ↑ v × w (엄지)
      |
      |
  w ↗
     /
    /
   /___→ v
 (검지)  (중지)
```

순서 중요:

```
v × w = -(w × v)

순서 바뀌면 방향 반대
```

3D 외적의 크기

크기 공식:

```
|v × w| = |v| |w| sin(θ)
```

여기서 θ = v와 w 사이 각

기하학적 의미:

평행사변형 넓이:

```
넓이 = 밑변 × 높이
     = |v| × |w| sin(θ)
```

시각화:

```
     w
    ↗
   / | h = |w| sin(θ)
  /  |
 /θ  |
/____v____
   |v|
```

특수한 경우:

θ = 0° (평행):

```
sin(0) = 0
|v × w| = 0
```

θ = 90° (수직):

```
sin(90°) = 1
|v × w| = |v| |w| (최대)
```

θ = 180° (반대):

```
sin(180°) = 0
|v × w| = 0
```

3D 외적의 계산 공식

복잡해 보이는 공식:

```
v × w = [v₂w₃ - v₃w₂]
        [v₃w₁ - v₁w₃]
        [v₁w₂ - v₂w₁]
```

외우기 어려움!

행렬식 트릭

특별한 행렬:

$$\begin{bmatrix} \hat{\imath} & \hat{\jmath} & \hat{k} \\ v_1 & v_2 & v_3 \\ w_1 & w_2 & w_3 \end{bmatrix}$$

주의:

- 첫 행이 **벡터** (숫자 아님!)
- 문법적으로 이상함
- 하지만 유용한 기억법

행렬식 "계산":

$$\det\begin{pmatrix} \hat{\imath} & \hat{\jmath} & \hat{k} \\ v_1 & v_2 & v_3 \\ w_1 & w_2 & w_3 \end{pmatrix}$$

코팩터 전개 (첫 행):

$$= \hat{\imath}\,\det\begin{pmatrix} v_2 & v_3 \\ w_2 & w_3 \end{pmatrix} - \hat{\jmath}\,\det\begin{pmatrix} v_1 & v_3 \\ w_1 & w_3 \end{pmatrix} + \hat{k}\,\det\begin{pmatrix} v_1 & v_2 \\ w_1 & w_2 \end{pmatrix}$$

계산:

$$= \hat{\imath}(v_2w_3 - v_3w_2) - \hat{\jmath}(v_1w_3 - v_3w_1) + \hat{k}(v_1w_2 - v_2w_1)$$

벡터 형태:

$$= \begin{bmatrix} v_2w_3 - v_3w_2 \\ -(v_1w_3 - v_3w_1) \\ v_1w_2 - v_2w_1 \end{bmatrix} = \begin{bmatrix} v_2w_3 - v_3w_2 \\ v_3w_1 - v_1w_3 \\ v_1w_2 - v_2w_1 \end{bmatrix}$$

이것이 v × w!

행렬식 트릭의 정당성

왜 작동하는가?

표기법의 마법:

```
[î   ĵ   k̂ ] = "벡터 공간의 기저"
```

행렬식 전개:

- 각 항이 기저벡터의 선형결합
- 계수가 정확히 외적 성분

다음 장에서:

- 이것이 우연이 아님
- 쌍대성으로 설명
- 깊은 의미 있음

3D 외적 예제

예 1: 표준 기저벡터

```
î = [1, 0, 0]
ĵ = [0, 1, 0]

î × ĵ = [0×0 - 0×1]   [0]
        [0×1 - 1×0] = [0] = k̂
        [1×1 - 0×0]   [1]

오른손 법칙 확인 ✓
```

예 2:

```
v = [1, 2, 3]
w = [4, 5, 6]

v × w = [2×6 - 3×5]   [12 - 15]   [-3]
        [3×4 - 1×6] = [12 - 6 ] = [ 6]
        [1×5 - 2×4]   [5 - 8  ]   [-3]
```

검증 (수직성):

```
v · (v × w) = 1(-3) + 2(6) + 3(-3) = -3 + 12 - 9 = 0 ✓
w · (v × w) = 4(-3) + 5(6) + 6(-3) = -12 + 30 - 18 = 0 ✓
```

기하학적 직관

2D 비유 연장:

2D:

```
평행사변형 → 넓이 (스칼라)
```

3D:

```
평행사변형 → 넓이 + 방향 (벡터)
```

방향이 중요한 이유:

- 3D에서 평행사변형은 다양한 방향
- 방향 정보 보존 필요
- 수직 벡터로 인코딩

외적의 성질

1. 반교환법칙:

```
v × w = -(w × v)
```

2. 분배법칙:

```
u × (v + w) = u × v + u × w
```

3. 스칼라 곱:

```
(cv) × w = c(v × w) = v × (cw)
```

4. 자기 자신과의 외적:

```
v × v = 0
```

5. 영벡터:

```
v × 0 = 0
```

6. 수직성:

```
(v × w) · v = 0
(v × w) · w = 0
```

내적과 외적 비교

내적 (Dot Product):

```
v · w = 스칼라
```

- 모든 차원
- "얼마나 같은 방향?"
- 투영과 관련
- 교환법칙: $v \cdot w = w \cdot v$

외적 (Cross Product):

```
v × w = 벡터 (3D만)
```

- 3D 전용
- "얼마나 수직?"
- 회전과 관련
- 반교환법칙: $v \times w = -(w \times v)$

응용 예시

1. 회전축 찾기:

두 벡터 v, w의 회전축:

```
축 = v × w
```

2. 법선 벡터:

평면 위 두 벡터 → 평면의 법선:

```
normal = v × w
```

3. 토크 (물리):

```
τ = r × F

r: 위치 벡터
F: 힘
τ: 토크 (회전력)
```

4. 각운동량:

```
L = r × p

p: 선운동량
L: 각운동량
```

5. 넓이 계산:

삼각형 넓이:

```
넓이 = |v × w| / 2
```

컴퓨터 그래픽스

법선 벡터 계산:

```
삼각형 정점: A, B, C

AB = B - A
AC = C - A

법선 = AB × AC
```

조명 계산:

- 법선 벡터 필요
- 빛 반사 계산
- 외적으로 구함

시야 판정:

```
카메라 방향 · 법선 > 0: 보임
카메라 방향 · 법선 < 0: 안 보임
```

로보틱스 응용

관절 회전:

```
회전축 = 링크1 × 링크2
```

속도 계산:

```
v = ω × r

ω: 각속도
r: 위치
v: 선속도
```

힘/토크 변환:

```
τ = r × F
```

다음 장 예고

더 깊은 이해:

- 왜 3D에만 존재?
- 행렬식 트릭의 진짜 의미
- 쌍대성 연결

선형변환 관점:

- 외적을 변환으로 이해
- 행렬식과의 근본적 연결
- 일반화 가능성

선택 사항:

- 핵심은 아님
- 하지만 매우 우아함
- 수학의 아름다움

핵심 직관 정리

1. 2D 외적:

```
v × w = 평행사변형 넓이 (부호 포함)
      = v₁w₂ - v₂w₁
```

2. 3D 외적:

```
v × w = 수직 벡터
방향: 오른손 법칙
크기: |v| |w| sin(θ) = 평행사변형 넓이
```

3. 계산 트릭:

```
v × w = det([î   ĵ   k̂ ])
           ([v₁  v₂  v₃])
           ([w₁  w₂  w₃])
```

4. 기하학적 의미:

```
외적 = 넓이 + 방향
```

5. 주요 성질:

```
v × w = -(w × v)  (반교환)
v × w ⊥ v, w       (수직성)
|v × w| = 평행사변형 넓이
```

---

2. 퀴즈

문제 1

2D 외적과 3D 외적의 차이를 설명하시오. 특히: (a) 2D 외적의 출력은 무엇이며 무엇을 의미하는가? (b) 3D 외적의 출력은 무엇이며 왜 벡터인가? (c) 2D 외적을 행렬식 det([v_1 w_1; v_2 w_2])로 계산할 수 있는 이유를 기하학적으로 설명하시오.

문제 2

3D 외적 v × w의 기하학적 의미를 세 가지 측면에서 설명하시오: (a) 방향: 왜 v와 w 모두에 수직인가? 오른손 법칙은 무엇인가? (b) 크기: |v × w| = |v| |w| sin(θ)가 왜 평행사변형의 넓이인가? (c) 부호: v × w = -(w × v)인 이유를 기하학적으로 설명하시오.

문제 3

다음 벡터들의 외적을 계산하고, 결과를 기하학적으로 해석하시오:

```
v = [1, 0, 0]  (x축 단위벡터 î)
w = [0, 1, 0]  (y축 단위벡터 ĵ)
```

(a) v × w를 계산하시오. (b) 결과가 z축 단위벡터 k̂인 이유를 오른손 법칙으로 설명하시오. (c) w × v를 계산하고 v × w와 비교하시오.

문제 4

3D 외적의 행렬식 트릭을 설명하시오:

```
v × w = det([î   ĵ   k̂ ])
          ([v₁  v₂  v₃])
          ([w₁  w₂  w₃])
```

(a) 이 표기법이 수학적으로 "이상한" 이유는 무엇인가? (b) 코팩터 전개를 이용하여 실제로 외적 공식이 나오는 과정을 보이시오. (c) 왜 이 트릭이 유용하며, 실제로는 무엇을 의미하는지 간략히 서술하시오.

문제 5

외적의 응용을 설명하시오: (a) 컴퓨터 그래픽스에서 삼각형의 법선 벡터를 구하는 방법 (b) 물리학에서 토크(torque) $\tau = r \times F$의 의미 (c) 왜 외적이 "회전"과 관련이 깊은지 기하학적으로 설명하시오.

---

3. 퀴즈 답

문제 1 답안

2D 외적과 3D 외적의 차이:

(a) 2D 외적:

출력:

```
v × w = 스칼라 (숫자)
```

의미:

1. 크기:

```
|v × w| = 평행사변형의 넓이
```

시각화:

```
      w
     ↗
    /
   /
  /______ v

넓이 = |v × w|
```

2. 부호:

```
v × w > 0: v에서 w로 반시계 회전
v × w < 0: v에서 w로 시계 회전
```

```
v × w = 0: v와 w가 평행
```

구체적 예:

```
v = [3, 0]  (x축 방향)
w = [0, 2]  (y축 방향)

v × w = 3×2 - 0×0 = 6

넓이 = 6 (사각형)
부호 = 양수 (반시계)
```

수학적 표현:

```
v × w = v₁w₂ - v₂w₁
```

(b) 3D 외적:

출력:

```
v × w = 벡터 (3차원)
```

왜 벡터?

이유 1: 방향 정보 필요

2D에서:

- 평행사변형은 xy평면에만
- 방향 = 반시계 or 시계 (2가지)
- 스칼라 부호로 충분

3D에서:

- 평행사변형은 무수히 많은 평면
- 각 평면마다 다른 "방향"
- 스칼라로는 불충분 → 벡터 필요

이유 2: 수직 벡터로 평면 표현

평면의 특성:

```
평면 = 두 벡터 v, w의 span
```

평면 표현 방법:

```
방법 1: v, w 직접 사용 (2개 필요)
```

```
방법 2: 법선 벡터 1개로 표현
```

외적이 선택:

```
v × w = 평면에 수직인 벡터
      + 크기 = 넓이
```

벡터의 세 정보:

1. 방향:

```
v와 w 모두에 수직
(오른손 법칙)
```

2. 크기:

```
|v × w| = 평행사변형 넓이
```

3. 부호 (방향의 위아래):

```
v × w vs w × v (반대)
```

구체적 예:

```
v = [1, 0, 0]
w = [0, 1, 0]

v × w = [0, 0, 1]

방향: z축 양의 방향
크기: 1 (단위 정사각형)
```

(c) 2D 외적과 행렬식의 기하학적 관계:

공식:

```
v × w = det([v₁  w₁])
           ([v₂  w₂])
```

행렬식의 기하학적 의미 복습:

행렬식 = 넓이 스케일링 계수

```
det(A) = 변환 후 넓이 / 원래 넓이
```

적용:

Step 1: 단위 정사각형 설정

원래 도형:

```
정점: O(0,0), î(1,0), ĵ(0,1), P(1,1)

넓이 = 1
```

Step 2: 선형변환 정의

변환 행렬:

```
A = [v₁  w₁]
    [v₂  w₂]
```

이것은:

```
î → v = [v₁, v₂]
ĵ → w = [w₁, w₂]
```

Step 3: 변환 후 도형

정사각형 → 평행사변형:

```
O(0,0) → (0,0)
î(1,0) → v
ĵ(0,1) → w
P(1,1) → v + w
```

Step 4: 넓이 계산

행렬식:

```
det(A) = 변환 후 넓이 / 1
       = 평행사변형 넓이
```

Step 5: 부호

```
det > 0: 방향 보존 (반시계)
det < 0: 방향 뒤집힘 (시계)
```

이것이 정확히 외적의 부호!

결론:

```
v × w = det([v₁  w₁])
           ([v₂  w₂])
```

의미:

- v, w를 열로 하는 행렬의 행렬식
- 단위 정사각형을 v, w 평행사변형으로 변환
- 행렬식 = 넓이 변화 = 평행사변형 넓이

시각적 정리:

```
단위 정사각형      변환 A      평행사변형
(î, ĵ)       ------->       (v, w)
넓이 = 1                넓이 = det(A)

det(A) = v₁w₂ - v₂w₁ = v × w
```

이것이 2D 외적을 행렬식으로 계산하는 기하학적 이유입니다!

문제 2 답안

3D 외적 v × w의 기하학적 의미:

(a) 방향: 수직성과 오른손 법칙

왜 v와 w 모두에 수직?

기하학적 이유:

평행사변형과 법선:

```
v와 w가 평면을 span
→ 평행사변형은 이 평면에 놓임
→ 평면을 특징짓는 것 = 법선 벡터
```

평면 방정식:

```
평면: ax + by + cz = d

법선 벡터: [a, b, c]
```

외적의 역할:

```
v × w = 평면의 법선 벡터
```

수학적 증명:

$v \times w = [v_2w_3 - v_3w_2, v_3w_1 - v_1w_3, v_1w_2 - v_2w_1]$라 하자.

v와 수직:

```
v · (v × w) = v₁(v₂w₃ - v₃w₂) + v₂(v₃w₁ - v₁w₃) + v₃(v₁w₂ - v₂w₁)
            = v₁v₂w₃ - v₁v₃w₂ + v₂v₃w₁ - v₂v₁w₃ + v₃v₁w₂ - v₃v₂w₁
```

```
            = v₁v₂w₃ - v₂v₁w₃ - v₁v₃w₂ + v₃v₁w₂ + v₂v₃w₁ - v₃v₂w₁
            = 0 ✓
```

w와 수직:

```
w · (v × w) = 0  (같은 방식으로 증명)
```

오른손 법칙:

손가락 배치:

```
1. 검지: v 방향을 가리킴
2. 중지: w 방향을 가리킴 (검지와 수직 유지)
3. 엄지: v × w 방향
```

시각화:

```
      엄지 ↑ (v × w)
           |
           |
     중지 ↗ (w)
         /
        /
    검지 → (v)
```

기억법:

```
v에서 w로 "나사" 돌리는 방향
→ 나사가 나가는 방향 = v × w
```

좌표계 예:

```
î × ĵ = k̂  (오른손 좌표계)

검지: x축 (î)
중지: y축 (ĵ)
엄지: z축 (k̂)
```

순서 바뀌면:

```
w × v = -(v × w)

손 뒤집기 = 방향 반대
```

(b) 크기: 평행사변형 넓이

공식:

$$|v \times w| = |v|\ |w| \sin(\theta)$$

기하학적 증명:

평행사변형 구성:

```
     w
    ↗
   /|
  / | h
 /θ |
/___v___
   |v|
```

높이 계산:

$$h = |w| \sin(\theta)$$

넓이:

$$\begin{aligned} \text{넓이} &= \text{밑변} \times \text{높이} \\ &= |v| \times |w| \sin(\theta) \\ &= |v \times w| \end{aligned}$$

대수적 확인:

외적 크기 제곱:

$$|v \times w|^2 = (v_2w_3 - v_3w_2)^2 + (v_3w_1 - v_1w_3)^2 + (v_1w_2 - v_2w_1)^2$$

전개하면 (복잡):

$$\begin{aligned} &= |v|^2\ |w|^2 - (v \cdot w)^2 \\ &= |v|^2\ |w|^2 - |v|^2\ |w|^2 \cos^2(\theta) \\ &= |v|^2\ |w|^2\ (1 - \cos^2(\theta)) \\ &= |v|^2\ |w|^2 \sin^2(\theta) \end{aligned}$$

따라서:

$$|v \times w| = |v|\ |w| \sin(\theta)\ ✓$$

특수한 경우:

평행 (θ = 0° or 180°):

```
sin(θ) = 0
|v × w| = 0

넓이 = 0 (평행사변형이 선으로 찌그러짐)
```

수직 (θ = 90°):

```
sin(90°) = 1
|v × w| = |v| |w|

넓이 최대 (직사각형)
```

(c) 부호: 반교환법칙

공식:

```
v × w = -(w × w)
```

기하학적 이유:

오른손 법칙 적용:

v × w:

```
검지: v
중지: w
엄지: ↑ (위)
```

w × v:

```
검지: w
중지: v
엄지: ↓ (아래)
```

방향이 정반대!

넓이는 같지만:

- 평행사변형은 같음
- 크기 |v × w| = |w × v|

하지만 방향이 반대:

- v × w는 위
- w × v는 아래

벡터 관점:

```
w × v = −(v × w)
```

수학적 확인:

```
v × w = [v₂w₃ − v₃w₂]
        [v₃w₁ − v₁w₃]
        [v₁w₂ − v₂w₁]

w × v = [w₂v₃ − w₃v₂]   [−(v₂w₃ − v₃w₂)]
        [w₃v₁ − w₁v₃] = [−(v₃w₁ − v₁w₃)]
        [w₁v₂ − w₂v₁]   [−(v₁w₂ − v₂w₁)]

      = −(v × w) ✓
```

2D 비유:

2D에서:

```
v × w = v₁w₂ − v₂w₁
w × v = w₁v₂ − w₂v₁ = −(v₁w₂ − v₂w₁) = −(v × w)
```

부호가 바뀜 = 회전 방향 바뀜

3D에서:

```
벡터 자체가 반대 = 법선 방향 반대
```

직관:

순서를 바꾸면:

- 평면은 같음
- 넓이는 같음
- 하지만 "위/아래" 반대

이것이 반교환법칙의 기하학적 의미입니다!

문제 3 답안

표준 기저벡터의 외적:

(a) v × w 계산:

주어진:

```
v = î = [1, 0, 0]
w = ĵ = [0, 1, 0]
```

외적 공식:

```
v × w = [v₂w₃ - v₃w₂]
        [v₃w₁ - v₁w₃]
        [v₁w₂ - v₂w₁]
```

대입:

```
v₁ = 1, v₂ = 0, v₃ = 0
w₁ = 0, w₂ = 1, w₃ = 0
```

계산:

```
첫 번째 성분: v₂w₃ - v₃w₂ = 0×0 - 0×1 = 0
두 번째 성분: v₃w₁ - v₁w₃ = 0×0 - 1×0 = 0
세 번째 성분: v₁w₂ - v₂w₁ = 1×1 - 0×0 = 1
```

결과:

```
v × w = [0]
        [0] = k̂ (z축 단위벡터)
        [1]
```

행렬식 트릭으로:

```
v × w = det([î   ĵ   k̂])
           ([1   0   0])
           ([0   1   0])

코팩터 전개 (첫 행):
= î det([0  0]) - ĵ det([1  0]) + k̂ det([1  0])
       ([1  0])        ([0  0])        ([0  1])

= î(0) - ĵ(0) + k̂(1)
= k̂
```

(b) 오른손 법칙 설명:

손 배치:

1. 검지: v = î 방향

```
검지를 x축 양의 방향으로
→
```

2. 중지: w = ĵ 방향

```
중지를 y축 양의 방향으로 (검지와 수직)
↑
```

3. 엄지: v × w 방향

```
엄지가 자동으로 z축 양의 방향
⊙ (종이에서 나오는 방향)
```

시각화:

```
     z (엄지, v × w)
     ↑
     |
     |
     |____→ x (검지, v = î)
    /
   /
  ↙
 y (중지, w = ĵ)
```

오른손 좌표계:

```
î × ĵ = k̂
```

이것이 오른손 좌표계의 정의!

나사 비유:

```
î에서 ĵ로 반시계 회전
→ 나사가 z축 양의 방향으로 나감
→ k̂
```

기하학적 의미:

xy평면:

- v = î와 w = ĵ가 span
- xy평면 위의 정사각형 (넓이 = 1)

법선 벡터:

- xy평면에 수직 = z축
- 방향은 오른손 법칙
- k̂ (z축 양의 방향)

(c) w × v 계산 및 비교:

계산:

```
w × v = ĵ × î
```

외적 공식:

```
w × v = [w₂v₃ − w₃v₂]
        [w₃v₁ − w₁v₃]
        [w₁v₂ − w₂v₁]
```

대입:

```
w₁ = 0, w₂ = 1, w₃ = 0
v₁ = 1, v₂ = 0, v₃ = 0
```

계산:

```
첫 번째 성분: 1×0 − 0×0 = 0
두 번째 성분: 0×1 − 0×0 = 0
세 번째 성분: 0×0 − 1×1 = −1
```

결과:

```
w × v = [0 ]
        [0 ] = −k̂
        [−1]
```

행렬식 트릭으로:

```
w × v = det([î   ĵ   k̂])
           ([0   1   0])
           ([1   0   0])

= î det([1  0]) − ĵ det([0  0]) + k̂ det([0  1])
       ([0  0])        ([1  0])        ([1  0])

= î(0) − ĵ(0) + k̂(−1)
= −k̂
```

비교:

v × w vs w × v:

```
v × w = k̂
w × v = −k̂

w × v = −(v × w) ✓
```

반교환법칙 확인!

기하학적 의미:

오른손 법칙:

î × ĵ:

```
검지: x축
중지: y축
엄지: z축 ↑ (양의 방향)
```

ĵ × î:

```
검지: y축
중지: x축
엄지: z축 ↓ (음의 방향)
```

방향이 정반대!

평면과 넓이:

- 같은 xy평면
- 같은 넓이 (1)
- 하지만 법선 방향 반대

순환 성질:

```
î × ĵ = k̂
ĵ × k̂ = î
k̂ × î = ĵ

ĵ × î = -k̂
k̂ × ĵ = -î
î × k̂ = -ĵ
```

이것이 3D 표준 기저의 외적 관계입니다!

문제 4 답안

3D 외적의 행렬식 트릭:

(a) 수학적으로 "이상한" 이유:

표준 행렬식:

```
det([a  b  c])
   ([d  e  f])
   ([g  h  i])
```

모든 원소 = 숫자 (스칼라)

외적의 "행렬식":

```
det([î   ĵ   k̂ ])
   ([v₁  v₂  v₃])
   ([w₁  w₂  w₃])
```

첫 행 = 벡터!

문제점:

1. 타입 불일치:

```
첫 행: 벡터
나머지: 스칼라

혼합된 타입!
```

2. 행렬식은 숫자를 반환:

```
일반 행렬식: det(A) = 스칼라
```

하지만:

```
외적 "행렬식" = 벡터?
```

모순!

3. 수학적으로 정의되지 않음:

```
det의 정의: n×n 숫자 행렬 → 숫자
```

벡터를 원소로 하는 행렬은:

```
정의역 밖!
```

왜 이상한가:

공식적으로는:

- 행렬식 아님
- 표기법의 남용
- 기억법 장치

하지만:

- 작동함!
- 올바른 답

- 매우 유용

다음 장에서:

- 쌍대성으로 정당화
- 실제 의미 있음
- 우연 아님

(b) 코팩터 전개:

주어진:

```
"det"([î   ĵ   k̂ ])
    ([v₁  v₂  v₃])
    ([w₁  w₂  w₃])
```

코팩터 전개 (첫 행):

일반 공식:

```
det(A) = Σᵢ (−1)^(1+i) a₁ᵢ det(M₁ᵢ)
```

여기서 M_{1i} = 1행과 i열 제거한 부분행렬

적용:

i = 1 (î 항):

```
(−1)^(1+1) î det([v₂  v₃])
                ([w₂  w₃])

= î(v₂w₃ − v₃w₂)
```

i = 2 (ĵ 항):

```
(−1)^(1+2) ĵ det([v₁  v₃])
                ([w₁  w₃])

= −ĵ(v₁w₃ − v₃w₁)
= ĵ(v₃w₁ − v₁w₃)
```

i = 3 (k̂ 항):

```
(−1)^(1+3) k̂ det([v₁  v₂])
                ([w₁  w₂])

= k̂(v₁w₂ − v₂w₁)
```

합계:

```
"det" = î(v₂w₃ - v₃w₂) + ĵ(v₃w₁ - v₁w₃) + k̂(v₁w₂ - v₂w₁)
```

벡터 형태:

```
= [v₂w₃ - v₃w₂]
  [v₃w₁ - v₁w₃]
  [v₁w₂ - v₂w₁]
```

이것이 정확히 v × w!

검증:

외적 정의:

```
v × w = [v₂w₃ - v₃w₂]
        [v₃w₁ - v₁w₃]
        [v₁w₂ - v₂w₁]
```

행렬식 트릭:

```
"det"([î   ĵ   k̂ ]) = [v₂w₃ - v₃w₂]
    ([v₁  v₂  v₃])   [v₃w₁ - v₁w₃]
    ([w₁  w₂  w₃])   [v₁w₂ - v₂w₁]
```

완전히 일치! ✓

(c) 유용성과 실제 의미:

왜 유용한가:

1. 기억하기 쉬움:

```
복잡한 공식:
v × w = [v₂w₃ - v₃w₂, v₃w₁ - v₁w₃, v₁w₂ - v₂w₁]

vs

간단한 트릭:
det([î  ĵ  k̂ ])
   ([v₁ v₂ v₃])
   ([w₁ w₂ w₃])
```

2. 계산 체계적:

- 행렬식은 익숙한 절차
- 코팩터 전개 알고리즘

- 실수 줄임

3. 패턴 인식:

```
각 성분 = 2×2 행렬식
```

실제 의미:

표기법의 마법:

벡터 î, ĵ, k̂를 "형식적 기호"로 취급:

```
î = "x성분을 추출하는 연산자"
ĵ = "y성분을 추출하는 연산자"
k̂ = "z성분을 추출하는 연산자"
```

행렬식 전개:

- 각 기저벡터에 스칼라 곱
- 선형결합 형성
- 결과 = 벡터

다음 장 예고:

쌍대성 관점:

```
외적 = 특별한 선형변환
행렬식 = 변환의 "부피"
```

이 둘의 연결:

- 깊은 수학적 의미
- 우연이 아님
- 매우 우아함

결론:

행렬식 트릭은:

1. **표기법의 남용** (공식적으로)
2. **유용한 기억법** (실용적으로)
3. **깊은 의미 있음** (수학적으로)

다음 장에서 "왜 작동하는가"의 진정한 답을 배웁니다!

문제 5 답안

외적의 응용:

(a) 컴퓨터 그래픽스: 법선 벡터

문제:

```
3D 모델의 삼각형
→ 조명 계산 위해 법선 벡터 필요
```

방법:

Step 1: 삼각형 정점

```
A, B, C (시계반대 방향)
```

Step 2: 두 변 벡터

```
AB = B - A
AC = C - A
```

시각화:

```
      C
     ↗
    /
   /
  /
 A───→ B
```

Step 3: 외적

```
n = AB × AC
```

방향:

- AB와 AC 모두에 수직
- 삼각형 평면의 법선
- 오른손 법칙 (평면 "위")

크기:

```
|n| = |AB| |AC| sin(θ)
    = 평행사변형 넓이
    = 2 × 삼각형 넓이
```

Step 4: 정규화 (단위벡터)

```
n̂ = n / |n|
```

응용:

1. 조명 계산 (람베르트 반사):

```
밝기 = max(0, L · n̂)

L: 광원 방향
n̂: 법선 벡터
```

2. 뒷면 제거 (Backface Culling):

```
if (카메라방향 · n̂ < 0)
    삼각형 그리지 않음
```

3. 부드러운 음영 (Smooth Shading):

- 정점마다 법선 계산
- 인접 삼각형 법선 평균

구체적 예:

```
A = [0, 0, 0]
B = [1, 0, 0]
C = [0, 1, 0]

AB = [1, 0, 0]
AC = [0, 1, 0]

n = AB × AC = [0, 0, 1] = k̂

법선: z축 양의 방향 (xy평면 위)
```

(b) 물리학: 토크 (Torque)

정의:

```
τ = r × F

r: 회전축에서 힘의 작용점까지 위치 벡터
F: 작용하는 힘
τ: 토크 (회전력)
```

의미:

크기:

```
|τ| = |r| |F| sin(θ)
    = r⊥ |F|
```

여기서 $r_\perp$ = F에 수직인 r의 성분 = "지렛대 팔"

시각화:

```
       F (힘)
      ↗
     /
    / θ
   /
  0───────→ r (위치)
  (회전축)
```

토크:

```
|τ| = |r| |F| sin(θ)
    = (지렛대 길이) × (힘)
```

방향:

- r와 F 모두에 수직
- 회전축 방향
- 오른손 법칙

오른손 법칙 (토크):

```
손가락: 회전 방향 (r → F)
엄지: τ 방향 (회전축)
```

예:

문 열기:

```
O = 경첩 (회전축)
r = 경첩에서 손잡이까지
F = 손이 미는 힘

τ = r × F
```

수직으로 밀 때 ($\theta = 90°$):

```
|τ| = |r| |F| (최대)
```

경첩 방향으로 밀 때 ($\theta = 0°$):

```
|τ| = 0 (문이 안 열림)
```

각가속도:

```
τ = I α
```

```
I: 관성 모멘트
α: 각가속도
```

(c) 외적과 회전의 관계

기하학적 연결:

1. 회전축:

두 벡터 v, w가 있을 때:

```
v × w = 회전축
```

의미:

- v를 w로 회전시키는 축
- 축 주위로 회전
- 방향 = 오른손 법칙

예:

```
v = î (x축)
w = ĵ (y축)

v × w = k̂ (z축)

의미: x축을 y축으로 회전 = z축 주위 회전
```

2. 회전각:

외적의 크기:

```
|v × w| = |v| |w| sin(θ)
```

sin(θ):

- 회전 정도
- 수직일 때 최대 (90°)
- 평행일 때 최소 (0°)

3. 각속도:

각속도 ω로 회전하는 물체:

```
v = ω × r

v: 선속도
```

```
ω: 각속도 벡터 (회전축 방향)
r: 위치 벡터
```

의미:

- 회전 → 선속도 생성
- 외적으로 표현

시각화:

```
     ω (회전축)
     ↑
     |
     |___→ r

v = ω × r (접선 방향)
```

4. 각운동량:

```
L = r × p

p = mv (선운동량)
L: 각운동량
```

보존:

- 외력 토크 = 0
- L 보존

5. 회전 변환:

벡터 v를 축 n 주위로 θ 회전:

```
v' = v cos(θ) + (n × v) sin(θ) + n(n · v)(1 - cos(θ))

(로드리게스 회전 공식)
```

n × v 항이 핵심!

직관적 이해:

왜 외적 = 회전?

평면과 법선:

- 회전은 평면 내 운동
- 평면 = 두 벡터 span
- 법선 = 회전축

오른손 법칙:

- 나사 돌리기

- 회전 방향 → 나사 나가는 방향
- 외적 = 나사 방향 = 회전축

넓이 = 회전 정도:

- 큰 넓이 = 큰 회전
- 작은 넓이 = 작은 회전
- 0 넓이 = 회전 없음 (평행)

결론:

외적은 회전의 모든 측면 인코딩:

1. **축:** 방향
2. **정도:** 크기 (sin(θ))
3. **방향성:** 부호 (오른손 법칙)

이것이 외적이 물리학과 공학에서 회전 표현에 핵심인 이유입니다!

요약:

외적의 세 가지 주요 응용:

1. **그래픽스:** 법선 벡터 → 조명/음영
2. **물리:** 토크/각운동량 → 회전 동역학
3. **일반:** 회전축/각속도 → 회전 운동

모두 "회전"이라는 공통 주제!

```
Dataview (inline field 'î det([v₂  v₃]) − ĵ det([v₁  v₃]) + k̂ det([v₁
v₂])
       ([w₂  w₃])       ([w₁  w₃])       ([w₁  w₂])'): Error:
-- PARSING FAILED --------------------------------------------------

> 1 | î det([v₂  v₃]) − ĵ det([v₁  v₃]) + k̂ det([v₁  v₂])
    |   ^
  2 |        ([w₂  w₃])       ([w₁  w₃])       ([w₁  w₂])

Expected one of the following:

'*' or '/' or '%', '+' or '-', '>=' or '<=' or '!=' or '=' or '>' or '<',
'and' or 'or'
```

```
Dataview (inline field 'î(v₂w₃ − v₃w₂) − ĵ(v₁w₃ − v₃w₁) + k̂(v₁w₂ −
v₂w₁)'): Error:
-- PARSING FAILED --------------------------------------------------

> 1 | î(v₂w₃ − v₃w₂) − ĵ(v₁w₃ − v₃w₁) + k̂(v₁w₂ − v₂w₁)
```

```
    |    ^

Expected one of the following:

'(', ')', '*' or '/' or '%', '+' or '-', ',', '.', '>=' or '<=' or '!=' or
'=' or '>' or '<', '[', 'and' or 'or', /[0-9\p{Letter}_-]/u, text
```

```
Dataview (inline field '[v₂w₃ − v₃w₂]
  [−(v₁w₃ − v₃w₁)]  = [v₃w₁ − v₁w₃]
  [v₁w₂ − v₂w₁]       [v₁w₂ − v₂w₁]'): Error:
-- PARSING FAILED --------------------------------------------------

> 1 | [v₂w₃ − v₃w₂]
    |   ^
  2 |   [−(v₁w₃ − v₃w₁)]  = [v₃w₁ − v₁w₃]
  3 |   [v₁w₂ − v₂w₁]       [v₁w₂ − v₂w₁]

Expected:

list ('[1, 2, 3]')
```

```
Dataview (inline field '|v|² |w|² − (v · w)²
= |v|² |w|² − |v|² |w|² cos²(θ)
= |v|² |w|² (1 − cos²(θ))
= |v|² |w|² sin²(θ)'): Error:
-- PARSING FAILED --------------------------------------------------

> 1 | |v|² |w|² − (v · w)²
    | ^
  2 | = |v|² |w|² − |v|² |w|² cos²(θ)
  3 | = |v|² |w|² (1 − cos²(θ))

Expected one of the following:

'(', 'null', boolean, date, duration, file link, list ('[1, 2, 3]'),
negated field, number, object ('{ a: 1, b: 2 }'), string, variable
```

```
Dataview (inline field '[v₂w₃ − v₃w₂]
  [v₃w₁ − v₁w₃]
  [v₁w₂ − v₂w₁]'): Error:
-- PARSING FAILED --------------------------------------------------

> 1 | [v₂w₃ − v₃w₂]
    |   ^
  2 |   [v₃w₁ − v₁w₃]
  3 |   [v₁w₂ − v₂w₁]
```

```
Expected:

list ('[1, 2, 3]')
```