CNU 데이터 분석 교육

8th lecture
"Linear Analysis-2"

2021 - 11 - 09

수학과 최명수

지난시간?

1. Linear Regreesion on 2-dimension

   -find a, b which minimize the loss function(RMSE)
   -by 'grid search' method

지도학습(Supervised learning)

뭘 지도했느냐?

	Y 키	X 몸무게
0	125.8	27.3
1	124.3	25.4
2	119.2	23.5
3	115.0	20.0
4	120.0	33.5

$$\text{키} = f(\text{몸무게}, a, b)$$
$$= a \times \text{몸무게} + b$$

$$Error(f, X, Y, a, b) = \sum_{i=0}^{n} (f(x_i, a, b) - y_i)^2$$
$$= \sum_{i=0}^{n} (ax_i + b - y_i)^2$$
$$where\ X = \{x_1, x_2, \cdots, x_n\}, \qquad Y = \{y_1, y_2, \cdots, y_n\}$$

충남대학교 수학과 최명수

오늘은 무엇을?

1. 다양한 loss function을 이용한 선형회귀모델 제작
   -MAE, log_cosh…

2. Linear Regression 함수 제작
   -코딩 능력 향상을 위한…

복습

1. 학생건강검사 결과분석 데이터를 불러온 후,
2. 키와 몸무게 열을 추출하여 결측치를 제거하고,
3. 선형함수(height_pred) 함수를 제작 한 후,
4. 임의의 300개의 데이터에 관하여 a의 범위를 (0.7, 1.3)으로, b의 범위를 (80, 120)으로 각각 30등분 하고,
5. Loss function을 RMSE로 설정한 후 Grid search method를 이용하여 최적의 a,b를 찾아내시오

6. 최적의 a,b를 이용하여 학습모델 결과를 시각화 하시오.

다양한 loss function

1. p-norm

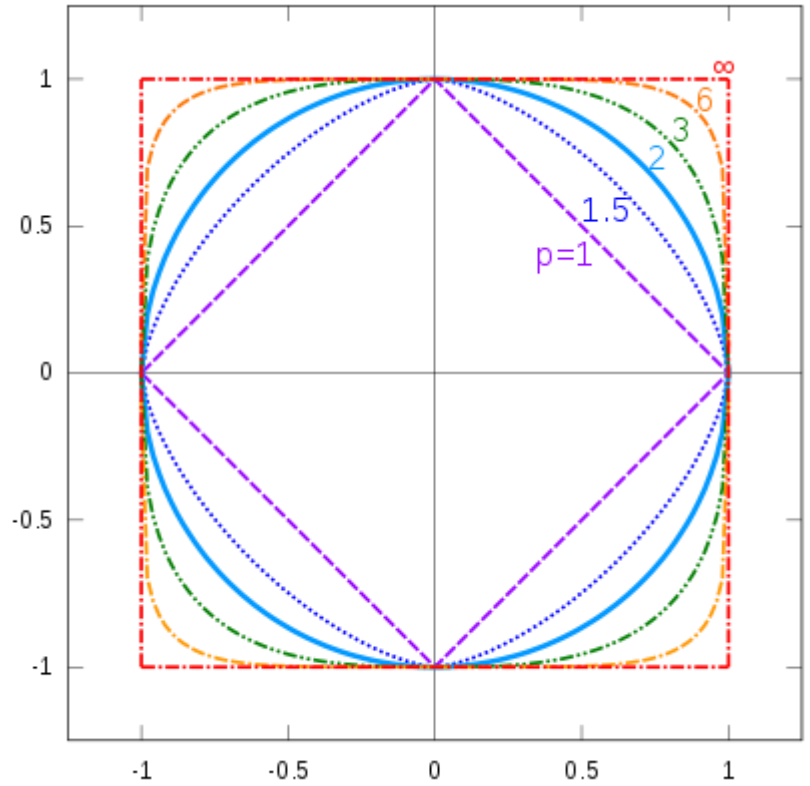
RMSE는 2-norm이다!

p-norm [edit]

Main article: L^p space

Let $p \geq 1$ be a real number. The p-norm (also called ℓ_p-norm) of vector $\mathbf{x} = (x_1, \ldots, x_n)$ is[9]

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



다양한 loss function

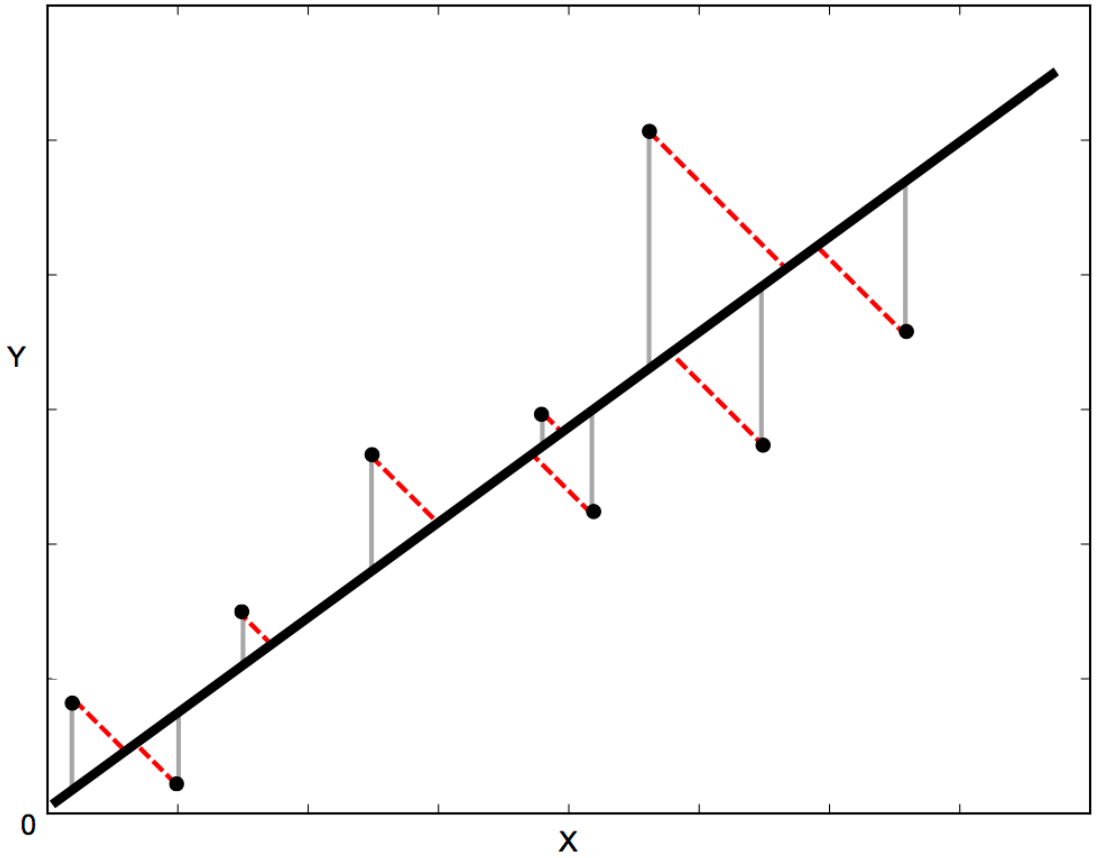
2. Orthogonal Distance

$$\text{distance}(ax + by + c = 0, (x_0, y_0)) = \frac{|ax_0 + by_0 + c|}{\sqrt{a^2 + b^2}}.$$

$$y = ax + b$$

$$dist(ax - y + b = 0, (x_0, y_0)) = \frac{|ax_0 - y_0 + b|}{\sqrt{a^2 + 1}}$$

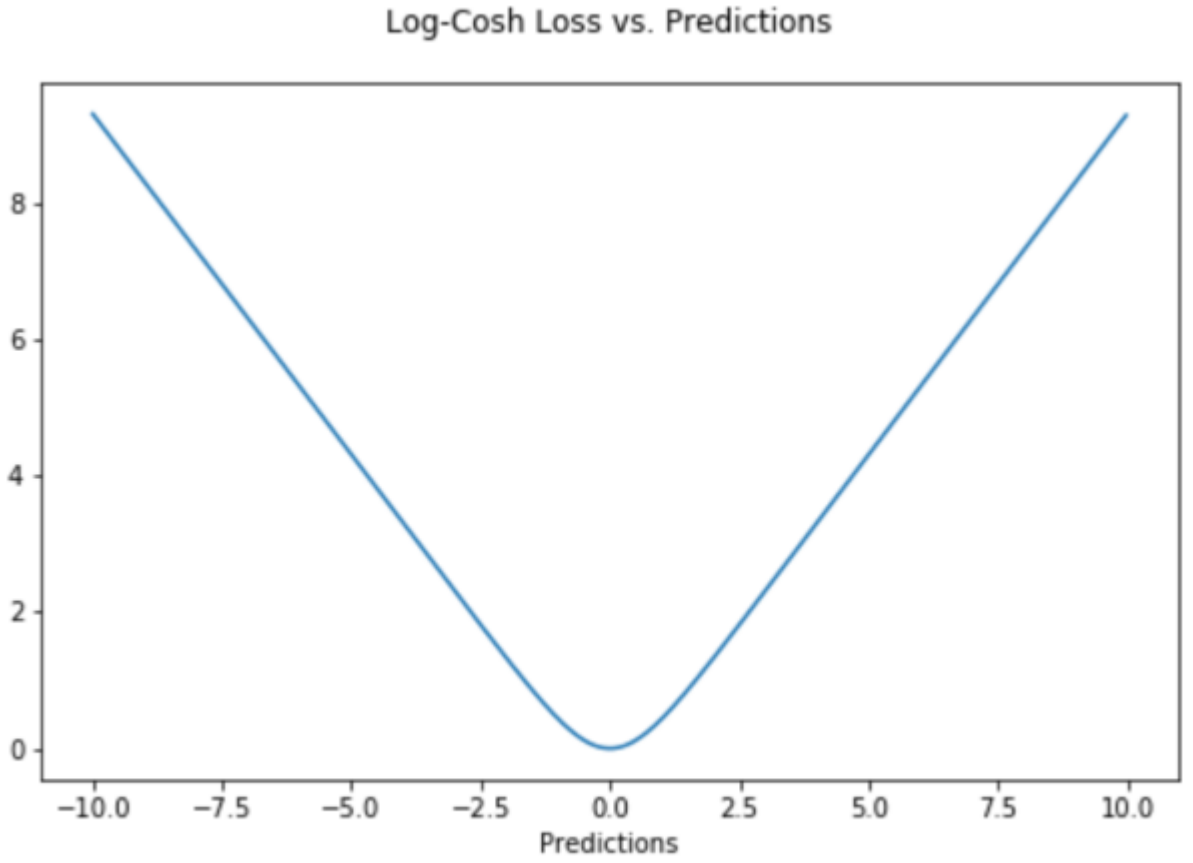
$$= \frac{|\hat{y} - y_0|}{\sqrt{a^2 + 1}} = \frac{|diff|}{\sqrt{a^2 + 1}}$$



다양한 loss function

3. log-cosh

$$L(y, y^p) = \sum_{i=1}^{n} \log(\cosh(y_i^p - y_i))$$

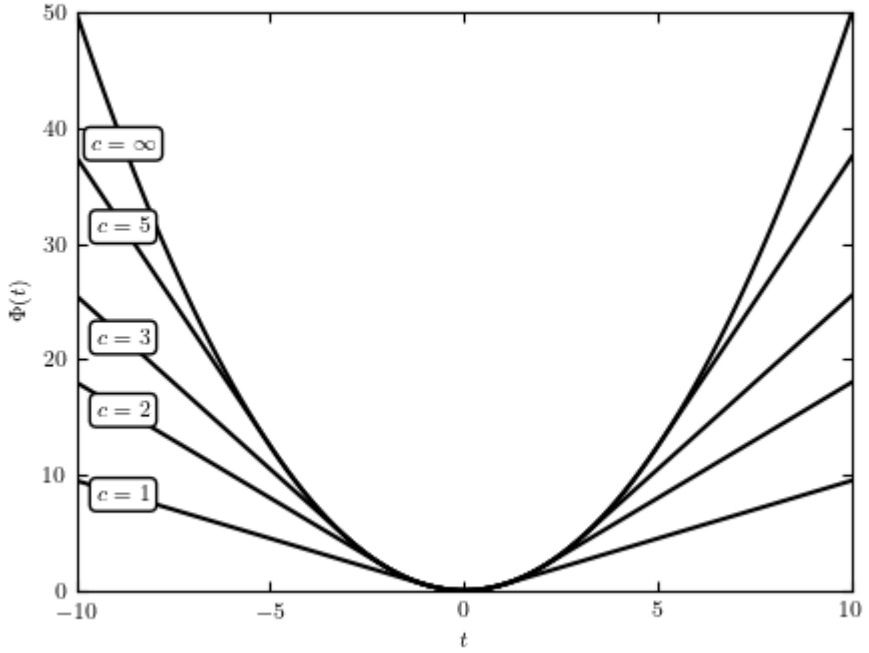


Plot of Log-cosh Loss (Y-axis) vs. Predictions (X-axis). True value = 0

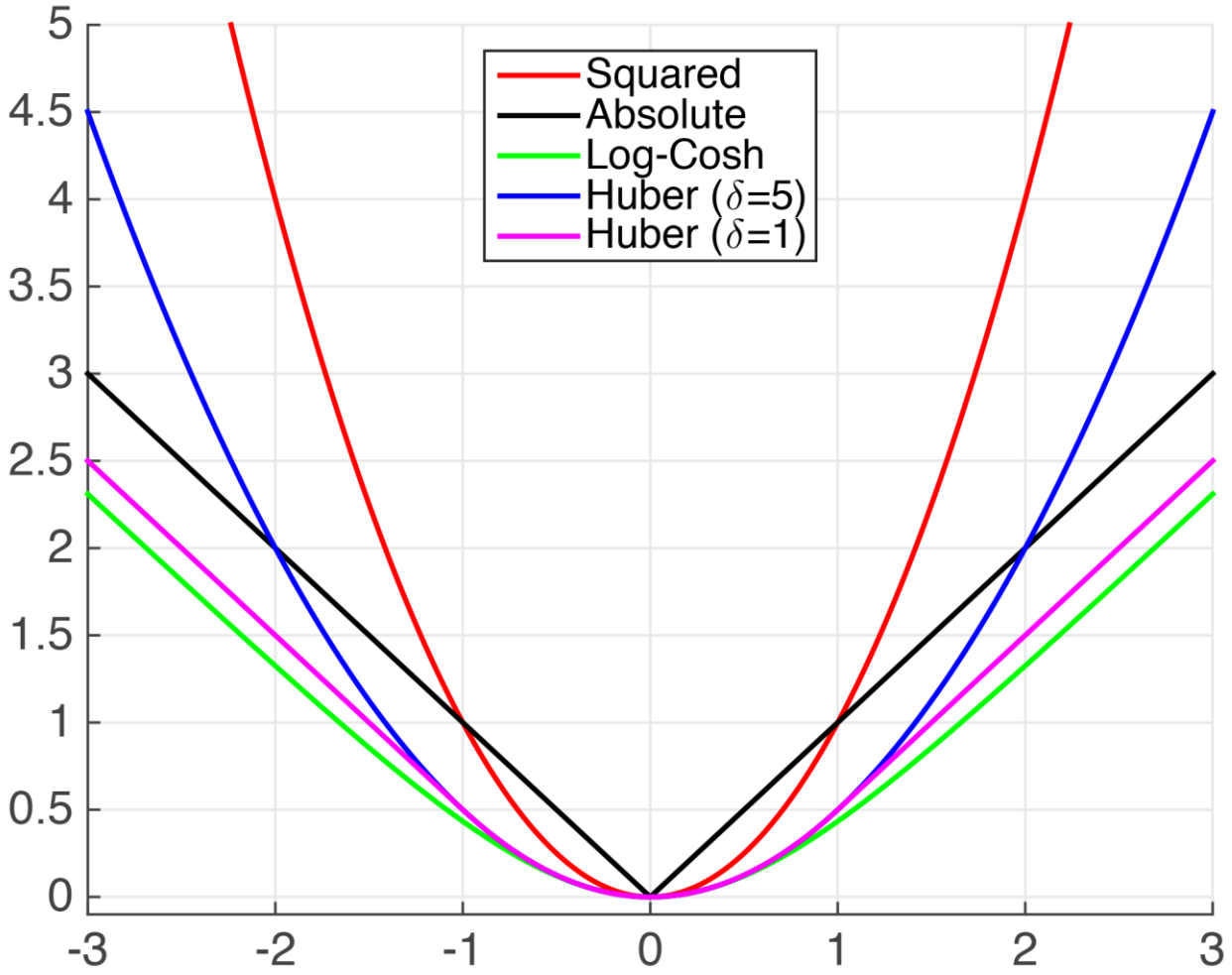
다양한 loss function

4. Huber function

$$L_\delta(a) = \begin{cases} \frac{1}{2}a^2 & \text{for } |a| \le \delta, \\ \delta \cdot \left(|a| - \frac{1}{2}\delta\right), & \text{otherwise.} \end{cases}$$



다양한 loss function



문제 1

다음 4개의 함수를 만드시오

P-norm(diff, p)

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

OD(diff)

$$y = ax + b$$

$$dist\big(ax - y + b = 0, (x_0, y_0)\big) = \frac{|ax_0 - y_0 + b|}{\sqrt{a^2 + 1}}$$

$$= \frac{|\hat{y} - y_0|}{\sqrt{a^2 + 1}} = \frac{|diff|}{\sqrt{a^2 + 1}}$$

log_cosh(diff)

$$L(y, y^p) = \sum_{i=1}^{n} \log(\cosh(y_i^p - y_i))$$

Huber(diff, delta)

$$L_\delta(a) = \begin{cases} \frac{1}{2}a^2 & \text{for } |a| \le \delta, \\ \delta \cdot \left(|a| - \frac{1}{2}\delta\right), & \text{otherwise.} \end{cases}$$

문제 1

정답

P-norm(diff, p)

```python
# p-norm
def p_norm(diff,n):
  return sum(abs(diff) ** n) ** (1/n)
```

OD(diff)

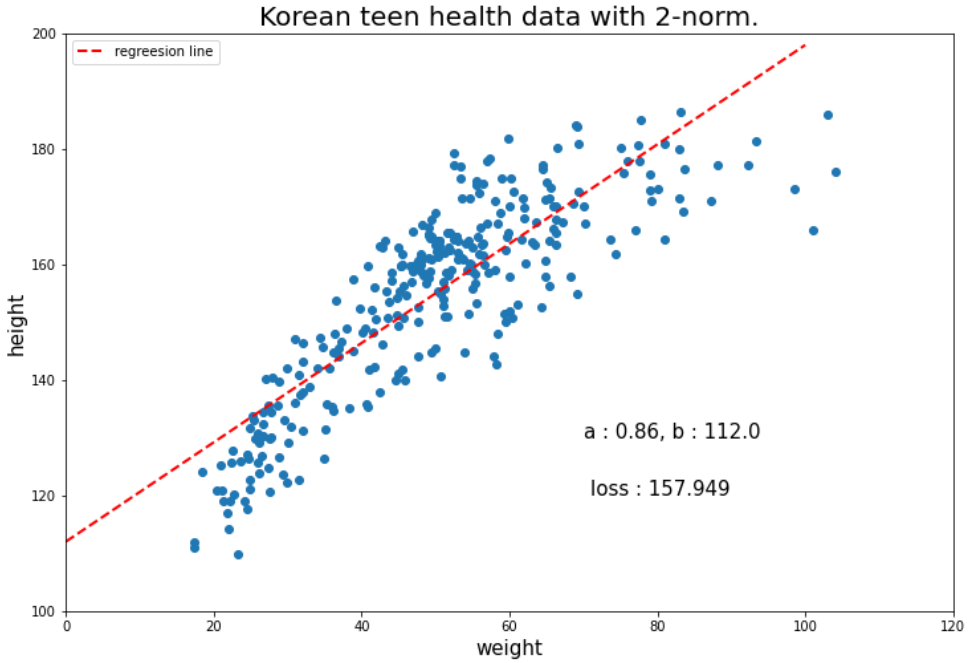
```python
# Orthogonal Distance
def OD(diff, a):
  return sum(abs(diff))/np.sqrt(a**2 + 1)
```

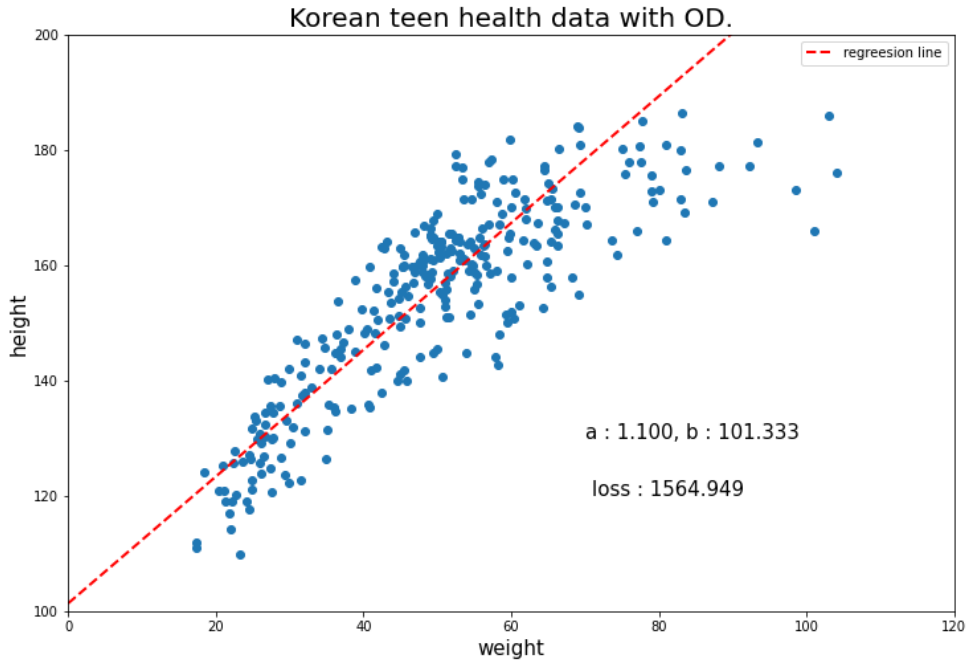
log_cosh(diff)

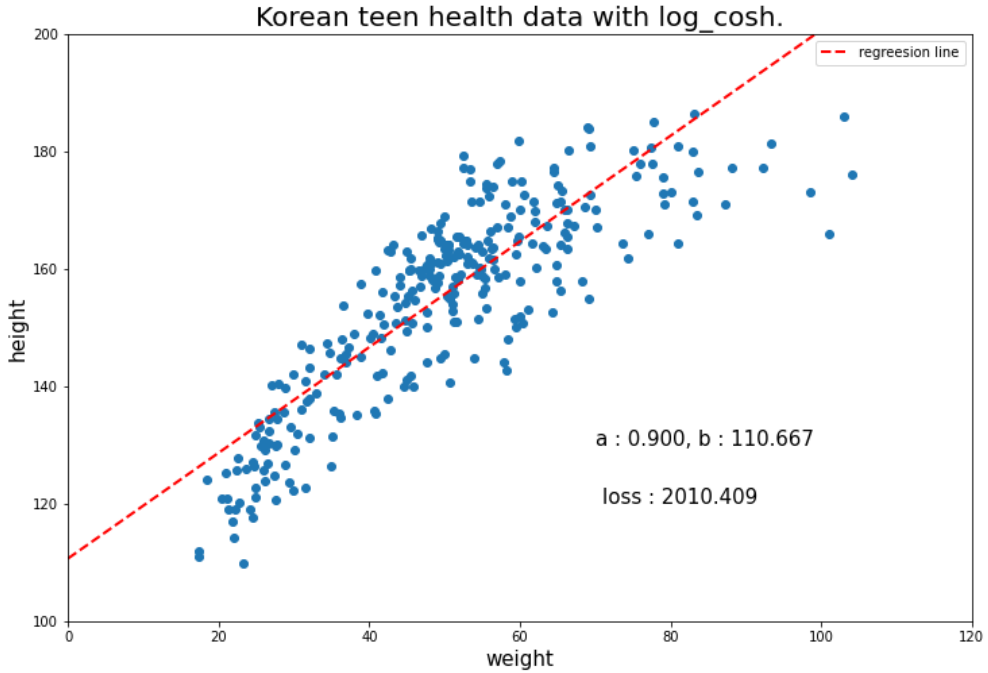
```python
# log cosh
def log_cosh(diff):
  return sum(np.log(np.cosh(diff)))
```

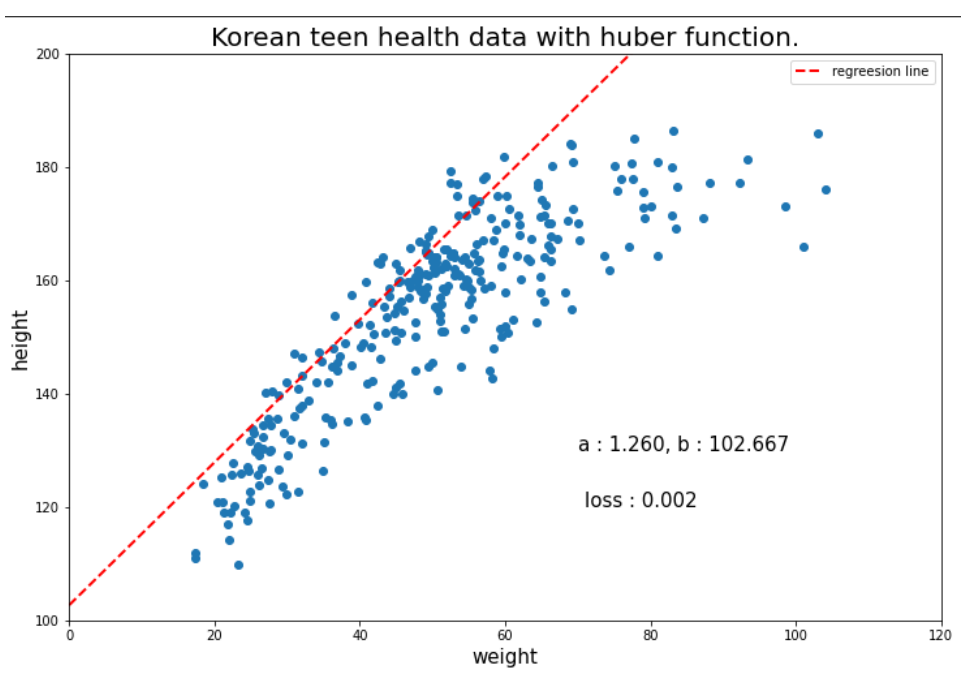
Huber(diff, delta)

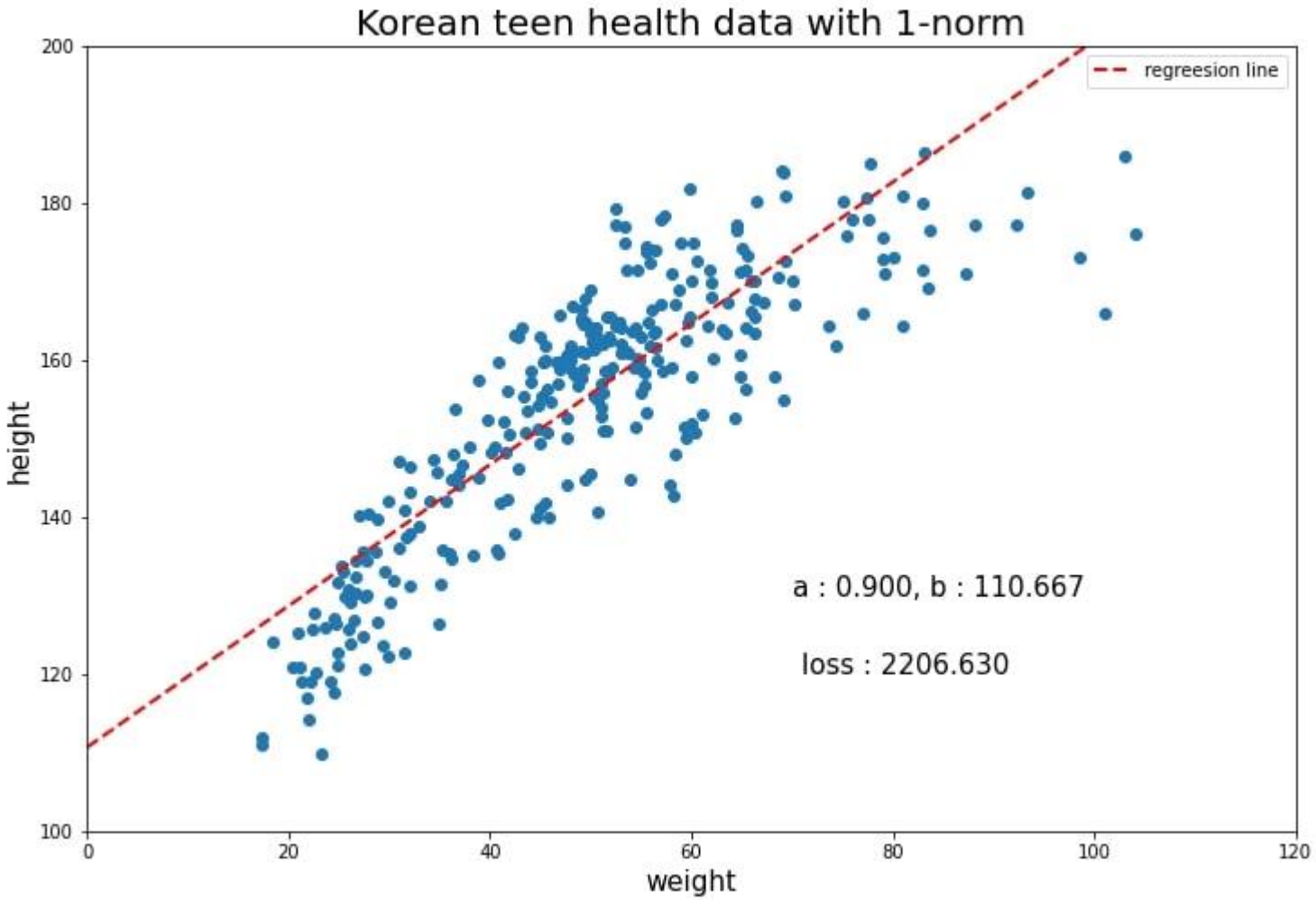
```python
# huber function
def huber(diff, delta):
  result = 0

  for value in diff:
    if abs(value) <= delta:
      loss = (value ** 2)/2
    else:
      loss =  delta * (abs(value) - delta/2)

  result += loss
  return(result)
```


Korean teen health data with 2-norm.
regreesion line
a : 0.86, b : 112.0
loss : 157.949
height
weight

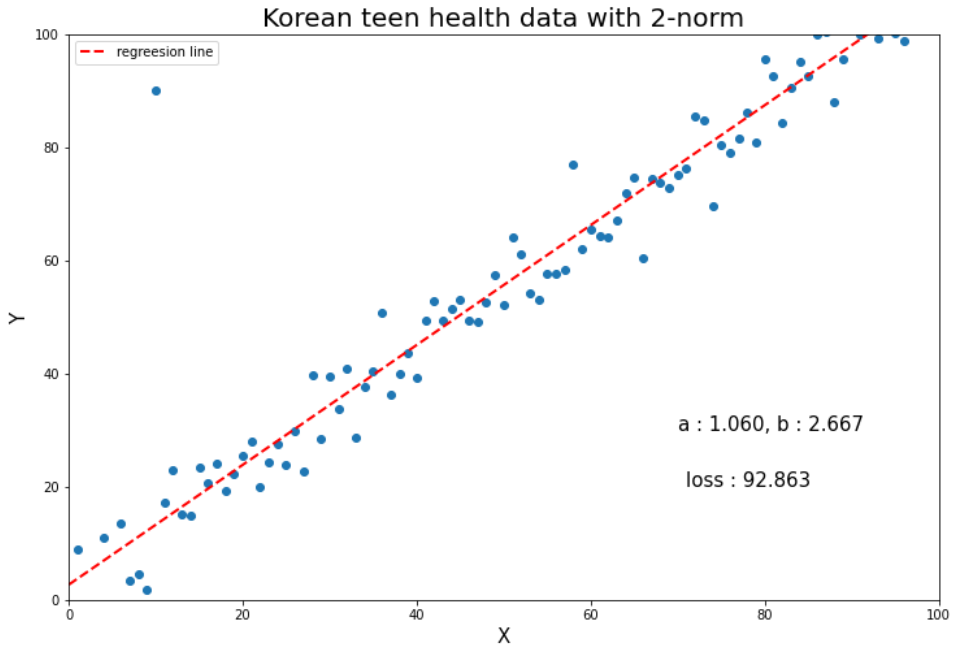

Korean teen health data with OD.
regreesion line
a : 1.100, b : 101.333
loss : 1564.949
height
weight

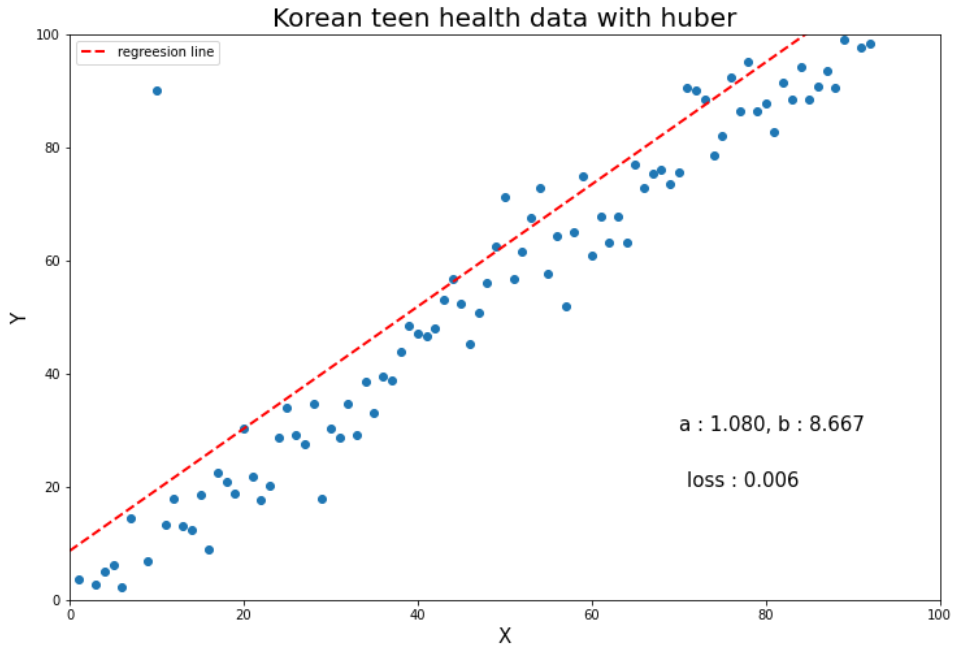

Korean teen health data with log_cosh.
regreesion line
a : 0.900, b : 110.667
loss : 2010.409
height
weight

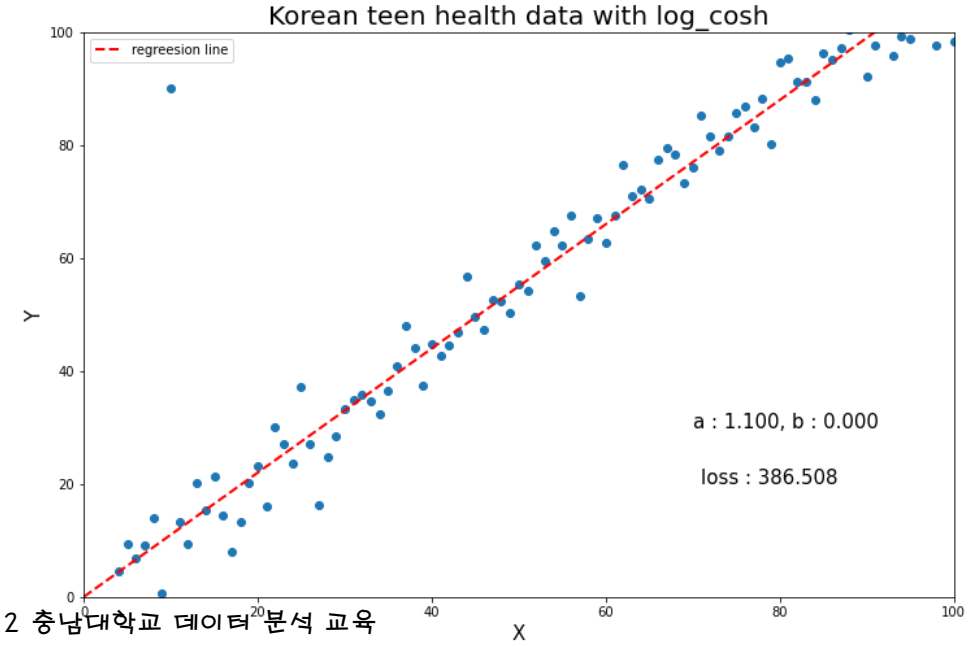

Korean teen health data with huber function.
regreesion line
a : 1.260, b : 102.667
loss : 0.002
height
weight

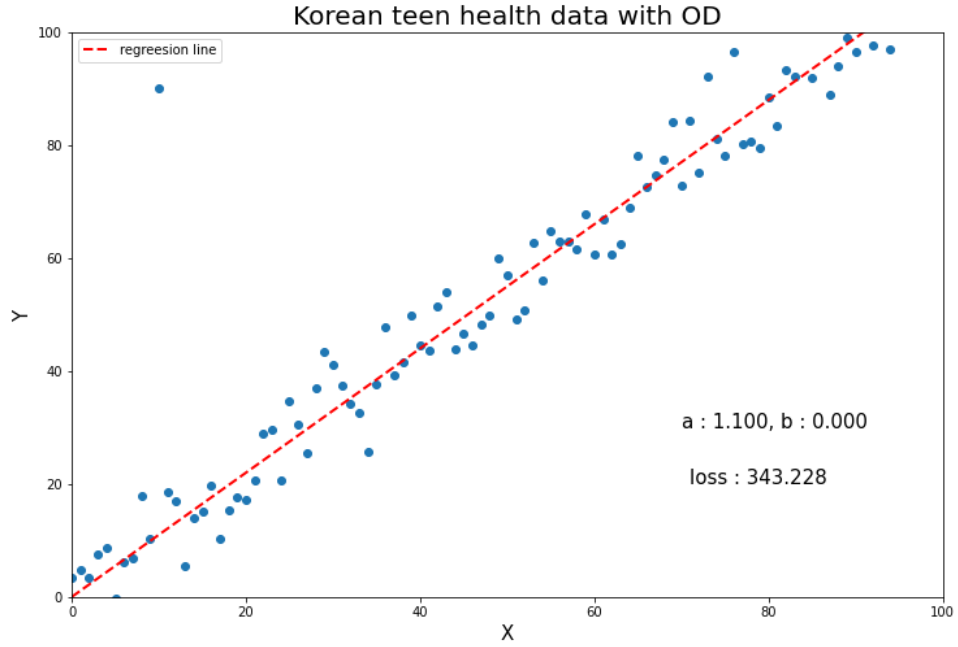

Korean teen health data with 1-norm
regreesion line
a : 0.900, b : 110.667
loss : 2206.630
height
weight
200
180
160
140
120
100
0
20
40
60
80
100
120

Loss function에 따른 민감도

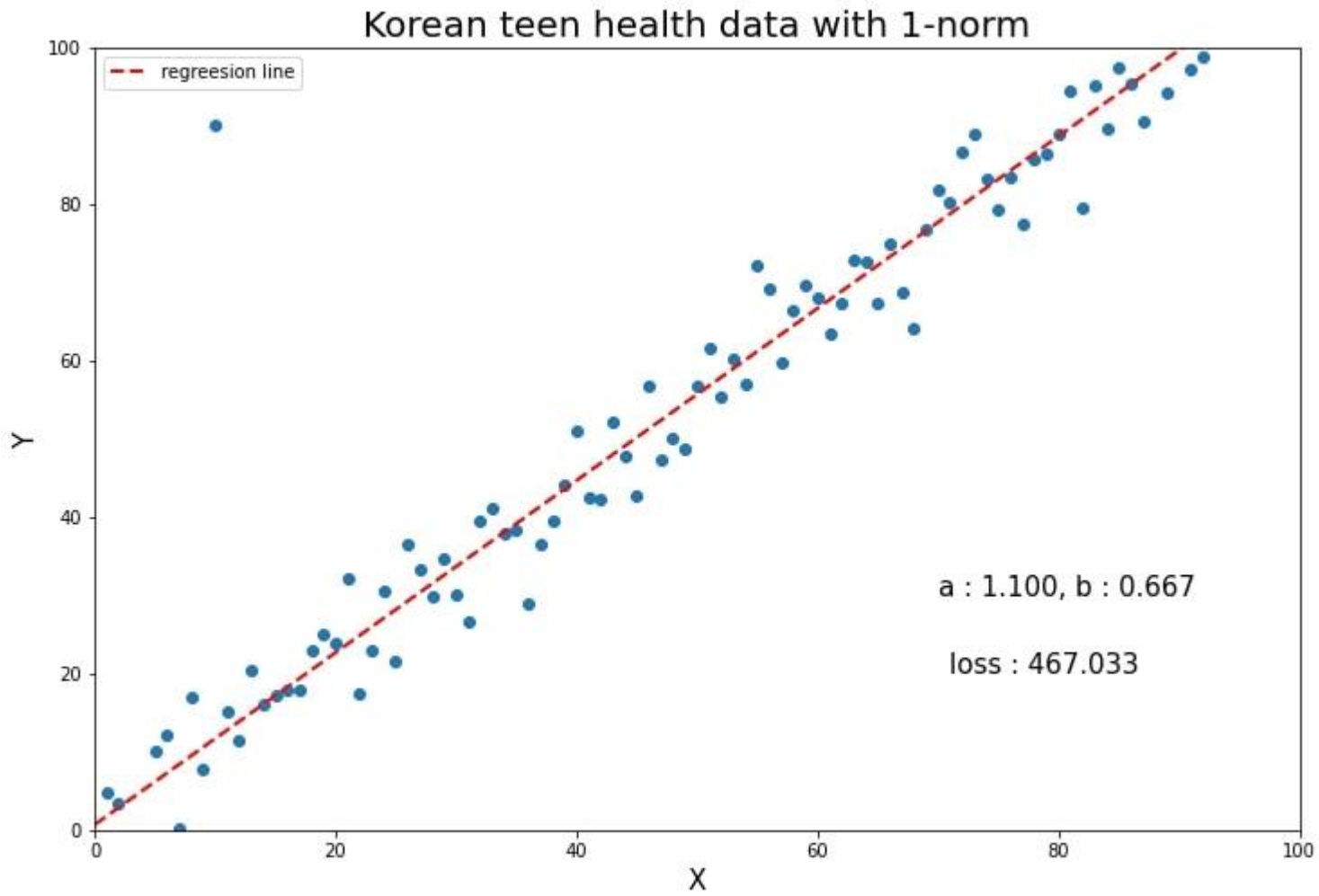








Loss function에 따른 민감도

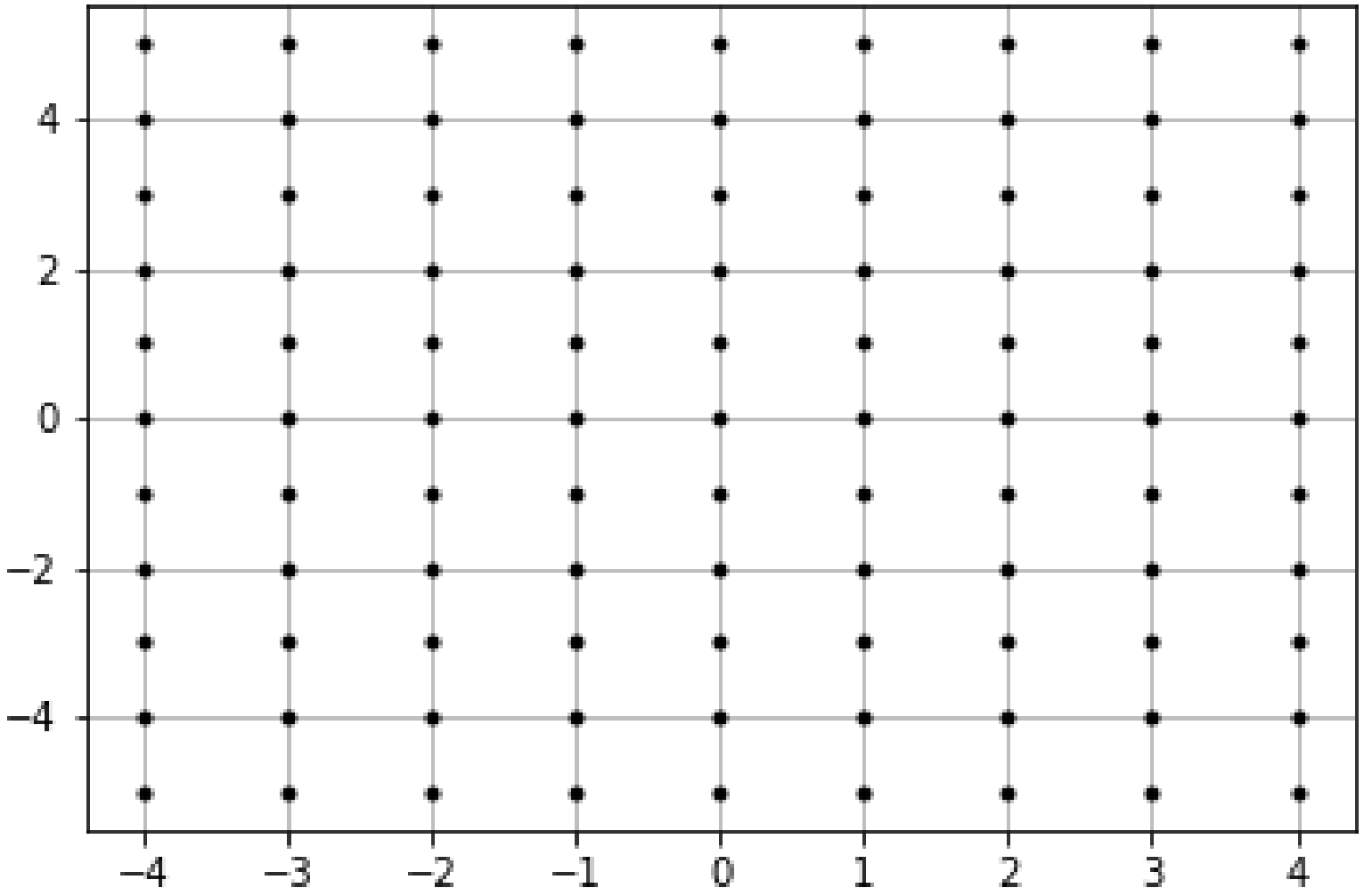


다양한 loss function

- Loss 값이 지들 맘대로라서 평가하기가 힘들다!!
  -> 다양한 loss에 대해서 일괄적인 평가수치는 없을까?

- Loss 값을 최소화 하는 방법을 수학적으로 살펴본다면?

mesh



https://www.geeksforgeeks.org/numpy-meshgrid-function/

정리 요약

1. Loss function에 따라 다른 성능을 보일 수 있다.

   -> 분석하고자 하는 데이터의 특성에 따라 선택하면 된다.

2. Loss값은 x, y 에 따른 3차원 곡면으로 생각할 수 있다.

   -> $z = f(x, y)$ 의 최소값 구하기?

아직 끝이 아니야…

문제2

- Grid Search method를 이용하여 최적의 파라미터(a,b)를 구해주는 일반적인 Linear Regression 함수를 만들어 보자.

- Input : X, Y, a_range, b_range, loss_function, loss_hyperparameter
- Ouput : min(a), min(b), min(loss), loss_list

* 참고 : loss_function은 위에서 사용한 4가지를 이용하는데, OD나 log_cosh는 hyperparameter 가 필요없다.

아직 끝이 아니야…

힌트

```
def LR(X, Y, a_range, b_range, loss_function, loss_hyperparameter = 2):

  a_list = [],  b_list = [], loss_list = []

  for a in a_range:
    for b in b_range:
      diff = Y - 직선함수(X, a, b)

      if loss_function == "p_norm":
        loss = p_norm(diff, loss_hyperparameter)
      elif loss_function == "다른거"
```

<- 이거 그대로 하면 오류남! 여백이 없어서 표시만 이렇게 한거니까 코딩 작성시는 Enter 키를 이용해서 밑으로 내려야 함!

아직 끝이 아니야…

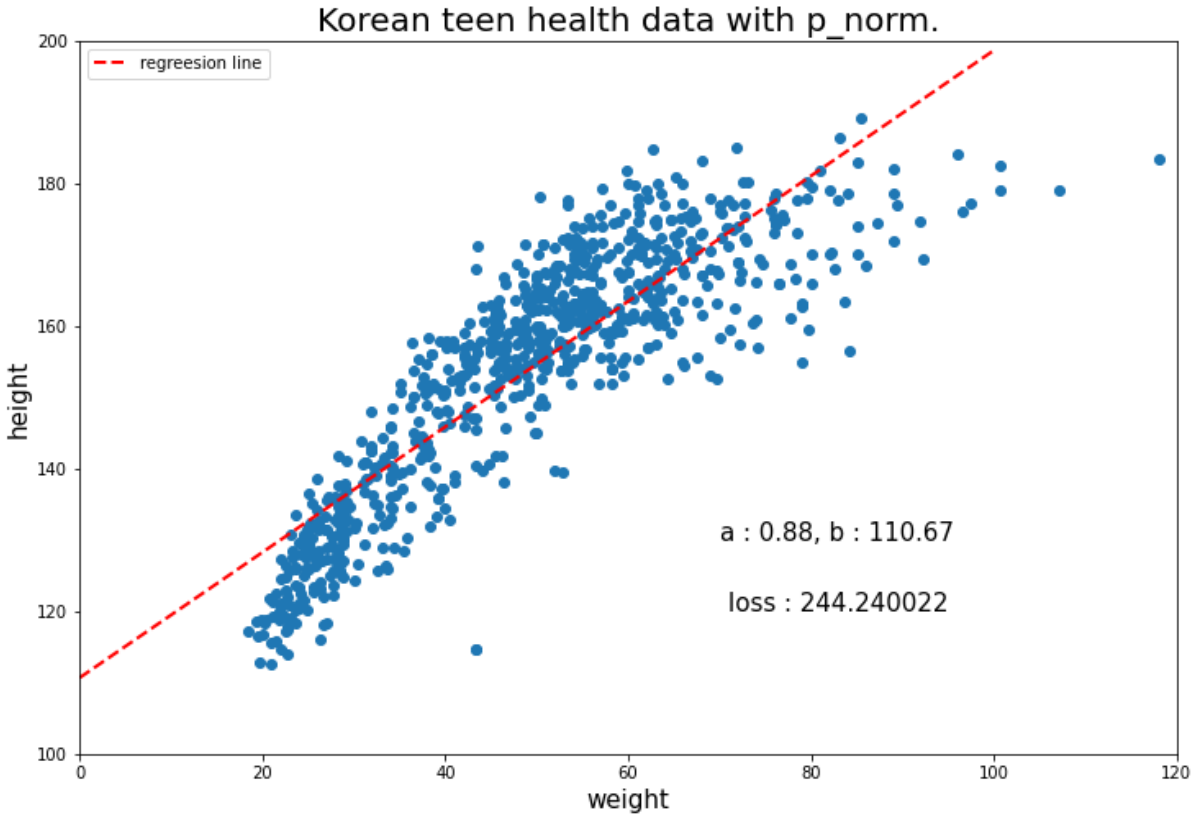
정답

```
def f(x,a,b):
  return a*x + b


def LR(X, Y, a_range, b_range, loss_function, loss_hyperparameter = 2):

  a_list = []
  b_list = []
  loss_list = []


  for a in a_range:
    for b in b_range:
      diff = Y - f(X, a, b)

      if loss_function == "p_norm":
        loss = p_norm(diff, loss_hyperparameter)

      elif loss_function == "OD":
        loss = OD(diff, a)

      elif loss_function == "log_cosh":
        loss = log_cosh(diff)

      elif loss_function == "huber":
        loss = huber(diff, loss_hyperparameter)

      else:
        print("Wrong loss function!")

      a_list.append(a)
      b_list.append(b)
      loss_list.append(loss)

  df_loss = pd.DataFrame([a_list, b_list, loss_list]).T
  df_loss.columns = ["a",'b','loss']

  df_min = df_loss[df_loss['loss'] == df_loss.min()[2]]

  return float(df_min["a"]), float(df_min["b"]), float(df_min['loss']), loss_list
```

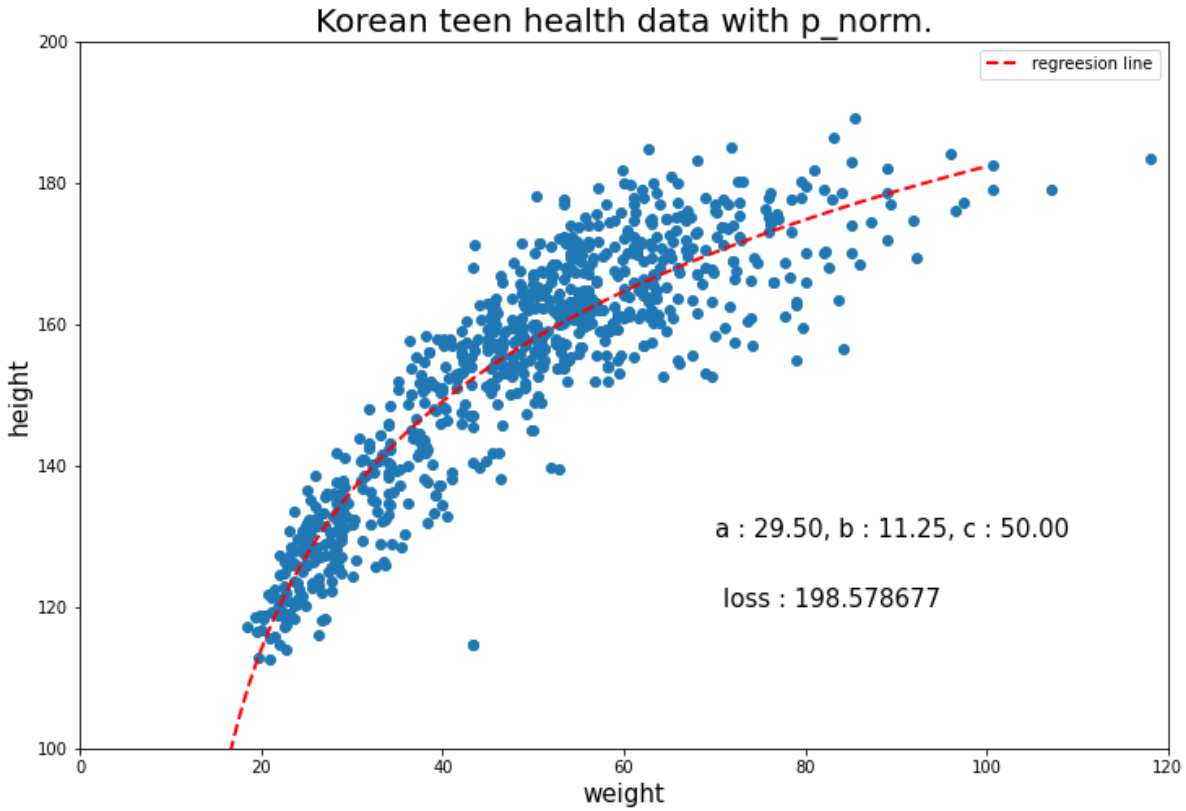
모델 바꾸기

Linear 모델말고 다른 모델로 한다면?

예를 들어 로그함수$(y = a \times log(x - b) + c)$를 이용해 추정한다면?



Linear regreesion



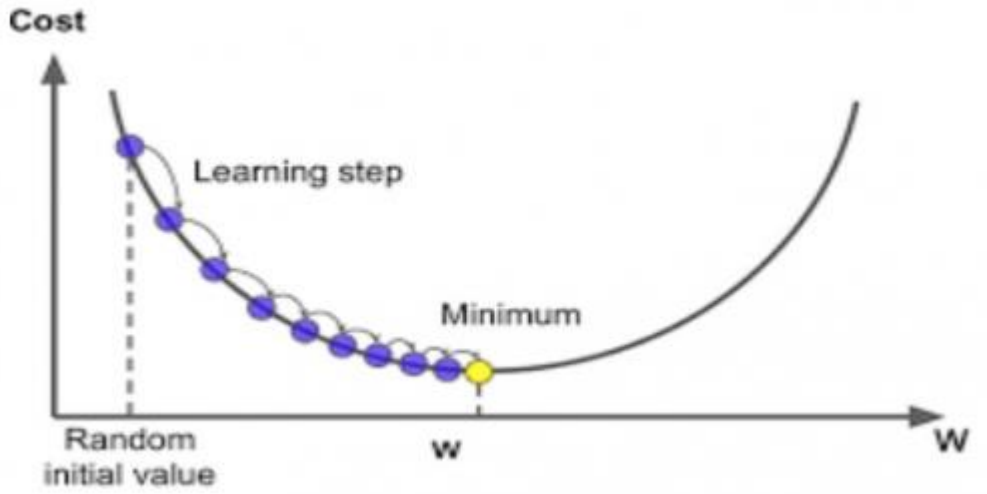
Log regreesion

공지

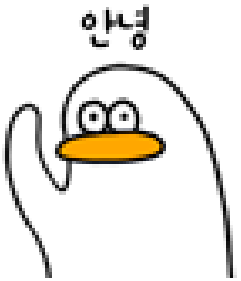
1. 모델은 만들면 모델이 정말 잘 작동되는지! 까지 확인을 해야한다.

2. 모델의 종류에 따라 성능도 많이 달라진다.

3. 다음시간에는 loss를 최소화 시키는 다른 방법들에 대해 배울 것



끝



담에봅시당