3-8. Orthogonal Projections

Projections(정사영, 투영, 투사)은 선형 변환(Linear transformation)의 중요한 종류입니다. rotation, reflection 처럼 그래픽스, coding theory, 통계학, 머신러닝에서 중요한 역할을 합니다.

머신러닝에서는 고차원에 데이터를 다룹니다. 고차원 데이터는 분석 및 시각화가 어려운 경우가 많습니다.

하지만 실제로 고차원 데이터는 소수의 차원만이 대부분의 정보를 담고 있고, 나머지 많은 차원은 데이터의 핵심 특성을 설명하는 데 꼭 필요하지 않는 경우가 많습니다.

고차원 데이터를 압축하거나 시각화할 때, 정보 손실이 발생합니다. 이러한 손실을 최소화하기 위해 가장 정보가 많은 차원(informative dimensions)을 찾는 것이 이상적일 것입니다.

Ch.1에서 언급했듯이, 데이터는 벡터로 표현될 수 있고, 이번 장에서는 데이터 압축의 핵심 도구에 대해서 다룹니다.

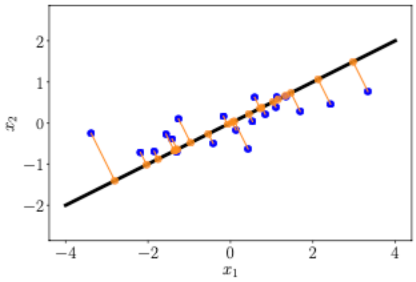
구체적으로 말하면, 원래의 고차원 데이터를 더 낮은 차원의 특징 공간(feature space)로 투영(projections)하여 낮은 차원의 공간에서 데이터를 분석하고 의미 있는 패턴을 추출할 수 있습니다.

예를 들어, PCA 또는 DNN과 같은 머신러닝 알고리즘은 차원 축소를 많이 활용합니다.

이번 장에서는 직교 투영(orthogonal projections)에 집중할 것이며, 이는 Ch.10, Ch.12에 활용됩니다.

또한 선형회귀도 직교 투영의 관점에서 해석할 수 있으며, 이는 Ch9에서 다룰 예정입니다.

Figure 3.9 Orthogonal projection (orange dots) of a two-dimensional dataset (blue dots) onto a one-dimensional subspace (straight line).



주어진 저차원 부분공간에 대해, 직교 투영은 고차원 데이터에서 가능한 많은 정보를 보존하며, 원래 데이터와 투영된 데이터 간의 차이(오차)를 최소화합니다.

예시

학생	국어	수학	과학	영어	체육	음악	미술
A	70	95	98	80	60	55	50
B	65	90	96	75	65	50	45
C	80	97	99	85	58	52	53
D	60	92	95	78	62	48	47

만약 우리가 이과 과목에 대해서만 분석하고자 한다면 수학/과학/영어 성적이 가장 중요한 차원일 수 있습니다.

반대로 음악/미술/체육/기술 가정 등은 이과 분석에 큰 영향을 주지 않을 수 있습니다. → 정보량이 낮은 차원

학생	수학	과학	영어
A	95	98	80
B	90	96	75
C	97	99	85
D	92	95	78

이렇게 고차원의 데이터를 저차원으로 압축하여 데이터를 분석하고 시각화하는 것이 쉬워질 것입니다.

정의 3.10 (Projection)

Definition 3.10 (Projection). Let V be a vector space and $U \subseteq V$ a subspace of V. A linear mapping $\pi : V \to U$ is called a *projection* if $\pi^2 = \pi \circ \pi = \pi$.

- $\pi^2 = \pi$ 는 투영을 한 번 더 해도 결과과 변하지 않는다는 것입니다.
- 즉, 이미 한 번 투영된 벡터는 다시 투영해도 똑같습니다.

예시

$V = \mathbb{R}^3$ (3차원 공간), $U = xy$-평면 $= \{(x, y, 0)\}$

투영 사상 π

- 어떤 벡터 $v = (x, y, z) \in \mathbb{R}^3$ 를 $\pi(v) = (x, y, 0)$으로 정의하면,
  z 축 방향으로 평면에 수직 투영하는 것입니다.
- $\pi^2(v) = \pi(\pi(v)) = \pi(x, y, 0) = (x, y, 0) = \pi(v)$

용어	의미
V	전체 벡터 공간
$U \subseteq V$	투영하려는 저차원 부분공간
$\pi : V \to U$	벡터를 U로 "투영하는" 선형 함수
$\pi^2 = \pi$	이미 투영된 벡터는 다시 투영해도 변하지 않음

선형 사상은 transformation matrix로 표현될 수 있으므로, 앞서 정의한 투영(projection)의 정의는 특수한 형태의 행렬 (투영 행렬)에도 동일하게 적용됩니다.

이러한 투영 행렬 P_π는 다음 조건을 만족합니다.

$P_\pi^2 = P_\pi$

내적 공간(inner product space)인 $(R^n, \langle\cdot,\cdot\rangle)$에서의 벡터를 부분공간(subspace) U에 직교 투영하는 방법을 유도할 것입니다.

먼저 1차원 부분 공간(직선, line)에 대한 투영부터 알아봅니다. 그리고 별도의 언급이 없으면 inner product는 dot product로 정의합니다.
→ $\langle x, y\rangle = x^T y$

3.8.1 Projection onto One-Dimensional Subspaces(Lines)

원점(origin)을 지나는 직선(1차원 부분공간)이 주어지고, 이 직선은 기저 벡터 $b \in \mathbb{R}^n$로 생성된다고 가정합니다.

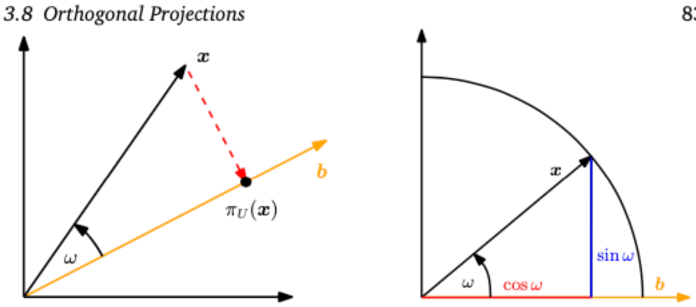
이 직선은 b에 의해 생성된 $\mathbb{R}^n$안의 1차원 부분공간 $U \subseteq \mathbb{R}^n$입니다.

우리가 벡터 $x \in \mathbb{R}^n$을 U 에 투영할 때, 목표는 **x와 가장 가까운 벡터 $\pi_U(x) \in U$를 찾는 것**입니다.

기하학적 관점에서, Projection $\pi_U(x)$는 다음 성질들을 가집니다.

- $\pi_U(x)$는 x와 가장 가까운 점이다. 즉, distance $||x - \pi_U(x)||$가 최소입니다.
- 따라서 $\pi_U(x)$에서 x까지의 벡터, $x - \pi_U(x)$ 는 부분공간 U, 벡터 b에 직교해야합니다.
- 직교 조건에 따라, 다음 내적 관계가 성립합니다.
  - $\langle x - \pi_U(x), b\rangle = 0$
- 또한 투영 벡터 $\pi_U(x)$ 는 부분공간 U 위에 존재하므로, 벡터 b의 스칼라 배수로 표현됩니다.
  - $\pi_U(x) = \lambda b,\ \lambda \in \mathbb{R}$

3.8 Orthogonal Projections 83



(a) Projection of $x \in \mathbb{R}^2$ onto a subspace U with basis vector b.

(b) Projection of a two-dimensional vector x with $\|x\| = 1$ onto a one-dimensional subspace spanned by b.

Figure 3.10 Examples of projections onto one-dimensional subspaces.

3단계를 통해, coordinate λ, projection $\pi_U(x), x \in \mathbb{R}^n$ 을 U로 mapping하는 projection matrix P_π 를 결정할 수 있습니다.

3단계 요약

단계	내용	결과
1단계	투영 좌표 λ 찾기	$\lambda = \frac{\langle x, b\rangle}{\langle b, b\rangle}$
2단계	투영 점 $\pi_U(x)$ 계산	$\pi_U(x) = \lambda b$
3단계	투영 행렬 P_π 유도	$P_\pi = \frac{bb^\top}{\|\|b\|\|^2}$

1. 투영 좌표 λ 구하기

▼ 왜 λ가 좌표가 될까?

$v = \begin{bmatrix}3\\5\end{bmatrix}$ 가 있을 때, 표준 기저 벡터 $e_1 = \begin{bmatrix}1\\0\end{bmatrix}$, $e_2 = \begin{bmatrix}0\\1\end{bmatrix}$을 통해

$v = 3e_1 + 5e_2$로 표현할 수 있습니다.

이때 좌표는 $(3, 5)$ 입니다.

벡터 $x \in \mathbb{R}^n$를 부분공간 $U = span(b)$ 위로 직교 투영하려고 합니다.

투영된 점은 $\pi_U(x) = \lambda b$ 형태입니다.

직교 조건과 내적의 bilinearity를 활용하여 다음의 식을 도출해 낼 수 있습니다.

$$\langle x - \lambda b, b\rangle = 0 \rightarrow \langle x, b\rangle - \lambda\langle b, b\rangle = 0 \rightarrow \lambda = \lambda = \frac{\langle x, b\rangle}{\langle b, b\rangle}$$

내적은 dot product로 정의했을 때, 다음의 식을 얻을 수 있습니다.

$$\lambda = \frac{b^T x}{b^T, b} = \frac{b^T x}{||b||^2}$$

만약 $||b|| = 1$ 인 경우, coordinate λ는 $b^T x$ 로 주어집니다.

2. 투영된 벡터 $\pi_U(x)$ 계산

$$\pi_U(x) = \lambda b = \frac{\langle x, b\rangle}{\langle b, b\rangle} b = \frac{b^T x}{||b||^2}$$

정의 3.1을 통해 $\pi_U(x)$ 길이를 다음과 같이 구할 수 있습니다.

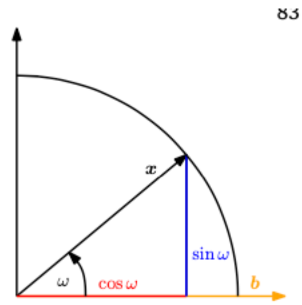
$$||\pi_U(x)|| = ||\lambda b|| = |\lambda|||b||$$

projection의 길이는 b의 길이에 λ배 한 것과 같습니다.

만약 내적으로 dot product를 사용한다면, 아래의 식을 얻을 수 있습니다.

$$\|\pi_U(\boldsymbol{x})\| \overset{(3.42)}{=} \frac{|\boldsymbol{b}^\top \boldsymbol{x}|}{\|\boldsymbol{b}\|^2} \|\boldsymbol{b}\| \overset{(3.25)}{=} |\cos\omega| \, \|\boldsymbol{x}\| \, \|\boldsymbol{b}\| \frac{\|\boldsymbol{b}\|}{\|\boldsymbol{b}\|^2} = |\cos\omega| \, \|\boldsymbol{x}\| \,. \tag{3.44}$$

w는 x와 b 사이의 각도입니다. 만약 $||x|| = 1$ 이라면 x는 unit circle에 놓여있으며, 이 때의 투영은 밑 그림에서 보여줍니다.



(b) Projection of a two-dimensional vector $\boldsymbol{x}$ with $\|\boldsymbol{x}\| = 1$ onto a one-dimensional subspace spanned by $\boldsymbol{b}$.

3. 투영 행렬 P_π 유도

선형 사상 $\pi_U(x) = P_\pi x$ 형태로 나타내고자 합니다.

dot product를 내적으로 사용한다면, 다음의 투영 행렬을 얻을 수 있습니다.

$$\pi_U(\boldsymbol{x}) = \lambda \boldsymbol{b} = \boldsymbol{b}\lambda = \boldsymbol{b}\frac{\boldsymbol{b}^\top \boldsymbol{x}}{\|\boldsymbol{b}\|^2} = \frac{\boldsymbol{b}\boldsymbol{b}^\top}{\|\boldsymbol{b}\|^2}\boldsymbol{x} \,, \tag{3.45}$$

$$\boldsymbol{P}_\pi = \frac{\boldsymbol{b}\boldsymbol{b}^\top}{\|\boldsymbol{b}\|^2} \,. \tag{3.46}$$

- bb^T는 symmetric matrix (rank 1)
- $||b||^2 = \langle b, b \rangle$ → scalar
- Projection Matrix P_π는 모든 벡터$x \in \mathbb{R}^n$을 원점으로 통과하고 방향이 b인 직선으로 projection합니다.

Remark

Projection $\pi_U(x) \in \mathbb{R}^n$은 여전히 n-차원 벡터입니다. 즉, 우리는 투영을 표현하기위해서 n-차원의 좌표가 필요 없습니다. 투영된 공간(부분 공간 U)에서 1차원 좌표λ 만 알면 충분합니다.

$x = \begin{bmatrix} 2 \\ 3 \end{bmatrix} \in \mathbb{R}^2, \ \ b = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

투영 결과 : $\pi_U(x) = \begin{bmatrix} 2.5 \\ 2.5 \end{bmatrix} = \lambda b$

$\lambda = 2.5$

- 투영 결과 벡터는 여전히 2차원이지만, b(기저 벡터)축 기준으로 보면 1차원 좌표(λ)값만 알면 충분합니다.

예시 3.10 (Projection onto a Line)

Example 3.10 (Projection onto a Line)
Find the projection matrix $\boldsymbol{P}_\pi$ onto the line through the origin spanned by $\boldsymbol{b} = \begin{bmatrix}1 & 2 & 2\end{bmatrix}^\top$. $\boldsymbol{b}$ is a direction and a basis of the one-dimensional subspace (line through origin).

With (3.46), we obtain

$$\boldsymbol{P}_\pi = \frac{\boldsymbol{b}\boldsymbol{b}^\top}{\boldsymbol{b}^\top\boldsymbol{b}} = \frac{1}{9}\begin{bmatrix}1\\2\\2\end{bmatrix}\begin{bmatrix}1 & 2 & 2\end{bmatrix} = \frac{1}{9}\begin{bmatrix}1 & 2 & 2\\2 & 4 & 4\\2 & 4 & 4\end{bmatrix}. \tag{3.47}$$

Let us now choose a particular $\boldsymbol{x}$ and see whether it lies in the subspace spanned by $\boldsymbol{b}$. For $\boldsymbol{x} = \begin{bmatrix}1 & 1 & 1\end{bmatrix}^\top$, the projection is

$$\pi_U(\boldsymbol{x}) = \boldsymbol{P}_\pi\boldsymbol{x} = \frac{1}{9}\begin{bmatrix}1 & 2 & 2\\2 & 4 & 4\\2 & 4 & 4\end{bmatrix}\begin{bmatrix}1\\1\\1\end{bmatrix} = \frac{1}{9}\begin{bmatrix}5\\10\\10\end{bmatrix} \in \mathrm{span}[\begin{bmatrix}1\\2\\2\end{bmatrix}]. \tag{3.48}$$

Note that the application of $\boldsymbol{P}_\pi$ to $\pi_U(\boldsymbol{x})$ does not change anything, i.e., $\boldsymbol{P}_\pi\pi_U(\boldsymbol{x}) = \pi_U(\boldsymbol{x})$. This is expected because according to Definition 3.10, we know that a projection matrix $\boldsymbol{P}_\pi$ satisfies $\boldsymbol{P}_\pi^2\boldsymbol{x} = \boldsymbol{P}_\pi\boldsymbol{x}$ for all $\boldsymbol{x}$.

Remark

Ch4 에서는 $\pi_U(x)$는 P_π의 eigenvector(고유벡터)이고, eigenvalue(고유값)은 1이라는 것을 보여줍니다.

3.8.2 Projection onto General Subspaces

이번 장에서는 1차원이 아닌 m 차원 $(m \geq 1)$ 부분 공간 $U \in \mathbb{R}^2$에 직교 투영하는 경우를 살펴봅니다.

3.8 Orthogonal Projections 85

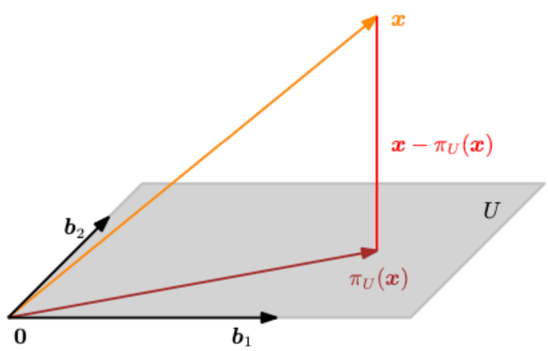


Figure 3.11 Projection onto a two-dimensional subspace U with basis $\boldsymbol{b}_1, \boldsymbol{b}_2$. The projection $\pi_U(\boldsymbol{x})$ of $\boldsymbol{x} \in \mathbb{R}^3$ onto U can be expressed as a linear combination of $\boldsymbol{b}_1, \boldsymbol{b}_2$ and the displacement vector $\boldsymbol{x} - \pi_U(\boldsymbol{x})$ is orthogonal to both $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$.

U의 ordered basis → $(b_1, \ldots, b_m)$

모든 U로의 projection $\pi_U(x)$는 U의 element입니다. 따라서 U의 기저 벡터 $(b_1, \ldots, b_m)$ 는 다음과 같이 선형 결합(Linear Combination)으로 표현할 수 있습니다.

$\pi_U(x) = \sum_{i=1}^{m} \lambda_i b_i$

1차원의 부분 공간으로 projection하는 경우와 마찬가지로 projection $\pi_U(x)$와 projection matrix P_π를 3단계로구할 수 있습니다.

1. 투영 벡터를 $\lambda_1, \ldots, \lambda_m$ 좌표로 표현

부분공간 U는 선형 독립인 $b_1, \ldots, b_m \in \mathbb{R}^n$ 의 span으로 표현됩니다.

$$\pi_U(\boldsymbol{x}) = \sum_{i=1}^{m} \lambda_i \boldsymbol{b}_i = \boldsymbol{B}\boldsymbol{\lambda}, \tag{3.49}$$

$$\boldsymbol{B} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_m] \in \mathbb{R}^{n \times m}, \quad \boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_m]^\top \in \mathbb{R}^m, \tag{3.50}$$

위 선형 결합은 x에서 closest **하다는 것은 mininum distance를 의미합니다. $\pi_U(x) \in U$ 와 $x \in \mathbb{R}^n$을 연결하는 모든 벡터는 U의 기저 벡터와 직교합니다.**

따라서 우리는 동시에 만족해야하는 m개의 조건을 얻습니다. (*내적은 dot product라고 가정*)

$$\langle \boldsymbol{b}_1, \boldsymbol{x} - \pi_U(\boldsymbol{x}) \rangle = \boldsymbol{b}_1^\top (\boldsymbol{x} - \pi_U(\boldsymbol{x})) = 0 \tag{3.51}$$

$$\vdots$$

$$\langle \boldsymbol{b}_m, \boldsymbol{x} - \pi_U(\boldsymbol{x}) \rangle = \boldsymbol{b}_m^\top (\boldsymbol{x} - \pi_U(\boldsymbol{x})) = 0 \tag{3.52}$$

$\pi_U(x) = B\lambda$ 이므로, 위 식은 다음과 같이 쓸 수 있습니다.

$$\boldsymbol{b}_1^\top (\boldsymbol{x} - \boldsymbol{B\lambda}) = 0$$

$$\vdots$$

$$\boldsymbol{b}_m^\top (\boldsymbol{x} - \boldsymbol{B\lambda}) = 0$$

우리는 homogeneous linear equation system을 얻을 수 있습니다.

$$\begin{bmatrix} \boldsymbol{b}_1^\top \\ \vdots \\ \boldsymbol{b}_m^\top \end{bmatrix} \begin{bmatrix} \boldsymbol{x} - \boldsymbol{B\lambda} \end{bmatrix} = \boldsymbol{0} \iff \boldsymbol{B}^\top (\boldsymbol{x} - \boldsymbol{B\lambda}) = \boldsymbol{0} \tag{3.55}$$

$$\iff \boldsymbol{B}^\top \boldsymbol{B\lambda} = \boldsymbol{B}^\top \boldsymbol{x} . \tag{3.56}$$

여기서 마지막 표현식을 **정규방정식(normal equation)** 이라고 부릅니다.

$b_1, \ldots, b_m$은 U의 기저이고, 선형독립입니다. 그래서 $B^T B \in R^{m \times m}$ 은 정사각행렬이며, 역행렬이 존재합니다. 따라서 coordinate를 다음과 같이 구할 수 있습니다.

$$\boldsymbol{\lambda} = (\boldsymbol{B}^\top \boldsymbol{B})^{-1} \boldsymbol{B}^\top \boldsymbol{x} . \tag{3.57}$$

$(B^T B)^{-1} B^T$ 행렬은 B의 pseudo-inverse(유사 역행렬)이라고 하며, 정사각 행렬이 아닌 행렬에 대해서도 계산할 수 있습니다.

- 유사 역행렬(무어-펜로즈 유사역행렬)을 사용할 수 있는 유일한 조건
  - $B^T B$ → positive definite
  - B → full rank
  - 실전에서는 수치적 안정성을 위해 ridge regularization을 사용해 다음처럼 계산합니다.
    - $\lambda = (B^T B + \epsilon I)^{-1} B^T x$
    - 여기서 ϵI는 jitter term 또는 ridge term이라 불리며, 수치적인 불안정성을 방지하고 $B^T B$가 양의 정부호가 되도록 돕습니다.

2. Projection $\pi_U(x) \in U$ 찾기

우리는 이미 $\pi_U(x) = B\lambda$라는 것을 알기 때문에, 식 3.58을 구할 수 있습니다.

$$\pi_U(\boldsymbol{x}) = \boldsymbol{B}(\boldsymbol{B}^\top \boldsymbol{B})^{-1} \boldsymbol{B}^\top \boldsymbol{x} . \tag{3.58}$$

3. Projection P_π 찾기

$P_{pi}(x) = \pi_U(x)$를 풀면 간단하게 projection matrix를 찾을 수 있습니다.

$$\boldsymbol{P}_\pi = \boldsymbol{B}(\boldsymbol{B}^\top \boldsymbol{B})^{-1} \boldsymbol{B}^\top . \tag{3.59}$$

Remark

일반적인 projection 행렬 공식은 1차원 subspace($\dim(U) = 1$)의 경우를 특수한 케이스로 포함합니다.

즉, 만약 $U = span(b)$ 처럼 하나의 벡터 $b \in \mathbb{R}^n$만으로 생성되는 1차원 공간이라면

- $B = b \in \mathbb{R}^{n \times 1} \rightarrow$ 열벡터
- $B^T B \in \mathbb{R} \rightarrow$ 스칼라 $b^T b$

따라서,

$$P_\pi = B(B^\top B)^{-1} B^\top = \frac{1}{b^\top b} bb^\top$$

이전의 1차원 투영 행렬 공식과 동일합니다.

$$P = \frac{bb^\top}{b^\top b}$$

예시 3.11 (Projection onto a Two-dimensional Subspace)

Example 3.11 (Projection onto a Two-dimensional Subspace)

For a subspace $U = \text{span}[\begin{bmatrix}1\\1\\1\end{bmatrix}, \begin{bmatrix}0\\1\\2\end{bmatrix}] \subseteq \mathbb{R}^3$ and $\boldsymbol{x} = \begin{bmatrix}6\\0\\0\end{bmatrix} \in \mathbb{R}^3$ find the coordinates $\boldsymbol{\lambda}$ of $\boldsymbol{x}$ in terms of the subspace U, the projection point $\pi_U(\boldsymbol{x})$ and the projection matrix $\boldsymbol{P}_\pi$.

First, we see that the generating set of U is a basis (linear independence) and write the basis vectors of U into a matrix $\boldsymbol{B} = \begin{bmatrix}1 & 0\\1 & 1\\1 & 2\end{bmatrix}$.

Second, we compute the matrix $\boldsymbol{B}^\top \boldsymbol{B}$ and the vector $\boldsymbol{B}^\top \boldsymbol{x}$ as

$$\boldsymbol{B}^\top \boldsymbol{B} = \begin{bmatrix}1 & 1 & 1\\0 & 1 & 2\end{bmatrix}\begin{bmatrix}1 & 0\\1 & 1\\1 & 2\end{bmatrix} = \begin{bmatrix}3 & 3\\3 & 5\end{bmatrix}, \quad \boldsymbol{B}^\top \boldsymbol{x} = \begin{bmatrix}1 & 1 & 1\\0 & 1 & 2\end{bmatrix}\begin{bmatrix}6\\0\\0\end{bmatrix} = \begin{bmatrix}6\\0\end{bmatrix}. \tag{3.60}$$

Third, we solve the normal equation $\boldsymbol{B}^\top \boldsymbol{B}\boldsymbol{\lambda} = \boldsymbol{B}^\top \boldsymbol{x}$ to find $\boldsymbol{\lambda}$:

$$\begin{bmatrix}3 & 3\\3 & 5\end{bmatrix}\begin{bmatrix}\lambda_1\\\lambda_2\end{bmatrix} = \begin{bmatrix}6\\0\end{bmatrix} \iff \boldsymbol{\lambda} = \begin{bmatrix}5\\-3\end{bmatrix}. \tag{3.61}$$

Fourth, the projection $\pi_U(\boldsymbol{x})$ of $\boldsymbol{x}$ onto U, i.e., into the column space of $\boldsymbol{B}$, can be directly computed via

$$\pi_U(\boldsymbol{x}) = \boldsymbol{B}\boldsymbol{\lambda} = \begin{bmatrix}5\\2\\-1\end{bmatrix}. \tag{3.62}$$

The corresponding *projection error* is the norm of the difference vector between the original vector and its projection onto U, i.e.,

$$\|\boldsymbol{x} - \pi_U(\boldsymbol{x})\| = \left\|\begin{bmatrix}1 & -2 & 1\end{bmatrix}^\top\right\| = \sqrt{6}. \tag{3.63}$$

Fifth, the projection matrix (for any $\boldsymbol{x} \in \mathbb{R}^3$) is given by

$$\boldsymbol{P}_\pi = \boldsymbol{B}(\boldsymbol{B}^\top \boldsymbol{B})^{-1}\boldsymbol{B}^\top = \frac{1}{6}\begin{bmatrix}5 & 2 & -1\\2 & 2 & 2\\-1 & 2 & 5\end{bmatrix}. \tag{3.64}$$

To verify the results, we can (a) check whether the displacement vector $\pi_U(\boldsymbol{x}) - \boldsymbol{x}$ is orthogonal to all basis vectors of U, and (b) verify that $\boldsymbol{P}_\pi = \boldsymbol{P}_\pi^2$ (see Definition 3.10).

Remark

투영된 백터 $\pi_U(x)$는 $\mathbb{R}^n$에 있는 벡터입니다. 다만 $\mathbb{R}^n$안의 m-차원 subspace $U \subseteq \mathbb{R}^n$ 위에 놓여있습니다.

하지만 이 벡터를 표현하기 위해 n개의 좌표가 필요한 것이 아닌, U의 기저 $\{b_1, \ldots, b_m\}$에 대한 m 개의 좌표 $\lambda_1, \ldots, \lambda_m$만 있으면 충분합니다.

지구상 위치를 위도와 경도만으로 표현할 수 있는 것 처럼, 3차원 공간 속 구의 표면 위의 한 점을 2개의 좌표로 충분히 표현 가능한 것과 같은 논리입니다.

Remark

어떠한 내적을 사용하냐에 따라 거리와 각도를 계산하는 것이 바뀔 수 있음을 주의해야합니다.

Projection은 선형방정식 $Ax = b$가 해가 없을 때 사용될 수 있습니다. A의 column vector들을 $a_1, a_2, \ldots, a_m$이라고 했을 때, $b = x_1a_1 + x_2a_2 + \cdots + x_ma_m$ 이 성립한다면 $(x_1, x_2, \ldots, x_m)^T$가 해가 됩니다.

결국 b는 A의 column vector들로 span 되는 subspace의 원소가 된다는 의미입니다. x는 좌표를 의미합니다.

b가 A의 column space로 span이 안될 때 근이 존재하지 않는다는 의미가 됩니다.

그럴 때는 우리는 근사해를 찾을 수 있습니다. A의 column vector 중 b와 가장 가까운 벡터를 찾는 것입니다.

$Ax = \pi_A(b)$를 생각할 수 있습니다. 이 식은 항상 해가 존재하며, 내적을 dot product로 가정했을 때, 구한 근사해를 가르켜 least-squares solution이라고 합니다. 이는 Ch9.4에 쓰일 reconstruction error를 구할 때 유용하게 쓰이고, 해당 error를 사용해서 PCA(주성분 분석)를 할 수 있습니다.

Remark

만약 U의 기저 벡터들이 orthonormal 인 경우, $B^TB = I$ 가 성립합니다.

따라서 λ를 구하는 과정이 매우 간소화 됩니다.

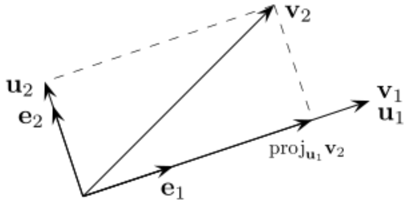
$$\pi_U(\boldsymbol{x}) = \boldsymbol{B}\boldsymbol{B}^\top \boldsymbol{x}$$

$$\boldsymbol{\lambda} = \boldsymbol{B}^\top \boldsymbol{x} .$$

우리는 이제 역행렬를 계산할 필요가 없다는 의미입니다.

3.8.3 Gram-schmidt Orthogonalization

정사영(Projections)은 Gram-schmidt의 핵심 방법입니다.



임의의 선형 독립 기저 $(b_1, b_2, \ldots, b_n)$을 직교(orthogonal) 또는 정규 직교(orthonormal) 기저 $(u_1, u_2, \ldots, u_n)$으로 변환하는 방법입니다.

두 기저는 같은 공간을 span 합니다. $\mathrm{span(b_1, b_2, \ldots, b_n) = span(u_1, u_2, \ldots, u_n)}$

$$\boldsymbol{u}_1 := \boldsymbol{b}_1 \tag{3.67}$$

$$\boldsymbol{u}_k := \boldsymbol{b}_k - \pi_{\mathrm{span}[\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k-1}]}(\boldsymbol{b}_k), \quad k = 2, \ldots, n . \tag{3.68}$$

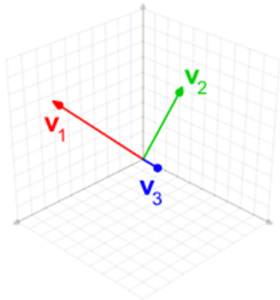
k번째 기저 벡터 b_k는 이미 만들어진 직교 벡터 $u_1, \ldots, u_{k-1}$가 생성하는 부분공간에 투영합니다.

이 투영 벡터를 b_k에서 빼주면, $u_1, \ldots, u_{k-1}$로 생성된 $k-1$-차원 부분공간에 직교하는 벡터 u_k가 얻어집니다.

이 과정을 모든 기저 벡터 $(b_1, b_2, \ldots, b_n)$에 대해 반복하면, 공간 V에 대한 직교 기저 $(u_1, u_2, \ldots, u_n)$를 만들 수 있습니다.

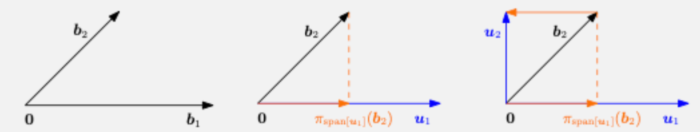
또한 각 u_k를 정규화하면 $||u_k|| = 1$인 정규 직교 기저(ONB)를 얻게 됩니다.

그람-슈미트 과정의 시각화



예시 3.12 (Gram-Schmidt Orthogonalization)

Example 3.12 (Gram-Schmidt Orthogonalization)



(a) Original non-orthogonal basis vectors b_1, b_2.

(b) First new basis vector $u_1 = b_1$ and projection of b_2 onto the subspace spanned by u_1.

(c) Orthogonal basis vectors u_1 and $u_2 = b_2 - \pi_{\mathrm{span}[u_1]}(b_2)$.

Figure 3.12 Gram-Schmidt orthogonalization. (a) non-orthogonal basis (b_1, b_2) of $\mathbb{R}^2$; (b) first constructed basis vector u_1 and orthogonal projection of b_2 onto $\mathrm{span}[u_1]$; (c) orthogonal basis (u_1, u_2) of $\mathbb{R}^2$.

Consider a basis (b_1, b_2) of $\mathbb{R}^2$, where

$$b_1 = \begin{bmatrix} 2 \\ 0 \end{bmatrix}, \quad b_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}; \tag{3.69}$$

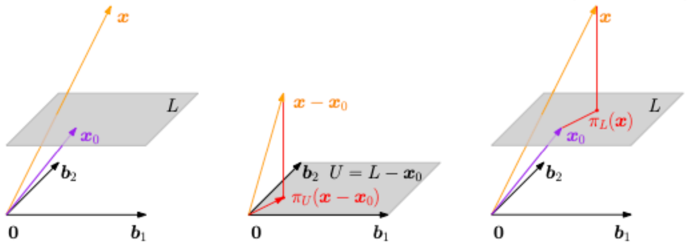
see also Figure 3.12(a). Using the Gram-Schmidt method, we construct an orthogonal basis (u_1, u_2) of $\mathbb{R}^2$ as follows (assuming the dot product as the inner product):

$$u_1 := b_1 = \begin{bmatrix} 2 \\ 0 \end{bmatrix}, \tag{3.70}$$

$$u_2 := b_2 - \pi_{\mathrm{span}[u_1]}(b_2) \overset{(3.45)}{=} b_2 - \frac{u_1 u_1^\top}{\|u_1\|^2} b_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{3.71}$$

3.8.3 Projection onto Affine Subspaces

90 *Analytic Geometry*



(a) Setting.

(b) Reduce problem to projection π_U onto vector subspace.

(c) Add support point back in to get affine projection π_L.

Figure 3.13 Projection onto an affine space. (a) original setting; (b) setting shifted by $-\boldsymbol{x}_0$ so that $\boldsymbol{x}-\boldsymbol{x}_0$ can be projected onto the direction space U; (c) projection is translated back to $\boldsymbol{x}_0 + \pi_U(\boldsymbol{x}-\boldsymbol{x}_0)$, which gives the final orthogonal projection $\pi_L(\boldsymbol{x})$.

These steps are illustrated in Figures 3.12(b) and (c). We immediately see that $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ are orthogonal, i.e., $\boldsymbol{u}_1^\top \boldsymbol{u}_2 = 0$.

지금까지, 어떻게 벡터를 lower-차원 부분공간 U에 정사영하는 방법을 알아봤습니다. 이제는 affine subspace에 정사영 시키는 방법에 대해 알아봅니다.

위 그림 (a)를 살펴보면, Affine space $L = x_0 + U$ 가 주어져있습니다. U의 기저 벡터는 b_1, b_2입니다.

L의 투영되는 orthogonal projection $\pi_L(x)$를 구하기 위해서, 위에서 했던 문제로 번형하여 쉽게 풀 수 있습니다.

x와 L에 support point x_0을 빼주면 vector subspace $U = L - x_0$을 얻을 수 있습니다.

에서 살펴본 방법을 사용하여 $\pi_U(x - x_0)$을 얻을 수 있습니다.

얻은 projection을 다시 x_0을 더해 L로 변환할 수 있으며, 다음과 같이 구할 수 있습니다.

$$\pi_L(\boldsymbol{x}) = \boldsymbol{x}_0 + \pi_U(\boldsymbol{x}-\boldsymbol{x}_0)\,, \tag{3.72}$$

$\pi_U(\cdot)$은 subspace U로의 정사영이며, 그림 (c)에서 L의 direction space입니다.

From Figure 3.13, it is also evident that the distance of $\boldsymbol{x}$ from the affine space L is identical to the distance of $\boldsymbol{x}-\boldsymbol{x}_0$ from U, i.e.,

$$\begin{aligned} d(\boldsymbol{x}, L) &= \|\boldsymbol{x} - \pi_L(\boldsymbol{x})\| = \|\boldsymbol{x} - (\boldsymbol{x}_0 + \pi_U(\boldsymbol{x}-\boldsymbol{x}_0))\| && (3.73a)\\ &= d(\boldsymbol{x}-\boldsymbol{x}_0, \pi_U(\boldsymbol{x}-\boldsymbol{x}_0)) = d(\boldsymbol{x}-\boldsymbol{x}_0, U)\,. && (3.73b)\end{aligned}$$

벡터 x에서 Affine subspace 사이의 거리도 알 수 있습니다. 그 거리는 위 식과 같습니다.

해당 부분은 Section12.1에서 Support Vector Machine 경계선을 긋는 Hyperplane(초평면)의 개념을 도입할 때 참고됩니다.

출처

- Mathmatics for Machine Learning (https://github.com/mml-book/mml-book.github.io)
- https://blog.naver.com/walk_along/222187729940
- https://junstar92.github.io/mml-study-note/2022/07/10/ch3-8.html
- https://ko.wikipedia.org/wiki/%EA%B7%B8%EB%9E%8C-%EC%8A%88%EB%AF%B8%ED%8A%B8_%EA%B3%BC%EC%A0%95#