데이터분석중급반
(To be renamed Basics to Algorithms (ML 1)

240107_Ridge Lasso Elastic Net

회귀분석의 성능을 평가할 때 대부분 실제값과 예측값의 차이를 가지고 평가했다.
둘의 관계만 살펴보게 되면 회귀계수에 대해서 과적합이 일어나서 회귀계수가 과적합이 되는 (기하급수적으로 커짐) 경우가 발생할 수 있다.

이렇게 되면 test 데이터에서는 성능이 잘 나오지 않을 수 있고, 예측하지 않은 데이터는 더 성능이 나빠질 수 있다.
규제를해서 일반화 성능 (과적합을 막기 위해) 높이려고 한다.

선형회귀 비용함수

Hypothesis : $h_\theta(x) = ax + b$

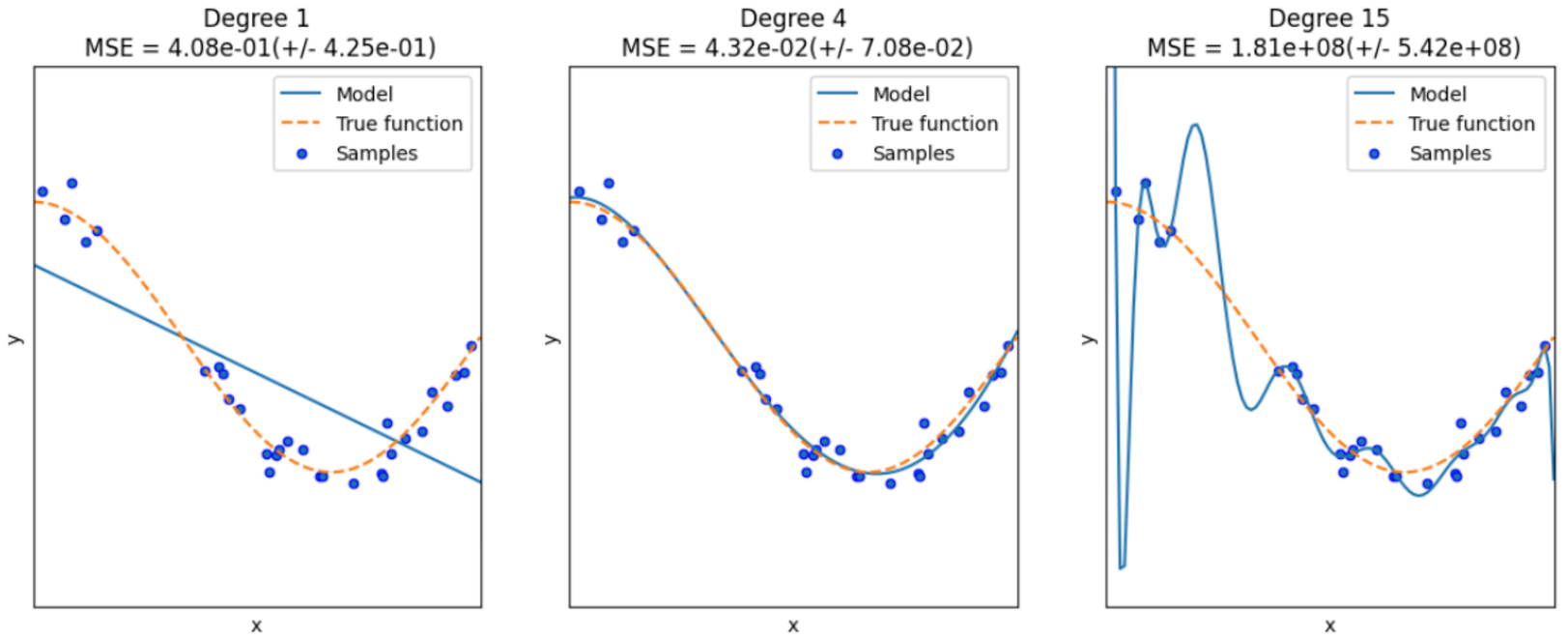
Parameters : a, b

Cost Function : $J(a,b) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$

Goal : $\underset{a,b}{\text{minimize}}\, J(a,b)$

Cost Function : 예측값과 실제값에 대한 차이를 통해서 이걸 최소화 하는 방법으로 진행했다.

Overfitting, Underfitting



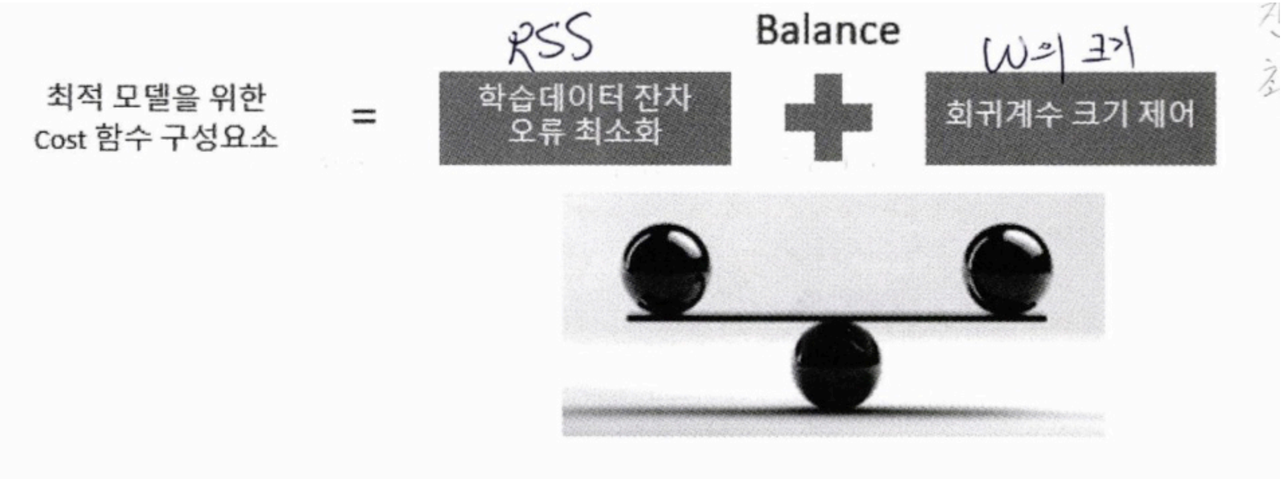
데이터의 특징을 너무 과하게 학습하게 되면 -> train 데이터에 너무나 디테일한 모든 것 까지 학습을 해서 test 진행시에 상당히 떨어지는 평가지표 값을 보인다 Overffiting
Udnerffiting 과소적합의 개념 -> 오히려 더 많은 패턴 (데이터를 넣어야 하는 경우)

기존의 선형적인 문제 해결이 아니라 비선형적인 문제일 경우 -> 다항회귀를 사용하게 된다.
이런 경우 잘못하면 Overffiting 될 가능성이 높다!

Regularization

RSS 선형회귀의 Cost Fucntion 개념

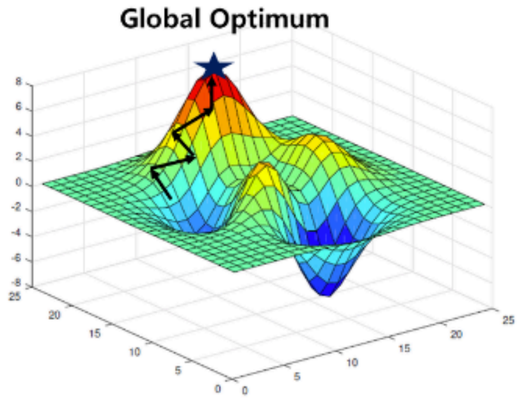
규제를 통해 Overffiting 현상을 낮추고, 일반화할 수 있도록 진행
회귀 계수를 규제한다.

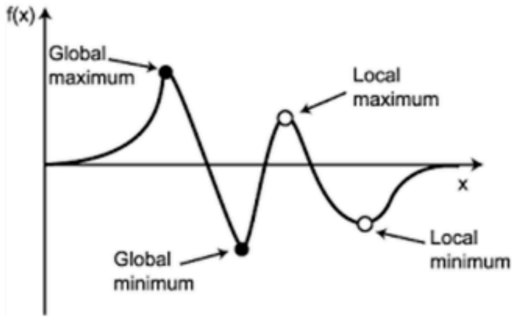


비용 함수 목표 = $Min(RSS(W)+alpha*||W||(2x2행렬)$

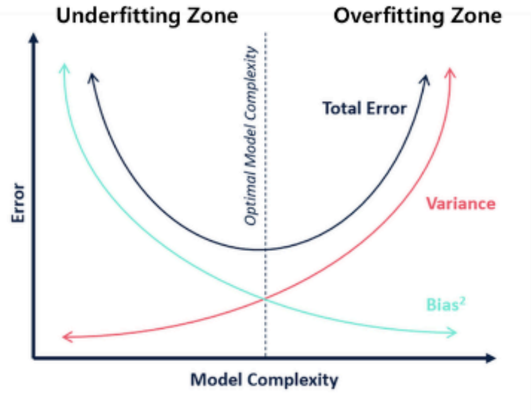
Good to Know Concepts

최적화(Optimization)와 과적합(Overffiting)

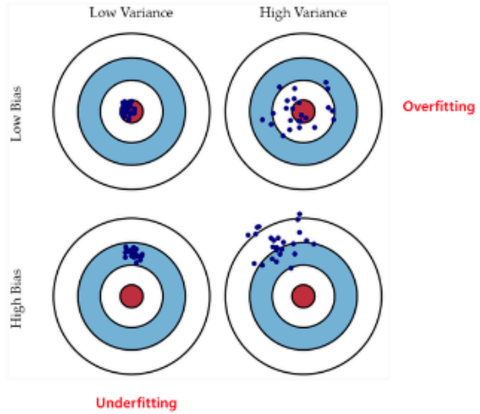




목적함수를 최소화 최대화 시키면서 찾는 방법을 최적화



x축 모델의 복잡도 y 에러



train data를 가지고- > test 데이터의 성능을 평가하는 것

가중치 규제

모델의 손실 함수 값이 너무 작아지지 않도록 특정한 값(함수) 추가

Weight 값이 과도하게 커지는 것 방지 및 데이터의 일반화 반영 가능

가중치 규제 L1(Lasso) , L2(Ridge)

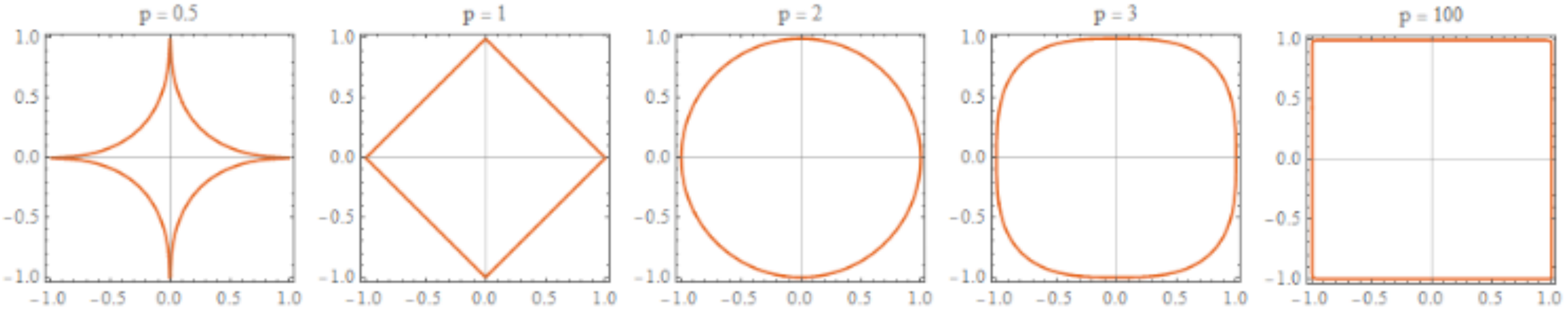
Norm

Norm 벡터의 절대적인 크기, 벡터간의 거리

모델의 손실함수에서 L1 , L2 함수 추가

$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p}, \quad 1 \leq p \leq \infty.$$

Norm은 유한 차원 벡터공간 벡터의 절대적 크기 혹은 거리



2차원 벡터 공간에서 p값 변화에 따른 p-norm의 분포 형태 (출처 :

L1 norm

L1 norm (Manhattan distance) p=1

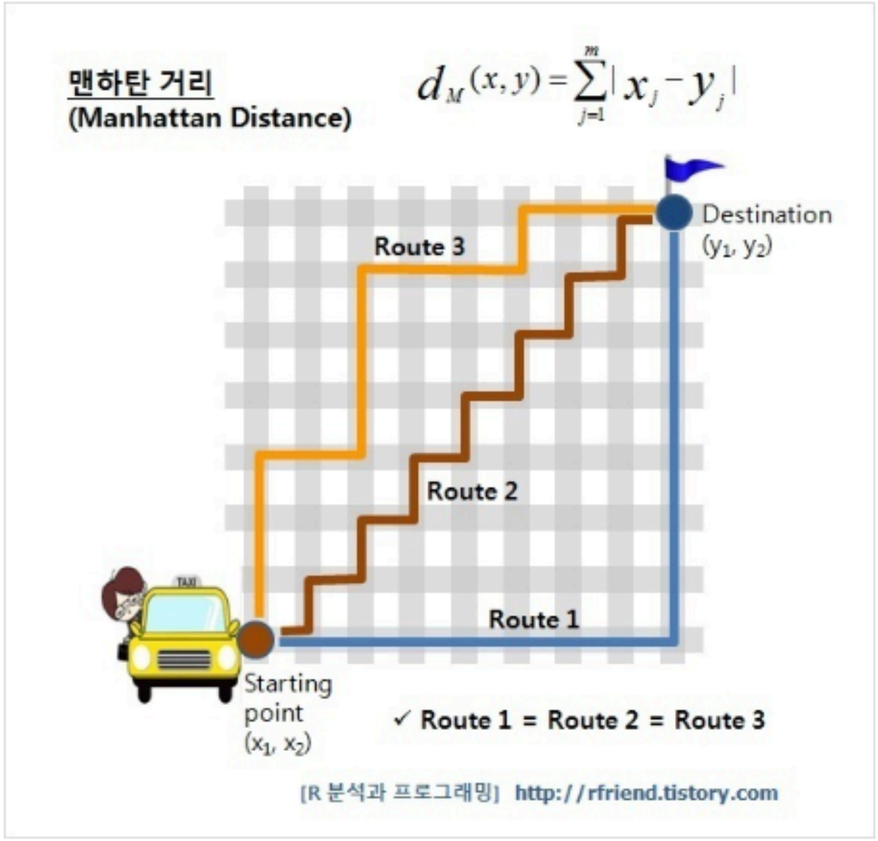
특정 방향으로만 움직일 수 있는 경우, 두 벡터 간의 최단 거리 찾기

$$d = | a_1 - b_1 | + | a_2 - b_2 |$$

L1 norm (Manhattan distance)

$$L = \sum_{i=1}^{n} |y_i - f(x_i)|$$

L1 Loss Function



L2 norm

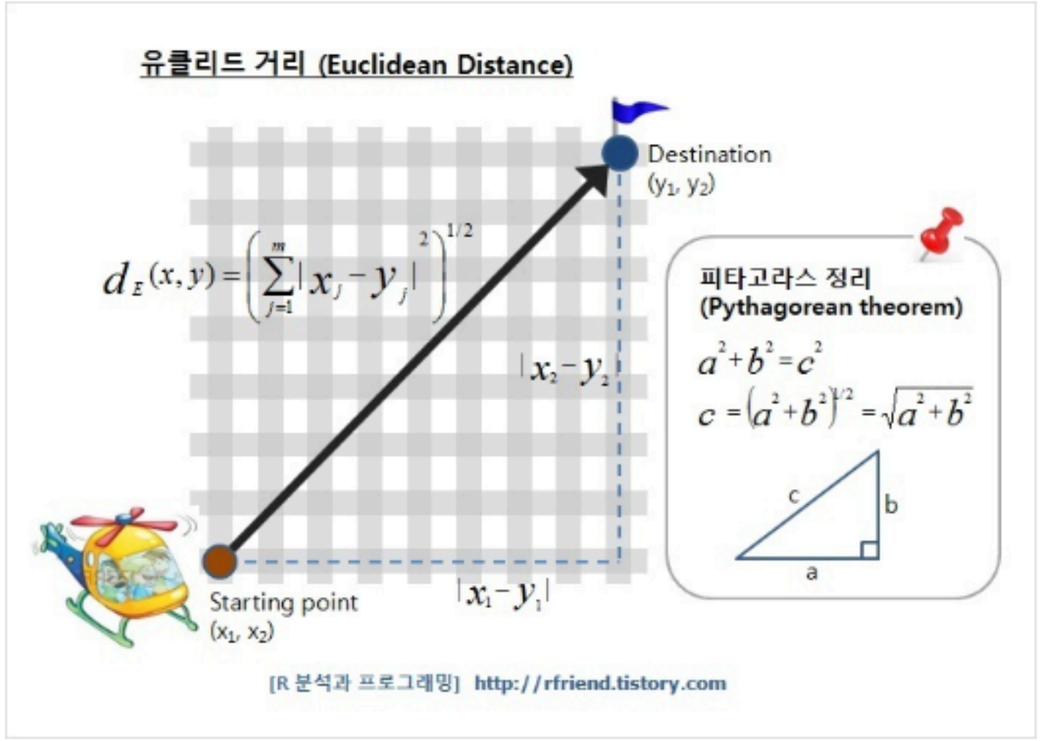
L2 norm (Euclidean distance), p=2 두 점사이의 최단거리

$$d = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2}$$

L2 norm (Manhattan distance)

$$L = \sum_{i=1}^{n} (y_i - f(x_i))^2$$

L2 Loss Function



L1 Regularization

$$\text{Cost} = \sum_{i=0}^{N}(y_i - \sum_{j=0}^{M} x_{ij}W_j)^2 + \lambda \sum_{j=0}^{M} |W_j|$$

$$w \rightarrow w - \boxed{\frac{\eta\lambda}{n} sgn(w)} - \eta \frac{\partial C0}{\partial w} \qquad sgn(w) = \begin{cases} 1, & w > 0 \\ 0, & w = 0 \\ -1, & w < 0 \end{cases}$$

L2 Regularization

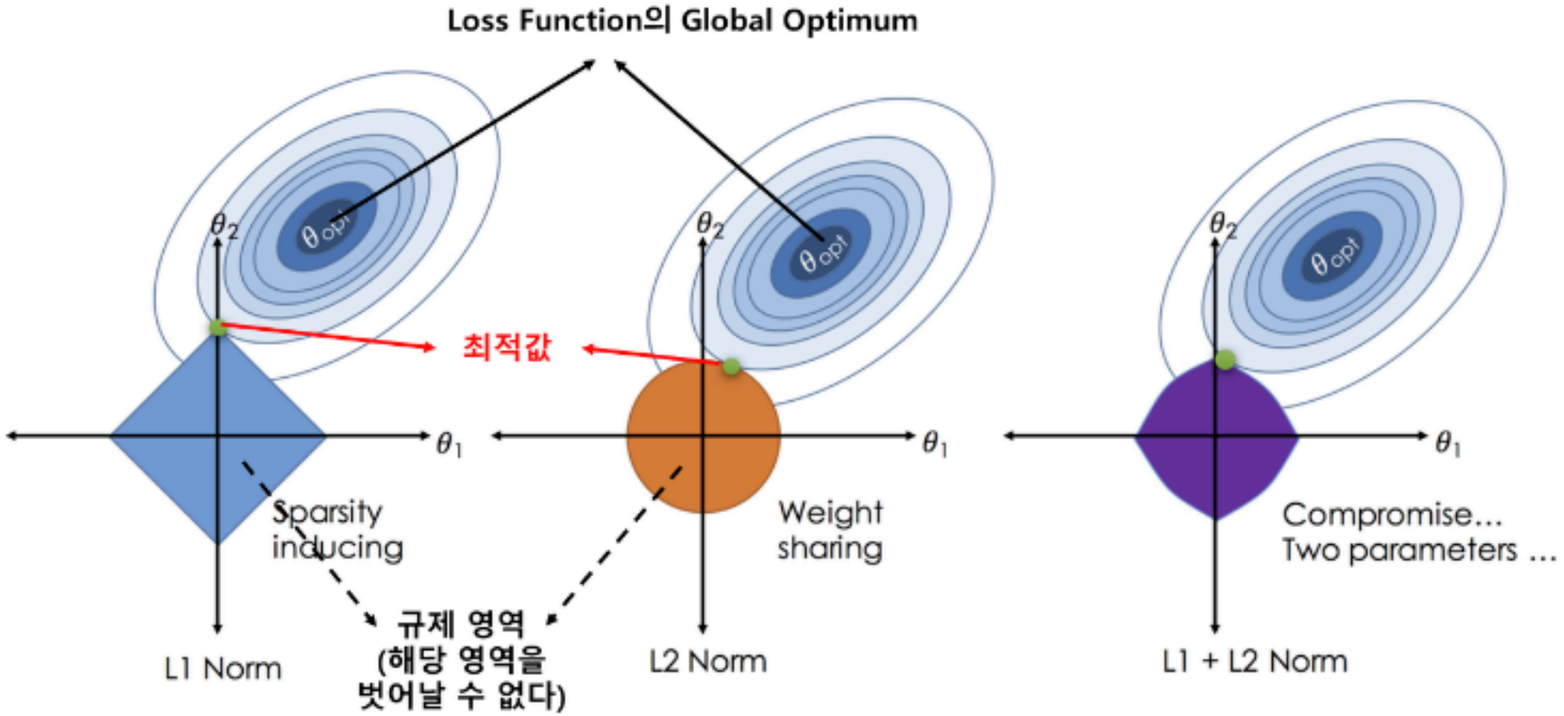
$$\text{Cost} = \sum_{i=0}^{N}(y_i - \sum_{j=0}^{M} x_{ij}W_j)^2 + \lambda \sum_{j=0}^{M} W_j^2$$

Loss function

Regularization Term

$$w \rightarrow w - \eta \frac{\partial C0}{\partial w} - \frac{\eta\lambda}{n}$$

$$= \boxed{\left(1 - \frac{\eta\lambda}{n}\right) w} - \eta \frac{\partial C_0}{\partial w}$$



계수 자체를 0 으로 만들어 버려서 가중치를 0으로 만듦
중요한 피처들을 선택할 때 하는 경우도 있다.
모델 Sparsity 0으로 만들어버리면 값자체가 0이 발생할 수 있음

최적화 그림을 보면 최적값에 대한 bias 손해보더라도 variance를 낮춰 Overffitig 낮춤

람다 값이 커질수록 규제 영역의 크기가 작아지게 되어 bias는 더 커지고 variance는 줄어들어 underffiting 가능

최적값은 규제 영역 내에 Global Optimum과 제일 가까운 지점

어떤 걸 써야 하나요?

L1 규제나 L2규제 중 어떤 것을 써야 하는가?

L1 norm 의 특징은 다른 점으로 이동하는데 다양한 방법이 있다.

L2 norm은 한 가지 방법이 있다.

데이터의 특성을 봐야겠지만, 기본적으로 Outlier 값들이 많은 경우는 오히려 L2가 영향을 더 많이 받을 수 있다.

기본적으로 모델링 전에 데이터의 특성, 분포 등을 바라보며 학습 데이터를 확인한다.

편미분 통해서 진행하게 되면서 -> L2 L1 가 달라지게 된다.
L1의 경우는 weight의 부호만 남게된다. weight의 크기에 따라 규제의 크기가 변하지 않은 경우가 잇다.
L2 보다는 효과 떨어질 수 있다.
(상대적인 것이지 절대적인 것은 아니다.)

L2 norm은 오차의 제곱을 사용한다. L1보다는 outlier 민감할 수 있다.

어떤 것이 좋다라고 판단하는 것 보다는 결국 데이터의 패턴을 보고 사용하면서 결국에는 특정 패턴이 보이면 어떤 규제가 더 성능 좋게 나올 수 있다.
L2 성능이 더 좋게 나온다고 하는 경우도 있고, 실제도 코드로 보면 L2 가 좋은 경우도 많다.