The Method of Sieves

Lecture Note
Advanced Microeconometrics
Anders Munk-Nielsen
Fall 2016, version 1.1

Abstract

This note introduces the method of sieves. The note is written assuming knowledge of the kernel regression estimator (Nadaraya-Watson) and written to relate the sieve estimation technique to this. The example of the partially linear model is used to motivate the method of sieves.

1 Introduction

This note introduces the method of sieves, an estimation technique relevant in situations where the researcher is not willing to impose functional form restrictions on the object of interest (or a part of it). One example might be the relationship between experience and wage income as in a partially linear model, which will be the example considered in this note. One simple way of capturing a non-linear relationship in one particular variable, z, might be to add a polynomial, $z, z^2, z^3, ...$. The method of sieves provides a formal framework for thinking about what such an approach implies. The framework extends to a very wide selection of estimation problems and has the clear advantage that it is typically easy to use.

1.1 The Partially Linear Model

Throughout this model, we will discuss the example of the partially linear model.

Partially linear model. In the partially linear model,

$$y_i = x_i\beta_o + h_o(z_i) + \varepsilon_i, \quad \mathbb{E}(\varepsilon_i|x_i, z_i) = 0, \tag{1}$$

where $x_i \in \mathbb{R}^P$ and $y_i, z_i, \varepsilon_i \in \mathbb{R}$.

Empirical examples: In a Mincer equation, y_i could be wage income and z_i could be experience while x_i would contain other controls such as years of schooling, parental background and industry dummies. In such an equation, we are particularly interested in exploring whether there are decreasing returns to experience and if so, when precisely they occur. In another example, z_i might be calendar time and we might wish to very precisely control for seasonal effects both within the year and over time. If the panel is long enough, one approach would be to add year dummies and month dummies. Alternatively, and also relevant for shorter panels, one might use a sieve estimator for h_o.

Naming conventions: We say that the model is:

Semi-nonparametric if we are interested in $h_o(\cdot)$,

Semiparametric if we are only interested in β_o,

Nonparametric if there is no $x_i\beta_o$-term and interest is in h_o,

Parametric if there is no h_o-term and interest is in β_o.

We will start by looking at the non-parametric regression model where $\beta_o = 0$ (Section 2) before we move to the semi-nonparametric version (Section 3). The proposed estimators herein will be based on a least squares characterization of the problem,

$$(\beta_o, h_o) = \underset{\beta\in\mathcal{B}, h\in\mathcal{H}}{\operatorname{argmin}} \mathbb{E}\left\{[y_i - x_i\beta - h(z_i)]^2\right\}. \tag{2}$$

However, sieve methods can also be used for maximum likelihood estimators, GMM, etc.

2 The Method of Sieves for Non-parametric Estimation

The core in the method of sieves is reducing the space in which we search for an estimator for the function $h_o(\cdot)$. In practice, we have no way of searching in the full space of continuous functions, so instead we limit our attention to e.g. polynomials of degree up to D, piecewise linear functions with D line segments over some interval. We will call D the *dimensionality* of the sieve. One very broad class of sieve estimators are captured by the *series estimator*,

$$h(z) = \sum_{d=0}^{D} \alpha_d p_d(z), \tag{3}$$

where $\{p_d(\cdot)\}_{d=1}^{D}$ are the *basis functions*, $p_d : \mathbb{R} \to \mathbb{R}$, and α_d are the *coefficients*.[1] One example is the polynomial basis functions, where $p_d(z) = z^d$, $z \in \mathbb{R}, d \in \mathbb{N}$. The clear advantage of the series estimator is that it can be expressed in terms of linear algebra. If $p^D(z) \equiv \Big(p_0(z), p_1(z), ..., p_D(z)\Big)$, we can write the approximator, h, as the linear index

$$h(z) = p^D(z)\alpha, \quad \alpha \in \mathbb{R}^{D+1}.$$

This form is very convenient because in many models, we will already be working with linear indices so the method of sieves amounts to adding additional terms that are just be basis functions of the sieve evaluated at z. However, the theoretical results concerning the sieve estimator are more general.

2.1 In Greater Generality

To formulate the problem and the model more generally, we must specify both the parameter space for the linear parameters, $\beta \in \mathcal{B} \subset \mathbb{R}^P$, and the function space wherein the unknown nonparametric function lives, $h_o \in \mathcal{H}$. For example, we might assume that $\mathcal{H}$ is the space of twice continuously differentiable functions on $\mathbb{R}$, $\mathcal{H} = \mathcal{C}^2(\mathbb{R})$. Let us start by considering the non-parametric estimation problem. The sample analogue of

[1] If z is multi-dimensional there are several ways of constructing the basis function, for example using products of one-dimensional basis functions. These are beyond the scope of this note.

the population problem (2) is

$$\min_{\beta\in\mathcal{B},h\in\mathcal{H}} N^{-1}\sum_{i=1}^{N}\left[y_i - h(z_i)\right]^2 .$$

The problem is infeasible because we do not have a technique for searching over all of $\mathcal{H}$, and even if we did there might be infinitely many solutions to the problem.

The core idea in the method of sieves is to replace $\mathcal{H}$ with a simpler space of lower *dimension*, $\mathcal{H}_D$, where $D \in \mathbb{N}$ denotes the dimension so that the sequence of sieve spaces "grow" in the sense that $\mathcal{H}_1 \subset \mathcal{H}_2 \subset ... \subset \mathcal{H}$. In the limit, the sieve space must become "dense" in $\mathcal{H}$ in the sense that for any $h^* \in \mathcal{H}$, the best approximation to h^* that can be found in $\mathcal{H}_C$ must get arbitrarily close to h^* as $D \to \infty$. In other words, we replace the infeasible sample problem with the *feasible* sample problem given by

$$\min_{\beta\in\mathcal{B},h\in\mathcal{H}_D} N^{-1}\sum_{i=1}^{N}\left[y_i - h(z_i)\right]^2 .$$

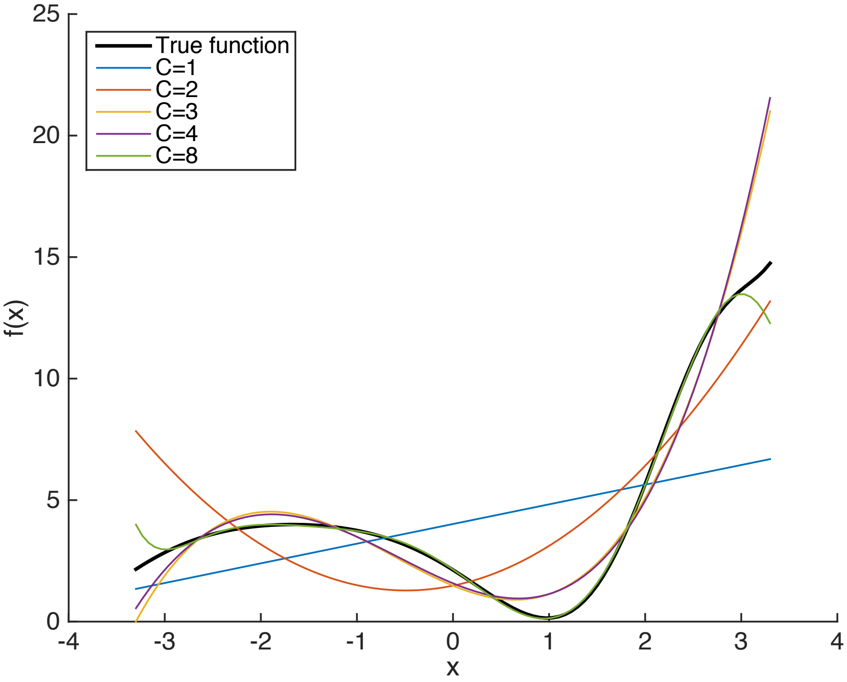
Let us consider the example of polynomial sieve spaces, where $\mathcal{H}_D$ denotes the space of polynomials of at most order D. We know from *Weierstrass' Approximation Theorem* that any continuously differentiable function over a compact interval can be approximated arbitrarily well by a polynomial (of sufficiently high degree). In other words, the space of polynomials becomes "dense" in the space of continuously differentiable functions as the order increases. Figure 1 shows polynomials of increasing complexity approximating a complex, non-polynomial function. We see that for $D = 1, 2$, the approximation is fairly bad but that the third and fourth order polynomials look quite similar. The 8th order polynomial shows an almost perfect match on $[-3; 3]$. Outside of the end points, all the polynomials diverge as we would expect.[2]

2.2 Non-parametric Sieve Estimation

When we work with the series estimator (3), the sieve estimation problem boils down to *i)* choosing the "basis functions", $p_d(\cdot)$, *ii)* choosing the dimension, D, and *iii)* estimating the coefficients on the basis functions, α_d. One example is to consider the

[2] Any polynomial has the property that if z becomes large or small enough, the polynomial will tend to ±∞.

Figure 1: Approximating Polynomials of Increasing Order



space of polynomial functions of at most order D, i.e. $p_d(z) = z^d$ for $d \in \mathbb{N}$. That is,

$$\mathcal{H}_C = \left\{ h : \mathbb{R} \to \mathbb{R} \middle| h(z) = \sum_{d=0}^{D} \alpha_d z^d, \ \ \alpha_d \in \mathbb{R}. \right\}.$$

Our sample problem becomes

$$\min_{\beta \in \mathcal{B}, \alpha \in \mathbb{R}^K} N^{-1} \sum_{i=1}^{N} \left(y_i - \sum_{d=0}^{D} \alpha_d z_i^d \right)^2 .$$

Note that if we write

$$p^D(z_i) \equiv (1, z_i, z_i^2, ..., z_i^D),$$

and use P^D to denote the $N \times D$ matrix with row $p^D(z_i)$, then the solution to the minimization problem is given by the standard OLS formula,

$$\hat{\alpha} = \left(P_D' P_D\right)^{-1} P_D' Y,$$

and our estimate of the function $h_o(z)$ would therefore be the function $\hat{h}(z) = p^D(z)\hat{\alpha}$. Note that since we have *chosen* D, the usual formulas for standard errors in OLS do not work. Accounting for the noise coming from the choice of D is tricky and will not be covered in this note.

3 Estimating the Partially Linear Model

3.1 A Kernel Approach: Robinson's Double Residual Method

The approach outlined here was first suggested by Robinson (1988) and named "Robinson's double residual method". Intuitively, the idea is to first subtract the linear influence of z on y and x respectively, thus *residualizing* the two variables. Then β_o can be estimated form these two residuals. Finally, the linear prediction $x\hat{\beta}$ can be subtracted from y to explore the nonparametric relationship between z and y, accounting for the linear index $x\hat{\beta}$.[3]

The Kernel regression approach to estimating β_o proceeds by first noting that if we take expectations on both sides of equation (1) conditional on z_i (and not x_i), we get

$$\mathbb{E}(y_i|z_i) = \mathbb{E}(x_i\beta_o|z_i) + h_o(z_i) + \mathbb{E}(\varepsilon_i|z_i).$$

If we subtract this from equation (1), we get

$$y_i - \mathbb{E}(y_i|z_i) = [x_i - \mathbb{E}(x_i|z_i)]\,\beta_o + \varepsilon_i - \mathbb{E}(\varepsilon_i|z_i).$$

If we introduce the notation, $\tilde{a}_i \equiv a_i - \mathbb{E}(a_i|z_i)$, i.e. the variable a_i minus its linear projection on z_i, then we can write the equation more compactly as

$$\tilde{y}_i = \tilde{x}_i\beta_o + \tilde{\varepsilon}_i.$$

We note that

$$\mathbb{E}(\tilde{\varepsilon}_i|x_i, z_i) = 0,$$

indicating that we can use OLS on the transformed data to estimate β_o consistently.[4]

To construct the transformed data, $\tilde{y}_i$, $\tilde{x}_{i1}$, ..., $\tilde{x}_{iP}$, we have to construct an estimate

[3] Alternatively, you can interpret it very much along the lines of the Frisch-Waugh-Lovell theorem.

[4] More precisely, for OLS consistency, we require $\mathbb{E}(\tilde{\varepsilon}_i|\tilde{x}_i) = 0$ but this turns out to be equivalent with the stated equation.

of the predictions, $m_Y(z_i) \equiv \mathbb{E}(y_i|z_i)$, $m_{X_p}(z_i) \equiv \mathbb{E}(x_{ip}|z_i)$. To do this, we can use the local constant Kernel regression estimator, which evaluated at z is given by

$$\hat{m}_Y(z) = \frac{\sum_{i=1}^N y_i K_b(z_i - z)}{\sum_{i=1}^N K_b(z_i - z)}. \tag{4}$$

The bandwidth, b, can for example be chosen using Silverman's rule of thumb (equation (5)).[5] The equation is similar for x,

$$\hat{m}_{X_p}(z) = \frac{\sum_{i=1}^N x_i K_b(z_i - z)}{\sum_{i=1}^N K_b(z_i - z)}.$$

Using this prediction method, we can construct the two sets of "estimated residuals",

$$\hat{\hat{y}}_i := y_i - \hat{m}_Y(z_i), \quad \hat{\hat{x}}_{ip} := x_{ip} - \hat{m}_{X_p}(z_i).$$

Then we estimate β_o using OLS on the transformed data, i.e.

$$\hat{\beta} = (\hat{\hat{X}}'\hat{\hat{X}})^{-1}\hat{\hat{X}}'\hat{\hat{Y}},$$

where $\hat{\hat{X}}$ is the $N \times P$ stacked matrix of $\hat{\hat{x}}_i$ and similarly for $\hat{\hat{Y}}$.

Note that since we have used a two-step approach to estimating $\hat{\beta}$, the usual OLS formula for standard errors does not apply. Obtaining asymptotically valid standard errors can be done for example using bootstrap methods although this is very computationally intensive. Asymptotic standard errors for $\hat{\beta}$, however, can under certain conditions be relatively simple (for example, $\hat{\beta}$ will often be $\sqrt{N}$-consistent in spite of the estimation noise coming from the residualization).

[5] Instead of using the local constant regression we can use the *local linear regression*, where $\hat{m}_Y(z) = \hat{a}(z) + \hat{b}(z)z$, where

$$\left(\hat{a}(z), \hat{b}(x)\right)' = [Z'\mathcal{K}(z)Z]^{-1}Z'\mathcal{K}(z)Y,$$

where $Y = (y_1, ..., y_N)'$, $\mathcal{K}(z)$ is an $N \times N$ diagonal matrix with (i, i)'th element $K_b(z_i - z)$ and Z is an $N \times 2$ matrix with i'th row $(1,\, z_i - z)$. This estimator performs better in the upper and lower ends of the support of z.

3.2 Estimating the Partially Linear Model Using Sieves

One example is to choose $\mathcal{H}_K$ to be the space of polynomial functions of at most order D. That is,

$$\mathcal{H}_C = \left\{ h : \mathbb{R} \to \mathbb{R} \middle| h(z) = \sum_{d=0}^{D} \alpha_d z^d, \;\; \alpha_d \in \mathbb{R}. \right\}.$$

Our sample problem becomes

$$\min_{\beta \in \mathcal{B}, \alpha \in \mathbb{R}^K} N^{-1} \sum_{i=1}^{N} \left(y_i - x_i\beta - \sum_{d=0}^{D} \alpha_d z_i^d \right)^2 .$$

Note that if we write

$$p^D(z_i) \equiv (1, z_i, z_i^2, ..., z_i^D),$$

and use P^D to denote the $N \times D$ matrix with row $p^D(z_i)$, then the solution is given by the standard OLS formula,

$$\begin{pmatrix} \hat{\beta} \\ \hat{\alpha} \end{pmatrix} = \left[(X, P_D)'(X, P_D) \right]^{-1} (X, P_D)'Y,$$

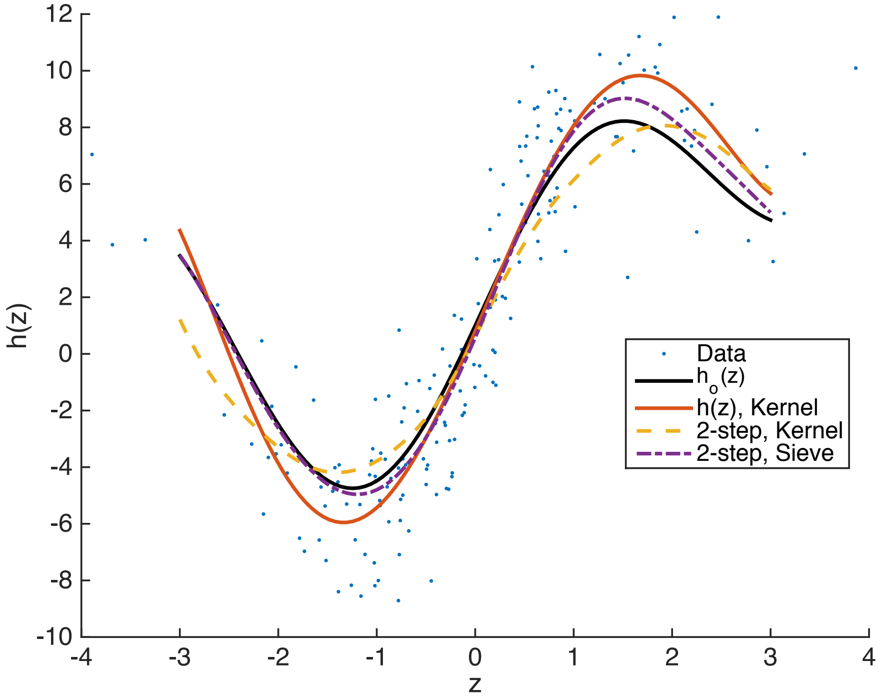
and our estimate of the function $h_o(z)$ would therefore be the function $\hat{h}(z) = p^D(z)\hat{\alpha}$.

Note that with the sieve method, there is no need for a two-step procedure and we recover $(\hat{\beta}, \hat{h})$ simultaneously. However, we are not allowed to use standard parametric inference procedures right away due to the fact that we have chosen D. Accounting from this can be very complicated and requires us to specify a *data driven* process for choosing D (i.e. as a function of N); we will discuss this more in Section 3.4.

Figure 2 shows the comparison of *1)* a simple 1-step kernel regression (which ignores the effect of the linear term, $x_i\beta$) *2)* a kernel-based two-step and *3)* a sieve-based two-step. The dataset has $N = 200$ observations. In the data generating process, x and z are positively correlated, meaning that the 1-step method that ignores $x\beta$ is biased. The kernel and sieve based approaches are both much closer to the true function but deviate somewhat, in particular towards the end-points of the data (for high/low z). If we were to increase N, both the kernel and sieve two-step estimators would converge to the true, black line, whereas the 1-step kernel estimator would not, due to the correlation between z and the omitted x.

We conclude this section with a brief comment on the use of sieve estimators more generally. The big advantage of sieve estimation is the ease of use; in many cases, we

Figure 2: Sieve Estimation: Dimension Choice



just pick D and treat the model as if it were a fully parametric model where we have just included $D + 1$ additional parameters, $\{\alpha_d\}_{d=0}^{D}$. This means that it is straightforward to include a semi-nonparametric component, $h(z)$, in a non-linear model such as probit. There are no general approaches to doing so with kernel methods.

3.3 The Choice of Sieve Space

The appropriate choice of Sieve space, $\mathcal{H}_D$, depends on the nature of the h_o function. The important requirement for the method of sieves to work is that h_o should be in the limit of $\mathcal{H}_D$ as $D \rightarrow \infty$. If this happens for a low value of D, all the better.[6] There is no general result on which function will be best; if the true function is a polynomial then of course a polynomial sieve works better. However, sieves differ for example in their properties at the limits of the set, $[a; b]$, over which the coefficients have been estimated; whereas a polynomial sieve diverges outside the set, a Gaussian sieve does not. Similarly, there exist sieves that are relevant if the researcher knows

[6] If for example the true function is a 4th order polynomial and you work with a polynomial sieve, then $h_o \in \mathcal{H}_D$ for all $D \geq 4$.

something about the function h_o a priori, such that the function should map into $[0;1]$, be monotonous, have a certain smoothness or non-smoothness.

3.4 Optimal Choice of Sieve Space Dimension (D)

Recall from kernel regression that in the choice of the bandwidth, the econometrician was faced with a tradeoff between *bias* and *variance*; if the bandwidth was chosen too small, the kernel estimator becomes jittery and looks like it simply "connects the data dots". This can be seen by the golden line in Figure 4. If the bandwidth is too large, the kernel estimator starts to look more like the unconditional average of y no matter the value of z (the purple line in Figure 4). In Figure 4, the true function is

$$h_o(z) = 5\sin(1.3z) + \exp(z - .1z^2).$$

Silverman suggested setting the *optimal bandwidth* for a Kernel estimator according to the rule

$$b = 1.059N^{-0.2}\min\left[\text{std}(z), Q_{0.75}(z) - Q_{0.25}(z)\right], \tag{5}$$

where $Q_p(z)$ denotes the p'th quantile of the random variable z. The rule is appropriate in a Gaussian setting. Figure 4 shows the cross-validation criterion for different values of the bandwidth. The optimal bandwidth occurs at $b \cong 0.2$ and we see that the bandwidth suggested by Silverman's rule is slightly larger but still fairly close.

For sieve estimators, a similar tradeoff exists between overfitting the individual observations in the finite sample (variance) and using sufficient information to capture the functional form (bias). However, no simple rule like (5) exists. Instead, one can use *cross-validation* to determine the optimal dimension. The basic idea in cross-validation is to explore how important each individual observation is for the estimated function; if we drop a single observation, we would still hope that the error for that particular data point would not become too large. If that is the case, we are perhaps over-fitting the neighboring points.

A simple cross-validation method is the *Jackknife cross-validation* criterion; here we sequentially drop a single observation and estimate the function using the remaining data points. We then calculate the squared error between the predicted value at the dropped observation compared to the true one (which we left out). When we have done this for all observations, we calculate the average squared error. To be more precise, suppose we are estimating a non-parametric function, so that $\mathbb{E}(y_i|z_i) = h_o(z_i)$.

Table 1: Examples of Sieve Spaces for Approximating $h_o : [a; b] \to \mathbb{R}$

Polynomial	These take the form $$h(z;\alpha) = \sum_{d=0}^{D} \alpha_d z^d.$$ They approximate smooth functions very well and are easy to program. Intuitively, you can think of this as a Taylor-approximation to the function. Note that $\lvert h(z;\alpha)\rvert \to \infty$ when $\lvert z\rvert \to \infty$.
Gaussian	This sieve has elements of the form $$h(z;\alpha) = \sum_{d=1}^{D} \alpha_d \phi^{(d)}(z),$$ where $\phi^{(d)}(\cdot)$ is the d'th derivative of the Gaussian density. The c'th derivative takes the form $\phi^{(d)}(z) = H_d(z)\phi(z)$, where $H_d(z)$ is a d'th order polynomial (the *Hermite* polynomial).[7] This means that $h(z;\alpha) \to 0$ when $z \to \pm\infty$, contrary to polynomial functions which diverge.
Linear spline	Linear splines are piecewise linear functions. They arise by picking a set of nodes, $\{n_d\}_{d=1}^{D}$, and constructing $$h(z;\alpha) = \alpha_0 + \sum_{d=1}^{D} \alpha_d (z - n_d)\mathbf{1}_{\{z \geq n_d\}}.$$ For example, one might set $n_1 = \min_i(z_i)$ and $n_D = \max_i(z_i)$ and pick the remaining nodes equidistantly between these. The resulting approximating function will be piecewise linear and have kink-points at each of the nodes.
Quadratic splines	Similarly to the linear counterparts, quadratic splines are piecewise quadratic functions with kinks at the nodes, $$h(z;\alpha) = \alpha_0 + \sum_{d=1}^{D} \left[\alpha_{1d}(z - n_d)\mathbf{1}_{\{z \geq n_d\}}\right] + \sum_{d=1}^{D} \left[\alpha_{2d}(z - n_d)^2\mathbf{1}_{\{z \geq n_d\}}\right].$$ The function h will be differentiable and smooth on each sub-interval $(n_d; n_{d+1})$, but will have a jump in the derivative at the nodes. There exists splines that are also differentiable at the nodes but these are more complicated, see Judd (1998).

Figure 3: Kernel Regression: Bandwidth Choice

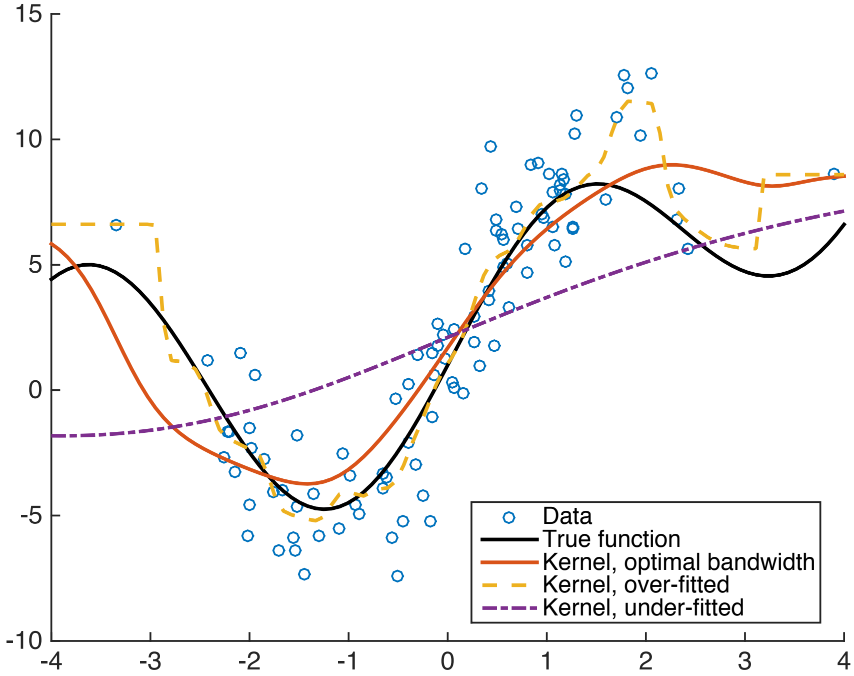
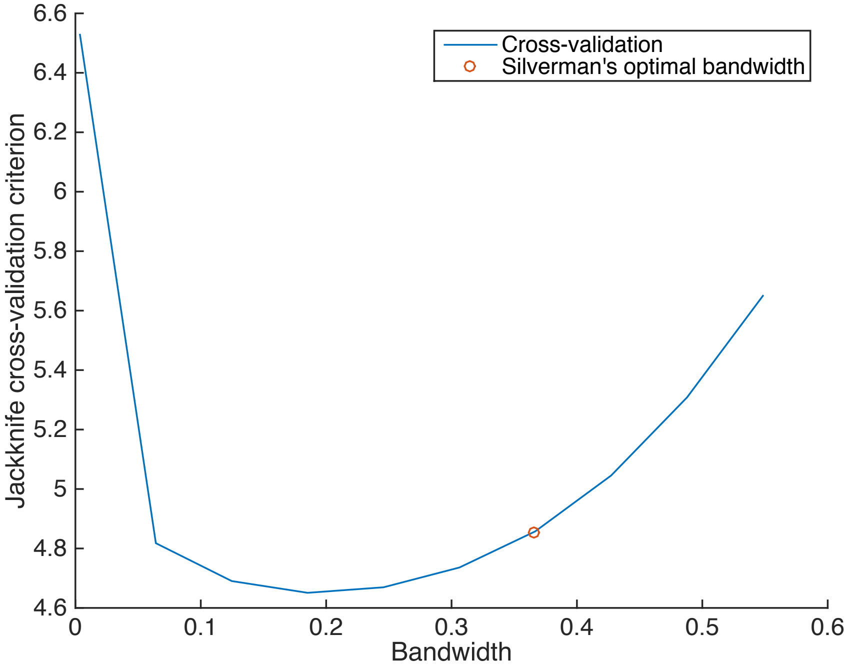


Figure 4: Kernel Regression: Bandwidth Choice by Cross-validation



Procedure (Jackknife cross-validation).

1. For each i:

  1.a: Estimate $\hat{h}_{[i]}(\cdot)$ by a sieve estimator with dimension D using the dataset $\{y_j, z_j\}_{j \neq i}$,

  1.b: Compute $\hat{e}_i := y_i - \hat{h}_{[i]}(z_i)$, where $\hat{h}_{[i]}(z_i)$ is our prediction for y_i for individual i given the estimated function $\hat{h}_{[i]}(\cdot)$.

2. Return $\text{CV}_D = N^{-1} \sum_{i=1}^{N} \hat{e}_i^2$.

3. Compute CV_D for a range of grid over possible values of $D \in \{D_1, ..., D_K\}$.

4. The preferred dimension is $D^* = \min_{D \in \{D_1,...,D_K\}} \text{CV}_D$, that is the dimension that gives the lowest cross-validation error.

Intuitively, if D is picked very low, we do not allow enough flexibility and get a high error due to *bias*; on the other hand, if D is too high, the function changes a lot when we drop an observation i, leading to a high error for the left-out observation. This indicates error due to too high *variance* of the prediction. Jackknife cross-validation is also referred to as *leave-one-out cross-validation*. The method can be overly costly computationally when N is very large; for that purpose, there exists alternative cross-validation criteria, where more than one observation is left out each time and there are different ways of picking which observations to exclude with different properties. These may reduce computational cost and can sometimes be just as good, sometimes not.

Figure 5 shows the cross-validation criterion for a range of values of the sieve dimension $D \in \{1, 2, ..., 13\}$. It turns out that $D^* = 7$ is the optimal dimension with $N = 1000$ observations, although $D = 5, 6, 7, 8, 9$ are all fairly close in criterion value. We note that the cross-validation is much less smooth than that for the kernel regression.

Figure 6 shows three different sieve estimators with $N = 100$ observations; one using the optimal dimension, one using too low D (under-fitting), and one using too high D (over-fitting). In particular in the ends of the support of z, we see that the over-fitted sieve has some very extreme movements in order to perfectly fit each of the data points. This means that when we sequentially drop these points, the curve will change dramatically, resulting in a bad cross-validation criterion. Conversely, the under-fitted graph doesn't capture enough of the curvature in the data.

Figure 5: Sieve Estimation: Dimension Choice

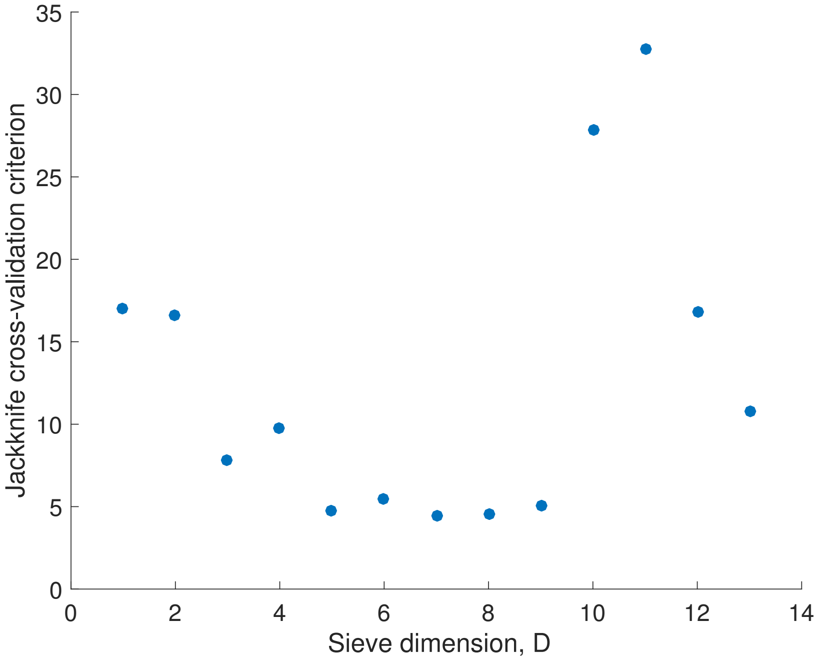
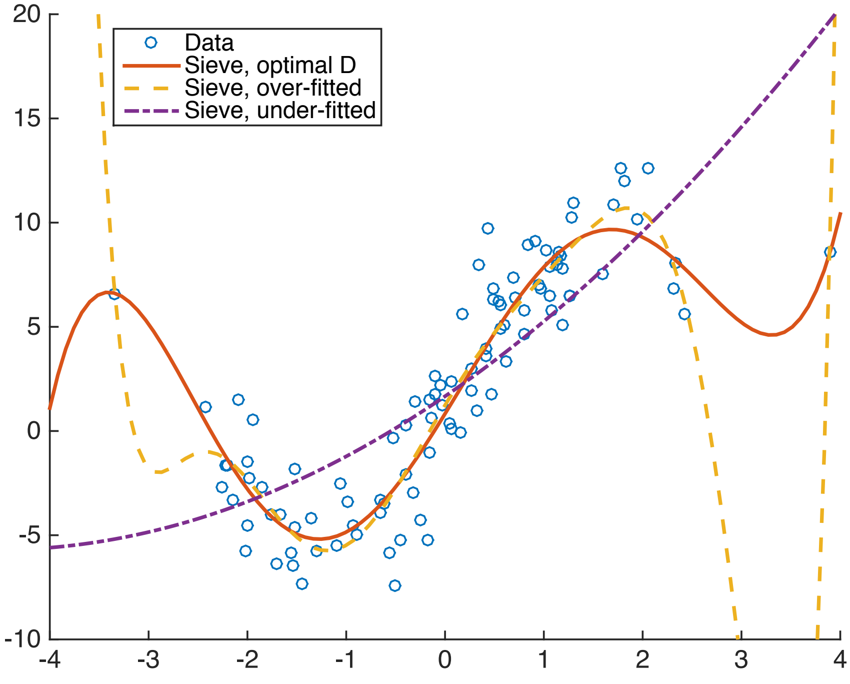


Figure 6: Sieve Estimation: Dimension Choice



3.5 Convergence Speed

Recall that in the standard parametric M-estimator framework, we obtain $\sqrt{N}$-consistent, asymptotically normal estimates of the parameters of the mode. However, in the fully nonparametric scenario, the best-case convergence rate was $N^{2/5}$ (if h_o is twice-differentiable and if the bandwidth is set correctly). Surprisingly, it turns out that in fairly general settings, it is possible to obtain $\sqrt{N}$-consistent estimators of β_o, in spite of the noise introduced by the fact that h will be estimated. This is in particular the case for the partially linear model, if the dimensionality grows at the "optimal rate", $D = D^*(N)$.

On the other hand, the rate of convergence of $\hat{h}$ to h_o is a much more complicated matter. This depends both on the properties one assumes about h_o (typically, how many times differentiable h_o is, the "smoothness") and on the properties of the sieve space employed; in particular, the so-called *entropy* of the sieve space, which must not grow too rapidly when the dimension grows. However, these are more theoretical concerns important for establishing convergence and asymptotic distributions than something the applied microeconometrician needs to worry about so long as common sieve spaces are used. The interested reader is referred to Chen (2007) for a more precise coverage of sieve estimation.

Nevertheless, due to the simultaneous estimation, sieve estimators tend to be more efficient than two-step estimators such as the Robinson double residual estimator.

References

Chen, X. 2007. "Large sample sieve estimation of semi-nonparametric models." *Handbook of Econometrics* 6:5549–5632.

Judd, K.L. 1998. *Numerical methods in economics.* MIT press.

Robinson, P.M. 1988. "Root-N-consistent semiparametric regression." *Econometrica: Journal of the Econometric Society* :931–954.