L1, L2 정리 (+ L1, L2 Regularization)

L1, L2

L1 distance을 사용하는 상황

- 각 축의 방향에 해석적인 의미가 있을 때
- 이미지에서 픽셀별 밝기가 독립적일 때
- 각 차원이 독립적인 특징(feature)을 나타내고 각 축이 개별적으로 의미 있는 경우
- 희소한 데이터에 강함 (많은 feature가 0일 때)*
  → ex. 텍스트 데이터를 벡터화한 경우 (Bag-of-Words)
- noise가 적고 개별 feature의 변화량이 직접적으로 의미 있을 때

L2 distance를 사용하는 상황

- 전체 공간에서 방향이 중요한 경우, 축의 방향에 의미가 없는 경우**
- 이미지에서 물체의 위치나 방향이 달라도 비슷한 것이라고 보고 싶을 때
- 데이터의 회전, 스케일 등에 대해 robust하고 싶을 때
- 각 feature가 상호작용하며 의미를 가지는 경우***
  → ex. 실제 공간적인 거리 측정, 위치 기반 거리
- feature들 간에 특별한 해석이 없이 단순한 벡터로서 거리를 비교하고 싶을 때

* L1이 희소한 데이터에 강한 이유는?

L1은 단순히 각 축의 절댓값 차이의 합이기에 대부분의 차원이 0이더라도 0이 아닌 몇 개의 feature만으로 거리 판단이 가능

L2는 모든 차원에서 제곱하기에 작은 노이즈들도 거리에 기여

> 즉, L1은 개별 차이의 합이므로 큰 차이가 나는 축에 중점적으로 민감하고 노이즈는 무시함

- L1은 몇 개의 중요한 feature만 차이가 나도 감지 가능
  → sparse한 데이터에 강함
- L2는 모든 feature를 고려
  → 노이즈에도 민감하고 값이 골고루 있을 때 더 잘 작동

** 방향과 축의 방향이란?

방향: 전체 공간의 방향

축의 방향: 개별 feature의 기준축

축의 방향의 의미가 있다
⇒ 각 축이 '혈압', '혈당', '나이'처럼 해석 가능한 독립적인 정보일 때
→ 이 경우 x축이든 y축이든 그 자체로 의미가 있으므로 각각 얼마씩 차이나는 지를 따져야 하니까 L1이 적합

방향이 중요하다
⇒ 데이터가 어떤 '방향성'을 갖고 변화하는가, 혹은 두 점 사이의 직선 거리가 중요한가
→ 이때는 좌표계의 축이 어떤 방향으로 되어 있는지보다 전체 벡터 방향이 중요하므로 L2가 적합

- L1은 축 하나하나에 의미 있는 경우 (축의 방향에 의미 있음)
- L2는 전체 벡터의 방향(방향성)이 중요한 경우 (축 자체의 이름이나 위치는 크게 중요하지 않음)

***** L2가 feature 상호작용에 적합한 이유는?**

L2는 각 차원의 차이를 제곱하고 모두 **합산**해서 하나의 거리로 만듦. 결국 특징들 간의 조합, 즉 벡터 전체의 움직임을 보는 것

- 두 사람이 키, 몸무게, 나이 등의 벡터로 표현됐을 때, 그 각각을 따로 비교하기보다는 "전체적인 유사도"를 보고 싶을 때
- 개별 feature가 아니라, 전체적인 특성의 조화가 중요한 경우

L1, L2 Regularization

L1/L2 Regularization (Lasso/Ridge)

L1, L2의 거리 측정 방식이 모델의 제약 방식으로 확장된 것

- Distance Metric
  → 두 점 사이가 얼마나 떨어졌는가
- Regularization
  → 모델의 파라미터 벡터가 얼마나 복잡한가

Lasso (L1 regularization)

: 어떤 feature는 필요 없다고 판단하고 0으로 만듦

Ridge (L2 regularization)

: 회전에도 안정적으로 weight들을 부드럽게 줄임

+)

회귀 문제를 풀 때 기본적으로 손실함수를 최소화

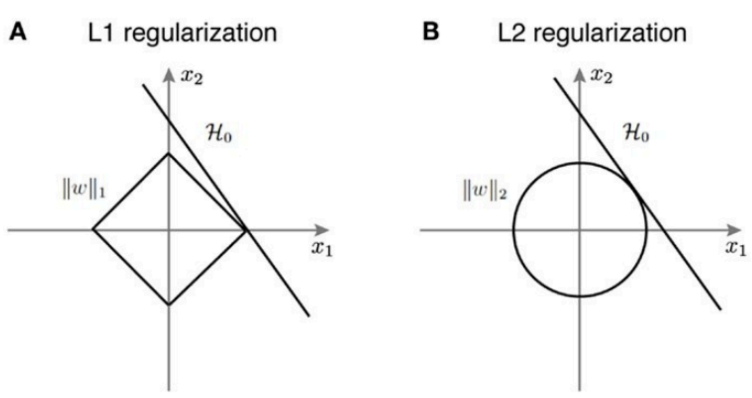
$$Loss(w) = MSE(w) + \lambda \cdot ||w||$$

- $MSE(w)$ → 예측오류
- $||w||$ → 복잡도 제약

결국 다음 두 조건을 동시에 만족시키는 가중치 w를 찾는다는 의미

1. 예측을 잘해야 하고 (예측오류 최소화)
2. 가중치가 너무 크면 X (복잡도 제한)

L1, L2 Reuglarization을 시각화해보면 다음과 같음



Lasso (L1) → "마름모 안에 들어있는 해를 찾아라"

- Lasso는 가중치의 절댓값합(=L1 norm)이 일정 값을 넘지 않도록 제한
- 이 조건을 만족하는 w 벡터들의 공간은 마름모꼴(2D), 혹은 다면체(고차원)
  :
  $|w1| + |w2| = c$ 라는 제약은 2D에서 그릴 때 마름모꼴을 만들기 때문

Ridge (L2) → "원 안에 들어있는 해를 찾아라"

- Ridge는 가중치의 제곱합(=L2 norm)이 일정한 값을 넘지 않도록 제약을 줌
- 그러면 w 벡터는 L2 norm이 c 이하인, 즉 반지름 c짜리 **원(혹은 구)** 안에 있어야 함
  ⇒
  해를 찾는 공간이 원 모양이고 그 원 안에서 **오차가 가장 작은 지점**을 찾는 것