Lecture Notes 26
36-705

Today we will discuss nonparametric density estimation and nonparametric regression. First, we need to define kernels.

1 Kernels

A kernel function $K(x)$ for $x \in \mathbb{R}$ is a function K such that $\int K(x)dx = 1$ and K is symmetric, i.e. $\int xK(x)dx = 0$. We will also assume that $K(x) \geq 0$ and $\int x^2K(x)dx$. Examples are: the Gaussian kernel

$$K(x) = \frac{1}{\sqrt{2\pi}}e^{-x^2/2}$$

the boxcar kernel

$$K(x) = I(|x| < 1/2)$$

and the Epanechnikov

$$K(x) = \frac{3}{4}(1-x^2)I(|x| < 1).$$

Given a kernel K and a number $h > 0$ called the bandwidth, we define

$$K_h(x) = \frac{1}{h}K\left(\frac{x}{h}\right).$$

Similarly, for $x \in \ mathbbR^d$ we define $K : \mathbb{R}^d \to \mathbb{R}$ where K is symmetric and integrates to 1. Then, given a symmetric positive definite bandwidth matrix H we define

$$K_H(x) = \frac{1}{|H|}K(H^{-1}x).$$

A common choice is to take $H = hI$ and

$$K_H(x) = \frac{1}{h^d}\prod_j K\left(\frac{x_j}{h}\right)$$

where K is a one-dimensional kernel.

2 Non-parametric Density Estimation

Let $Y_1, \ldots, Y_n \sim p$. We'll focus on the case $Y_i \in \mathbb{R}$. We want to estimate p nonparametrically. A common estimator is the kernel density estimator defined by

$$\widehat{p}(y) = \frac{1}{n}\sum_i K_h(Y_i - y)$$

where K_h is a kernel with bandwidth h. You should think of $h = h_n$ as a number that decreases with sample size. We will assume that $p''(y) < \infty$.

Let's analyze this estimator. First

$$\begin{aligned}
\mathbb{E}[\widehat{p}(y)] &= \int K_h(u-y)p(u)du = \int K(t)p(y+th)dt \\
&\approx \int K(t)\left[p(y) + thp'(y) + \frac{t^2h^2}{2}p''(y)\right]dt \\
&= p(y) + hp'(y)\int tK(t)dt + \frac{h^2p''(y)}{2}\int t^2K(t)dt \\
&= p(y) + c_1(y)h^2
\end{aligned}$$

where $c_1(y) = p''(y)\int t^2K(t)dt/2$. So the bias is $c_1(y)h^2$.

Now we find the variance. We have

$$\text{Var}[\widehat{p}(y)] = \frac{1}{n}\text{Var}[K_h(Y-y)] = \frac{1}{n}\mathbb{E}[K_h^2(Y-y)] - \frac{1}{n}(\mathbb{E}[K_h(Y-y)])^2.$$

Now

$$\begin{aligned}
\mathbb{E}[K_h^2(Y-y)] &= \int \frac{1}{h^2}K^2((u-y)/h)p(u)du = \frac{1}{h}\int K^2(t)p(y+th)dt \\
&= \frac{1}{h}\int K^2(t)[p(y) + thp'(y) + \cdots]dt \approx \frac{c_2p(y)}{h}
\end{aligned}$$

where $c_2 = \int t^2K(t)dt$. Next

$$(\mathbb{E}[K_h(Y-y)])^2 \approx (p(y) + c_1(y)h^2)^2 \approx p^2(y).$$

So

$$\text{Var}[\widehat{p}(y)] \approx \frac{c_2np(y)}{nh} + \frac{p(y)}{n} \approx \frac{p(y)}{nh}.$$

Note that, if $h \to 0$ and $nh \to \infty$ as $n \to \infty$ then the bias and variance go to 0 and hence $\widehat{p}(y) \xrightarrow{P} p(y)$.

Next, consider the *integrated mean squared error* IMSE:

$$\text{IMSE} = \mathbb{E}[\int (\widehat{p}(y) - p(y))^2 dy] = \int \mathbb{E}[(\widehat{p}(y) - p(y))^2] dy$$
$$= \int \left(c_1^2(y) h^4 + \frac{c_2 p(y)}{nh} \right) dy = c_1 h^4 + \frac{c_2}{nh}$$

where $c_1 = \int c^2(y) dy$.

Here we say the bias-variance tradeoff. As h increases, the bias increases and the variance decreases and vice versa. The IMSE is minimized by choosing

$$h_n = \left(\frac{c_2}{4 c_1 n} \right)^{1/5} \approx \left(\frac{1}{n} \right)^{1/5}.$$

With this choice, we see that

$$\text{IMSE} = O\left(\frac{1}{n} \right)^{4/5}.$$

In practice, h is usually chosen by a version of cross-validation. In d dimensions it turns out that the IMSE is $O(n^{-4/(4+d)})$. The effect of dimension is brutal and is called the curse of dimensionality.

3 Non-parametric Regression

We observe $(X_1, Y_1), \ldots, (X_n, Y_n) \sim P$ and our goal is to estimate the regression function

$$r(x) = \mathbb{E}[Y | X = x].$$

We *integrated* squared loss

$$L(\widehat{r}, r) = \int (\widehat{r}(x) - r(x))^2 dx.$$

The risk is then

$$R(\widehat{r}, r) = \mathbb{E}\left(\int (\widehat{r}(x) - r(x))^2 dx \right).$$

We will assume that $r''(y) < \infty$.

As in the case of point estimation we have a bias variance decomposition. First we define the point-wise bias:

$$b(x) = \mathbb{E}(\widehat{r}(x)) - r(x),$$

and the point-wise variance:

$$v(x) = \mathbb{E}\Big(\widehat{r}(x) - \mathbb{E}(\widehat{r}(x))\Big)^2.$$

Now, as before we can verify that:

$$R(\widehat{r}, r) = \int b^2(x)dx + \int v(x)dx.$$

A natural strategy in non-parametric regression is to locally average the data, i.e. our estimate of the regression function at any point will be the average of the Y values in a small neighborhood of the point.

The width of this neighborhood will determine the bias and variance. Too large a neighborhood will result in high bias and low variance (this is called oversmoothing) and too small a neighborhood will result in low bias but large variance (this is known as undersmoothing).

4 Optimal Regression Function

Suppose we knew the joint distribution over (X, Y). One could alternatively begin by defining the risk of an estimate $\widehat{r}$ as

$$R(\widehat{r}) = \mathbb{E}(Y - \widehat{r}(X))^2.$$

This risk simply measures the prediction error, i.e. the expected error we make in predicting Y when we use the function $\widehat{r}(X)$. This risk is minimized by the conditional expectation, i.e. we have the following theorem.

Theorem 1 *The risk R is minimized by*

$$r(x) = \mathbb{E}(Y|X = x).$$

Proof: Let $g(x)$ be any function of x. Then

$$\begin{aligned}
R(g) &= \mathbb{E}(Y - g(X))^2 = \mathbb{E}(Y - r(X) + r(X) - g(X))^2 \\
&= \mathbb{E}(Y - r(X))^2 + \mathbb{E}(r(X) - g(X))^2 + 2\mathbb{E}((Y - r(X))(r(X) - g(X))) \\
&\geq \mathbb{E}(Y - r(X))^2 + 2\mathbb{E}((Y - r(X))(r(X) - g(X))) \\
&= \mathbb{E}(Y - r(X))^2 + 2\mathbb{E}\mathbb{E}\left((Y - r(X))(r(X) - g(X)) \;\middle|\; X\right) \\
&= \mathbb{E}(Y - r(X))^2 + 2\mathbb{E}\Big((\mathbb{E}(Y|X) - r(X))(r(X) - g(X))\Big) \\
&= \mathbb{E}(Y - r(X))^2 + 2\mathbb{E}\Big((r(X) - r(X))(r(X) - g(X))\Big) \\
&= \mathbb{E}(Y - r(X))^2 = R(r).
\end{aligned}$$

■

5 Kernel Regression

One of the most basic ways of doing non-parametric regression is called kernel regression. We will analyze kernel regression when we only have one covariate. The general case is not very different. The estimator is defined as:

$$\widehat{r}(x) = \sum_{i=1}^{n} w_i(x) Y_i,$$

where the weights assign more importance to points near x. This is called a kernel regressor when the weights are chosen according to a kernel, i.e. we have weights:

$$w_i(x) = \frac{K\left(\frac{x-X_i}{h}\right)}{\sum_{i=1}^{n} K\left(\frac{x-X_i}{h}\right)} = \frac{K_h(X_i - x)}{\sum_j K_h(X_i - x)}$$

where, as before, the bandwidth h controls the amount of smoothing.

To analyze this estimator, note that we can write

$$Y_i = r(X_i) + \epsilon_i$$

where ϵ_i has mean 0. Now

$$\begin{aligned}
\widehat{r}(x) &= \frac{\sum_i Y_i K_h(X_i - x)}{\sum_i K_h(X_i - x)} = \frac{\frac{1}{n}\sum_i Y_i K_h(X_i - x)}{\frac{1}{n}\sum_i K_h(X_i - x)} \\
&= \frac{\frac{1}{n}\sum_i Y_i K_h(X_i - x)}{\widehat{p}(x)} = \frac{\frac{1}{n}\sum_i Y_i K_h(X_i - x)}{p(x) + o_P(1)} \\
&\approx \frac{\frac{1}{n}\sum_i Y_i K_h(X_i - x)}{p(x)}.
\end{aligned}$$

Let's find the mean and variance of the numerator. We have

$$\begin{aligned}
\mathbb{E}[Y K_h(X - x)] &= \int\int y K_h(u - x) p(x, y) du dy = \int K_h(u - x) \int y p(y|u) dy p(u) du \\
&= \int K_h(u - x) r(u) p(u) du = \int K(t) r(x + th) p(x + th) dt \\
&\approx \int K(t) \left[r(x) + thr'(x) + \frac{t^2 h^2}{2} r''(x) \right] \left[p(x) + thp'(x) + \frac{t^2 h^2}{2} p''(x) \right] dt \\
&= r(x)p(x) + \frac{ch^2}{2}[r(x)p''(x) + 2r'(x)p'(x) + r''(x)p(x)]
\end{aligned}$$

where $c = \int t^2 K(t)$. Hence,

$$\mathbb{E}[\widehat{r}(x)] = r(x) + Ch^2.$$

By a similar calculation

$$\text{Var}[\widehat{r}(x)] = \frac{C}{nh}.$$

We conclude that

$$\text{IMSE} = ch^4 + \frac{c}{nh}$$

where now we use c generically to define constants. As in density estimation, the best bandwidth is $h_n \asymp n^{-1/5}$ and the risk is $n^{-4/5}$.

The analysis reveals that the bias depends on $p'(x)$ and $p(x)$. These terms can be removed by using better estimators.

6 The general case

So far we assumes that $r''(y) < \infty$. More generally, suppose that the β^{th} derivative of $r(x)$ is bounded, and we are in d-dimensions. In this case the bias will be roughly:

$$b^2(x) \approx h^{2\beta},$$

and the variance:

$$v(x) \approx \frac{1}{nh^d},$$

and balancing these will lead to the rate of convergence:

$$R(\widehat{r}, r) \approx n^{-2\beta/(2\beta+d)}.$$

This reveals another crucial feature of non-parametrics. In linear regression, the rate of convergence is typically something like:

$$R(\widehat{\beta}, \beta) \approx \frac{d}{n}.$$

In both cases, the situation gets worse as d increases, however in non-parametrics the situation gets *exponentially* worse. This is often colloquially referred to as the *curse of dimensionality.*

7 RKHS regression

There is another method also referred to as kernel regression. More precisely, it is Reproducing Kernel Hilbert Space (RKHS) regression. We will not cover this in much detail but here is te general idea.

A symmetric bivariate function $K : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is positive semidefinite (PSD) if for all integers $n \geq 1$ and elements $\{x_i\}_{i=1}^n$ where each $x_i \in \mathcal{X}$, the $n \times n$ matrix K with elements $K_{ij} := K(x_i, x_j)$ is positive semidefinite.

Here are a few standard examples:

1. Linear kernel: When $\mathcal{X} = \mathbb{R}^d$ then $K(x_i, x_j) = \langle x_i, x_j \rangle = \sum_{u=1}^d x_{iu} x_{ju}$, is the linear kernel and is PSD.
2. Polynomial kernel: Again when $\mathcal{X} = \mathbb{R}^d$ then $K(x_i, x_j) = (\langle x_i, x_j \rangle)^m$, is the homogenous polynomial kernel of degree $m \geq 2$. This kernel is also PSD. The inhomogenous polynomial kernel $K(x_i, x_j) = (1 + \langle x_i, x_j \rangle)^m$ is also PSD.
3. Gaussian kernel: Perhaps the most popular kernel in machine learning is the Gaussian kernel. Here we take $K(x_i, x_j) = \exp(-\|x_i - x_j\|_2^2/(2\sigma^2))$.

Given dataa $\{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ we are going to estimate the function r by a function r_α which we will assume has the form:

$$r_\alpha(x) = \sum_{i=1}^n \alpha_i K(x_i, x),$$

where we need to estimate the α_i's. To do this we will minimize a least-squares type objective:

$$\widehat{\alpha} = \arg\min_\alpha \frac{1}{2} \sum_{i=1}^n (Y_i - r_\alpha(X_i))^2 + \lambda \text{Pen}(r_\alpha).$$

The penalty we will use is something called an RKHS norm penalization and it takes the form:

$$\text{Pen}(r_\alpha) = \alpha^T K \alpha,$$

where K is the gram matrix, i.e. $K_{ij} = K(x_i, x_j)$. This penalty encourages the function to be smooth but this is not easy to see without going into more detail. Observe that we can write:

$$\begin{bmatrix} r_\alpha(X_1) \\ r_\alpha(X_2) \\ \vdots \\ r_\alpha(X_n) \end{bmatrix} = K\alpha,$$

so the RKHS regression objective simplifies to:

$$\widehat{\alpha} = \arg\min_{\alpha} \frac{1}{2}\|Y - K\alpha\|_2^2 + \lambda \alpha^T K \alpha,$$

which we can solve in closed form (just by taking derivatives and setting to zero) as:

$$\widehat{\alpha} = (K + \lambda I)^{-1} Y.$$

Our estimated regression function then takes the form:

$$r_{\widehat{\alpha}}(x) = \sum_{i=1}^{n} \widehat{\alpha}_i K(X_i, x).$$

Superficially there are similarities between RKHS regression and kernel regression. They both produce a function whose value at a given point is a weighted combination of the Y_i values at other points. In kernel regression the weights are easy to interpret, while in RKHS regression the weights are the solution to a least squares problem and are not directly interpretable.

From a practical standpoint, RKHS regression typically has two tuning parameters: the penalty parameter λ and usually some RKHS parameter (for instance the RKHS kernel bandwidth for a Gaussian kernel).

There are two types of problems one could ponder: (1) we want to fit a function that is Lipschitz or Holder smooth (as we analyzed in the first half): in this case, it is perhaps natural to use kernel regression and somewhat more artificial to use RKHS regression (2) we want to fit a function in a particular RKHS, in this case it is perhaps more natural to use RKHS regression.

From a theoretical standpoint, RKHS regression is usually analyzed using variants of the Rademacher complexity results we saw earlier in the course, i.e. they are not directly analyzed in terms of the bias and variance (this is because the RKHS regression procedure is naturally viewed as ERM over an RKHS). This means that the rates of convergence are typically specified in terms of properties of the RKHS and the data-generating distribution, i.e. a typical measure of complexity of an RKHS is the decay-rate of eigenvalues of the kernel gram matrix. This is quite unlike kernel regression where the function class is something simple (Lipschitz functions), and the measure of complexity is just the smoothness of the function.

RKHS regression is not the only alternative to kernel regression. Often you will see methods like k-NN regression (where you predict at a point by averaging the y values of the k-closest points), local polynomial regression (where you chop up the domain and fit (low-degree) polynomials in each piece of the domain) and orthogonal series estimators or projection estimators (where you expand the regression function in a orthogonal basis – say of sine/cosine type functions – and then estimate the coefficients in this basis).