7.3 Regularization

Machine Learning 1: Foundations

Marius Kloft *(TUK)*

What is Regularization?

What is Regularization?

SVM, LR, and ANN employ regularization:

$$\min_{[W_l] \; b,\mathbf{w}} \underbrace{\frac{1}{2}\|\mathbf{w}\|^2 \left[+\frac{1}{2}\sum_{l=1}^{L}\|W_l\|^2\right]}_{\text{regularizer } R(\theta)} + \underbrace{C}_{\substack{\text{regularization}\\\text{constant}}} \underbrace{\sum_{i=1}^{n} \ell\left(y_i\left(\langle \mathbf{w}, \phi(\mathbf{x}_i)\rangle + b\right)\right)}_{\text{loss } L(\theta)},$$

denoting $\theta := (b, \mathbf{w}, [, W])$, with the gray terms only for ANNs.

What is Regularization?

SVM, LR, and ANN employ regularization:

$$\min_{[W_l]\ b,\mathbf{w}} \underbrace{\frac{1}{2}\|\mathbf{w}\|^2 \left[+\frac{1}{2}\sum_{l=1}^{L}\|W_l\|^2\right]}_{\text{regularizer } R(\theta)} + \underbrace{C}_{\substack{\text{regularization}\\ \text{constant}}} \underbrace{\sum_{i=1}^{n}\ell\left(y_i\left(\langle \mathbf{w}, \phi(\mathbf{x}_i)\rangle + b\right)\right)}_{\text{loss } L(\theta)},$$

denoting $\theta := (b, \mathbf{w}, [, W])$, with the gray terms only for ANNs.

Influence of the regularization constant C

- high C

What is Regularization?

SVM, LR, and ANN employ regularization:

$$\min_{[W,]\ b,\mathbf{w}} \underbrace{\frac{1}{2}\|\mathbf{w}\|^2 \left[+\frac{1}{2}\sum_{l=1}^{L}\|W_l\|^2\right]}_{\text{regularizer } R(\theta)} + \underbrace{C}_{\substack{\text{regularization}\\ \text{constant}}} \underbrace{\sum_{i=1}^{n} \ell\left(y_i\left(\langle \mathbf{w}, \phi(\mathbf{x}_i)\rangle + b\right)\right)}_{\text{loss } L(\theta)},$$

denoting $\theta := (b, \mathbf{w}, [, W])$, with the gray terms only for ANNs.

Influence of the regularization constant C

- high C
  $\Rightarrow$ focus on getting the loss small, not the regularizer

What is Regularization?

SVM, LR, and ANN employ regularization:

$$\min_{[W,]\ b,\mathbf{w}} \underbrace{\frac{1}{2}\|\mathbf{w}\|^2 \left[+\frac{1}{2}\sum_{l=1}^{L}\|W_l\|^2\right]}_{\text{regularizer } R(\theta)} + \underbrace{C}_{\substack{\text{regularization}\\ \text{constant}}} \underbrace{\sum_{i=1}^{n} \ell\left(y_i\left(\langle \mathbf{w}, \phi(\mathbf{x}_i)\rangle + b\right)\right)}_{\text{loss } L(\theta)},$$

denoting $\theta := (b, \mathbf{w}, [, W])$, with the gray terms only for ANNs.

Influence of the regularization constant C

- high C

  $\Rightarrow$ focus on getting the loss small, not the regularizer

  $\Rightarrow$ low regularization & high overfitting

What is Regularization?

SVM, LR, and ANN employ regularization:

$$\min_{[W,]\ b,\mathbf{w}} \underbrace{\frac{1}{2}\|\mathbf{w}\|^2 \left[+\frac{1}{2}\sum_{l=1}^{L}\|W_l\|^2\right]}_{\text{regularizer } R(\theta)} + \underbrace{C}_{\substack{\text{regularization}\\ \text{constant}}} \underbrace{\sum_{i=1}^{n} \ell\left(y_i\left(\langle \mathbf{w}, \phi(\mathbf{x}_i)\rangle + b\right)\right)}_{\text{loss } L(\theta)},$$

denoting $\theta := (b, \mathbf{w}, [, W])$, with the gray terms only for ANNs.

Influence of the regularization constant C

- high C
  $\Rightarrow$ focus on getting the loss small, not the regularizer
  $\Rightarrow$ low regularization & high overfitting
- low $C \Rightarrow$ high regularization & low overfitting

What is Regularization?

SVM, LR, and ANN employ regularization:

$$\min_{[W_l]\ b,\mathbf{w}} \ \underbrace{\frac{1}{2}\|\mathbf{w}\|^2 \left[+\frac{1}{2}\sum_{l=1}^{L}\|W_l\|^2\right]}_{\text{regularizer } R(\theta)} + \underbrace{C}_{\substack{\text{regularization}\\ \text{constant}}} \underbrace{\sum_{i=1}^{n} \ell\left(y_i\left(\langle \mathbf{w}, \phi(\mathbf{x}_i)\rangle + b\right)\right)}_{\text{loss } L(\theta)},$$

denoting $\theta := (b, \mathbf{w}, [, W])$, with the gray terms only for ANNs.

Influence of the regularization constant C

- high C
  $\Rightarrow$ focus on getting the loss small, not the regularizer
  $\Rightarrow$ low regularization & high overfitting
- low $C \Rightarrow$ high regularization & low overfitting

Why does it work?

Why It Works: Example of Polynomial Kernel

Recall: prediction functions are degree-m polynomials:

$$f(\mathbf{x}) = \langle \mathbf{w}, \phi(\mathbf{x}) \rangle = \sum_{i=(i_1,\ldots,i_d)\in\mathbb{N}_0^d : \sum_{j=1}^d i_j \leq m} w_i c_i x_1^{i_1} \cdots x_d^{i_d}$$

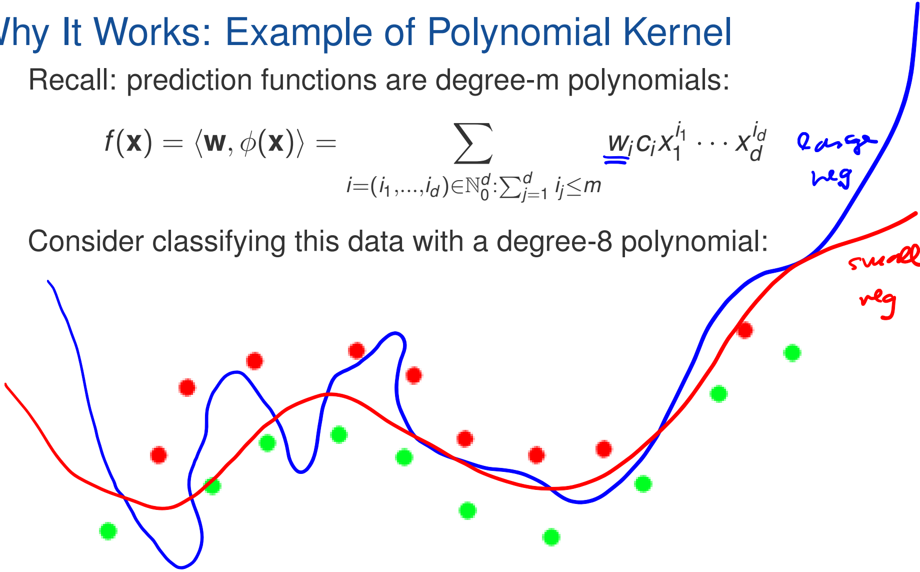
+b +b

Why It Works: Example of Polynomial Kernel

Recall: prediction functions are degree-m polynomials:

$$f(\mathbf{x}) = \langle \mathbf{w}, \phi(\mathbf{x}) \rangle = \sum_{i=(i_1,\ldots,i_d)\in\mathbb{N}_0^d : \sum_{j=1}^d i_j \leq m} w_i c_i x_1^{i_1} \cdots x_d^{i_d}$$

Consider classifying this data with a degree-8 polynomial:



The more regularization, the smaller the coefficients

- leads to smoother functions

Why Regularization Works: Regularizer in Constraint

Using a Lagrangian argument, one can show that

$$\min_{\theta} \quad \underbrace{R(\theta)}_{\text{regularizer}} + \underbrace{C}_{\substack{\text{regularization}\\ \text{constant}}} \underbrace{L(\theta)}_{\text{loss}}$$

Why Regularization Works: Regularizer in Constraint

Using a Lagrangian argument, one can show that

$$\min_{\theta} \underbrace{R(\theta)}_{\text{regularizer}} + \underbrace{C}_{\substack{\text{regularization}\\ \text{constant}}} \underbrace{L(\theta)}_{\text{loss}}$$

is equivalent to

$$\begin{aligned} \min_{\theta} \quad & L(\theta) \\ \text{s.t.} \quad & R(\theta) \leq \tilde{C} \end{aligned}$$

for some adequate choice of $\tilde{C}$.

Proof

$$\min_{\theta} \quad L(\theta) \quad \text{s.t.} \quad R(\theta) \leq \tilde{C}$$

$$\overset{(L)}{=} \max_{\lambda \geq 0} \min_{\theta} \quad L(\theta) + \lambda \cdot \left(R(\theta) - \tilde{C} \right)$$

$$\overset{(L)}{=} \min_{\theta} \max_{\lambda \geq 0} \quad L(\theta) + \lambda \cdot \left(R(\theta) - \tilde{C} \right)$$

$$= \min_{\theta} \quad L(\theta) + \lambda^* \left(R(\theta) \cancel{- \tilde{C}} \right)$$

$$\hat{=} \min_{\theta} \quad L(\theta) + \underbrace{\lambda^*}_{=: C} R(\theta)$$

Denote (θ^*, λ^*) the optimum of min-max problem

$\square$

Why Regularization Works: Regularizer in Constraint

Interpretation of SVM/LR/ANN with regularizer in constraint:

Why Regularization Works: Regularizer in Constraint

Interpretation of SVM/LR/ANN with regularizer in constraint:

- By changing $\tilde{C}$ we can control the size of the set

$$\left\{\theta : R(\theta) \leq \tilde{C}\right\},$$

from which we pick the parameters $\theta := (b, \mathbf{w}, [, W])$ of the classifier

Why Regularization Works: Regularizer in Constraint

Interpretation of SVM/LR/ANN with regularizer in constraint:

- By changing $\tilde{C}$ we can control the size of the set

$$\left\{\theta : R(\theta) \leq \tilde{C}\right\},$$

from which we pick the parameters $\theta := (b, \mathbf{w}, [, W])$ of the classifier

Bottom line: The **larger** the set, the more likely the algorithm will pick a function

- that describes the training data well
- but does not generalize well