LOSS FUNCTION

Loss function for different models: (on a data point)

Linear Regression: $\hat{y} = Wx + b$

- L2 loss (distance): $\ell(\theta | x, y) = \frac{1}{2} \| y - \hat{y} \|_2^2$

Binary classification: $\hat{y} = \sigma(Wx + b)$

- Binary cross entropy: $\ell(\theta | x, y) = -y \log(\hat{y}) - (1-y) \log(1-\hat{y})$

Multi-class classification: $\hat{y} = \text{softmax}(Wx + b)$

- Cross entrop loss: $\ell(\theta | x, y) = -\sum_{c=1}^{C} 1_{y=c} \log(\hat{y}_c)$

$\Rightarrow$ basically $-\log(\hat{y}_y)$ ⇢ grondtruth

Loss function of model is expected loss over dataset

$$L(\theta | D) = E_{(x,y)\sim D}\left[\ell(\theta | x, y) \right]$$

Minimizing Loss function (Training model)

Problem statement: Find $\theta^* = \arg\min_{\theta} L(\theta | D)$

Use gradient descent: $\theta' = \theta - \underbrace{\varepsilon}_{\text{learning rate}} \left[\underbrace{\nabla_\theta L(\theta)}_{\text{gradient}} \right]^T$

Initialization:

Usually random, according to Gaussian: $N(\mu, \sigma)$

or Uniform: $\text{Uniform}(a, b)$