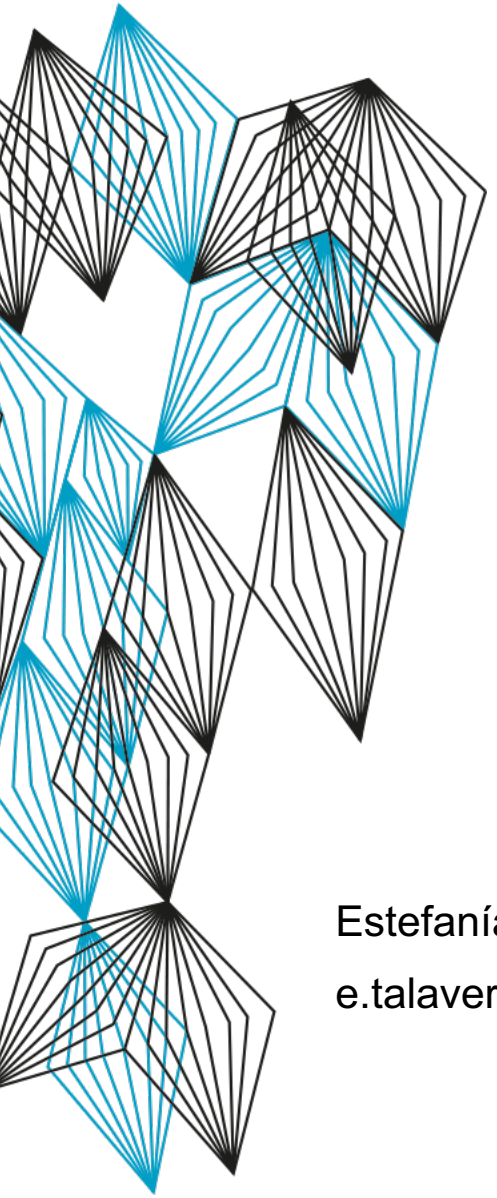


ARTIFICIAL INTELLIGENCE & CYBER SECURITY

NEURAL NETWORKS

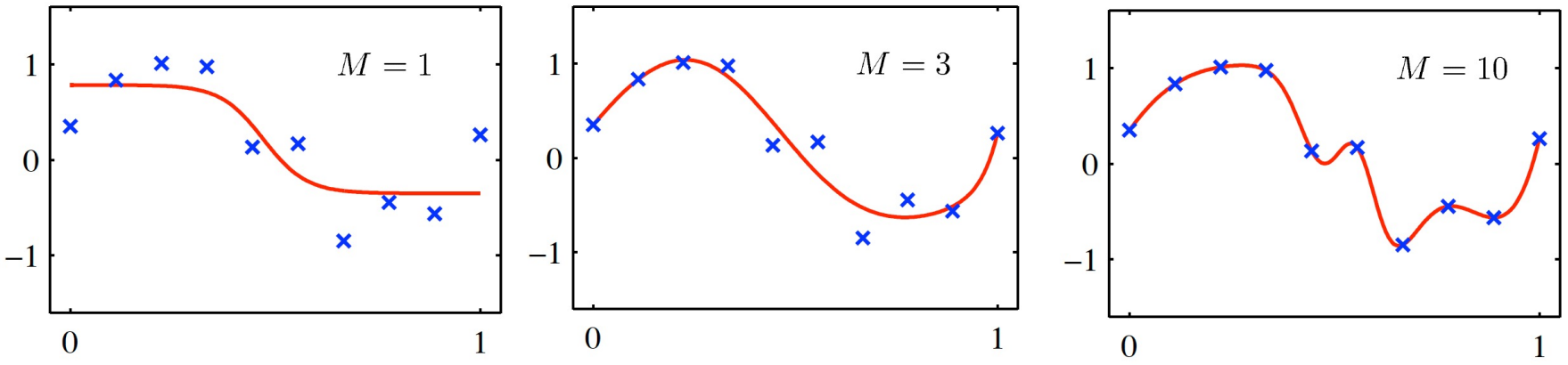
Regularization

Estefanía Talavera Martínez

e.talaveramartinez@utwente.nl

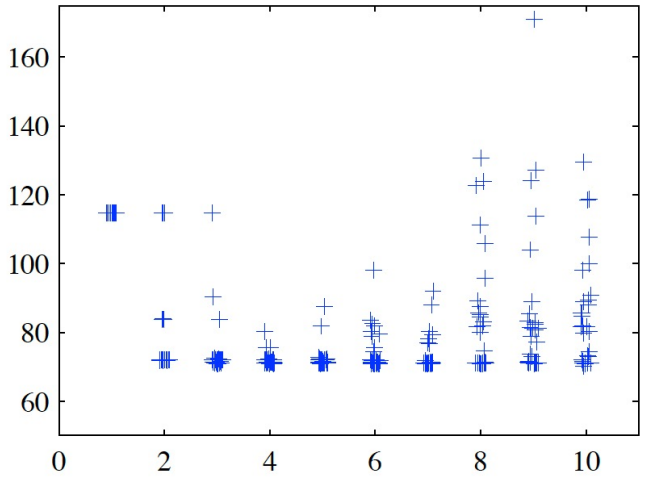
REGULARISATION

The number of input and output units is generally imposed by the problem, but the number of hidden units may vary



REGULARISATION

Yet the generalisation performance is not a simple function of M



Example: error on left-out data

- 30 random starts per size
- Initial weights sampled from a Gaussian distribution

In this particular case, the lowest validation error was for M = 8

WEIGHT DECAY

Again, the traditional technique: penalise large weights

L2 regularization

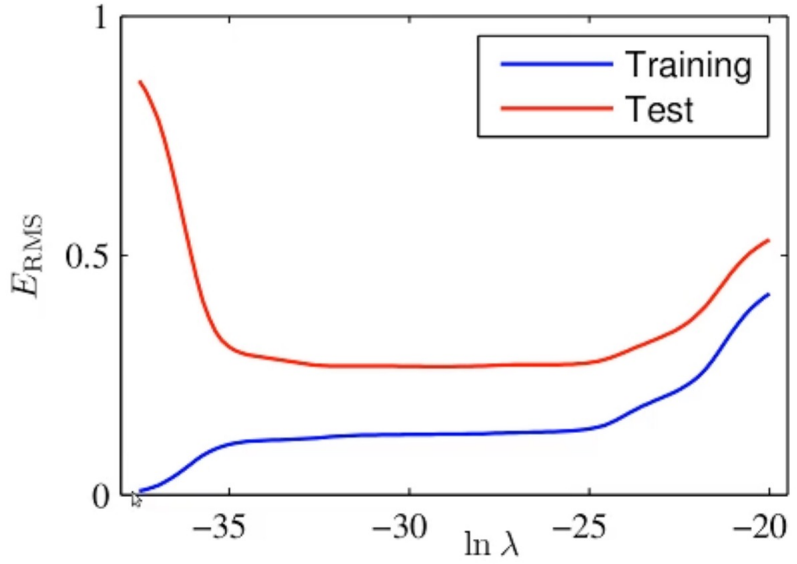
$$\tilde{E}(\mathbf{w}) = E(\mathbf{w}) + \frac{\lambda}{2}\mathbf{w}\cdot\mathbf{w},$$

Treat the weights of each layer separately, and do not constrain the biases

$$\tilde{E}(\mathbf{w}) = E(\mathbf{w}) + \frac{\lambda_1}{2}\sum_{w\in\mathcal{W}_1} w^2 + \frac{\lambda_2}{2}\sum_{w\in\mathcal{W}_2} w^2$$

Parameter shrinkage

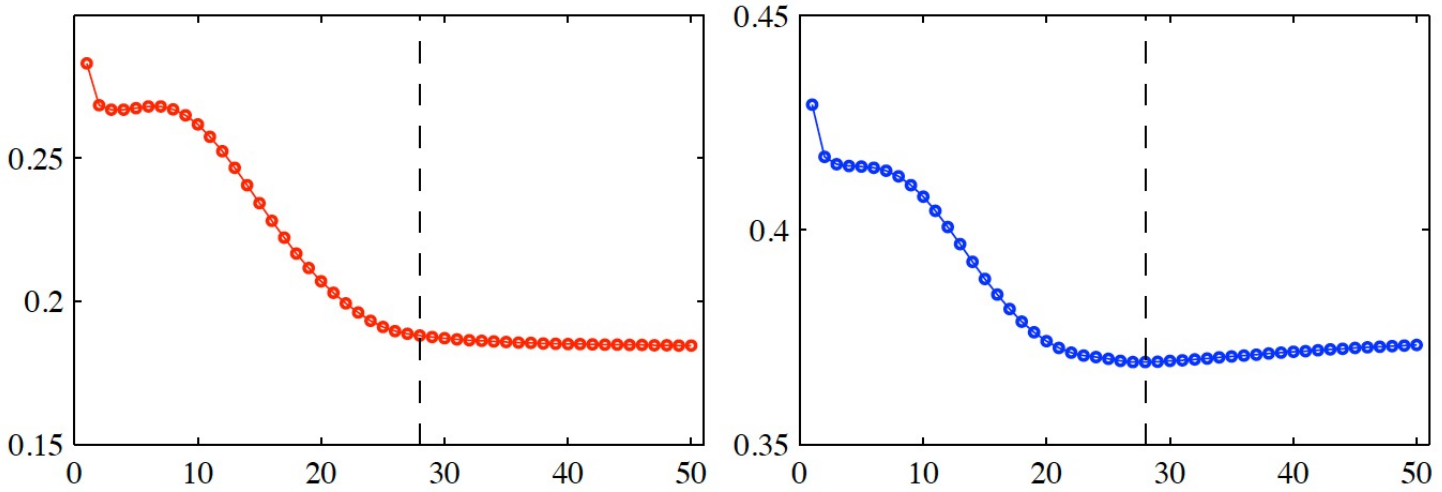
Example: Polynomial curve fitting



EARLY STOPPING

AN ALTERNATIVE IS TO STOP TRAINING WHEN

PERFORMANCE ON A VALIDATION DATASET STARTS TO DEGRADE



This is similar to weight decay: if we start from the origin, stopping early restricts the weights to small values

TECHNIQUES

WEIGHT-DECAY:

PENALIZE LARGE WEIGHTS USING PENALTIES OR CONSTRAINTS ON THEIR SQUARED VALUES

EARLY STOPPING:

START WITH SMALL WEIGHTS AND STOP THE LEARNING BEFORE IT OVERFITS.