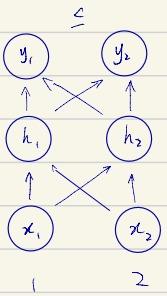
1.1)



$$W^{(1)} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad \vec{b}^{(1)} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad \phi^{(1)} = \begin{cases} z & \text{if } z > 0 \\ -z & \text{if } z \le 0 \end{cases} = |z|$$

$$W^{(2)} = \begin{bmatrix} 0.5 & -0.5 \\ 0.5 & 0.5 \end{bmatrix} \qquad \vec{b}^{(2)} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad \phi^{(2)} = z \quad \text{(linear)}$$

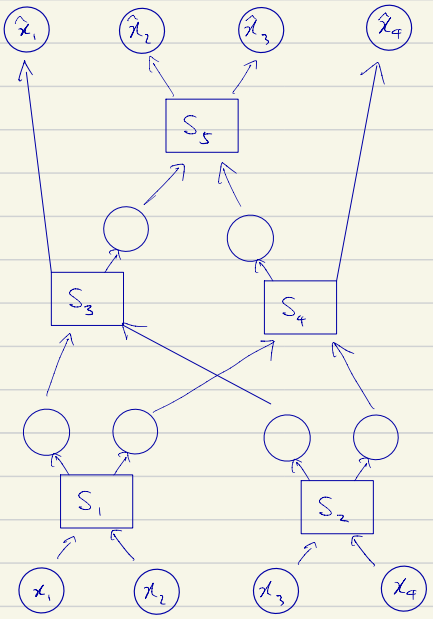
$h_1 = x_1 + x_2$

$h_2 = x_1 - x_2$

$y_1 = 0.5\,h_1 + 0.5\,h_2$

$y_2 = 0.5\,h_1 - 0.5\,h_2$

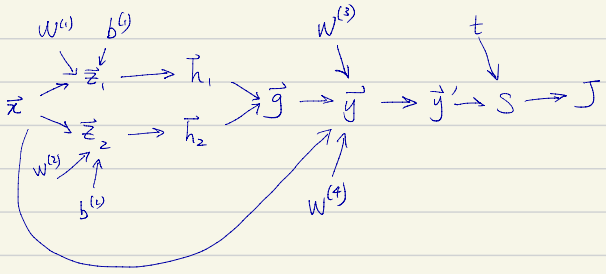
1.2) using merge sort



1.3)

$$W^{(3)} = [0\ 0\ 1\ 0]$$

2.1.1)



2.1.2)

Dm $\bar{J} = 1$

$\bar{S} = \bar{J}\frac{\partial J}{\partial S} = -\bar{J}$

N×1 $\bar{y}' = \bar{S}\frac{\partial S}{\partial y'}^T = \bar{S}\left[\frac{\mathbb{I}(t=1)}{y'_1} \; \frac{\mathbb{I}(t=2)}{y'_2} \; \cdots \; \frac{\mathbb{I}(t=N)}{y'_n}\right]^T = \bar{S}\frac{1}{\vec{y}'^T} \circ \text{one-hot}^T(t) = \bar{S}\frac{1}{\vec{y}'^T} \circ [0\,0\,0\,..\,0\,1\,0\,\cdots\,0]^T$ (where the t^{th} index is 1)

N×1 $\bar{y} = \text{softmax}'(y)\,\bar{y}'$

M×1 $\bar{g} = W^{(3)T}\bar{y}$

M×1 $\bar{h}_1 = \begin{bmatrix} h_{21} & & 0 \\ & h_{22} & \\ 0 & & \ddots \; h_{2m} \end{bmatrix}\begin{bmatrix} \bar{g} \\ 1 \end{bmatrix} = \bar{g} \circ h_2$

M×1 $\bar{h}_2 = \begin{bmatrix} h_{11} & & 0 \\ & h_{12} & \\ 0 & & \ddots \; h_{1m} \end{bmatrix}\begin{bmatrix} \bar{g} \\ 1 \end{bmatrix} = \bar{g} \circ h_1$

M×1 $\bar{z}_1 = \begin{bmatrix} \mathbb{I}(z_{11}>0) & & 0 \\ & \mathbb{I}(z_{12}>0) & \\ 0 & & \ddots \; \mathbb{I}(z_{1m}>0) \end{bmatrix}\begin{bmatrix} \bar{h}_1 \\ 1 \end{bmatrix} = \bar{h}_1 \circ \mathbb{I}(z_1>0)$, where $\mathbb{I}(z_1>0) = \begin{bmatrix} \mathbb{I}(z_{11}>0) \\ \mathbb{I}(z_{12}>0) \\ \vdots \\ \mathbb{I}(z_{1m}>0) \end{bmatrix}$

M×1 $\bar{z}_2 = \begin{bmatrix} \sigma(z_{21})(1-\sigma(z_{21})) & & 0 \\ & \sigma(z_{22})(1-\sigma(z_{22})) & \\ 0 & & \ddots \; \sigma(z_{2m})(1-\sigma(z_{2m})) \end{bmatrix}\begin{bmatrix} \bar{h}_2 \\ 1 \end{bmatrix} = \bar{h}_2 \circ \sigma(z_2) \circ (1-\sigma(z_2))$

since: $\sigma(z) = \frac{1}{1+e^{-z}}$

$\sigma'(z) = \frac{d}{dz}(1+e^{-z})^{-1}$

$= -(1+e^{-z})^{-2}(-e^{-z})$

$= \frac{e^{-z}}{(1+e^{-z})^2}$

$= \frac{1}{1+e^{-z}}\left(1 - \frac{1}{1+e^{-z}}\right)$

$= \sigma(z)(1-\sigma(z))$

D×1 $\bar{x} = W^{(1)T}\bar{z}_1 + W^{(2)T}\bar{z}_2 + W^{(4)T}\bar{y}$

2.2.1) $\boldsymbol{x} = \begin{bmatrix} 1 \\ 3 \\ 1 \end{bmatrix}$ $\bar{\boldsymbol{y}} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$ $\vec{x} \xrightarrow{W^{(1)}} \vec{z} \to \vec{h} \xrightarrow{W^{(2)}} \vec{y} \to$

forward pass:

$$\boldsymbol{z} = \begin{bmatrix} 1 & 2 & 1 \\ -2 & 1 & 0 \\ 1 & -2 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 8 \\ 1 \\ -6 \end{bmatrix}$$

$$\boldsymbol{h} = \text{Relu} \begin{bmatrix} 8 \\ 1 \\ -6 \end{bmatrix} = \begin{bmatrix} 8 \\ 1 \\ 0 \end{bmatrix}$$

backward pass:

$$\bar{\boldsymbol{h}} = W^{(2)^T} \bar{\boldsymbol{y}} = \begin{bmatrix} -2 & 4 & 1 \\ 1 & -2 & -3 \\ -3 & 4 & 6 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ -4 \\ 7 \end{bmatrix}$$

$$\bar{\boldsymbol{z}} = \bar{\boldsymbol{h}} \circ \mathbb{I}(\boldsymbol{z} > 0) = \begin{bmatrix} 3 \\ -4 \\ 7 \end{bmatrix} \circ \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 3 \\ -4 \\ 0 \end{bmatrix}$$

$$\frac{\partial J}{\partial W^{(1)}} = \boldsymbol{x} \bar{\boldsymbol{z}}^T = \begin{bmatrix} 1 \\ 3 \\ 1 \end{bmatrix} \begin{bmatrix} 3 & -4 & 0 \end{bmatrix} = \begin{bmatrix} 3 & -4 & 0 \\ 9 & -12 & 0 \\ 3 & -4 & 0 \end{bmatrix}$$

$$\frac{\partial J}{\partial W^{(2)}} = \boldsymbol{h} \bar{\boldsymbol{y}}^T = \begin{bmatrix} 8 \\ 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 8 & 8 & 8 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\left\| \frac{\partial J}{\partial W^{(1)}} \right\|_F^2 = \left\| \begin{bmatrix} 3 & -4 & 0 \\ 9 & -12 & 0 \\ 3 & -4 & 0 \end{bmatrix} \right\|_F^2 = 9 + 16 + 81 + 144 + 9 + 16$$

$$= \boxed{275}$$

$$\left\| \frac{\partial J}{\partial W^{(2)}} \right\|_F^2 = \left\| \begin{bmatrix} 8 & 8 & 8 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \right\|_F^2 = 64 + 64 + 64 + 1 + 1 + 1 = \boxed{195}$$

2-2-2)

$$\left\| \frac{\partial J}{\partial W^{(1)}} \right\|_F^2 = \text{trace}\left(\frac{\partial J}{\partial W^{(1)}}^T \frac{\partial J}{\partial W^{(1)}} \right)$$

$$= \text{trace}\left(\bar{z} x^T x \bar{z}^T \right)$$

$$= \text{trace}\left(x^T x \bar{z}^T \bar{z} \right) \quad \text{(Cyclic Property of Trace)}$$

$$= (x^T x)(\bar{z}^T \bar{z}) \quad \text{(Scalar Multiplication)}$$

$$= \|x\|_2^2 \, \|\bar{z}\|_2^2$$

$$= \left\| \begin{bmatrix} 1 \\ 3 \\ 1 \end{bmatrix} \right\|_2^2 \left\| \begin{bmatrix} 3 \\ -4 \\ 0 \end{bmatrix} \right\|_2^2$$

$$= (1+9+1)(9+16+0)$$

$$= 11 \cdot 25$$

$$= \boxed{275}$$ same as previous

$$\left\| \frac{\partial J}{\partial W^{(2)}} \right\|_F^2 = \text{trace}\left(\frac{\partial J}{\partial W^{(2)}}^T \frac{\partial J}{\partial W^{(2)}} \right)$$

$$= \text{trace}\left(\bar{y} h^T h \bar{y}^T \right)$$

$$= \text{trace}\left(h^T h \bar{y}^T \bar{y} \right) \quad \text{(Cyclic Property of Trace)}$$

$$= (h^T h)(\bar{y}^T \bar{y}) \quad \text{(Scalar Multiplication)}$$

$$= \|h\|_2^2 \, \|\bar{y}\|_2^2$$

$$= \left\| \begin{bmatrix} 8 \\ 1 \\ 0 \end{bmatrix} \right\|_2^2 \left\| \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right\|_2^2$$

$$= (64+1+0)(1+1+1) = 65 \cdot 3 = \boxed{195}$$ same as previous

2-2-3)

	T (Naive)	T (Efficient)	M (Naive)	M (Efficient)
Forward Pass	D^2NK ①	$D^2N(K-1)$ ①	$O(D^2K+DNK)$	$O(D^2K+DNK)$
Backward Pass	$D^2N(2K-1)$ ②	$D^2N(K-1)$ ③	$O(D^2NK)$	$O(D^2K+DNK)$
Gradient Norm Computation	D^2NK	$NK(2D+1)$	$O(D^2NK)$	$O(DNK)$

model: $h_1 = W_1 x$

$h_2 = W_2 h_1$

$\vdots$

$h_{K-1} = W_{K-1} h_{K-2}$

$y = h_K = W_K h_{K-1}$

general assumptions: $\bar{y}$ is given and we only need to compute what is needed to calculate the gradient norms of all the weights.

note ①: assuming $\bar{y}$ is given, forward pass only needs to compute $h_1, h_2, \cdots h_{K-1}$ since we do not need the output $y = h_K$ for computing the gradient norms of the weights. (hence the $K-1$)

note ②: assuming $\bar{y}$ is given. then we only need to compute $\bar{h}_{K-1}, \bar{h}_{K-2} \cdots \bar{h}_1$ and $\frac{\partial y}{\partial W^{(K)}}, \frac{\partial y}{\partial W^{(K-1)}} \cdots \frac{\partial y}{\partial W^{(2)}}, \frac{\partial y}{\partial W^{(1)}}$ for computing the gradient norms of the weights. (hence the $2K-1$)

note ③: assuming $\bar{y}$ is given, then we only need to compute $\bar{h}_{K-1}, \bar{h}_{K-2} \cdots \bar{h}_1$ for computing the gradient norms of the weights. (hence the $K-1$)

3) $X = n\begin{bmatrix} \\ \\ \\ d \end{bmatrix}$ $\quad t = [\,]n \quad \hat{w} = [\,]d \quad \varepsilon = [\,]n$

3.1) $\mathcal{L} = \frac{1}{n}\|X\hat{w} - t\|_2^2$

$$\frac{d\mathcal{L}}{d\hat{w}} = \frac{2}{n}X^T(X\hat{w} - t) = \frac{2}{n}(X^TX\hat{w} - X^Tt)$$

3.2.1)

set $\frac{d\mathcal{L}}{d\hat{w}} = \frac{2}{n}(X^TX\hat{w} - X^Tt) = 0$

$\Leftrightarrow X^TX\hat{w} - X^Tt = 0$

$\Leftrightarrow X^TX\hat{w} = X^Tt$

$\Leftrightarrow \hat{w} = (X^TX)^{-1}X^Tt$

(X^TX invertible since $n > d$)

3.2.2)

error $= \frac{1}{n}\|X\hat{w} - t\|_2^2$, $\quad \hat{w} = (X^TX)^{-1}X^Tt \quad t = Xw^* + \varepsilon$

$= \frac{1}{n}\|X(X^TX)^{-1}X^Tt - t\|_2^2$

$= \frac{1}{n}\|X(X^TX)^{-1}X^T(Xw^* + \varepsilon) - (Xw^* + \varepsilon)\|_2^2$

$= \frac{1}{n}\|X(X^TX)^{-1}X^TXw^* + X(X^TX)^{-1}X^T\varepsilon - Xw^* - \varepsilon\|_2^2$

$= \frac{1}{n}\|Xw^* + X(X^TX)^{-1}X^T\varepsilon - Xw^* - \varepsilon\|_2^2$

$= \frac{1}{n}\|X(X^TX)^{-1}X^T\varepsilon - \varepsilon\|_2^2$

$= \frac{1}{n}\|(X(X^TX)^{-1}X^T - I)\varepsilon\|_2^2$ as wanted.

expectation of error:

$$\text{Error} = \frac{1}{n}\left\| \left(X(X^TX)^{-1}X^T - I\right)\varepsilon \right\|_2^2$$

$$= \frac{1}{n}\left[\left(X(X^TX)^{-1}X^T - I\right)\varepsilon\right]^T\left(X(X^TX)^{-1}X^T - I\right)\varepsilon$$

where $A = X(X^TX)^{-1}X^T$

$A^T = X(X^TX)^{-1T}X^T$

$$= \frac{1}{n}\left(\left[A - I\right)\varepsilon\right]^T(A - I)\varepsilon\Big)$$

$$= \frac{1}{n}\left(\varepsilon^T(A-I)^T(A-I)\varepsilon\right)$$

$$= \frac{1}{n}\varepsilon^T\left(A^TA - A - A^T + I\right)\varepsilon$$

$$= \frac{1}{n}\varepsilon^T\left(X(X^TX)^{-1T}\cancel{X^TX(X^TX)^{-1}}X^T - X(X^TX)^{-1}X^T - X(X^TX)^{-1T}X^T + I\right)\varepsilon$$

$$= \frac{1}{n}\varepsilon^T\left(X(X^TX)^{-1T}X^T - X(X^TX)^{-1}X^T - X(X^TX)^{-1T}X^T + I\right)\varepsilon$$

$$= \frac{1}{n}\varepsilon^T\left(\cancel{A^T} - A - \cancel{A^T} + I\right)\varepsilon$$

$$= \frac{1}{n}\varepsilon^T(I - A)\varepsilon$$

$$= \frac{1}{n}\left(\varepsilon^T - \varepsilon^TA\right)\varepsilon$$

$$= \frac{1}{n}\left(\varepsilon^T\varepsilon - \varepsilon^TA\varepsilon\right)$$

$$\Rightarrow E(\text{error}) = E\left[\frac{1}{n}\left(\varepsilon^T\varepsilon - \varepsilon^TA\varepsilon\right)\right]$$

$$= \frac{1}{n}E\left[\varepsilon^T\varepsilon - \varepsilon^TA\varepsilon\right]$$

$$= \frac{1}{n}\left(E\left[\varepsilon^T\varepsilon\right] - E\left[\varepsilon^TA\varepsilon\right]\right)$$ by linearity of expectation

$$= \frac{1}{n}\left(n\sigma^2 - \sigma^2 d\right)$$ see next page

$$= \sigma^2 - \frac{\sigma^2 d}{n}$$

$$= \boxed{\sigma^2\left(1 - \frac{d}{n}\right)}$$

$$E[\boldsymbol{\varepsilon}^T\boldsymbol{\varepsilon}]$$

$$= E[\varepsilon_1^2 + \varepsilon_2^2 + \dots + \varepsilon_n^2]$$

$$= E[\varepsilon_1^2] + E[\varepsilon_2^2] + \dots + E[\varepsilon_n^2] \quad \text{by linearity of Expectation}$$

$$= n\sigma^2 \quad \text{since} \quad \text{var}(\varepsilon) = E(\varepsilon^2) - [E(\varepsilon)]^2 \quad \text{and} \quad E(\varepsilon) = 0.$$

$$E[\boldsymbol{\varepsilon}^T\mathbf{A}\boldsymbol{\varepsilon}] = E\left[\begin{bmatrix} \varepsilon_1 & \varepsilon_2 & \cdots & \varepsilon_n \end{bmatrix} \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & & & \\ \vdots & & \ddots & \\ A_{n1} & & & A_{nn} \end{bmatrix} \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix} \right]$$

$$= E\left[\begin{bmatrix} \sum_{i=1}^n \varepsilon_i A_{i1} & \sum_{i=1}^n \varepsilon_i A_{i2} & \cdots & \sum_{i=1}^n \varepsilon_i A_{in} \end{bmatrix} \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix} \right]$$

$$= E\left[\sum_{j=1}^n \sum_{i=1}^n \varepsilon_i \varepsilon_j A_{ij} \right]$$

$$= E\left[\sum_{i=1}^n A_{ii}\varepsilon_i^2 + \sum_{\substack{i,j \\ \text{where } i \neq j}}^n A_{ij}\varepsilon_i\varepsilon_j \right]$$

$$= \sum_{i=1}^n E[A_{ii}\varepsilon_i^2] + \sum_{\substack{i,j \\ \text{where } i \neq j}}^n E[A_{ij}\varepsilon_i\varepsilon_j] \quad \text{by linearity of expectation}$$

$$= \sum_{i=1}^n A_{ii} E[\varepsilon_i^2] + \sum_{\substack{i,j \\ \text{where } i \neq j}}^n A_{ij} E[\varepsilon_i] E[\varepsilon_j] \quad \text{since } \varepsilon_i \text{ and } \varepsilon_j \text{ is independent when } i \neq j.$$

$$= \sum_{i=1}^n A_{ii}\sigma^2 + \sum_{\substack{i,j \\ \text{where } i \neq j}}^n A_{ij}(0)(0)$$

$$= \sigma^2 \sum_{i=1}^n A_{ii}$$

$$= \sigma^2 \, tr(\mathbf{A}) \quad , \text{ recall } \mathbf{A} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T$$

$$= \sigma^2 \, tr(\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T)$$

$$= \sigma^2 \, tr((\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{X}) \quad \text{by cyclic property of trace.}$$

$$= \sigma^2 \, tr(\mathbf{I}) \quad , \mathbf{I} \text{ is dimension } d \times d$$

$$= \sigma^2 d$$

3.3.2)

$\hat{w} = X^T a$ for some $a \in \mathbb{R}^n$

substitute this into

$$\frac{d\mathcal{L}}{d\hat{w}} = \frac{2}{n} X^T (X\hat{w} - t) = 0$$

$$\Rightarrow \frac{2}{n} X^T (X X^T a - t) = 0$$

$$\Rightarrow X^T (X X^T a - t) = 0$$

$$\Rightarrow X X^T (X X^T a - t) = X \cdot 0$$

$$\Rightarrow (X X^T a - t) = (X X^T)^{-1} X \cdot 0, \quad X X^T \text{ is invertible since } d > n.$$

$$\Rightarrow X X^T a - t = 0$$

$$\Rightarrow X X^T a = t$$

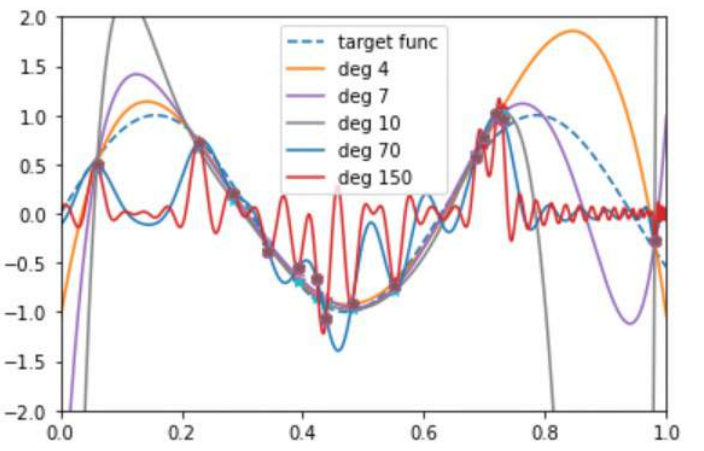
$$\Rightarrow a = (X X^T)^{-1} t$$

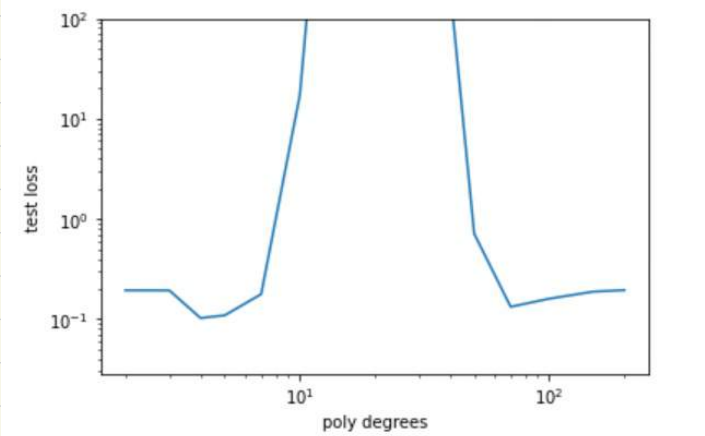
$$\Rightarrow \hat{w} = X^T (X X^T)^{-1} t \quad , \text{ which is unique.}$$

3.3.4)

```
# to be implemented; fill in the derived solution for the underparameterized (d<n) and overparameterized (d>n) problem

def fit_poly(X, d,t):
  X_expand = poly_expand(X, d=d, poly_type = poly_type)
  n = X.shape[0]
  if d > n:
    W = (X_expand.T @ np.linalg.inv(X_expand @ X_expand.T)) @ t
  else:
    W = (np.linalg.inv(X_expand.T @ X_expand) @ X_expand.T) @ t
  return W
```





conclusion:

overparametrization does not always lead to overfitting. From the second graph above, we see that test loss for polynomial degree 10^2 is about the same as test loss for degree ≈ 4, and much less than test loss for degrees between 10 – 50.