

Learning From Data

Lecture 12

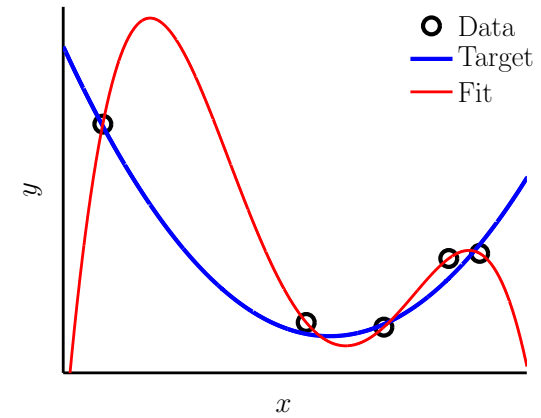
Regularization

Constraining the Model
Weight Decay
Augmented Error

M. Magdon-Ismail
CSCI 4100/6100

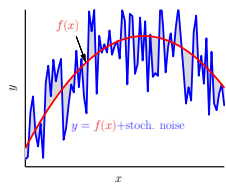
RECAP: Overfitting

Fitting the data more than is warranted

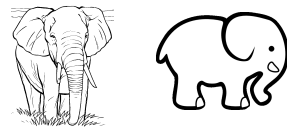
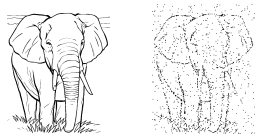
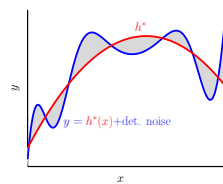


RECAP: Noise is Part of y We Cannot Model

Stochastic Noise



Deterministic Noise



Stochastic and Deterministic Noise Hurt Learning

Human: Good at extracting the simple pattern, ignoring the noise and complications.

Computer: Pays equal attention to all pixels. Needs help simplifying → (features, regularization).

Regularization

What is regularization?

A **cure** for our **tendency to fit** (get distracted by) the noise, hence improving E_{out} .

How does it work?

By **constraining** the model so that we cannot fit the noise.

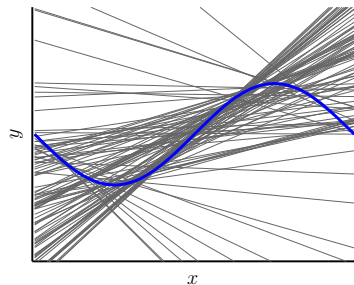
↑
putting on the brakes



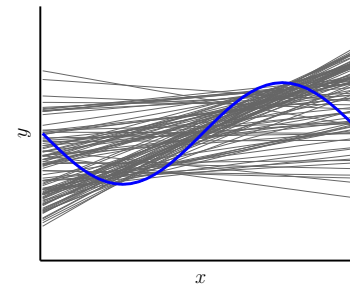
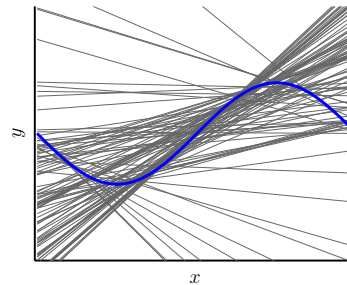
Side effects?

The medication will have **side effects** – if we cannot fit the noise, maybe we cannot fit f (the signal)?

Constraining the Model: Does it Help?



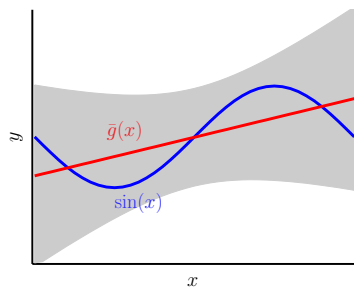
Constraining the Model: Does it Help?



constrain weights to be smaller

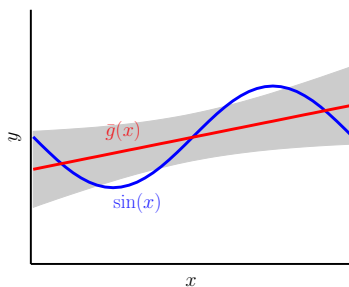
... and the winner is:

Bias Goes Up A Little



no regularization

bias = 0.21

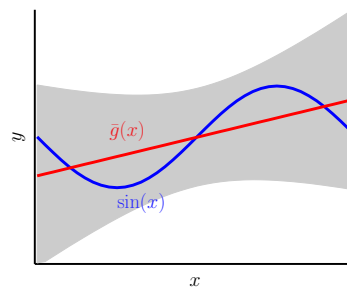


regularization

bias = 0.23

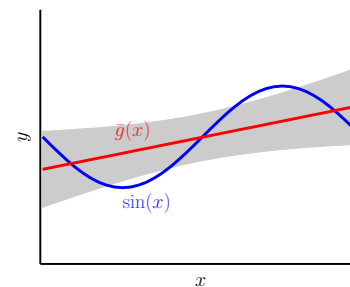
← side effect

Variance Drop is Dramatic!



no regularization

bias = 0.21
var = 1.69



regularization

bias = 0.23
var = 0.33

← side effect

← treatment

(Constant model had bias=0.5 and var=0.25.)

Regularization in a Nutshell

VC analysis:

$$E_{\text{out}}(g) \leq E_{\text{in}}(g) + \Omega(\mathcal{H})$$

If you use a simpler \mathcal{H} and get a good fit, then your E_{out} is better.

Regularization takes this a step further:

If you use a **'simpler'** h and get a good fit, then is your E_{out} better?

Polynomials of Order Q - A Useful Testbed

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

Standard Polynomial

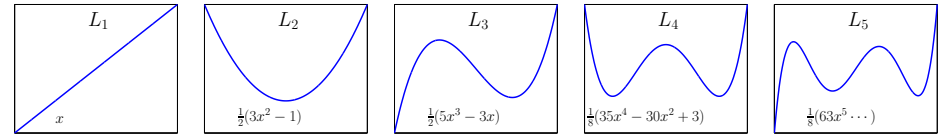
$$\mathbf{z} = \begin{bmatrix} 1 \\ x \\ x^2 \\ \vdots \\ x^Q \end{bmatrix} \quad h(x) = \mathbf{w}^T \mathbf{z}(x) = w_0 + w_1 x + \dots + w_Q x^Q$$

Legendre Polynomial

$$\mathbf{z} = \begin{bmatrix} 1 \\ L_1(x) \\ L_2(x) \\ \vdots \\ L_Q(x) \end{bmatrix} \quad h(x) = \mathbf{w}^T \mathbf{z}(x) = w_0 + w_1 L_1(x) + \dots + w_Q L_Q(x)$$

we're using linear regression

allows us to treat the weights 'independently'



RECAP: Linear Regression

$$\underbrace{(x_1, y_1), \dots, (x_N, y_N)}_{\mathbf{X} \quad \mathbf{y}} \quad \longrightarrow \quad \underbrace{(\mathbf{z}_1, y_1), \dots, (\mathbf{z}_N, y_N)}_{\mathbf{Z} \quad \mathbf{y}}$$

$$\begin{aligned} \min : E_{\text{in}}(\mathbf{w}) &= \frac{1}{N} \sum_{n=1}^N (\mathbf{w}^T \mathbf{z}_n - y_n)^2 \\ &= \frac{1}{N} (\mathbf{Z}\mathbf{w} - \mathbf{y})^T (\mathbf{Z}\mathbf{w} - \mathbf{y}) \end{aligned}$$

$$\mathbf{w}_{\text{lin}} = (\mathbf{Z}^T \mathbf{Z})^{-1} \mathbf{Z}^T \mathbf{y}$$

linear regression fit ↗

Constraining The Model: \mathcal{H}_{10} vs. \mathcal{H}_2

$$\mathcal{H}_{10} = \left\{ h(x) = w_0 + w_1 \Phi_1(x) + w_2 \Phi_2(x) + w_3 \Phi_3(x) + \dots + w_{10} \Phi_{10}(x) \right\}$$

$$\mathcal{H}_2 = \left\{ h(x) = w_0 + w_1 \Phi_1(x) + w_2 \Phi_2(x) + w_3 \Phi_3(x) + \dots + w_{10} \Phi_{10}(x) \right\}$$

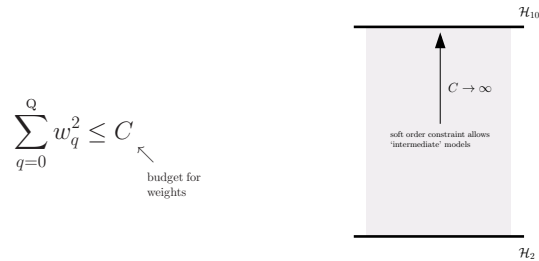
such that: $w_3 = w_4 = \dots = w_{10} = 0$

↗
a 'hard' order constraint that sets some weights to zero

$$\mathcal{H}_2 \subset \mathcal{H}_{10}$$

Soft Order Constraint

Don't set weights explicitly to zero (e.g. $w_3 = 0$).
Give a budget and let the learning choose.



Soft Order Constrained Model \mathcal{H}_C

$$\mathcal{H}_{10} = \left\{ h(x) = w_0 + w_1\Phi_1(x) + w_2\Phi_2(x) + w_3\Phi_3(x) + \dots + w_{10}\Phi_{10}(x) \right\}$$

$$\mathcal{H}_2 = \left\{ h(x) = w_0 + w_1\Phi_1(x) + w_2\Phi_2(x) + w_3\Phi_3(x) + \dots + w_{10}\Phi_{10}(x) \right\}$$

such that: $w_3 = w_4 = \dots = w_{10} = 0$

$$\mathcal{H}_C = \left\{ h(x) = w_0 + w_1\Phi_1(x) + w_2\Phi_2(x) + w_3\Phi_3(x) + \dots + w_{10}\Phi_{10}(x) \right\}$$

such that: $\sum_{q=0}^{10} w_q^2 \leq C$

a 'soft' budget constraint on the sum of weights

VC-perspective: \mathcal{H}_C is smaller than $\mathcal{H}_{10} \implies$ better generalization.

Fitting the Data

The optimal weights

$$\mathbf{w}_{\text{reg}} \in \mathcal{H}_C$$

regularized

should minimize the in-sample error, but be within the budget.

\mathbf{w}_{reg} is a solution to

$$\min : E_{\text{in}}(\mathbf{w}) = \frac{1}{N}(\mathbf{Z}\mathbf{w} - \mathbf{y})^T(\mathbf{Z}\mathbf{w} - \mathbf{y})$$

$$\text{subject to: } \mathbf{w}^T\mathbf{w} \leq C$$

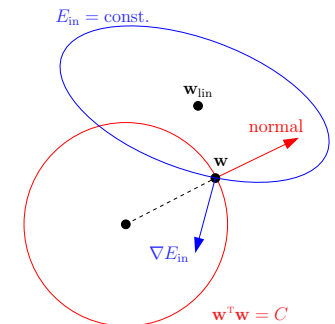
Solving For \mathbf{w}_{reg}

$$\min : E_{\text{in}}(\mathbf{w}) = \frac{1}{N}(\mathbf{Z}\mathbf{w} - \mathbf{y})^T(\mathbf{Z}\mathbf{w} - \mathbf{y})$$

$$\text{subject to: } \mathbf{w}^T\mathbf{w} \leq C$$

Observations:

- Optimal \mathbf{w} tries to get as 'close' to \mathbf{w}_{lin} as possible.
Optimal \mathbf{w} will use full budget and be on the surface $\mathbf{w}^T\mathbf{w} = C$.
- Surface $\mathbf{w}^T\mathbf{w} = C$, at optimal \mathbf{w} , should be perpendicular to ∇E_{in} .
Otherwise can move along the surface and decrease E_{in} .
- Normal** to surface $\mathbf{w}^T\mathbf{w} = C$ is the vector \mathbf{w} .
- Surface is $\perp \nabla E_{\text{in}}$; surface is \perp **normal**.
 ∇E_{in} is parallel to **normal** (but in opposite direction).



$$\nabla E_{\text{in}}(\mathbf{w}_{\text{reg}}) = -2\lambda_C \mathbf{w}_{\text{reg}}$$

λ_C , the Lagrange multiplier, is positive.
The 2 is for mathematical convenience.

Solving For \mathbf{w}_{reg}

$E_{\text{in}}(\mathbf{w})$ is minimized, **subject to: $\mathbf{w}^T \mathbf{w} \leq C$**

$$\Leftrightarrow \nabla E_{\text{in}}(\mathbf{w}_{\text{reg}}) + 2\lambda_C \mathbf{w}_{\text{reg}} = \mathbf{0}$$

$$\Leftrightarrow \nabla (E_{\text{in}}(\mathbf{w}) + \lambda_C \mathbf{w}^T \mathbf{w}) \Big|_{\mathbf{w}=\mathbf{w}_{\text{reg}}} = \mathbf{0}$$

$E_{\text{in}}(\mathbf{w}) + \lambda_C \mathbf{w}^T \mathbf{w}$ is minimized, **unconditionally**

There is a correspondence: $C \uparrow \quad \lambda_C \downarrow$

The Augmented Error

Pick a C and minimize

$E_{\text{in}}(\mathbf{w})$ **subject to: $\mathbf{w}^T \mathbf{w} \leq C$**



Pick a λ_C and minimize

$$E_{\text{aug}}(\mathbf{w}) = E_{\text{in}}(\mathbf{w}) + \lambda_C \mathbf{w}^T \mathbf{w} \quad \text{unconditionally}$$

\nwarrow
A penalty for the 'complexity' of h , measured by the size of the weights.

We can pick any budget C . Translation: we are free to pick any multiplier λ_C

What's the right C ? \leftrightarrow What's the right λ_C ?

Linear Regression With Soft Order Constraint

$$E_{\text{aug}}(\mathbf{w}) = \frac{1}{N} (\mathbf{Z}\mathbf{w} - \mathbf{y})^T (\mathbf{Z}\mathbf{w} - \mathbf{y}) + \lambda_C \mathbf{w}^T \mathbf{w}$$

\nwarrow Convenient to set $\lambda_C = \frac{\lambda}{N}$

$$E_{\text{aug}}(\mathbf{w}) = \frac{(\mathbf{Z}\mathbf{w} - \mathbf{y})^T (\mathbf{Z}\mathbf{w} - \mathbf{y}) + \lambda \mathbf{w}^T \mathbf{w}}{N}$$

\nwarrow called 'weight decay' as the penalty encourages smaller weights

Unconditionally minimize $E_{\text{aug}}(\mathbf{w})$.

The Solution for \mathbf{w}_{reg}

$$\begin{aligned} \nabla E_{\text{aug}}(\mathbf{w}) &= 2\mathbf{Z}^T (\mathbf{Z}\mathbf{w} - \mathbf{y}) + 2\lambda \mathbf{w} \\ &= 2(\mathbf{Z}^T \mathbf{Z} + \lambda \mathbf{I}) \mathbf{w} - 2\mathbf{Z}^T \mathbf{y} \end{aligned}$$

Set $\nabla E_{\text{aug}}(\mathbf{w}) = \mathbf{0}$

$$\mathbf{w}_{\text{reg}} = (\mathbf{Z}^T \mathbf{Z} + \lambda \mathbf{I})^{-1} \mathbf{Z}^T \mathbf{y}$$

\uparrow
 λ determines the amount of regularization

Recall the unconstrained solution ($\lambda = 0$):

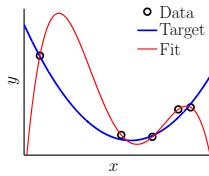
$$\mathbf{w}_{\text{lin}} = (\mathbf{Z}^T \mathbf{Z})^{-1} \mathbf{Z}^T \mathbf{y}$$

A Little Regularization ...

Minimizing $E_{in}(\mathbf{w}) + \frac{\lambda}{N} \mathbf{w}^T \mathbf{w}$ with different λ 's

$\lambda = 0$

$\lambda = 0.0001$



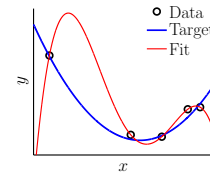
Overfitting

... Goes A Long Way

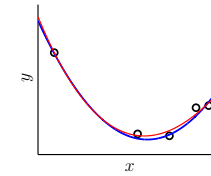
Minimizing $E_{in}(\mathbf{w}) + \frac{\lambda}{N} \mathbf{w}^T \mathbf{w}$ with different λ 's

$\lambda = 0$

$\lambda = 0.0001$



Overfitting



Wow!

Don't Overdose

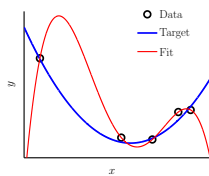
Minimizing $E_{in}(\mathbf{w}) + \frac{\lambda}{N} \mathbf{w}^T \mathbf{w}$ with different λ 's

$\lambda = 0$

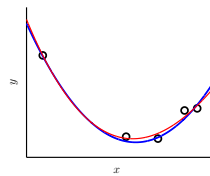
$\lambda = 0.0001$

$\lambda = 0.01$

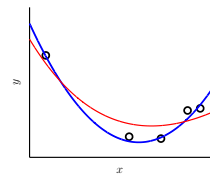
$\lambda = 1$



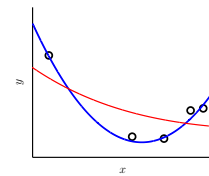
Overfitting



→

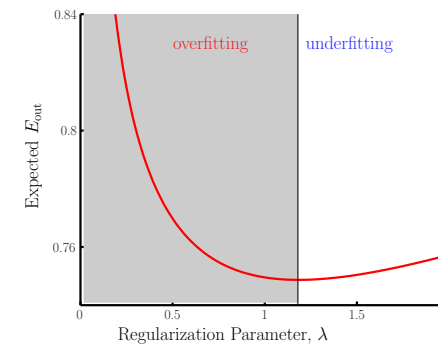


→

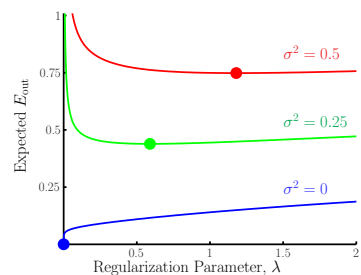


Underfitting

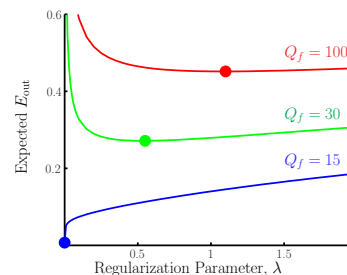
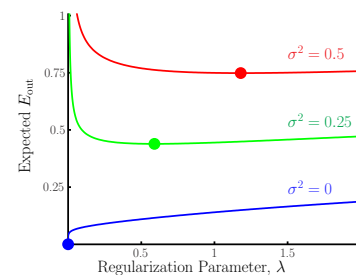
Overfitting and Underfitting



More Noise Needs More Medicine

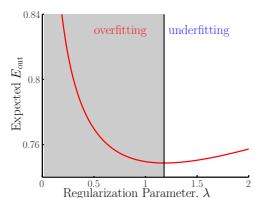


... Even For Deterministic Noise



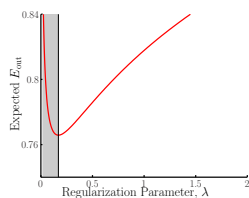
Variations on Weight Decay

Uniform Weight Decay



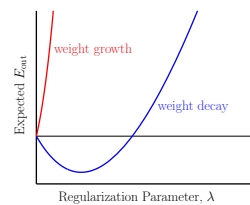
$$\sum_{q=0}^Q w_q^2$$

Low Order Fit



$$\sum_{q=0}^Q qw_q^2$$

Weight Growth!



$$\sum_{q=0}^Q \frac{1}{w_q^2}$$

Choosing a Regularizer – A Practitioner's Guide

The perfect regularizer:

constrain in the 'direction' of the target function.
target function is unknown (going around in circles ☺).

The guiding principle:

constrain in the 'direction' of **smoother** (usually simpler) hypotheses
hurts your ability to fit the 'high frequency' noise
smoother and simpler $\xrightarrow{\text{usually means}}$ weight decay not weight growth.

What if you choose the wrong regularizer?

You still have λ to play with — **validation**.

How Does Regularization Work?

Stochastic noise \rightarrow nothing you can do about that.

Good features \rightarrow helps to reduce deterministic noise.

Regularization:

Helps to combat what noise remains, especially when N is small.

Typical modus operandi: sacrifice a little **bias** for a **huge** improvement in **var**.

VC angle: you are using a smaller \mathcal{H} without sacrificing too much E_{in}

Augmented Error as a Proxy for E_{out}

$$E_{\text{aug}}(h) = E_{\text{in}}(h) + \frac{\lambda}{N} \Omega(h) \quad \leftarrow \text{this was } w'w$$

\updownarrow

$$E_{\text{out}}(h) \leq E_{\text{in}}(h) + \Omega(\mathcal{H}) \quad \leftarrow \text{this was } O\left(\sqrt{\frac{d_{\text{VC}}}{N} \ln N}\right)$$

E_{aug} can beat E_{in} as a proxy for E_{out} .

\leftarrow depends on choice of λ

