CS281B/Stat241B (Spring 2008) Statistical Learning Theory **Lecture: 10**

Kernel ridge regression, Gaussian processes, and ensemble methods

Lecturer: Peter Bartlett *Scribe: Kevin Canini*

1 Loss & maximum likelihood

A classification or regression problem is typically formulated with a cost term of the form:

$$\frac{1}{n}\sum \phi(y_i, f(x_i)) + \text{penalty}(f)$$

where $\phi(\cdot,\cdot)$ is the estimation error and penalty(f) is the regularization term.

For certain loss functions, we can interpret minimizing this expression as maximizing the probability of the data. This viewpoint is not useful, although, for the hinge loss function. In that case, because ϕ is not differentiable, the minimizer of $\mathbb{E}\left[\phi(Y, f(X))|X\right]$ is not an invertible function of the conditional probability $P(Y = 1|X)$.

2 Kernel ridge regression

Ridge regression adds a regularization penalty (scaled by λ) to the cost term, as follows:

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2 + \lambda\|f\|_{\mathcal{H}}^2$$

As alluded to earlier, minimizing the empirical risk of a data set, using the above cost term, is equivalent to maximizing the log-likelihood of the data for a certain probability model and loss function.

If we model $Y = f(X) + Z$, where Z is additive noise, the kernel regression formulation is

$$\begin{aligned} &\min \lambda\|w\|^2 + \sum \xi_i^2 \\ &\text{s.t. } \xi_i = y_i - \langle w, x_i\rangle \end{aligned}$$

Computing the Lagrangian and using calculus to minimize over w and ξ, gives

$$\begin{aligned} w &= \frac{1}{2\lambda}\sum \alpha_i x_i \\ \xi &= \frac{\alpha_i}{2} \end{aligned}$$

and hence the solution to the dual is

$$\alpha = 2\lambda(K + \lambda I)^{-1} y$$

with

$$\begin{aligned} K_{ij} &= \langle x_i, x_j \rangle \\ y &= (y_1, \ldots, y_n)' \end{aligned}$$

Thus, the solution is

$$f(x) = y'(K + \lambda I)^{-1} k,$$

where $k = (k(x_1, x), \ldots, k(x_n, x))'$ is the vector of inner products between the data and the new point, x.

3 Bayesian viewpoint: Gaussian processes

There is a Gaussian process interpretation of kernel ridge regression. We first define a prior on the regression function f. Suppose that f is drawn from a Gaussian process, such that for all n and all $x_1, \ldots, x_n \in \mathcal{X}$, there is a matrix $\Sigma \in \Re^{n \times n}$ and $(f(x_1), \ldots, f(x_n))' \sim N(0, \Sigma)$. The entry $\Sigma_{i,j}$ of the matrix Σ specifies the covariance between $f(x_i)$ and $f(x_j)$.

Consider a model $y = f(x) + \xi$ with $\xi \sim N(0, \sigma^2)$, and suppose that we observe data $(x_1, y_1), \ldots, (x_n, y_n)$ and we wish to predict $f(x_0)$, where x_0 is a new test point. Consider the posterior distribution of

$$(f(x_0), f(x_1), \ldots, f(x_n))$$

given this data. It is

$$\mathbb{P}\left((f(x_0), \ldots, f(x_n))' | x_0, x_1, \ldots, x_n, y_1, \ldots, y_n\right) \propto \mathbb{P}\left(y|t\right) \mathbb{P}\left(t_0, t | x_0, x_1, \ldots, x_n\right)$$

$$\begin{aligned} \mathbb{P}\left(y|t\right) &\propto e^{-\frac{1}{2\sigma^2}\|y-t\|^2} \\ \mathbb{P}\left(t_0, t | x_0, x_1, \ldots, x_n\right) &\propto e^{-\frac{1}{2}(t_0, t')\Sigma^{-1}(t_0, t)'} \end{aligned}$$

where

1. Σ is the prior covariance of $(f(x_0), f(x_1), \ldots, f(x_n))'$,
2. $t = (f(x_1), \ldots, f(x_n))'$, and
3. $t_0 = f(x_0)$.

It is an easy calculation to see that the posterior mean of $t_0 = f(x_0)$ is $y'(\Sigma + \sigma^2 I)^{-1} k$, where k is the first column of Σ. Notice that we can interpret the Gram matrix K in kernel ridge regression as the covariance of a Gaussian process prior.

4 Ensemble methods

In pattern classification problems, we use ensemble methods to form a "committee" of classifiers, using some sort of voting schemes. The hope is that even though any single classifier might not perform well, the ensemble performs better.

For example, if $f_i : \mathcal{X} \to \{\pm 1\}$, we can take a majority vote among $f_1(x), \ldots, f_M(x)$ to determine $f(x)$.

The underlying functions can be many things, e.g.

- linear threshold functions: $\sum \alpha_i f_i(x) = \sum \alpha_i \text{ sign}(w_i' x)$
- decision trees
- decision stumps: a decision tree with a single test, e.g., $y = \mathbf{1}[x_7 \geq 3]$
- a dictionary of simple functions

4.1 AdaBoost (Freund-Schapire '95)

We start with a uniform distribution over the n data points:

$$D_1(i) = \frac{1}{n} \text{ for } i = 1, \ldots, n$$

and the function $F_0(x) = 0$.

We go through a specified number of iterations; for each $t \in \{1, \ldots, T\}$, choose $f_t \in G$ to (approximately) minimize $\epsilon_t = \sum_{i=1}^n D_t(i)\mathbf{1}[f_t(x_i) \neq y_1]$. Then make the following updates:

$$F_t = F_{t-1} + \alpha_t f_t$$

$$D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} e^{\alpha_t} & \text{if } f_t(x_i) \neq y_i \\ e^{-\alpha_t} & \text{otherwise} \end{cases}$$

Here,

$$\alpha_t = \frac{1}{2} \log \frac{1 - \epsilon_t}{\epsilon_t}$$

$$Z_t = 2\sqrt{\epsilon_t(1 - \epsilon_t)}$$

AdaBoost will overfit if you let T get very large, e.g., as big as n^2. However, T any smaller than linear in n will not overfit in a precise sense (for a suitably rich class of base classifiers, AdaBoost with $T \to \infty$ slower than linearly in n will be *universally consistent*, that is, the risk of the classifier it produces will approach the Bayes risk).

4.1.1 Theorem:

The empirical probability

$$\begin{aligned} \hat{P}(YF_T(X) \leq 0) = \frac{1}{n}|\{i : y_i F_t(x_i) \leq 0\}| &\leq \prod_{t=1}^T Z_t \\ &= \prod_{t=1}^T 2\sqrt{\epsilon_t(1 - \epsilon_t)} \end{aligned}$$

Furthermore, if $\epsilon_t \leq \frac{1}{2} - \gamma$ for all t, then

$$\begin{aligned} \prod_{i=1}^T Z_t &\leq \prod_{i=1}^T 2\sqrt{\frac{1}{2^2} - \gamma^2} \\ &= (1 - 4\gamma^2)^{T/2} \\ &\leq \delta, \text{ for } T \geq \frac{\ln 1/\delta}{2\gamma^2} \end{aligned}$$