이 장의 목적

내적의 깊은 이해:

- 전통적 정의 (계산)
- 기하학적 의미 (투영)
- 선형변환 관점 (쌍대성)

왜 이제야 내적을?

- 전통적으로는 맨 처음 배움
- 하지만 **선형변환을 알아야** 진정한 의미 이해
- 쌍대성(duality)이라는 깊은 개념

전통적 내적 정의

수치적 정의:

```
v · w = [v₁, v₂] · [w₁, w₂] = v₁w₁ + v₂w₂
```

과정:

1. 두 벡터의 각 좌표 짝짓기
2. 각 쌍을 곱하기
3. 모두 더하기

예:

```
[1, 2] · [3, 4] = 1×3 + 2×4 = 3 + 8 = 11
```

일반화 (n차원):

```
v · w = Σᵢ vᵢwᵢ = v₁w₁ + v₂w₂ + ... + vₙwₙ
```

기하학적 의미: 투영

내적의 기하학적 해석:

```
v · w = (v를 w 방향으로 투영한 길이) × |w|
```

또는 대칭적으로:

```
v · w = (w를 v 방향으로 투영한 길이) × |v|
```

시각화:

1. w 방향으로 v를 투영:

```
        v
       ↗
      /
     /
    /_________ w
   /  ↓
proj  (v를 w로 투영)
```

2. 투영 길이 측정:

- v의 끝에서 w에 수직선
- 원점에서 수직선까지 거리 = 투영 길이

3. 내적:

```
v · w = (투영 길이) × |w|
```

투영 길이의 부호

같은 방향:

```
v와 w가 예각 (<90°)
→ 투영 길이 > 0
→ v · w > 0
```

수직:

```
v와 w가 직각 (=90°)
→ 투영 길이 = 0
→ v · w = 0
```

반대 방향:

```
v와 w가 둔각 (>90°)
→ 투영 길이 < 0
→ v · w < 0
```

시각적:

```
예각:    v ↗   w →    (양수)
수직:    v ↑   w →    (0)
둔각:  v ↙     w →    (음수)
```

순서 무관성

대칭성:

```
v · w = w · v
```

기하학적 이유:

v → w 투영:

```
(v를 w로) × |w|
```

w → v 투영:

```
(w를 v로) × |v|
```

같은 이유: 평행사변형의 넓이를 이용한 기하학적 증명

- 두 경우 모두 같은 값

직관:

- 투영의 "총량"은 순서와 무관
- 기하학적으로 대칭적

스케일링 성질

벡터를 스케일하면:

```
(2v) · w = 2(v · w)
```

기하학적 의미:

v를 2배로:

- 투영 길이도 2배
- 따라서 내적도 2배

일반적:

```
(cv) · w = c(v · w)
```

단위벡터와의 내적

단위벡터 û (|û| = 1):

```
v · û = (v를 û로 투영한 길이) × 1
      = 투영 길이 그 자체
```

의미:

- 단위벡터와의 내적 = 그 방향으로의 좌표
- 매우 중요한 해석!

예:

```
v = [3, 4]
î = [1, 0] (x축 단위벡터)

v · î = 3 = v의 x좌표
```

핵심 질문

두 가지 미스터리:

1. **순서가 왜 무관한가?**
   - 투영의 대칭성이 왜?
2. **좌표 곱셈이 왜 투영과 같은가?**
   - $v_1w_1 + v_2w_2$가 왜 기하학적 투영?

답: 쌍대성(Duality)!

nD → 1D 선형변환

복습: 1×n 행렬

```
[u₁  u₂] [x]
         [y] = u₁x + u₂y
```

의미:

- 2D 벡터를 입력
- 숫자(1D)를 출력

선형성:

- 직선이 균등 간격 유지
- 원점 고정

선형변환의 시각화 (2D → 1D)

입력 공간 (2D):

- 균등 간격의 점들

출력 공간 (1D):

- 수직선

선형성 확인:

```
입력: ──●──●──●── (2D 직선)
          ↓ 변환
출력: ─●─●─●─ (1D 직선)
```

균등 간격 유지!

1×n 행렬의 기하학적 의미

1×2 행렬:

```
[u₁  u₂]
```

의미:

- 2D → 1D 변환
- 완전히 두 숫자로 결정

어떻게?

기저벡터의 도착지:

```
î = [1, 0] → u₁
ĵ = [0, 1] → u₂
```

행렬 형태:

```
[u₁  u₂] = [î가 가는 곳, ĵ가 가는 곳]
```

행렬-벡터 곱셈 = 내적

핵심 통찰:

```
[u₁  u₂] [x]   = u₁x + u₂y
         [y]
```

이것은:

```
[u₁, u₂] · [x, y] = u₁x + u₂y
```

내적과 완전히 동일!

쌍대성 (Duality)

정의: 서로 다른 두 수학적 대상 사이의 자연스럽지만 놀라운 대응

선형대수에서:

```
벡터 ↔ 선형변환 (nD → 1D)
```

구체적:

벡터 $u = [u_1, u_2]$:

- 2D 공간의 화살표

쌍대 (dual):

- 1×2 행렬 [u_1 u_2]
- 2D → 1D 선형변환

관계:

```
u와의 내적 = u의 쌍대 변환 적용
```

왜 쌍대성이 존재하는가?

기하학적 이유:

단위벡터 û를 생각:

```
û = [u₁, u₂]  (|û| = 1)
```

û 방향의 수직선 정의:

```
û를 따라 놓인 1D 수직선
```

투영 연산:

```
임의의 벡터 v를 이 수직선에 투영
→ 숫자 하나 얻음
```

이것이 선형변환!

변환 행렬:

```
[u₁  u₂]
```

변환 적용:

```
[u₁  u₂] [v₁]
         [v₂] = u₁v₁ + u₂v₂ = û · v
```

투영이 선형변환인 이유

선형성 확인:

1. 가법성:

```
proj(v + w) = proj(v) + proj(w)
```

2. 스케일링:

```
proj(cv) = c·proj(v)
```

기하학적 직관:

- 투영은 "선형적"
- 격자선 유지
- 비례성 유지

û의 좌표의 의미

단위벡터 û = [u_1, u_2]:

u_1의 의미:

```
u₁ = î를 û 방향 수직선에 투영한 값
```

u_2의 의미:

```
u₂ = ĵ를 û 방향 수직선에 투영한 값
```

왜 중요한가?

변환 행렬:

```
[u₁  u₂] = [î의 도착지, ĵ의 도착지]
```

임의의 벡터 v = [x, y]:

```
[u₁  u₂] [x]   = x·u₁ + y·u₂
         [y]

= x(î의 투영) + y(ĵ의 투영)
= v의 투영
```

선형결합이 보존됨!

기하학적 증명

설정:

- 대각선 방향 단위벡터 û
- 대각선을 따라 놓인 수직선

질문:

```
[u₁  u₂] [x]   이 왜 v를 û로 투영?
         [y]
```

답:

1. 기저벡터 투영:

```
î → u₁ (수직선 위 값)
ĵ → u₂ (수직선 위 값)
```

2. 선형성:

```
v = xî + yĵ

투영(v) = 투영(xî + yĵ)
        = x·투영(î) + y·투영(ĵ)
        = x·u₁ + y·u₂
```

3. 결론:

```
[u₁  u₂] [x]   = 투영(v)
         [y]
```

일반 벡터로의 확장

단위벡터가 아닌 u:

```
u = [u₁, u₂]  (|u| ≠ 1)
```

쌍대 변환:

```
[u₁  u₂]
```

변환 적용:

```
[u₁  u₂] [v₁]   = u₁v₁ + u₂v₂
         [v₂]

= u · v (내적)
```

기하학적 의미:

```
= (v를 u 방향으로 투영) × |u|
```

왜?

- $u = |u| \cdot \hat{u}$ ($\hat{u}$는 단위벡터)
- $u \cdot v = |u| (\hat{u} \cdot v)$
- $= |u| \times$ ($\hat{u}$ 방향 투영)

내적 공식의 유도

왜 $v_1w_1 + v_2w_2$인가?

선형변환 관점:

벡터 $w = [w_1, w_2]$:

- 쌍대 변환: $[w_1\ w_2]$

벡터 $v = [v_1, v_2]$:

변환 적용:

```
[w₁  w₂] [v₁]   = w₁v₁ + w₂v₂
         [v₂]
```

이것이 내적 공식!

기하학적 해석:

```
= v를 w로 투영 × |w|
```

결론: 수치 공식 ↔ 기하학적 투영 둘이 같은 이유 = 쌍대성!

대칭성의 증명

왜 $v \cdot w = w \cdot v$?

변환 관점:

$v \cdot w$:

- w의 쌍대 변환을 v에 적용

$w \cdot v$:

- v의 쌍대 변환을 w에 적용

둘 다:

```
= v₁w₁ + v₂w₂
```

기하학적:

- 투영의 대칭성
- 평행사변형의 넓이 논증

쌍대성의 중요성

벡터의 이중성:

1. 기하학적 객체:

- 공간의 화살표
- 방향과 크기

2. 선형 함수:

- nD → 1D 변환
- 투영 연산

어느 관점?

- 상황에 따라 유용한 것 선택
- 같은 대상의 다른 얼굴

예: 그래디언트 (미적분):

- 벡터로 표현
- 실제로는 방향 도함수 (선형 함수)
- 쌍대성의 완벽한 예!

쌍대성의 일반적 정의

수학 전반: 서로 다른 종류의 수학적 대상 사이의 자연스럽고 놀라운 대응

선형대수의 경우:

벡터의 쌍대:

- 그 벡터가 인코딩하는 선형변환

선형변환의 쌍대 (nD → 1D):

- 공간 내의 특정 벡터

양방향 대응:

```
벡터 ←→ 선형변환
```

내적의 요약

표면적 이해:

- 기하학적 도구
- 투영 이해
- 벡터 방향 판단

이것만으로도 충분히 유용!

깊은 이해:

- 한 벡터를 변환으로 "번역"
- 쌍대성의 구체적 예
- 수치 ↔ 기하학 연결

3차원으로의 확장

3D 내적:

```
v · w = v₁w₁ + v₂w₂ + v₃w₃
```

기하학적 의미:

```
= (v를 w로 투영) × |w|
```

쌍대성:

```
1×3 행렬 [w₁  w₂  w₃]
```

변환:

```
3D → 1D
```

모든 개념이 자연스럽게 확장!

다음 장 예고: 외적

외적 (Cross Product):

- 3D에서만 존재
- 두 벡터 → 벡터
- 쌍대성의 또 다른 예

공식이 복잡해 보이지만:

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

쌍대성으로 이해하면 자연스러움!

핵심 직관 정리

1. 내적의 세 얼굴:

```
수치적: v₁w₁ + v₂w₂ + ...
기하학적: (투영 길이) × |w|
변환적: 1×n 행렬 적용
```

2. 쌍대성:

```
벡터 ↔ nD → 1D 선형변환
```

3. 투영 = 선형변환:

```
특정 방향으로의 투영은 선형적
```

4. 좌표의 의미:

```
벡터 u의 좌표 = 기저벡터를 u로 투영한 값
```

5. 공식의 기원:

```
v · w = [w₁  w₂] [v₁]   (선형변환)
                 [v₂]
      = 투영 × |w| (기하학)
```

---

2. 퀴즈

문제 1

내적의 세 가지 정의를 설명하시오: (1) 수치적 정의, (2) 기하학적 정의(투영), (3) 선형변환 관점. 그리고 이 세 정의가 왜 동등한지 쌍대성의 개념을 사용하여 서술하시오.

문제 2

벡터 v와 w의 내적 v · w가:

- 양수일 때
- 0일 때
- 음수일 때 각각 무엇을 의미하는지 기하학적으로 설명하시오. 특히 투영의 관점에서 부호가 결정되는 원리를 서술하시오.

문제 3

왜 v · w = w · v인지(내적의 교환법칙) 두 가지 관점에서 설명하시오: (a) 투영의 대칭성 관점 (b) 선형변환 관점 그리고 둘이 같은 이유를 논리적으로 서술하시오.

문제 4

쌍대성(duality)의 개념을 설명하시오. 특히: (a) 일반적으로 쌍대성이란 무엇인가? (b) 선형대수에서 벡터와 선형변환 (nD → 1D)의 쌍대 관계는 무엇인가? (c) 단위벡터 û = [u_1, u_2]에서 u_1과 u_2가 나타내는 기하학적 의미를 쌍대성 관점에서 설명하시오.

문제 5

내적 공식 v · w = v_1w_1 + v_2w_2가 왜 "v를 w 방향으로 투영한 길이 × |w|"와 같은지 선형변환의 관점에서 증명하시오. 특히 기저벡터 î, ĵ의 투영과 선형성을 이용하여 단계별로 서술하시오.

---

3. 퀴즈 답

문제 1 답안

내적의 세 가지 정의:

---

(1) 수치적 정의:

공식:

```
v · w = v₁w₁ + v₂w₂ + ... + vₙwₙ
```

과정:

1. 각 좌표 쌍을 곱함
2. 모든 곱을 더함

예 (2D):

```
[3, 1] · [2, 4] = 3×2 + 1×4 = 6 + 4 = 10
```

특징:

- 계산이 간단
- 좌표에 의존
- "왜?"는 알기 어려움

---

(2) 기하학적 정의 (투영):

공식:

```
v · w = (v를 w로 투영한 길이) × |w|
```

또는 대칭적으로:

```
v · w = (w를 v로 투영한 길이) × |v|
```

시각화:

```
       v
      ↗
     /
    /
   /__________ w
  /   ↓
proj
```

투영 길이:

- v의 끝에서 w에 수직선
- 원점에서 수직선까지 거리

부호:

- 같은 방향 (예각): 양수
- 수직: 0
- 반대 방향 (둔각): 음수

특징:

- 기하학적 직관
- 좌표 무관
- 순서 대칭성 명확

---

(3) 선형변환 관점:

1×n 행렬로 표현:

```
w = [w₁, w₂, ..., wₙ]

쌍대 변환: [w₁  w₂  ...  wₙ] (1×n 행렬)
```

변환 적용:

```
[w₁  w₂  ...  wₙ] [v₁]
                    [v₂]  = w₁v₁ + w₂v₂ + ... + wₙvₙ
                    [⋮ ]
                    [vₙ]
```

의미:

- nD → 1D 선형변환
- w의 쌍대 변환을 v에 적용

특징:

- 변환 이론과 연결
- 선형성 명확
- 쌍대성의 예

세 정의의 동등성:

쌍대성 (Duality)을 통한 연결:

핵심 개념:

```
벡터 ↔ nD → 1D 선형변환
```

단계별 증명:

Step 1: 투영은 선형변환

투영 연산:

```
v를 û 방향으로 투영 (û는 단위벡터)
```

선형성 확인:

```
proj(v + w) = proj(v) + proj(w)  ✓
proj(cv) = c·proj(v)  ✓
```

따라서 투영 = 선형변환

Step 2: 선형변환 = 행렬

2D → 1D 선형변환:

```
기저벡터의 도착지로 완전 결정
î → u₁
ĵ → u₂

행렬: [u₁  u₂]
```

Step 3: 기저 투영 = 좌표

단위벡터 $\hat{u} = [u_1, u_2]$로의 투영:

```
u₁ = î를 û로 투영한 값
u₂ = ĵ를 û로 투영한 값
```

Step 4: 선형결합 보존

임의의 $v = x\hat{i} + y\hat{j}$:

```
투영(v) = 투영(xî + yĵ)
        = x·투영(î) + y·투영(ĵ)  (선형성)
        = x·u₁ + y·u₂
```

행렬 형태:

```
[u₁  u₂] [x]   = x·u₁ + y·u₂
         [y]
```

Step 5: 스케일링 고려

일반 벡터 $w = [w_1, w_2]$:

```
w = |w| · û (û는 단위벡터)

v · w = |w| · (v · û)
      = |w| × (v를 û로 투영)
      = (v를 w로 투영) × |w|  (기하학적)
```

수치적:

```
[w₁  w₂] [v₁]   = w₁v₁ + w₂v₂  (변환)
         [v₂]
```

결론:

```
수치적 정의 = 선형변환 = 기하학적 투영
```

쌍대성의 역할:

벡터 w:

1. **기하학적 객체** (화살표)
   - 투영의 방향
2. **선형 함수** (1×n 행렬)
   - nD → 1D 변환

이 두 관점이 **자연스럽게 대응**:

```
w의 좌표 = 변환 행렬의 원소
         = 기저벡터의 투영값
```

쌍대성 덕분에:

- 수치 계산이 기하학적 의미를 가짐
- 투영이 행렬 곱셈으로 표현됨
- 세 정의가 하나로 통합

이것이 내적의 진정한 아름다움입니다!

문제 2 답안

내적의 부호와 기하학적 의미:

기본 공식 (투영):

```
v · w = (v를 w로 투영한 길이) × |w|
```

---

(1) $v \cdot w > 0$ (양수):

기하학적 조건:

```
v와 w 사이 각도 θ < 90° (예각)
```

시각화:

```
     v
    ↗
   /
  /  θ < 90°
/_________ w
   ↓
  proj > 0
```

투영 길이:

- v의 끝에서 w에 수직선을 그음
- 수직선이 w의 **양의 방향**에 떨어짐
- 투영 길이 > 0

내적:

```
v · w = (양수) × |w| > 0
```

의미:

- 두 벡터가 **같은 방향**을 향함
- "협력적" 관계
- 한 벡터가 다른 벡터 방향으로 "진행"

구체적 예:

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

v · w = 3×2 + 1×1 = 7 > 0

각도: cos⁻¹(7 / (√10 × √5)) ≈ 18.4° < 90° ✓
```

---

(2) v · w = 0:

기하학적 조건:

```
v와 w 사이 각도 θ = 90° (직각)
```

시각화:

```
v
↑
|
|  θ = 90°
└──────→ w
  ↓
 proj = 0
```

투영 길이:

- v의 끝에서 w에 수직선을 그음
- 수직선이 **원점**에 떨어짐
- 투영 길이 = 0

내적:

```
v · w = 0 × |w| = 0
```

의미:

- 두 벡터가 **직교(perpendicular)**
- 서로 "독립적"
- 한 벡터가 다른 벡터 방향으로 전혀 진행 안 함

수학적 용어:

- Orthogonal (직교)
- 매우 중요한 개념!

구체적 예:

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

v · w = 3×2 + (-2)×3 = 6 - 6 = 0 ✓

각도: 90° ✓
```

응용:

- 직교 기저
- 직교 분해
- 그람-슈미트 과정

---

(3) v · w < 0 (음수):

기하학적 조건:

```
v와 w 사이 각도 θ > 90° (둔각)
```

시각화:

```
v ↖
   \
    \  θ > 90°
     \_________ w
       ↓
      proj < 0
```

투영 길이:

- v의 끝에서 w에 수직선을 그음
- 수직선이 w의 **음의 방향**에 떨어짐
- 투영 길이 < 0 (부호가 음!)

내적:

```
v · w = (음수) × |w| < 0
```

의미:

- 두 벡터가 **반대 방향**을 향함
- "대립적" 관계
- 한 벡터가 다른 벡터와 반대로 "진행"

극단적 경우 (θ = 180°):

```
v와 w가 정반대
v · w = -|v| × |w| (최소값)
```

구체적 예:

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

v · w = 1×(-2) + 2×(-1) = -2 - 2 = -4 < 0

각도: cos⁻¹(-4 / (√5 × √5)) = cos⁻¹(-4/5) ≈ 143° > 90° ✓
```

투영의 부호 결정 원리:

수식적:

```
v · w = |v| |w| cos(θ)
```

cos(θ)의 부호:

```
θ < 90°: cos(θ) > 0 → v · w > 0
θ = 90°: cos(θ) = 0 → v · w = 0
θ > 90°: cos(θ) < 0 → v · w < 0
```

기하학적 직관:

투영 = "그림자"

태양이 w 방향에서 비춤:

```
w 방향 ←☀

v의 그림자가 w 위에 어디?
```

예각:

- 그림자가 w의 양의 방향
- 투영 > 0

직각:

- 그림자가 원점
- 투영 = 0

둔각:

- 그림자가 w의 음의 방향
- 투영 < 0

요약 표:

각도	내적 부호	투영 방향	벡터 관계
θ < 90°	> 0	양의 방향	같은 방향
θ = 90°	= 0	원점	직교
θ > 90°	< 0	음의 방향	반대 방향

결론: 내적의 부호는 두 벡터의 상대적 방향을 나타내며, 이는 투영의 방향으로 기하학적으로 이해할 수 있습니다.

문제 3 답안

내적의 교환법칙: v · w = w · v

(a) 투영의 대칭성 관점:

두 가지 투영:

방법 1: v를 w로 투영

```
        v
       ↗
      /
     /
    /__________ w
   /  ↓
  p₁  (투영 길이)

 v · w = p₁ × |w|
```

방법 2: w를 v로 투영

```
        w
       ↗
      /
     /
    /__________ v
   /  ↓
  p₂  (투영 길이)

 w · v = p₂ × |v|
```

질문: 왜 같은가?

기하학적 증명 (평행사변형 이용):

1. 평행사변형 구성:

```
      D ────── C
     /|       /|
    / |      / |
   /  |h    /  |
  A ──|──── B  |
      |   /    |
      |  /     |
      | /      |
      |/       |
      O
```

- O = 원점
- OA = v
- OB = w
- OACB = 평행사변형

2. 넓이 계산 (v 기준):

```
넓이 = |w| × h₁
```

여기서 h_1 = v를 w에 수직인 방향으로 측정한 높이

하지만:

$$h_1 = |v| \sin(\theta)$$

따라서:

$$넓이 = |w| \times |v| \sin(\theta)$$

3. 넓이 계산 (w 기준):

$$넓이 = |v| \times h_2$$

여기서 h_2 = w를 v에 수직인 방향으로 측정한 높이

$$h_2 = |w| \sin(\theta)$$

따라서:

$$넓이 = |v| \times |w| \sin(\theta)$$

4. 같은 넓이:

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

내적 연결:

투영을 이용한 넓이:

$$넓이^2 = |v|^2\ |w|^2 - (v \cdot w)^2$$

(이것은 라그랑주 항등식)

대칭성이 보장됨!

더 직접적인 증명:

공식:

$$v \cdot w = |v|\ |w| \cos(\theta)$$
$$w \cdot v = |w|\ |v| \cos(\theta)$$

곱셈의 교환법칙:

$$|v|\ |w| = |w|\ |v|$$

따라서:

```
v · w = w · v ✓
```

(b) 선형변환 관점:

변환으로 표현:

v · w (w의 변환을 v에 적용):

```
[w₁  w₂] [v₁]   = w₁v₁ + w₂v₂
         [v₂]
```

w · v (v의 변환을 w에 적용):

```
[v₁  v₂] [w₁]   = v₁w₁ + v₂w₂
         [w₂]
```

같은 이유:

수의 곱셈 교환법칙:

```
w₁v₁ = v₁w₁
w₂v₂ = v₂w₂
```

따라서:

```
w₁v₁ + w₂v₂ = v₁w₁ + v₂w₂
```

행렬 표기법:

```
[w₁  w₂] [v₁]   = [v₁  v₂] [w₁]
         [v₂]              [w₂]
```

(c) 두 관점이 같은 이유:

핵심 연결: 쌍대성

1. 투영과 변환의 동일성:

벡터 w로의 투영:

```
기하학적: (v를 w로 투영) × |w|
변환적: [w₁  w₂] [v₁]
                 [v₂]
```

둘이 같음! (쌍대성)

2. 대칭성의 보존:

기하학적 대칭성:

```
평행사변형은 대칭적
→ 투영 관계 대칭적
```

대수적 대칭성:

```
수의 곱셈 교환법칙
→ 행렬 곱셈에서도 교환
```

3. 쌍대성이 보장:

```
기하학 ←쌍대성→ 대수
   ↓               ↓
 대칭적          대칭적
```

쌍대성 덕분에:

- 기하학적 대칭성 = 대수적 대칭성
- 투영의 순서 무관 = 곱셈의 순서 무관

직관적 이해:

투영의 본질:

```
v와 w의 "얼마나 같은 방향?"
```

방향성은 상대적:

- v가 w 방향을 보는 것
- w가 v 방향을 보는 것
- 같은 "방향 일치도"

변환의 본질:

```
한 벡터를 다른 벡터의 "좌표계"로 변환
```

좌표계 변환은 상대적:

- v를 w 좌표로
- w를 v 좌표로
- 같은 "상대적 위치"

결론:

$v \cdot w = w \cdot v$인 이유:

1. **투영 관점:**
   - 평행사변형의 기하학적 대칭성
   - $|v|\,|w| \cos(\theta)$의 곱셈 교환법칙
2. **변환 관점:**
   - 수의 곱셈 교환법칙
   - $w_1v_1 + w_2v_2 = v_1w_1 + v_2w_2$
3. **통합:**
   - 쌍대성이 두 관점을 연결
   - 기하학적 대칭성 = 대수적 대칭성

이것이 내적의 교환법칙이 성립하는 깊은 이유입니다!

문제 4 답안

쌍대성(Duality)의 개념:

(a) 쌍대성의 일반적 정의:

정의:

```
서로 다른 두 종류의 수학적 대상 사이의
자연스럽지만 놀라운 대응 관계
```

핵심 요소:

1. 두 종류의 대상:

- 타입 A와 타입 B
- 겉보기에는 전혀 다름

2. 자연스러운 대응:

- A ↔ B 사이의 일대일 대응
- 구조를 보존하는 방식

3. 놀라운 연결:

- 직관적으로 예상 못함
- 하지만 깊이 생각하면 필연적

예시 (다른 분야):

기하학:

- 점 ↔ 선 (사영기하학)
- 내부 ↔ 외부 (위상수학)

논리:

- AND ↔ OR (불 대수)
- ∀ ↔ ∃ (양화사)

물리:

- 위치 ↔ 운동량 (푸리에 변환)
- 시간 ↔ 주파수

공통점:

- 두 관점이 상호 변환 가능
- 한쪽에서 어려운 것이 다른 쪽에서 쉬움
- 더 깊은 통찰 제공

(b) 선형대수의 쌍대성:

벡터와 선형변환의 대응:

타입 A: 벡터

```
v = [v₁, v₂, ..., vₙ]
```

- n차원 공간의 화살표
- 기하학적 객체
- 방향과 크기

타입 B: 선형변환 (nD → 1D)

```
f: ℝⁿ → ℝ
```

- 벡터를 숫자로 변환
- 함수적 객체
- 행렬로 표현: $[v_1\ v_2\ \ldots\ v_n]$

자연스러운 대응:

벡터 v의 쌍대:

```
v = [v₁, v₂, ..., vₙ]

쌍대 v* = 선형변환 fᵥ
```

여기서:

```
fᵥ(w) = v · w
```

선형변환 f의 쌍대:

```
f: ℝⁿ → ℝ  (선형)

쌍대 f* = 벡터 v
```

여기서:

```
f(w) = v · w  (모든 w에 대해)
```

양방향 대응:

```
벡터 ↔ nD → 1D 선형변환
```

구조 보존:

```
(v + w)* = v* + w*  (가법성)
(cv)* = c·v*  (스케일링)
```

(c) 단위벡터 $\hat{u} = [u_1, u_2]$의 좌표 의미:

설정:

```
û = [u₁, u₂]  (|û| = 1, 단위벡터)
```

쌍대 변환:

```
f_û(v) = û · v
```

행렬 표현:

```
[u₁  u₂]
```

기하학적 의미:

```
v를 û 방향으로 투영
```

---

u_1의 의미:

정의:

```
u₁ = î를 û로 투영한 값
```

상세 설명:

1. 표준 기저벡터 î:

```
î = [1, 0]
```

2. û로 투영:

```
proj_û(î) = (î · û) / |û|
         = (1·u₁ + 0·u₂) / 1
         = u₁
```

3. 기하학적 그림:

```
    û
   ↗
  /
 / u₁
/___î___→
    ↑
   u₁
```

4. 쌍대성 관점:

변환 행렬 [u_1 u_2]:

- 첫 번째 열 원소 = î의 도착지
- î → u_1

5. 좌표 의미:

```
u₁ = û의 x 성분
   = û이 x축으로 얼마나 가는가
   = x축을 û로 투영한 값
```

---

u_2의 의미:

정의:

```
u₂ = ĵ를 û로 투영한 값
```

상세 설명:

1. 표준 기저벡터 ĵ:

```
ĵ = [0, 1]
```

2. û로 투영:

```
proj_û(ĵ) = (ĵ · û) / |û|
         = (0·u₁ + 1·u₂) / 1
         = u₂
```

3. 기하학적 그림:

```
  ĵ     û
  ↑    ↗
  |   /
  | / u₂
  |/_____
     ↑
    u₂
```

4. 쌍대성 관점:

변환 행렬 [u_1 u_2]:

- 두 번째 열 원소 = ĵ의 도착지
- ĵ → u_2

5. 좌표 의미:

```
u₂ = û의 y 성분
   = û이 y축으로 얼마나 가는가
   = y축을 û로 투영한 값
```

쌍대성의 심오한 의미:

벡터의 좌표 = 기저의 투영값

일반 벡터 v = [x, y]:

```
v = xî + yĵ
```

û로 투영:

```
proj_û(v) = proj_û(xî + yĵ)
          = x·proj_û(î) + y·proj_û(ĵ)  (선형성!)
          = x·u₁ + y·u₂
```

행렬 형태:

```
[u₁  u₂] [x]   = x·u₁ + y·u₂
         [y]
```

결론:

u_1, u_2는:

1. **벡터 관점:**
   - û의 좌표 성분
2. **변환 관점:**
   - 기저벡터의 도착지
   - 변환 행렬의 원소
3. **투영 관점:**
   - 기저벡터를 û로 투영한 값

쌍대성 덕분에 이 세 관점이 하나로 통합!

직관:

```
"벡터의 좌표"는
"기저를 그 방향으로 투영한 값"
```

이것이 쌍대성의 핵심 통찰입니다!

문제 5 답안

내적 공식의 증명 (선형변환 관점):

증명할 명제:

```
v · w = v₁w₁ + v₂w₂  (수치)
     = (v를 w로 투영) × |w|  (기하학)
```

증명 전략:

1. 투영을 선형변환으로 이해
2. 기저벡터 투영 확인
3. 선형성으로 일반화

Step 1: **단위벡터의 경우 (|w| = 1)**

설정:

```
û = [u₁, u₂]  (|û| = 1)
```

투영 연산:

```
v를 û 방향으로 투영 → 숫자
```

목표: 이것이 선형변환임을 보이고, 행렬을 찾기

Step 2: 기저벡터 투영

î = [1, 0]를 û로 투영:

기하학적으로:

```
proj_û(î) = u₁
```

이유:

- î는 x축 단위벡터
- û = $[u_1, u_2]$
- u_1 = û의 x 성분 = x축으로의 투영

ĵ = [0, 1]를 û로 투영:

기하학적으로:

```
proj_û(ĵ) = u₂
```

이유:

- ĵ는 y축 단위벡터
- u_2 = û의 y 성분 = y축으로의 투영

Step 3: 선형변환 행렬

기저벡터의 도착지:

```
î → u₁
ĵ → u₂
```

2D → 1D 선형변환 행렬:

```
[u₁  u₂]
```

의미:

- 첫 번째 열: î의 도착지

- 두 번째 열: ĵ의 도착지

Step 4: 임의 벡터 투영

벡터 v = [v_1, v_2]:

선형결합으로 표현:

```
v = v₁î + v₂ĵ
```

투영 (선형성 이용):

```
proj_û(v) = proj_û(v₁î + v₂ĵ)

선형성:
          = v₁·proj_û(î) + v₂·proj_û(ĵ)
          = v₁·u₁ + v₂·u₂
```

행렬 형태:

```
[u₁  u₂] [v₁]   = v₁u₁ + v₂u₂
         [v₂]
```

결론 (단위벡터):

```
v · û = v₁u₁ + v₂u₂ = proj_û(v)
```

Step 5: 일반 벡터로 확장

일반 벡터 w = [w_1, w_2] (|w| ≠ 1):

단위벡터 분해:

```
w = |w| · û

여기서 û = w / |w| = [w₁/|w|, w₂/|w|]
```

û의 성분:

```
u₁ = w₁ / |w|
u₂ = w₂ / |w|
```

투영:

```
proj_û(v) = v₁u₁ + v₂u₂
          = v₁(w₁/|w|) + v₂(w₂/|w|)
          = (v₁w₁ + v₂w₂) / |w|
```

w 방향 투영:

```
proj_w의 길이 = |w| × proj_û(v)
              = |w| × (v₁w₁ + v₂w₂) / |w|
              = v₁w₁ + v₂w₂
```

내적:

```
v · w = (v를 w로 투영한 길이) × |w|
      = proj_w(v) × |w|
      = (v₁w₁ + v₂w₂) / |w| × |w|
      = v₁w₁ + v₂w₂
```

선형변환 관점 재정리:

w의 쌍대 변환:

```
f_w(v) = [w₁  w₂] [v₁]
                   [v₂]
```

기저벡터:

```
î → w₁
ĵ → w₂
```

임의 벡터 $\mathbf{v = v_1\hat{i} + v_2\hat{j}}$:

```
f_w(v) = v₁·w₁ + v₂·w₂
```

기하학적 의미:

```
= v를 w/|w| 방향으로 투영 × |w|
= (v를 w로 투영) × |w|
```

완전한 증명:

명제:

```
v · w = v₁w₁ + v₂w₂ = (v를 w로 투영) × |w|
```

증명:

1. **투영 = 선형변환**
   - 가법성: proj(u + v) = proj(u) + proj(v)
   - 스케일링: proj(cu) = c·proj(u)
2. **기저 투영:**
   - proj_w(î) = w_1 / |w| × |w| = w_1
   - proj_w(ĵ) = w_2 / |w| × |w| = w_2
3. **변환 행렬:**
   - [w_1 w_2]
4. **선형성 적용:**
   ```
   v = v₁î + v₂ĵ

   v · w = [w₁  w₂] [v₁]
                     [v₂]
         = v₁w₁ + v₂w₂  (수치)
   ```
5. **기하학적 해석:**
   ```
   = (v를 w/|w|로 투영) × |w|
   = (v를 w로 투영) × |w|  (기하학)
   ```

Q.E.D.

핵심 통찰:

쌍대성이 연결:

```
수치 계산 ← [선형변환] → 기하학적 투영
```

w의 좌표 = 기저의 투영값

```
w₁ = î를 w로 투영
w₂ = ĵ를 w로 투영
```

선형결합이 보존됨:

```
v = v₁î + v₂ĵ

투영(v) = v₁·투영(î) + v₂·투영(ĵ)
        = v₁w₁ + v₂w₂
```

이것이 내적 공식의 진정한 의미입니다!

```
Dataview (inline field '(v를 u 방향으로 투영) × |u|'): Error:
-- PARSING FAILED --------------------------------------------------
```

```
> 1 | (v를 u 방향으로 투영) × |u|
    |     ^

Expected one of the following:

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

```
Dataview (inline field 'v를 w로 투영 × |w|'): Error:
-- PARSING FAILED --------------------------------------------------

> 1 | v를 w로 투영 × |w|
    |    ^

Expected one of the following:

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

```
Dataview (inline field 'v₁w₁ + v₂w₂'): Error:
-- PARSING FAILED --------------------------------------------------

> 1 | v₁w₁ + v₂w₂
    |  ^

Expected one of the following:

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

```
Dataview (inline field '(v를 w로 투영) × |w|'): Error:
-- PARSING FAILED --------------------------------------------------

> 1 | (v를 w로 투영) × |w|
    |     ^

Expected one of the following:

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

```
Dataview (inline field 'v를 w/|w| 방향으로 투영 × |w|
= (v를 w로 투영) × |w|'): Error:
-- PARSING FAILED --------------------------------------------------

> 1 | v를 w/|w| 방향으로 투영 × |w|
```

```
    |    ^
  2 | = (v를 w로 투영) × |w|

Expected one of the following: 

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

```
Dataview (inline field '(v를 w/|w|로 투영) × |w|
= (v를 w로 투영) × |w|  (기하학)'): Error: 
-- PARSING FAILED --------------------------------------------------

> 1 | (v를 w/|w|로 투영) × |w|
    |      ^
  2 | = (v를 w로 투영) × |w|  (기하학)

Expected one of the following: 

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