L1 ,L2

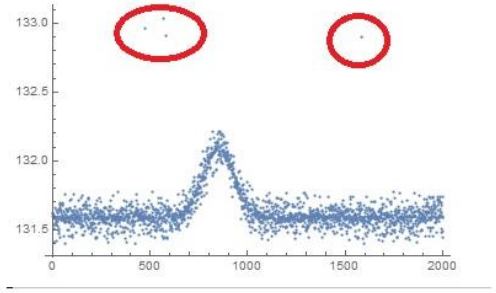
1. Loss

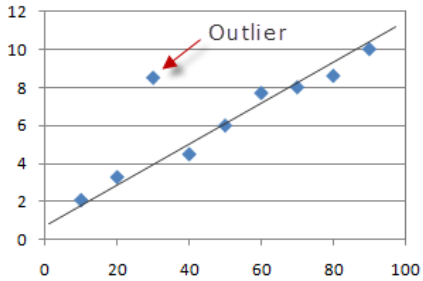
   A. L1 = $\sum_{i=1}^{n}|y_i - f(x_i)|$: 예측 값과 정답 값의 오차를 절대 값.

   B. L2 = $\sum_{i=1}^{n}|y_i - f(x_i)|$^2 : 예측 값과 정답 값의 오차의 제곱 값.

2. L1, L2 Loss의 Robustness

   A. 정의 : oulier에 대한 영향도를 나타 냄.





   B. L2 Loss : outlier위 그림에서 보이듯이 보통 큰 오차값을 나타냄. L2같은 경우 제곱이므로 outlier에 대해 더큰 오차를 주게 된다. 그러므로 outlier에 대해 민감 하게 작용한다. 그러므로 Robustness가 작다.

   C. L1 Loss : Outlier의 오차를 절대값 하여 사용하기 때문에 L2에 비해 outlier에 대해 덜 민감하게 작용. 그러므로 Robustness가 크다.

3. L1, L2 Loss의 Stability

   A. 정의 : 비슷한 Data에 대해 얼마나 일관적인 값을 예측 하는가?

   B. L2 Loss : Noise한 데이터의 경우 비슷한 범위안에 들어오기 때문에 오차값이 작다. 그러므로 0~1사이의 값들의 오차는 더 작아지기 때문에 예측 값에 변화가 거의 없다. 때문에 Stability가 크다.

C. L1 Loss : L1은 오차를 그대로 사용하기 때문에 오차가 그대로 영향에 미친다. 그러므로 Stability가 작다.

4. L1, L2 Loss의 결론

L2는 Outlier에 대해 영향도가 크고 노이즈에 대해 영향도가 작다.

L1은 Outler에 대해 영향도가 작고 노이즈에 대해 영향도가 크다.

여기서의 영향도의 작고, 크고는 상대적 이야기.

그러므로 ouliers가 효과적으로 적당히 무시되기를 바라면 L1이고 outlers를 주의 깊게 보고싶다면 L2를 쓰는 것이 좋을 듯.

5. L1, L2 Regularization

A. Regularization정의

i. 오버피팅을 방지하는 것. 딥러닝에서 말하자면 모델의 웨이트들이 너무 완벽하게 overfit되지 않도록 정규화 요소를 추가하는 것.

최소화 (손실(데이터 | 모델) + 복잡도(모델))

B. L2 Regularization

i. $C = C_0 + \frac{\lambda}{2n}\sum_w w^2$ C : Cost(Loss) Function, n : data갯수, w: 가중치

위 식을 backpropagation 하면 C_0값이 최소가 되도록 학습함과 동시에 $\frac{\lambda}{2n}\sum_w w^2$의 w들도 최소가 되는 방향으로 학습을 진행.

여기서 경사하강법에 의해 업데이이트 되는 w를 생각하면 다음과 같습니다.

$w_{updata} = w - \alpha\frac{\partial}{\partial w}C$ (기존 C = (wx – y)^2)

위에서 새롭게 정의한 C를 대입하여 편미분 하면

$$w_{updata} = \mathrm{w} - \alpha\left(\frac{\partial}{\partial w}C_0 + \frac{\lambda}{2n}w\right)$$

$$w_{updata} = \left(1 - \frac{\alpha\lambda}{n}\right)w - \alpha\frac{\partial}{\partial w}C_0$$

즉, 기존 Regularization이 없을 때와 비교하면 업데이트 w에 $\left(1 - -\frac{\alpha\lambda}{n}\right)$ 의 텀이 추가 되었다는 것이다. 그러므로 w가 작아지는 방향으로 학습이 진행된다. 그러므로 weight decay라고 말하기도 한다. 이는 특정 가중치가 비이상적으로 커져서 학습의 효과에 큰 영향을 끼치는 것을 막아준다.

C. L1 Regularization

i. $\mathrm{C} = C_0 + \frac{\lambda}{2n}\sum_w |w|$ C : Cost(Loss) Function, n : data갯수, w: 가중치

$$w_{updata} = \mathrm{w} - \alpha\left(\frac{\partial}{\partial w}C_0 + \frac{\lambda}{2n}sgn(w)\right)$$

$$w_{updata} = w - \alpha\frac{\lambda}{2n}sgn(w) - \alpha\frac{\partial}{\partial w}C_0$$

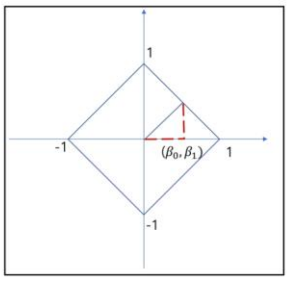
$sgn(x)$: 부호 함수로써 $\begin{cases} 1 & x > 0 \\ 0 & x = 0 \\ -1 & x < 0 \end{cases} \quad (x \in \mathbb{R})$ 이다.

즉, weght값 자체를 줄이는 것이 아니라 w의 부호에 따라 상수 값을 빼주는 방식으로 Regularization을 해주는 형태로 진행.

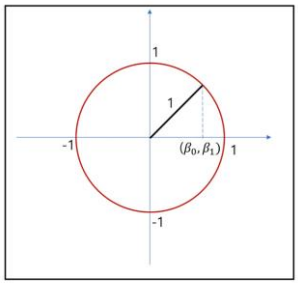
D. L2, L1 Regularization 의 결론

L1 : 웨이트에 상수 값을 빼주기 때문에 작은 가중치는 거의 0으로 수렴되어 진다. 때문에 몇몇 중요한 가중치만 살아남게 되고 이를 Feature Selection이라고 한다. 이러한 이유로 L1의 경우 가중치는 saprse한 형태를 가질 수 있다.

L2 : 위 내용에서 설명 완료.



L1: $||\beta||_1 = |\beta_0| + |\beta_1| = 1$



L2: $||\beta||_2 = \sqrt{(\beta_0)^2 + (\beta_1)^2} = 1$

6. Lasso Regression, Ridge Regression

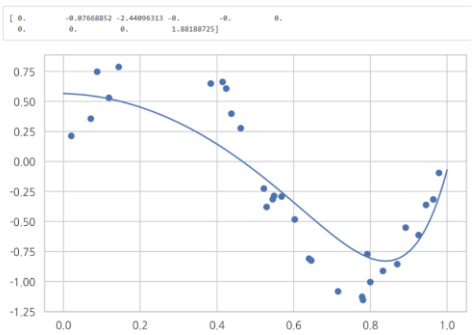
A. Lasso Regression : L1 Regularization

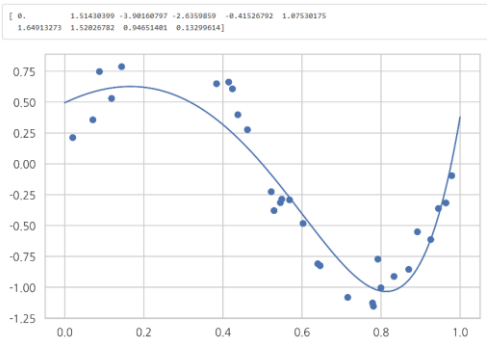
가중치들 절대값의 합을 최소화 하는 것을 제약조건으로 추가

$$\mathbf{w} = \arg\min_{w}(\sum_{i=1}^{N} e_i^2 + \lambda \sum_{j=1}^{N} |w_j|)$$

B. Ridge Regression : L2 Regularization

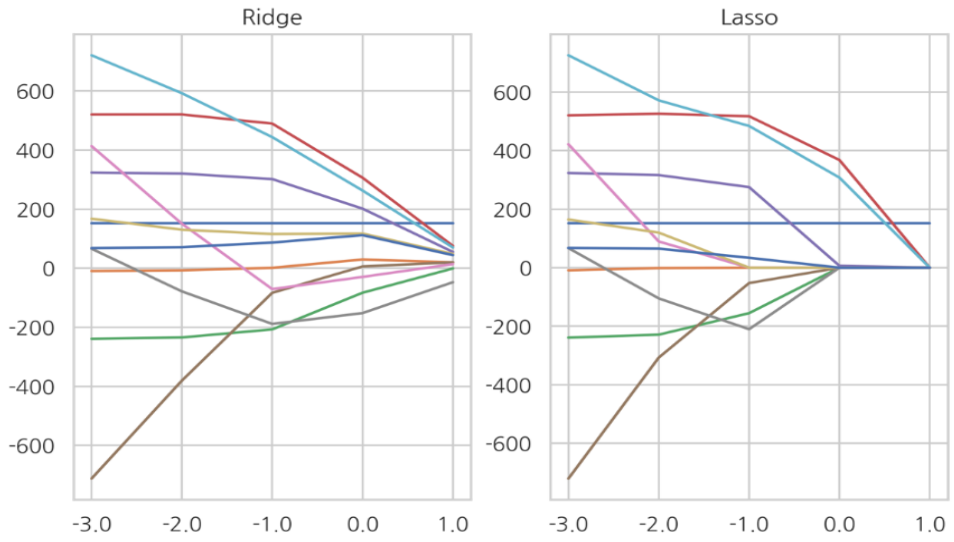
$$\mathbf{w} = \arg\min_{w}(\sum_{i=1}^{N} e_i^2 + \lambda \sum_{j=1}^{N} w_j^2)$$





Lasso Ridge

결론적으로, Ridge 모형은 가중치 계수를 한꺼번에 축소 시키는데 반해 Lasso 모형의 경우 일부 가중치 계수가 먼저 0으로 가는 특성이 있다.(밑의 그림이 w의 변화 추이를 보여주고 있음)


Ridge
Lasso
600
400
200
0
-200
-400
-600
-3.0
-2.0
-1.0
0.0
1.0