Orthogonal Projection

Orthogonal Projection

- b의 사영($\hat{b}$)은 $col\,A$(A의 컬럼벡터로 이루어진 span) 위에 존재한다.
- $\hat{b}$ 의 식

$$\hat{\mathbf{b}} = f(\mathbf{b}) = A\hat{\mathbf{x}} = A(A^TA)^{-1}A^T\mathbf{b}$$

  - f: 사영시키는 함수
  - $A(A^TA)^{-1}A^T$: 행렬
  - b: 벡터
  - 위의 식은 선형의 성질을 만족한다.(선 선형결합, 후 사영 = 선 각각을 사영, 후 선형결합)

Orthogonal Sets & Orthonormal Sets

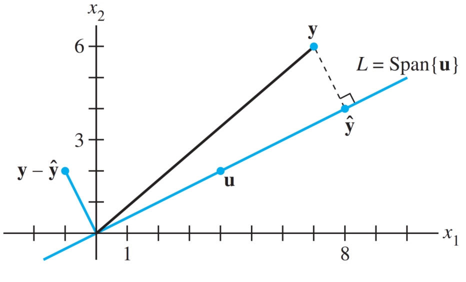
- **Orthogonal Sets**
  - 어떤 벡터들의 집합(공간)이 있다고 했을 때, 그 중에서 어떤 서로다른 i, j번째 벡터를 내적하면, 언제나 0이 되는 경우
- **Orthonormal Sets**
  - 각 벡터들이 Orthogonal 조건을 만족하며, 그 길이가 모두 1인 집합(unit vectors)
- → Orthogonal/Orthonormal set은 언제나 선형독립이다.
  (새 벡터가 추가되면, span의 dimension이 추가된다.)

Orthogonal Basis 찾기

📌 **Subspace의 basis 찾는 방법**

- *Subspace: 선형결합에 닫혀있는 벡터들의 부분집합
- ⇒ 순차적으로 하나씩 벡터를 고른다.
  subspace를 충분히 span하도록 하는 최소한의 재료벡터를 찾는다.
  재료벡터들이 모두 중복없이 선형독립을 만족하도록 한다.
- 위의 subspace's basis 찾는 방법을 활용하여, orthogonal basis를 찾을 수 있다.
  (orthogonal vector set이 기존 subspace의 basis보다 까다로움)

- Gram-Schmidt process를 통해, Orthogonal basis를 찾으면 된다.
  - 그램-슈미츠 과정: 기존 벡터를 사영시켜서, 새로운 basis로 사용한다.
    ⇒ QR (Matrix 분해)의 과정. Orthogonalize.
- Line) 사영한 벡터 구하기



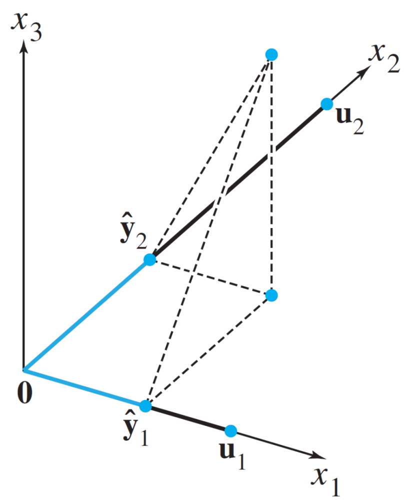
  - 위의 그림에서 y 벡터를 사영한 결과인 $\hat{y}$의 식은 아래와 같다. (내적 식 참고)

$$\hat{\mathbf{y}} = \text{proj}_L \mathbf{y} = \frac{\mathbf{y}\cdot\mathbf{u}}{\mathbf{u}\cdot\mathbf{u}}\mathbf{u}$$

- u가 단위벡터라면, 아래와 같이 표현할 수 있다.

$$\hat{\mathbf{y}} = \text{proj}_L \mathbf{y} = (\mathbf{y}\cdot\mathbf{u})\mathbf{u}$$

- Plane) 사영한 벡터 구하기



- 삼수선 정리에 의해, y 점에서 각 축과 면에 내린 수선의 발은 모두 직각이다.
- 위의 그림에서 y 벡터를 사영한 결과인 $\hat{y}$의 식은 아래와 같다. (내적 식 참고)

$$\hat{\mathbf{y}} = \text{proj}_L \mathbf{y} = \frac{\mathbf{y}\cdot\mathbf{u}_1}{\mathbf{u}_1\cdot\mathbf{u}_1}\mathbf{u}_1 + \frac{\mathbf{y}\cdot\mathbf{u}_2}{\mathbf{u}_2\cdot\mathbf{u}_2}\mathbf{u}_2$$

- u_1, u_2 벡터가 단위벡터라면, 아래와 같이 표현할 수 있다.

$$\hat{\mathbf{y}} = \text{proj}_L \mathbf{y} = (\mathbf{y}\cdot\mathbf{u}_1)\mathbf{u}_1 + (\mathbf{y}\cdot\mathbf{u}_2)\mathbf{u}_2$$

Orthogonal Projection을 선형변환으로 나타내기

$$\begin{aligned}
\hat{\mathbf{b}} &= f(\mathbf{b}) = (\mathbf{b}\cdot\mathbf{u}_1)\mathbf{u}_1 + (\mathbf{b}\cdot\mathbf{u}_2)\mathbf{u}_2 \\
&= (\mathbf{u}_1^T\mathbf{b})\mathbf{u}_1 + (\mathbf{u}_2^T\mathbf{b})\mathbf{u}_2 \\
&= \mathbf{u}_1(\mathbf{u}_1^T\mathbf{b}) + \mathbf{u}_2(\mathbf{u}_2^T\mathbf{b}) \\
&= (\mathbf{u}_1\mathbf{u}_1^T)\mathbf{b} + (\mathbf{u}_2\mathbf{u}_2^T)\mathbf{b} \\
&= (\mathbf{u}_1\mathbf{u}_1^T + \mathbf{u}_2\mathbf{u}_2^T)\mathbf{b} \\
&= [\mathbf{u}_1 \quad \mathbf{u}_2]\begin{bmatrix}\mathbf{u}_1^T \\ \mathbf{u}_2^T\end{bmatrix}\mathbf{b} = UU^T\mathbf{b} \quad \Rightarrow \text{linear transformation!}
\end{aligned}$$

- orthonormal basis u_1, u_2 가 있다고 할 때,
  결합법칙과 sum of rank 1 outer product 표현방식을 사용하여,
  b 벡터의 사영인 $\hat{b}$을 선형변환의 형태로 나타낼 수 있다.
- U = projection 하고자 하는 subspace의 orthogonal/orthonormal한 벡터를 컬럼으로 가지는 matrix
- $A = U = [u_1 u_2]$가 orthonormal columns 라면, u_1, u_2 벡터는 서로 수직이고, 각각의 길이는 1이다. 따라서, 아래의 식을 만족한다.

$$C = A^T A = \begin{bmatrix} \mathbf{u}_1^T \\ \mathbf{u}_2^T \end{bmatrix} [\mathbf{u}_1 \quad \mathbf{u}_2] = I. \text{ Thus,}$$

$$\hat{\mathbf{b}} = A\hat{\mathbf{x}} = A(A^T A)^{-1} A^T \mathbf{b} = A(I)^{-1} A^T \mathbf{b} = AA^T \mathbf{b} = UU^T \mathbf{b}$$

u1, u2 벡터가 서로 orthogonal 을 만족하지 않는 경우

- 두 벡터의 각도가 직각이 아닌 경우,
  두개 이상의 피처가 비슷한 방향성을 가지면, 작은 차이가 계수값의 크고 작음에 많은 영향을 준다.
- 왜냐면, projection 을 하려는 방향은 다른 벡터의 방향에 영향을 받기 때문
- 따라서, Ridge/Lasso와 같은 Regularization을 사용한다. → 벡터들을 각각의 축에 붙여버리는 효과를 줘서, 솔루션이 덜 민감해지게 한다.
- 90도라면, 다른 벡터의 방향에 상관없이 수선의 발을 내려서, projection한 벡터를 구할 수 있으므로, 다음 강의에서, orthogonal 하지 않은 벡터들을 orthogonal 하도록 구하는 과정이 나온다.