Chu, Chi-Yang; Henderson, Daniel J.; Parmeter, Christopher F.

Article

Plug-in bandwidth selection for kernel density estimation with discrete data

Econometrics

Provided in Cooperation with:
MDPI – Multidisciplinary Digital Publishing Institute, Basel

Suggested Citation: Chu, Chi-Yang; Henderson, Daniel J.; Parmeter, Christopher F. (2015) : Plug-in bandwidth selection for kernel density estimation with discrete data, Econometrics, ISSN 2225-1146, MDPI, Basel, Vol. 3, Iss. 2, pp. 199-214, https://doi.org/10.3390/econometrics3020199

This Version is available at:
https://hdl.handle.net/10419/171823

Econometrics **2015**, *3*, 199-214; doi:10.3390/econometrics3020199

OPEN ACCESS

econometrics

ISSN 2225-1146

www.mdpi.com/journal/econometrics

Article

Plug-in Bandwidth Selection for Kernel Density Estimation with Discrete Data

Chi-Yang Chu [1], Daniel J. Henderson [1] and Christopher F. Parmeter [2,*]

[1] Department of Economics, Finance, and Legal Studies, University of Alabama, Tuscaloosa, AL 35487-0224, USA; E-Mails: cchu2@crimson.ua.edu (C.-Y.C.); djhender@cba.ua.edu (D.J.H.)

[2] Department of Economics, University of Miami, Coral Gables, FL 33124-6520, USA

* Author to whom correspondence should be addressed; E-Mail: cparmeter@bus.miami.edu; Tel.: +1-305-284-4397.

Academic Editor: Kerry Patterson

Received: 27 November 2014 / Accepted: 11 March 2015 / Published: 31 March 2015

Abstract: This paper proposes plug-in bandwidth selection for kernel density estimation with discrete data via minimization of mean summed square error. Simulation results show that the plug-in bandwidths perform well, relative to cross-validated bandwidths, in non-uniform designs. We further find that plug-in bandwidths are relatively small. Several empirical examples show that the plug-in bandwidths are typically similar in magnitude to their cross-validated counterparts.

Keywords: nonparametric; kernel; discrete variable; bandwidth selection; plug-in

JEL classifications: C14

---

1. Introduction

Bandwidth selection plays an important role in nonparametric density estimation. An appropriate bandwidth can help yield an estimated density that is close to the true density; however, a poorly chosen bandwidth can severely distort the true underlying features of the density. Thus, judicious choice of bandwidth is suggested. A range of alternatives exists for practitioners to select bandwidths, the most common being data-driven and plug-in methods.

Several data-driven approaches exist which choose the bandwidth via minimizing the distance between the true and estimated density. In the continuous data setting these methods are shown to converge slowly and display erratic finite sample performance [1]. Unlike data-driven methods, plug-in methods [2,3] require *a priori* assumptions about the unknown distribution of the data and then seek to minimize the asymptotic mean integrated square error (AMISE) of a density estimator $\widehat{f}(x)$. In the case of a single continuous variable with Gaussian kernel, the optimal bandwidth for normally distributed data is $h_{opt} = 1.06\sigma n^{-\frac{1}{5}}$, where σ is the standard deviation of x and n is the sample size. This method is often employed for preliminary analysis. Even in the case where the data are not Gaussian, use of this "optimal" bandwidth often gives an accurate representation of the distribution.

Plug-in bandwidth selection in the continuous data setting is quite common and lauded for both its practical and theoretical performance (see [1]). However, in the discrete data setting, much less is known about the relative performance of plug-in type selection rules relative to cross-validation; as noted in [4], cross-validation has the ability to smooth out uniformly distributed variables. Indeed, much of the pioneering work on discrete data kernel smoothing has focused exclusively on data-driven bandwidth selection. Here we seek to study, in the univariate setting, the performance of a plug-in bandwidth selector.

Much of the extant literature on estimation of discrete densities focus on multivariate binary discrimination.[1] Here, our focus will be on univariate multinomial distributions. Adopting the kernel approach, the underlying density, $p(x)$, is estimated by $\widehat{p}(x) = \frac{1}{n}\sum_{i=1}^{n} l(\cdot)$, with $l(\cdot)$ a kernel function for discrete data. We first begin by deriving a plug-in bandwidth for Aitchison and Aitken's [6] unordered kernel via minimization of mean summed squared error (MSSE).[2] This bandwidth provides intuition for relationships between sample size, true probabilities, and the number of categories. In addition, we derive plug-in bandwidths for Wang and van Ryzin's [10] ordered kernel function as well as other discrete kernel functions [4,11,12]. Although we have closed form solutions for the unordered cases, we cannot derive them for the general ordered case. It is noted that the way we derive plug-in bandwidths is similar to [5] with the main differences being discussed when we introduce the methodology.

For this set of kernel functions, our main findings are: (1) the plug-in bandwidths perform well when the simulated data are not uniform, (2) the plug-in bandwidths for ordered kernels are relatively small because the kernels have geometric functional forms, (3) in the case of three categories, the plug-in bandwidth for the Li and Racine ordered kernel has a flatness property, which results from the structural form of the formula for the bandwidth, and (4) our simulated and empirical examples show that the plug-in bandwidth are typically similar to those from cross-validation routines. Although we find some evidence of finite sample gains in the unordered case, these gains diminish with the sample size.

The remainder of this paper is organized as follows: Section 2 introduces the discrete kernel functions and derives the plug-in bandwidths. Section 3 shows the finite sample performance via simulations. Section 4 gives several empirical illustrations. Section 5 concludes.

---

[1] Hall [5] adopts the kernel approach to derive the optimal bandwidth for Aitchison and Aitken's [6] kernel function via minimization of mean summed square error (MSSE) and the result is applied to discriminant analysis.

[2] Titterington [7] applies Aitchison and Aitken's [6] kernel function to the approach in [8] and derives the optimal bandwidth (see also [9]).

2. Methodology

When estimating a univariate probability function, an intuitive approach is to use the sample frequency of occurrence as the estimator of cell probability (*i.e.*, the frequency approach). Its mathematical representation is

$$\widetilde{p}(x) = \frac{1}{n}\sum_{i=1}^{n} 1(X_i = x) = \frac{n_x}{n}$$

where X is a discrete random variable with support $S = \{0, 1, \ldots, c-1\}$, x takes c different values, n_x is the number of observations that are equal to x ($\sum_{x=0}^{c-1} n_x = n$), and $1(X_i = x) = 1$ if $X_i = x$ and zero otherwise. $\widetilde{p}(x)$ is a consistent estimator of $p(x)$ as long as n_x grows proportionally to n as n goes to infinity, which implies that the sample size n is much larger than the number of categories c. In many situations, the number of categories is close to or even greater than the sample size and this results in sparse data. A way to solve this problem is to borrow information from nearby cells to improve estimation of each cell probability (e.g., the kernel approach). Below we highlight methods to smooth over both unordered and ordered discrete data via kernel functions. We derive plug-in bandwidths for each kernel function and make comparisons with cross-validation methods.

2.1. Aitchison and Aitken (1976)

Aitchison and Aitken [6] were the first to introduce a kernel function to smooth unordered discrete variables. The kernel function they proposed is

$$l(X_i, x, \lambda) = \begin{cases} 1-\lambda \text{ if } X_i = x \\ \frac{\lambda}{c-1} \quad \text{if } X_i \neq x \end{cases}$$

where λ is the smoothing parameter (bandwidth), which is bounded between 0 and $\frac{c-1}{c}$. When $\lambda = 0$, $l(X_i, x, \lambda)$ collapses to the indicator function $1(X_i = x)$ and is identical to the frequency approach. Alternatively, when $\lambda = \frac{c-1}{c}$, $l(X_i, x, \lambda)$ gives uniform weighting. In other words, the weights for $X_i = x$ and $X_i \neq x$ are the same. Using this kernel function, a probability function can be estimated as

$$\widehat{p}(x) = \frac{1}{n}\sum_{i=1}^{n} l(X_i, x, \lambda) \tag{1}$$

where the sum of the kernel weights is equal to one and this ensures that $\widehat{p}(x)$ is a proper probability estimate lying between 0 and 1. It can be shown that the bias and variance of $\widehat{p}(x)$ are

$$Bias\left[\widehat{p}(x)\right] = \lambda\left[\frac{1 - c \cdot p(x)}{c-1}\right]$$

and

$$Var\left[\widehat{p}(x)\right] = \frac{p(x)\left[1-p(x)\right]}{n}\left(1 - \frac{\lambda c}{c-1}\right)^2$$

respectively (see also [13] or [14]). In contrast to the frequency approach, the kernel approach introduces bias, but can significantly reduce the variance and hence mean squared error (MSE).

To derive the optimal bandwidth λ^* (the plug-in bandwidth λ^{PI}), we minimize mean sum squared error (MSSE) which aggregates (pointwise) MSE over the support of x and hence is a global measure. Formally, we have

$$MSSE\left[\widehat{p}\left(x\right)\right] = \sum_{x=0}^{c-1}\left(Bias\left[\widehat{p}\left(x\right)\right]\right)^2 + \sum_{x=0}^{c-1} Var\left[\widehat{p}\left(x\right)\right]$$

We take the first derivative of MSSE with respect to λ as

$$\frac{\partial\, MSSE\left[\widehat{p}\left(x\right)\right]}{\partial\,\lambda} = 2\lambda\sum_{x=0}^{c-1}\left[\frac{1-cp(x)}{c-1}\right]^2 - \frac{2}{n}\left(1-\frac{\lambda c}{c-1}\right)\left(\frac{c}{c-1}\right)\sum_{x=0}^{c-1} p(x)\left[1-p(x)\right]$$

and setting this equal to zero leads to the optimal bandwidth

$$\lambda^{PI} = \frac{c-1}{c}\left\{1+\frac{n\sum\limits_{x=0}^{c-1}\left[\frac{1}{c}-p(x)\right]^2}{\sum\limits_{x=0}^{c-1} p(x)\left[1-p(x)\right]}\right\}^{-1} \tag{2}$$

Note that λ^{PI} converges to 0 (the lower bound) as n goes to infinity and converges to $\frac{c-1}{c}$ (the upper bound) as $p(x)$ approximates the uniform distribution. In practice, $p(x)$ can be replaced with $\widetilde{p}(x) = \frac{n_x}{n}$ to obtain $\widehat{\lambda}^{PI}$.

For the specific cases where $c = 2$ and 3, λ^{PI} is given as (where $p(x) \equiv p_x$)

$$\lambda^{PI}_{c=2} = \frac{1}{2}\left[1+\frac{n\left(\frac{1}{2}-p_1\right)^2}{p_1\left(1-p_1\right)}\right]^{-1}$$

and

$$\lambda^{PI}_{c=3} = \frac{2}{3}\left\{1+\frac{n\left[\left(\frac{1}{3}-p_0\right)^2+\left(\frac{1}{3}-p_1\right)^2+\left(\frac{1}{3}-p_0\right)\left(\frac{1}{3}-p_1\right)\right]}{p_0\left(1-p_0\right)+p_1\left(1-p_1\right)-p_0p_1}\right\}^{-1}$$

respectively.

As mentioned in the introduction, our approach to deriving an optimal bandwidth is similar to [5]. However, a key difference is that [5] focuses on the estimation of a multivariate binary distribution for discriminant analysis and hence the variables of interest only take on the values 0 and 1. With two cells, Aitchison and Aitken's [6] kernel function is fixed to be equal to $1-\lambda$ or λ. In addition, for computational convenience, [5] eliminates higher-order terms from the expectation and variance.

2.2. Wang and van Ryzin (1981)

Wang and van Ryzin [10] propose a kernel function to smooth ordered discrete variables (e.g., schooling years). Their kernel function is

$$l(X_i, x, \lambda) = \begin{cases} 1-\lambda & \text{if } X_i = x \\ \frac{1}{2}(1-\lambda)\lambda^{|X_i-x|} & \text{if } X_i \neq x \end{cases}$$

where $\lambda \in [0,1]$ and $|X_i - x|$ determines how much information is used from a nearby cell ($X_i \neq x$) to improve the estimation of probability. The larger the distance between X_i and x, the less information

used in estimation. It can be shown that this kernel function cannot give uniform weighting without proper rescaling. Further note that for this and the remaining kernel functions, that when this kernel is used to construct the probabilities as in Equation (1), it is not a proper probability estimate because the sum of the kernel weights is not equal to one. However, the estimate can be normalized via $\widehat{p}(x) / \sum_{x=0}^{c-1} \widehat{p}(x)$.

2.2.1. Two Categories ($c = 2$)

In the case of $c = 2$, the ordered kernel has the same interpretation as the unordered kernel, but we discuss it here for completeness. Using the same approach as we did for the Aitchison and Aitken kernel, it can be shown that $\lambda_{c=2}^{PI}$ is a root from the cubic equation

$$\frac{\partial\, MSSE\left[\widehat{p}(x)\right]}{\partial\, \lambda} = A\lambda^3 + B\lambda^2 + C\lambda + D$$

where the coefficients A, B, C, and D consist of n and $p(x)$.[3] For this cubic equation, the discriminant (Δ) is

$$\Delta = 18ABCD - 4B^3D + B^2C^2 - 4AC^3 - 27A^2D^2$$

where if

$$\Delta \begin{cases} < 0, \text{ one real root and two imaginary roots} \\ > 0, \text{ three distinct real roots} \\ = 0, \text{ multiple real roots} \end{cases}$$

exist. When $\Delta < 0$, the unique real root is

$$\lambda_{c=2}^{PI} = -\frac{1}{3A}\left(\frac{\alpha}{\beta} + \beta + B\right)$$

where

$$\alpha = B^3 - 3AC,$$

$$\beta = \left[\frac{\gamma + \left(\gamma^2 - 4\alpha^3\right)^{1/2}}{2}\right]^{1/3}$$

and

$$\gamma = 2B^3 - 9ABC + 27A^2D$$

It is difficult to tell the sign of the discriminant without entering real values and hence a numerical method is adopted. Figure 1 shows the plug-in bandwidths for three separate sample sizes ($n = 25$, 50, and 100). We see that they are largest when the true probability is near uniform and they decrease as the sample size increases.

---

[3] The exact form of the coefficients are $A = 2\left[\left(1 - p_1^2\right) + p_1^2\right] + \frac{1}{n}\left[4\left(1 - p_1\right)p_1\right]$, $B = 3\left[6\left(1 - p_1\right)p_1 - 1\right] - \frac{1}{n}\left[18\left(1 - p_1\right)p_1\right]$, $C = 5 - 18\left(1 - p_1\right)p_1 + \frac{1}{n}\left[26\left(1 - p_1\right)p_1\right]$, and $D = -\frac{1}{n}\left[12\left(1 - p_1\right)p_1\right]$.

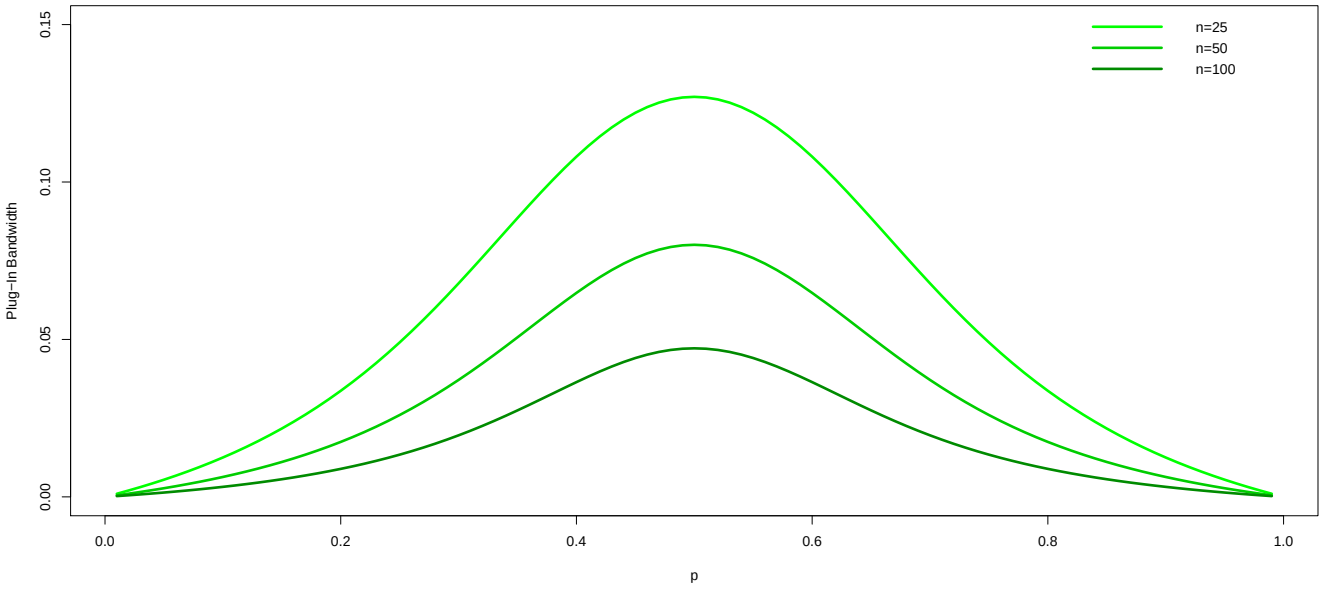

Figure 1. Plug-in bandwidth for the Wang and van Ryzin kernel with two cells ($c = 2$) for three different sample sizes ($n = 25, 50$ and 100).

2.2.2. Three Categories ($c = 3$)

Similar to Section 2.2.1, $\lambda_{c=3}^{PI}$ is a root from the quintic equation

$$\frac{\partial\, MSSE\left[\widehat{p}\,(x)\right]}{\partial\, \lambda} = E\lambda^5 + F\lambda^4 + G\lambda^3 + H\lambda^2 + I\lambda + J$$

where the coefficients E, F, G, H, I, and J consist of parameters n and $p(x)$.[4] Although there exist no formula for roots, numerical methods can be deployed to solve for the roots of the higher order polynomial. Using Brent's [15] method, our algorithm finds the same outcome in essentially each setting (one unique real root and two pairs of imaginary roots). Figure 2 shows the plug-in bandwidths for three separate sample sizes ($n = 25$, 50, and 100). We see that they are again largest when the true probability is near uniform and they decrease as the sample size increases.

---

[4] The exact form of the coefficients are $E = 6h$, $F = 5\,(g - 2h)$, $G = 4\,(f + h - c - 2g)$, $H = 3\,(c + e + g - b - 2f)$, $I = 2\,(a + b + d + f - 2e)$, and $J = e - 2d$, where $a = p_0^2 + p_1^2 + p_2^2$, $b = -2p_1\,(1 - p_1)$, $c = -2p_0p_2$, $d = \frac{1}{n}\left[p_0\,(1 - p_0) + p_1\,(1 - p_1) + p_2\,(1 - p_2)\right]$, $e = -\frac{1}{n}\left[2p_1\,(1 - p_1)\right]$, $f = \frac{1}{4}\left[2p_1^2 + (1 - p_1)^2\right] + \frac{1}{n}\left\{\frac{1}{4}\left[p_0\,(1 - p_0) + 2p_1\,(1 - p_1) + p_2\,(1 - p_2) - 10p_0p_2\right]\right\}$, $g = \frac{1}{2}p_1\,(1 - p_1) - \frac{1}{n}\left[\frac{1}{2}p_1\,(1 - p_1)\right]$, and $h = \frac{1}{4}\left(p_0^2 + p_2^2\right) + \frac{1}{n}\left\{\frac{1}{4}\left[p_0\,(1 - p_0) + p_2\,(1 - p_2)\right]\right\}$.

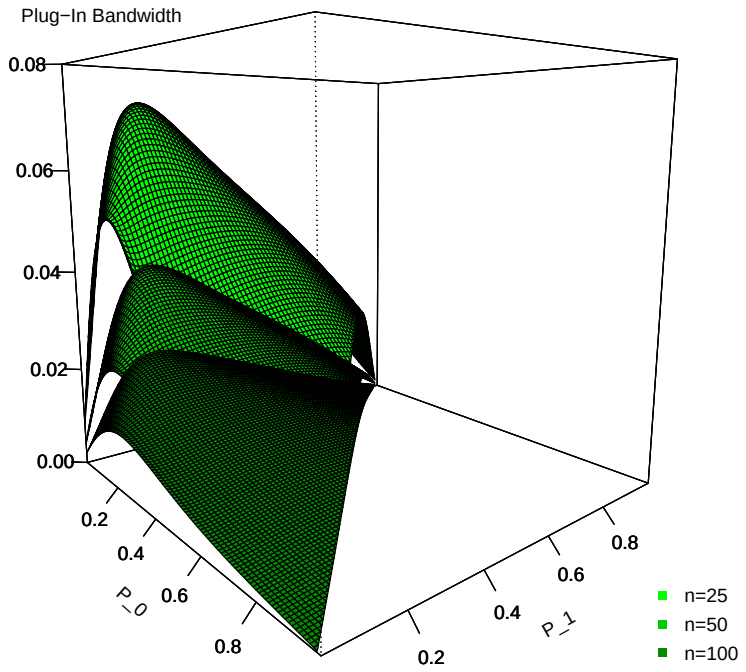

Figure 2. Plug-in bandwidth for the Wang and van Ryzin kernel with three cells ($c = 3$) for three different sample sizes ($n = 25, 50$ and 100).

2.2.3. Geometric Property of the Kernel Function

Notice from Figure 1 and especially Figure 2 that the plug-in bandwidths are relative small. Rajagopalan and Lall [16] mention that weights for the Wang and van Ryzin kernel drop off rapidly because of the geometric form of the kernel function. They argue that this kernel should not be used to smooth sparse data and propose an alternative approach.

To circumvent the drop-off problem, Rajagopalan and Lall [16] develop a discrete-version MSE optimal kernel which can be expressed in terms of the bandwidth and hence it is feasible to obtain the optimal bandwidth via minimizing MSSE. To avoid this problem with the Aitchinson and Aitken kernel, Titterington [7] derives an optimal bandwidth with the restriction that it must be greater than or equal to one-half.

2.3. (Unordered) Li and Racine

Li and Racine [11,12] and Ouyang, Li, and Racine [4] develop unordered and ordered kernel functions which are similar to Aitchison and Aitken's [6] and Wang and van Ryzin's [10] kernel functions, respectively. The former is introduced here and the latter will be shown in the next subsection. Their unordered kernel function is

$$l(X_i, x, \lambda) = \begin{cases} 1 \text{ if } X_i = x \\ \lambda \text{ if } X_i \neq x \end{cases}$$

where $\lambda \in [0, 1]$. This kernel has a similar limiting behavior to the Aitchison and Aitken kernel: when $\lambda = 0$, $l(X_i, x, \lambda)$ collapses to the indicator function $1(X_i = x)$ and when $\lambda = 1$, $l(X_i, x, \lambda)$ gives uniform weighting.

The plug-in bandwidth (λ^{PI}) for this kernel function is

$$\lambda^{PI} = \left\{ 1 + \frac{n \sum_{x=0}^{c-1} \left[1 - p(x)\right]^2}{\sum_{x=0}^{c-1} p(x)\left[1 - p(x)\right]} \right\}^{-1}.$$

λ^{PI} converges to 0 (the lower bound) as n goes to infinity, but doesn't converge to a specific value as $p(x)$ approximates the uniform distribution because c does not enter the kernel function directly.

For the specific cases where $c = 2$ and 3, λ^{PI} is given as

$$\lambda^{PI}_{c=2} = \left\{ 1 + \frac{n\left[\frac{1}{2} - p_1\left(1 - p_1\right)\right]}{p_1\left(1 - p_1\right)} \right\}^{-1} \tag{3}$$

and

$$\lambda^{PI}_{c=3} = \left\{ 1 + \frac{n\left[1 - p_0\left(1 - p_0\right) - p_1\left(1 - p_1\right) + p_0 p_1\right]}{p_0\left(1 - p_0\right) + p_1\left(1 - p_1\right) - p_0 p_1} \right\}^{-1}$$

respectively.

2.4. (Ordered) Li and Racine

The ordered Li and Racine kernel function is defined as

$$l(X_i, x, \lambda) = \begin{cases} 1 & \text{if } X_i = x \\ \lambda^{|X_i - x|} & \text{if } X_i \neq x \end{cases}$$

where $\lambda \in [0, 1]$ and $|X_i - x|$ has the same intuition as for the Wang and van Ryzin kernel (and has the same limiting behavior as the Li and Racine unordered kernel function).

For the case where $c = 2$, λ^{PI} can be derived as

$$\lambda^{PI}_{c=2} = \left\{ 1 + \frac{n\left[{p_1}^2 + \left(1 - p_1\right)^2\right]}{2p_1\left(1 - p_1\right)} \right\}^{-1}$$

and can be shown to be identical to Equation (3) and this is intuitive given that we have two categories.

For the case where $c = 3$, $\lambda^{PI}_{c=3}$ is a root from the cubic equation

$$\frac{\partial\, MSSE\left[\widehat{p}\,(x)\right]}{\partial\, \lambda} = K\lambda^3 + L\lambda^2 + M\lambda + N$$

where the coefficients K, L, M, and N consist of n and $p(x)$.[5]

Figure 3 shows the plug-in bandwidths for three separate sample sizes ($n = 25$, 50, and 100). We again see that they decrease as the sample size increases. However, the curves are relatively flat.

---

[5] The exact form of the coefficients are $K = 2\left(p_0^2 + p_2^2\right) + \frac{1}{n}\left\{2\left[p_0\left(1 - p_0\right) + p_2\left(1 - p_2\right)\right]\right\}$, $L = 3p_1\left(1 - p_1\right) - \frac{1}{n}\left[3p_1\left(1 - p_1\right)\right]$, $M = 2p_1^2 + \left(1 - p_1\right)^2 + \frac{1}{n}\left[p_0\left(1 - p_0\right) + 2p_1\left(1 - p_1\right) + p_2\left(1 - p_2\right) - 6p_0 p_2\right]$, and $N = -\frac{1}{n}\left[2p_1\left(1 - p_1\right)\right]$.

Specifically, for any given p_1 and n, L and N are fixed, because a trade-off occurs between p_0 and p_2 in K and M. In other words, K and M are essentially constant when p_0 (or p_2) changes. Therefore, the impact in the plug-in bandwidth of a change in p_0 (or p_2) is limited. Again, the figure shows that the plug-in bandwidth is relatively small in each case and the logic is the same as we discussed for the Wang and van Ryzin kernel.

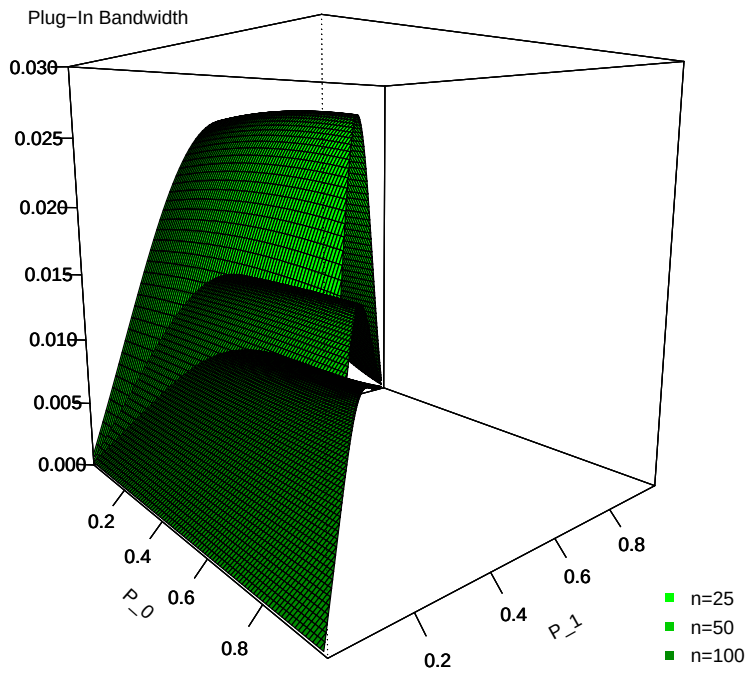

Figure 3. Plug-in bandwidth for the Li and Racine ordered kernel with three cells ($c = 3$) for three different sample sizes ($n = 25, 50$ and 100).

2.5. Plug-in versus Cross-Validated Bandwidths

To evaluate the performance of the plug-in bandwidth, we compare the plug-in bandwidth with a data-driven method, such as least-squares cross-validation (LSCV).[6] The feasible cross-validation function proposed by Ouyang, Li, and Racine [4] is given as

$$LSCV(\lambda) = \sum_{x=0}^{c-1} \widehat{p}(x)^2 - \frac{2}{n} \sum_{i=1}^{n} \widehat{p}_{-i}(X_i)$$

where $\widehat{p}_{-i}(X_i) = \frac{1}{n-1} \sum_{j=1,\, j \neq i}^{n} l(X_i, X_j, \lambda)$ is the leave-one-out estimator of $p(\cdot)$. If we were to use the Aitchison and Aitken kernel, the kernel estimator of $p(x)$ can be written as

$$\widehat{p}(x) = \widetilde{p}(x) \left(1 - \frac{\lambda c}{c-1}\right) + \frac{1}{c} \left(\frac{\lambda c}{c-1}\right)$$

[6] In the continuous case, LSCV yields a bandwidth that minimizes the integrated squared error of a density estimator $\widehat{f}(x)$ and which is obtained via $\min_{\lambda} \int \left[\widehat{f}(x) - f(x)\right]^2 dx$.

Similarly, the leave-one-out estimator of $p(X_i)$ can be written as

$$\widehat{p}_{-i}(X_i) = \frac{n}{n-1}\left(1 - \frac{\lambda c}{c-1}\right)\widetilde{p}(X_i) + \frac{n}{n-1}\left(\frac{\lambda}{c-1}\right) - \frac{1-\lambda}{n-1}$$

Taking the first derivative of the cross-validation function with respect to λ and setting this equal to zero leads to the optimal bandwidth

$$\widehat{\lambda}^{LSCV} = \frac{1}{n-1}\left(\frac{c-1}{c}\right)\frac{1 - S_{\widetilde{p}^2}}{S_{\widetilde{p}^2} - \frac{1}{c}} \tag{4}$$

where $S_{\widetilde{p}^2} \equiv \sum_{x=0}^{c-1}\widetilde{p}(x)^2 = \frac{1}{n}\sum_{i=1}^{n}\widetilde{p}(X_i)$. Note that this is a closed form solution and hence we could perform cross-validation without an optimization function.

For comparison, if we expand Equation (2), it can be rearranged such that

$$\begin{aligned}\widehat{\lambda}^{PI} &= \frac{1}{n}\left(\frac{c-1}{c}\right)\frac{1 - S_{\widetilde{p}^2}}{S_{\widetilde{p}^2} - \frac{1}{c}} - \left(\frac{c-1}{c}\right)\frac{(1 - S_{\widetilde{p}^2})^2}{n(1 - S_{\widetilde{p}^2})(S_{\widetilde{p}^2} - \frac{1}{c}) + n^2(S_{\widetilde{p}^2} - \frac{1}{c})^2} \\ &= \frac{1}{n}\left(\frac{c-1}{c}\right)\frac{1 - S_{\widetilde{p}^2}}{S_{\widetilde{p}^2} - \frac{1}{c}} + O\left(\frac{1}{n^2}\right)\end{aligned} \tag{5}$$

We can see that Equations (4) and (5) are asymptotically equivalent.[7] However, they are different in finite samples. The impact of this will be discussed further in the next section. It is straightforward to derive the cross-validated bandwidth for the unordered Li and Racine kernel, but simple closed-form comparisons for the ordered kernels (we believe) do not exist. That being said, the cross-validated bandwidth for the ordered kernels are similar to what we show in Sections 2.2 and 2.4. For example, for the Wang and van Ryzin kernel with $c = 3$, both the plug-in and cross-validated bandwidths are a unique real root from a quintic equation.

3. Simulations

3.1. Settings

For our simulations, the data are generated (as shown in Tables 1 and 2) by using multinomial ($c = 2$ and 3) and beta-binomial ($c = 3$) distributions for unordered and ordered kernels, respectively (we consider four scenarios for each distribution). The beta-binomial distribution is the discrete-version the beta distribution and has two parameters (α and β) that can determine the first four moments (see [17,18] for details on its simulation). We use the value $r = \frac{p_{\max}}{p_{\min}}$ as a measure of the design. We have a uniform design as r goes to one and empty cells (few observations in almost all categories) as r goes to infinity. We consider two measures (relative bias and relative MSE) to evaluate the performance of the plug-in bandwidth and take the average over $10,000$ replications for each sample size $n = 25$, 50, and 100. The measures considered are as follows:

---

[7] We would like to thank an anonymous referee for pointing this out to us.

(1) Ratio of bias of the estimated probability function with the cross-validated bandwidth to that with the estimated plug-in bandwidth,

$$R^{Bias} \equiv \frac{Bias[\widehat{p}_{LSCV}(X)]}{Bias[\widehat{p}_{PI}(X)]} = \frac{\frac{1}{n}\sum_{i=1}^{n}|\widehat{p}_{LSCV}(X_i) - p(X_i)|}{\frac{1}{n}\sum_{i=1}^{n}|\widehat{p}_{PI}(X_i) - p(X_i)|}$$

(2) Ratio of mean square error of the estimated probability function with the cross-validated bandwidth to that with the estimated plug-in bandwidth,

$$R^{MSE} \equiv \frac{MSE[\widehat{p}_{LSCV}(X)]}{MSE[\widehat{p}_{PI}(X)]} = \frac{\frac{1}{n}\sum_{i=1}^{n}[\widehat{p}_{LSCV}(X_i) - p(X_i)]^2}{\frac{1}{n}\sum_{i=1}^{n}[\widehat{p}_{PI}(X_i) - p(X_i)]^2}$$

Table 1. Monte Carlo simulation scenarios for unordered kernels.

Scenario	$c = 2$		$c = 3$	
	Probability	r	**Probability**	r
i	$p_0 = 0.25, p_1 = 0.75$	3.0	$p_0 = 0.15, p_1 = 0.35, p_1 = 0.50$	$3.\overline{3}$
ii	$p_0 = 0.30, p_1 = 0.70$	$2.\overline{3}$	$p_0 = 0.25, p_1 = 0.30, p_1 = 0.45$	2.5
iii	$p_0 = 0.40, p_1 = 0.60$	1.5	$p_0 = 0.25, p_1 = 0.35, p_1 = 0.40$	1.6
iv	$p_0 = 0.50, p_1 = 0.50$	1.0	$p_0 = 1/3, p_1 = 1/3, p_1 = 1/3$	1.0

Table 2. Monte Carlo simulation scenarios for ordered kernels.

Scenario	$c = 3$		
	Probability	**Shape Parameters**	r
i	$p_0 = 0.50, p_1 = 0.33, p_2 = 0.17$	$\alpha = 1, \beta = 2$	2.9
ii	$p_0 = 0.25, p_1 = 0.50, p_2 = 0.25$	$\alpha = 50, \beta = 50$	2.0
iii	$p_0 = 0.40, p_1 = 0.20, p_2 = 0.40$	$\alpha = 0.33, \beta = 0.33$	2.0
iv	$p_0 = 1/3, p_1 = 1/3, p_2 = 1/3$	$\alpha = 1, \beta = 1$	1.0

3.2. Non-Uniform Designs

Simulation results for the performance evaluation of the plug-in bandwidth via measures (1) and (2) are shown in Tables 3 and 4 for unordered and ordered kernels, respectively. The first three sections of each table are for the non-uniform design case. The relatively small sample sizes are to show finite differences, but we note that as we increase the sample size, the ratios go to one. For unordered kernels, the plug-in bandwidths appear to perform better than the cross-validated bandwidths. For ordered kernels, we have two scenarios where the cross-validated bandwidths perform better and one where the plug-in bandwidth has finite sample gains. There appears to be no finite sample dominance by either approach in the ordered case.

Table 3. Simulation results for the performance evaluation of the plug-in bandwidth for unordered kernels.

	$\lambda_{AA,\,c=2}$		$\lambda_{ULR,\,c=2}$		$\lambda_{AA,\,c=3}$		$\lambda_{ULR,\,c=3}$	
n	R^{Bias}	R^{MSE}	R^{Bias}	R^{MSE}	R^{Bias}	R^{MSE}	R^{Bias}	R^{MSE}
	p_0= 0.25 , p_1= 0.75				p_0= 0.15 , p_1= 0.35 , p_2= 0.50			
25	1.0867	1.2172	1.2815	1.6257	1.0658	1.1553	1.0566	1.1279
50	1.0290	1.0755	1.1366	1.3371	1.0428	1.1105	1.0633	1.1455
100	1.0058	1.0149	1.0628	1.1429	1.0135	1.0333	1.0381	1.0757
	p_0= 0.30 , p_1= 0.70				p_0= 0.25 , p_1= 0.30 , p_2= 0.45			
25	1.1377	1.1363	1.3230	1.6476	1.0701	1.0678	0.9615	0.8178
50	1.0581	1.1553	1.2105	1.5256	1.1075	1.1871	1.0885	1.0740
100	1.0159	1.0454	1.1033	1.2546	1.0945	1.2030	1.1463	1.2760
	p_0= 0.40 , p_1= 0.60				p_0= 0.25 , p_1= 0.35 , p_2= 0.40			
25	1.1282	1.2659	1.1528	1.0279	0.9927	0.9169	0.7839	0.5766
50	1.1264	1.2033	1.2443	1.2908	1.0601	1.0635	0.9077	0.7624
100	1.1104	1.2362	1.2694	1.5157	1.0904	1.1660	1.0290	1.0040
	p_0= 0.50 , p_1= 0.50				p_0= 1/3 , p_1= 1/3 , p_2= 1/3			
25	0.5692	0.7130	0.3289	0.3902	0.5133	0.5906	0.2866	0.2705
50	0.6138	0.7150	0.3480	0.3652	0.5076	0.5814	0.2730	0.2455
100	0.5901	0.7049	0.3237	0.3402	0.4938	0.5692	0.2586	0.2275

Table 4. Simulation results for the performance evaluation of the plug-in bandwidth for ordered kernels.

	$\lambda_{WR,\,c=3}$		$\lambda_{OLR,\,c=3}$	
n	R^{Bias}	R^{MSE}	R^{Bias}	R^{MSE}
	$p_0 = 0.50, p_1 = 0.33, p_3 = 0.17$			
25	0.9561	0.9153	0.9825	0.9458
50	0.9843	0.9642	0.9972	0.9874
100	1.0010	0.9936	1.0069	1.0065
	$p_0 = 0.25, p_1 = 0.50, p_3 = 0.25$			
25	0.9178	0.8504	0.9118	0.8268
50	0.9467	0.9007	0.9401	0.8894
100	0.9775	0.9485	0.9773	0.9456
	$p_0 = 0.40, p_1 = 0.20, p_3 = 0.40$			
25	1.0019	0.9783	1.0035	0.9749
50	1.0096	1.0046	1.0085	1.0015
100	1.0072	1.0085	1.0054	1.0046
	$p_0 = 1/3, p_1 = 1/3, p_3 = 1/3$			
25	0.8894	0.8052	0.8804	0.7841
50	0.9257	0.8680	0.9258	0.8642
100	0.9655	0.9262	0.9677	0.9290

3.3. Uniform Designs

In the uniform designs the cross-validated approach possesses substantial gains in finite samples relative to the plug-in bandwidth selector. In the unordered case, the gains appear to persist even as the sample size grows, whereas in the ordered case, while gains exist, they decrease with the sample size.

The asymptotic behavior of $\widehat{\lambda}_{AA}^{PI}$, λ_{AA}^{LSCV}, and λ_{ULR}^{LSCV} is akin to [19], which shows that LSCV can remove irrelevant discrete variables from a conditional density by divergence of bandwidths to their respective upper bounds or, equivalently, by the shrinkage of their respective marginal distributions toward a uniform distribution. However, λ_{WR}^{LSCV} and λ_{OLR}^{LSCV} do not exhibit such behavior.

In uniform designs, the change in ratios for the unordered kernels is different from that for ordered kernels: the former (*i.e.*, the Aitchison and Aitken kernel) implies that the cross-validated bandwidth converges to the upper bound faster than the plug-in bandwidth, but for the latter, both converge to the lower bound at a relatively similar rate.

In a uniform design, when n goes to infinity, $\widetilde{p}(x)$ converges in probability to $\frac{1}{c}$ and hence $\sum_{x=0}^{c-1}\left[\frac{1}{c}-\widetilde{p}(x)\right]^2$ in Equation (2) approximates zero. In other words, when n goes to infinity, there are two opposite forces driving $\widehat{\lambda}_{AA}^{PI}$: one is n and the other is $\sum_{x=0}^{c-1}\left[\frac{1}{c}-\widetilde{p}(x)\right]^2$. Based on our simulation results, $\sum_{x=0}^{c-1}\left[\frac{1}{c}-\widetilde{p}(x)\right]^2$ dominates n, and hence, $\widehat{\lambda}_{AA}^{PI}$ converges to the upper bound.[8] In addition, $\widehat{\lambda}_{AA}^{PI}$ converges slower than $\widehat{\lambda}_{AA}^{LSCV}$ due to the big O term in $\widehat{\lambda}_{AA}^{PI}$.

The plug-in bandwidths for the other three kernels don't have such competing forces in play and all of them converge to the lower bound as n goes to infinity. However, the cross-validated bandwidths for unordered kernels converge to the upper bound, but to the lower bound for ordered kernels.

4. Empirical Illustrations

In this section we consider two empirical examples to complement our Monte Carlo simulations. While each of these examples is simplistic, they will shed insight into the difference between estimated bandwidths when applied to empirical data. Specifically, we hope to see the relative sizes of the bandwidths from each procedure as well as examine how they change with the sample size.

For unordered kernels, we consider the travel mode choice (between Sydney and Melbourne, Australia) data from [20]. This data consists of $n = 210$ observations and $c = 4$ categories (air, train, bus, and car). We note here that we likely have a non-uniform design as the relative proportions for air, train, bus, and car are 0.28, 0.30, 0.14, and 0.28, respectively. For ordered kernels, we consider the well-studied salary data from [21]. This data consists of $n = 147$ observations. Instead of the $c = 6, 12$ or 28 cell cases typically considered, we further condense the number of cells to $c = 3$ to make closer comparisons to our simulations. We also appear to have a non-uniform design here as the relative proportions for low, middle, and high salaries are 0.26, 0.49, and 0.25, respectively. For each data set,

---

[8] Equations (4) and (5) reveal that the cross-validated and plug-in bandwidths converge to their (identical) upper bound, $\frac{c-1}{c}$, as n goes to infinity due to $S_{\widetilde{p}^2}-\frac{1}{c}\approx 0$.

we consider both the full-sample size as well as random sub-samples of size $n = 25$, 50 and 100 (again, consistent with the simulations).

Table 5 gives the plug-in and cross-validated bandwidths for each kernel for each data set for each sample size. The most glaring observation is that the plug-in bandwidth is always smaller than the corresponding cross-validated bandwidth, sometimes strikingly so. Further, the relative difference is not uniform across ordered and unordered kernels. We see that the plug-in and cross-validated bandwidths are most similar for the Aitchinson and Aitken kernel. The ratio of the cross-validated to the plug-in bandwidth is just above unity. However, for each of the other kernels, this ratio is many times larger (and increases with the sample size). Finally, as expected, the bandwidths each tend towards zero (non-uniform design) as the sample size increases.

Table 5. Empirical comparisons between plug-in and cross-validated bandwidths.

	Travel Mode ($c = 4$)				**Salary** ($c = 3$)			
n	λ_{AA}		λ_{ULR}		λ_{WR}		λ_{OLR}	
	PI	**LSCV**	**PI**	**LSCV**	**PI**	**LSCV**	**PI**	**LSCV**
25	0.2138	0.3114	0.0117	0.1507	0.0440	0.2324	0.0267	0.1625
50	0.1950	0.2688	0.0062	0.1226	0.0224	0.1495	0.0137	0.0930
100	0.1719	0.2253	0.0032	0.0969	0.0123	0.0811	0.0067	0.0475
Full	0.1372	0.1687	0.0015	0.0676	0.0083	0.0565	0.0046	0.0316

5. Conclusions

Bandwidth selection is sine qua non for practical kernel smoothing. In the continuous data setting, aside from being well-studied, plug-in methods are praised for their performance relative to data-driven methods. Yet, migrating to the discrete setting, relatively little discussion on plug-in bandwidth selection exists, especially in comparison to the matured literature surrounding the continuous data setting. Here, we offer a simple plug-in bandwidth selection rule for univariate density estimation where the data possesses a multinomial distribution and compare with the corresponding data-driven bandwidth selectors.

Simulation results show that plug-in bandwidths for unordered kernels perform well in a non-uniform design; similar to their performance in continuous data settings. We see that the plug-in bandwidths for ordered kernels are relatively small and provide little smoothing. Moreover, at least in the case of three categories, the plug-in bandwidth for the Li and Racine ordered kernel possesses a flatness property. The empirical examples show that the plug-in bandwidths are smaller than the cross-validated bandwidths, quite different than the continuous data setting where it is commonly seen that plug-in bandwidths tend to provide more smoothing than data-driven bandwidths.

Acknowledgments

We thank the editor and two anonymous referees for invaluable feedback which greatly improved the structure of the paper. All errors are ours alone. A detailed appendix of all derivations is available at www.the-smooth-operators.com.

Author Contributions

All authors contributed equally to the project.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Jones, M.C.; Marron, J.S.; Sheather, S.J. A brief survey of bandwidth selection for density estimation. *J. Am. Stat. Assoc.* **1996**, *91*, 401–407.
2. Deheuvels, P. Estimation nonparamétrique de la densité par histogrammes généralisés. *Rev. Stat. Appl.* **1977**, *25*, 5–42.
3. Silverman, B.W. *Density Estimation for Statistics and Data Analysis*; Chapman and Hall: London, UK, 1986.
4. Ouyang, D.; Li, Q.; Racine, J. Cross-Validation and the estimation of probability distributions with categorical data. *J. Nonparametr. Statist.* **2006**, *18*, 69–100.
5. Hall, P. On nonparametric multivariate binary discrimination. *Biometrika* **1981**, *68*, 287–294.
6. Aitchison, J.; Aitken, C.G.G. Multivariate binary discrimination by the kernel method. *Biometrika* **1976**, *63*, 413–420.
7. Titterington, D.M. A comparative study of kernel-based density estimates for categorical data. *Technometrics* **1980**, *22*, 259–268.
8. Good, I.J. *The Estimation of Probabilities*; Massachusetts Institute of Technology Press: Cambridge, MA, USA, 1965.
9. Fienberg, S.E.; Holland, P.W. Simultaneous estimation of multinomial cell probabilities. *J. Am. Stat. Assoc.* **1973**, *68*, 683–691.
10. Wang, M.; van Ryzin, J. A class of smooth estimators for discrete distributions. *Biometrika* **1981**, *68*, 301–309.
11. Li, Q.; Racine, J. Nonparametric estimation of distributions with categorical and continuous data. *J. Multivar. Anal.* **2003**, *86*, 266–292.
12. Li, Q.; Racine, J. Nonparametric estimation of regression functions with both categorical and continuous data. *J. Econom.* **2004**, *119*, 99–130.
13. Henderson, D.J.; Parmeter, C.F. *Applied Nonparametric Econometrics*; Cambridge University Press: Cambridge, UK, 2015.
14. Li, Q.; Racine, J. *Nonparametric Econometrics: Theory and Practice*; Princeton University Press: Princeton, NJ, USA, 2007.
15. Brent, R. *Algorithms for Minimization without Derivatives*; Englewood Cliffs: Bergen County, NJ, USA; Prentice-Hall: Upper Saddle River, NJ, USA, 1973.
16. Rajagopalan, B.; Lall, U. A kernel estimator for discrete distribution. *Nonparametr. Statist.* **1995**, *4*, 409–426.
17. Dong, J.; Simonoff, J.S. The construction and properties of boundary kernels for smoothing sparse multinomials. *J. Comput. Graph. Stat.* **1994**, *3*, 57–66.

18. Jacob, P.; Oliveira, P.E. Relative smoothing of discrete distributions with sparse observations. *J. Stat. Comput. Simul.* **2011**, *81*, 109–112.
19. Hall, P.; Racine, J.; Li, Q. Cross-Validation and the estimation of conditional probability densities. *J. Am. Stat. Assoc.* **2004**, *99*, 1015–1026.
20. Greene, W. *Econometric Analysis*; Prentice Hall: Upper Saddle River, NJ, USA, 2011.
21. Simonoff, J.S. *Smoothing Methods in Statistics*; Springer-Verlag: New York, NY, USA, 1996.