STAT 425: Introduction to Nonparametric Statistics **Winter 2018**

Lecture 9: Regression: Regressogram and Kernel Regression

Instructor: Yen-Chi Chen

Reference: Chapter 5 of *All of nonparametric statistics*.

9.1 Introduction

Let $(X_1, Y_1), \cdots, (X_n, Y_n)$ be a bivariate random sample. In the regression analysis, we are often interested in the regression function

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

Sometimes, we will write

$$Y_i = m(X_i) + \epsilon_i,$$

where ϵ_i is a mean 0 noise. The simple linear regression model is to assume that $m(x) = \beta_0 + \beta_1 x$, where β_0 and β_1 are the intercept and slope parameter. In this lecture, we will talk about methods that direct estimate the regression function $m(x)$ without imposing any parametric form of $m(x)$.

9.2 Regressogram (Binning)

We start with a very simple but extremely popular method. This method is called regressogram but people often call it binning approach. You can view it as

$$\text{regressogram} = \text{regression} + \text{histogram}.$$

For simplicity, we assume that the covariates X_i's are from a distribution over $[0, 1]$.

Similar to the histogram, we first choose M, the number of bins. Then we partition the interval $[0, 1]$ into M equal-width bins:

$$B_1 = \left[0, \frac{1}{M}\right), B_2 = \left[\frac{1}{M}, \frac{2}{M}\right), \cdots, B_{M-1} = \left[\frac{M-2}{M}, \frac{M-1}{M}\right), B_M = \left[\frac{M-1}{M}, 1\right].$$

When $x \in B_\ell$, we estimate $m(x)$ by

$$\widehat{m}_M(x) = \frac{\sum_{i=1}^n Y_i I(X_i \in B_\ell)}{\sum_{i=1}^n I(X_i \in B_\ell)} = \text{average of the responses whose covariates is in the same bin as } x.$$

Bias. The bias of a regressogram estimator is

$$\textbf{bias}(\widehat{m}_M(x)) = O\left(\frac{1}{M}\right).$$

Variance. The variation of a regressogram estimator is

$$\mathsf{Var}(\widehat{m}_M(x)) = O\left(\frac{M}{n}\right).$$

Therefore, the MSE and MISE will be at rate

$$\textbf{MSE} = O\left(\frac{1}{M^2}\right) + O\left(\frac{M}{n}\right), \quad \textbf{MISE} = O\left(\frac{1}{M^2}\right) + O\left(\frac{M}{n}\right),$$

leading to the optimal number of bins $M^* \asymp n^{1/3}$ and the optimal convergence rate $O(n^{-2/3})$, the same as the histogram.

Similar to the histogram, the regressogram has a slower convergence rate compared to many other competitors (we will introduce several other candidates). However, they (histogram and regressogram) are still very popular because the construction of an estimator is very simple and intuitive; practitioners with little mathematical training can easily master these approaches.

9.3 Kernel Regression

Given a point x_0, assume that we are interested in the value $m(x_0)$. Here is a simple method to estimate that value. When $m(x_0)$ is smooth, an observation $X_i \approx x_0$ implies $m(X_i) \approx m(x_0)$. Thus, the response value $Y_i = m(X_i) + \epsilon_i \approx m(x_0) + \epsilon_i$. Using this observation, to reduce the noise ϵ_i, we can use the sample average. Thus, an estimator of $m(x_0)$ is to take the average of those responses whose covariate are close to x_0.

To make it more concrete, let $h > 0$ be a threshold. The above procedure suggests to use

$$\widehat{m}_{\mathsf{loc}}(x_0) = \frac{\sum_{i:|X_i - x_0| \le h} Y_i}{n_h(x_0)} = \frac{\sum_{i=1}^n Y_i I(|X_i - x_0| \le h)}{\sum_{i=1}^n I(|X_i - x_0| \le h)}, \tag{9.1}$$

where $n_h(x_0)$ is the number of observations where the covariate $X : |X_i - x_0| \le h$. This estimator, $\widehat{m}_{\mathsf{loc}}$, is called the *local average* estimator. Indeed, to estimate $m(x)$ at any given point x, we are using a local average as an estimator.

The local average estimator can be rewritten as

$$\widehat{m}_{\mathsf{loc}}(x_0) = \frac{\sum_{i=1}^n Y_i I(|X_i - x_0| \le h)}{\sum_{i=1}^n I(|X_i - x_0| \le h)} = \sum_{i=1}^n \frac{I(|X_i - x_0| \le h)}{\sum_{\ell=1}^n I(|X_\ell - x_0| \le h)} \cdot Y_i = \sum_{i=1}^n W_i(x_0) Y_i, \tag{9.2}$$

where

$$W_i(x_0) = \frac{I(|X_i - x_0| \le h)}{\sum_{\ell=1}^n I(|X_\ell - x_0| \le h)} \tag{9.3}$$

is a weight for each observation. Note that $\sum_{i=1}^n W_i(x_0) = 1$ and $W_i(x_0) > 0$ for all $i = 1, \cdots, n$; this implies that $W_i(x_0)$'s are indeed weights. Equation (9.2) shows that the local average estimator can be written as a *weighted average* estimator so the i-th weight $W_i(x_0)$ determines the contribution of response Y_i to the estimator $\widehat{m}_{\mathsf{loc}}(x_0)$.

In constructing the local average estimator, we are placing a hard-thresholding on the neighboring points– those within a distance h are given equal weight but those outside the threshold h will be ignored completely. This hard-thresholding leads to an estimator that is not continuous.

To avoid problem, we consider another construction of the weights. Ideally, we want to give more weights to those observations that are close to x_0 and we want to have a weight that is 'smooth'. The Gaussian function $G(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$ seems to be a good candidate. We now use the Gaussian function to construct an estimator. We first construct the weight

$$W_i^G(x_0) = \frac{G\left(\frac{x_0 - X_i}{h}\right)}{\sum_{\ell=1}^n G\left(\frac{x_0 - X_\ell}{h}\right)}.$$

The quantity $h > 0$ is the similar quantity to the threshold in the local average but now it acts as the *smoothing bandwidth* of the Gaussian. After constructing the weight, our new estimator is

$$\widehat{m}_{\mathsf{G}}(x_0) = \sum_{i=1}^{n} W_i^G(x_0) Y_i = \sum_{i=1}^{n} \frac{G\left(\frac{x_0 - X_i}{h}\right)}{\sum_{\ell=1}^{n} G\left(\frac{x_0 - X_\ell}{h}\right)} Y_i = \frac{\sum_{i=1}^{n} Y_i G\left(\frac{x_0 - X_i}{h}\right)}{\sum_{\ell=1}^{n} G\left(\frac{x_0 - X_\ell}{h}\right)}. \tag{9.4}$$

This new estimator has a weight that changes more smoothly than the local average and is smooth as we desire.

Observing from equation (9.1) and (9.4), one may notice that these *local* estimators are all of a similar form:

$$\widehat{m}_h(x_0) = \frac{\sum_{i=1}^{n} Y_i K\left(\frac{x_0 - X_i}{h}\right)}{\sum_{\ell=1}^{n} K\left(\frac{x_0 - X_\ell}{h}\right)} = \sum_{i=1}^{n} W_i^K(x_0) Y_i, \quad W_i^K(x_0) = \frac{K\left(\frac{x_0 - X_i}{h}\right)}{\sum_{\ell=1}^{n} K\left(\frac{x_0 - X_\ell}{h}\right)}, \tag{9.5}$$

where K is some function. When K is a Gaussian, we obtain estimator (9.4); when K is a uniform over $[-1, 1]$, we obtain the local average (9.1). The estimator in equation (9.5) is called the *kernel regression* estimator or Nadaraya-Watson estimator[1]. The function K plays a similar role as the kernel function in the KDE and thus it is also called the *kernel function*. And the quantity $h > 0$ is similar to the smoothing bandwidth in the KDE so it is also called the smoothing bandwidth.

9.3.1 Theory

Now we study some statistical properties of the estimator $\widehat{m}_h$. We skip the details of derivations[2].

Bias. The bias of the kernel regression at a point x is

$$\mathbf{bias}(\widehat{m}_h(x)) = \frac{h^2}{2} \mu_K \left(m''(x) + 2 \frac{m'(x) p'(x)}{p(x)} \right) + o(h^2),$$

where $p(x)$ is the probability density function of the covariates $X_1, \cdots, X_n$ and $\mu_K = \int x^2 K(x) dx$ is the same constant of the kernel function as in the KDE.

The bias has two components: a curvature component $m''(x)$ and a *design* component $\frac{m'(x)p'(x)}{p(x)}$. The curvature component is similar to the one in the KDE; when the regression function curved a lot, kernel smoothing will smooth out the structure, introducing some bias. The second component, also known as the *design bias*, is a new component compare to the bias in the KDE. This component depends on the density of covariate $p(x)$. Note that in some studies, we can choose the values of covariates so the density $p(x)$ is also called the *design* (this is why it is known as the design bias).

Variance. The variance of the estimator is

$$\mathsf{Var}(\widehat{m}_h(x)) = \frac{\sigma^2 \cdot \sigma_K^2}{p(x)} \cdot \frac{1}{nh} + o\left(\frac{1}{nh}\right),$$

where $\sigma^2 = \mathsf{Var}(\epsilon_i)$ is the error of the regression model and $\sigma_K^2 = \int K^2(x) dx$ is a constant of the kernel function (the same as in the KDE). This expression tells us possible sources of variance. First, the variance increases when σ^2 increases. This makes perfect sense because σ^2 is the noise level. When the noise level is large, we expect the estimation error increases. Second, the density of covariate $p(x)$ is inversely related to the variance. This is also very reasonable because when $p(x)$ is large, there tends to be more data points

[1] https://en.wikipedia.org/wiki/Kernel_regression

[2] if you are interested in the derivation, check http://www.ssc.wisc.edu/~bhansen/718/NonParametrics2.pdf and http://www.maths.manchester.ac.uk/~peterf/MATH38011/NPR%20N-W%20Estimator.pdf

around x, increasing the size of sample that we are averaging from. Last, the convergence rate is $O\left(\frac{1}{nh}\right)$, which is the same as the KDE.

MSE and MISE. Using the expression of bias and variance, the MSE at point x is

$$\textbf{MSE}(\widehat{m}_h(x)) = \frac{h^4}{4}\mu_K^2\left(m''(x) + 2\frac{m'(x)p'(x)}{p(x)}\right)^2 + \frac{\sigma^2\cdot\sigma_K^2}{p(x)}\cdot\frac{1}{nh} + o(h^4) + o\left(\frac{1}{nh}\right)$$

and the MISE is

$$\textbf{MISE}(\widehat{m}_h) = \frac{h^4}{4}\mu_K^2\int\left(m''(x) + 2\frac{m'(x)p'(x)}{p(x)}\right)^2 dx + \frac{\sigma^2\cdot\sigma_K^2}{nh}\int\frac{1}{p(x)}dx + o(h^4) + o\left(\frac{1}{nh}\right). \tag{9.6}$$

Optimizing the major components in equation (9.6) (the AMISE), we obtain the optimal value of the smoothing bandwidth

$$h_{\mathsf{opt}} = C^*\cdot n^{-1/5},$$

where C^* is a constant depending on p and K.

9.3.2 Uncertainty and Confidence Intervals

How do we assess the quality of our estimator $\widehat{m}_h(x)$?

We can use the bootstrap to do it. In this case, empirical bootstrap, residual bootstrap, and wild bootstrap all can be applied. But note that each of them relies on slightly different assumptions. Let $(X_1^*, Y_1^*), \cdots, (X_n^*, Y_n^*)$ be the bootstrap sample. Applying the bootstrap sample to equation (9.5), we obtain a bootstrap kernel regression, denoted as $\widehat{m}_h^*$. Now repeat the bootstrap procedure B times, this yields

$$\widehat{m}_h^{*(1)}, \cdots, \widehat{m}_h^{*(B)},$$

B bootstrap kernel regression estimator. Then we can estimate the variance of $\widehat{m}_h(x)$ by the sample variance

$$\widehat{\mathsf{Var}}_B(\widehat{m}_h(x)) = \frac{1}{B-1}\sum_{\ell=1}^{B}\left(\widehat{m}_h^{*(\ell)}(x) - \bar{\widehat{m}}_{h,B}^*(x)\right), \quad \bar{\widehat{m}}_{h,B}^*(x) = \frac{1}{B}\sum_{\ell=1}^{B}\widehat{m}_h^{*(\ell)}(x).$$

Similarly, we can estimate the MSE as what we did in Lecture 5 and 6. However, when using the bootstrap to estimate the uncertainty, one has to be very careful because when h is either too small or too large, the bootstrap estimate may fail to converge its target.

When we choose $h = O(n^{-1/5})$, the bootstrap estimate of the variance is consistent but the bootstrap estimate of the MSE might not be consistent. The main reason is: it is easier for the bootstrap to estimate the variance than the bias. Thus, when we choose h in such a way, both bias and the variance contribute a lot to the MSE so we cannot ignore the bias. However, in this case, the bootstrap cannot estimate the bias consistently so the estimate of the MSE is not consistent.

Confidence interval. To construct a confidence interval of $m(x)$, we will use the following property of the kernel regression:

$$\sqrt{nh}\left(\widehat{m}_h(x) - \mathbb{E}(\widehat{m}_h(x))\right) \xrightarrow{D} N\left(0, \frac{\sigma^2\cdot\sigma_K^2}{p(x)}\right)$$

$$\frac{\widehat{m}_h(x) - \mathbb{E}(\widehat{m}_h(x)}{\mathsf{Var}(\widehat{m}_h(x))} \xrightarrow{D} N(0,1).$$

The variance depends on three quantities: σ^2, σ_K^2, and $p(x)$. The quantity σ_K^2 is known because it is just a characteristic of the kernel function. The density of covariates $p(x)$ can be estimated using a KDE. So what remains unknown is the noise level σ^2. A good news is: we can estimate it using the residuals. Recall that residuals are

$$e_i = Y_i - \widehat{Y}_i = Y_i - \widehat{m}_h(X_i).$$

When $\widehat{m}_h \approx m$, the residual becomes an approximation to the noise ϵ_i. The quantity $\sigma^2 = \mathsf{Var}(\epsilon_1)$ so we can use the sample variance of the residuals to estimate it (note that the average of residuals is 0):

$$\widehat{\sigma}^2 = \frac{1}{n - 2\nu + \widetilde{\nu}} \sum_{i=1}^{n} e_i^2, \tag{9.7}$$

where $\nu, \widetilde{\nu}$ are quantities acting as degree-of-freedom in which we will explain later. Thus, a $1-\alpha$ CI can be constructed using

$$\widehat{m}_h(x) \pm z_{1-\alpha/2} \frac{\widehat{\sigma} \cdot \sigma_K}{\sqrt{\widehat{p}_n(x)}},$$

where $\widehat{p}_n(x)$ is the KDE of the covariates.

Bias issue.

9.3.3 Resampling Techniques

Cross-validation.

Bootstrap approach.

http://faculty.washington.edu/yenchic/17Sp_403/Lec8-NPreg.pdf

9.3.4 Relation to KDE

Many theoretical results of the KDE apply to the nonparametric regression. For instance, we can generalize the MISE into other types of error measurement between $\widehat{m}_h$ and m. We can also use derivatives of $\widehat{m}_h$ as estimators of the corresponding derivatives of m. Moreover, when we have a multivariate covariate, we can use either a radial basis kernel or a product kernel to generalize the kernel regression to multivariate case.

The KDE and the kernel regression has a very interesting relationship. Using the given bivariate random sample $(X_1, Y_1), \cdots, (X_n, Y_n)$, we can estimate the joint PDF $p(x, y)$ as

$$\widehat{p}_n(x, y) = \frac{1}{nh^2} \sum_{i=1}^{n} K\left(\frac{X_i - x}{h}\right) K\left(\frac{Y_i - y}{h}\right).$$

This joint density estimator also leads to a marginal density estimator of X:

$$\widehat{p}_n(x) = \int \widehat{p}_n(x, y) dy = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{X_i - x}{h}\right).$$

Now recalled that the regression function is the conditional expectation

$$m(x) = \mathbb{E}(Y|X = x) = \int y p(y|x) dy = \int y \frac{p(x, y)}{p(x)} dy = \frac{\int y p(x, y) dy}{p(x)}.$$

Replacing $p(x,y)$ and $p(x)$ by their corresponding estimators $\widehat{p}_n(x,y)$ and $\widehat{p}_n(x)$, we obtain an estimate of $m(x)$ as

$$\begin{aligned}
\widehat{m}_n(x) &= \frac{\int y\widehat{p}_n(x,y)dy}{\widehat{p}_n(x)} \\
&= \frac{\int y \frac{1}{nh^2}\sum_{i=1}^n K\left(\frac{X_i-x}{h}\right) K\left(\frac{Y_i-y}{h}\right) dy}{\frac{1}{nh}\sum_{i=1}^n K\left(\frac{X_i-x}{h}\right)} \\
&= \frac{\sum_{i=1}^n K\left(\frac{X_i-x}{h}\right) \cdot \int y \cdot K\left(\frac{Y_i-y}{h}\right)\frac{dy}{h}}{\sum_{i=1}^n K\left(\frac{X_i-x}{h}\right)} \\
&= \frac{\sum_{i=1}^n K\left(\frac{X_i-x}{h}\right) Y_i}{\sum_{i=1}^n K\left(\frac{X_i-x}{h}\right)} \\
&= \frac{\sum_{i=1}^n Y_i K\left(\frac{X_i-x}{h}\right)}{\sum_{i=1}^n K\left(\frac{X_i-x}{h}\right)} \\
&= \widehat{m}_h(x).
\end{aligned}$$

Note that when $K(x)$ is symmetric, $\int y \cdot K\left(\frac{Y_i-y}{h}\right)\frac{dy}{h} = Y_i$. Namely, we may understand the kernel regression as an estimator inverting the KDE of the joint PDF into a regression estimator.

9.4 Linear Smoother

Now we are going to introduce a very important notion called linear smoother. Linear smoother is a collection of many regression estimators that have nice properties. The linear smoother is an estimator of the regression function in the form that

$$\widehat{m}(x) = \sum_{i=1}^n \ell_i(x) Y_i, \tag{9.8}$$

where $\ell_i(x)$ is some function depending on $X_1, \cdots, X_n$ but not on any of $Y_1, \cdots, Y_n$.

The residual for the i-th observation can be written as

$$e_j = Y_j - \widehat{m}(X_j) = Y_j - \sum_{i=1}^n \ell_i(X_j) Y_i.$$

Let $e = (e_1, \cdots, e_n)^T$ be the vector of residuals and define an $n \times n$ matrix L as $L_{ij} = \ell_j(X_i)$:

$$L = \begin{pmatrix}
\ell_1(X_1) & \ell_2(X_1) & \ell_3(X_1) & \cdots & \ell_n(X_1) \\
\ell_1(X_2) & \ell_2(X_2) & \ell_3(X_2) & \cdots & \ell_n(X_2) \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
\ell_1(X_n) & \ell_2(X_n) & \ell_3(X_n) & \cdots & \ell_n(X_n)
\end{pmatrix}$$

Then the predicted vector $\widehat{\mathbb{Y}} = (\widehat{Y}_1, \cdots, \widehat{Y}_n)^T = LY$, where $\mathbb{Y} = (Y_1, \cdots, Y_n)^T$ is the vector of observed Y_i's and $e = \mathbb{Y} - \widehat{\mathbb{Y}} = \mathbb{Y} - L\mathbb{Y} = (I - L)\mathbb{Y}$.

Example: Linear Regression. For the linear regression, let $\mathbb{X}$ denotes the data matrix (first column is all value 1 and second column is $X_1, \cdots, X_n$). We know that $\widehat{\beta} = (\mathbb{X}^T\mathbb{X})^{-1}\mathbb{X}^T\mathbb{Y}$ and $\widehat{\mathbb{Y}} = \mathbb{X}\widehat{\beta} = \mathbb{X}(\mathbb{X}^T\mathbb{X})^{-1}\mathbb{X}^T\mathbb{Y}$. This implies that the matrix L is

$$L = \mathbb{X}(\mathbb{X}^T\mathbb{X})^{-1}\mathbb{X}^T,$$

which is also the projection matrix in linear regression. Thus, the linear regression is a linear smoother.

Example: Regressogram. The regressogram is also a linear smoother. Let $B_1, \cdots, B_m$ be the bins of the covariate and define $B(x)$ be the bin such that x belongs to. Then

$$\ell_j(x) = \frac{I(X_j \in B(x))}{\sum_{i=1}^n I(X_i \in B(x))}.$$

Example: Kernel Regression. As you may expect, the kernel regression is also a linear smoother. Recall from equation (9.5)

$$\widehat{m}_h(x_0) = \frac{\sum_{i=1}^n Y_i K\left(\frac{x_0 - X_i}{h}\right)}{\sum_{\ell=1}^n K\left(\frac{x_0 - X_\ell}{h}\right)} = \sum_{i=1}^n W_i^K(x_0) Y_i, \quad W_i^K(x_0) = \frac{K\left(\frac{x_0 - X_i}{h}\right)}{\sum_{\ell=1}^n K\left(\frac{x_0 - X_\ell}{h}\right)}$$

so

$$\ell_j(x) = \frac{K\left(\frac{x - X_j}{h}\right)}{\sum_{\ell=1}^n K\left(\frac{x - X_\ell}{h}\right)}.$$

9.4.1 Variance of Linear Smoother

The linear smoother has an unbiased estimator of the underlying noise level σ^2. Recall that then noise level $\sigma^2 = \mathsf{Var}(\epsilon_i)$.

We need to use two tricks about variance and covariance matrix. For a matrix A and a random variable X,

$$\mathsf{Cov}(AX) = A\mathsf{Cov}(X)A^T.$$

Thus, the covariance matrix of the residual vector

$$\mathsf{Cov}(e) = \mathsf{Cov}((I - L)\mathbb{Y}) = (I - L)\mathsf{Cov}(\mathbb{Y})(I - L^T).$$

Because $Y_1, \cdots, Y_n$ are IID, $\mathsf{Cov}(\mathbb{Y}) = \sigma^2 \mathbb{I}_n$, where $\mathbb{I}_n$ is the $n \times n$ identity matrix. This implies

$$\mathsf{Cov}(e) = (I - L)\mathsf{Cov}(\mathbb{Y})(I - L^T) = \sigma^2(I - L - L^T + LL^T).$$

Now taking matrix trace in both side,

$$\mathsf{Tr}(\mathsf{Cov}(e)) = \sum_{i=1}^n \mathsf{Var}(e_i) = \sigma^2 \mathsf{Tr}(I - L - L^T + LL^T) = \sigma^2(n - \nu - \nu + \widetilde{\nu}),$$

where $\nu = \mathsf{Tr}(L)$ and $\widetilde{\nu} = \mathsf{Tr}(LL^T)$. Because the residual square is approximately $\mathsf{Var}(e_i)$, we have

$$\sum_{i=1}^n e_i^2 \approx \sum_{i=1}^n \mathsf{Var}(e_i) = \sigma^2(n - 2\nu + \widetilde{\nu}).$$

Thus, we can estimate σ^2 by

$$\widehat{\sigma}^2 = \frac{1}{n - 2\nu + \widetilde{\nu}} \sum_{i=1}^n e_i^2, \tag{9.9}$$

which is what we did in equation (9.7). The quantity ν is called the degree of freedom. In the linear regression case, $\nu = \widetilde{\nu} = p + 1$, the number of covariates so the variance estimator $\widehat{\sigma}^2 = \frac{1}{n-p-1}\sum_{i=1}^n e_i^2$. If you have learned the variance estimator of a linear regression, you should be familiar with this estimator.

The degree of freedom ν is easy to interpret in the linear regression. And the power of equation (9.9) is that it works for every linear smoother as long as the errors ϵ_i's are IID. So it shows how we can define *effective degree of freedom* for other complicated regression estimator.