3. 고유값(eighenvalues) 및 고유벡터(eigenvectors)

1. 고유벡터

   고유벡터는 행렬의 변환에서 방향이 변하지 않는 벡터입니다. 보통 일반 벡터는 어떤 변환을 거치면 방향이 바뀝니다. 그러나 고유벡터는 특정한 조건 하에서 방향이 변하지 않고, 단지 크기만 변합니다.

   이것을 공식으로 나타내면

   $$Av = \lambda v$$

   - A : 정방행렬 (예: 2x2, 3×3 같은 사각형 행렬)
   - v : 고유벡터 (행렬 A에 의해 방향이 변하지 않는 벡터)
   - λ : 고유값 (변환 후 크기가 변화된 정도)

   고유벡터는 방향을 유지하면서 크기만 변하는 벡터입니다. 이때 변하는 크기가 고유값입니다.

2. 고유값

   고유값은 고유벡터가 변환될 때 얼마나 크기(스케일)가 변했는지를 나타내는 값입니다. 즉, 고유벡터가 행렬 변환을 받았을 때 벡터의 크기가 얼마나 커지거나 작아지는지를 나타내는 값이 고유값입니다.

   - 양수 고유값 : 고유벡터의 방향은 변하지 않고 크기만 커지거나 작아짐
   - 음수 고유값 : 고유벡터의 방향이 반대로 뒤집어짐
   - 0인 고유값 : 벡터가 완전히 사라지거나 변환 후 0이 됨.

3. 고유값과 고유벡터 구하기

   행렬 A의 고유값과 고유벡터를 구하는 방법은 특성 방정식을 푸는 것입니다.

   특정 방저식은 다음과 같습니다

   $$det(A - \lambda I) = 0$$

- λ는 고유값
- I는 단위행렬
- det는 행렬의 행렬식을 의미합니다.

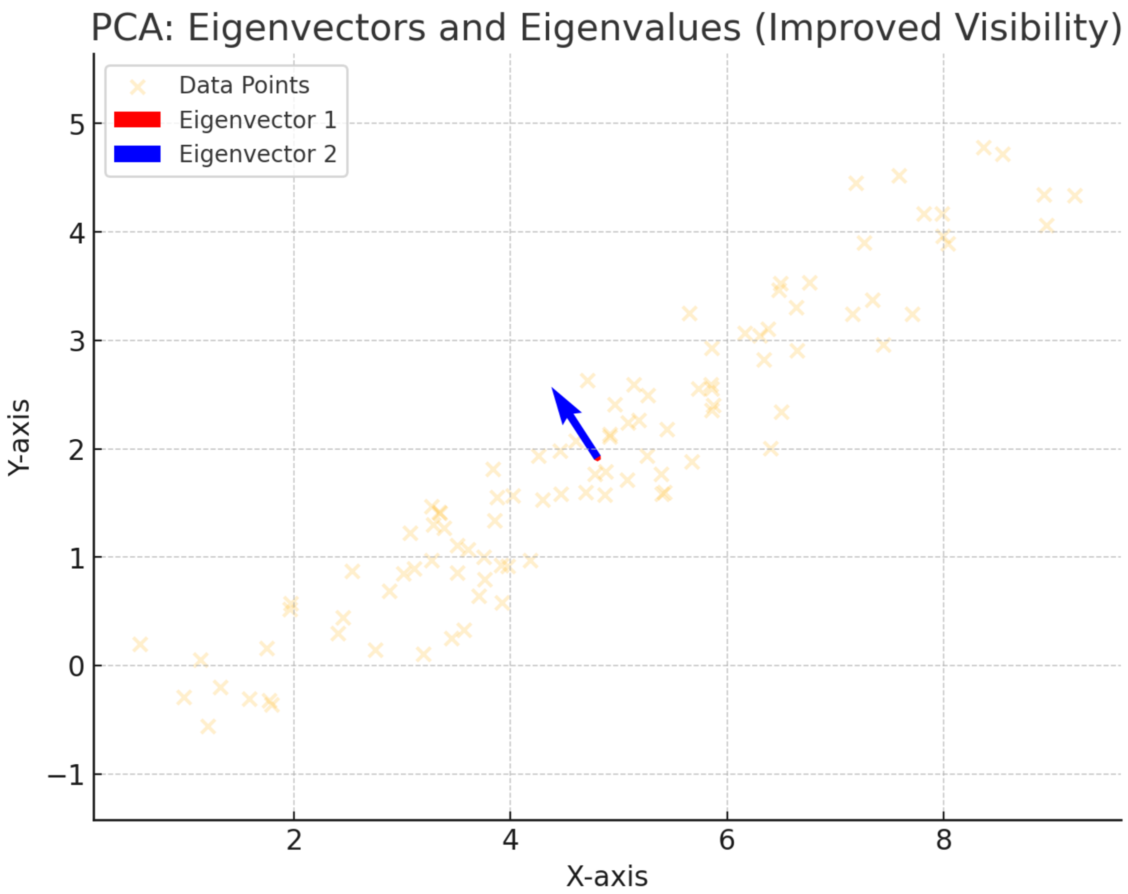
이 특성 방정식을 풀면 행렬 A의 고유값 λ를 구할 수 있습니다. 고유값을 알면 그에 대응하는 고유벡터 v를 구할 수 있습니다.

4. 고유값 및 고유벡터의 기하학적 의미

고유값과 고유벡터를 기하학적으로 보면, 행렬은 벡터를 변환시키는 역할을 합니다. 대부분의 벡터는 행렬 변환을 받으면 방향과 크기가 모두 바뀝니다. 하지만 고유벡터는 방향이 변하지 않고, 크기만 고유값만큼 변화하게 됩니다.

이 개념은 다음과 같은 예시로 생각할 수 있습니다.

- 행렬은 벡터를 공간에서 회전하거나 늘리거나 줄이는 변환을 할 수 있습니다.
- 고유벡터는 그러한 변환속에서도 방향을 유지하는 벡터입니다.
- 고유값은 이 벡터가 얼마나 늘어나거나 줄어들었는지를 나타내는 값입니다.



- 붉은색 화살표는 첫번째 고유벡터(주성분 1), 파란색화살표는 두번째 고유벡터(주성분2)
- 붉은벡터 → 데이터 분산이 가장 큰 방향
- 파란벡터 → 그보다 작은 분산을 설명하는 방향

고유값과 고유벡터의 활용

1. PCA (주성분 분석)

   PCA는 데이터를 더 적은 차원으로 축소하는 차원 축소 방법입니다. PCA에서 고유값과 고유벡터는 매우 중요한 역할을 합니다.

   - 고유값 : 데이터의 분산을 설명하는데, 큰 고유값은 해당 방향(축)이 데이터의 분산이 크다는 것을 의미합니다.
   - 고유벡터 : 고유값에 대응하는 고유벡터는 그 데이터의 주성분 축을 나타냅니다. 즉, 고유값이 큰 고유벡터들은 데이터의 분산이 큰 방향을 나타내며, 이 축들을 기준으로 데이터를 차원 축소 합니다.

2. 딥러닝에서의 사용

   딥러닝에서도 고유값은 모델의 학습 과정이나 가중치 행렬의 특성을 분석하는 데 사용할 수 있습니다.
   예를 들어, 학습 중에 가중치 행렬의 고유값 분포가 신경망의 수렴 속도나 안정성에 영향을 미칠 수 있습니다. 큰 고유값이 있는 경우, 그 방향으로 매우 빠르게 수렴하거나, 반대로 너무 크면 불안정해 질 수 있습니다.

고유값 및 고유벡터의 정의

$$A\mathbf{v} = \lambda\mathbf{v}$$

특성 방정식

$$\det(A - \lambda I) = 0$$

예시

- 2×2 행렬

$$A = \begin{pmatrix} 4 & 2 \\ 1 & 3 \end{pmatrix}$$
$$\det(A - \lambda I) = \det \begin{pmatrix} 4 - \lambda & 2 \\ 1 & 3 - \lambda \end{pmatrix} = 0 \;\cdots \text{특성방정식}$$
$$(4 - \lambda)(3 - \lambda) - (1)(2) = 0 \;\cdots \text{행렬식계산}$$
$$\lambda^2 - 7\lambda + 10 = 0 \;\cdots \text{전개}$$

- 고유값 계산
  - 근의공식 사용한 계산

$$\lambda = \frac{-(-7) \pm \sqrt{(-7)^2 - 4(1)(10)}}{2(1)}$$
$$\lambda = \frac{7 \pm \sqrt{49 - 40}}{2} = \frac{7 \pm \sqrt{9}}{2}$$
$$\lambda_1 = \frac{7 + 3}{2} = 5, \quad \lambda_2 = \frac{7 - 3}{2} = 2$$

- 첫번째 고유벡터 계산
  - 고유값 $\lambda_1 = 5$에 대해, 다음 방정식을 계산

$$A - 5I = \begin{pmatrix} 4 & 2 \\ 1 & 3 \end{pmatrix} - 5 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} -1 & 2 \\ 1 & -2 \end{pmatrix}$$
$$\text{방정식 및 연립방정식}$$
$$\begin{pmatrix} -1 & 2 \\ 1 & -2 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$
$$-1 \cdot v_1 + 2 \cdot v_2 = 0 \quad \Rightarrow \quad v_1 = 2v$$
$$\text{결과}$$
$$\mathbf{v}_1 = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

- 두번째 고유벡터 계산
  - 고유값 $\lambda_2 = 2$

$$A - 2I = \begin{pmatrix} 4 & 2 \\ 1 & 3 \end{pmatrix} - 2\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 2 & 2 \\ 1 & 1 \end{pmatrix}$$

방정식및연립방정식

$$\begin{pmatrix} 2 & 2 \\ 1 & 1 \end{pmatrix}\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$2v_1 + 2v_2 = 0 \quad \Rightarrow \quad v_1 = -v_2$$

결과

$$\mathbf{v}_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

```
import numpy as np

# 2x2 행렬 정의
A = np.array([[4, 2],
              [1, 3]])

# 고유값 및 고유벡터 계산
eigenvalues, eigenvectors = np.linalg.eig(A)

# 고유값 출력
print("고유값:")
print(eigenvalues)

# 고유벡터 출력
print("\n고유벡터:")
print(eigenvectors)

# 고유값 : [5. 2.]
'''
고유벡터 = [[ 0.89442719 -0.70710678]
                        [ 0.4472136   0.70710678]]
'''
```

- 고유값이 λ =5 와 λ = 2 입니다. 즉, 이 행렬은 어떤 벡터의 크기를 5배 또는 2배로 변환합니다.
- 대응하는 고유벡터는 각각의 고유 값에 대해 방향을 유지하는 벡터입니다. 첫 번째 고유벡터는 [0.984, 0.447]로, 이 벡터의크기는 5배로 변환됩니다. 두 번째 고유벡터는 [-0.707, 0.707]로, 이 벡터의 크기는 2배로 변환 됩니다.