Loss Functions For Regression

Second Study Session

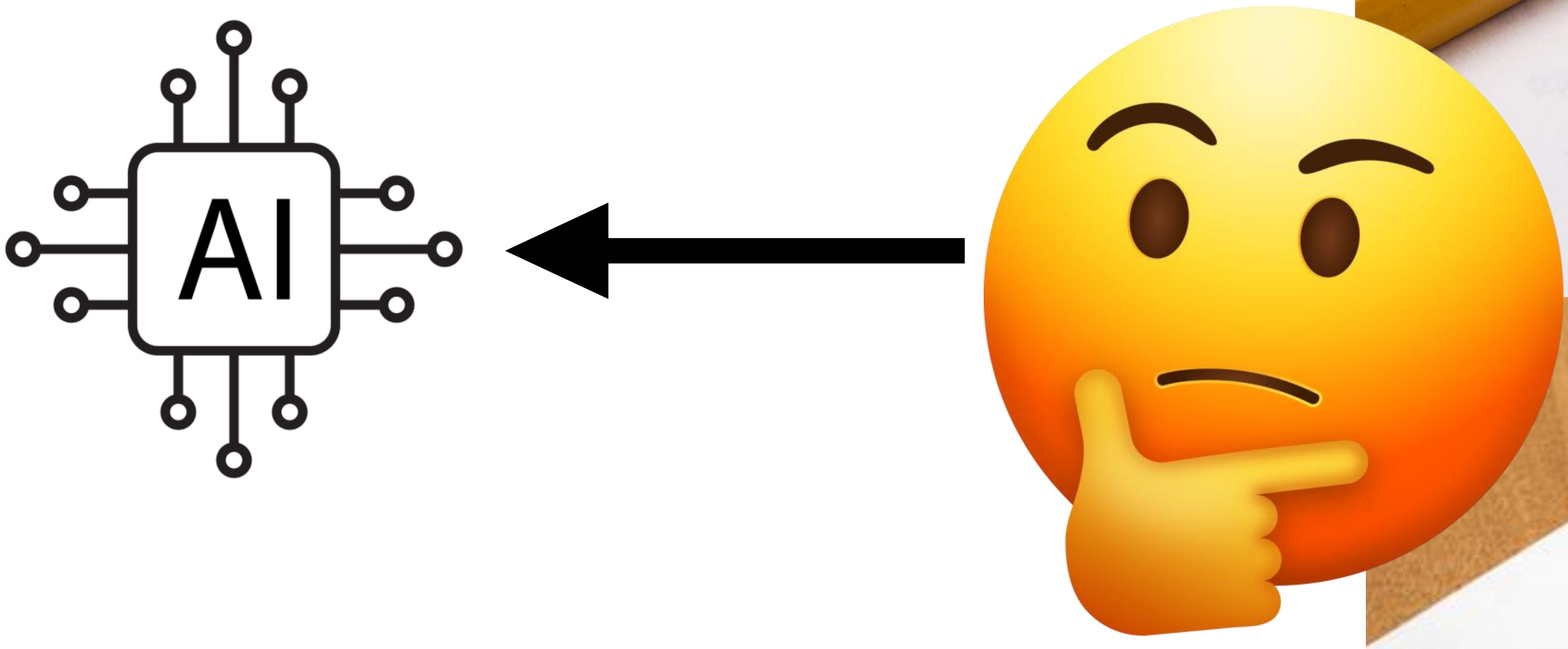
Lee JaeHyeong

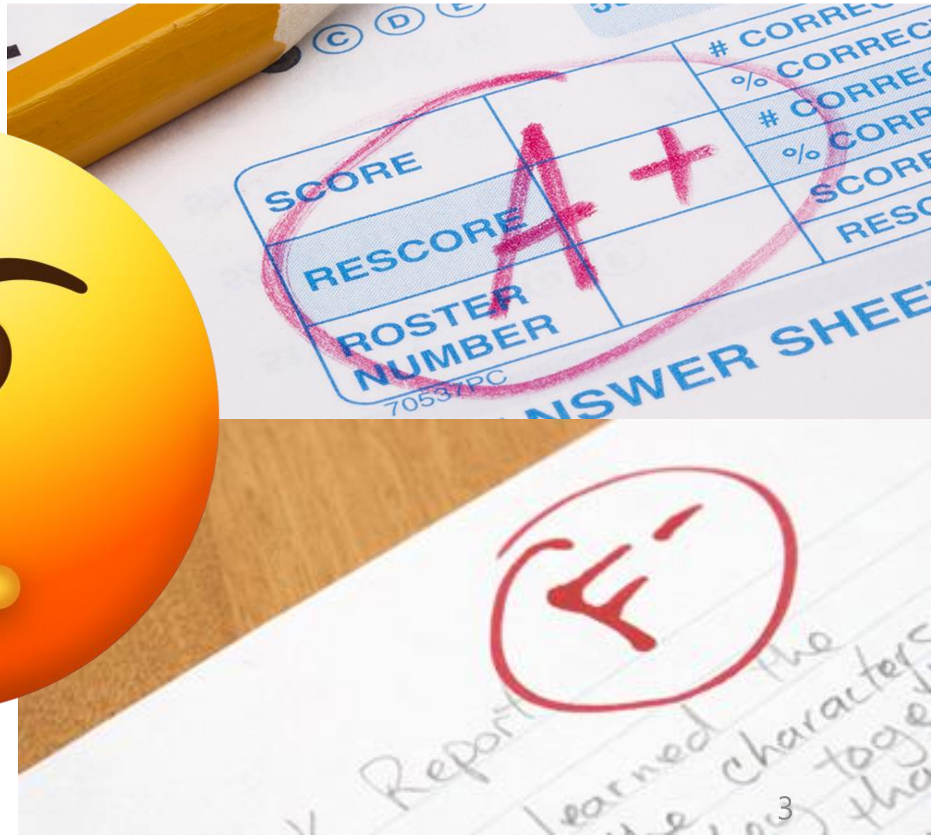
Context

- ✓ Loss Function?
- ✓ Regression vs Classification
- ✓ 3 Loss Functions for Regression
- ✓ Summary

Loss function?

How to say our model is working **good or bad**?
by which **criterion**?

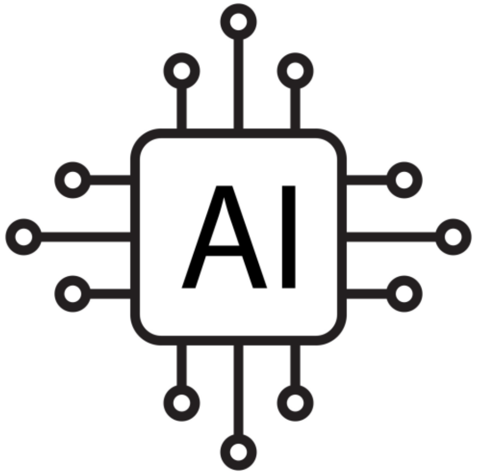


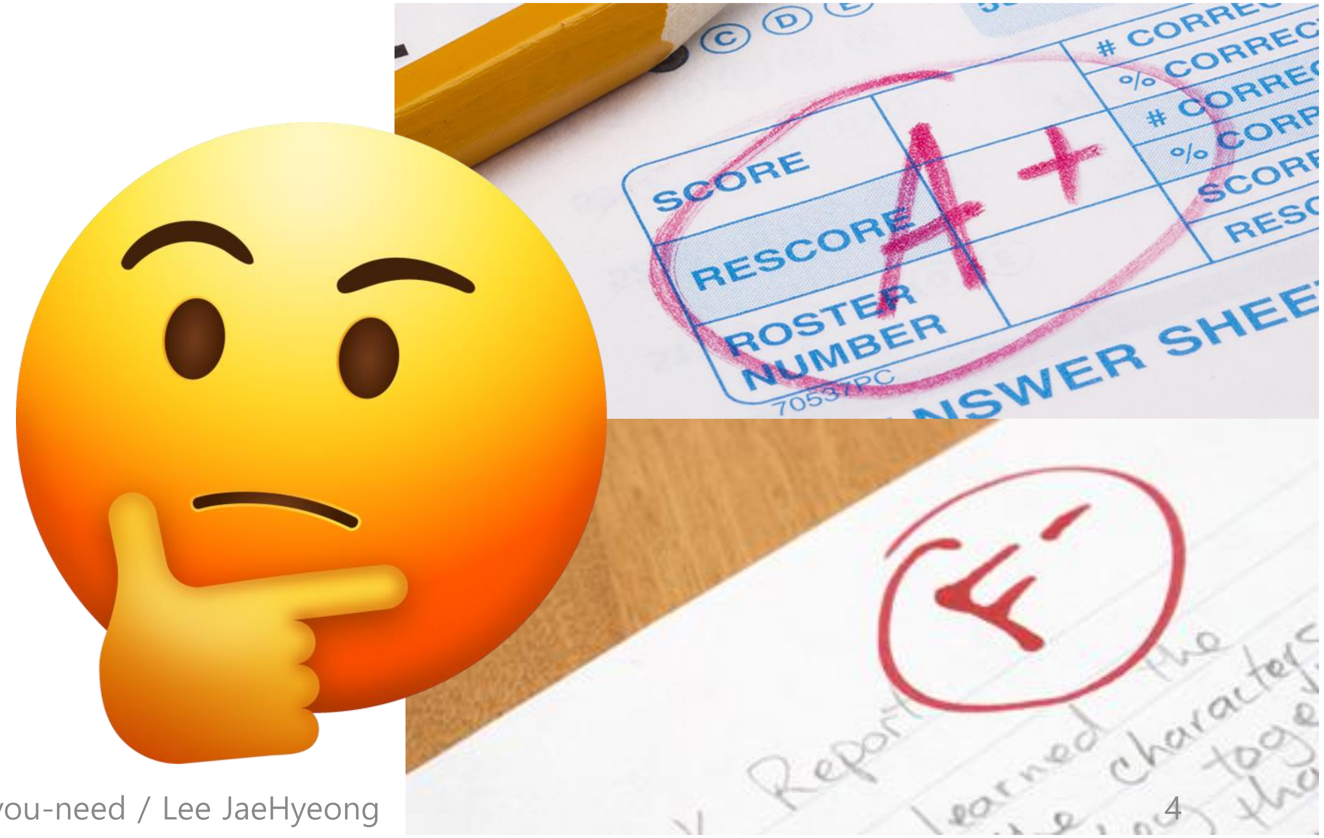


Loss function?

How to say our model is working **good or bad**?
by which **criterion**?

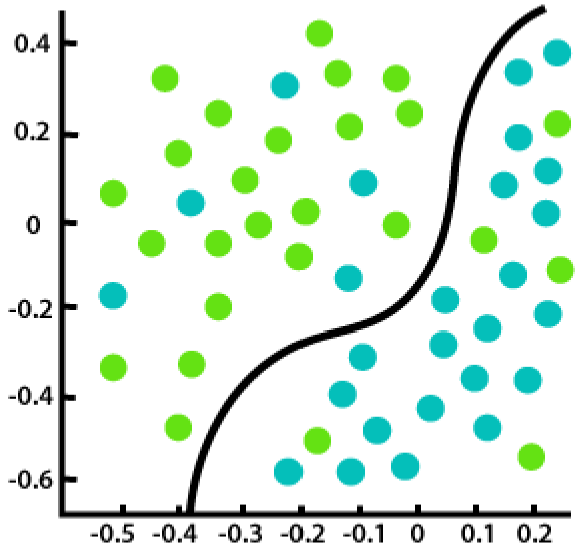
loss function !



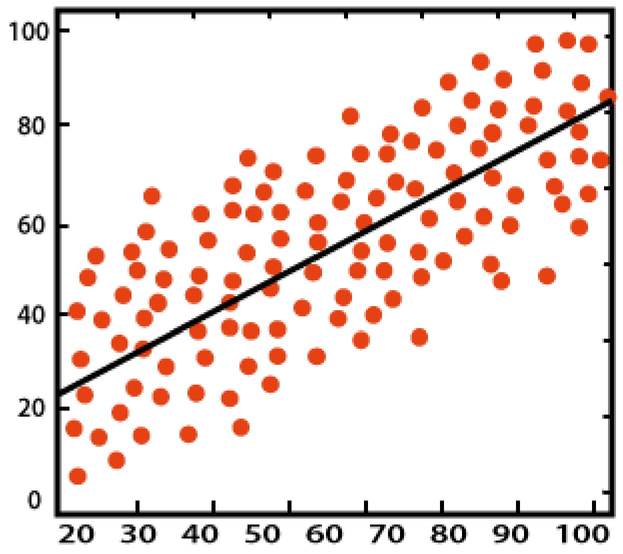


Regression vs Classification

- Classification
  - predict sample's class
- Regression
  - predict continuous value



Classification



Regression

3 Loss Functions for Regression

- MSE (L2 loss)
- MAE (L1 loss)
- Smooth Mean Absolute Error (Huber loss)

Mean Squared Error

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

MSE = mean squared error
n = number of data points
Y_i = observed values
$\hat{Y}_i$ = predicted values

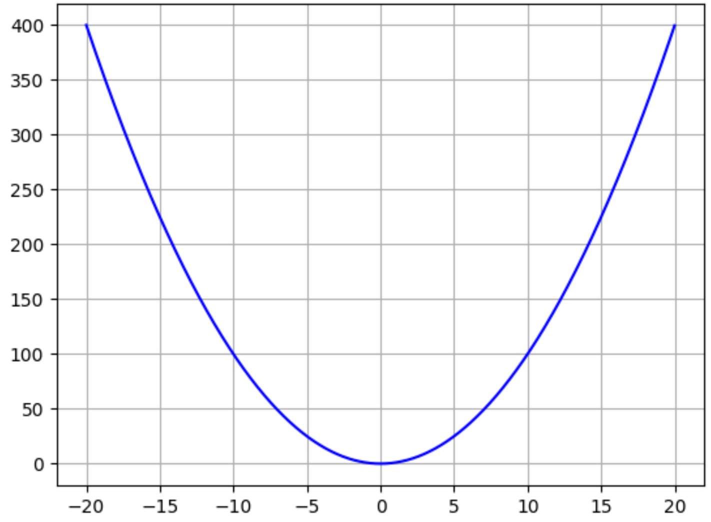
- L2 distance between prediction and ground truth becomes loss score.

Mean Squared Error

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

MSE = mean squared error
n = number of data points
Y_i = observed values
$\hat{Y}_i$ = predicted values

- L2 distance between prediction and ground truth becomes loss score.



loss curve in terms of *"gt – pred"*

Mean Squared Error

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

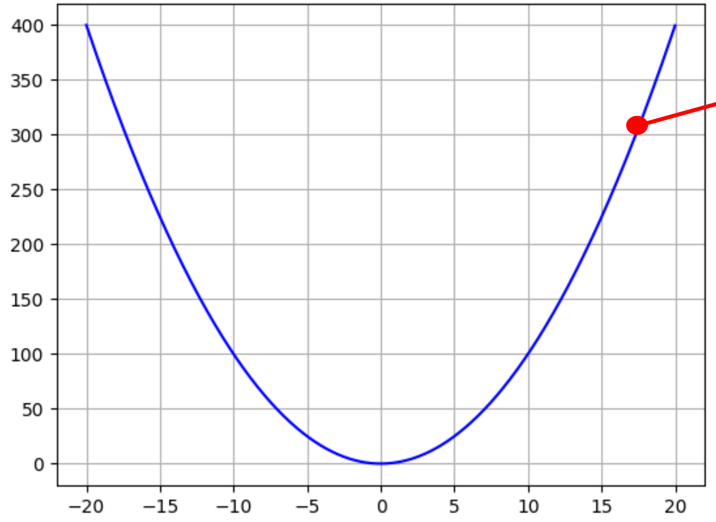
MSE = mean squared error

n = number of data points

Y_i = observed values

$\hat{Y}_i$ = predicted values

- L2 distance between prediction and ground truth becomes loss score.



$$\frac{\partial MSE}{\partial pred_i} = \frac{1}{n}\sum_{i=1}^{n} -1 * 2(y_i - pred_i)$$

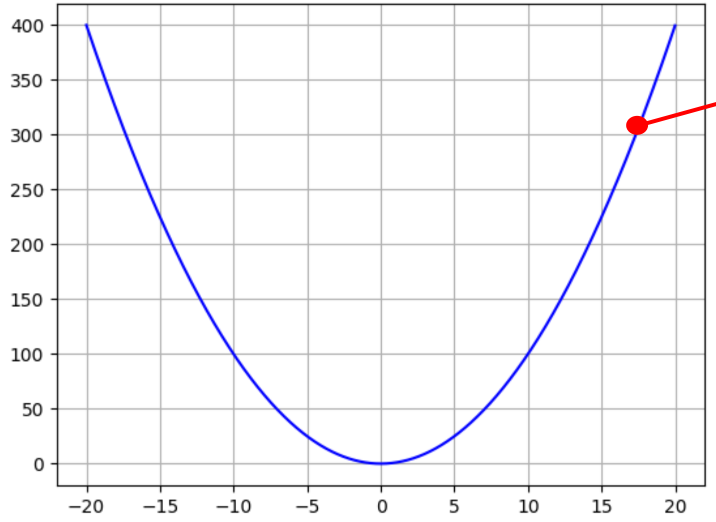
loss curve in terms of *"gt – pred"*

Mean Squared Error

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

MSE = mean squared error
n = number of data points
Y_i = observed values
$\hat{Y}_i$ = predicted values

- L2 distance between prediction and ground truth becomes loss score.



$$\frac{\partial MSE}{\partial pred_i} = \frac{1}{n}\sum_{i=1}^{n} -1 * 2(y_i - pred_i)$$

gradients will be scaled by "$y_i - pred_i$"!
high loss, large optimization steps,
low loss, small optimization steps.

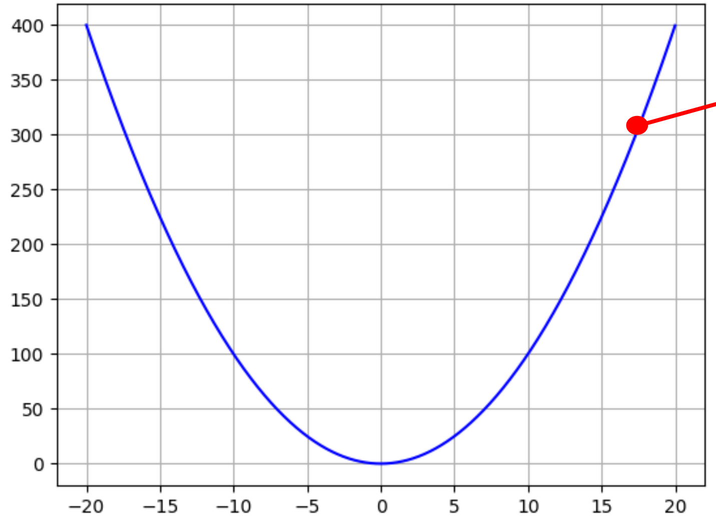
loss curve in terms of *"gt – pred"*

Mean Squared Error

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

MSE = mean squared error
n = number of data points
Y_i = observed values
$\hat{Y}_i$ = predicted values

- L2 distance between prediction and ground truth becomes loss score.



$$\frac{\partial MSE}{\partial pred_i} = \frac{1}{n}\sum_{i=1}^{n} -1 * 2(y_i - pred_i)$$

gradients will be scaled by "$y_i - pred_i$"!
high loss, large optimization steps,
low loss, small optimization steps.
sensitive to outliers!

loss curve in terms of *"gt – pred"*

Mean Absolute Error

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$

MAE = mean absolute error
y_i = prediction
x_i = true value
n = total number of data points

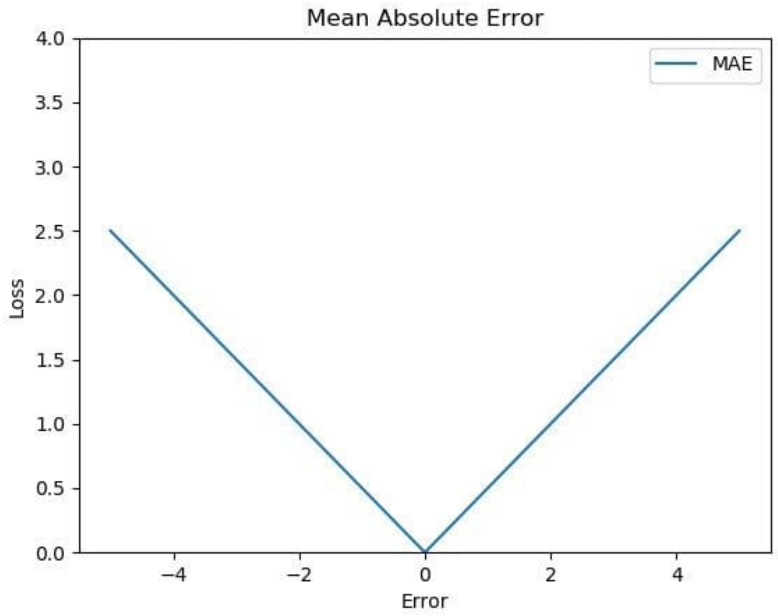
- L1 distance between prediction and ground truth

Mean Absolute Error

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$

MAE = mean absolute error
y_i = prediction
x_i = true value
n = total number of data points

- L1 distance between prediction and ground truth



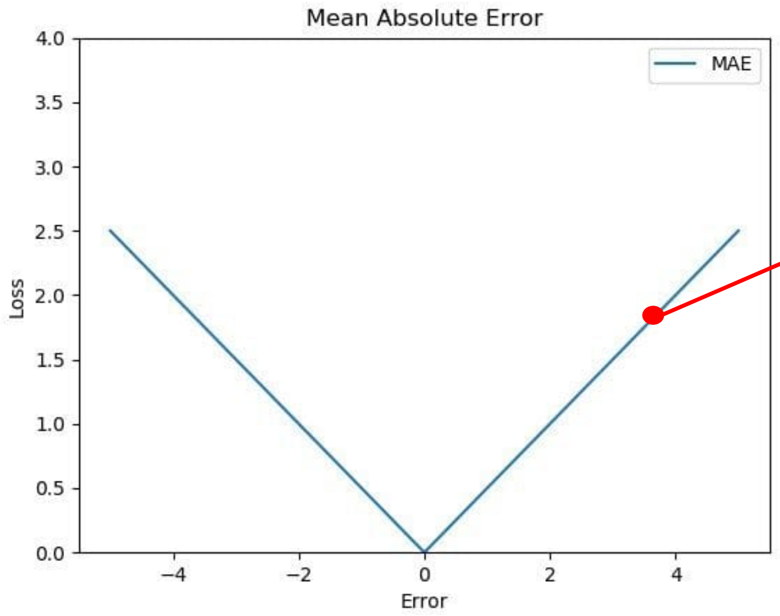
loss curve in terms of *"gt - pred"*

Mean Absolute Error

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$

MAE = mean absolute error
y_i = prediction
x_i = true value
n = total number of data points

- L1 distance between prediction and ground truth



$$\frac{d\mathrm{MAE}}{dy_{\text{pred}}} = \begin{cases} +1, & y_{\text{pred}} > y_{\text{true}} \\ -1, & y_{\text{pred}} < y_{\text{true}} \end{cases}$$

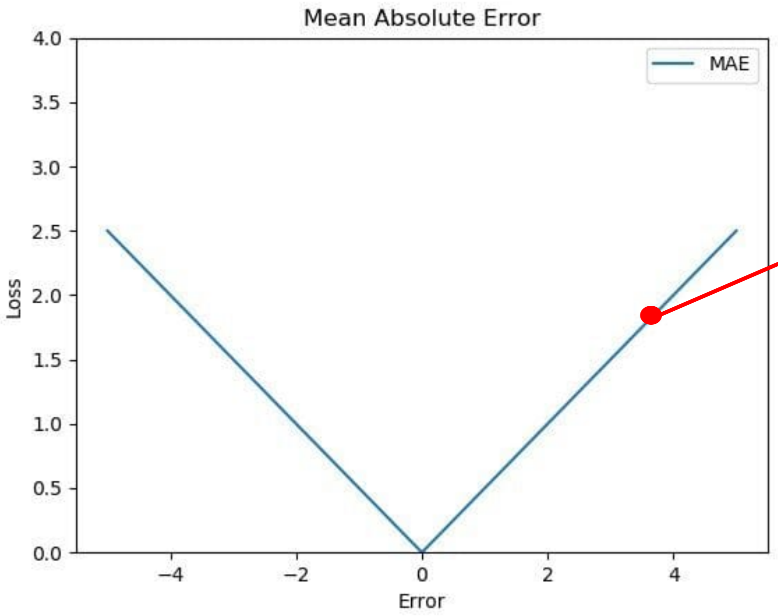
loss curve in terms of *"gt - pred"*

Mean Absolute Error

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$

MAE = mean absolute error
y_i = prediction
x_i = true value
n = total number of data points

- L1 distance between prediction and ground truth



$$\frac{d\mathrm{MAE}}{dy_{\mathrm{pred}}} = \begin{cases} +1, & y_{\mathrm{pred}} > y_{\mathrm{true}} \\ -1, & y_{\mathrm{pred}} < y_{\mathrm{true}} \end{cases}$$

optimization steps aren't related with loss value !

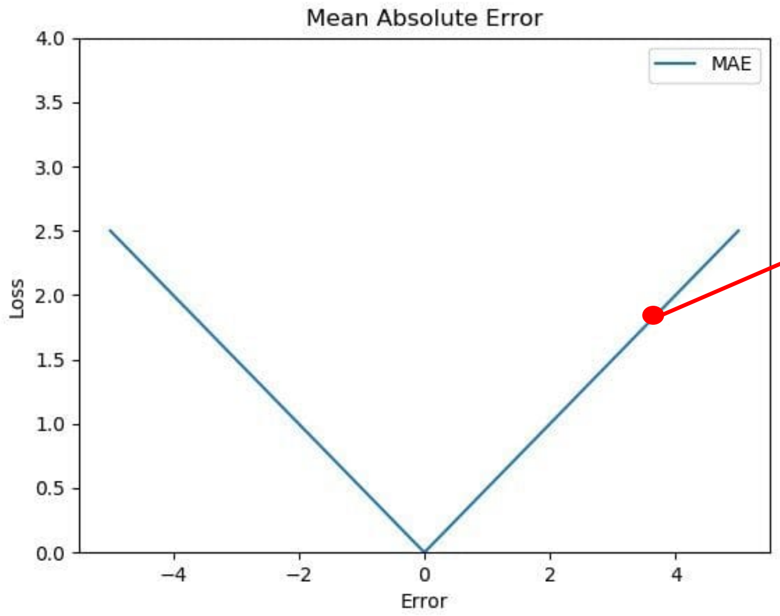
loss curve in terms of *"gt - pred"*

Mean Absolute Error

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$

MAE = mean absolute error
y_i = prediction
x_i = true value
n = total number of data points

- L1 distance between prediction and ground truth



$$\frac{d\mathrm{MAE}}{dy_{\text{pred}}} = \begin{cases} +1, & y_{\text{pred}} > y_{\text{true}} \\ -1, & y_{\text{pred}} < y_{\text{true}} \end{cases}$$

optimization steps aren't related with loss value !
MAE gives same weight for all samples!

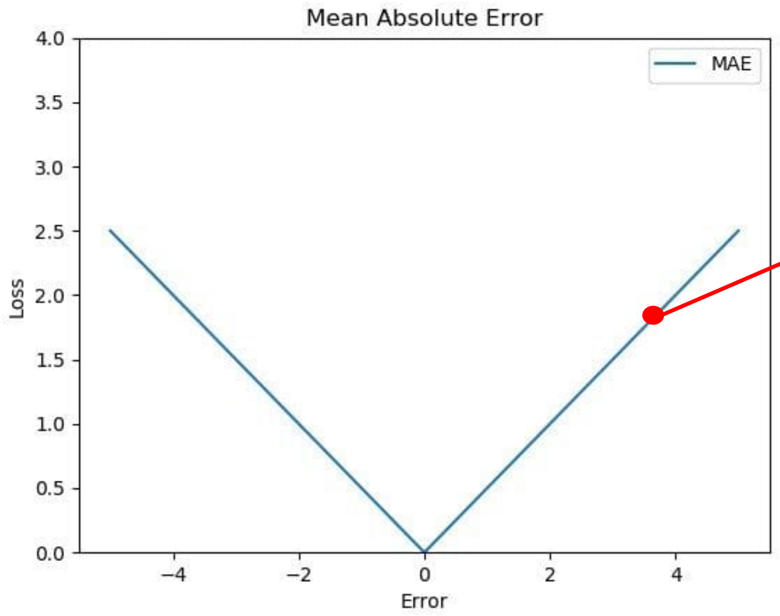
loss curve in terms of *"gt - pred"*

Mean Absolute Error

$$\text{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$

MAE = mean absolute error
y_i = prediction
x_i = true value
n = total number of data points

- L1 distance between prediction and ground truth



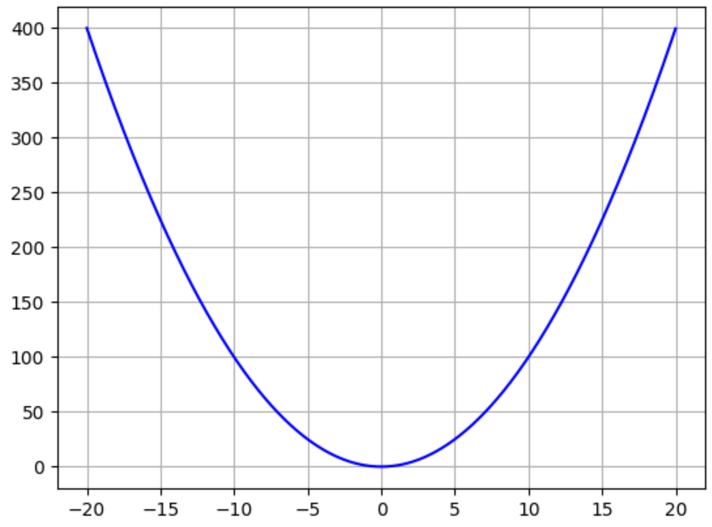
$$\frac{d\text{MAE}}{dy_{\text{pred}}} = \begin{cases} +1, & y_{\text{pred}} > y_{\text{true}} \\ -1, & y_{\text{pred}} < y_{\text{true}} \end{cases}$$

optimization steps aren't related with loss value !
MAE gives same weight for all samples!
-> more robust to outliers

loss curve in terms of *"gt - pred"*

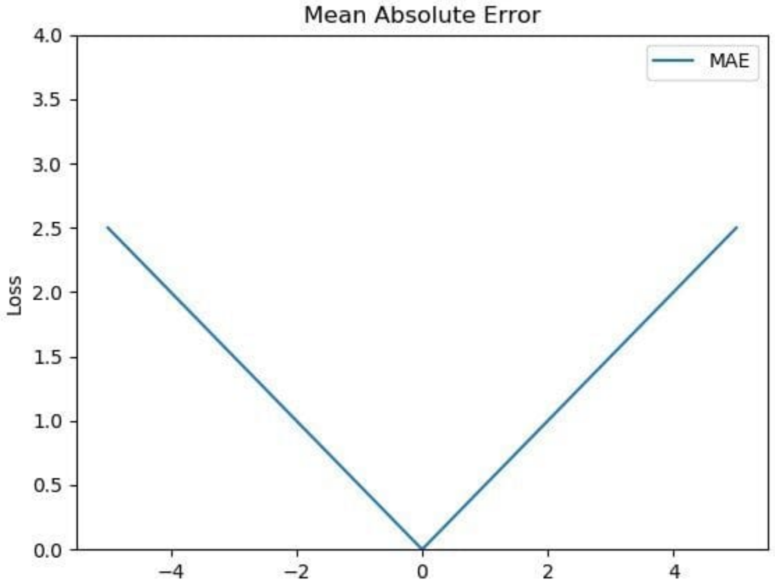
Comparison

MSE Loss



$$\frac{\partial MSE}{\partial pred_i} = \frac{1}{n}\sum_{i=1}^{n} -1 * 2(y_i - pred_i)$$

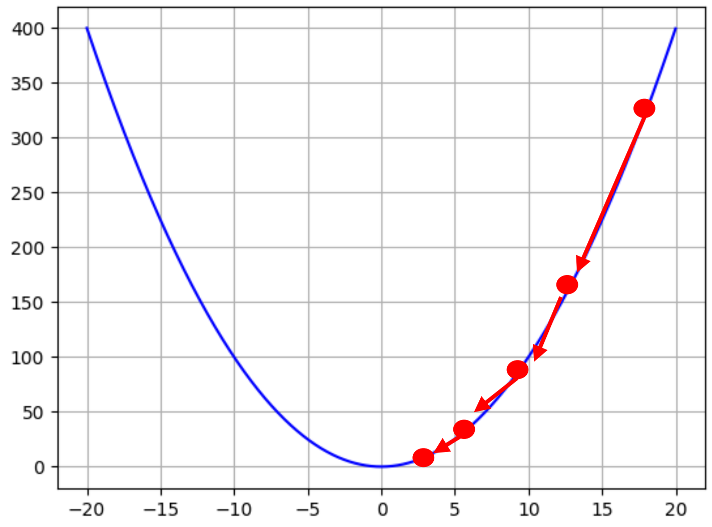
MAE Loss



$$\frac{d\mathrm{MAE}}{dy_{\mathrm{pred}}} = \begin{cases} +1, & y_{\mathrm{pred}} > y_{\mathrm{true}} \\ -1, & y_{\mathrm{pred}} < y_{\mathrm{true}} \end{cases}$$

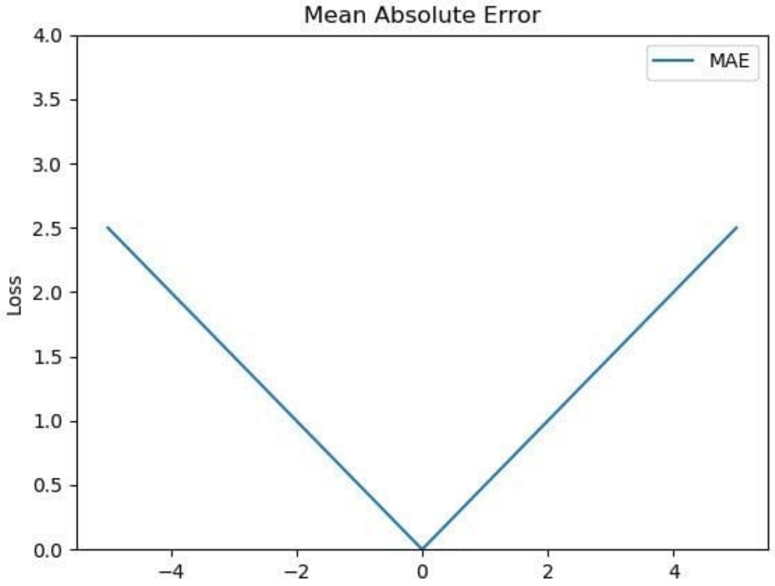
Comparison

MSE Loss



$$\frac{\partial MSE}{\partial pred_i} = \frac{1}{n}\sum_{i=1}^{n} -1 * 2(y_i - pred_i)$$

MAE Loss

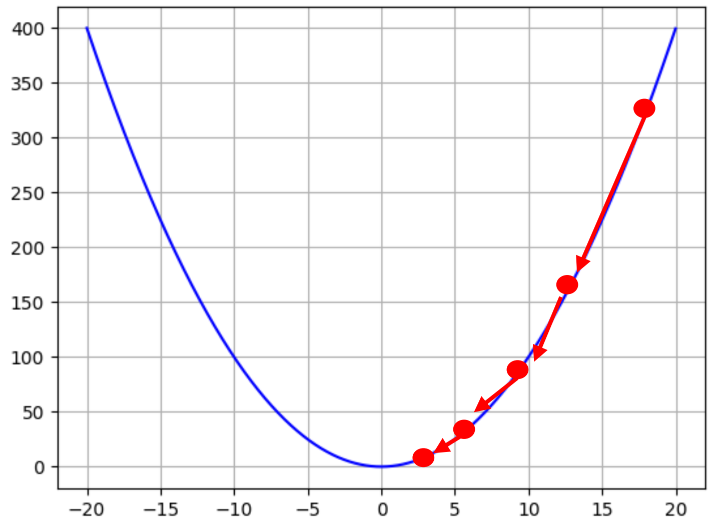


$$\frac{d\mathrm{MAE}}{dy_{\mathrm{pred}}} = \begin{cases} +1, & y_{\mathrm{pred}} > y_{\mathrm{true}} \\ -1, & y_{\mathrm{pred}} < y_{\mathrm{true}} \end{cases}$$

high loss -> gradient scaled -> bigger optim steps

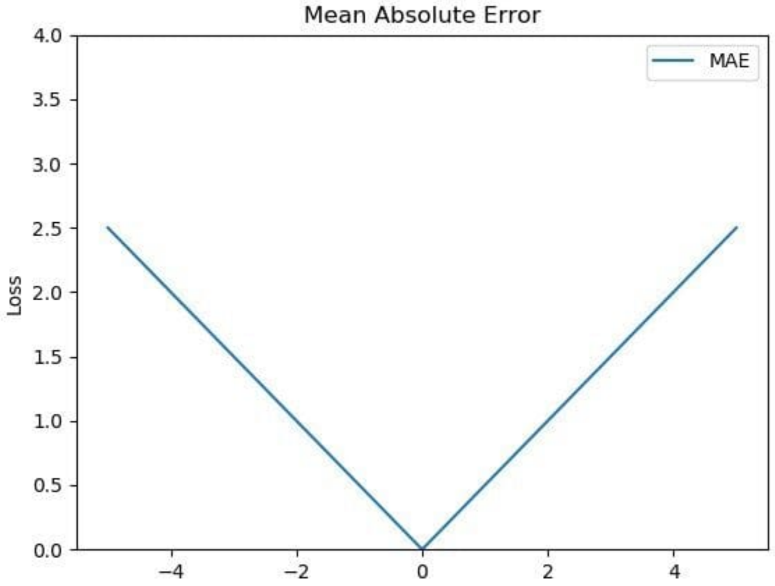
Comparison

MSE Loss



$$\frac{\partial MSE}{\partial pred_i} = \frac{1}{n}\sum_{i=1}^{n} -1 * 2(y_i - pred_i)$$

MAE Loss



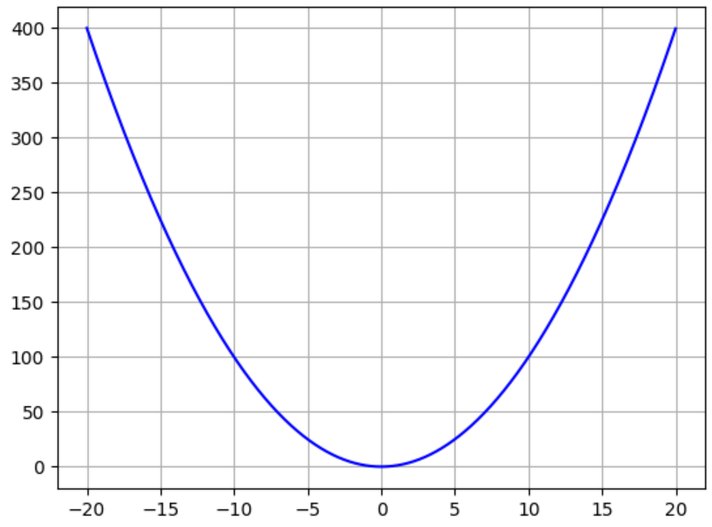
$$\frac{d\text{MAE}}{dy_{\text{pred}}} = \begin{cases} +1, & y_{\text{pred}} > y_{\text{true}} \\ -1, & y_{\text{pred}} < y_{\text{true}} \end{cases}$$

high loss -> gradient scaled -> bigger optim steps

faster convergence / stable optimization when loss is low

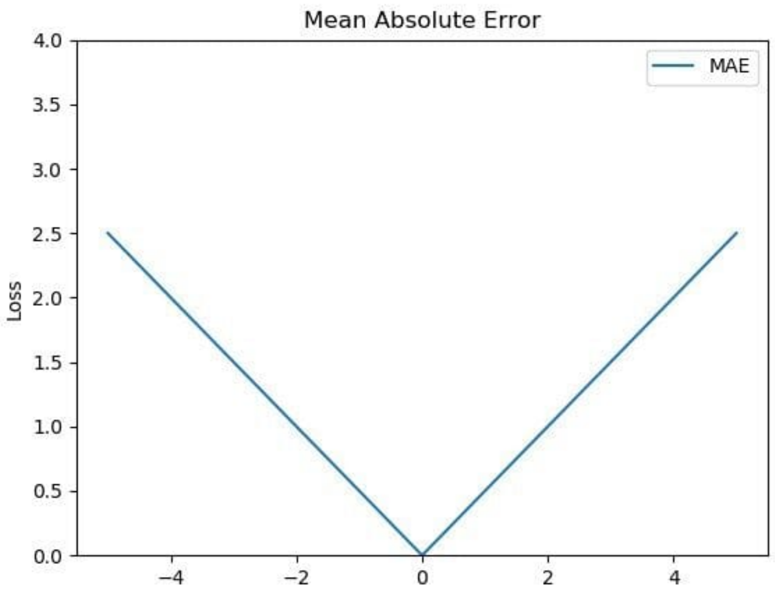
Comparison

MSE Loss



$$\frac{\partial MSE}{\partial pred_i} = \frac{1}{n}\sum_{i=1}^{n} -1 * 2(y_i - pred_i)$$

MAE Loss

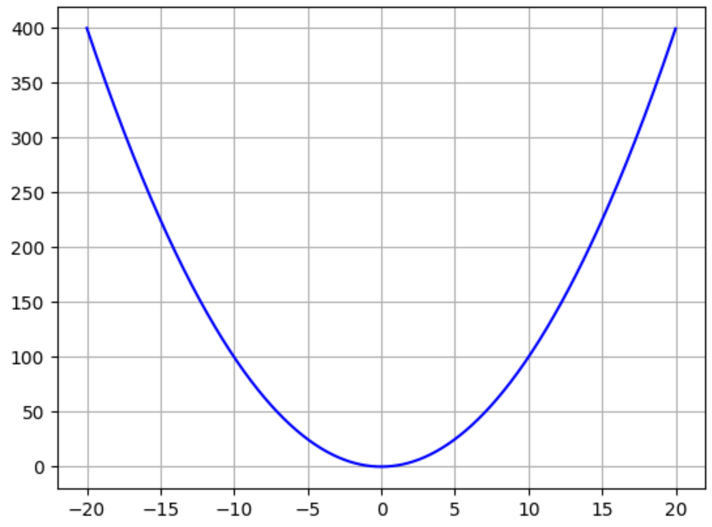


$$\frac{d\mathrm{MAE}}{dy_{\mathrm{pred}}} = \begin{cases} +1, & y_{\mathrm{pred}} > y_{\mathrm{true}} \\ -1, & y_{\mathrm{pred}} < y_{\mathrm{true}} \end{cases}$$

gradient <-> loss value are independent

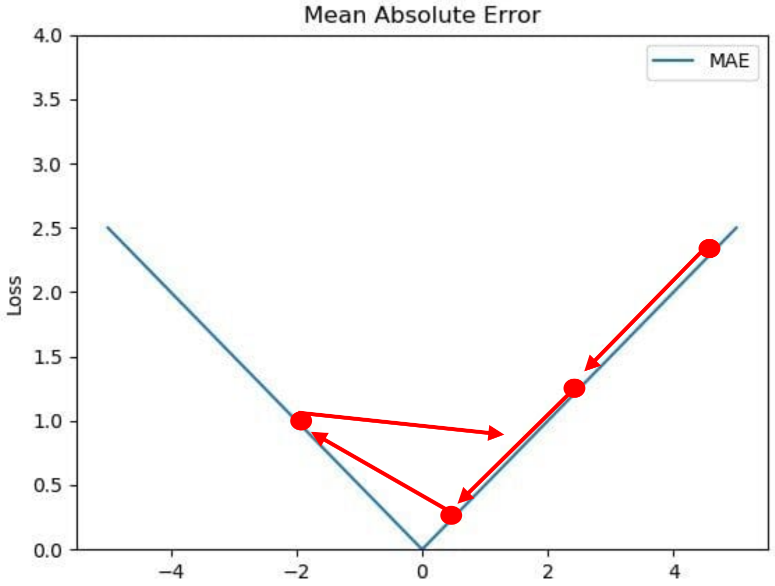
Comparison

MSE Loss



$$\frac{\partial MSE}{\partial pred_i} = \frac{1}{n}\sum_{i=1}^{n} -1 * 2(y_i - pred_i)$$

MAE Loss



$$\frac{d\mathrm{MAE}}{dy_{\mathrm{pred}}} = \begin{cases} +1, & y_{\mathrm{pred}} > y_{\mathrm{true}} \\ -1, & y_{\mathrm{pred}} < y_{\mathrm{true}} \end{cases}$$

gradient <-> loss value are independent

can bounce when loss becomes 0 !

Huber Loss

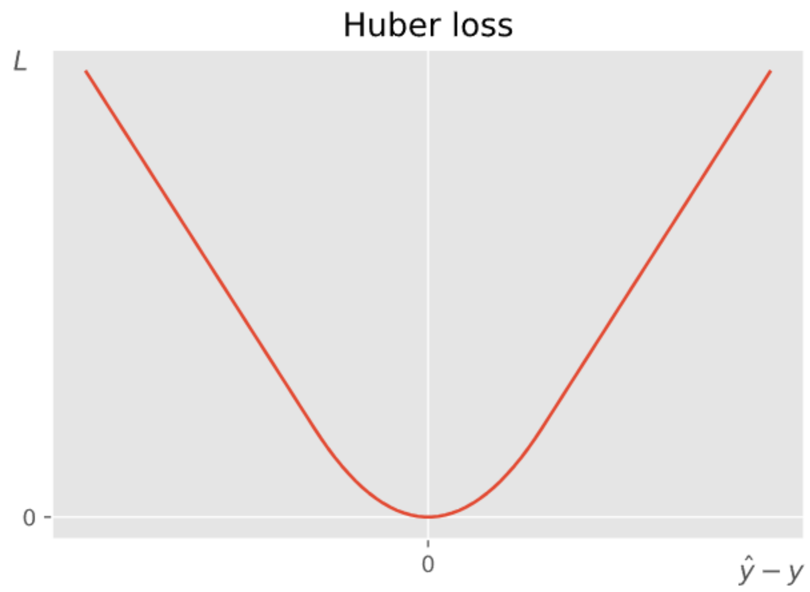
$$L_\delta(y, f(x)) = \begin{cases} \frac{1}{2}(y - f(x))^2 & \text{for} |y - f(x)| \le \delta, \\ \delta\, |y - f(x)| - \frac{1}{2}\delta^2 & \text{otherwise.} \end{cases}$$

- Act like MAE when residuals are larger than delta.
- Act like MSE when residuals are smaller then delta.
- robust to outliers, no bouncing when loss ~ 0 !

Huber Loss

$$L_\delta(y, f(x)) = \begin{cases} \frac{1}{2}(y - f(x))^2 & \text{for} |y - f(x)| \le \delta, \\ \delta \, |y - f(x)| - \frac{1}{2}\delta^2 & \text{otherwise.} \end{cases}$$

- Act like MAE when residuals are larger than delta.
- Act like MSE when residuals are smaller then delta.
- robust to outliers, no bouncing when loss ~ 0 !



Huber Loss

$$L_\delta(y, f(x)) = \begin{cases} \frac{1}{2}(y - f(x))^2 & \text{for} |y - f(x)| \le \delta, \\ \delta\,|y - f(x)| - \frac{1}{2}\delta^2 & \text{otherwise.} \end{cases}$$

- Act like MAE when residuals are larger than delta.
- Act like MSE when residuals are smaller then delta.
- robust to outliers, no bouncing when loss ~ 0 !

have to find hyper-parameter delta by training.

Takeaways

1. When outliers are just corrupted data, and we are focusing on general regression, use MAE.

2. When outliers are important feature that need to be considered, use MSE.

3. MAE can make bounce when loss becomes smaller, use Huber Loss instead. (MAE + MSE)

Thank You !