Semiparametric and Nonparametric Methods in Political Science

Lecture 3: Applications of Nonparametric Techniques

Michael Peress, University of Rochester and Yale University

Overview

- In lecture 1, we considered "easy" semiparametric estimators that did not require estimating an infinite dimensional quantity
  - These estimators retained parametric ($\sqrt{N}$) convergence rates
- In lecture 2, we considered nonparametric density estimation
  - In the one dimensional case, convergence was slower than ($\sqrt{N}$)
  - In higher dimensions, convergence became slower and slower ($N^{2/(4+d)}$)
- In lecture 3, we will cover nonparametric regression techniques
  - $y_n = g_0(x_n) + \varepsilon_n$ (nonparametric regression)
  - $\Pr(y_n = 1 | x_n) = G_0(x_n)$ (nonparametric binary choice)
  - In both cases, curse of dimensionality ($N^{2/(4+d)}$)

Overview

- In lecture 3, we will apply the “toolbox” from lecture 2 to study “hard” semiparametric and nonparametric problems
  - In some cases, we can estimate the parameter of interest at the parametric rate ($\sqrt{N}$), but require estimating an infinite dimensional quantity in the process (semiparametric estimation)
    - $y_n = g_0(\beta_0' x_n) + \varepsilon_n$ (linear index model)
    - $\Pr(y_n = 1 | x_n) = G_0(\beta_0' x_n)$ (semiparametric binary choice)
  - In other cases, the parameter of interest will be infinite dimensional, but we will control for nuisance variables using a parametric component
    - $y_n = g_0(x_n) + \beta_0' z_n + \varepsilon_n$ (partially linear model)

Kernel Regression

- Consider the relationship, $y_n = g_0(x_n) + \varepsilon_n$, where (x_n, ε_n) are iid and $E[\varepsilon_n \mid x_n] = 0$.
- The (locally constant) Kernel estimator is defined by,

$$\hat{g}(x;h) = \tfrac{1}{N}\sum_{n=1}^{N} w_n(x;h)\, y_n$$

where $w_n(x;h) = \dfrac{\frac{1}{h}K\left(\frac{x-x_n}{h}\right)}{\frac{1}{Nh}\sum_{m=1}^{N} K\left(\frac{x-x_m}{h}\right)}$

- Motivation: to evaluate $E[y_n \mid x]$, look at average value of y_n for x_n ‘s that are close to x (weighted by their closeness)

Kernel Regression

- Why it works (heuristic proof of consistency):

$$\hat{g}(x) = \frac{\frac{1}{hN}\sum_{n=1}^{N} K\left(\frac{x-x_n}{h}\right) Y_n}{\frac{1}{hN}\sum_{m=1}^{N} K\left(\frac{x-x_m}{h}\right)} = \frac{\frac{1}{hN}\sum_{n=1}^{N} K\left(\frac{x-x_n}{h}\right) g_0(x_n)}{\frac{1}{hN}\sum_{m=1}^{N} K\left(\frac{x-x_m}{h}\right)} + \frac{\frac{1}{hN}\sum_{n=1}^{N} K\left(\frac{x-x_n}{h}\right) \varepsilon_n}{\frac{1}{hN}\sum_{m=1}^{N} K\left(\frac{x-x_m}{h}\right)}$$

$$\downarrow_{LLN} \qquad\qquad \downarrow_{LLN}$$

$$\frac{\frac{1}{hN}\frac{1}{h}\int_{x'} K\left(\frac{x-x'}{h}\right) g_0(x') f_0(x') dx'}{\frac{1}{h}\int_{x'} K\left(\frac{x-x'}{h}\right) f_0(x') dx'} \qquad \frac{0}{\frac{1}{h}\int_{x'} K\left(\frac{x-x'}{h}\right) f_0(x') dx'}$$

$$\downarrow_{h\to 0} \qquad\qquad \downarrow_{h\to 0}$$

$$\frac{g_0(x) f_0(x)}{f_0(x)} \qquad\qquad \frac{0}{f_0(x)}$$

- Hence, $\hat{g}(x) \xrightarrow{prob.} g_0(x)$

Kernel Regression

- Kernel regression estimators have many properties similar to kernel density estimators

$$Bias[\hat{g}(x;h)] = \tfrac{h^2}{2 f_0(x)} \mu_2 \left(g_0''(x) f_0(x) + 2 g_0'(x) f_0'(x) \right) + O(N^{-1}h^{-1}) + o(h^2)$$

$$Var(\hat{g}(x;h)) = \tfrac{\sigma_0^2(x)}{Nhf_0(x)} \nu_2 + o(N^{-1}h^{-1})$$

- We can derive the IMSE to be,

$$IMSE(\hat{g}) = \tfrac{1}{Nh} \nu_2 \int_x \frac{\sigma_0^2(x)}{f_0(x)} dx + h^4 \mu_2^2 \int_x \left(\frac{g_0''(x) f_0(x)}{2 f_0(x)} + \frac{g_0'(x) f_0'(x)}{f_0(x)} \right)^2 dx$$

$$+ o(h^4) + o(N^{-1}h^{-1})$$

Kernel Regression

- Minimizing this expression yields,

$$h^* = \left(\frac{\nu_2 \int_x \frac{\sigma_0^{\,2}(x)}{f_0(x)} dx}{\mu_2^{\,2} \int_x \left(\frac{g_0''(x) f_0(x) + 2 g_0'(x) f_0'(x)}{f_0(x)} \right)^2 dx} \right)^{1/5} N^{-1/5}$$

- We therefore find that the IMSE has an error of $O(N^{-4/5})$ when the optimal bandwidth is selected
- Plug in rule is <u>really</u> messy
- Normal reference rule won't really work since we would still have to guess g_0

Kernel Regression

- Cross validation:

$$CV(h) = \tfrac{1}{N}\sum_{n=1}^{N}(y_n - \hat{g}_{(n)}(x_n;h))^2$$

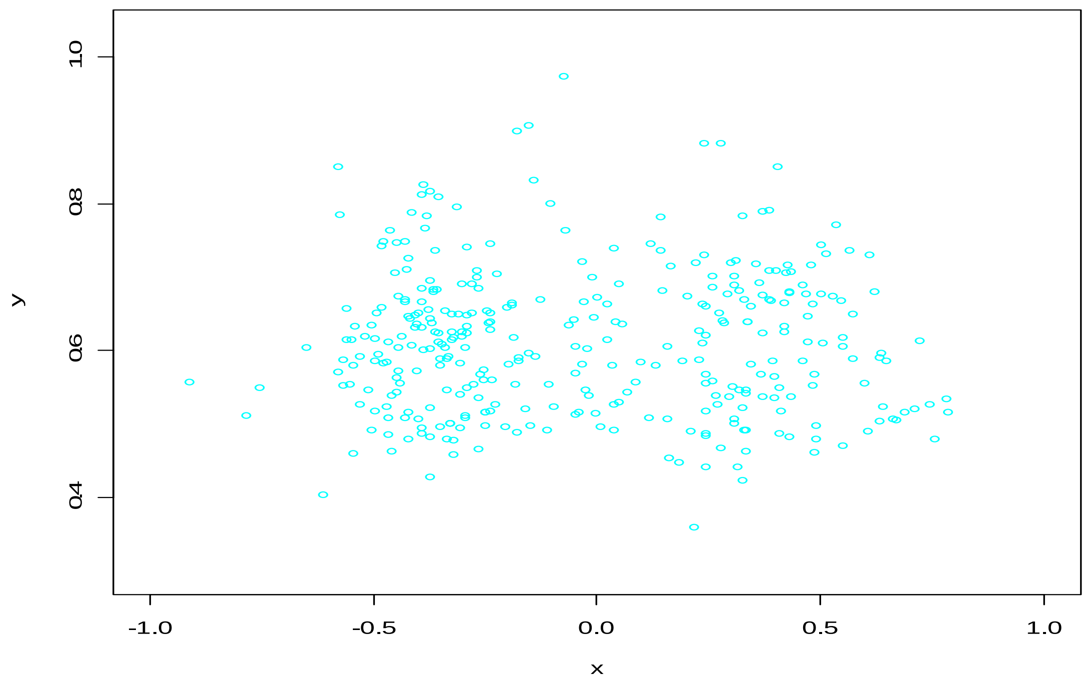
where,

$$\hat{g}_{(n)}(x;h) = \frac{\frac{1}{Nh}\sum_{m\neq n} K\left(\frac{x-X_m}{h}\right) y_m}{\frac{1}{Nh}\sum_{m\neq n} K\left(\frac{x-X_m}{h}\right)}$$

- Easy alternative: use ROT for density f_0 even though this rule is not specifically optimized for kernel regression

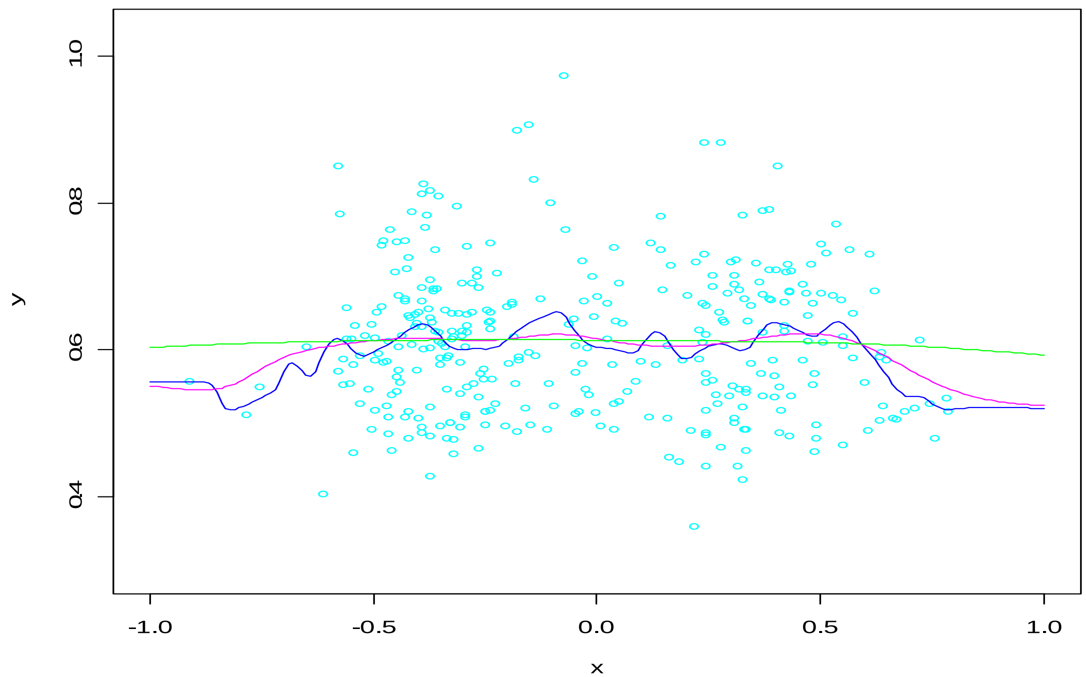
Kernel Regression

- Example: Effect of Position on Vote Share for Senate Incumbents



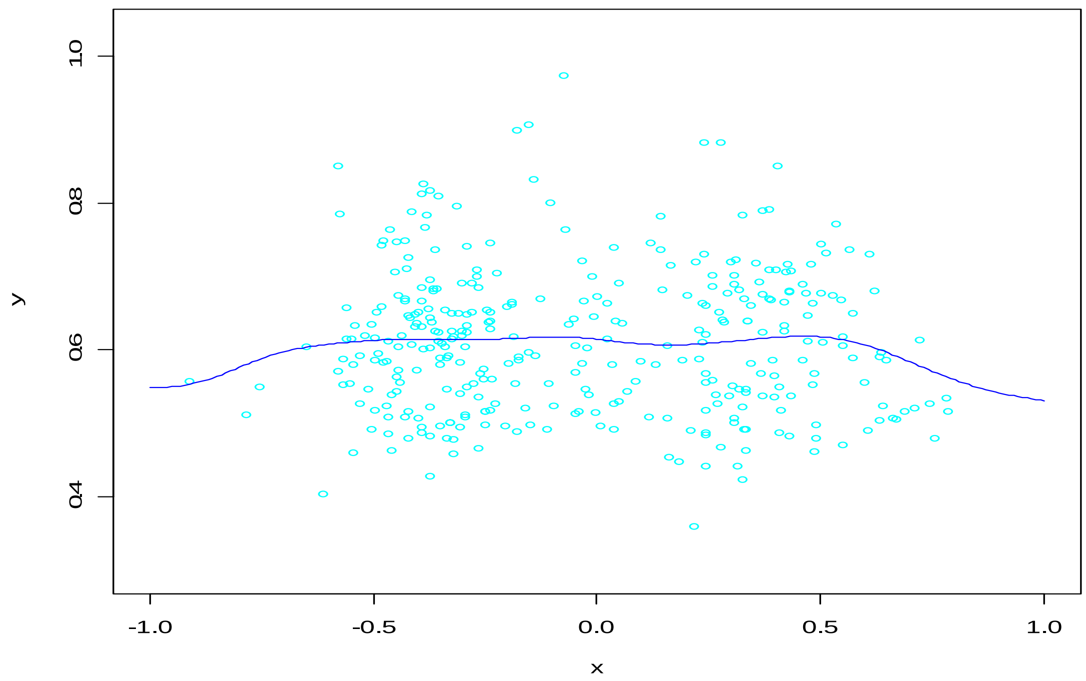
Kernel Regression

- Example: Effect of Position on Vote Share for Senate Incumbents (h=.03, .01, and .3)



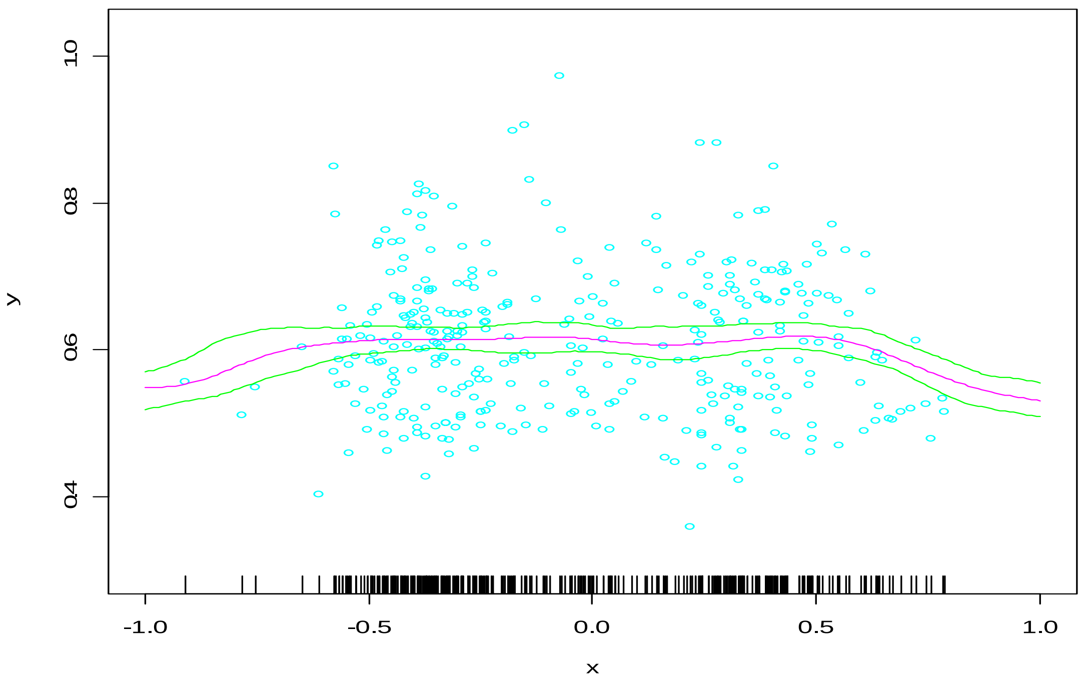
Kernel Regression

- Example: Effect of Position on Vote Share for Senate Incumbents (hROT=0.126)



Kernel Regression

- Example: Effect of Position on Vote Share for Senate Incumbents w/ Bootstrapped Standard Errors:



Kernel Regression

- For the Locally Linear estimator, we have,

$$\hat{f}(x) = \hat{\beta}_0(x) + \hat{\beta}_1(x)x$$

where,

$$(\hat{\beta}_0(x), \hat{\beta}_1(x)) = \underset{(\beta_0,\beta_1)}{\arg\min} \tfrac{1}{N}\sum_{n=1}^{N} w_n(x;h)(y_n - \beta_0 - \beta_1 x_n)^2$$

- We can determine that,

$$Bias[\hat{g}(x;h)] = \tfrac{1}{2}\mu_2 h^2 g_0''(x) + O(N^{-1}h^{-1}) + o(h^2)$$

$$Var(\hat{g}(x;h)) = \tfrac{\sigma_0^2(x)}{Nhf_0(x)}\nu_2 + o(N^{-1}h^{-1})$$

Kernel Regression

- We can derive the IMSE to be,

$$IMSE(\hat{g}) = \tfrac{1}{Nh}\nu_2 \int_x \frac{\sigma_0{}^2(x)}{f_0(x)} dx + \tfrac{1}{4}\mu_2{}^2 h^4 \int_x (g_0 "(x))^2 dx + o(h^4) + o(N^{-1}h^{-1})$$

- Minimizing this expression yields,

$$h^* = \left(\frac{\nu_2 \int_x \frac{\sigma_0{}^2(x)}{f_0(x)} dx}{\mu_2{}^2 \int_x (g_0 "(x))^2 dx} \right)^{-1/5} N^{-1/5}$$

- The variance is the same as the locally constant estimator, but the bias is different
- The bias of the estimator will tends towards OLS rather than a flat curve

Multivariate Kernel Regression

- Consider the relationship, $y_n = g_0(x_n) + \varepsilon_n$, where (x_n, ε_n) are iid and $E[\varepsilon_n \mid x_n] = 0$
- The (locally constant) Kernel estimator is defined by,

$$\hat{g}(x;h) = \tfrac{1}{N} \sum_{n=1}^{N} w_n(x;h) y_n$$

where,

$$w_n(x;h) = \frac{\frac{1}{h^d} \prod_{i=1}^{d} K\left(\frac{x_i - x_{n,i}}{h}\right)}{\frac{1}{Nh^d} \sum_{m=1}^{N} \prod_{i=1}^{d} K\left(\frac{x_i - x_{m,i}}{h}\right)}$$

Kernel Binary Choice

- Consider the relationship, $\Pr(y_n = 1 | x_n) = G_0(x_n)$
- The Kernel estimator is defined by,

$$\hat{G}(x;h) = \tfrac{1}{N} \sum_{n=1}^{N} w_n(x;h) y_n$$

where,

$$w_n(x;h) = \frac{\frac{1}{h^d} \prod_{i=1}^{d} K\left(\frac{x_i - x_{n,i}}{h}\right)}{\frac{1}{Nh^d} \sum_{m=1}^{N} \prod_{i=1}^{d} K\left(\frac{x_i - x_{m,i}}{h}\right)}$$

- Notice that this is the same estimator and the Kernel regression estimator

Semiparametric Binary Choice

- Parametric binary choice (i.e. probit)

$$\Pr(y_n = 1 \mid x_n) = \Phi(\beta' x_n)$$

- Nonparametric binary choice

$$\Pr(y = 1 \mid x) = G_0(x)$$

- Semiparametric binary choice

$$\Pr(y_n = 1 \mid x_n) = G_0(\beta' x_n)$$

Semiparametric Binary Choice

- Why consider semiparametric binary choice model?
  - Parametric binary choice (i.e. probit)

$$\Pr(y_n = 1 \mid x_n) = \Phi(\beta' x_n)$$

  - Marginal effect of x_k

$$\tfrac{\partial}{\partial x_k} \Pr(y_n = 1 \mid x) = \beta_k \phi(\beta' x)$$

  - Magnitude of $\beta_k \phi(\beta' x)$ is largest when $\beta' x = 0$
  - Same will hold for any symmetric unimodal density
  - Fully nonparametric model is too general (hard to report results) and suffers from curse of dimensionality

Semiparametric Binary Choice

- For semiparametric binary choice model,

$$\tfrac{\partial}{\partial x_k}\Pr(y_n = 1 \mid x) = \beta_k g(\beta' x)$$

- If g is not symmetric, then $\tfrac{\partial}{\partial x_k}\Pr(y_n = 1 \mid x)$ need not peak when $\beta' x = 0$
- In an application to campaigning, this assumption implies that moderate voters are most sensitive to campaigning (maybe)
- In an application to GOTV, this implies that voters with a predicted probability of voting of 0.5 are most sensitive to GOTV (maybe)
- These are assumptions!
- Semiparametric binary choice model allows us to relax/test these assumptions

Semiparametric Binary Choice

- It would be nice if there were an "easy" semiparametric estimator that did not require estimating G_0 (i.e. like OLS w/ robust standard errors does not require estimating $E[\varepsilon_n^{\,2} \mid x_n]$ in order to deal w/ heteroskedasticity)
- Manski's maximum score estimator is an attempt to provide such an estimator

$$\hat{\beta} = \underset{\beta:\beta'\beta=1}{\arg\min} \tfrac{1}{N} \sum_{n=1}^{N} 1\{y_n = 1\}1\{\beta' x_n > 0\} + 1\{y_n = 0\}1\{\beta' x_n < 0\}$$

- The Maximum Score Estimator is consistent under the assumption that $G_0(0) = 0$ (i.e. the median error is normalized to zero)
- It is consistent even when the errors exhibit non-normality/time series dependence
- Great, right?

Semiparametric Binary Choice

- First drawback of MSE:
  - It does not provide an estimate of G_0, which is a problem, if G_0 is of interest (i.e. testing for asymmetric campaign effects)
  - As long as G is a nuisance parameter (i.e. as long as we are only interested in robustness to asymmetric campaign effects), doesn't matter

Semiparametric Binary Choice

- Second drawback of MSE:
  - Theory is “weird”
  - The estimator is consistent, under very broad assumptions about error term in nonparametric probit model
  - The estimator converges slowly ($N^{1/3}$)
  - The estimator is not asymptotically normal (Kim and Pollard, 1989)
  - Even worse, bootstrap is inconsistent for MSE (Abrevaya and Huang, 2005)

Semiparametric Binary Choice

- The Semiparametric Kernel Estimator:
  - Define $z_n = \beta' x_n$
  - If we knew β, we would have $\Pr(y_n = 1 | z_n) = G_0(z_n)$
  - We can form,

$$\hat{G}(z) = \hat{P}(y_n = 1 | z) = \frac{\frac{1}{hN}\sum_{n=1}^{N} K\left(\frac{z_n - z}{h}\right) y_n}{\frac{1}{hN}\sum_{n=1}^{N} K\left(\frac{z_n - z}{h}\right)}$$

Semiparametric Binary Choice

- Approach we use embeds kernel estimator in log-likelihood
- Consider the estimator given by,

$$\hat{\beta} = \underset{\beta:\beta_0=0,\beta_1=1}{\arg\min} \tfrac{1}{N} \sum_{n=1}^{N} y_n \log \hat{G}(\beta' x_n;\beta) + (1-y_n)\log(1-\hat{G}(\beta' x_n;\beta))$$

where,

$$z_n(\beta) = \beta' x_n$$

$$\hat{G}(z;\beta) = \frac{\frac{1}{hN}\sum_{n=1}^{N} K\left(\frac{z_n(\beta)-z}{h}\right) y_n}{\frac{1}{hN}\sum_{n=1}^{N} K\left(\frac{z_n(\beta)-z}{h}\right)}$$

- We must impose the restrictions $\beta_0 = 0$ and $\beta_1 = 1$ for identification

Semiparametric Binary Choice

- The estimator is $\sqrt{N}$ -consistent and asymptotically normal for β_0
- Large sample properties of this (and many other estimators) follow from Andrew's (1994) MINPIN theorem
- Define,

$$\hat{\beta} \in \arg\max_{\beta \in \mathrm{B}} \tfrac{1}{N} \sum_{n=1}^{N} \psi(x_n, y_n; \beta, \hat{G}(\beta))$$

- We have,

$$\sqrt{N}(\hat{\beta} - \beta_0) \xrightarrow{dist.} N(0, Q_{\beta\beta}{}^{-1} V_{\beta}^{Q} Q_{\beta\beta}{}^{-1})$$

where,

$$Q_{\beta\beta} = E\left[\tfrac{\partial^2 \psi}{\partial \beta^2}(x_n, y_n; \beta_0, G_0)\right], \quad V_{\beta}^{Q} = E\left[\tfrac{\partial \psi}{\partial \beta}(x_n, y_n; \beta_0, G_0) \tfrac{\partial \psi}{\partial \beta}(x_n, y_n; \beta_0, G_0)'\right]$$

Semiparametric Binary Choice

- Where can then estimate,

$$\hat{Q}_{\beta\beta} = \tfrac{1}{N}\sum_{n=1}^{N} \tfrac{\partial^2\psi}{\partial\beta^2}(x_n, y_n; \hat{\beta}, \hat{G}), \qquad \hat{V}_{\beta}^{Q} = \tfrac{1}{N}\sum_{n=1}^{N} \tfrac{\partial\psi}{\partial\beta}(x_n, y_n; \hat{\beta}, \hat{G})\tfrac{\partial\psi}{\partial\beta}(x_n, y_n; \hat{\beta}, \hat{G})'$$

- Alternatively, we can use the bootstrap to conduct inference about β_0 and G_0 (especially if G_0 is of direct interest)

Semiparametric Binary Choice

- Selecting the Bandwidth:
  - "Lazy" rule of thumb:
    - For each β, form $z_n = \beta' x_n$
    - Compute h based on normal reference rule for the density of z_n (notice that there is a different h each time the objective function is evaluated at β
    - This is an ad-hoc rule, but at least you get the rates correct (i.e. $h = cN^{-1/5}$)
  - Plug in rule not available (as far as I know) because asymptotic formulas get VERY complicated

Semiparametric Binary Choice

- Cross Validation:

$$\tfrac{1}{N}\sum_{n=1}^{N} y_n \log \hat{G}_{(n)}(z_n;h) + (1-y_n)\log(1-\hat{G}_{(n)}(z_n;h))$$

where $\hat{G}_{(n)}$ is the leave-one-out estimator

- Two approaches:
  - Iterate between MLE and CV
  - Simultaneously maximize
  - I don’t really recommend either approach

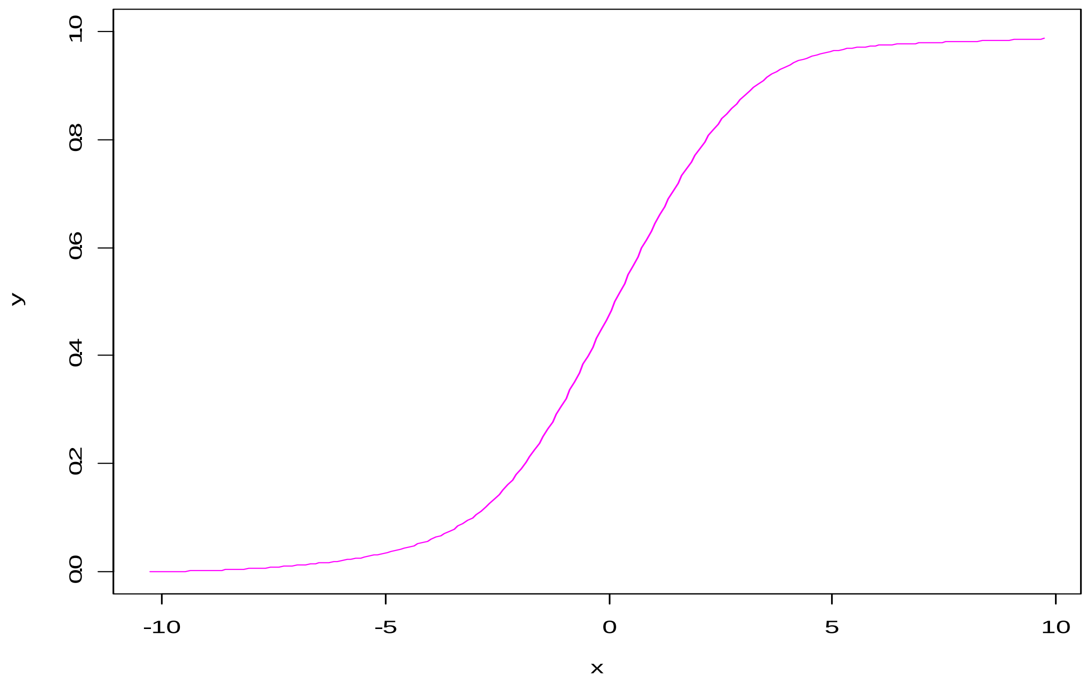
Semiparametric Binary Choice

- Example: Semiparametric Model of Presidential Vote in 2004 Election (coefficient estimates)

	Probit		Probit (Normalized)		Semiparametric (Normalized)		
	est.	se	est.	se	est.	se	boot se
(Intercept)	-0.056	(0.146)	0.000		0.000		
Prox. Diff.	0.393	(0.044)	1.000		1.000		
Party Dem.	-0.967	(0.228)	-2.460		-2.341	(0.949)	[0.811]
Party Rep.	1.049	(0.210)	2.668		2.766	(0.652)	[0.699]
Black	-1.336	(0.361)	-3.398		-3.374	(1.022)	[0.992]
Female	0.161	(0.171)	0.411		0.427	(0.424)	[0.452]
South	0.227	(0.196)	0.577		0.553	(0.459)	[0.485]

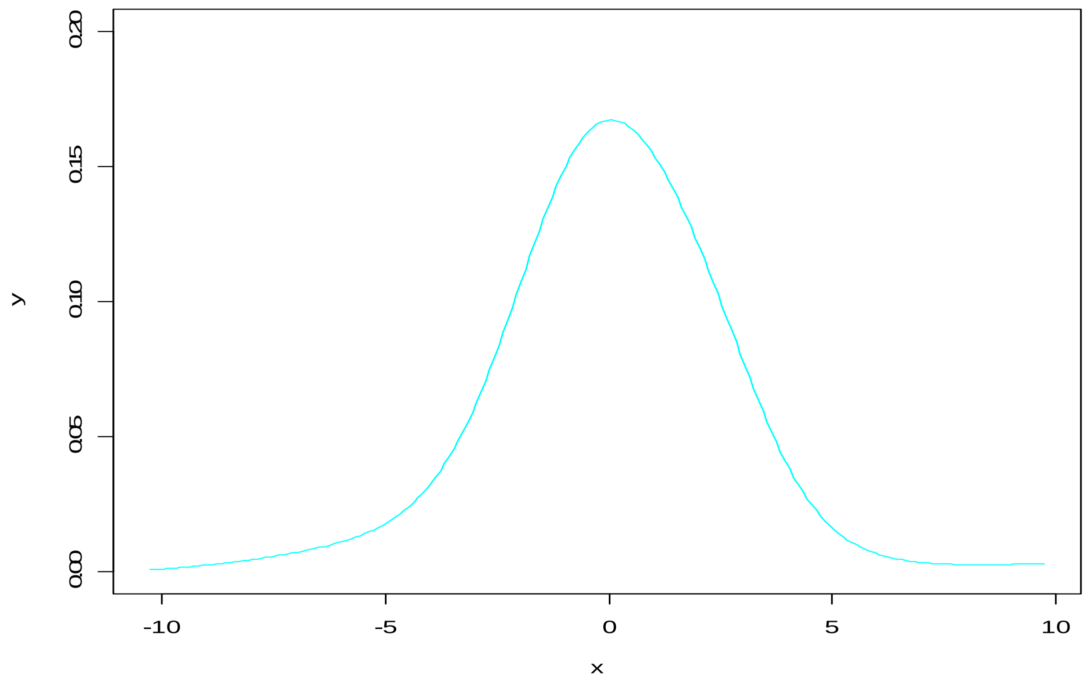
Semiparametric Binary Choice

- Example: Semiparametric Model of Presidential Vote in 2004 Election (estimate of G)



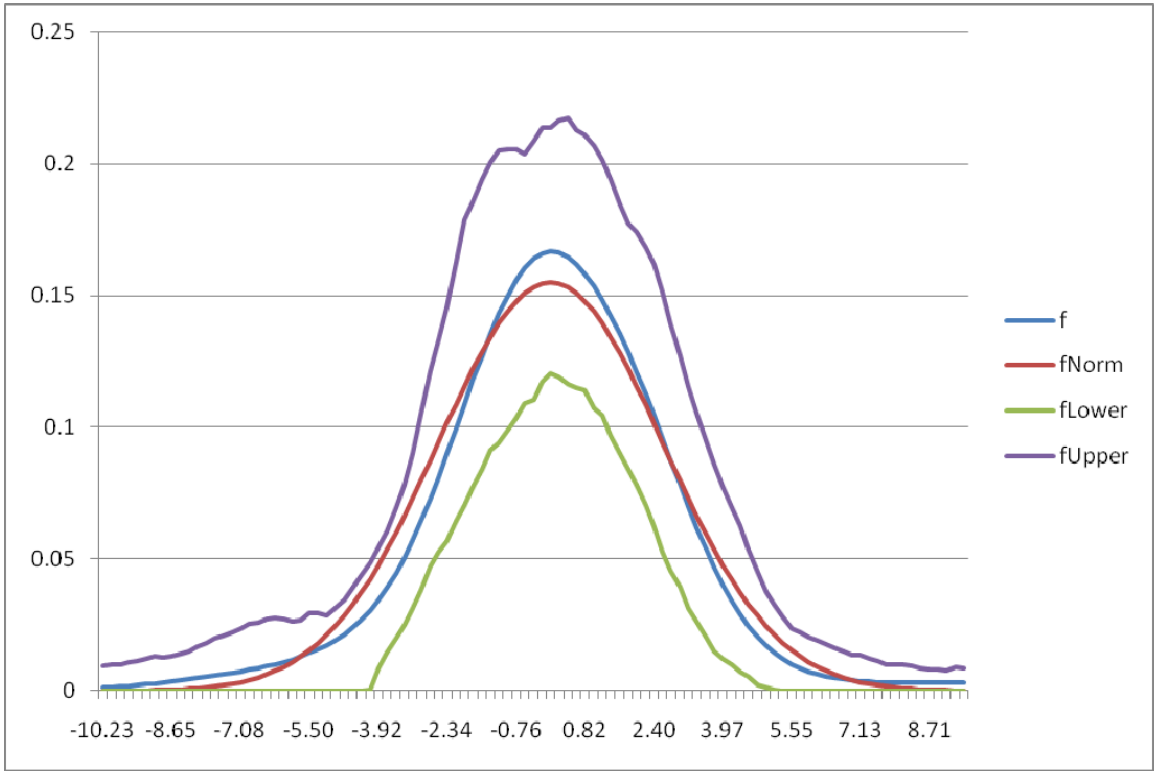
Semiparametric Binary Choice

- Example: Semiparametric Model of Presidential Vote in 2004 Election (estimate of g)



Semiparametric Binary Choice

- Example: Semiparametric Model of Presidential Vote in 2004 Election (estimate of g w/ bootstrapped CIs)



Semiparametric Binary Choice

- Back to Marginal Effects:

$$\tfrac{\partial}{\partial x_k}\Pr(y_n = 1 \mid x) = \beta_k g(\beta' x)$$

  - Marginal effects depend on estimate of g
  - Consider (weighted) average marginal effect

$$\theta_k = \tfrac{\partial}{\partial x_k}\int_x \Pr(y_n = 1 \mid x) w(x) dx = \beta_k \int_x g(\beta' x) w(x) dx$$

  - $\hat{\theta}_k = \hat{\beta}_k \int_x \hat{g}(\hat{\beta}' x) w(x) dx$ is $\sqrt{N}$ –consistent (average derivatives can be estimated at the parametric rate)

Semiparametric Binary Choice

- This is a very generalizable principal- even when we require preliminary estimates of infinite dimensional quantities, we are ultimately interested in finite dimensional quantities, which can often be estimated at the parametric rate
- Exceptions are statistics which are not smooth (such as the mode)

Partially Linear Models

- Kernel regression estimators are of limited use on their own
- Most social science applications involve multiple explanatory variables
- As with binary choice, fully nonparametric approach suffers from the curse of dimensionality
- Partially linear model provide a way of having multiple regressors with one degree of nonparametrics

$$y_n = \beta_0' z_n + g_0(x_n) + \varepsilon_n$$

- If β_0 is of interest and g_0 is a nuisance parameter, we have a semi-parametric model
- If g_0 is of interest and β_0 is a nuisance parameter, we have a low-dimensional nonparametric model (picture w/ controls)

Partially Linear Models

- Suppose that we knew the value of β, we could define $w_n = y_n - \beta' z_n$ and consider the model, $w_n = g_0(x_n) + \varepsilon_n$, applying the Kernel regression estimator

$$\hat{g}(x;\beta) = \frac{\frac{1}{hN}\sum_{n=1}^{N} K\left(\frac{x_n - x}{h}\right)(y_n - \beta' z_n)}{\frac{1}{hN}\sum_{n=1}^{N} K\left(\frac{x_n - x}{h}\right)}$$

- We can plug this estimator into the equation above to obtain $y_n - \hat{g}(x_n;\beta) = \beta' z_n + \varepsilon_n$
- We can then estimate β_0 using least squares,

$$\hat{\beta} = \arg\min_{\beta} \frac{1}{N}\sum_{n=1}^{N} (y_n - \hat{g}(x_n;\beta) - \beta' z_n)^2$$

Partially Linear Models

- For this model (unlike the semiparametric binary choice model) we can apply some computational tricks
  - Let us write,

$$\tilde{y}_n = y_n - \frac{\frac{1}{hN}\sum_{m=1}^{N} K\left(\frac{x_m - x_n}{h}\right) y_m}{\frac{1}{hN}\sum_{m=1}^{N} K\left(\frac{x_m - x_n}{h}\right)}, \qquad \tilde{z}_n = z_n - \frac{\frac{1}{hN}\sum_{m=1}^{N} K\left(\frac{x_m - x_n}{h}\right) z_m}{\frac{1}{hN}\sum_{m=1}^{N} K\left(\frac{x_m - x_n}{h}\right)}$$

  - We have that,

$$\hat{\beta} = \left[\frac{1}{N}\sum_{n=1}^{N} \tilde{z}_n \tilde{z}_n'\right]^{-1} \left[\frac{1}{N}\sum_{n=1}^{N} \tilde{z}_n \tilde{y}_n\right]$$

Partially Linear Models

- Large sample distribution:

$$\sqrt{N}(\hat{\beta}-\beta_0)\xrightarrow{prob.} N(Q_{zz}{}^{-1}V_{z\varepsilon}Q_{zz}{}^{-1})$$

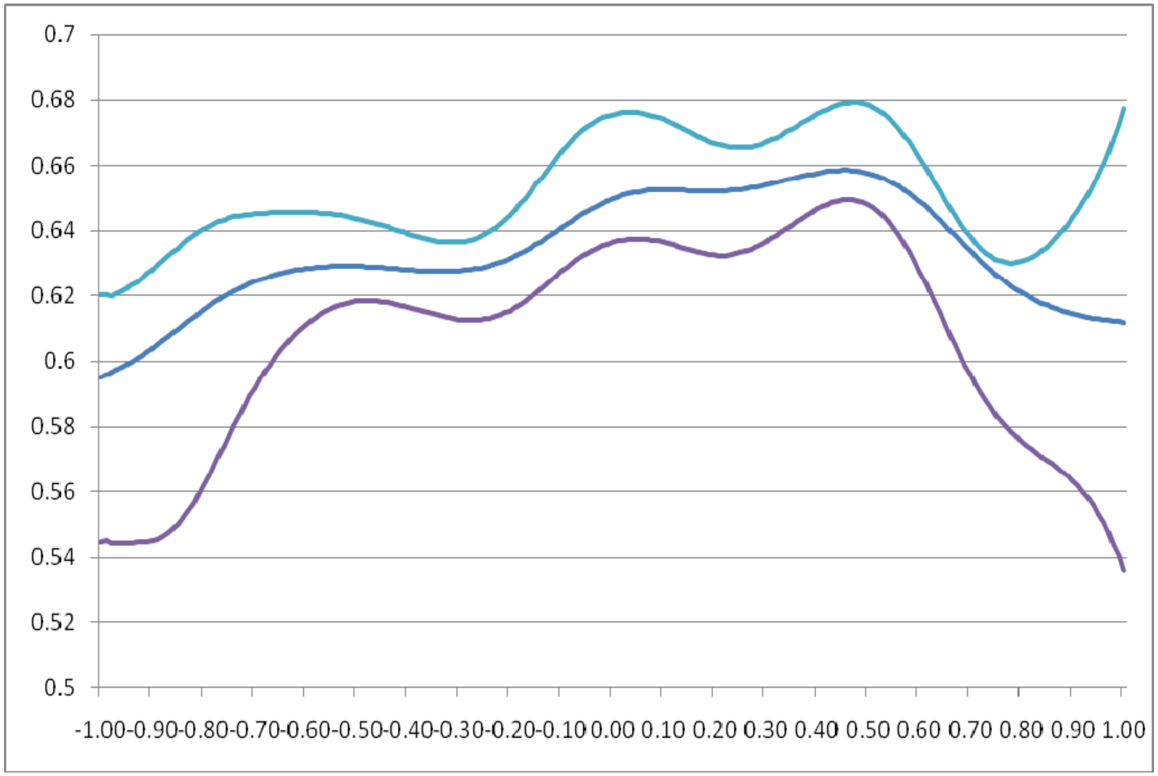
where we can estimate,

$$\tilde{\varepsilon}_n = y_n - \hat{g}(x_n) - \hat{\beta}' z_n, \qquad \hat{Q}_{zz} = \tfrac{1}{N}\sum_{n=1}^{N} \tilde{z}_n \tilde{z}_n', \qquad \hat{V}_{z\varepsilon} = \tfrac{1}{N}\sum_{n=1}^{N} \tilde{\varepsilon}_n{}^2 \tilde{z}_n \tilde{z}_n'$$

- If inferences about g are desired, use bootstrap
- Once again, average marginal effects of x_n can be estimated at parametric rate

Partially Linear Models

- Example: Candidate Positioning in Senate Elections w/ Controls



Partially Linear Models

- Example: Candidate Positioning in Senate Elections w/ Controls

	Beta	Se	Boot. Se
st_pop	0.020	(0.006)	[0.008]
st_south	-0.004	(0.011)	[0.012]
st_unemp	-0.002	(0.005)	[0.002]
inc_dem	0.024	(0.038)	[0.038]
inc_tenure	0.002	(0.002)	[0.001]
inc_spend	-0.005	(0.005)	[0.002]
ch_qual	-0.021	(0.004)	[0.003]
ch_spend	-0.006	(0.003)	[0.002]

Single Index Models

- The single index model is given by,

$$y_n = g_0(\beta_0 ' x_n) + \varepsilon_n$$

- Kernel estimator,

$$\hat{\beta} = \arg\min_{\beta} \tfrac{1}{N} \sum_{n=1}^{N} (y_n - \hat{g}(x_n;\beta))^2$$

where,

$$\hat{g}(x;\beta) = \frac{\frac{1}{hN} \sum_{n=1}^{N} K\left(\frac{\beta'(x_n - x)}{h}\right) y_n}{\frac{1}{hN} \sum_{n=1}^{N} K\left(\frac{\beta'(x_n - x)}{h}\right)}$$

- β can be estimated at rate $\sqrt{N}$

Other Semiparametric Problems

- ATEs in nonparametric models can be estimated at parametric rate
- Most test statistics (e.g. the density is normal, the regression function is monotonic) can be estimated at parametric rate – see Hall and Yatchew (2005)
- Marginal effects that project to entire populations (e.g. average derivatives) can be estimated at parametric rate

Alternative Nonparametric Estimators

- k-Nearest neighbor estimator:
  - Consider the multivariate nonparametric regression problem:

$$y_n = g_0(x_n) + \varepsilon_n$$

  - The k-NN estimator is given by,

$$\hat{g}(x;k) = \tfrac{1}{k}\sum_{n=1}^{N} 1_{nk}(x) y_n$$

    where $I_{nk} = 1 \Leftrightarrow x_n$ is one of the k closest points to x

  - Issues:
    - Selecting k
    - Computation

Alternative Nonparametric Estimators

- Sieve estimator:

$$\hat{g}(x) = \sum_{i=1}^{m} a_i h_i(x)$$

  - Here, $\{h_i(x)\}_{i=1}^{\infty}$ are basis functions
  - For example, if $h_i(x) = x^i$, we have $\hat{g}_0(x) = \sum_{i=1}^{m} a_i x^i$
  - Issues:
    - Selecting m
    - Becomes very complicated in higher dimensions

Alternative Nonparametric Estimators

- Smoothing splines:

$$\hat{g} = \arg\max_{g} \tfrac{1}{N} \sum_{n=1}^{N} (y_n - g(x_n))^2 - \lambda \int_x (g''(x))^2 dx$$

  - Solution is a cubic spline with knots at all the data points
  - Computation involves linear algebra
  - Easy to impose shape restrictions (i.e. monotonicity) – becomes quadratic programming problem
  - Issues:
    - Selecting λ (smoothing parameter)

Concluding Thoughts

- Many “easy” semiparametric estimators exist, which provide robustness at little cost
  - No need to estimate infinite dimensional quantities of interest
  - Very easy to apply
  - Basic principle extends (i.e. conventional ideal point estimators remain consistent if errors terms are correlated across multiple votes on the same bill)
- Fully nonparametric estimators can be applied
  - The cost is slower convergence rates (and the curse of dimensionality)
  - Somewhat difficult to apply
  - Kernel estimators are not necessarily the best, but they are the easiest (and achieve optimal rates of convergence)

Concluding Thoughts

- Optimize tradeoff between robustness and efficiency via models with parametric and nonparametric components
    - Semiparametric modeling
        - If parameter of interest is finite dimensional, parametric rate can be achieved
        - Sandwich estimators can be applied for inference
        - Test statistics can often be estimated at the parametric rate
    - One dimensional infinite dimensional parameter of interest
        - Parametric rate is not achieved, but curse of dimensionality is avoided
        - Implementation of these flexible models is more difficult, but problems are not insurmountable