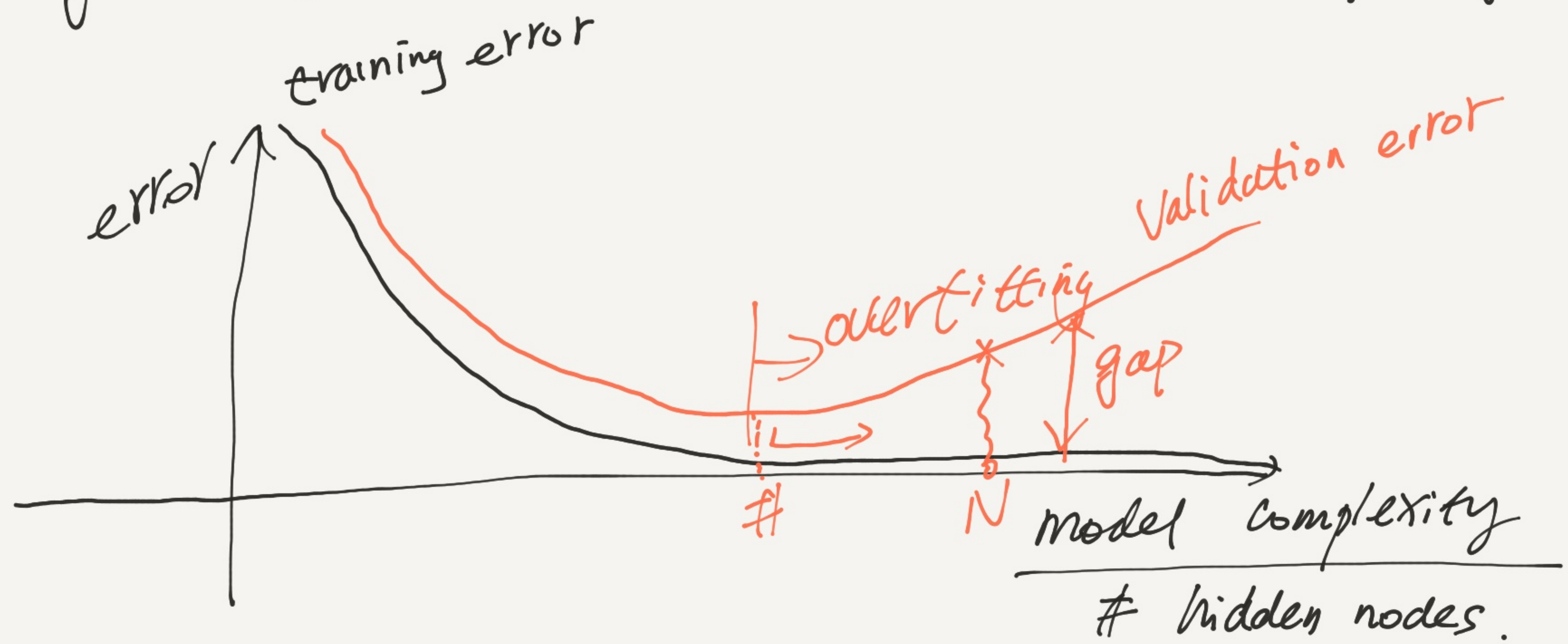
Lecture 16.

Regularization

1. a set strategies used in DL to solve the overfitting issue.



2. three main strategies

(1) Add constraints on model parameters.

$$\min L(w)$$

subject to. $f_i(w) \le 0 \quad i=1,2,\cdots,n$

$h_i(w) = 0 \quad i=1,2,\cdots,p$

Lagrangian method: to convert constrained problem to unconstrained problem:

$$\min L_1(w) = L(w) + \underbrace{\sum_{i=1}^{n} \lambda_i f_i(w) + \sum_{i=1}^{p} v_i \cdot h_i(w)}_{\text{penalty term / regularization term}}$$

DL with regularization:

$$\mathcal{L}(w) = \mathcal{L}_0(w) + \lambda \underbrace{\mathcal{L}_1(w)}_{\text{Regularization term}}$$

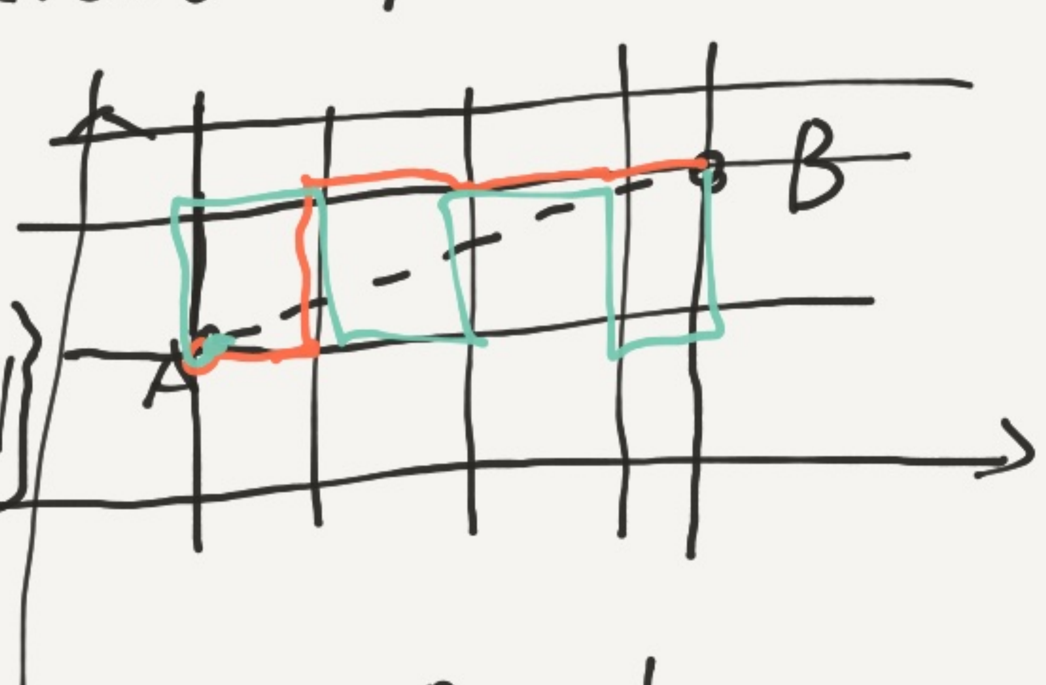
λ: hyper parameter

l_2 norm: $\|w\|_2 = \sqrt{w_1^2 + w_2^2 + \cdots + w_n^2}$

$$\mathcal{L}_1 = \|w\|_2^2 = w_1^2 + w_2^2 + \cdots + w_n^2$$

l_1 norm: $\|w\|_1 = |w_1| + |w_2| + \cdots + |w_n|$

city distance. $|\cdot|$: absolute operation.



l_∞ norm: $\|w\|_\infty = \max\{|w_1|, |w_2|, \cdots, |w_n|\}$

l_p norm: $\|w\|_p = \left(|w_1|^p + |w_2|^p + \cdots + |w_n|^p\right)^{\frac{1}{p}}$

(2) drop out strategy.

(3) Add more data: Data Augmentation → (DL Text book)

$$(x_i, y_i) \longrightarrow \{(x_i^*, y_i)\}$$

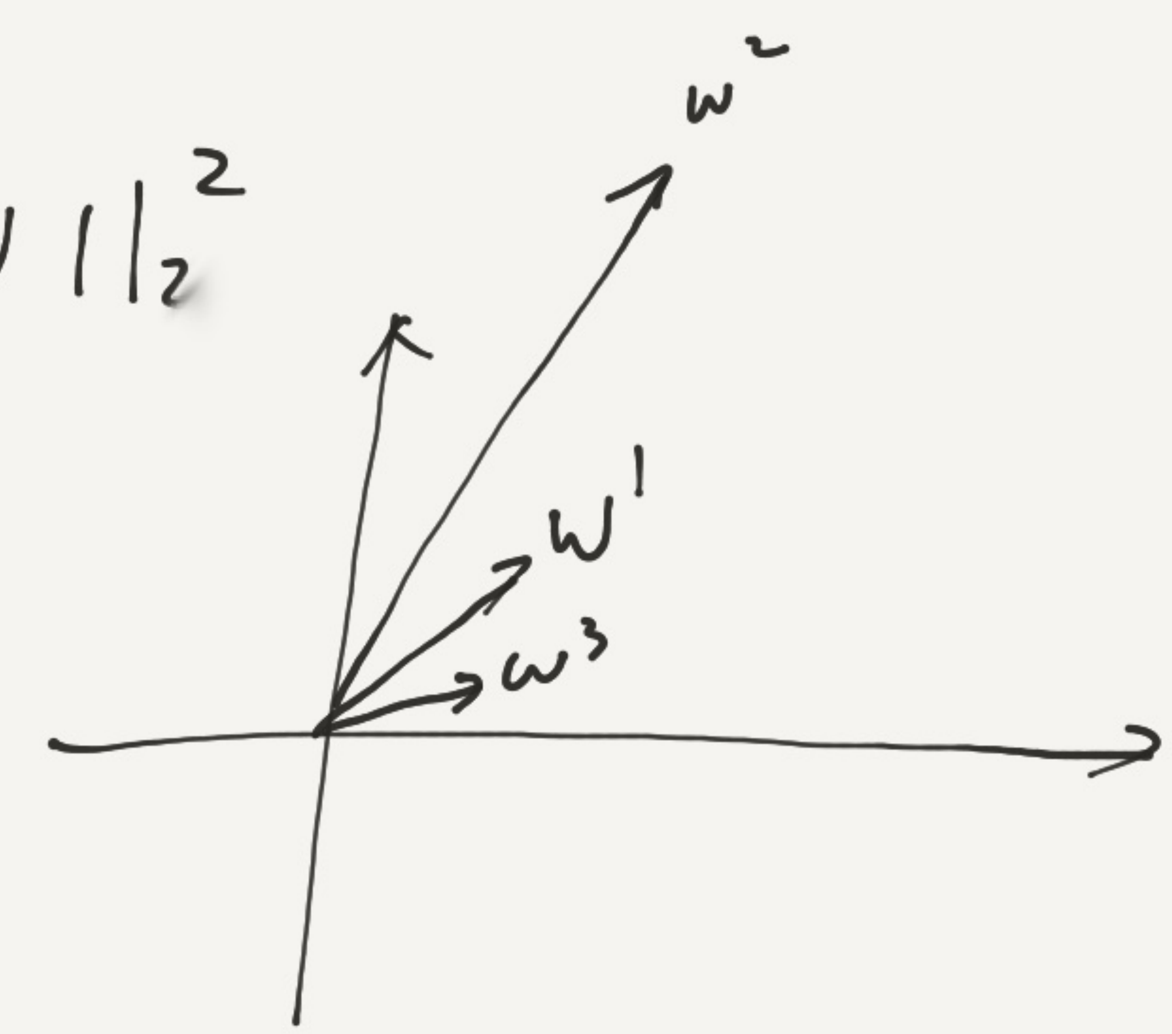
$$\boxed{8} \longrightarrow \boxed{8} \qquad \boxed{6} \xrightarrow{X} \boxed{9}$$

$$\boxed{8} \longrightarrow \boxed{8}$$

$$\left(\tfrac{1}{2}x^2\right)' = x$$

3. l_2 norm regularization

$$\min \mathcal{L}(w) = \underset{CE,\ mse}{\underset{\downarrow}{\mathcal{L}_0(w)}} + \frac{\lambda}{2}\|w\|_2^2$$



Defines preference for small model paramers,

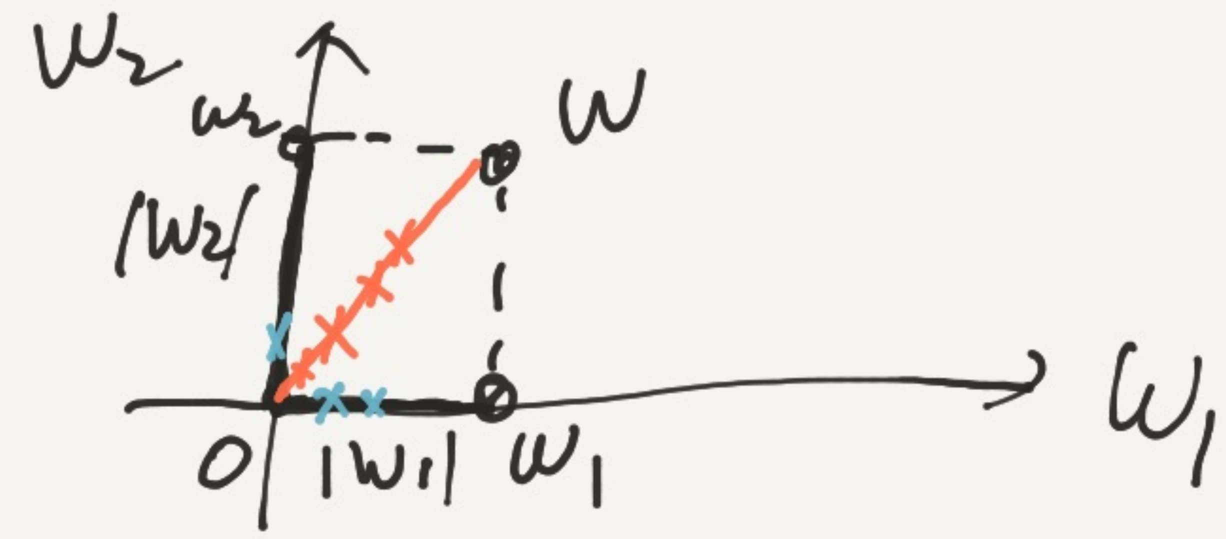
4. ℓ_1 regularization

$$\mathcal{L}(w) = \mathcal{L}_0(w) + \lambda \cdot \|w\|_1$$
$$= \mathcal{L}_0(w) + \lambda\left(|w_1| + |w_2| + \cdots + |w_n|\right)$$

leads to more zero model parameters.

$\downarrow$

auto. feature selection



$$g(w^T x + w_0) = g(w_1 x_1 + w_2 x_2 + \cdots + w_n x_n + w_0)$$

0

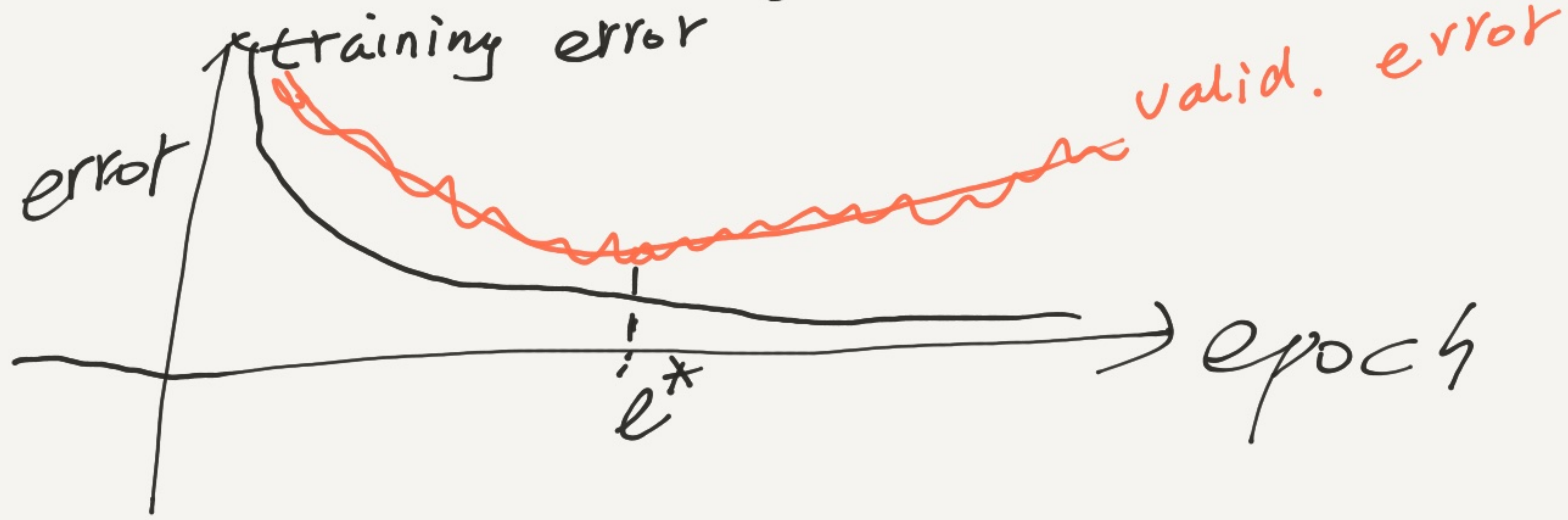
Gradient vanishing : $(0.9)^{100} \simeq 0$

$(1)^{100} = 1$

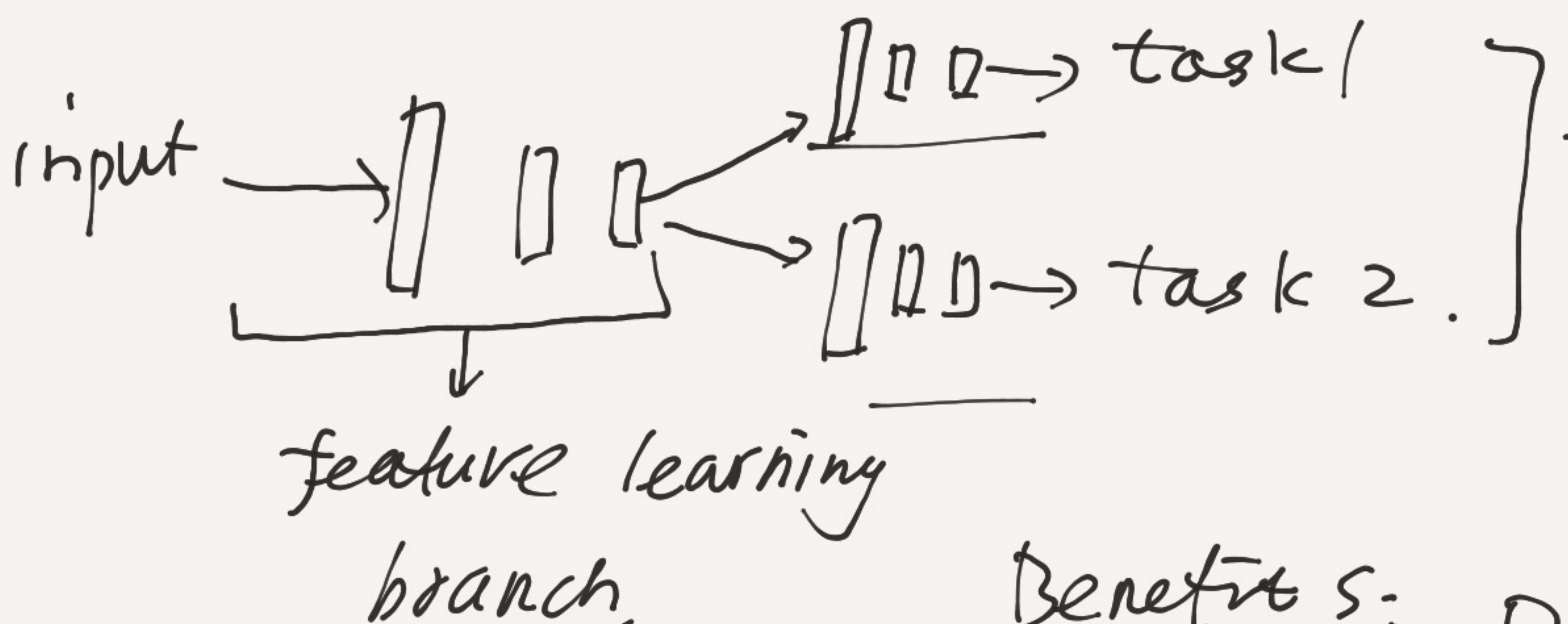
physics informed ML / DL.

Other Regularization strategies.

① Early stopping (training)



② multitask learning (Read shared paper)



two output branches.

$$\mathcal{L} = \mathcal{L}_{task1} + \lambda \mathcal{L}_{task2}$$

Benefits: 1). learn meaningful features for multiple tasks.

2). Needs less training data