Weight 'decay'

Minimizing $\quad E_{\text{in}}(\mathbf{w}) + \frac{\lambda}{N}\,\mathbf{w}^{\intercal}\mathbf{w} \quad$ is called weight *decay*. Why?

Gradient descent:

$$\mathbf{w}(t+1) = \mathbf{w}(t) \; - \; \eta\,\nabla E_{\text{in}}\left(\mathbf{w}(t)\right) \; - 2\,\eta\,\frac{\lambda}{N}\,\mathbf{w}(t)$$

$$= \mathbf{w}(t)\;(1 - 2\eta\frac{\lambda}{N}) - \eta\,\nabla E_{\text{in}}\left(\mathbf{w}(t)\right)$$

Applies in neural networks:

$$\mathbf{w}^{\intercal}\mathbf{w} = \sum_{l=1}^{L}\sum_{i=0}^{d^{(l-1)}}\sum_{j=1}^{d^{(l)}}\left(w_{ij}^{(l)}\right)^2$$

Variations of weight decay

Emphasis of certain weights: $$\sum_{q=0}^{Q} \gamma_q \, w_q^2$$

Examples:

$$\gamma_q = 2^q \implies \text{low-order fit}$$

$$\gamma_q = 2^{-q} \implies \text{high-order fit}$$

Neural networks: different layers get different γ's

Tikhonov regularizer: $\mathbf{w}^{\mathsf{T}} \Gamma^{\mathsf{T}} \Gamma \mathbf{w}$

Even weight growth!

We 'constrain' the weights to be large - bad!

Practical rule:

stochastic noise is 'high-frequency'

deterministic noise is also non-smooth

$\Longrightarrow$ constrain learning towards smoother hypotheses