

COMP5212: Machine Learning

Lecture 10

Minhao CHENG

VC dimension

Definition

- The VC dimension of a hypothesis set \mathcal{H} , denoted by $d_{VC}(\mathcal{H})$, is the largest value of N for which $m_{\mathcal{H}}(N) = 2^N$
 - “The most points \mathcal{H} can shatter”
- $N \leq d_{VC}(\mathcal{H}) \Rightarrow \mathcal{H}$ can shatter N points
- $k > d_{VC}(\mathcal{H}) \Rightarrow \mathcal{H}$ cannot be shattered
- The smallest **break point** is 1 above VC-dimension

VC dimension

The growth function

- In terms of a break point k :

$$\bullet \quad m_{\mathcal{H}}(N) \leq \sum_{i=0}^{k-1} \binom{N}{i}$$

- In terms of the VC dimension d_{VC} :

$$\bullet \quad m_{\mathcal{H}}(N) \leq \sum_{i=0}^{d_{\text{VC}}} \binom{N}{i}$$

VC dimension

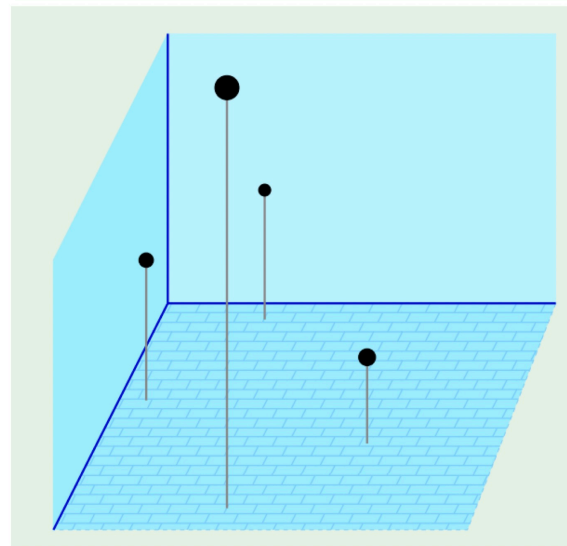
VC dimension of linear classifier

- For $d = 2$, $d_{VC} = 3$

VC dimension

VC dimension of linear classifier

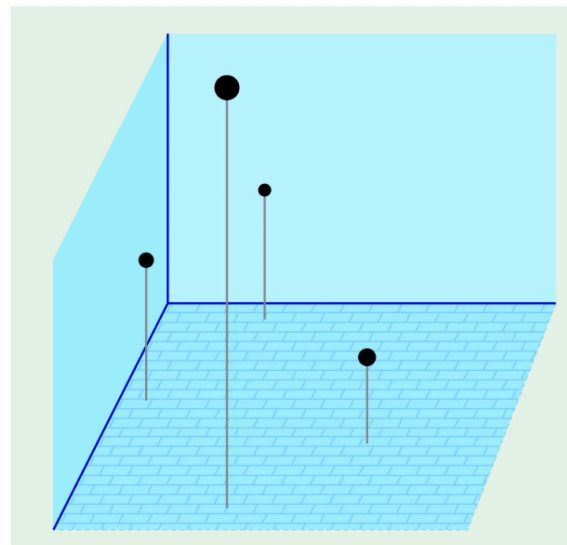
- For $d = 2$, $d_{VC} = 3$
- What if $d > 2$?



VC dimension

VC dimension of linear classifier

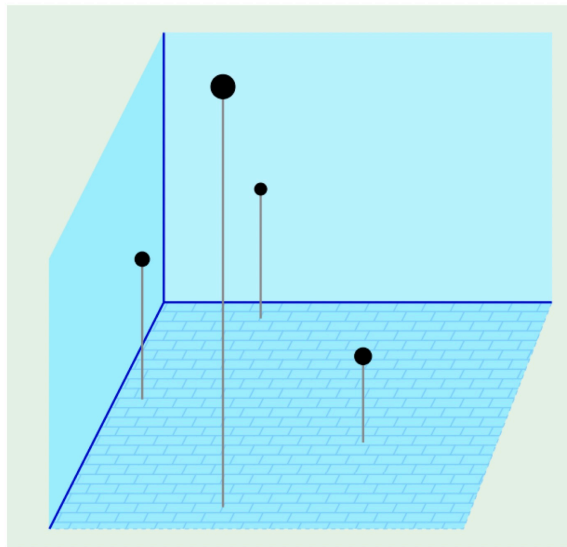
- For $d = 2$, $d_{VC} = 3$
- What if $d > 2$?
- In general,
 - $d_{VC} = d + 1$



VC dimension

VC dimension of linear classifier

- For $d = 2$, $d_{VC} = 3$
- What if $d > 2$?
- In general,
 - $d_{VC} = d + 1$
- We will prove $d_{VC} \geq d + 1$ and $d_{VC} \leq d + 1$



VC dimension

VC dimension of linear classifier

- A set of $N = d + 1$ points in \mathbb{R}^d shattered by the linear hyperplane

$$X = \begin{bmatrix} -\mathbf{x}_1^\top - \\ -\mathbf{x}_2^\top - \\ -\mathbf{x}_3^\top - \\ \vdots \\ -\mathbf{x}_{d+1}^\top - \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 1 & 0 & 1 & & 0 \\ & \vdots & & \ddots & 0 \\ 1 & 0 & \dots & 0 & 1 \end{bmatrix}$$

VC dimension

VC dimension of linear classifier

- A set of $N = d + 1$ points in \mathbb{R}^d shattered by the linear hyperplane

$$X = \begin{bmatrix} -\mathbf{x}_1^\top - \\ -\mathbf{x}_2^\top - \\ -\mathbf{x}_3^\top - \\ \vdots \\ -\mathbf{x}_{d+1}^\top - \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 1 & 0 & 1 & & 0 \\ & \vdots & & \ddots & 0 \\ 1 & 0 & \dots & 0 & 1 \end{bmatrix}$$

- X is invertible!

VC dimension

Can we shatter the dataset?

- For any $y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{d+1} \end{bmatrix} = \begin{bmatrix} \pm 1 \\ \pm 1 \\ \vdots \\ \pm 1 \end{bmatrix}$, can be find w satisfying
 - $\text{sign}(Xw) = y$

VC dimension

Can we shatter the dataset?

- For any $y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{d+1} \end{bmatrix} = \begin{bmatrix} \pm 1 \\ \pm 1 \\ \vdots \\ \pm 1 \end{bmatrix}$, can be find w satisfying
 - $\text{sign}(Xw) = y$
 - Easy! Just set $w = X^{-1}y$

VC dimension

Can we shatter the dataset?

- For any $y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{d+1} \end{bmatrix} = \begin{bmatrix} \pm 1 \\ \pm 1 \\ \vdots \\ \pm 1 \end{bmatrix}$, can be find w satisfying
 - $\text{sign}(Xw) = y$
 - Easy! Just set $w = X^{-1}y$

VC dimension

Can we shatter the dataset?

- This implies what?
 - [a] $d_{VC} = d + 1$
 - [b] $d_{VC} \leq d + 1$
 - [c] $d_{VC} \geq d + 1$
 - [d] No conclusion

VC dimension

Can we shatter the dataset?

- This implies what?

- [a] $d_{VC} = d + 1$

- [b] $d_{VC} \leq d + 1$

- [c] $d_{VC} \geq d + 1$

- [d] No conclusion

VC dimension

Can we shatter the dataset?

- To show $d_{VC} \leq d + 1$
 - [a] There are $d + 1$ points we cannot shatter
 - [b] There are $d + 2$ points we cannot shatter
 - [c] We cannot shatter any set of $d + 1$ points
 - [d] We cannot shatter any set of $d + 2$ points

VC dimension

VC dimension of linear classifier

- To show $d_{VC} \leq d + 1$, we need to show
 - We cannot shatter any set of $d + 2$ points

VC dimension

VC dimension of linear classifier

- To show $d_{VC} \leq d + 1$, we need to show
 - We cannot shatter any set of $d + 2$ points
- For any $d + 2$ points
 - $x_1, x_2, \dots, x_{d+1}, x_{d+2}$
- More points than dimensions \Rightarrow linear dependent

- $$x_j = \sum_{i \neq j} a_i x_i$$

- Where not all a_i 's are zeros

VC dimension

VC dimension of linear classifier

- $x_j = \sum_{i \neq j} a_i x_i$
- Now we construct a dichotomy that cannot be generated:
 - $y_i = \begin{cases} \text{sign}(a_i) & \text{if } i \neq j \\ -1 & \text{if } i = j \end{cases}$

VC dimension

VC dimension of linear classifier

- $x_j = \sum_{i \neq j} a_i x_i$
- Now we construct a dichotomy that cannot be generated:
 - $y_i = \begin{cases} \text{sign}(a_i) & \text{if } i \neq j \\ -1 & \text{if } i = j \end{cases}$
- For all $i \neq j$, assume the labels are correct:
 $\text{sign}(a_i) = \text{sign}(w^T x_i) \Rightarrow a_i w^T x_i > 0$

VC dimension

VC dimension of linear classifier

- $x_j = \sum_{i \neq j} a_i x_i$

- Now we construct a dichotomy that cannot be generated:

- $y_i = \begin{cases} \text{sign}(a_i) & \text{if } i \neq j \\ -1 & \text{if } i = j \end{cases}$

- For all $i \neq j$, assume the labels are correct: $\text{sign}(a_i) = \text{sign}(w^T x_i) \Rightarrow a_i w^T x_i > 0$
- Therefore, $y_j = \text{sign}(w^T x_j) = +1$ (cannot be -1)

VC dimension

Putting it together

- We proved for d -dimensional linear hyperplane
 - $d_{VC} \geq d + 1$ and $d_{VC} \leq d + 1 \Rightarrow d_{VC} = d + 1$
- Number of parameters w_0, \dots, w_d
 - $d + 1$ parameters!

VC dimension

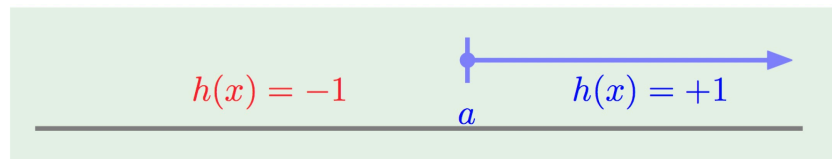
Putting it together

- We proved for d -dimensional linear hyperplane
 - $d_{VC} \geq d + 1$ and $d_{VC} \leq d + 1 \Rightarrow d_{VC} = d + 1$
- Number of parameters w_0, \dots, w_d
 - $d + 1$ parameters!
- Parameters create degrees of freedom

VC dimension

Examples

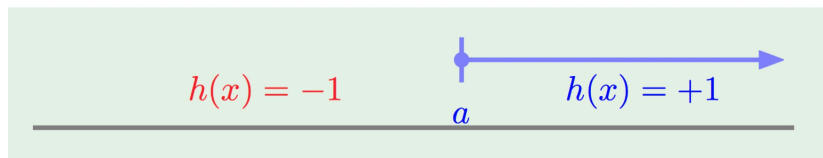
- Positive rays: 1 parameters, $d_{VC} = 1$



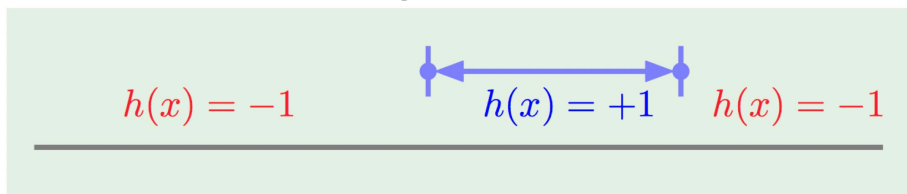
VC dimension

Examples

- Positive rays: 1 parameters, $d_{VC} = 1$



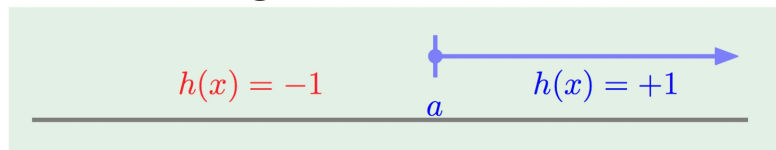
- Positive intervals: 2 parameters, $d_{VC} = 2$



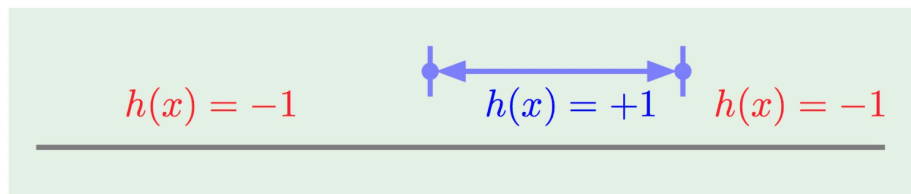
VC dimension

Examples

- Positive rays: 1 parameters, $d_{VC} = 1$



- Positive intervals: 2 parameters, $d_{VC} = 2$



- Not always true ...
 - d_{VC} measures the **effective** number of parameters

VC dimension

Number of data points needed

- $\mathbb{P}[|E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon] \leq \underbrace{4m_{\mathcal{H}}(2N)}_{\delta} e^{-\frac{1}{8}\epsilon^2 N}$
- If we want certain ϵ and δ , how does N depend on d_{VC} ?

VC dimension

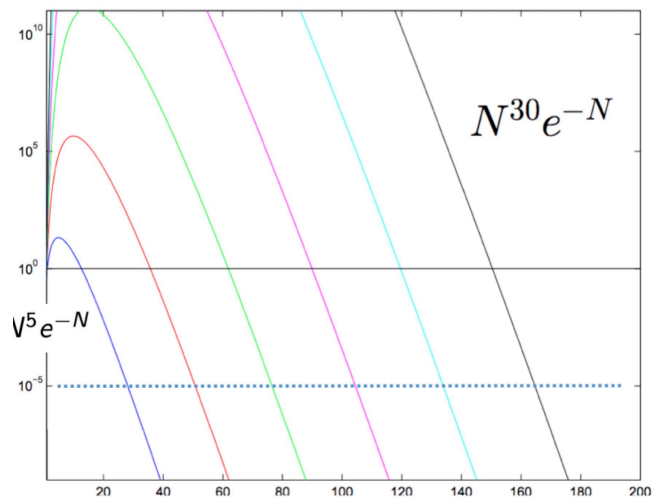
Number of data points needed

- $$\mathbb{P}[|E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon] \leq \underbrace{4m_{\mathcal{H}}(2N)}_{\delta} e^{-\frac{1}{8}\epsilon^2 N}$$
- If we want certain ϵ and δ , how does N depend on d_{VC} ?
- Need $N^{\epsilon} e^{-N} = \text{small value}$

VC dimension

Number of data points needed

- $\mathbb{P}[|E_{\text{in}}(g) - E_{\text{out}}(g)| > \epsilon] \leq \underbrace{4m_{\mathcal{H}}(2N)}_{\delta} e^{-\frac{1}{8}\epsilon^2 N}$
- If we want certain ϵ and δ , how does N depend on d_{VC} ?
- Need $N^{d_{\text{VC}}} e^{-N} = \text{small value}$



N is almost linear with d_{VC}

Regularization

Regularization

The polynomial model

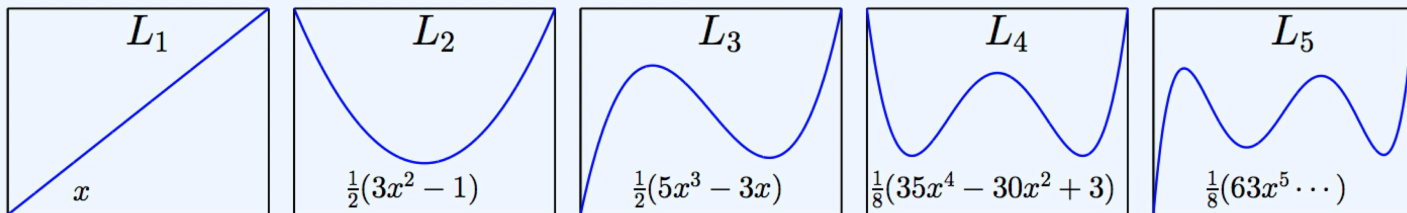
- \mathcal{H}_Q : polynomials of order Q

$$\mathcal{H}_Q = \left\{ \sum_{q=0}^Q w_q L_q(x) \right\}$$

- Linear regression in the \mathcal{Z} space with

- $z = [1, L_1(x), \dots, L_Q(x)]$

Legendre polynomials:



Regularization

Unconstrained solution

- Input $(x_1, y_1), \dots, (x_N, y_N) \rightarrow (z_1, y_1), \dots, (z_N, y_N)$

- Linear regression:

- Minimize: $E_{\text{tr}}(w) = \frac{1}{N} \sum_{n=1}^N (w^T z_n - y_n)^2$

- Minimize: $\frac{1}{N} (Zw - y)^T (Zw - y)$

- Solution $w_{\text{tr}} = (Z^T Z)^{-1} Z^T y$

Regularization

Constraining the weights

- Hard constraint: \mathcal{H}_2 is constrained version of \mathcal{H}_{10} (with $w_q = 0$ for $q > 2$)

Regularization

Constraining the weights

- Hard constraint: \mathcal{H}_2 is constrained version of \mathcal{H}_{10} (with $w_q = 0$ for $q > 2$)
- Soft-order constraint: $\sum_{q=0}^Q w_q^2 \leq C$

Regularization

Constraining the weights

- Hard constraint: \mathcal{H}_2 is constrained version of \mathcal{H}_{10} (with $w_q = 0$ for $q > 2$)

- Soft-order constraint: $\sum_{q=0}^Q w_q^2 \leq C$

- The problem given soft-order constraint:

- Minimize $\frac{1}{N}(Zw - y)^T(Zw - y)$ s.t. $\underbrace{w^T w}_{\text{smaller hypothesis space}} \leq C$

- Solution w_{reg} instead of w_{tr}

Regularization

Equivalent to the unconstrained version

- Constrained version:

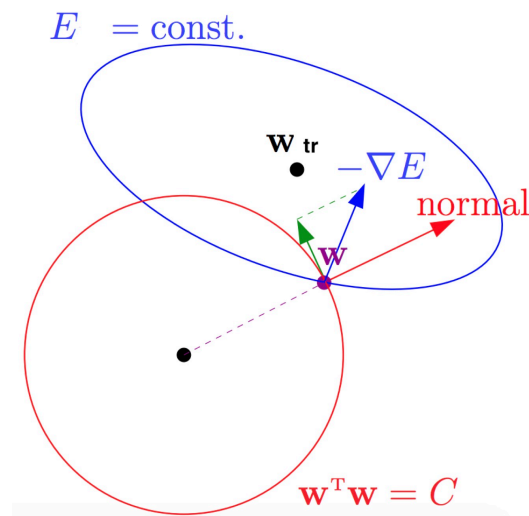
- $\min_w E_{\text{tr}}(w) = \frac{1}{N}(Zw - y)^T(Zw - y)$

- s.t. $w^T w \leq C$

- Optimal when

- $\nabla E_{\text{tr}}(w_{\text{reg}}) \propto -w_{\text{reg}}$

- Why? If $-\nabla E_{\text{tr}}(w_{\text{reg}})$ and w are not parallel, can decrease $E_{\text{tr}}(w)$ without violating the constraint



Regularization

Equivalent to the unconstrained version

- Constrained version:

$$\bullet \min_w E_{\text{tr}}(w) = \frac{1}{N}(Zw - y)^T(Zw - y) \quad \text{s.t. } w^T w \leq C$$

- Optimal when

- $\nabla E_{\text{tr}}(w_{\text{reg}}) \propto -w_{\text{reg}}$

- Assume $\nabla E_{\text{tr}}(w_{\text{reg}}) = -2\frac{\lambda}{N}w_{\text{reg}} \Rightarrow \nabla E_{\text{tr}}(w_{\text{reg}}) + 2\frac{\lambda}{N}w_{\text{reg}} = 0$

Regularization

Equivalent to the unconstrained version

- Constrained version:

- $\min_w E_{\text{tr}}(w) = \frac{1}{N}(Zw - y)^T(Zw - y) \quad \text{s.t. } w^T w \leq C$

- Optimal when

- $\nabla E_{\text{tr}}(w_{\text{reg}}) \propto -w_{\text{reg}}$

- Assume $\nabla E_{\text{tr}}(w_{\text{reg}}) = -2\frac{\lambda}{N}w_{\text{reg}} \Rightarrow \nabla E_{\text{tr}}(w_{\text{reg}}) + 2\frac{\lambda}{N}w_{\text{reg}} = 0$

- w_{reg} is also the solution of [unconstrained problem](#)

- $\min_w E_{\text{tr}}(w) + \frac{\lambda}{N}w^T w$ (Ridge regression!)

Regularization

Equivalent to the unconstrained version

- Constrained version:

- $\min_w E_{\text{tr}}(w) = \frac{1}{N}(Zw - y)^T(Zw - y) \quad \text{s.t. } w^T w \leq C$

- Optimal when

- $\nabla E_{\text{tr}}(w_{\text{reg}}) \propto -w_{\text{reg}}$

- Assume $\nabla E_{\text{tr}}(w_{\text{reg}}) = -2\frac{\lambda}{N}w_{\text{reg}} \Rightarrow \nabla E_{\text{tr}}(w_{\text{reg}}) + 2\frac{\lambda}{N}w_{\text{reg}} = 0$

- w_{reg} is also the solution of **unconstrained problem**

- $\min_w E_{\text{tr}}(w) + \frac{\lambda}{N}w^T w$ (Ridge regression!) C ↑ λ ↓

Regularization

Ridge regression solution

L-2 regularization

- $\min_w E_{\text{reg}}(w) = \frac{1}{N} \left((Zw - y)^T (Zw - y) + \lambda w^T w \right)$
- $\nabla E_{\text{reg}}(w) = 0 \Rightarrow Z^T Z(w - y) + \lambda w = 0$

Regularization

Ridge regression solution

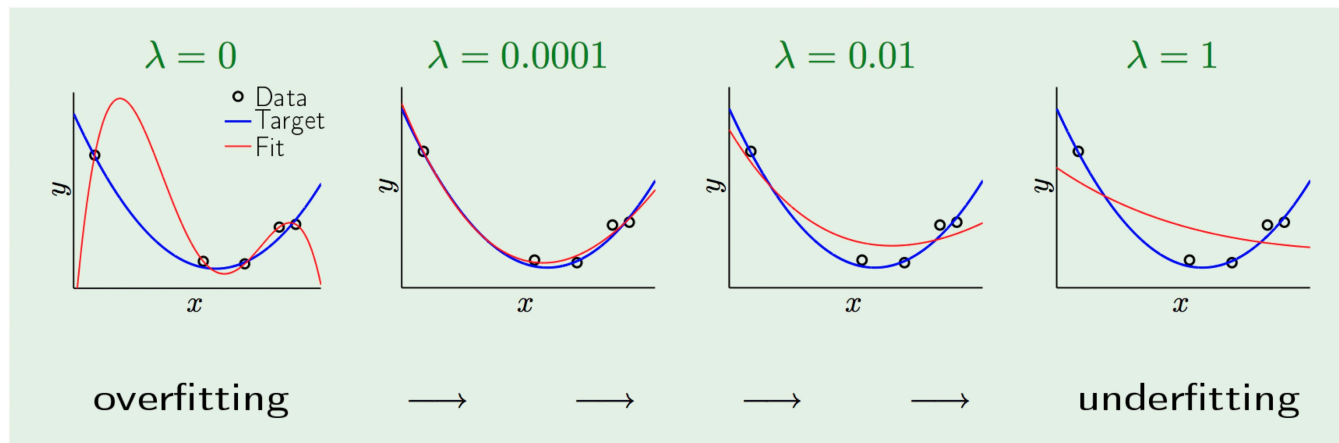
- $\min_w E_{\text{reg}}(w) = \frac{1}{N} \left((Zw - y)^T (Zw - y) + \lambda w^T w \right)$
- $\nabla E_{\text{reg}}(w) = 0 \Rightarrow Z^T Z(w - y) + \lambda w = 0$ Damping factor
- So, $w_{\text{reg}} = (Z^T Z + \lambda I)^{-1} Z^T y$ (with regularization) as opposed to $w_{\text{tr}} = (Z^T Z)^{-1} Z^T y$ (without regularization)

More numerical stable, as it could improve the $Z^T Z$ condition by adding λI

Regularization

The result

$$\min_w E_{\text{tr}}(w) + \frac{\lambda}{N} w^T w$$



Regularization

Equivalent to “weight decay”

- Consider the general case

- $\min_w E_{\text{tr}}(w) + \frac{\lambda}{N} w^T w$

Regularization

Equivalent to “weight decay”

- Consider the general case

- $\min_w E_{\text{tr}}(w) + \frac{\lambda}{N} w^T w$

- Gradient descent:

$$w_{t+1} = w_t - \eta (\nabla E_{\text{tr}}(w_t) + 2 \frac{\lambda}{N} w_t)$$

- $$= w_t \underbrace{\left(1 - 2\eta \frac{\lambda}{N}\right)}_{\text{weight decay}} - \eta \nabla E_{\text{tr}}(w_t)$$

Regularization

Variations of weight decay

- Emphasis of certain weights:

- $\sum_{q=0}^Q \gamma_q w_q^2$

- Example 1: $\gamma_q = 2^q \Rightarrow$ low-order fit
- Example 2: $\gamma_q = 2^{-q} \Rightarrow$ high-order fit

Regularization

Variations of weight decay

- Emphasis of certain weights:

- $\sum_{q=0}^Q \gamma_q w_q^2$

- Example 1: $\gamma_q = 2^q \Rightarrow$ low-order fit
- Example 2: $\gamma_q = 2^{-q} \Rightarrow$ high-order fit
- General Tikhonov regularizer:
 - $w^T H w$ with a positive semi-definite H

Regularization

Variations of weight decay

- Calling the regularizer $\Omega = \Omega(h)$, we minimize
 - $E_{\text{reg}}(h) = E_{\text{tr}}(h) + \frac{\lambda}{N}\Omega(h)$
- In general, $\Omega(h)$ can be any measurement for the “size” of h

Regularization

L2 vs L1 regularizer

- L1-regularizer: $\Omega(w) = \|w\|_1 = \sum_q |w_q|$
- Usually leads to a sparse solution (only few w_q will be nonzero)

