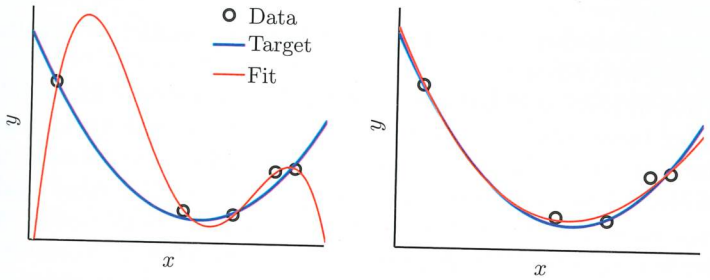
related to the deterministic noise in that it captures the model's inability to approximate f. The var term is indirectly impacted by both types of noise, capturing a model's susceptibility to being led astray by the noise.

4.2 Regularization

Regularization is our first weapon to combat overfitting. It constrains the learning algorithm to improve out-of-sample error, especially when noise is present. To whet your appetite, look at what a little regularization can do for our first overfitting example in Section 4.1. Though we only used a very small 'amount' of regularization, the fit improves dramatically.



without regularization | with regularization

Now that we have your attention, we would like to come clean. Regularization is as much an art as it is a science. Most of the methods used successfully in practice are heuristic methods. However, these methods are grounded in a mathematical framework that is developed for special cases. We will discuss both the mathematical and the heuristic, trying to maintain a balance that reflects the reality of the field.

Speaking of heuristics, one view of regularization is through the lens of the VC bound, which bounds E_{out} using a model complexity penalty $\Omega(\mathcal{H})$:

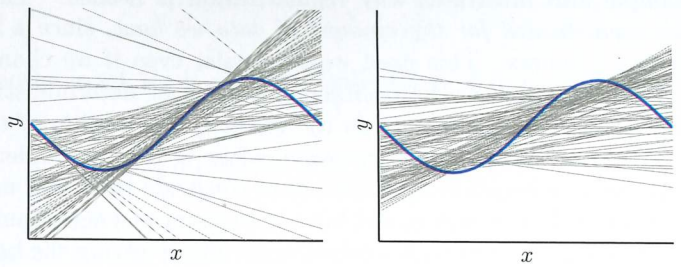
$$E_{\text{out}}(h) \leq E_{\text{in}}(h) + \Omega(\mathcal{H}) \qquad \text{for all } h \in \mathcal{H}. \tag{4.1}$$

So, we are better off if we fit the data using a simple $\mathcal{H}$. Extrapolating one step further, we should be better off by fitting the data using a 'simple' h from $\mathcal{H}$. The essence of regularization is to concoct a measure $\Omega(h)$ for the complexity of an individual hypothesis. Instead of minimizing $E_{\text{in}}(h)$ alone, one minimizes a combination of $E_{\text{in}}(h)$ and $\Omega(h)$. This avoids overfitting by constraining the learning algorithm to fit the data well using a simple hypothesis.

Example 4.1. One popular regularization technique is *weight decay*, which measures the complexity of a hypothesis h by the size of the coefficients used to represent h (e.g. in a linear model). This heuristic prefers mild lines with

small offset and slope, to wild lines with bigger offset and slope. We will get to the mechanics of weight decay shortly, but for now let's focus on the outcome.

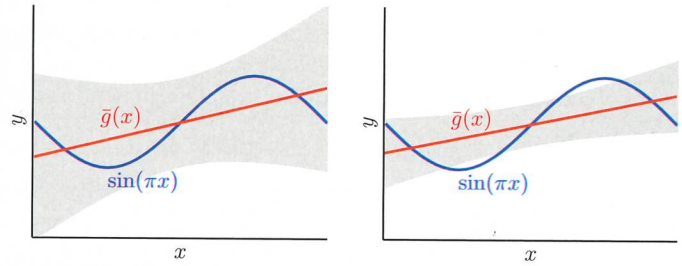
We apply weight decay to fitting the target $f(x) = \sin(\pi x)$ using $N = 2$ data points (as in Example 2.8). We sample x uniformly in $[-1, 1]$, generate a data set and fit a line to the data (our model is $\mathcal{H}_1$). The figures below show the resulting fits on the same (random) data sets with and without regularization.



without regularization | with regularization

Without regularization, the learned function varies extensively depending on the data set. As we have seen in Example 2.8, a constant model scored $E_{\text{out}} = 0.75$, handily beating the performance of the (unregularized) linear model that scored $E_{\text{out}} = 1.90$. With a little weight decay regularization, the fits to *the same data sets* are considerably less volatile. This results in a significantly lower $E_{\text{out}} = 0.56$ that beats both the constant model and the unregularized linear model.

The bias-variance decomposition helps us to understand how the regularized version beat both the unregularized version as well as the constant model.



without regularization
bias = 0.21;
var = 1.69.

with regularization
bias = 0.23;
var = 0.33.

Average hypothesis $\bar{g}$ (red) with var(x) indicated by the gray shaded region that is $\bar{g}(x) \pm \sqrt{\text{var}(x)}$.

As expected, regularization reduced the var term rather dramatically from 1.69 down to 0.33. The price paid in terms of the bias (quality of the average fit) was

modest, only slightly increasing from 0.21 to 0.23. The result was a significant decrease in the expected out-of-sample error because bias+var decreased. This is the crux of regularization. By constraining the learning algorithm to select 'simpler' hypotheses from $\mathcal{H}$, we sacrifice a little bias for a significant gain in the var. □

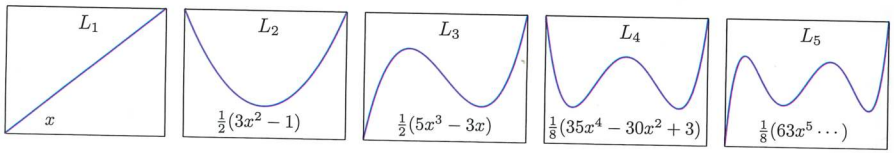
This example also illustrates why regularization is needed. The linear model is *too sophisticated for the amount of data we have,* since a line can perfectly fit any 2 points. This need would persist even if we changed the target function, as long as we have either stochastic or deterministic noise. The need for regularization depends on the *quantity and quality of the data.* Given our meager data set, our choices were either to take a simpler model, such as the model with constant functions, or to constrain the linear model. It turns out that using the complex model but constraining the algorithm toward simpler hypotheses gives us more flexibility, and ends up giving the best E_{out}. In practice, this is the rule not the exception.

Enough heuristics. Let's develop the mathematics of regularization.

4.2.1 A Soft Order Constraint

In this section, we derive a regularization method that applies to a wide variety of learning problems. To simplify the math, we will use the concrete setting of regression using Legendre polynomials, the polynomials of increasing complexity used in Exercise 4.2. So, let's first formally introduce you to the Legendre polynomials.

Consider a learning model where $\mathcal{H}$ is the set of polynomials in one variable $x \in [-1, 1]$. Instead of expressing the polynomials in terms of consecutive powers of x, we will express them as a combination of Legendre polynomials in x. Legendre polynomials are a standard set of polynomials with nice analytic properties that result in simpler derivations. The zeroth-order Legendre polynomial is the constant $L_0(x) = 1$, and the first few Legendre polynomials are illustrated below.



As you can see, when the order of the Legendre polynomial increases, the curve gets more complex. Legendre polynomials are orthogonal to each other within $x \in [-1, 1]$, and any regular polynomial can be written as a linear combination of Legendre polynomials, just like it can be written as a linear combination of powers of x.

Polynomial models are a special case of linear models in a space $\mathcal{Z}$, under a nonlinear transformation $\Phi\colon \mathcal{X} \to \mathcal{Z}$. Here, for the Qth order polynomial model, Φ transforms x into a vector $\mathbf{z}$ of Legendre polynomials,

$$\mathbf{z} = \begin{bmatrix} 1 \\ L_1(x) \\ \vdots \\ L_Q(x) \end{bmatrix}.$$

Our hypothesis set $\mathcal{H}_Q$ is a linear combination of these polynomials,

$$\mathcal{H}_Q = \left\{ h \;\middle|\; h(x) = \mathbf{w}^{\mathrm{T}}\mathbf{z} = \sum_{q=0}^{Q} w_q L_q(x) \right\}_{\mathbf{w}\in\mathbb{R}^{Q+1}},$$

where $L_0(x) = 1$. As usual, we will sometimes refer to the hypothesis h by its weight vector $\mathbf{w}$.[2] Since each h is linear in $\mathbf{w}$, we can use the machinery of linear regression from Chapter 3 to minimize the squared error

$$E_{\text{in}}(\mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{w}^{\mathrm{T}}\mathbf{z}_n - y_n)^2. \tag{4.2}$$

The case of polynomial regression with squared-error measure illustrates the main ideas of regularization well, and facilitates a solid mathematical derivation. Nonetheless, our discussion will generalize in practice to non-linear, multi-dimensional settings with more general error measures. The baseline algorithm (without regularization) is to minimize E_{in} over the hypotheses in $\mathcal{H}_Q$ to produce the final hypothesis $g(x) = \mathbf{w}_{\text{lin}}^{\mathrm{T}}\mathbf{z}$, where $\mathbf{w}_{\text{lin}} = \operatorname{argmin}_{\mathbf{w}} E_{\text{in}}(\mathbf{w})$.

Exercise 4.4

Let $\mathrm{Z} = \begin{bmatrix}\mathbf{z}_1 & \dots & \mathbf{z}_N\end{bmatrix}^{\mathrm{T}}$ be the data matrix (assume Z has full column rank); let $\mathbf{w}_{\text{lin}} = (\mathrm{Z}^{\mathrm{T}}\mathrm{Z})^{-1}\mathrm{Z}^{\mathrm{T}}\mathbf{y}$; and let $\mathrm{H} = \mathrm{Z}(\mathrm{Z}^{\mathrm{T}}\mathrm{Z})^{-1}\mathrm{Z}^{\mathrm{T}}$ (the hat matrix of Exercise 3.3). Show that

$$E_{\text{in}}(\mathbf{w}) = \frac{(\mathbf{w}-\mathbf{w}_{\text{lin}})^{\mathrm{T}}\mathrm{Z}^{\mathrm{T}}\mathrm{Z}(\mathbf{w}-\mathbf{w}_{\text{lin}}) + \mathbf{y}^{\mathrm{T}}(\mathrm{I}-\mathrm{H})\mathbf{y}}{N}, \tag{4.3}$$

where I is the identity matrix.

(a) What value of $\mathbf{w}$ minimizes E_{in}?

(b) What is the minimum in-sample error?

The task of regularization, which results in a final hypothesis $\mathbf{w}_{\text{reg}}$ instead of the simple $\mathbf{w}_{\text{lin}}$, is to constrain the learning so as to prevent overfitting the

[2] We used $\tilde{\mathbf{w}}$ and $\tilde{d}$ for the weight vector and dimension in $\mathcal{Z}$. Since we are explicitly dealing with polynomials and $\mathcal{Z}$ is the only space around, we use $\mathbf{w}$ and Q for simplicity.

data. We have already seen an example of constraining the learning; the set $\mathcal{H}_2$ can be thought of as a constrained version of $\mathcal{H}_{10}$ in the sense that some of the $\mathcal{H}_{10}$ weights are required to be zero. That is, $\mathcal{H}_2$ is a subset of $\mathcal{H}_{10}$ defined by $\mathcal{H}_2 = \{\mathbf{w} \mid \mathbf{w} \in \mathcal{H}_{10}; w_q = 0 \text{ for } q \geq 3\}$. Requiring some weights to be 0 is a *hard* constraint. We have seen that such a hard constraint on the order can help, for example $\mathcal{H}_2$ is better than $\mathcal{H}_{10}$ when there is a lot of noise and N is small. Instead of requiring some weights to be zero, we can force the weights to be small but not necessarily zero through a softer constraint such as

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

This is a 'soft order' constraint because it only encourages each weight to be small, without changing the order of the polynomial by explicitly setting some weights to zero. The in-sample optimization problem becomes:

$$\min_{\mathbf{w}} E_{\text{in}}(\mathbf{w}) \quad \text{subject to} \quad \mathbf{w}^{\mathrm{T}}\mathbf{w} \leq C. \tag{4.4}$$

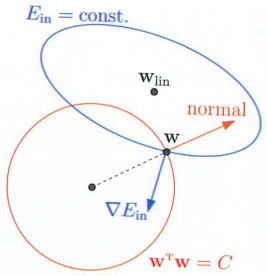
The data determines the optimal weight sizes, given the total budget C which determines the amount of regularization; the larger C is, the weaker the constraint and the smaller the amount of regularization. We can define the soft-order-constrained hypothesis set $\mathcal{H}(C)$ by

$$\mathcal{H}(C) = \{h \mid h(x) = \mathbf{w}^{\mathrm{T}}\mathbf{z} \ , \ \mathbf{w}^{\mathrm{T}}\mathbf{w} \leq C\}.$$

Equation (4.4) is equivalent to minimizing E_{in} over $\mathcal{H}(C)$. If $C_1 < C_2$, then $\mathcal{H}(C_1) \subset \mathcal{H}(C_2)$ and so $d_{\text{vc}}(\mathcal{H}(C_1)) \leq d_{\text{vc}}(\mathcal{H}(C_2))$, and we expect better generalization with $\mathcal{H}(C_1)$. Let the regularized weights $\mathbf{w}_{\text{reg}}$ be the solution to (4.4).

Solving for $\mathbf{w}_{\text{reg}}$. If $\mathbf{w}_{\text{lin}}^{\mathrm{T}}\mathbf{w}_{\text{lin}} \leq C$ then $\mathbf{w}_{\text{reg}} = \mathbf{w}_{\text{lin}}$ because $\mathbf{w}_{\text{lin}} \in \mathcal{H}(C)$. If $\mathbf{w}_{\text{lin}} \notin \mathcal{H}(C)$, then not only is $\mathbf{w}_{\text{reg}}^{\mathrm{T}}\mathbf{w}_{\text{reg}} \leq C$, but in fact $\mathbf{w}_{\text{reg}}^{\mathrm{T}}\mathbf{w}_{\text{reg}} = C$ ($\mathbf{w}_{\text{reg}}$ uses the entire budget C; see Problem 4.10).

We thus need to minimize E_{in} subject to the *equality* constraint $\mathbf{w}^{\mathrm{T}}\mathbf{w} = C$. The situation is illustrated to the right. The weights $\mathbf{w}$ must lie on the surface of the sphere $\mathbf{w}^{\mathrm{T}}\mathbf{w} = C$; the normal vector to this surface at $\mathbf{w}$ is the vector $\mathbf{w}$ itself (also in red). A surface of constant E_{in} is shown in blue; this surface is a quadratic surface (see Exercise 4.4) and the normal to this surface is $\nabla E_{\text{in}}(\mathbf{w})$. In this case, $\mathbf{w}$ cannot be optimal because $\nabla E_{\text{in}}(\mathbf{w})$ is not parallel to the red normal vector. This means that $\nabla E_{\text{in}}(\mathbf{w})$ has some non-zero component along the constraint surface, and by moving a small amount in the opposite direction of this component we can improve E_{in}, while still



remaining on the surface. If $\mathbf{w}_{\text{reg}}$ is to be optimal, then for some positive parameter λ_C

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

i.e., ∇E_{in} must be parallel to $\mathbf{w}_{\text{reg}}$, the normal vector to the constraint surface (the scaling by 2 is for mathematical convenience and the negative sign is because ∇E_{in} and $\mathbf{w}$ are in opposite directions). Equivalently, $\mathbf{w}_{\text{reg}}$ satisfies

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

because $\nabla(\mathbf{w}^{\mathrm{T}}\mathbf{w}) = 2\mathbf{w}$. So, for some $\lambda_C > 0$, $\mathbf{w}_{\text{reg}}$ locally minimizes

$$E_{\text{in}}(\mathbf{w}) + \lambda_C \mathbf{w}^{\mathrm{T}}\mathbf{w}. \tag{4.5}$$

The parameter λ_C and the vector $\mathbf{w}_{\text{reg}}$ (both of which depend on C and the data) must be chosen so as to simultaneously satisfy the gradient equality and the weight norm constraint $\mathbf{w}_{\text{reg}}^{\mathrm{T}}\mathbf{w}_{\text{reg}} = C$.[3] That $\lambda_C > 0$ is intuitive since we are enforcing smaller weights, and minimizing $E_{\text{in}}(\mathbf{w}) + \lambda_C \mathbf{w}^{\mathrm{T}}\mathbf{w}$ would not lead to smaller weights if λ_C were negative. Note that if $\mathbf{w}_{\text{lin}}^{\mathrm{T}}\mathbf{w}_{\text{lin}} \leq C$, $\mathbf{w}_{\text{reg}} = \mathbf{w}_{\text{lin}}$ and minimizing (4.5) still holds with $\lambda_C = 0$. Therefore, we have an equivalence between solving the constrained problem (4.4) and the unconstrained minimization of (4.5). This equivalence means that minimizing (4.5) is similar to minimizing E_{in} using a smaller hypothesis set, which in turn means that we can expect better generalization by minimizing (4.5) than by just minimizing E_{in}.

Other variations of the constraint in (4.4) can be used to emphasize some weights over the others. Consider the constraint $\sum_{q=0}^{Q} \gamma_q w_q^2 \leq C$. The importance γ_q given to weight w_q determines the type of regularization. For example, $\gamma_q = q$ or $\gamma_q = e^q$ encourages a low-order fit, and $\gamma_q = (1+q)^{-1}$ or $\gamma_q = e^{-q}$ encourages a high-order fit. In extreme cases, one recovers hard-order constraints by choosing some $\gamma_q = 0$ and some $\gamma_q \to \infty$.

Exercise 4.5 [Tikhonov regularizer]

A more general soft constraint is the *Tikhonov* regularization constraint

$$\mathbf{w}^{\mathrm{T}}\Gamma^{\mathrm{T}}\Gamma\mathbf{w} \leq C$$

which can capture relationships among the w_i (the matrix Γ is the Tikhonov regularizer).

(a) What should Γ be to obtain the constraint $\sum_{q=0}^{Q} w_q^2 \leq C$?

(b) What should Γ be to obtain the constraint $(\sum_{q=0}^{Q} w_q)^2 \leq C$?

[3] λ_C is known as a Lagrange multiplier and an alternate derivation of these same results can be obtained via the theory of Lagrange multipliers for constrained optimization.