Kernel Regression

Ethan Ligon

March 30, 2022

Introduction

We return to the problem which motivates us. We're interested in estimating things like:

- $\mathbb{E}(y|X = x)$ (Conditional expectations)
- $f(y|x)$ (Conditional pdf)
- Extends to estimating any (smooth?) function.

Note

$$\mathbb{E}(y|X = x) = \int y f(y|x) dy,$$

So if we can estimate $f(y|x)$ we can compute expectations. In previous lecture, we discussed methods for estimating *unconditional* densities $f(y)$. Today we return to the conditional case.

Nonparametric Regression

The basic non-parametric regression model can be written in the form

$$y = m(X) + \epsilon$$
$$\mathbb{E}(\epsilon|X) = 0$$
$$\mathbb{E}(\epsilon^2|X) = \sigma^2(X).$$

The idea is to exploit the conditional moment restriction to estimate m and perhaps σ^2.

Additional Assumptions

- $m(x)$ continuous;
- Marginal density $f(x)$ continuous. (If X is discrete, taking just a few different values, then just compare, e.g., $f(y|x_1)$ with $f(y|x_2)$).

Kernel Regression

There are a variety of approaches to estimating $m(x)$ and $\sigma^2(x)$; today we'll focus on *kernel regression*.

Kernel

As with KDE, we start with a kernel, which must integrate to one. But there are other desirable properties:

Non-negativity $k(u) \geq 0$ for all u. (In this case we can interpret k as a probability density function.)

Boundedness $\int |u|^r k(u) du < \infty$ for all positive integers r.

Symmetry $k(u) = k(-u)$. (Note that boundedness & symmetry imply $\int uk(u)du = 0$.)

Normalized $\int u^2 k(u) du = 1$

Differentiable $\hat{f}$ will inherit differentiability of kernel, and often one prefers "smooth" estimates.

Kernel Regression Estimator

Since $\mathbb{E}(u|X) = 0$ and kernel is bounded, we have

$$\mathbb{E}\left(k\left(\frac{X-x}{h}\right)u\right) = 0,$$

where $h > 0$ is a "bandwidth" parameter. Then working with the basic non-parametric regression model, we have

$$k\left(\frac{X-x}{h}\right)y = k\left(\frac{X-x}{h}\right)m(x) + k\left(\frac{X-x}{h}\right)u,$$

and

$$\mathbb{E}k\left(\frac{X-x}{h}\right)y = \mathbb{E}k\left(\frac{X-x}{h}\right)m(x),$$

so that

$$m(x) = \frac{\mathbb{E}k\left(\frac{X-x}{h}\right)y}{\mathbb{E}k\left(\frac{X-x}{h}\right)}.$$

Kernel Regression Estimator

Now, let $\{(X_i, y_i)\}$ be a random sample of n observations. Then from

$$m(x) = \frac{\mathbb{E}k\left(\frac{X-x}{h}\right)y}{\mathbb{E}k\left(\frac{X-x}{h}\right)}$$

applying the analogy principle we obtain

$$\hat{m}(x) = \frac{\sum_i k_i(x)y_i}{\sum_i k_i(x)},$$

where $k_i(x)$ is a shorthand for $k\left(\frac{X_i-x}{h}\right)$. This is the *kernel regression estimator*.

Residuals

The MSE or IMSE is not in general a feasible way to evaluate the estimator, but we can compute the nonparametric residuals:

$$e_i = y_i - \hat{m}(x_i)$$

Squaring this gives an estimator of MSE(x_i), while we can construct an estimator of the *expected* MSE using

$$\widehat{\text{EMSE}} = \frac{1}{n}\sum_{i=1}^{n} e_i^2.$$

(Question: Why is this a reasonable way to estimate an integral?)

"Overfitting"

A problem with this is that the estimator is specifically designed to fit at exactly the sample points, so the IMSE estimated this way can be be expected to be *smaller* than at other points. Note that this problem gets worse as $h \to \infty$.

Leave-one-out (cross-validation) estimator

A standard solution to this problem is based on the old idea of the "jack-knife", which involves calculating $\hat{m}_{-j}(x)$ which *leaves out* the jth observation in estimation:

$$\hat{m}_{-j}(x) = \frac{\sum_{i \neq j} k_i(x) y_i}{\sum_{i \neq j} k_i(x)}.$$

This gives us corresponding residuals

$$e_{-i} = y_i - \hat{m}_{-i}(x_i).$$

Since $\hat{m}_{-i}$ is not a function of (y_i, x_i) this eliminates the problem of overfitting, and we can estimate the IMSE as

$$\widehat{\mathsf{EMSE}} = \frac{1}{n} \sum_{i=1}^{n} e_{-i}^2.$$

Computation

Doing this directly would be very expensive! We'd have to compute n estimates. We can make things much simpler by noticing an important facts:

The Kernel Trick

For estimating the EMSE we only care about evaluating $\hat{m}$ at the points where we have data. This means that we can turn the problem of calculating the EMSE from a problem involving sums of functions into a problem that just relies on matrices of real numbers. The key matrix is called the "Gram" or kernel matrix:

$$\boldsymbol{G}(h) = \left[k\left(\frac{x_i - x_j}{h} \right) \right].$$

Note that $\boldsymbol{G}$ is $n \times n$, symmetric, and has diagonal elements all given by $k(0)$ (which don't depend on the bandwidth).

Estimation using the Gram matrix

With the Gram matrix in hand we can re-write the kernel regression estimator evaluated *at the data* $\boldsymbol{x}$:

$$\underset{n\times 1}{\hat{m}(\boldsymbol{x})} = \frac{\boldsymbol{G}\boldsymbol{y}}{\boldsymbol{G}\ell_n},$$

where ℓ_n is a column vector of ones.

Leave one out

Let $\boldsymbol{G}_- = \boldsymbol{G} - \mathrm{diag}\boldsymbol{G}$. Then the n-vector of "leave-one-out" estimators is

$$\underset{n\times 1}{\hat{m}_-(\boldsymbol{x})} = \frac{G_-\boldsymbol{y}}{G_-\ell_n},$$

and "leave-one-out residuals" are simply

$$\boldsymbol{e}_- = \boldsymbol{y} - \hat{m}_-(\boldsymbol{x}).$$

EMSE, Bias, Variance

Once we have e_- we have very simple estimators for sample bias and variance of the estimator:

Bias	$\mathbb{E}\epsilon = \mathbb{E}\frac{1}{n}\sum_{i=1}^n e_{-i}$
Variance	$\text{Var}(\epsilon) = \mathbb{E}\frac{1}{n}\sum_{i=1}^n e_{-i}^2 - \mathbb{E}\left(\frac{1}{n}\sum_{i=1}^n e_{-i}\right)^2$
EMSE	$\text{Bias}^2 + \text{Variance}$

Bandwidth selection

With a feasible estimator for the IMSE our problem of bandwidth selection can be addressed by finding the value of h that minimizes $\widehat{EMSE}$.