the solution $\mathbf{W} = (\mathbf{X}\mathbf{X}^\top)^\dagger \mathbf{X}\mathbf{Y}$ is the one with the minimal norm and is often preferred for that reason. Thus, we will write the solutions as

$$\mathbf{W} = \begin{cases} (\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{X}\mathbf{Y} & \text{if } \mathbf{X}\mathbf{X}^\top \text{ is invertible,} \\ (\mathbf{X}\mathbf{X}^\top)^\dagger \mathbf{X}\mathbf{Y} & \text{otherwise.} \end{cases} \tag{11.11}$$

The matrix $\mathbf{X}\mathbf{X}^\top$ can be computed in $O(mN^2)$. The cost of its inversion or that of computing its pseudo-inverse is in $O(N^3)$.[19] Finally, the multiplication with $\mathbf{X}$ and $\mathbf{Y}$ takes $O(mN^2)$. Therefore, the overall complexity of computing the solution $\mathbf{W}$ is in $O(mN^2 + N^3)$. Thus, when the dimension of the feature space N is not too large, the solution can be computed efficiently.

While linear regression is simple and admits a straightforward implementation, it does not benefit from a strong generalization guarantee, since it is limited to minimizing the empirical error without controlling the norm of the weight vector and without any other regularization. Its performance is also typically poor in most applications. The next sections describe algorithms with both better theoretical guarantees and improved performance in practice.

11.3.2 Kernel ridge regression

We first present a learning guarantee for regression with bounded linear hypotheses in a feature space defined by a PDS kernel. This will provide a strong theoretical support for the *kernel ridge regression* algorithm presented in this section. The learning bounds of this section are given for the squared loss. Thus, in particular, the generalization error of a hypothesis h is defined by $R(h) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[(h(x) - y)^2\right]$.

Theorem 11.11 *Let $K\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a PDS kernel, $\Phi\colon \mathcal{X} \to \mathbb{H}$ a feature mapping associated to K, and $\mathcal{H} = \{x \mapsto \mathbf{w} \cdot \mathbf{\Phi}(x)\colon \|\mathbf{w}\|_{\mathbb{H}} \leq \Lambda\}$. Assume that there exists $r > 0$ such that $K(x,x) \leq r^2$ and $M > 0$ such that $|h(x) - y| < M$ for all $(x,y) \in \mathcal{X} \times \mathcal{Y}$. Then, for any $\delta > 0$, with probability at least $1 - \delta$, each of the following inequalities holds for all $h \in \mathcal{H}$:*

$$R(h) \leq \widehat{R}_S(h) + 4M\sqrt{\frac{r^2\Lambda^2}{m}} + M^2\sqrt{\frac{\log\frac{1}{\delta}}{2m}}$$

$$R(h) \leq \widehat{R}_S(h) + \frac{4M\Lambda\sqrt{\mathrm{Tr}[\mathbf{K}]}}{m} + 3M^2\sqrt{\frac{\log\frac{2}{\delta}}{2m}}\,.$$

[19] In the analysis of the computational complexity of the algorithms discussed in this chapter, the cubic-time complexity of matrix inversion can be replaced by a more favorable complexity $O(N^{2+\omega})$, with $\omega = .376$ using asymptotically faster matrix inversion methods such as that of Coppersmith and Winograd.

Proof: By the bound on the empirical Rademacher complexity of kernel-based hypotheses (theorem 6.12), the following holds for any sample S of size m:

$$\widehat{\mathfrak{R}}_S(\mathcal{H}) \leq \frac{\Lambda\sqrt{\mathrm{Tr}[\mathbf{K}]}}{m} \leq \sqrt{\frac{r^2\Lambda^2}{m}},$$

which implies that $\mathfrak{R}_m(\mathcal{H}) \leq \sqrt{\frac{r^2\Lambda^2}{m}}$. Combining these inequalities with the learning bounds of Theorem 11.3 yield immediately the inequalities claimed. □

The learning bounds of the theorem suggests minimizing a trade-off between the empirical squared loss (first term on the right-hand side), and the norm of the weight vector (upper bound Λ on the norm appearing in the second term), or equivalently the norm squared. Kernel ridge regression is defined by the minimization of an objective function that has precisely this form and thus is directly motivated by the theoretical analysis just presented:

$$\min_{\mathbf{w}} F(\mathbf{w}) = \lambda\|\mathbf{w}\|^2 + \sum_{i=1}^{m} (\mathbf{w}\cdot\mathbf{\Phi}(x_i) - y_i)^2 . \tag{11.12}$$

Here, λ is a positive parameter determining the trade-off between the regularization term $\|\mathbf{w}\|^2$ and the empirical mean squared error. The objective function differs from that of linear regression only by the first term, which controls the norm of $\mathbf{w}$. As in the case of linear regression, the problem can be rewritten in a more compact form as

$$\min_{\mathbf{W}} F(\mathbf{W}) = \lambda\|\mathbf{W}\|^2 + \|\mathbf{X}^\top\mathbf{W} - \mathbf{Y}\|^2, \tag{11.13}$$

where $\mathbf{X} \in \mathbb{R}^{N\times m}$ is the matrix formed by the feature vectors, $\mathbf{X} = \left[\, \Phi(x_1) \;\ldots\; \Phi(x_m) \,\right]$, $\mathbf{W} = \mathbf{w}$, and $\mathbf{Y} = (y_1, \ldots, y_m)^\top$. Here too, F is convex, by the convexity of $\mathbf{w} \mapsto \|\mathbf{w}\|^2$ and that of the sum of two convex functions, and is differentiable. Thus F admits a global minimum at $\mathbf{W}$ if and only if

$$\nabla F(\mathbf{W}) = 0 \Leftrightarrow (\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})\mathbf{W} = \mathbf{X}\mathbf{Y} \Leftrightarrow \mathbf{W} = (\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}\mathbf{X}\mathbf{Y}. \tag{11.14}$$

Note that the matrix $\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I}$ is always invertible, since its eigenvalues are the sum of the non-negative eigenvalues of the symmetric positive semidefinite matrix $\mathbf{X}\mathbf{X}^\top$ and $\lambda > 0$. Thus, kernel ridge regression admits a closed-form solution.

An alternative formulation of the optimization problem for kernel ridge regression equivalent to (11.12) is

$$\min_{\mathbf{w}} \sum_{i=1}^{m} (\mathbf{w}\cdot\mathbf{\Phi}(x_i) - y_i)^2 \quad \text{subject to: } \|\mathbf{w}\|^2 \leq \Lambda^2.$$

This makes the connection with the bounded linear hypothesis set of theorem 11.11 even more evident. Using slack variables ξ_i, for all $i \in [m]$, the problem can be

equivalently written as

$$\min_{\mathbf{w}} \sum_{i=1}^{m} \xi_i^2 \quad \text{subject to: } (\|\mathbf{w}\|^2 \leq \Lambda^2) \wedge \big(\forall i \in [m],\ \xi_i = y_i - \mathbf{w} \cdot \mathbf{\Phi}(x_i)\big).$$

This is a convex optimization problem with differentiable objective function and constraints. To derive the equivalent dual problem, we introduce the Lagrangian $\mathcal{L}$, which is defined for all $\boldsymbol{\xi}, \mathbf{w}, \boldsymbol{\alpha}'$, and $\lambda \geq 0$ by

$$\mathcal{L}(\boldsymbol{\xi}, \mathbf{w}, \boldsymbol{\alpha}', \lambda) = \sum_{i=1}^{m} \xi_i^2 + \sum_{i=1}^{m} \alpha_i'(y_i - \xi_i - \mathbf{w} \cdot \mathbf{\Phi}(x_i)) + \lambda(\|\mathbf{w}\|^2 - \Lambda^2)\,.$$

The KKT conditions lead to the following equalities:

$$\begin{aligned}
&\nabla_{\mathbf{w}}\mathcal{L} = -\sum_{i=1}^{m} \alpha_i' \mathbf{\Phi}(x_i) + 2\lambda \mathbf{w} = 0 && \implies && \mathbf{w} = \frac{1}{2\lambda} \sum_{i=1}^{m} \alpha_i' \mathbf{\Phi}(x_i)\\
&\nabla_{\xi_i}\mathcal{L} = 2\xi_i - \alpha_i' = 0 && \implies && \xi_i = \alpha_i'/2\\
&\forall i \in [m], \alpha_i'(y_i - \xi_i - \mathbf{w} \cdot \mathbf{\Phi}(x_i)) = 0\\
&\lambda(\|\mathbf{w}\|^2 - \Lambda^2) = 0.
\end{aligned}$$

Plugging in the expressions of $\mathbf{w}$ and ξ_is in that of $\mathcal{L}$ gives

$$\begin{aligned}
\mathcal{L} &= \sum_{i=1}^{m} \frac{\alpha_i'^2}{4} + \sum_{i=1}^{m} \alpha_i' y_i - \sum_{i=1}^{m} \frac{\alpha_i'^2}{2} - \frac{1}{2\lambda} \sum_{i,j=1}^{m} \alpha_i' \alpha_j' \mathbf{\Phi}(x_i)^\top \mathbf{\Phi}(x_j)\\
&\quad + \lambda\Big(\frac{1}{4\lambda^2} \|\sum_{i=1}^{m} \alpha_i' \mathbf{\Phi}(x_i)\|^2 - \Lambda^2\Big)\\
&= -\frac{1}{4} \sum_{i=1}^{m} \alpha_i'^2 + \sum_{i=1}^{m} \alpha_i' y_i - \frac{1}{4\lambda} \sum_{i,j=1}^{m} \alpha_i' \alpha_j' \mathbf{\Phi}(x_i)^\top \mathbf{\Phi}(x_j) - \lambda\Lambda^2\\
&= -\lambda \sum_{i=1}^{m} \alpha_i^2 + 2 \sum_{i=1}^{m} \alpha_i y_i - \sum_{i,j=1}^{m} \alpha_i \alpha_j \mathbf{\Phi}(x_i)^\top \mathbf{\Phi}(x_j) - \lambda\Lambda^2,
\end{aligned}$$

with $\alpha_i' = 2\lambda\alpha_i$. Thus, the equivalent dual optimization problem for KRR can be written as follows:

$$\max_{\boldsymbol{\alpha} \in \mathbb{R}^m} -\lambda \boldsymbol{\alpha}^\top \boldsymbol{\alpha} + 2\boldsymbol{\alpha}^\top \mathbf{Y} - \boldsymbol{\alpha}^\top (\mathbf{X}^\top \mathbf{X}) \boldsymbol{\alpha}\,, \tag{11.15}$$

or, more compactly, as

$$\max_{\boldsymbol{\alpha} \in \mathbb{R}^m} G(\boldsymbol{\alpha}) = -\boldsymbol{\alpha}^\top (\mathbf{K} + \lambda \mathbf{I}) \boldsymbol{\alpha} + 2\boldsymbol{\alpha}^\top \mathbf{Y}\,, \tag{11.16}$$

where $\mathbf{K} = \mathbf{X}^\top \mathbf{X}$ is the kernel matrix associated to the training sample. The objective function G is concave and differentiable. The optimal solution is obtained

by differentiating the function and setting it to zero:

$$\nabla G(\boldsymbol{\alpha}) = 0 \iff 2(\mathbf{K} + \lambda \mathbf{I})\boldsymbol{\alpha} = 2\mathbf{Y} \iff \boldsymbol{\alpha} = (\mathbf{K} + \lambda \mathbf{I})^{-1}\mathbf{Y}\,. \tag{11.17}$$

Note that $(\mathbf{K}+\lambda\mathbf{I})$ is invertible, since its eigenvalues are the sum of the eigenvalues of the SPSD matrix $\mathbf{K}$ and $\lambda > 0$. Thus, as in the primal case, the dual optimization problem admits a closed-form solution. By the first KKT equation, $\mathbf{w}$ can be determined from $\boldsymbol{\alpha}$ by

$$\mathbf{w} = \sum_{i=1}^{m} \alpha_i \boldsymbol{\Phi}(\mathbf{x}_i) = \mathbf{X}\boldsymbol{\alpha} = \mathbf{X}(\mathbf{K} + \lambda \mathbf{I})^{-1}\mathbf{Y}. \tag{11.18}$$

The hypothesis h solution can be given as follows in terms of $\boldsymbol{\alpha}$:

$$\forall x \in \mathcal{X}, \quad h(x) = \mathbf{w} \cdot \boldsymbol{\Phi}(x) = \sum_{i=1}^{m} \alpha_i K(x_i, x)\,. \tag{11.19}$$

Note that the form of the solution, $h = \sum_{i=1}^{m} \alpha_i K(x_i, \cdot)$, could be immediately predicted using the Representer theorem, since the objective function minimized by KRR falls within the general framework of theorem 6.11. This also could show that $\mathbf{w}$ could be written as $\mathbf{w} = \mathbf{X}\alpha$. This fact, combined with the following simple lemma, can be used to determine $\boldsymbol{\alpha}$ in a straightforward manner, without the intermediate derivation of the dual problem.

Lemma 11.12 *The following identity holds for any matrix* $\mathbf{X}$*:*

$$(\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1}\mathbf{X} = \mathbf{X}(\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I})^{-1}\,.$$

Proof: Observe that $(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})\mathbf{X} = \mathbf{X}(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})$. Left-multiplying by $(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})^{-1}$ this equality and right-multiplying it by $(\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}$ yields the statement of the lemma. □

Now, using this lemma, the primal solution of $\mathbf{w}$ can be rewritten as follows:

$$\mathbf{w} = (\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1}\mathbf{X}\mathbf{Y} = \mathbf{X}(\mathbf{X}^\top\mathbf{X} + \lambda \mathbf{I})^{-1}\mathbf{Y} = \mathbf{X}(\mathbf{K} + \lambda \mathbf{I})^{-1}\mathbf{Y}.$$

Comparing with $\mathbf{w} = \mathbf{X}\alpha$ gives immediately $\boldsymbol{\alpha} = (\mathbf{K} + \lambda \mathbf{I})^{-1}\mathbf{Y}$.

Our presentation of the KRR algorithm was given for linear hypotheses with no offset, that is we implicitly assumed $b = 0$. It is common to use this formulation and to extend it to the general case by augmenting the feature vector $\boldsymbol{\Phi}(x)$ with an extra component equal to one for all $x \in \mathcal{X}$ and the weight vector $\mathbf{w}$ with an extra component $b \in \mathbb{R}$. For the augmented feature vector $\boldsymbol{\Phi}'(x) \in \mathbb{R}^{N+1}$ and weight vector $\mathbf{w}' \in \mathbb{R}^{N+1}$, we have $\mathbf{w}' \cdot \boldsymbol{\Phi}'(x) = \mathbf{w} \cdot \boldsymbol{\Phi}(x) + b$. Nevertheless, this formulation does not coincide with the general KRR algorithm where a solution of the form $x \mapsto \mathbf{w} \cdot \boldsymbol{\Phi}(x) + b$ is sought. This is because for the general KRR, the regularization term is $\lambda\|\mathbf{w}\|$, while for the extension just described it is $\lambda\|\mathbf{w}'\|$.

Table 11.1
Comparison of the running-time complexity of KRR for computing the solution or the prediction value of a point in both the primal and the dual case. κ denotes the time complexity of computing a kernel value; for polynomial and Gaussian kernels, $\kappa = O(N)$.

	Solution	Prediction
Primal	$O(mN^2 + N^3)$	$O(N)$
Dual	$O(\kappa m^2 + m^3)$	$O(\kappa m)$

In both the primal and dual cases, KRR admits a closed-form solution. Table 11.1 gives the time complexity of the algorithm for computing the solution and the one for determining the prediction value of a point in both cases. In the primal case, determining the solution $\mathbf{w}$ requires computing matrix $\mathbf{X}\mathbf{X}^\top$, which takes $O(mN^2)$, the inversion of $(\mathbf{X}\mathbf{X}^\top + \lambda\mathbf{I})$, which is in $O(N^3)$, and multiplication with $\mathbf{X}$, which is in $O(mN^2)$. Prediction requires computing the inner product of $\mathbf{w}$ with a feature vector of the same dimension that can be achieved in $O(N)$. The dual solution first requires computing the kernel matrix $\mathbf{K}$. Let κ be the maximum cost of computing $K(x, x')$ for all pairs $(x, x') \in \mathcal{X} \times \mathcal{X}$. Then, $\mathbf{K}$ can be computed in $O(\kappa m^2)$. The inversion of matrix $\mathbf{K} + \lambda\mathbf{I}$ can be achieved in $O(m^3)$ and multiplication with $\mathbf{Y}$ takes $O(m^2)$. Prediction requires computing the vector $(K(x_1, x), \ldots, K(x_m, x))^\top$ for some $x \in \mathcal{X}$, which requires $O(\kappa m)$, and the inner product with $\boldsymbol{\alpha}$, which is in $O(m)$.

Thus, in both cases, the main step for computing the solution is a matrix inversion, which takes $O(N^3)$ in the primal case, $O(m^3)$ in the dual case. When the dimension of the feature space is relatively small, solving the primal problem is advantageous, while for high-dimensional spaces and medium-sized training sets, solving the dual is preferable. Note that for relatively large matrices, the space complexity could also be an issue: the size of relatively large matrices could be prohibitive for memory storage and the use of external memory could significantly affect the running time of the algorithm.

For sparse matrices, there exist several techniques for faster computations of the matrix inversion. This can be useful in the primal case where the features can be relatively sparse. On the other hand, the kernel matrix $\mathbf{K}$ is typically dense; thus, there is less hope for benefiting from such techniques in the dual case. In such cases, or, more generally, to deal with the time and space complexity issues arising when m and N are large, approximation methods using low-rank approximations via the Nyström method or the partial Cholesky decomposition can be used very effectively.

The KRR algorithm admits several advantages: it benefits from favorable theoretical guarantees since it can be derived directly from the generalization bound we

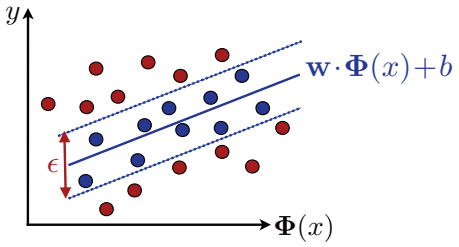

Figure 11.4
SVR attempts to fit a "tube" with width ϵ to the data. Training data within the "epsilon tube" (blue points) incur no loss.

presented; it admits a closed-form solution, which can make the analysis of many of its properties convenient; and it can be used with PDS kernels, which extends its use to non-linear regression solutions and more general features spaces. KRR also admits favorable stability properties that we discuss in chapter 14.

The algorithm can be generalized to learning a mapping from $\mathcal{X}$ to $\mathbb{R}^p$, $p > 1$. This can be done by formulating the problem as p independent regression problems, each consisting of predicting one of the p target components. Remarkably, the computation of the solution for this generalized algorithm requires only a single matrix inversion, e.g., $(\mathbf{K} + \lambda \mathbf{I})^{-1}$ in the dual case, regardless of the value of p.

One drawback of the KRR algorithm, in addition to the computational issues for determining the solution for relatively large matrices, is the fact that the solution it returns is typically not sparse. The next two sections present two sparse algorithms for linear regression.

11.3.3 Support vector regression

In this section, we present the *support vector regression* (SVR) algorithm, which is inspired by the SVM algorithm presented for classification in chapter 5. The main idea of the algorithm consists of fitting a tube of width $\epsilon > 0$ to the data, as illustrated by figure 11.4. As in binary classification, this defines two sets of points: those falling inside the tube, which are ϵ-close to the function predicted and thus not penalized, and those falling outside, which are penalized based on their distance to the predicted function, in a way that is similar to the penalization used by SVMs in classification.

Using a hypothesis set $\mathcal{H}$ of linear functions: $\mathcal{H} = \{x \mapsto \mathbf{w} \cdot \mathbf{\Phi}(x) + b \colon \mathbf{w} \in \mathbb{R}^N, b \in \mathbb{R}\}$, where $\mathbf{\Phi}$ is the feature mapping corresponding some PDS kernel K, the optimization problem for SVR can be written as follows:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_{i=1}^{m} \left|y_i - (\mathbf{w} \cdot \mathbf{\Phi}(x_i) + b)\right|_\epsilon, \tag{11.20}$$

where $|\cdot|_\epsilon$ denotes the *ϵ-insensitive loss*:

$$\forall y, y' \in \mathcal{Y}, \quad |y' - y|_\epsilon = \max(0, |y' - y| - \epsilon). \tag{11.21}$$

The use of this loss function leads to sparse solutions with a relatively small number of support vectors. Using slack variables $\xi_i \geq 0$ and $\xi'_i \geq 0$, $i \in [m]$, the optimization problem can be equivalently written as

$$\begin{aligned} \min_{\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\xi}'} \ & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m}(\xi_i + \xi'_i) \\ \text{subject to } & (\mathbf{w} \cdot \boldsymbol{\Phi}(x_i) + b) - y_i \leq \epsilon + \xi_i \\ & y_i - (\mathbf{w} \cdot \boldsymbol{\Phi}(x_i) + b) \leq \epsilon + \xi'_i \\ & \xi_i \geq 0, \xi'_i \geq 0, \ \forall i \in [m]. \end{aligned} \tag{11.22}$$

This is a convex quadratic program (QP) with affine constraints. Introducing the Lagrangian and applying the KKT conditions leads to the following equivalent dual problem in terms of the kernel matrix $\mathbf{K}$:

$$\begin{aligned} \max_{\boldsymbol{\alpha}, \boldsymbol{\alpha}'} \ & -\epsilon(\boldsymbol{\alpha}' + \boldsymbol{\alpha})^\top \mathbf{1} + (\boldsymbol{\alpha}' - \boldsymbol{\alpha})^\top \mathbf{y} - \frac{1}{2}(\boldsymbol{\alpha}' - \boldsymbol{\alpha})^\top \mathbf{K}(\boldsymbol{\alpha}' - \boldsymbol{\alpha}) \\ \text{subject to: } & (\mathbf{0} \leq \boldsymbol{\alpha} \leq \mathbf{C}) \wedge (\mathbf{0} \leq \boldsymbol{\alpha}' \leq \mathbf{C}) \wedge ((\boldsymbol{\alpha}' - \boldsymbol{\alpha})^\top \mathbf{1} = 0)\,. \end{aligned} \tag{11.23}$$

Any PDS kernel K can be used with SVR, which extends the algorithm to non-linear regression solutions. Problem (11.23) is a convex QP similar to the dual problem of SVMs and can be solved using similar optimization techniques. The solutions $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}'$ define the hypothesis h returned by SVR as follows:

$$\forall x \in \mathcal{X}, \quad h(x) = \sum_{i=1}^{m}(\alpha'_i - \alpha_i)K(\mathbf{x}_i, \mathbf{x}) + b\,, \tag{11.24}$$

where the offset b can be obtained from a point x_j with $0 < \alpha_j < C$ by

$$b = -\sum_{i=1}^{m}(\alpha'_i - \alpha_i)K(x_i, x_j) + y_j + \epsilon, \tag{11.25}$$

or from a point x_j with $0 < \alpha'_j < C$ via

$$b = -\sum_{i=1}^{m}(\alpha'_i - \alpha_i)K(x_i, x_j) + y_j - \epsilon. \tag{11.26}$$

By the complementarity conditions, for all $i \in [m]$, the following equalities hold:

$$\begin{aligned} \alpha_i\big((\mathbf{w} \cdot \boldsymbol{\Phi}(x_i) + b) - y_i - \epsilon - \xi_i\big) &= 0 \\ \alpha'_i\big((\mathbf{w} \cdot \boldsymbol{\Phi}(x_i) + b) - y_i + \epsilon + \xi'_i\big) &= 0. \end{aligned}$$

Thus, if $\alpha_i \neq 0$ or $\alpha'_i \neq 0$, that is if x_i is a support vector, then, either $(\mathbf{w} \cdot \mathbf{\Phi}(x_i) + b) - y_i - \epsilon = \xi_i$ holds or $y_i - (\mathbf{w} \cdot \mathbf{\Phi}(x_i) + b) - \epsilon = \xi'_i$. This shows that support vectors points lying outside the ϵ-tube. Of course, at most one of α_i or α'_i is non-zero for any point x_i: the hypothesis either overestimates or underestimates the true label by more than ϵ. For the points within the ϵ-tube, we have $\alpha_j = \alpha'_j = 0$; thus, these points do not contribute to the definition of the hypothesis returned by SVR. Thus, when the number of points inside the tube is relatively large, the hypothesis returned by SVR is relatively sparse. The choice of the parameter ϵ determines a trade-off between sparsity and accuracy: larger ϵ values provide sparser solutions, since more points can fall within the ϵ-tube, but may ignore too many key points for determining an accurate solution.

The following generalization bounds hold for the ϵ-insensitive loss and kernel-based hypotheses and thus for the SVR algorithm. We denote by $\mathcal{D}$ the distribution according to which sample points are drawn and by $\widehat{\mathcal{D}}$ the empirical distribution defined by a training sample of size m.

Theorem 11.13 *Let $K \colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a PDS kernel, let $\Phi \colon \mathcal{X} \to \mathbb{H}$ be a feature mapping associated to K and let $\mathcal{H} = \{x \mapsto \mathbf{w} \cdot \mathbf{\Phi}(x) \colon \|\mathbf{w}\|_{\mathbb{H}} \leq \Lambda\}$. Assume that there exists $r > 0$ such that $K(x, x) \leq r^2$ and $M > 0$ such that $|h(x) - y| \leq M$ for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$. Fix $\epsilon > 0$. Then, for any $\delta > 0$, with probability at least $1 - \delta$, each of the following inequalities holds for all $h \in \mathcal{H}$,*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[|h(x) - y|_\epsilon\right] \leq \mathop{\mathbb{E}}_{(x,y)\sim\widehat{\mathcal{D}}}\left[|h(x) - y|_\epsilon\right] + 2\sqrt{\frac{r^2\Lambda^2}{m}} + M\sqrt{\frac{\log\frac{1}{\delta}}{2m}}$$

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[|h(x) - y|_\epsilon\right] \leq \mathop{\mathbb{E}}_{(x,y)\sim\widehat{\mathcal{D}}}\left[|h(x) - y|_\epsilon\right] + \frac{2\Lambda\sqrt{\mathrm{Tr}[\mathbf{K}]}}{m} + 3M\sqrt{\frac{\log\frac{2}{\delta}}{2m}}.$$

Proof: Since for any $y' \in \mathcal{Y}$, the function $y \mapsto |y - y'|_\epsilon$ is 1-Lipschitz, the result follows Theorem 11.3 and the bound on the empirical Rademacher complexity of $\mathcal{H}$. □

These results provide theoretical guarantees for the SVR algorithm. Notice, however, that the theorem does not provide guarantees for the expected loss of the hypotheses in terms of the squared loss. For $0 < \epsilon < 1/4$, the inequality $|x|^2 \leq |x|_\epsilon$ holds for all x in $[-\eta'_\epsilon, -\eta_\epsilon] \cup [\eta_\epsilon, \eta'_\epsilon]$ with $\eta_\epsilon = \frac{1-\sqrt{1-4\epsilon}}{2}$ and $\eta'_\epsilon = \frac{1+\sqrt{1-4\epsilon}}{2}$. For small values of ϵ, $\eta_\epsilon \approx 0$ and $\eta'_\epsilon \approx 1$, thus, if $M = 2r\lambda \leq 1$, then, the squared loss can be upper bounded by the ϵ-insensitive loss for almost all values of $(h(x) - y)$ in $[-1, 1]$ and the theorem can be used to derive a useful generalization bound for the squared loss.

More generally, if the objective is to achieve a small squared loss, then, SVR can be modified by using the *quadratic ϵ-insensitive loss*, that is the square of the ϵ-insensitive loss, which also leads to a convex QP. We will refer by *quadratic SVR* to

this version of the algorithm. Introducing the Lagrangian and applying the KKT conditions leads to the following equivalent dual optimization problem for quadratic SVR in terms of the kernel matrix $\mathbf{K}$:

$$\max_{\boldsymbol{\alpha},\boldsymbol{\alpha}'} \; -\epsilon(\boldsymbol{\alpha}'+\boldsymbol{\alpha})^\top \mathbf{1} + (\boldsymbol{\alpha}'-\boldsymbol{\alpha})^\top \mathbf{y} - \frac{1}{2}(\boldsymbol{\alpha}'-\boldsymbol{\alpha})^\top \Big(\mathbf{K} + \frac{1}{C}\mathbf{I}\Big)(\boldsymbol{\alpha}'-\boldsymbol{\alpha}) \tag{11.27}$$
$$\text{subject to: } (\boldsymbol{\alpha} \geq \mathbf{0}) \wedge (\boldsymbol{\alpha}' \geq \mathbf{0}) \wedge (\boldsymbol{\alpha}'-\boldsymbol{\alpha})^\top \mathbf{1} = 0)\,.$$

Any PDS kernel K can be used with quadratic SVR, which extends the algorithm to non-linear regression solutions. Problem (11.27) is a convex QP similar to the dual problem of SVMs in the separable case and can be solved using similar optimization techniques. The solutions $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}'$ define the hypothesis h returned by SVR as follows:

$$h(x) = \sum_{i=1}^{m} (\alpha'_i - \alpha_i) K(\mathbf{x}_i, \mathbf{x}) + b\,, \tag{11.28}$$

where the offset b can be obtained from a point x_j with $0 < \alpha_j < C$ or $0 < \alpha'_j < C$ exactly as in the case of SVR with (non-quadratic) ϵ-insensitive loss. Note that for $\epsilon = 0$, the quadratic SVR algorithm coincides with KRR as can be seen from the dual optimization problem (the additional constraint $(\boldsymbol{\alpha}'-\boldsymbol{\alpha})^\top \mathbf{1} = 0$ appears here due to use of an offset b). The following generalization bound holds for quadratic SVR. It can be shown in a way that is similar to the proof of theorem 11.13 using the fact that the quadratic ϵ-insensitive function $x \mapsto |x|^2_\epsilon$ is $2M$-Lipschitz over the interval $[-M, +M]$.

Theorem 11.14 *Let $K\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a PDS kernel, let $\Phi\colon \mathcal{X} \to \mathbb{H}$ be a feature mapping associated to K and let $\mathcal{H} = \{x \mapsto \mathbf{w} \cdot \boldsymbol{\Phi}(x)\colon \|\mathbf{w}\|_{\mathbb{H}} \leq \Lambda\}$. Assume that there exists $r > 0$ such that $K(x,x) \leq r^2$ and $M > 0$ such that $|h(x) - y| \leq M$ for all $(x,y) \in \mathcal{X} \times \mathcal{Y}$. Fix $\epsilon > 0$. Then, for any $\delta > 0$, with probability at least $1-\delta$, each of the following inequalities holds for all $h \in \mathcal{H}$,*

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[|h(x)-y|^2_\epsilon\right] \leq \mathop{\mathbb{E}}_{(x,y)\sim\widehat{\mathcal{D}}}\left[|h(x)-y|^2_\epsilon\right] + 4M\sqrt{\frac{r^2\Lambda^2}{m}} + M^2\sqrt{\frac{\log\frac{1}{\delta}}{2m}}$$

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}\left[|h(x)-y|^2_\epsilon\right] \leq \mathop{\mathbb{E}}_{(x,y)\sim\widehat{\mathcal{D}}}\left[|h(x)-y|^2_\epsilon\right] + \frac{4M\Lambda\sqrt{\mathrm{Tr}[\mathbf{K}]}}{m} + 3M^2\sqrt{\frac{\log\frac{2}{\delta}}{2m}}.$$

This theorem provides a strong justification for the quadratic SVR algorithm. Alternative convex loss functions can be used to define regression algorithms, in particular the *Huber loss* (see figure 11.5), which penalizes smaller errors quadratically and larger ones only linearly.

SVR admits several advantages: the algorithm is based on solid theoretical guarantees, the solution returned is sparse, and it allows a natural use of PDS kernels,

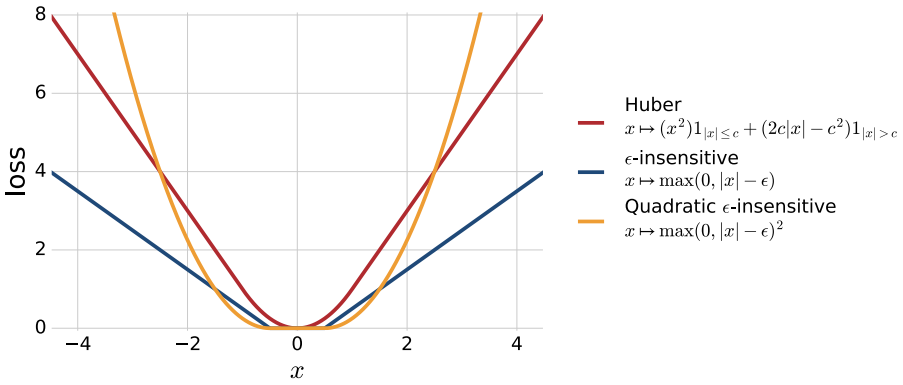

Figure 11.5
Alternative loss functions that can be used in conjunction with SVR.

which extend the algorithm to non-linear regression solutions. SVR also admits favorable stability properties that we discuss in chapter 14. However, one drawback of the algorithm is that it requires the selection of two parameters, C and ϵ. These can be selected via cross-validation, as in the case of SVMs, but this requires a relatively larger validation set. Some heuristics are often used to guide the search for their values: C is searched near the maximum value of the labels in the absence of an offset ($b = 0$) and for a normalized kernel, and ϵ is chosen close to the average difference of the labels. As already discussed, the value of ϵ determines the number of support vectors and the sparsity of the solution. Another drawback of SVR is that, as in the case of SVMs or KRR, it may be computationally expensive when dealing with large training sets. One effective solution in such cases, as for KRR, consists of approximating the kernel matrix using low-rank approximations via the Nyström method or the partial Cholesky decomposition. In the next section, we discuss an alternative sparse algorithm for regression.

11.3.4 Lasso

Unlike the KRR and SVR algorithms, the Lasso (least absolute shrinkage and selection operator) algorithm does not admit a natural use of PDS kernels. Thus, here, we assume that the input space $\mathcal{X}$ is a subset of $\mathbb{R}^N$ and consider a family of linear hypotheses $\mathcal{H} = \{x \mapsto \mathbf{w} \cdot \mathbf{x} + b \colon \mathbf{w} \in \mathbb{R}^N, b \in \mathbb{R}\}$.

Let $S = \big((\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\big) \in (\mathcal{X} \times \mathcal{Y})^m$ be a labeled training sample. Lasso is based on the minimization of the empirical squared error on S with a regularization term depending on the norm of the weight vector, as in the case of the ridge regression, but using the L_1 norm instead of the L_2 norm and without squaring the