

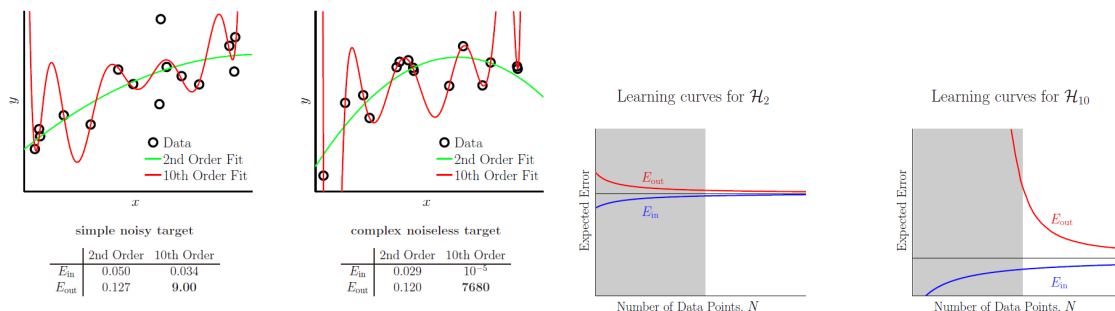
# OVERFITTING

## What is overfitting

Overfitting literally means "Fitting the data more than is warranted." The main case of overfitting is when you pick the hypothesis with lower  $E_{in}$ , and it results in higher  $E_{out}$ . This means that  $E_{in}$  alone is no longer a good guide for learning.

*Classic overfitting: simple target with excessively complex  $H$ .  $E_{in} \approx 0$ ;  $E_{out} \gg 0$*

Here is an example of overfitting:



We conclude that:

Number of data points $\uparrow$	Overfitting $\downarrow$
Noise $\uparrow$	Overfitting $\uparrow$
Target complexity $\uparrow$	Overfitting $\uparrow$

## Two types of noise

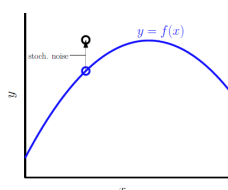
We would like to learn from  $\circ$ :

$$y_n = f(x_n)$$

Unfortunately, we only observe  $\circ$ :

$$y_n = f(x_n) + \text{'stochastic noise'}$$

no one can model this



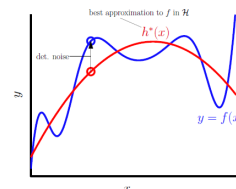
We would like to learn from  $\circ$ :

$$y_n = h^*(x_n)$$

Unfortunately, we only observe  $\circ$ :

$$y_n = f(x_n) = h^*(x_n) + \text{'deterministic noise'}$$

$H$  cannot model this



## Stochastic Noise — Data Error

Stochastic Noise: fluctuations/measurement errors we cannot model. Re-measure  $y_n$  stochastic noise changes. change  $H$  stochastic noise the same.

## Deterministic Noise — Model Error

Deterministic Noise: the part of  $f$  we cannot model. re-measure  $y_n$  deterministic noise the same. change  $H$  deterministic noise changes.

Using the bias and variance decomposition view we have:

$$E_D[E_{out}] = \sigma^2 + bias + var$$

The first two terms reflect the direct impact of the stochastic and deterministic noise. The variance of the stochastic noise is  $\sigma^2$  and the bias is directly related to the deterministic noise in that it captures the model's inability to approximate  $f$ . The var term is indirectly impacted by both types of noise, capturing a model's susceptibility to being led astray by the noise.

## How to fix it!

## Regularization

### Using the soft constraint

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

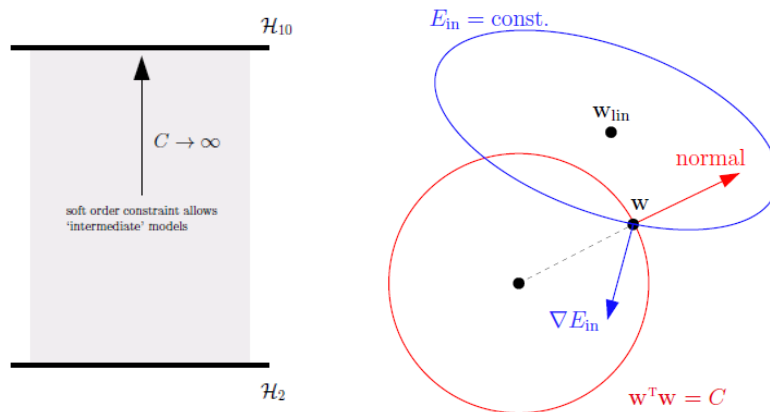
$$\sum_{q=0}^Q w_q^2 \leq C$$

Then  $w_{reg}$  is a solution to

$$\begin{aligned} \min : E_{in}(w) &= \frac{1}{N} (Zw - y)^T (Zw - y) \\ \text{subject to : } w^T w &\leq C \end{aligned}$$

Observations:

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



Then problem can be changed to:

$$\min : E_{aug}(w) = E_{in}(w) + \lambda_C w^T w$$

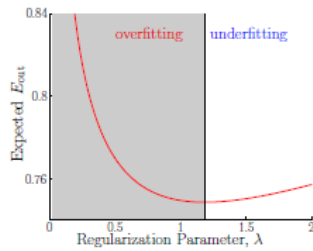
How does regularization work:

*Stochastic noise* —  $\rightarrow$  nothing you can do about that.

*Good features* —  $\rightarrow$  helps to reduce deterministic noise.

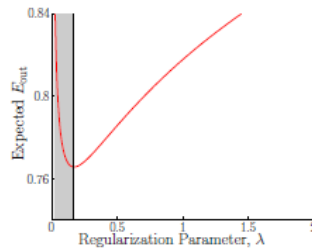
*Regularization*: Helps to combat what noise remains, especially when  $N$  is small. Typical modus operandi: sacrifice a little *bias* for a huge improvement in *var*. VC angle: you are using a smaller  $H$  without sacrificing too much  $E_{in}$

## Uniform Weight Decay



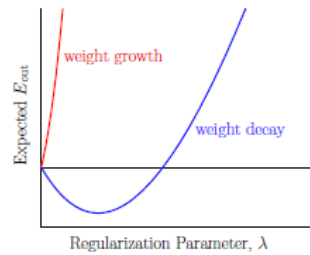
$$\sum_{q=0}^Q w_q^2$$

## Low Order Fit



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

## Weight Growth!



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

The first encourages all weights to be small, uniformly; the second pays more attention to the higher order weights, encouraging a lower order fit. In the shaded region,  $E_{out}$  increases as you decrease  $E_{in}$  (decrease  $\lambda$ ) - the regularization parameter is too small and there is not enough of a constraint on the learning, leading to decreased performance because of overfitting. In the unshaded region, the regularization parameter is too large, over-constraining the learning and not giving it enough flexibility to fit the data, leading to decreased performance because of underfitting.

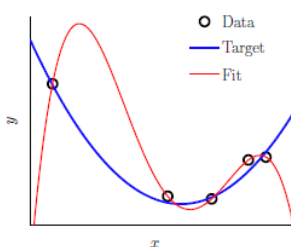
Even though we don't know either the target function or the noise, regularization helps by reducing the impact of the noise. Most common models have hypothesis sets which are naturally parameterized so that smaller parameters lead to smoother hypotheses. Thus, a weight decay type of regularizer constrains the learning towards smoother hypotheses. This helps, because stochastic noise is 'high frequency' (non-smooth). Similarly, deterministic noise (the part of the target function which cannot be modeled) also tends to be non-smooth. Thus, constraining the learning towards smoother hypotheses 'hurts' our ability to overfit the noise more than it hurts our ability to fit the useful information. These are empirical observations, not theoretically justifiable statements.

$$\lambda = 0$$

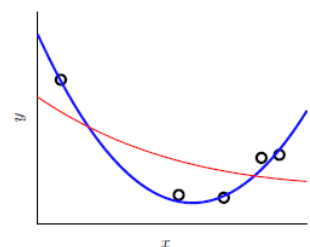
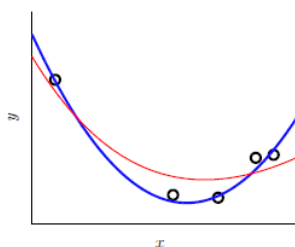
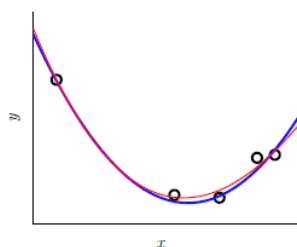
$$\lambda = 0.0001$$

$$\lambda = 0.01$$

$$\lambda = 1$$



Overfitting



Underfitting

## Lasso&Ridge regularization

### Introduction

Prior to lasso, the most widely used method for choosing which covariates to include was stepwise selection, which **only improves prediction accuracy in certain cases, such as when only a few covariates have a strong relationship with the outcome**. However, in other cases, it can make prediction error worse. Also, at the time, ridge regression was the most popular technique for improving prediction accuracy. Ridge regression improves prediction error by shrinking large regression coefficients in order to reduce overfitting, but it does not perform covariate selection and therefore does not help to make the model more interpretable.

**Lasso** is able to achieve both of these goals by forcing the sum of the absolute value of the regression coefficients to be less than a fixed value, which **forces certain coefficients to be set to zero**, effectively choosing a simpler model that does not

include those coefficients. This idea is similar to **ridge regression**, in which the sum of the squares of the **coefficients is forced to be less than a fixed value**, though in the case of ridge regression, this only shrinks the size of the coefficients, **it does not set any of them to zero**.

#### General form

Lasso regularization can be extended to a wide variety of objective functions such as those for generalized linear models, generalized estimating equations, proportional hazards models, and M-estimators in general, in the obvious way.[1][3] Given the objective function

$$\frac{1}{N} \sum_{i=1}^N f(x_i, y_i, \alpha, \beta)$$

the lasso regularized version of the estimator will be the solution to

$$\min_{\alpha, \beta} \frac{1}{N} \sum_{i=1}^N f(x_i, y_i, \alpha, \beta) \text{ subject to } \|\beta\|_1 \leq t$$

where only  $\beta$  is penalized while  $\alpha$  is free to take any allowed value, just as  $\beta_0$  was not penalized in the basic case.

Consider a sample consisting of  $N$  cases, each of which consists of  $p$  covariates and a single outcome. Let  $y_i$  be the outcome and  $x_i := (x_1, x_2, \dots, x_p)^T$  be the covariate vector for the  $i$ th case. Then the objective of lasso is to solve. It can be helpful to rewrite

$$\min_{\beta \in \mathbb{R}^p} \left\{ \frac{1}{N} \|y - X\beta\|_2^2 \right\} \text{ subject to } \|\beta\|_1 \leq t. \text{ Lasso}$$

$$\min_{\beta \in \mathbb{R}^p} \left\{ \frac{1}{N} \|y - X\beta\|_2^2 \right\} \text{ subject to } \|\beta\|_2 \leq t. \text{ Ridge}$$

in the so-called Lagrangian form

$$\min_{\beta \in \mathbb{R}^p} \left\{ \frac{1}{N} \|y - X\beta\|_2^2 + \lambda \|\beta\|_1 \right\} \text{ Lasso}$$

$$\min_{\beta \in \mathbb{R}^p} \left\{ \frac{1}{N} \|y - X\beta\|_2^2 + \lambda \|\beta\|_2 \right\} \text{ Ridge}$$

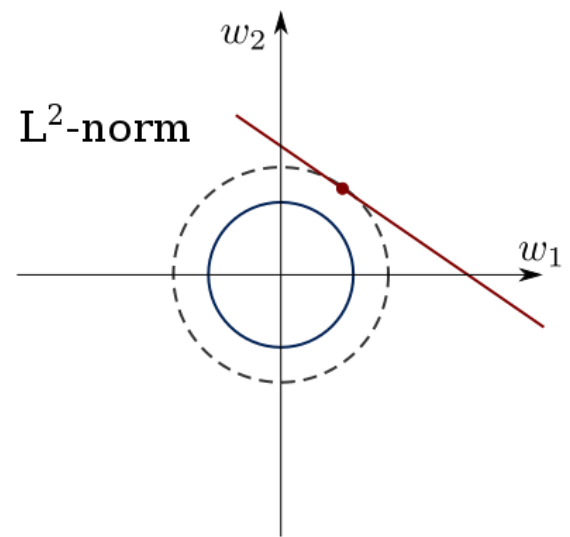
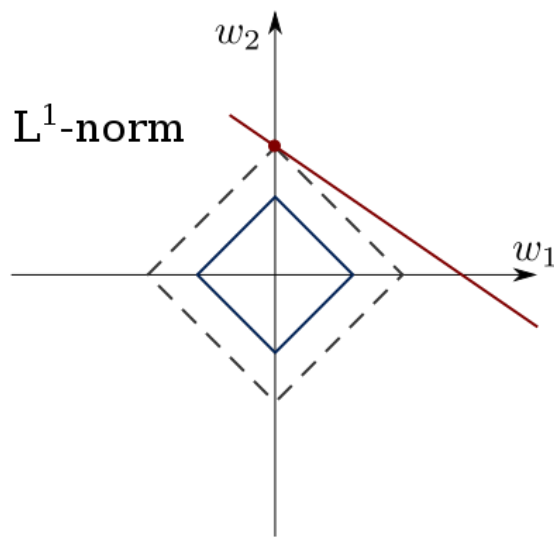
where the exact relationship between  $t$  and  $\lambda$  is data dependent.

#### Geometric interpretation

As discussed above, lasso can set coefficients to zero, while ridge regression, which appears superficially similar, cannot. This is due to the difference in the shape of the constraint boundaries in the two cases. Both lasso and ridge regression can be interpreted as minimizing the same objective function

$$\min_{\beta_0, \beta} \left\{ \frac{1}{N} \|y - \beta_0 - X\beta\|_2^2 \right\}$$

but with respect to different constraints:  $\|\beta\|_1 \leq t$  for lasso and  $\|\beta\|_2 \leq t$  for ridge. From the figure, one can see that the constraint region defined by the  $\ell^1$  norm is a square rotated so that its corners lie on the axes (in general a cross-polytope), while the region defined by the  $\ell^2$  norm is a circle (in general an  $n$ -sphere), which is rotationally invariant and, therefore, has no corners. As seen in the figure, a convex object that lies tangent to the boundary, such as the line shown, is likely to encounter a corner (or in higher dimensions an edge or higher-dimensional equivalent) of a hypercube, for which some components of  $\beta$  are identically zero, while in the case of an  $n$ -sphere, the points on the boundary for which some of the components of  $\beta$  are zero are not distinguished from the others and the convex object is no more likely to contact a point at which some components of  $\beta$  are zero than one for which none of them are.



## Validation

Regularization attempts to minimize  $E_{out}$  by working through the equation and concocting a heuristic term that emulates the penalty term. Validation, on the other hand, cuts to the chase and estimates the out-of-sample error directly.

$$E_{out}(g) = E_{in}(g) + \text{overfit penalty} \quad \text{Regularization}$$

VC bounds this using a complexity error bar for  $H$  regularization estimates this through a heuristic complexity penalty for  $g$ .

$$E_{out}(g) = E_{in}(g) + \text{overfit penalty} \quad \text{Validation}$$

validation estimates this directly

There is a difference between a validation set and a test set. Although the validation set will not be directly used for training, it will be used in making certain choices in the learning process. The minute a set affects the learning process in any way, it is no longer a test set.