Semiparametric regression in Stata

Vincenzo Verardi

2013 UK Stata Users Group meeting
London, UK

September 2013

Semiparametric regression models

Semiparametric regression

- Deals with the introduction of some very general non-linear functional forms in regression analysis
- Generally used to fit a parametric model in which the functional form of a subset of the explanatory variables is not known and/or in which the distribution of the error term cannot be assumed to be of a specific type beforehand.
- Most popular semiparametric regression models are the **partially linear** models and **single index** models

Remark: for an excellent description of semiparametric estimators, we advice to read the book by Ahamada and Flachaire (2010) from which many of the explanations available here come from.

Semiparametric regression models

Partially linear models

- The partially linear model is defined as: $y = X\beta + m(z) + \varepsilon$
- Advantage 1: This model allows "any" form of the unknown function m
- Advantage 2: $\hat{\beta}$ is $\sqrt{n}$-consistent

Single index models

- The single index model is defined as: $y = g(X\beta) + \varepsilon$
- Advantage 1: generalizes the linear regression model (which assumes $g(\cdot)$ is linear)
- Advantage 2: the curse of dimensionality is avoided as there is only one nonparametric dimension

PLM example

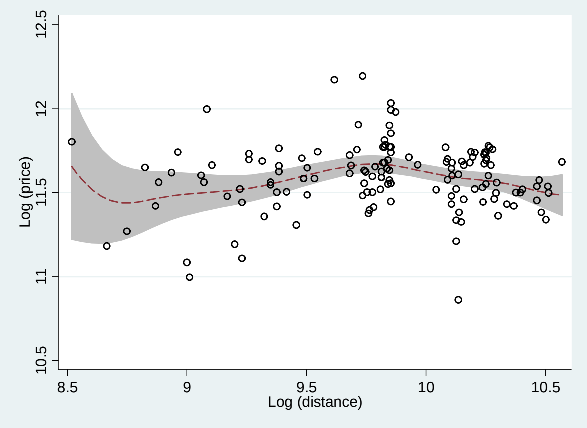
Hedonic pricing equation of houses

Wooldridge (2000): What was the effect of a local garbage incinerator on housing prices in North Andover in 1981?

```
. semipar lprice larea lland rooms bath age if y81==1, nonpar(ldist)

                                                      Number of obs =     142
                                                      R-squared     =  0.6863
                                                      Adj R-squared =  0.6748
                                                      Root MSE      =  0.1859
------------------------------------------------------------------------------
      lprice |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       larea |   .3266051    .070965     4.60   0.000     .1862768    .4669334
       lland |   .0790684   .0318007     2.49   0.014     .0161847    .1419521
       rooms |    .026588   .0266849     1.00   0.321    -.0261795    .0793554
       baths |   .1611464   .0400458     4.02   0.000     .0819585    .2403342
         age |  -.0029953   .0009564    -3.13   0.002    -.0048865   -.0011041
------------------------------------------------------------------------------
```

PLM Example

Hedonic pricing equation of houses
Non-parametric part



Single index example

Titanic accident

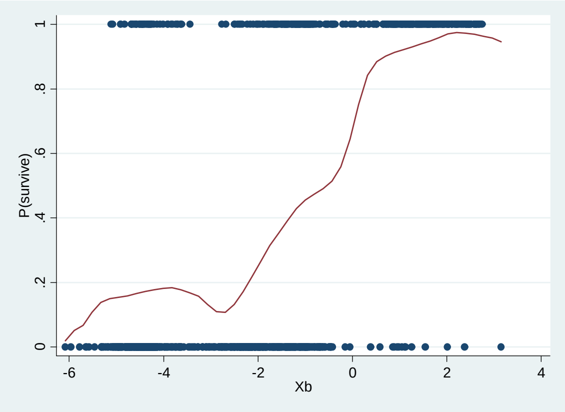
What was the probability of surviving the accident?

```
. xi: sml survived female age i.pclass
i.pclass         _Ipclass_1-3        (naturally coded; _Ipclass_1 omitted)

Iteration 0:   log likelihood = -485.15013

…

Iteration 6:   log likelihood = -471.17626

SML Estimator - Klein & Spady (1993)                Number of obs   =       1046
                                                    Wald chi2(4)    =      27.30
Log likelihood = -471.17626                         Prob > chi2     =     0.0000

------------------------------------------------------------------------------
    survived |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      female |   3.220109   .6381056     5.05   0.000     1.969445    4.470772
         age |  -.0334709   .0076904    -4.35   0.000    -.0485438   -.0183981
  _Ipclass_2 |  -1.360299    .370819    -3.67   0.000    -2.087091   -.6335076
  _Ipclass_3 |  -3.605414   .8002326    -4.51   0.000    -5.173842   -2.036987
------------------------------------------------------------------------------
```

Single index example

Titanic accident

Non-parametric part



Partially linear models

Quantitative dependent variable models

- Fractional polynomials
- Splines
- Additive models
- Yatchew's difference estimator
- Robinson's double residual estimator
- ...

Qualitative dependent variable models

- Fractional polynomials
- Splines
- Generalized additive models
- ...

Fractional polynomial

The **partially linear model** is defined as: $y = X\beta + m(z) + \varepsilon$

- In fractional polynomial models, $m(z) = \sum_{i=1}^{k} \gamma_i z^{p_i}$
- Powers p_i are taken from a predetermined set $S = \{-2, -1, -0.5, 0, 0.5, 1, 2, 3\}$ where z^0 is taken as $ln(z)$
- Generally $k = 2$ is sufficient to have a good fit
- For ℓ "repeated" powers p, we have $\sum_{i=1}^{\ell} \gamma_i z^p \left[\ln(z)\right]^{i-1}$
- All combinations of powers are fitted and the "best" fitting model (e.g. according to the AIC) is retained.
- As a fully parametric model, it is extremely easy to handle and can be generalized to non-linear regression models
- This model can be extended to qualitative dependent variable models without major problems

Spline regression

The **partially linear model** is defined as: $y = X\beta + m(z) + \varepsilon$

- In spline regression models
  $$m(z) = \sum_{j=1}^{p} \gamma_j z^j + \sum_{\ell=1}^{q} \gamma_{p\ell}(z - k_\ell)_+^p + \varepsilon$$
- Polynomial splines tend to be highly correlated. To deal with this, splines can be represented as B-spline bases which are, in essence, a rescaling of each of the piecewise functions.
- This model can easily be extended to qualitative dependent variable models
- Spline estimation is sensitive to the choice of the number of knots and their position. To reduce the impact of this choice, a penalization term can be introduced
- Penalized splines: estimate γ minimizing the following criterion $\sum_{i=1}^{n} \left[y_i - x_i^t\beta - m(z_i)\right]^2 + \lambda \int \left[m''(z)\right]^2 dz$

Additive models

The **partially linear model** is defined as: $y = X\beta + m(z) + \varepsilon$

- This is a special case of an additive separable model $y = \beta_0 + \sum_{d=1}^{D} m_d(z_d) + \varepsilon$ that can be estimated using backfitting
- The backfitting algorithm (that is equivalent to a penalized likelihood approach)
  - Initializes $\hat{\beta}_0 = \bar{y}$; $\hat{m}_d \equiv m_d^0$, $\forall d$ such that $\sum_d \hat{m}_d = 0$
  - Repeats till convergence:
    - For each predictor j :

$$\hat{m}_d \leftarrow smooth\left[\left(y - \hat{\beta}_0 - \sum_{k \neq d}^{D} \hat{m}_k\right) |z_d\right]$$

$$\hat{m}_d \leftarrow \hat{m}_d - \overline{\hat{m}}_d$$

- This algorithm can easily be extended to qualitative dependent variable models

Yatchew's (1998) difference estimator

The **partially linear model** is defined as: $y = X\beta + m(z) + \varepsilon$

- For the difference estimator, start by sorting the data according to z
- Estimate the model in difference $\Delta y = \Delta X\beta + \Delta m(z) + \Delta\varepsilon$
- If m is smooth, single-valued with bounded first derivative and if z has a compact support, $\Delta m(z)$ cancels out when the number of observation increases. Parameter vector β can be consistently estimated without modelling $m(z)$ explicitly
- Finally $m(z)$ can be estimated regressing $(y - X\hat{\beta})$ on z nonparametrically
- By selecting the order of differencing sufficiently large (and the optimal differencing weights), the estimator approaches asymptotic efficiency

Robinson's (1988) double residual estimator

The **partially linear model** is defined as: $y = X\beta + m(z) + \varepsilon$

- For the double residual estimator, take the expected value conditioning on z: $E(y|z) = E(X|z)\beta + m(z) + \underbrace{E(\varepsilon|z)}_{0}$
- We therefore have that $\underbrace{y - E(y|z)}_{\varepsilon_1} = \underbrace{(X - E(X|z))}_{\varepsilon_2}\beta + \varepsilon$
- By estimating $E(y|z)$ and $E(X|z)$ using some nonparmatetric regression method and replacing them in the above equation, it is possible to estimate β consistently without modelling $m(z)$ explicitly: $\hat{\beta} = \left(\hat{\varepsilon}_2'\hat{\varepsilon}_2\right)^{-1}\hat{\varepsilon}_2'\hat{\varepsilon}_1$
- Finally $m(z)$ can be estimated by regressing $\left(y - X\hat{\beta}\right)$ on z nonparametrically
- This estimator reaches the asymptotic efficiency bound $V = \frac{\sigma^2_\varepsilon}{n\sigma^2_{\varepsilon_2}}$

Some estimators available in Stata

Semi-non-parametric

Fractional polynomials: `fracpoly` and `mfp`
Splines: `mkspline`, `bspline`, `mvrs`
Penalized splines: `pspline`
Generalized additive models: `gam`

Semi-parametric

Yatechew's partially linear regression: `plreg`
Robinson's partially linear regression: `semipar`

In this talk we will concentrate on `semipar`. However, many of the presented results could be used for the other estimators !

Example

Let us generate a weird semiparametric model

- `set obs 1000`
- `drawnorm e`
- `generate z=(uniform()-0.5)*30`
- `generate x1=z+invnorm(uniform())`
- `generate x2=z+invnorm(uniform())`
- `generate x3=z+invnorm(uniform())`
- `generate y=x1+x2+x3+e`
- `replace y=(10*sin(abs(z)))*(z<_pi)+y`

To be useful, the partially linear model should estimate consistently both the parametric AND the non-parametric part. If one of the two is poorly estimated the other one will be as well. Let us compare the estimators in Stata

Fractional polynomial regression

```
. mfp regress y z x*, df(1, z:10)

      Source |       SS       df       MS              Number of obs =    1000
-------------+------------------------------           F(  8,   991) = 3869.80
       Model |  700273.966     8  87534.2457           Prob > F      =  0.0000
    Residual |  22416.2848   991  22.6198635           R-squared     =  0.9690
-------------+------------------------------           Adj R-squared =  0.9687
       Total |  722690.251   999  723.413664           Root MSE      =   4.756

------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       Iz__1 |  -784.7258   72.07625   -10.89   0.000    -926.1654   -643.2862
       Iz__2 |  -279.7523   29.01214    -9.64   0.000    -336.6846     -222.82
       Iz__3 |   789.2751   73.33961    10.76   0.000     645.3563    933.1938
       Iz__4 |  -505.0903   45.56326   -11.09   0.000    -594.5019   -415.6788
       Iz__5 |   107.7887   9.476345    11.37   0.000     89.19267    126.3847
      Ix1__1 |   1.157587   .1521613     7.61   0.000     .8589915    1.456182
      Ix2__1 |   .9479614   .1540668     6.15   0.000     .6456268    1.250296
      Ix3__1 |   .8754499   .1511476     5.79   0.000     .5788437    1.172056
       _cons |   4.988257   .2741892    18.19   0.000     4.450199    5.526315
------------------------------------------------------------------------------
```

Robinson's estimator

```
. semipar y x*, nonpar(z) generate(fit) partial(res)

                                                 Number of obs =    1000
                                                 R-squared     =  0.5745
                                                 Adj R-squared =  0.5733
                                                 Root MSE      =  1.4149
------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
          x1 |   .9962337   .0454645    21.91   0.000     .9070166    1.085451
          x2 |   .9165930   .0458836    19.98   0.000     .8265534    1.006633
          x3 |   .9914365   .0450222    22.02   0.000     .9030874    1.079786
------------------------------------------------------------------------------
```

Penalized spline

```
. pspline y z x*, nois

Computing standard errors:

Mixed-effects REML regression                   Number of obs      =      1000
Group variable: _all                            Number of groups   =         1

                                                Obs per group: min =      1000
                                                               avg =    1000.0
                                                               max =      1000


                                                Wald chi2(4)       =   2364.68
Log restricted-likelihood = -1650.8448          Prob > chi2        =    0.0000

------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
           z |   9.412364    .662555    14.21   0.000      8.11378    10.71095
          x1 |   .9449280   .0371332    25.45   0.000     .8721484    1.017708
          x2 |   .9317772   .0370458    25.15   0.000     .8591688    1.004386
          x3 |   1.027681   .0364005    28.23   0.000      .956337    1.099024
       _cons |   146.2157   10.03695    14.57   0.000     126.5437    165.8878
------------------------------------------------------------------------------
```

Spline

```
. mvrs regress y z x*, df(1, z:10)

      Source |       SS       df       MS              Number of obs =    1000
-------------+------------------------------           F( 12,   987) = 3264.65
       Model |  704930.142    12  58744.1785           Prob > F      =  0.0000
    Residual |  17760.1088   987  17.9940312           R-squared     =  0.9754
-------------+------------------------------           Adj R-squared =  0.9751
       Total |  722690.251   999  723.413664           Root MSE      =  4.2419

------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         z_0 |  -.4446125   2.147522    -0.21   0.836    -4.658846    3.769621
         z_1 |   .4644359   .1346419     3.45   0.001     .2002187    .7286531
         z_2 |  -1.314159   .1341581    -9.80   0.000    -1.577427   -1.050891
         z_3 |  -1.651472   .1343905   -12.29   0.000    -1.915196   -1.387749
         z_4 |   .5623887   .1342405     4.19   0.000     .2989591    .8258183
         z_5 |  -.6660937   .1342368    -4.96   0.000     -.929516   -.4026713
         z_6 |   .9780222   .1344052     7.28   0.000     .7142694    1.241775
         z_7 |   1.726668   .1342929    12.86   0.000     1.463136    1.990201
         z_8 |  -.3566616   .1343096    -2.66   0.008    -.6202267   -.0930965
          x1 |   1.256455   .1360496     9.24   0.000     .9894751    1.523434
          x2 |   .8465489   .1373653     6.16   0.000     .5769873    1.116111
          x3 |   .8592235   .1350626     6.36   0.000     .5941807    1.124266
       _cons |   1.664401   .1506893    11.05   0.000     1.368693     1.96011
------------------------------------------------------------------------------
```

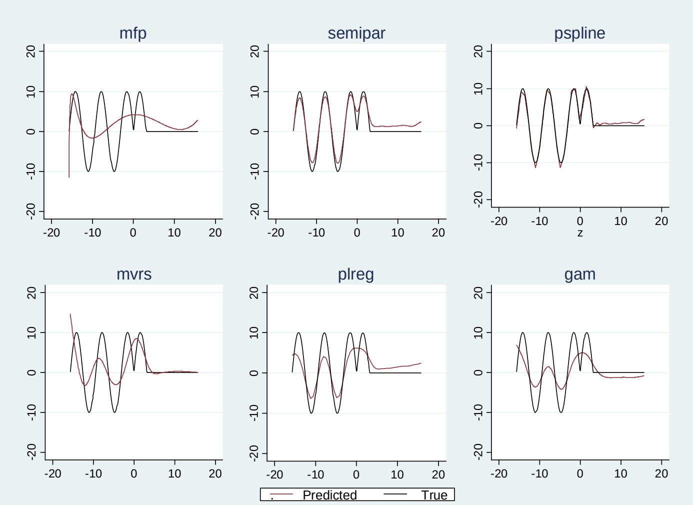
Yatchew's estimator

```
. plreg y x*, nlf(z) gen(fit)

Partial Linear regression model with Yatchew's weighting matrix

      Source |       SS       df       MS              Number of obs =     999
-------------+------------------------------           F(3, 996)     =  858.29
       Model | 2756.201275     3  918.733758           Prob > f      =  0.0000
    Residual | 1066.136271   996  1.07041794           R-squared     =  0.7211
-------------+------------------------------           Adj R-squared =  0.7202
       Total |    3822.338   999  3.82616371           Root MSE      =  1.0346

------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
          x1 |   .9114451   .0411167    22.17   0.000     .8307597    .9921304
          x2 |   .9425252   .0399196    23.61   0.000     .8641891    1.020861
          x3 |   1.024437   .0394415    25.97   0.000     .9470392    1.101835
------------------------------------------------------------------------------
Significance test on z: V =774.383 P>|V| = 0.000
------------------------------------------------------------------------------
```

Generalized additive model

```
. gam y x* z, df(1, z:10)

1000 records merged.

Generalized Additive Model with family gauss, link ident.

Model df     =    13.999                         No. of obs =      1000
Deviance     =   12934.9                         Dispersion =   13.1186
-------------------------------------------------------------------------
          y |   df     Lin. Coef.  Std. Err.       z         Gain   P>Gain
------------+------------------------------------------------------------
         x1 |      1    1.159738   .1156261    10.030           .        .
         x2 |      1    .8660830   .1169652     7.405           .        .
         x3 |      1    .9199053   .1148958     8.006           .        .
          z |  9.999  -.0322593    .2027343    -0.159    1088.570   0.0000
      _cons |      1    2.52637     .114536    22.057           .        .
-------------------------------------------------------------------------
```

Predicted non-linear function



Example from plreg (Yatchew, 2003)

Assess scale economies in electricity distribution.

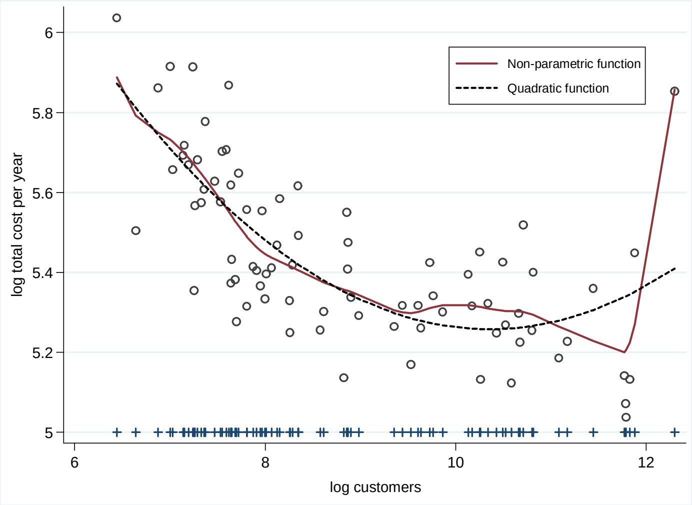
- Data for that example come from the survey of 81 municipal electricity distributors in Ontario, Canada, in 1932.
- The cost of distributing electricity is modeled in a simple Cobb–Douglas framework, where
  - **tc** is the log of total cost per customer
  - **cust** is the log of number of customers
- Control variables are the log of wage rate (**wage**), of price of capital (**pcap**), of kilowatt hours per customer (**kwh**) and of kilometers of distribution wire per customer (**kmwire**), a dummy variable for the public utility commissions that deliver additional services (**puc**), the remaining life of distribution assets (**life**) and the load factor (**lf**).

Running semipar using the example from plreg

```
. semipar tc wage pcap puc kwh life lf kmwire, nonpar(cust) gen(func)

                                                    Number of obs =       81
                                                    R-squared     =   0.5839
                                                    Adj R-squared =   0.5445
                                                    Root MSE      =   0.1323
```

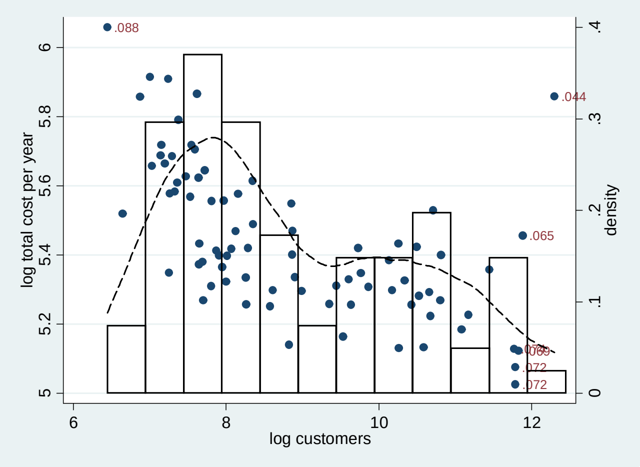
tc	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
wage	.7098844	.2837827	2.50	0.015	.1444351	1.275334
pcap	.5182129	.0662044	7.83	0.000	.3862978	.6501281
puc	-.0661499	.0340252	-1.94	0.056	-.1339466	.0016468
kwh	.022051	.0781643	0.28	0.779	-.1336947	.1777967
life	-.5178481	.1060321	-4.88	0.000	-.7291218	-.3065744
lf	1.310515	.3926239	3.34	0.001	.5281949	2.092835
kmwire	.3681217	.07709	4.78	0.000	.2145166	.5217269

Running semipar using the example from plreg

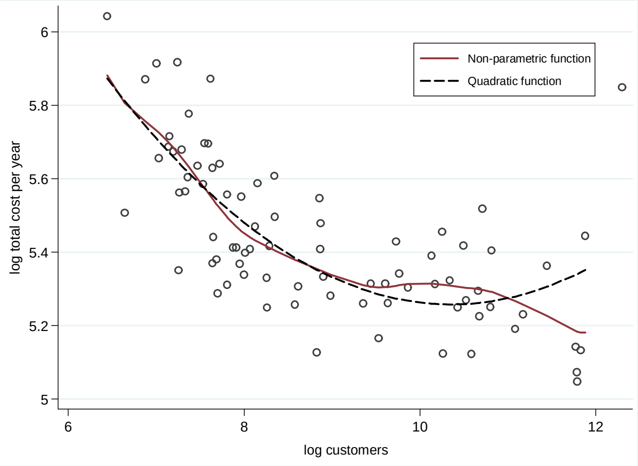


Semiparametric estimators are noisy in sparse regions

Trimming



Same example with trimming at f(z)=0.05



Testing for a specific parametric form

Test

- Hardle and Mammen (1993) propose a testing procedure based on square deviations between the nonparametric kernel estimator $\hat{m}(z_i)$ (with bandwidth h) and a parametric regression $\hat{f}(z_i, \theta)$
- The test statistic they propose is $T_n = n\sqrt{h}\sum_{i=1}^{n}\left(\hat{m}(z_i) - \hat{f}(z_i, \theta)\right)^2 \pi(z_i)$ where $\pi(\cdot)$ is an optional weight function
- To obtain critical values, Hardle and Mammen (1993) suggest using wild bootstrap
- An absence of rejection of the null means that the polynomial adjustment is suitable

Testing for a parametric fit in the above example

```
. semipar tc wage pcap puc kwh life lf kmwire, nonpar(cust) test(2)

…

Simulation the distribution of the test statistic

bootstrap replicates (100)
----+--- 1 ---+--- 2 ---+--- 3 ---+--- 4 ---+--- 5

..................................................    50
..................................................   100


H0: Parametric and non-parametric fits are not different
-------------------------------------------------------
Standardized Test statistic T: 1.2186131
Critical value (95%): 1.9599639
Approximate P-value: .24
```

Testing for a parametric fit in the above example

```
. semipar tc wage pcap puc kwh life lf kmwire, nonpar(cust) test(1)

…

Simulation the distribution of the test statistic

bootstrap replicates (100)
----+--- 1 ---+--- 2 ---+--- 3 ---+--- 4 ---+--- 5

..................................................    50
..................................................   100


H0: Parametric and non-parametric fits are not different
-------------------------------------------------------
Standardized Test statistic T: 1.193676
Critical value (95%): 1.959964
Approximate P-value: .25
```

Testing for a parametric fit in the above example

```
. semipar tc wage pcap puc kwh life lf kmwire, nonpar(cust) test(0)

…

Simulation the distribution of the test statistic

bootstrap replicates (100)
----+--- 1 ---+--- 2 ---+--- 3 ---+--- 4 ---+--- 5

..................................................    50
..................................................   100


H0: Parametric and non-parametric fits are not different
-------------------------------------------------------
Standardized Test statistic T: 3.5257177
Critical value (95%): 1.959964
Approximate P-value: 0
```

Robinson's double residual estimator

The **partially linear model** is defined as: $y = X\beta + m(z) + \varepsilon$

- For the double residual estimator, take the expected value conditioning on z: $E(y|z) = E(X|z)\beta + m(z) + \underbrace{E(\varepsilon|z)}_{0}$
- We therefore have that $\underbrace{y - E(y|z)}_{\varepsilon_1} = \underbrace{(X - E(X|z))}_{\varepsilon_2}\beta + \varepsilon$
- By estimating $E(y|z)$ and $E(X|z)$ using some nonparmatetric regression method and replacing them in the above equation, it is possible to estimate consistently β without modelling explicitly $m(z)$: $\hat{\beta} = \left(\hat{\varepsilon}_2'\hat{\varepsilon}_2\right)^{-1}\hat{\varepsilon}_2'\hat{\varepsilon}_1$

Dealing with heteroskedasticity and clustering

Robust-to-heteroskedasticity covariance matrix

- To deal with heteroskedasticity the parametric part of the model could be estimated using FGLS
- Since the estimations are not biased in case of heteroskedasticity, a simple alternative is to correct the variance of the betas using Hubert-White sandwich covariance matrix
- In case of general heteroskedasticity:
  $V(\hat{\beta}) = \left(\hat{\varepsilon}_2'\hat{\varepsilon}_2\right)^{-1}\hat{\varepsilon}_2'\hat{\varepsilon}\hat{\varepsilon}'\hat{\varepsilon}_2\left(\hat{\varepsilon}_2'\hat{\varepsilon}_2\right)^{-1}$
- For clustered data: $V(\hat{\beta}) = \left(\hat{\varepsilon}_2'\hat{\varepsilon}_2\right)^{-1}\sum_{j=1}^{n_c} u_j u_j' \left(\hat{\varepsilon}_2'\hat{\varepsilon}_2\right)^{-1}$ with $u_j = \sum_i \hat{\varepsilon}_i x_i$ where $\hat{\varepsilon}_i$ is the residual for the ith observation and x_i is a row vector of predictors including the constant and n_c is the number of clusters.

Previous example BUT controlling for heteroskedasticity

```
. semipar tc wage pcap puc kwh life lf kmwire, nonpar(cust) robust

                                          Number of obs =       81
                                          R-squared     =   0.5839
                                          Adj R-squared =   0.5445
                                          Root MSE      =   0.1323
```

tc	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
wage	.7098844	.3174935	2.24	0.028	.0772648	1.342504
pcap	.5182129	.0656184	7.90	0.000	.3874655	.6489604
puc	-.0661499	.0327918	-2.02	0.047	-.131489	-.0008108
kwh	.022051	.0919709	0.24	0.811	-.1612051	.2053071
life	-.5178481	.1022124	-5.07	0.000	-.7215108	-.3141854
lf	1.310515	.3713168	3.53	0.001	.5706503	2.05038
kmwire	.3681217	.0700106	5.26	0.000	.2286225	.507621

Generating clustered data in the example

Expanding the dataset without bringing new information

- Generate an identifier for each individual (`gen id=_n`)
- Expand the dataset 3 times (`expand 3`)
- In this case we have perfect within cluster correlation
- If we use a standard (or even a robust-to-heteroskedasticity) covariance matrix, the inflation of n would shrink the standard errors and inflate the t-statistics.
- We must use the clustered variance.

Extended dataset without cluster correction

```
. semipar tc wage pcap puc kwh life lf kmwire, nonpar(cust)

                                                    Number of obs =      243
                                                    R-squared     =   0.5866
                                                    Adj R-squared =   0.5744
                                                    Root MSE      =   0.1256
```

tc	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
wage	.6889357	.1572608	4.38	0.000	.3791215	.9987499
pcap	.5102208	.036622	13.93	0.000	.438073	.5823687
puc	-.0692528	.0190201	-3.64	0.000	-.1067236	-.031782
kwh	.0222381	.0433129	0.51	0.608	-.0630912	.1075675
life	-.5154635	.0588754	-8.76	0.000	-.631452	-.399475
lf	1.327293	.2186896	6.07	0.000	.8964597	1.758126
kmwire	.3594247	.0430594	8.35	0.000	.2745947	.4442546

Extended dataset with the cluster correction

```
. semipar tc wage pcap puc kwh life lf kmwire, nonpar(cust) cluster(id)

                                                  Number of obs =      243
                                                  R-squared     =   0.5866
                                                  Adj R-squared =   0.5744
                                                  Root MSE      =   0.1256
```

tc	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
wage	.6889357	.3076639	2.24	0.028	.076665	1.301206
pcap	.5102208	.0632224	8.07	0.000	.3844043	.6360374
puc	-.0692528	.031856	-2.17	0.033	-.1326482	-.0058573
kwh	.0222381	.088764	0.25	0.803	-.1544078	.1988841
life	-.5154635	.100355	-5.14	0.000	-.7151762	-.3157508
lf	1.327293	.3579018	3.71	0.000	.6150456	2.03954
kmwire	.3594247	.0688593	5.22	0.000	.2223902	.4964591

Dealing with endogeneity

Standard IV

- We have a model $y = X\beta + \varepsilon$ where $E(\varepsilon|X) \neq 0$
- We need to find relevant and exogenous instruments W and estimate $\hat{\beta}_{IV} = \left(X'P_W X\right)^{-1} X'P_W y$
- $P_W = W(W'W)^{-1}W'$ is the part of X explained by W i.e. OLS fitted values from $X = W + \nu$
- Equivalently, $\hat{\beta}_{IV} = \left(X'X\right)^{-1} X'y$ of the model $y = X\beta + \gamma\hat{\nu} + \varepsilon$ (CFA)

Dealing with endogeneity in the parametric part

Semiparametric IV

- We have a model $y = X\beta + m(z) + \varepsilon$ where $E(\varepsilon|X) \neq 0$ but $E(\varepsilon|z) = 0$

- The double residual estimator is an OLS estimation of

$$\underbrace{y - \widehat{E(y|z)}}_{\tilde{y}} = \underbrace{\left(X - \widehat{E(X|z)}\right)}_{\tilde{X}}\beta + \varepsilon$$

- We therefore have that

$$\hat{\beta} = \left(\tilde{X}' P_W \tilde{X}\right)^{-1} \tilde{X}' P_W \tilde{y}$$

$$V\left(\hat{\beta}\right) = \sigma^2_\varepsilon \left(\tilde{X}' P_W \tilde{X}\right)^{-1}$$

Dealing with endogeneity in the non-parametric part

Semiparametric IV

- We have a model $y = X\beta + m(z) + \varepsilon$ where $E(\varepsilon|X) = 0$ but $E(\varepsilon|z) \neq 0$
- For the double residual estimator, take the expected value conditioning on z: $E(y|z) = E(X|z)\beta + m(z) + \underbrace{E(\varepsilon|z)}_{\neq 0}$
- However in this case $E(y|z)$ and $E(X|z)$ cannot be consistently estimated using a nonparametric regression since z is endogenous
- An appealing solution would be to condition on W: $E(y|W) = E(X|W)\beta + E(m(z)|W) + E(\varepsilon|W)$ but in this case the non-parametric part does not cancel out
- It is a complicated problem!

Dealing with endogeneity in the non-parametric part

Semiparametric IV

- Assume that W is correlated to z, not to ε, such that $z = W\pi + \nu$ and $E(\nu|W) = 0$
- If $E(\varepsilon|z,\nu) = \rho\nu$, then $\varepsilon = \rho\nu + \eta$ and the partially linear model becomes $y = X\beta + m(z) + \rho\nu + \eta$
- Applying the double residual principle, we have $y - E(y|z) = (X - E(X|z))\beta + \rho(\nu - E(\nu|z)) + \eta$
- ν should be estimated using the residuals fitted from $z = W\pi + \nu$ (i.e. the first stage of IV)

Dealing with endogeneity in the non-parametric part

Stata example

Let's reproduce the return-to-education example of Wooldridge as presented in ivreg2.sthlp

- `bcuse/mroz.dta`

First Stage

- `reg educ age kidslt6 kidsge6 exper expersq`
- `predict res, res`

Second stage

- `bootstrap: semipar lwage exper expersq res, nonpar(educ) nograph`

Dealing with endogeneity in the non-parametric part

Stata example - first stage

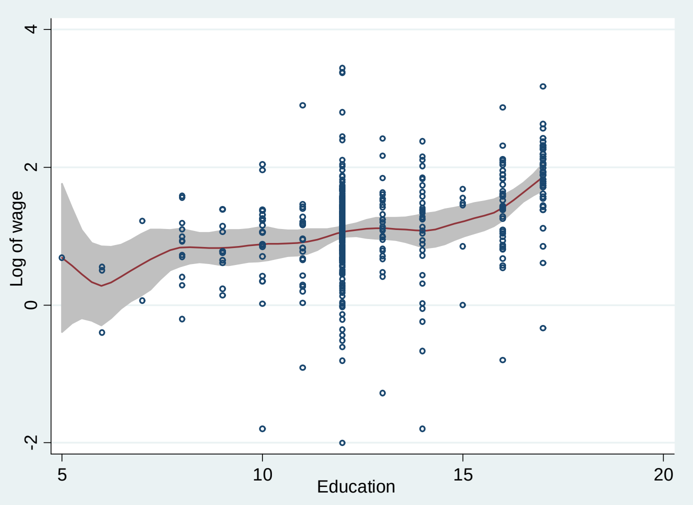
```
. reg educ age kidslt6 kidsge6 exper expersq

      Source |       SS       df       MS              Number of obs =     753
-------------+------------------------------           F(  5,   747) =    7.31
       Model |  182.382773     5  36.4765545           Prob > F      =  0.0000
    Residual |  3727.65707   747   4.9901701           R-squared     =  0.0466
-------------+------------------------------           Adj R-squared =  0.0403
       Total |  3910.03984   752  5.19952106           Root MSE      =  2.2339

------------------------------------------------------------------------------
        educ |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         age |  -.0397596   .0126845    -3.13   0.002     -.064661   -.0148582
     kidslt6 |   .3237357   .1746409     1.85   0.064    -.0191097    .6665812
     kidsge6 |  -.1630517   .0686703    -2.37   0.018    -.2978615   -.0282419
       exper |   .0942983   .0293858     3.21   0.001     .0366097     .151987
     expersq |  -.0022822   .0009627    -2.37   0.018    -.0041721   -.0003923
       _cons |   13.52568    .631423    21.42   0.000      12.2861    14.76525
------------------------------------------------------------------------------
```

Dealing with endogeneity on the non-parametric part

Stata example - second stage

```
. bootstrap: semipar lwage exper expersq res, nonpar(educ) nograph
(running semipar on estimation sample)

Bootstrap replications (50)
----+--- 1 ---+--- 2 ---+--- 3 ---+--- 4 ---+--- 5
..................................................    50
------------------------------------------------------------------------------
             |   Observed   Bootstrap                         Normal-based
       lwage |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       exper |    .041717   .0168872     2.47   0.013     .0086187    .0748152
     expersq |  -.0007824   .0004895    -1.60   0.110    -.0017418     .000177
         res |  -.0020053   .0983329    -0.02   0.984    -.1947342    .1907236
------------------------------------------------------------------------------
```

Dealing with endogeneity in the non-parametric part



Dealing with unobserved heterogeneity

Panel data

- Consider a general panel data semiparametric model
  $y_{i,\tau} = x_{i,\tau}^t \beta + m(z_{i,\tau}) + \alpha_i + \varepsilon_{i,\tau}, \quad i = 1, ..., n; \tau = 1, ..., T$
- A first difference estimator would be
  $\Delta y_{i,\tau} = \Delta x_{i,\tau}^t \beta + [m(z_{i,\tau}) - m(z_{i,\tau-1})] + \Delta\varepsilon_{i,\tau}$
- Baltagi and Li (2002) show that $[m(z_{i,\tau}) - m(z_{i,\tau-1})]$ can be estimated by series estimator $[p^k(z_{i,\tau}) - p^k(z_{i,\tau-1})]\gamma$ and suggest fitting
  $\Delta y_{i,\tau} = \Delta x_{i,\tau}^t \beta + [p^k(z_{i,\tau}) - p^k(z_{i,\tau-1})]\gamma + \Delta\varepsilon_{i,\tau}$
- Having estimated $\hat{\beta}$, it is easy to fit the fixed effects $\hat{\alpha}_i$ and estimate the error component residual
  $\hat{u}_{i,\tau} = y_{i,\tau} - x_{i,\tau}^t \hat{\beta} - \hat{\alpha}_i = m(z_{i,\tau}) + \varepsilon_{i,\tau}$.
- The curve m can be fitted by regressing $\hat{u}_{i,\tau}$ on $z_{i,\tau}$ using some standard non-parametric regression estimator.

Dealing with unobserved heterogeneity

Simple example

- `set obs 1000`
- `drawnorm x1-x3 e`
- `gen d=round(uniform()*250)`
- `replace x3=x3+d/100` $\Longrightarrow$ corr(d,x3)=0.55
- `gen y=x1+x2+x3+x3^2+d+e`
- `bysort d: gen t=_n`
- `tsset d t`
- `xtsemipar y x1 x2, nonpar(x3)`

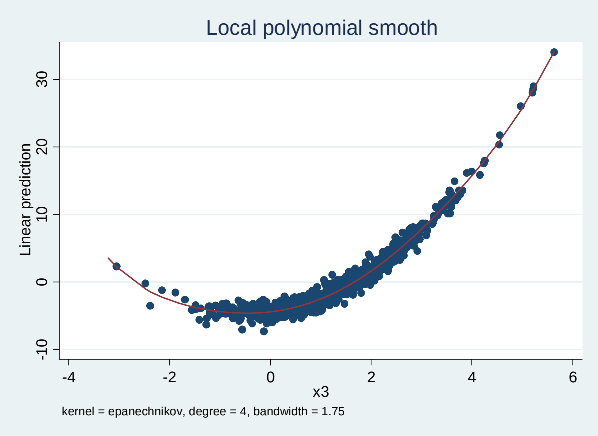
Dealing with unobserved heterogeneity

Simple example

```
. xtsemipar y x1 x2, nonpar(x3)

                                                   Number of obs        =     754
                                                   Within R-squared     =  0.9515
                                                   Adj Within R-squared =  0.9511
                                                   Root MSE             =  1.4122
------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
          x1 |   .9098837   .0370427    24.56   0.000     .8371636    .9826037
          x2 |   1.011729   .0356399    28.39   0.000     .9417624    1.081695
------------------------------------------------------------------------------
```

Dealing with unobserved heterogeneity

Simple example



Marginal effect

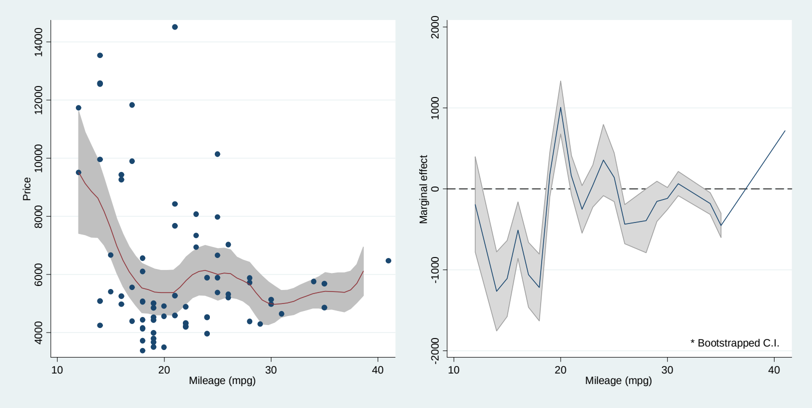
Presenting the results

- To get an idea of the marginal effects, we could look at the first derivative of the estimated function on each point.
- Stata function `dydx` is very useful here (beware of repeated values)
- **Example**

```
semipar price weight, nonpar(mpg) gen(party) ci
bysort mpg:  gen ok=(_n==1)
dydx party mpg if ok==1, gen(fprim)
bysort mpg:  replace fprim=fprim[1]

twoway (line fprim mpg)
```

Marginal effect

Plotting the marginal effects



Single index models

Single index models

- Defined as: $y = g(X\beta) + \varepsilon$
- In terms of rates of convergence it is as accurate as a parametric model for the estimation of β and as accurate as a one-dimensional nonparametric model for the estimation of $g(\cdot)$
- The specification is more flexible than a parametric model and avoids the curse of dimensionality
- $g(\cdot)$ is analogous to a link function in a generalized linear model, except that it is unknown and must be estimated.
- The conditional mean function is $E(y|X) = g(X\beta)$
    - Ichimura (1993) SLS
    - Klein and Spady (1993) Binary choice estimator

Single index models

Ichimura (1993) semiparametric least squares

- The single index regression model is: $y = g(X\beta) + \varepsilon$
- If $g(\cdot)$ were known, β could be estimated by NLS: $\hat{\beta} = \arg\min_{\beta} \sum_{i=1}^{n} (y_i - g(x_i^t\beta))^2$
- Since $g(\cdot)$ is unknown, Ichimura suggests replacing $g(\cdot)$ with a leave-one-out Nadaraya-Watson estimator of $g(\cdot)$
- The coefficient of one continuous variable is set to 1. Such a normalization is required because rescaling of the vector β by a constant and a similar rescaling of the function g by the inverse of the constant will produce the same regression function.
- `sls.ado` available from Michael Barker upon request

Single index models

Klein and Spady (1993) semiparametric binary choice estimator

- The single index regression model is: $y = 1(g(X\beta) + \varepsilon > 0)$
- If $g(\cdot)$ were known, you could estimate β by ML:
  $\hat{\beta} = \arg\max_{\beta} \sum_{i=1}^{n} (y_i \ln(g(x_i^t \beta)) + (1 - y_i)\ln(1 - g(x_i^t \beta)))$
- Since $g(\cdot)$ is unknown,Klein and Spady suggest replacing $g(\cdot)$ with a leave-one-out Nadaraya-Watson estimator of $g(\cdot)$
- In the context of binary choice, Klein and Spady estimator is preferable to Ichimura's as it can be shown to be more efficient.
- `sml.ado`

Single index models

Simple example - Klein and Spady

- `set obs 1000`
- `drawnorm x1 x2 x3 e`
- `gen y=x1+x2+x3+e>0`
- `sml y x*`
- `matrix B=e(b)`
- `matrix V=e(V)`
- `predict Xb`
- `lpoly y Xb, gen(F) at(Xb) gaussian`

Note: Here $g(\cdot) = F(\cdot)$

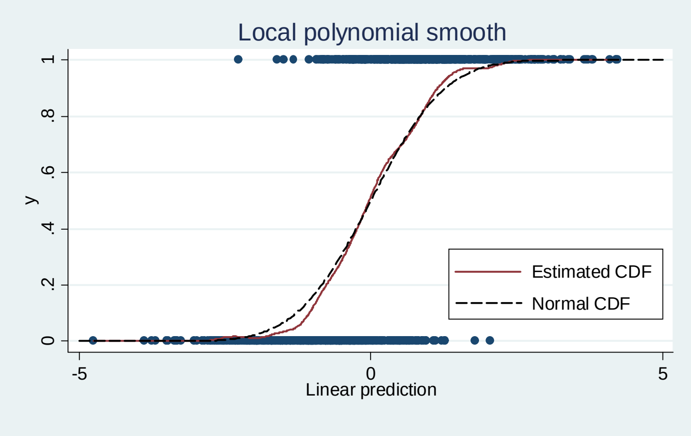
Single index models

Simple example - Klein and Spady

```
. sml y x*


Iteration 0:   log likelihood =  -355.3734
..
Iteration 3:   log likelihood = -355.33626

SML Estimator - Klein & Spady (1993)               Number of obs   =       1000
                                                   Wald chi2(3)    =      23.00
Log likelihood = -355.33626                        Prob > chi2     =     0.0000

------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
          x1 |   1.018119   .2221681     4.58   0.000     .5826774    1.453561
          x2 |   1.034679   .2210081     4.68   0.000     .6015106    1.467847
          x3 |   .9120784   .1939964     4.70   0.000     .5318524    1.292304
------------------------------------------------------------------------------

. matrix B=e(b)
. matrix V=e(V)
. predict Xb
. lpoly y Xb, gen(F) at(Xb) gaussian
```

Single index models

Simple example - Klein and Spady



Single Index Model - Klein and Spady

Marginal effects

$$mfx : \frac{\partial p(y=1)}{\partial X_j} = \frac{\partial F(X\beta)}{\partial X_j} = \frac{\partial F(X\beta)}{\partial (X\beta)} \frac{\partial (X\beta)}{\partial X_j} = f(X\beta)\beta_j$$

In Stata

```
dydx F Xb, gen(f)
local j 0
foreach var of varlist x1 x2 x3 {
local j=‘j’+1
gen margin‘var’=f*B[1,‘j’]
}
matrix M=J(3,3,0)
```

Single index models

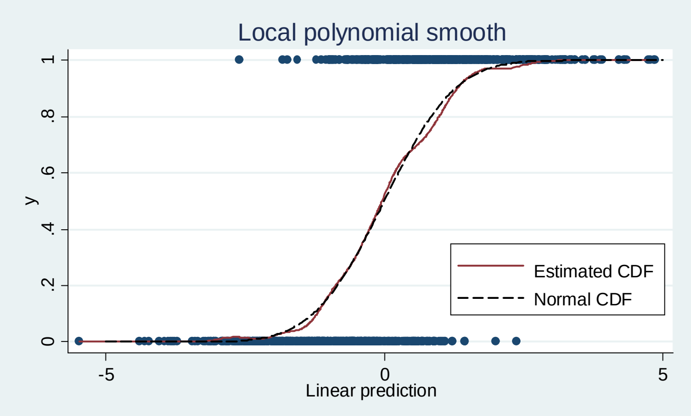
Simple example - Ichimura

```
. sls y x3 x1 x2
initial:       SSq(b) =  102.93577
alternative:   SSq(b) =  102.08216
rescale:       SSq(b) =  102.07341
rescale eq:    SSq(b) =  95.025004
SLS 0:   SSq(b) =  95.025004
...
SLS 11:  SSq(b) =   54.77692
------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
Index        |
          x1 |   1.063685   1.119289     0.95   0.342     -1.13008     3.25745
          x2 |   1.010571   1.139634     0.89   0.375    -1.223071    3.244214
          x3 |          1  (offset)
------------------------------------------------------------------------------

                   b
marginx1  .19322335
marginx2  .18357503
```

"`margins, dydx(*) predict(ey) force`" should work as well but beware of S.E.

Single index models

Simple example - Ichimura



Conclusion

Conclusion

- Stata has several semi-parametric and semi-non-parametric estimators readily available
- The practical implementation is easy and fast
- These estimators are much more flexible than pure parametric models and at the same time do not suffer from the curse of dimensionality
- Most of the violations of the Gauss-Markov assumption can be easily tackled
- Some work is still needed to make the marginal effects available after the estimations

References

References

- Ahamada, I. and Flachaire, E. (2010), **Non-Parametric Econometrics**. Oxford University Press
- Hastie, T.J. and Tibshirani, R.J. (1990), **Generalized Additive Models**. Chapman and Hall.
- Libois, F. and Verardi, V. (2013), **Semiparametric Fixed-effects Estimator**, *Stata Journal* 13(2): 329-336.
- Robinson, P. (1988), **Root-n-consistent semi-parametric regression**, *Econometrica* 56: 931-54.
- Royston, P. and Sauerbrei, W. (2008), **Multivariable Model - Building: A Pragmatic Approach to Regression Anaylsis based on Fractional Polynomials for Modelling Continuous Variables**. Wiley.

References

References

- Ruppert, D., Wand, M. P. and Carroll, R.J. (2003), **Semiparametric Regression**. Cambridge University Press.
- Verardi, V. and Debarsy, N. (2012), **Robinson's square root of N consistent semiparametric regression estimator in Stata**, *Stata Journal* 12(4): 726-735.
- Yatchew, A. (1998), **Nonparametric regression techniques in economics**, *Journal of Economic Literature* 36: 669-721.
- Yatchew, A. (2003), **Semiparametric Regression for the Applied Econometrician**. Cambridge University
- Wooldridge, J. (2000), **Introductory Econometrics: A Modern Approach**. South-Western

Stata Commands

Available in Standard 13

- `fracpoly`, `mfp` (Fractional polynomials)
- `mvrs`, `mkspline` (Splines)

Available from SSC

- `bspline` (Basis splines) - Roger Newson
- `pspline` (Penalized Splines) - Ben Jann and Roberto Gutierrez
- `xtsemipar` (Baltagi and Li's F.E. estimator)
- `semipar` (Robinson's double residual estimator)
- `plreg` (Yatchew's difference estimator) - Michael Lokshin
- `gam` (Generalized additive models) - Patrick Royston and Gareth Ambler

Available form Stata Journal

- `sml` (Klein and spady's estimator) - Giuseppe De Luca

Available from the author upon request

- `sls` (Ichimura's estimator) - Michael Barker