Estimating and Interpreting Effects for Nonlinear and Nonparametric Models

Enrique Pinzón

September 18, 2018

Objective

- Build a unified framework to ask questions about model estimates
- Learn to apply this unified framework using Stata

A Brief Introduction to Stata and How I Work

- A look at the Stata Interface
- From dialog boxes to do-files
- Loading your data
  - Excel
  - Delimited (comma, tab, or other)
  - ODBC (open data base connectivity)
  - Fred, SAS, Haver
- "Big data"
  - 120,000 variables 20 billion observations (MP)
  - 32,767 variables 2.14 billion observations (SE)
- Stata resources
  https://www.stata.com/links/resources-for-learning-stata/

A Brief Introduction to Stata and How I Work

- A look at the Stata Interface
- From dialog boxes to do-files
- Loading your data
  - Excel
  - Delimited (comma, tab, or other)
  - ODBC (open data base connectivity)
  - Fred, SAS, Haver
- "Big data"
  - 120,000 variables 20 billion observations (MP)
  - 32,767 variables 2.14 billion observations (SE)
- Stata resources
  https://www.stata.com/links/resources-for-learning-stata/

Factor variables

- Distinguish between discrete and continuous variables
- Way to create “dummy-variables”, interactions, and powers
- Works with most Stata commands

Using factor variables

```
. import excel apsa, firstrow

. tabulate d1
         d1 |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |      2,000       20.00       20.00
          1 |      2,000       20.00       40.00
          2 |      2,044       20.44       60.44
          3 |      2,037       20.37       80.81
          4 |      1,919       19.19      100.00
------------+-----------------------------------
      Total |     10,000      100.00

. summarize 1.d1
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
        1.d1 |     10,000          .2      .40002          0          1

. summarize i.d1
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
          d1 |
          1  |     10,000          .2      .40002          0          1
          2  |     10,000       .2044    .4032827          0          1
          3  |     10,000       .2037    .4027686          0          1
          4  |     10,000       .1919    .3938145          0          1
```

Using factor variables

```
. import excel apsa, firstrow

. tabulate d1
         d1 |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |      2,000       20.00       20.00
          1 |      2,000       20.00       40.00
          2 |      2,044       20.44       60.44
          3 |      2,037       20.37       80.81
          4 |      1,919       19.19      100.00
------------+-----------------------------------
      Total |     10,000      100.00

. summarize 1.d1
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
        1.d1 |     10,000          .2      .40002          0          1

. summarize i.d1
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
          d1 |
          1  |     10,000          .2      .40002          0          1
          2  |     10,000       .2044    .4032827          0          1
          3  |     10,000       .2037    .4027686          0          1
          4  |     10,000       .1919    .3938145          0          1
```

Using factor variables

```
. import excel apsa, firstrow

. tabulate d1
         d1 |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |      2,000       20.00       20.00
          1 |      2,000       20.00       40.00
          2 |      2,044       20.44       60.44
          3 |      2,037       20.37       80.81
          4 |      1,919       19.19      100.00
------------+-----------------------------------
      Total |     10,000      100.00

. summarize 1.d1
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
        1.d1 |     10,000          .2      .40002          0          1

. summarize i.d1
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
          d1 |
          1  |     10,000          .2      .40002          0          1
          2  |     10,000       .2044    .4032827          0          1
          3  |     10,000       .2037    .4027686          0          1
          4  |     10,000       .1919    .3938145          0          1
```

Using factor variables

```
. import excel apsa, firstrow

. tabulate d1
         d1 |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |      2,000       20.00       20.00
          1 |      2,000       20.00       40.00
          2 |      2,044       20.44       60.44
          3 |      2,037       20.37       80.81
          4 |      1,919       19.19      100.00
------------+-----------------------------------
      Total |     10,000      100.00

. summarize 1.d1
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
        1.d1 |     10,000          .2      .40002          0          1

. summarize i.d1
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
          d1 |
          1  |     10,000          .2      .40002          0          1
          2  |     10,000       .2044    .4032827          0          1
          3  |     10,000       .2037    .4027686          0          1
          4  |     10,000       .1919    .3938145          0          1
```

Using factor variables

```
. summarize ibn.d1

    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
          d1 |
          0  |     10,000          .2      .40002          0          1
          1  |     10,000          .2      .40002          0          1
          2  |     10,000       .2044    .4032827          0          1
          3  |     10,000       .2037    .4027686          0          1
          4  |     10,000       .1919    .3938145          0          1

. summarize ib2.d1

    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
          d1 |
          0  |     10,000          .2      .40002          0          1
          1  |     10,000          .2      .40002          0          1
          3  |     10,000       .2037    .4027686          0          1
          4  |     10,000       .1919    .3938145          0          1
```

Using factor variables

```
. summarize ibn.d1
```

Variable	Obs	Mean	Std. Dev.	Min	Max
d1					
0	10,000	.2	.40002	0	1
1	10,000	.2	.40002	0	1
2	10,000	.2044	.4032827	0	1
3	10,000	.2037	.4027686	0	1
4	10,000	.1919	.3938145	0	1

```
. summarize ib2.d1
```

Variable	Obs	Mean	Std. Dev.	Min	Max
d1					
0	10,000	.2	.40002	0	1
1	10,000	.2	.40002	0	1
3	10,000	.2037	.4027686	0	1
4	10,000	.1919	.3938145	0	1

Using factor variables

```
. summarize d1##d2
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
          d1 |
          1  |     10,000          .2      .40002          0          1
          2  |     10,000       .2044    .4032827          0          1
          3  |     10,000       .2037    .4027686          0          1
          4  |     10,000       .1919    .3938145          0          1
             |
        1.d2 |     10,000       .4986     .500023          0          1
-------------+---------------------------------------------------------
             |
       d1#d2 |
        1 1  |     10,000       .1009    .3012113          0          1
        2 1  |     10,000       .1007    .3009461          0          1
        3 1  |     10,000       .1035     .304626          0          1
        4 1  |     10,000       .0922    .2893225          0          1
```

Using factor variables

```
. summarize c.x1##c.x1 c.x1#c.x2 c.x1#i.d1, separator(4)
```

Variable	Obs	Mean	Std. Dev.	Min	Max
x1	10,000	.0110258	.9938621	-4.095795	3.714316
c.x1#c.x1	10,000	.9877847	1.416602	4.18e-09	16.77553
c.x1#c.x2	10,000	.000208	1.325283	-7.469295	6.45778
d1#c.x1					
1	10,000	.0044334	.4516058	-3.021819	3.286315
2	10,000	.0008424	.4432188	-4.095795	3.178586
3	10,000	.0025783	.4533505	-3.374062	3.428311
4	10,000	-.0014739	.4379122	-3.161604	3.714316

Models and Quantities of Interest

- We usually model an outcome of interest, Y, conditional on covariates of interest X:
  - $E(Y|X) = X\beta$ (regression)
  - $E(Y|X) = \exp(X\beta)$ (poisson)
  - $E(Y|X) = P(Y|X) = \Phi(X\beta)$ (probit)
  - $E(Y|X) = P(Y|X) = [\exp(X\beta)][1\iota + \exp(X\beta)]^{-1}$ (logit)
  - $E(Y|X) = g(X)$ (nonparametric regression)

Models and Quantities of Interest

- We usually model an outcome of interest, Y, conditional on covariates of interest X:
  - $E(Y|X) = X\beta$ (regression)
  - $E(Y|X) = \exp(X\beta)$ (poisson)
  - $E(Y|X) = P(Y|X) = \Phi(X\beta)$ (probit)
  - $E(Y|X) = P(Y|X) = [\exp(X\beta)][1\iota + \exp(X\beta)]^{-1}$ (logit)
  - $E(Y|X) = g(X)$ (nonparametric regression)

Models and Quantities of Interest

- We usually model an outcome of interest, Y, conditional on covariates of interest X:
  - $E(Y|X) = X\beta$ (regression)
  - $E(Y|X) = \exp(X\beta)$ (poisson)
  - $E(Y|X) = P(Y|X) = \Phi(X\beta)$ (probit)
  - $E(Y|X) = P(Y|X) = [\exp(X\beta)][1\iota + \exp(X\beta)]^{-1}$ (logit)
  - $E(Y|X) = g(X)$ (nonparametric regression)

Models and Quantities of Interest

- We usually model an outcome of interest, Y, conditional on covariates of interest X:
  - $E(Y|X) = X\beta$ (regression)
  - $E(Y|X) = \exp(X\beta)$ (poisson)
  - $E(Y|X) = P(Y|X) = \Phi(X\beta)$ (probit)
  - $E(Y|X) = P(Y|X) = [\exp(X\beta)][1\iota + \exp(X\beta)]^{-1}$ (logit)
  - $E(Y|X) = g(X)$ (nonparametric regression)

Questions

- Population averaged
  - Does a medicaid expansion improve health outcomes ?
  - What is the effect of a minimum wage increase on employment ?
  - What is the effect on urban violence indicators, during the weekends of moving back the city curfew ?
- At a point
  - What is the effect of loosing weight for a 36 year, overweight hispanic man?
  - What is the effect on urban violence indicators, during the weekends of moving back the city curfew, for a large city, in the southwest of the United States ?

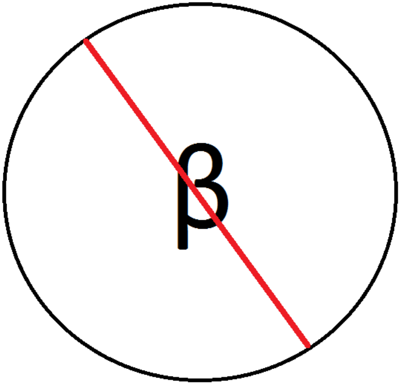
Questions

- Population averaged
  - Does a medicaid expansion improve health outcomes ?
  - What is the effect of a minimum wage increase on employment ?
  - What is the effect on urban violence indicators, during the weekends of moving back the city curfew ?
- At a point
  - What is the effect of loosing weight for a 36 year, overweight hispanic man?
  - What is the effect on urban violence indicators, during the weekends of moving back the city curfew, for a large city, in the southwest of the United States ?

Questions

- Population averaged
  - Does a medicaid expansion improve health outcomes ?
  - What is the effect of a minimum wage increase on employment ?
  - What is the effect on urban violence indicators, during the weekends of moving back the city curfew ?
- At a point
  - What is the effect of loosing weight for a 36 year, overweight hispanic man?
  - What is the effect on urban violence indicators, during the weekends of moving back the city curfew, for a large city, in the southwest of the United States ?

What are the answers?



A linear model

$$y = \beta_0 + x_1\beta_1 + x_2\beta_2 + x_1^2\beta_3 + x_2^2\beta_4 + x_1x_2\beta_5$$
$$+ d_1\beta_6 + d_2\beta_7 + d_1d_2\beta_8 + x_2d_1\beta_9 + \varepsilon$$

- x_1 and x_2 are continuous, d_2 is binary, and d_1 has 5 categories.
- There are interactions of continuous and categorical variables
- This is simulated data

A linear model

$$y = \beta_0 + x_1\beta_1 + x_2\beta_2 + x_1^2\beta_3 + x_2^2\beta_4 + x_1x_2\beta_5 \\ + d_1\beta_6 + d_2\beta_7 + d_1d_2\beta_8 + x_2d_1\beta_9 + \varepsilon$$

- x_1 and x_2 are continuous, d_2 is binary, and d_1 has 5 categories.
- There are interactions of continuous and categorical variables
- This is simulated data

Regression results

```
. regress yr c.x1##c.x2 c.x1#c.x1 c.x2#c.x2 i.d1##i.d2 c.x2#i.d1
      Source |       SS           df       MS      Number of obs   =    10,000
-------------+----------------------------------   F(18, 9981)     =    388.10
       Model |  335278.744        18  18626.5969   Prob > F        =    0.0000
    Residual |  479031.227     9,981  47.9943119   R-squared       =    0.4117
-------------+----------------------------------   Adj R-squared   =    0.4107
       Total |  814309.971     9,999   81.439141   Root MSE        =    6.9278

------------------------------------------------------------------------------
          yr |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
          x1 |   -1.04884   .1525255    -6.88   0.000    -1.347821   -.7498593
          x2 |   .4749664   .4968878     0.96   0.339    -.4990339    1.448967
             |
   c.x1#c.x2 |    1.06966   .1143996     9.35   0.000     .8454139    1.293907
             |
   c.x1#c.x1 |  -1.061312    .048992   -21.66   0.000    -1.157346   -.9652779
             |
   c.x2#c.x2 |   1.177785   .1673487     7.04   0.000      .849748    1.505822
             |
          d1 |
          1  |  -1.504705   .5254654    -2.86   0.004    -2.534723   -.4746865
          2  |  -3.727184   .5272623    -7.07   0.000    -4.760725   -2.693644
          3  |  -6.522121   .5229072   -12.47   0.000    -7.547125   -5.497118
          4  |   -8.80982   .5319266   -16.56   0.000    -9.852503   -7.767136
             |
        1.d2 |   1.615761   .3099418     5.21   0.000     1.008212    2.223309
             |
       d1#d2 |
         1 1 |  -3.649372   .4383277    -8.33   0.000    -4.508582   -2.790161
         2 1 |  -5.994454    .435919   -13.75   0.000    -6.848943   -5.139965
         3 1 |  -8.457034   .4364173   -19.38   0.000      -9.3125   -7.601568
         4 1 |  -11.04842   .4430598   -24.94   0.000     -11.9169   -10.17993
             |
     d1#c.x2 |
          1  |    1.11805   .3626989     3.08   0.002     .4070865    1.829013
          2  |   1.918298   .3592232     5.34   0.000     1.214149    2.622448
          3  |   3.484255   .3594559     9.69   0.000     2.779649    4.188861
          4  |   4.260699    .362315    11.76   0.000     3.550488    4.970909
```

Effects: x_2

Suppose we want to study the marginal effect of x_2

$$\frac{\partial E\left(y|x_1, x_2, d_1, d_2\right)}{\partial x_2}$$

This is given by

$$\frac{\partial E\left(y|x_1, x_2, d_1, d_2\right)}{\partial x_2} = \beta_2 + 2x_2\beta_4 + x_1\beta_5 + d_1\beta_9$$

- I can compute this effect for every individual in my sample and then average to get a population averaged effect
- I could evaluate this conditional on values of the different covariates, or even values of importance for x_2

Effects: x_2

Suppose we want to study the marginal effect of x_2

$$\frac{\partial E\left(y|x_1, x_2, d_1, d_2\right)}{\partial x_2}$$

This is given by

$$\frac{\partial E\left(y|x_1, x_2, d_1, d_2\right)}{\partial x_2} = \beta_2 + 2x_2\beta_4 + x_1\beta_5 + d_1\beta_9$$

- I can compute this effect for every individual in my sample and then average to get a population averaged effect
- I could evaluate this conditional on values of the different covariates, or even values of importance for x_2

Population averaged effect manually

```
. regress, coeflegend
      Source |       SS           df       MS      Number of obs   =    10,000
-------------+----------------------------------   F(18, 9981)     =    388.10
       Model |  335278.744        18  18626.5969   Prob > F        =    0.0000
    Residual |  479031.227     9,981  47.9943119   R-squared       =    0.4117
-------------+----------------------------------   Adj R-squared   =    0.4107
       Total |  814309.971     9,999   81.439141   Root MSE        =    6.9278
------------------------------------------------------------------------------
          yr |      Coef.  Legend
-------------+----------------------------------------------------------------
          x1 |   -1.04884  _b[x1]
          x2 |   .4749664  _b[x2]
             |
   c.x1#c.x2 |    1.06966  _b[c.x1#c.x2]
             |
   c.x1#c.x1 |  -1.061312  _b[c.x1#c.x1]
             |
   c.x2#c.x2 |   1.177785  _b[c.x2#c.x2]
             |
          d1 |
          1  |  -1.504705  _b[1.d1]
          2  |  -3.727184  _b[2.d1]
          3  |  -6.522121  _b[3.d1]
          4  |   -8.80982  _b[4.d1]
             |
        1.d2 |   1.615761  _b[1.d2]
             |
       d1#d2 |
        1 1  |  -3.649372  _b[1.d1#1.d2]
        2 1  |  -5.994454  _b[2.d1#1.d2]
        3 1  |  -8.457034  _b[3.d1#1.d2]
        4 1  |  -11.04842  _b[4.d1#1.d2]
             |
     d1#c.x2 |
          1  |    1.11805  _b[1.d1#c.x2]
          2  |   1.918298  _b[2.d1#c.x2]
          3  |   3.484255  _b[3.d1#c.x2]
          4  |   4.260699  _b[4.d1#c.x2]
```

Population averaged effect manually

```
 generate double dydx2 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 + ///
_b[1.d1#c.x2]*1.d1 + _b[2.d1#c.x2]*2.d1 + ///
_b[3.d1#c.x2]*3.d1 + _b[4.d1#c.x2]*4.d1
```

Population averaged effect manually

```
. list dydx2 in 1/10, sep(0)

     +-----------+
     |     dydx2 |
     |-----------|
  1. | 4.6587219 |
  2. | 4.3782089 |
  3. | 7.8509027 |
  4. | 10.018247 |
  5. | 7.4219045 |
  6. | 7.2065007 |
  7. | 3.6052012 |
  8. | 5.4846114 |
  9. | 6.3144353 |
 10. | 5.9827419 |
     +-----------+

. summarize dydx2

    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
       dydx2 |     10,000     5.43906    2.347479  -2.075498   12.90448
```

Population averaged effect manually

```
. list dydx2 in 1/10, sep(0)

     +-----------+
     |     dydx2 |
     |-----------|
  1. | 4.6587219 |
  2. | 4.3782089 |
  3. | 7.8509027 |
  4. | 10.018247 |
  5. | 7.4219045 |
  6. | 7.2065007 |
  7. | 3.6052012 |
  8. | 5.4846114 |
  9. | 6.3144353 |
 10. | 5.9827419 |
     +-----------+

. summarize dydx2
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
       dydx2 |     10,000     5.43906    2.347479  -2.075498   12.90448
```

`margins`

- A way to compute effects of interest and their standard errors
- Fundamental to construct our unified framework
- Consumes factor variable notation
- Operates over Stata `predict`, $\widehat{E(Y|X)} = X\widehat{\beta}$

margins, dydx(*)

```
. margins, dydx(x2)
Average marginal effects                        Number of obs     =     10,000
Model VCE    : OLS

Expression   : Linear prediction, predict()
dy/dx w.r.t. : x2
```

	dy/dx	Delta-method Std. Err.	t	P>|t|	[95% Conf.	Interval]
x2	5.43906	.1188069	45.78	0.000	5.206174	5.671945

- Expression, default prediction $E(Y|X) = X\beta$
  - This means you could access other Stata predictions
  - Or any function of the coefficients
- Delta method is the way the standard errors are computed

Expression

```
. margins, expression(_b[c.x2] +                                  ///
> _b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 +                          ///
> _b[1.d1#c.x2]*1.d1 + _b[2.d1#c.x2]*2.d1 +                            ///
> _b[3.d1#c.x2]*3.d1 + _b[4.d1#c.x2]*4.d1)
Warning: expression() does not contain predict() or xb().
Predictive margins                              Number of obs     =     10,000
Model VCE    : OLS
Expression   : _b[c.x2] + _b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 + _b[1.d1#c.x2]*1.d1 +
               _b[2.d1#c.x2]*2.d1 + _b[3.d1#c.x2]*3.d1 + _b[4.d1#c.x2]*4.d1
------------------------------------------------------------------------------
             |            Delta-method
             |     Margin   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       _cons |    5.43906   .1188069    45.78   0.000     5.206202    5.671917
------------------------------------------------------------------------------
```

Delta Method and Standard Errors

We get our standard errors from the central limit theorem.

$$\widehat{\beta} - \beta \xrightarrow{d} N(0, V)$$

We can get standard errors for any smooth function $g()$ of $\widehat{\beta}$ with

$$g\left(\widehat{\beta}\right) - g(\beta) \xrightarrow{d} N\left(0, g'(\beta)' V g'(\beta)\right)$$

Effect of x_2: revisited

$$\frac{\partial E\left(y|x_1, x_2, d_1, d_2\right)}{\partial x_2} = \beta_2 + 2x_2\beta_4 + x_1\beta_5 + d_1\beta_9$$

- We averaged this function but could evaluate it at different values of the covariates for example:
  - What is the average marginal effect of x_2 for different values of d_1
  - What is the average marginal effect of x_2 for different values of d_1 and x_1

Effect of x_2: revisited

$$\frac{\partial E\left(y|x_1, x_2, d_1, d_2\right)}{\partial x_2} = \beta_2 + 2x_2\beta_4 + x_1\beta_5 + d_1\beta_9$$

- We averaged this function but could evaluate it at different values of the covariates for example:
  - What is the average marginal effect of x_2 for different values of d_1
- Counterfactual: What if everyone in the population had a level of $d_1 = 0$. What if $d_1 = 1$, ...

Effect of x_2: revisited

$$\frac{\partial E\left(y|x_1, x_2, d_1, d_2\right)}{\partial x_2} = \beta_2 + 2x_2\beta_4 + x_1\beta_5 + d_1\beta_9$$

- We averaged this function but could evaluate it at different values of the covariates for example:
  - What is the average marginal effect of x_2 for different values of d_1
- Counterfactual: What if everyone in the population had a level of $d_1 = 0$. What if $d_1 = 1$, ...

Different values of d_1 a counterfactual

```
generate double dydx2 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 + ///
_b[1.d1#c.x2]*1.d1 + _b[2.d1#c.x2]*2.d1 + ///
_b[3.d1#c.x2]*3.d1 + _b[4.d1#c.x2]*4.d1

generate double dydx2_d10 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2

generate double dydx2_d11 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 + ///
_b[1.d1#c.x2]

generate double dydx2_d12 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 + ///
_b[2.d1#c.x2]
```

Different values of d_1 a counterfactual

```
 generate double dydx2 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 + ///
_b[1.d1#c.x2]*1.d1 + _b[2.d1#c.x2]*2.d1 + ///
_b[3.d1#c.x2]*3.d1 + _b[4.d1#c.x2]*4.d1

generate double dydx2_d10 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2

generate double dydx2_d11 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 + ///
_b[1.d1#c.x2]

generate double dydx2_d12 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 + ///
_b[2.d1#c.x2]
```

Different values of d_1 a counterfactual

```
 generate double dydx2 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 + ///
_b[1.d1#c.x2]*1.d1 + _b[2.d1#c.x2]*2.d1 + ///
_b[3.d1#c.x2]*3.d1 + _b[4.d1#c.x2]*4.d1

generate double dydx2_d10 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2

generate double dydx2_d11 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 + ///
_b[1.d1#c.x2]

generate double dydx2_d12 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 + ///
_b[2.d1#c.x2]
```

Different values of d_1 a counterfactual

```
 generate double dydx2 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 + ///
_b[1.d1#c.x2]*1.d1 + _b[2.d1#c.x2]*2.d1 + ///
_b[3.d1#c.x2]*3.d1 + _b[4.d1#c.x2]*4.d1

generate double dydx2_d10 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2

generate double dydx2_d11 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 + ///
_b[1.d1#c.x2]

generate double dydx2_d12 = _b[c.x2] + ///
_b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 + ///
_b[2.d1#c.x2]
```

Average marginal effect of x_2 at counterfactuals: manually

```
. summarize dydx2_*
```

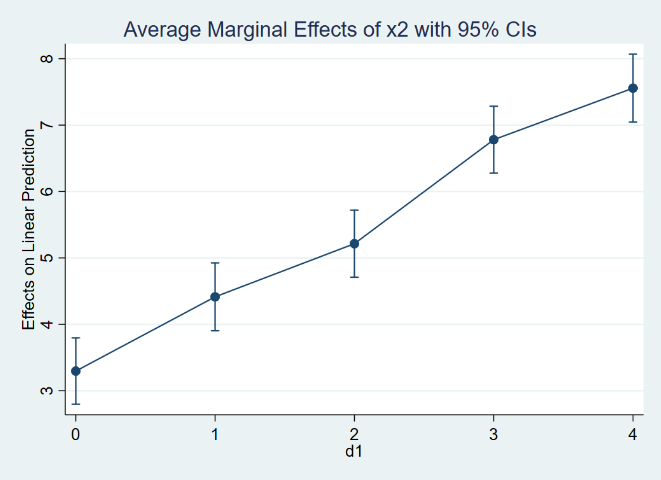
Variable	Obs	Mean	Std. Dev.	Min	Max
dydx2_d10	10,000	3.295979	1.7597	-2.411066	9.288564
dydx2_d11	10,000	4.414028	1.7597	-1.293017	10.40661
dydx2_d12	10,000	5.214277	1.7597	-.4927681	11.20686
dydx2_d13	10,000	6.780233	1.7597	1.073188	12.77282
dydx2_d14	10,000	7.556677	1.7597	1.849632	13.54926

Average marginal effect of x_2 at counterfactuals: `margins`

```
. margins d1, dydx(x2)
Average marginal effects                        Number of obs     =     10,000
Model VCE    : OLS

Expression   : Linear prediction, predict()
dy/dx w.r.t. : x2
```

	dy/dx	Delta-method Std. Err.	t	P>|t|	[95% Conf.	Interval]
x2						
d1						
0	3.295979	.2548412	12.93	0.000	2.796439	3.795519
1	4.414028	.2607174	16.93	0.000	3.90297	4.925087
2	5.214277	.2575936	20.24	0.000	4.709342	5.719212
3	6.780233	.2569613	26.39	0.000	6.276537	7.283929
4	7.556677	.2609514	28.96	0.000	7.04516	8.068195

Graphically: `marginsplot`



Thou shalt not be fooled by overlapping confidence intervals

$$Var(a - b) = Var(a) + Var(b) - 2Cov(a, b)$$

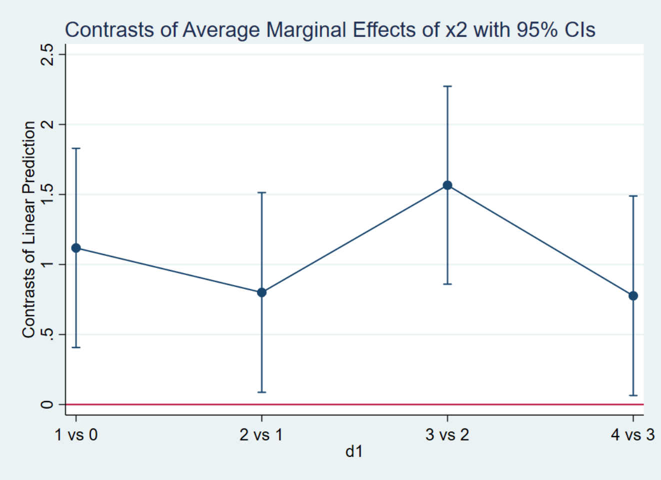
- You have $Var(a)$ and $Var(b)$
- You do not have $2Cov(a, b)$

Thou shalt not be fooled by overlapping confidence intervals

```
. margins ar.d1, dydx(x2) contrast(nowald)
Contrasts of average marginal effects
Model VCE    : OLS
Expression   : Linear prediction, predict()
dy/dx w.r.t. : x2
```

	Contrast dy/dx	Delta-method Std. Err.	[95% Conf.	Interval]
x2				
d1				
(1 vs 0)	1.11805	.3626989	.4070865	1.829013
(2 vs 1)	.8002487	.3638556	.0870184	1.513479
(3 vs 2)	1.565956	.3603585	.859581	2.272332
(4 vs 3)	.7764441	.3634048	.0640974	1.488791

Thou shalt not be fooled by overlapping confidence intervals



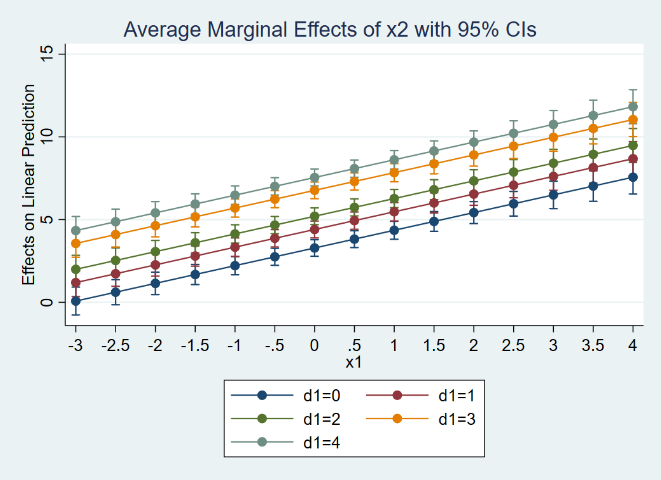
Effect of x_2: revisited

$$\frac{\partial E\left(y|x_1, x_2, d_1, d_2\right)}{\partial x_2} = \beta_2 + 2x_2\beta_4 + x_1\beta_5 + d_1\beta_9$$

- We averaged this function but could evaluate it at different values of the covariates for example:
  - What is the average marginal effect of x_2 for different values of d_1 and x_1

Effect of x_2: revisited

```
margins d1, dydx(x2) at(x1=(-3(.5)4))
```



Put on your calculus hat or ask a different question

$$\frac{\partial E\left(y|.\right)}{\partial x_2}$$

- This is our object of interest
- By definition it is the change in $E\left(y|.\right)$ for an infinitesimal change in x_2
- Sometimes people talk about this as a unit change in x_2

Put on your calculus hat or ask a different question

$$\frac{\partial E\left(y|.\right)}{\partial x_2}$$

- This is our object of interest
- By definition it is the change in $E\left(y|.\right)$ for an infinitesimal change in x_2
- Sometimes people talk about this as a unit change in x_2

Put on your calculus hat or ask a different question

```
. margins, dydx(x2)
Average marginal effects                        Number of obs     =     10,000
Model VCE    : OLS

Expression   : Linear prediction, predict()
dy/dx w.r.t. : x2
```

	dy/dx	Delta-method Std. Err.	t	P>|t|	[95% Conf.	Interval]
x2	5.43906	.1188069	45.78	0.000	5.206174	5.671945

```
. quietly predict double xb0
. quietly replace x2 = x2 + 1
. quietly predict double xb1
. generate double diff = xb1 - xb0
. summarize diff
```

Variable	Obs	Mean	Std. Dev.	Min	Max
diff	10,000	6.616845	2.347479	-.8977125	14.08226

Put on your calculus hat or ask a different question

```
. margins, dydx(x2)
Average marginal effects                        Number of obs     =     10,000
Model VCE    : OLS
Expression   : Linear prediction, predict()
dy/dx w.r.t. : x2
```

	dy/dx	Delta-method Std. Err.	t	P>|t|	[95% Conf.	Interval]
x2	5.43906	.1188069	45.78	0.000	5.206174	5.671945

```
. quietly predict double xb0
. quietly replace x2 = x2 + 1
. quietly predict double xb1
. generate double diff = xb1 - xb0
. summarize diff
```

Variable	Obs	Mean	Std. Dev.	Min	Max
diff	10,000	6.616845	2.347479	-.8977125	14.08226

Put on your calculus hat or ask a different question

```
. margins, at(x2 = generate(x2)) at(x2=generate(x2+1))
Predictive margins                              Number of obs     =     10,000
Model VCE    : OLS
Expression   : Linear prediction, predict()
1._at        : x2               = x2
2._at        : x2               = x2+1

------------------------------------------------------------------------------
             |            Delta-method
             |     Margin   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         _at |
          1  |   -.599745   .0692779    -8.66   0.000    -.7355437   -.4639463
          2  |     6.0171   .1909195    31.52   0.000     5.642859     6.39134
------------------------------------------------------------------------------

. margins, at(x2 = generate(x2)) at(x2=generate(x2+1)) contrast(at(r) nowald)
Contrasts of predictive margins
Model VCE    : OLS
Expression   : Linear prediction, predict()
1._at        : x2               = x2
2._at        : x2               = x2+1

--------------------------------------------------------------
             |            Delta-method
             |   Contrast   Std. Err.     [95% Conf. Interval]
-------------+------------------------------------------------
         _at |
   (2 vs 1)  |   6.616845   .1779068      6.268111    6.965578
--------------------------------------------------------------

. summarize diff
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
        diff |     10,000    6.616845    2.347479  -.8977125   14.08226
```

Put on your calculus hat or ask a different question

```
. margins, at(x2 = generate(x2)) at(x2=generate(x2+1))
Predictive margins                              Number of obs     =     10,000
Model VCE    : OLS
Expression   : Linear prediction, predict()
1._at        : x2              = x2
2._at        : x2              = x2+1
------------------------------------------------------------------------------
             |            Delta-method
             |     Margin   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         _at |
          1  |   -.599745   .0692779    -8.66   0.000    -.7355437   -.4639463
          2  |     6.0171   .1909195    31.52   0.000     5.642859     6.39134
------------------------------------------------------------------------------
```

```
. margins, at(x2 = generate(x2)) at(x2=generate(x2+1)) contrast(at(r) nowald)
Contrasts of predictive margins
Model VCE    : OLS
Expression   : Linear prediction, predict()
1._at        : x2              = x2
2._at        : x2              = x2+1
--------------------------------------------------------------
             |            Delta-method
             |   Contrast   Std. Err.     [95% Conf. Interval]
-------------+------------------------------------------------
         _at |
   (2 vs 1)  |   6.616845   .1779068      6.268111    6.965578
--------------------------------------------------------------
```

```
. summarize diff
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
        diff |     10,000    6.616845    2.347479  -.8977125   14.08226
```

Put on your calculus hat or ask a different question

```
. margins, at(x2 = generate(x2)) at(x2=generate(x2+1))
Predictive margins                              Number of obs     =     10,000
Model VCE     : OLS
Expression    : Linear prediction, predict()
1._at         : x2              = x2
2._at         : x2              = x2+1
------------------------------------------------------------------------------
             |            Delta-method
             |     Margin   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         _at |
          1  |   -.599745   .0692779    -8.66   0.000    -.7355437   -.4639463
          2  |     6.0171   .1909195    31.52   0.000     5.642859     6.39134
------------------------------------------------------------------------------

. margins, at(x2 = generate(x2)) at(x2=generate(x2+1)) contrast(at(r) nowald)
Contrasts of predictive margins
Model VCE     : OLS
Expression    : Linear prediction, predict()
1._at         : x2              = x2
2._at         : x2              = x2+1
--------------------------------------------------------------
             |            Delta-method
             |   Contrast   Std. Err.     [95% Conf. Interval]
-------------+------------------------------------------------
         _at |
   (2 vs 1)  |   6.616845   .1779068      6.268111    6.965578
--------------------------------------------------------------

. summarize diff
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
        diff |     10,000    6.616845    2.347479  -.8977125   14.08226
```

Ask a different question

- Marginal effects have a meaning in some contexts but are misused
- It is difficult to interpret infinitesimal changes but we do not need to
- We can ask about meaningful questions by talking in units that mean something to the problem we care about

A 10 percent increase in x_2

```
. margins, at(x2 = generate(x2)) at(x2=generate(x2*1.1)) ///
>                    contrast(at(r) nowald)

Contrasts of predictive margins
Model VCE    : OLS

Expression   : Linear prediction, predict()

1._at        : x2                  = x2

2._at        : x2                  = x2*1.1

------------------------------------------------------------------
             |            Delta-method
             |   Contrast   Std. Err.     [95% Conf. Interval]
-------------+----------------------------------------------------
         _at |
   (2 vs 1)  |   .7562394   .0178679      .7212147     .791264
------------------------------------------------------------------
```

What we learned

$$\frac{\partial E\left(y|x_1, x_2, d_1, d_2\right)}{\partial x_2} = \beta_2 + 2x_2\beta_4 + x_1\beta_5 + d_1\beta_9$$

- Population averaged
- Counterfactual values of d_1
- Counterfactual values for d_1 and x_1
- Exploring a fourth dimensional surface

What we learned

$$\frac{\partial E\left(y|x_1, x_2, d_1, d_2\right)}{\partial x_2} = \beta_2 + 2x_2\beta_4 + x_1\beta_5 + d_1\beta_9$$

- Population averaged
- Counterfactual values of d_1
- Counterfactual values for d_1 and x_1
- Exploring a fourth dimensional surface

Discrete covariates

$$E\left(Y|d = d_1, \ldots\right) - E\left(Y|d = d_0, \ldots\right)$$
$$\ldots$$
$$E\left(Y|d = d_k, \ldots\right) - E\left(Y|d = d_0, \ldots\right)$$

- The effect is the difference of the object of interest evaluated at the different levels of the discrete covariate relative to a base level
- It can be interpreted as a treatment effect

Effect of d_1

```
. margins d1
Predictive margins                              Number of obs     =     10,000
Model VCE     : OLS
Expression    : Linear prediction, predict()

             |            Delta-method
             |     Margin   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
          d1 |
          0  |    3.77553   .1550097    24.36   0.000      3.47168    4.079381
          1  |   1.784618   .1550841    11.51   0.000     1.480622    2.088614
          2  |  -.6527544   .1533701    -4.26   0.000    -.9533906   -.3521181
          3  |  -2.807997   .1535468   -18.29   0.000     -3.10898   -2.507014
          4  |  -5.461784   .1583201   -34.50   0.000    -5.772123   -5.151445
```

```
. margins r.d1, contrast(nowald)
Contrasts of predictive margins
Model VCE     : OLS
Expression    : Linear prediction, predict()

             |            Delta-method
             |   Contrast   Std. Err.     [95% Conf. Interval]
-------------+------------------------------------------------
          d1 |
    (1 vs 0) |  -1.990912   .2193128     -2.420809   -1.561015
    (2 vs 0) |  -4.428285   .2180388     -4.855685   -4.000884
    (3 vs 0) |  -6.583527   .2182232     -7.011289   -6.155766
    (4 vs 0) |  -9.237314   .2215769     -9.671649   -8.802979
```

Effect of d_1

```
. margins d1
Predictive margins                              Number of obs     =     10,000
Model VCE    : OLS
Expression   : Linear prediction, predict()
------------------------------------------------------------------------------
             |            Delta-method
             |     Margin   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
          d1 |
          0  |    3.77553   .1550097    24.36   0.000      3.47168    4.079381
          1  |   1.784618   .1550841    11.51   0.000     1.480622    2.088614
          2  |  -.6527544   .1533701    -4.26   0.000    -.9533906   -.3521181
          3  |  -2.807997   .1535468   -18.29   0.000     -3.10898   -2.507014
          4  |  -5.461784   .1583201   -34.50   0.000    -5.772123   -5.151445
------------------------------------------------------------------------------

. margins r.d1, contrast(nowald)
Contrasts of predictive margins
Model VCE    : OLS
Expression   : Linear prediction, predict()
--------------------------------------------------------------
             |            Delta-method
             |   Contrast   Std. Err.     [95% Conf. Interval]
-------------+------------------------------------------------
          d1 |
   (1 vs 0)  |  -1.990912   .2193128     -2.420809   -1.561015
   (2 vs 0)  |  -4.428285   .2180388     -4.855685   -4.000884
   (3 vs 0)  |  -6.583527   .2182232     -7.011289   -6.155766
   (4 vs 0)  |  -9.237314   .2215769     -9.671649   -8.802979
--------------------------------------------------------------
```

Effect of d_1

```
. margins r.d1, contrast(nowald)
Contrasts of predictive margins
Model VCE    : OLS
Expression   : Linear prediction, predict()
```

	Contrast	Delta-method Std. Err.	[95% Conf.	Interval]
d1				
(1 vs 0)	-1.990912	.2193128	-2.420809	-1.561015
(2 vs 0)	-4.428285	.2180388	-4.855685	-4.000884
(3 vs 0)	-6.583527	.2182232	-7.011289	-6.155766
(4 vs 0)	-9.237314	.2215769	-9.671649	-8.802979

```
. margins, dydx(d1)
Average marginal effects                    Number of obs     =     10,000
Model VCE    : OLS
Expression   : Linear prediction, predict()
dy/dx w.r.t. : 1.d1 2.d1 3.d1 4.d1
```

	dy/dx	Delta-method Std. Err.	t	P>|t|	[95% Conf.	Interval]
d1						
1	-1.990912	.2193128	-9.08	0.000	-2.420809	-1.561015
2	-4.428285	.2180388	-20.31	0.000	-4.855685	-4.000884
3	-6.583527	.2182232	-30.17	0.000	-7.011289	-6.155766
4	-9.237314	.2215769	-41.69	0.000	-9.671649	-8.802979

Note: dy/dx for factor levels is the discrete change from the base level.

Effect of d_1

```
. margins r.d1, contrast(nowald)
Contrasts of predictive margins
Model VCE      : OLS
Expression     : Linear prediction, predict()
```

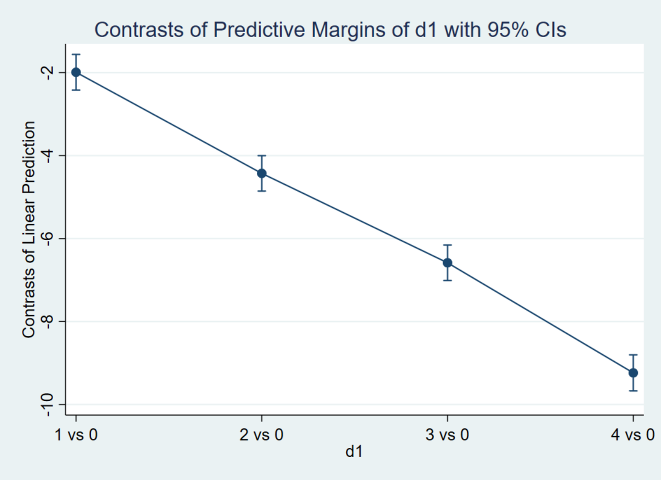
	Contrast	Delta-method Std. Err.	[95% Conf.	Interval]
d1				
(1 vs 0)	-1.990912	.2193128	-2.420809	-1.561015
(2 vs 0)	-4.428285	.2180388	-4.855685	-4.000884
(3 vs 0)	-6.583527	.2182232	-7.011289	-6.155766
(4 vs 0)	-9.237314	.2215769	-9.671649	-8.802979

```
. margins, dydx(d1)
Average marginal effects                       Number of obs     =     10,000
Model VCE      : OLS
Expression     : Linear prediction, predict()
dy/dx w.r.t. : 1.d1 2.d1 3.d1 4.d1
```

	dy/dx	Delta-method Std. Err.	t	P>|t|	[95% Conf.	Interval]
d1						
1	-1.990912	.2193128	-9.08	0.000	-2.420809	-1.561015
2	-4.428285	.2180388	-20.31	0.000	-4.855685	-4.000884
3	-6.583527	.2182232	-30.17	0.000	-7.011289	-6.155766
4	-9.237314	.2215769	-41.69	0.000	-9.671649	-8.802979

Note: dy/dx for factor levels is the discrete change from the base level.

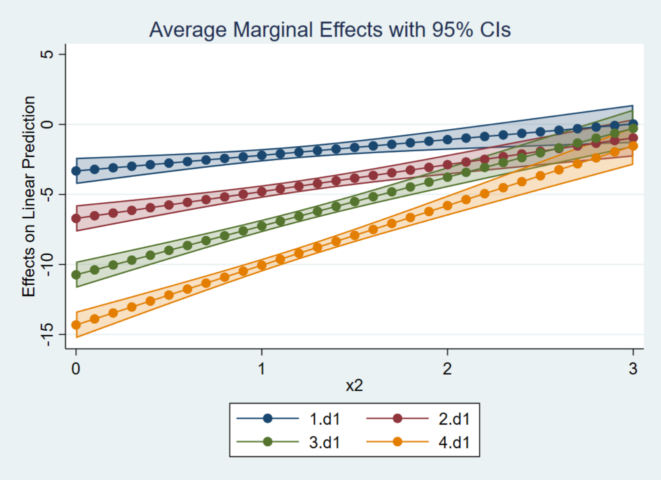
Effect of d_1



Contrasts of Predictive Margins of d1 with 95% CIs
Contrasts of Linear Prediction
-2
-4
-6
-8
-10
1 vs 0
2 vs 0
3 vs 0
4 vs 0
d1

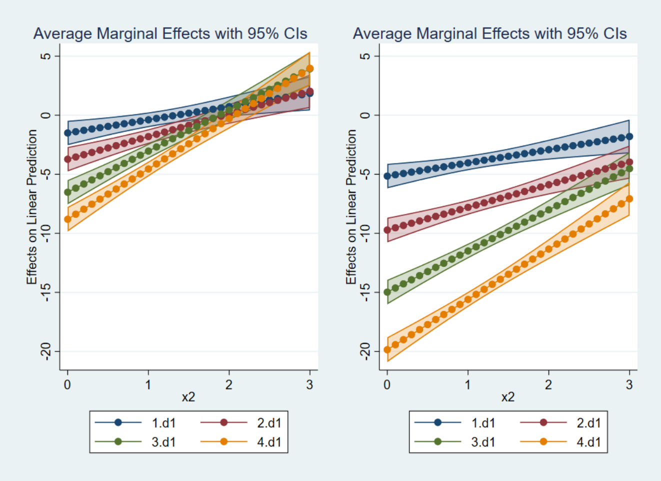
Effect of d_1 for x_2 counterfactuals

```
margins, dydx(d1) at(x2=(0(.5)3))
marginsplot, recastci(rarea) ciopts(fcolor(%30))
```



Effect of d_1 for x_2 and d_2 counterfactuals

```
margins 0.d2, dydx(d1) at(x2=(0(.5)3))
margins 1.d2, dydx(d1) at(x2=(0(.5)3))
marginsplot, recastci(rarea) ciopts(fcolor(%30))
```



Effect of x_2 and d_1 or x_2 and x_1

- We can think about changes of two variables at a time
- This is a bit trickier to interpret and a bit trickier to compute
- `margins` allows us to solve this problem elegantly

A change in x_2 and d_1

```
. margins r.d1, dydx(x2) contrast(nowald)
Contrasts of average marginal effects
Model VCE      : OLS

Expression     : Linear prediction, predict()
dy/dx w.r.t. : x2
```

	Contrast dy/dx	Delta-method Std. Err.	[95% Conf.	Interval]
x2				
d1				
(1 vs 0)	1.11805	.3626989	.4070865	1.829013
(2 vs 0)	1.918298	.3592232	1.214149	2.622448
(3 vs 0)	3.484255	.3594559	2.779649	4.188861
(4 vs 0)	4.260699	.362315	3.550488	4.970909

A change in d_1 and d_2

```
. margins r.d1, dydx(d2) contrast(nowald)
Contrasts of average marginal effects
Model VCE    : OLS

Expression   : Linear prediction, predict()
dy/dx w.r.t. : 1.d2
```

	Contrast dy/dx	Delta-method Std. Err.	[95% Conf.	Interval]
0.d2	(base outcome)			
1.d2				
d1				
(1 vs 0)	-3.649372	.4383277	-4.508582	-2.790161
(2 vs 0)	-5.994454	.435919	-6.848943	-5.139965
(3 vs 0)	-8.457034	.4364173	-9.3125	-7.601568
(4 vs 0)	-11.04842	.4430598	-11.9169	-10.17993

```
Note: dy/dx for factor levels is the discrete change from the
      base level.
```

A change in x_2 and x_1

```
. margins, expression(_b[c.x2] +                                        ///
> _b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 +                         ///
> _b[1.d1#c.x2]*1.d1 + _b[2.d1#c.x2]*2.d1 +                           ///
> _b[3.d1#c.x2]*3.d1 + _b[4.d1#c.x2]*4.d1)                            ///
> dydx(x1)
Warning: expression() does not contain predict() or xb().
Average marginal effects                        Number of obs     =     10,000
Model VCE    : OLS
Expression   : _b[c.x2] + _b[c.x1#c.x2]*c.x1 + 2*_b[c.x2#c.x2]*c.x2 + _b[1.d1#c.x2]*1.d1 +
               _b[2.d1#c.x2]*2.d1 + _b[3.d1#c.x2]*3.d1 + _b[4.d1#c.x2]*4.d1
dy/dx w.r.t. : x1
```

	dy/dx	Delta-method Std. Err.	z	P>|z|	[95% Conf.	Interval]
x1	1.06966	.1143996	9.35	0.000	.8454411	1.293879

Framework

- An object of interest, $E(Y|X)$
- Questions
  - $\frac{\partial E(Y|X)}{\partial x_k}$
  - $E(Y|d = d_{level})$ - $E(Y|d = d_{base})$
  - Both
  - Second order terms, double derivatives
- Explore the surface
  - Population averaged
  - Effects at fixed values of covariates (counterfactuals)

Framework

- An object of interest, $E(Y|X)$
- Questions
  - $\frac{\partial E(Y|X)}{\partial x_k}$
  - $E(Y|d = d_{level})$ - $E(Y|d = d_{base})$
  - Both
  - Second order terms, double derivatives
- Explore the surface
  - Population averaged
  - Effects at fixed values of covariates (counterfactuals)

Framework

- An object of interest, $E(Y|X)$
- Questions
  - $\frac{\partial E(Y|X)}{\partial x_k}$
  - $E(Y|d = d_{level})$ - $E(Y|d = d_{base})$
  - Both
  - Second order terms, double derivatives
- Explore the surface
  - Population averaged
  - Effects at fixed values of covariates (counterfactuals)

Binary outcome models

- The data generating process is given by:

$$y = \begin{cases} 1 & \text{if} \quad y^* = x\beta + \varepsilon > 0 \\ 0 & \text{otherwise} \end{cases}$$

- We make an assumption on the distribution of ε, f_ε
  - Probit: ε follows a standard normal distribution
  - Logit: ε follows a standard logistic distribution
  - By construction $P(y = 1|x) = F(x\beta)$
- This gives rise to two models:
  1. If $F(.)$ is the standard normal distribution we have a **Probit**
  2. If $F(.)$ is the logistic distribution we have a **Logit** model
- $P(y = 1|x) = E(y|x)$

Binary outcome models

- The data generating process is given by:

$$y = \begin{cases} 1 & \text{if} \quad y^* = x\beta + \varepsilon > 0 \\ 0 & \text{otherwise} \end{cases}$$

- We make an assumption on the distribution of ε, f_ε
  - Probit: ε follows a standard normal distribution
  - Logit: ε follows a standard logistic distribution
  - By construction $P(y = 1|x) = F(x\beta)$
- This gives rise to two models:
  1. If $F(.)$ is the standard normal distribution we have a **Probit**
  2. If $F(.)$ is the logistic distribution we have a **Logit** model
- $P(y = 1|x) = E(y|x)$

Binary outcome models

- The data generating process is given by:

$$y = \begin{cases} 1 & \text{if} \quad y^* = x\beta + \varepsilon > 0 \\ 0 & \text{otherwise} \end{cases}$$

- We make an assumption on the distribution of ε, f_ε
  - Probit: ε follows a standard normal distribution
  - Logit: ε follows a standard logistic distribution
  - By construction $P(y = 1|x) = F(x\beta)$
- This gives rise to two models:
  1. If $F(.)$ is the standard normal distribution we have a **Probit**
  2. If $F(.)$ is the logistic distribution we have a **Logit** model
- $P(y = 1|x) = E(y|x)$

Effects

- The change in the conditional probability due to a change in a covariate is given by

$$\frac{\partial P(y|x)}{\partial x_k} = \frac{\partial F(x\beta)}{\partial x_k}\beta_k = f(x\beta)\beta_k$$

- This implies that:
  1. The value of the object of interest depends on x
  2. The β coefficients only tell us the sign of the effect given that $f(x\beta) > 0$ almost surely
- For a categorical variable (factor variables)

$$F(x\beta|d = d_l) - F(x\beta|d = d_0)$$

Coefficient table

```
. probit ypr c.x1##c.x2 i.d1##i.d2 i.d1#c.x1, nolog
Probit regression                               Number of obs     =     10,000
                                                LR chi2(16)       =    2942.75
                                                Prob > chi2       =     0.0000
Log likelihood = -5453.1739                     Pseudo R2         =     0.2125
```

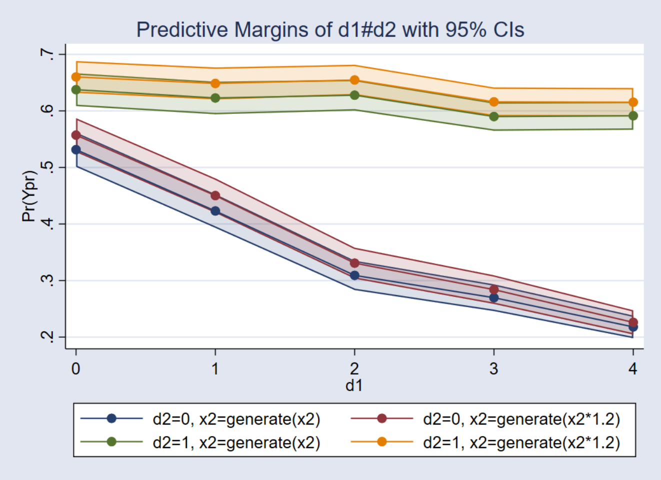
ypr	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
x1	-.3271742	.0423777	-7.72	0.000	-.4102329	-.2441155
x2	.3105438	.023413	13.26	0.000	.2646551	.3564325
c.x1#c.x2	.3178514	.0258437	12.30	0.000	.2671987	.3685041
d1						
1	-.2927285	.057665	-5.08	0.000	-.4057498	-.1797072
2	-.6605838	.0593125	-11.14	0.000	-.7768342	-.5443333
3	-.9137215	.0647033	-14.12	0.000	-1.040538	-.7869054
4	-1.27621	.0718132	-17.77	0.000	-1.416961	-1.135459
1.d2	.2822199	.057478	4.91	0.000	.1695651	.3948747
d1#d2						
1 1	.2547359	.0818174	3.11	0.002	.0943767	.4150951
2 1	.6621119	.0839328	7.89	0.000	.4976066	.8266171
3 1	.8471544	.0893541	9.48	0.000	.6720237	1.022285
4 1	1.26051	.0999602	12.61	0.000	1.064592	1.456429
d1#c.x1						
1	-.2747025	.0422351	-6.50	0.000	-.3574819	-.1919232
2	-.5640486	.0452423	-12.47	0.000	-.6527219	-.4753753
3	-.9452172	.0512391	-18.45	0.000	-1.045644	-.8447905
4	-1.220619	.0608755	-20.05	0.000	-1.339933	-1.101306

Effects of x_2

```
. margins, at(x2=generate(x2)) at(x2=generate(x2*1.2))
Predictive margins                              Number of obs     =     10,000
Model VCE    : OIM

Expression   : Pr(ypr), predict()

1._at        : x2               = x2

2._at        : x2               = x2*1.2

------------------------------------------------------------------------------
             |            Delta-method
             |     Margin   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         _at |
          1  |   .4817093   .0043106   111.75   0.000     .4732607    .4901579
          2  |   .5039467   .0046489   108.40   0.000     .4948349    .5130585
------------------------------------------------------------------------------
```

Effects of x_2 at values of d_1 and d_2

```
margins d1#d2,
at(x2=generate(x2))at(x2=generate(x2*1.2))
```



Logit vs. Probit

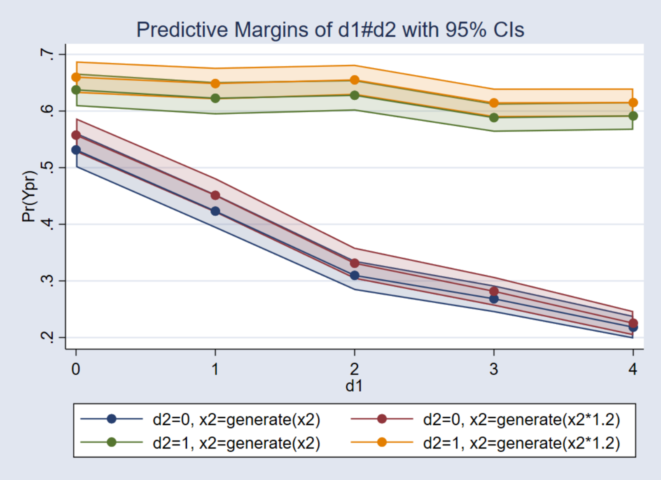
```
 . quietly logit ypr c.x1##c.x2 i.d1##i.d2 i.d1#c.x1
. quietly margins d1#d2, at(x2=generate(x2))at(x2=generate(x2*1.2)) post
. estimates store logit
. quietly probit ypr c.x1##c.x2 i.d1##i.d2 i.d1#c.x1
. quietly margins d1#d2, at(x2=generate(x2))at(x2=generate(x2*1.2)) post
. estimates store probit
```

Logit vs. Probit

```
. estimates table probit logit
```

Variable	probit	logit
_at#d1#d2		
1 0 0	.53151657	.53140462
1 0 1	.63756257	.63744731
1 1 0	.42306578	.42322182
1 1 1	.62291206	.62262466
1 2 0	.30922733	.30975991
1 2 1	.62783902	.62775349
1 3 0	.26973385	.26845746
1 3 1	.59004519	.58834989
1 4 0	.21809081	.21827411
1 4 1	.5914183	.59140961
2 0 0	.55723572	.55751404
2 0 1	.66005549	.65979041
2 1 0	.4502963	.45117594
2 1 1	.64854781	.64854287
2 2 0	.33082849	.33120501
2 2 1	.65472273	.65506022
2 3 0	.28400721	.28169093
2 3 1	.61605961	.61442653
2 4 0	.22609365	.22538232
2 4 1	.6154092	.61499622

Logit vs. Probit



Fractional models and quasilikelihood (pseudolikelihood)

- Likelihood models assume we know the unobservable and all it's moments
- Quasilikelihood models are agnostic about anything but the first moment
- Fractional models use the likelihood of a probit or logit to model outcomes in $[0, 1]$. The unobservable of the probit and logit does not generate values in $(0, 1)$
- Stata has an implementation for fractional probit and fractional logit models

The model

$$E(Y|X) = F(X\beta)$$

- $F(.)$ is a known c.d.f
- No assumptions are made about the distribution of the unobservable

Two fractional model examples

```
. clear
. set obs 10000
number of observations (_N) was 0, now 10,000
. set seed 111
. generate e  = rnormal()
. generate x  = rchi2(5)-3
. generate xb = .5*(1 - x)
. generate yp = xb + e > 0
. generate yf = normal(xb + e)
```

- In both cases $E(Y|X) = \Phi(X\theta)$
- For `yp`, the probit, $\theta = \beta$
- For `yf`, $\theta = \frac{\beta}{\sqrt{1+\sigma^2}}$

Two fractional model examples

```
. clear
. set obs 10000
number of observations (_N) was 0, now 10,000
. set seed 111
. generate e  = rnormal()
. generate x  = rchi2(5)-3
. generate xb = .5*(1 - x)
. generate yp = xb + e > 0
. generate yf = normal(xb + e)
```

- In both cases $E(Y|X) = \Phi(X\theta)$
- For `yp`, the probit, $\theta = \beta$
- For `yf`, $\theta = \frac{\beta}{\sqrt{1+\sigma^2}}$

Two fractional model estimates

```
. quietly fracreg probit yp x
. estimates store probit
. quietly fracreg probit yf x
. estimates store frac
. estimates table probit frac, eq(1)
```

Variable	probit	frac
x	-.50037834	-.35759981
_cons	.48964237	.34998136

```
. display .5/sqrt(2)
.35355339
```

Fractional regression output

```
. fracreg probit ypr c.x1##c.x2 i.d1##i.d2 i.d1#c.x1
Iteration 0:   log pseudolikelihood = -7021.8384
Iteration 1:   log pseudolikelihood = -5515.9431
Iteration 2:   log pseudolikelihood = -5453.7326
Iteration 3:   log pseudolikelihood = -5453.1743
Iteration 4:   log pseudolikelihood = -5453.1739
Fractional probit regression                    Number of obs     =     10,000
                                                Wald chi2(16)     =    1969.26
                                                Prob > chi2       =     0.0000
Log pseudolikelihood = -5453.1739               Pseudo R2         =     0.2125
```

ypr	Coef.	Robust Std. Err.	z	P>|z|	[95% Conf.	Interval]
x1	-.3271742	.0421567	-7.76	0.000	-.4097998	-.2445486
x2	.3105438	.0232016	13.38	0.000	.2650696	.356018
c.x1#c.x2	.3178514	.0254263	12.50	0.000	.2680168	.3676859
d1						
1	-.2927285	.0577951	-5.06	0.000	-.4060049	-.1794521
2	-.6605838	.0593091	-11.14	0.000	-.7768275	-.54434
3	-.9137215	.0655808	-13.93	0.000	-1.042258	-.7851855
4	-1.276209	.0720675	-17.71	0.000	-1.417459	-1.134959
1.d2	.2822199	.057684	4.89	0.000	.1691613	.3952784
d1#d2						
1 1	.2547359	.0817911	3.11	0.002	.0944284	.4150435
2 1	.6621119	.0839477	7.89	0.000	.4975774	.8266464
3 1	.8471544	.0896528	9.45	0.000	.6714382	1.022871
4 1	1.260509	.0999594	12.61	0.000	1.064592	1.456425
d1#c.x1						
1	-.2747025	.041962	-6.55	0.000	-.3569466	-.1924585
2	-.5640486	.0447828	-12.60	0.000	-.6518212	-.4762759
3	-.9452172	.0514524	-18.37	0.000	-1.046062	-.8443723
4	-1.220618	.0615741	-19.82	0.000	-1.341301	-1.099935

Robust standard errors

- In general, this means we are agnostic about the $E(\varepsilon\varepsilon'|X)$, about the conditional variance
- The intuition from linear regression (heteroskedasticity) does not extend
- In nonlinear likelihood-based models like probit and logit this is not the case

Robust standard errors

- In general, this means we are agnostic about the $E(\varepsilon\varepsilon'|X)$, about the conditional variance
- The intuition from linear regression (heteroskedasticity) does not extend
- In nonlinear likelihood-based models like probit and logit this is not the case

Nonlinear likelihood models and heteroskedasticity

```
 . clear
. set seed 111
. set obs 10000
number of observations (_N) was 0, now 10,000
. generate x = rbeta(2,3)
. generate e1 = rnormal(0, x)
. generate e2 = rnormal(0, 1)
. generate y1 = .5 - .5*x + e1 >0
. generate y2 = .5 - .5*x + e2 >0
```

Nonlinear likelihood models and heteroskedasticity

```
. probit y1 x, nolog
Probit regression                               Number of obs     =     10,000
                                                LR chi2(1)        =    1409.02
                                                Prob > chi2       =     0.0000
Log likelihood = -4465.3713                     Pseudo R2         =     0.1363

------------------------------------------------------------------------------
          y1 |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
           x |   -2.86167   .0812023   -35.24   0.000    -3.020824   -2.702517
       _cons |   2.090816   .0415858    50.28   0.000     2.009309    2.172322
------------------------------------------------------------------------------

. probit y2 x, nolog
Probit regression                               Number of obs     =     10,000
                                                LR chi2(1)        =      62.36
                                                Prob > chi2       =     0.0000
Log likelihood = -6638.0701                     Pseudo R2         =     0.0047

------------------------------------------------------------------------------
          y2 |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
           x |  -.5019177   .0636248    -7.89   0.000    -.6266199   -.3772154
       _cons |   .4952327   .0290706    17.04   0.000     .4382554      .55221
------------------------------------------------------------------------------
```

Nonparametric regression

- Nonparametric regression is agnostic
- Unlike parametric estimation, nonparametric regression assumes no functional form for the relationship between outcomes and covariates.
- You do not need to know the functional form to answer important research questions
- You are not subject to problems that arise from misspecification

Nonparametric regression

- Nonparametric regression is agnostic
- Unlike parametric estimation, nonparametric regression assumes no functional form for the relationship between outcomes and covariates.
- You do not need to know the functional form to answer important research questions
- You are not subject to problems that arise from misspecification

Nonparametric regression

- Nonparametric regression is agnostic
- Unlike parametric estimation, nonparametric regression assumes no functional form for the relationship between outcomes and covariates.
- You do not need to know the functional form to answer important research questions
- You are not subject to problems that arise from misspecification

Mean Function

- Some parametric functional form assumptions.
  - regression: $E(Y|X) = X\beta$
  - probit: $E(Y|X) = \Phi(X\beta)$
  - Poisson: $E(Y|X) = \exp(X\beta)$
- The relationship of interest is also a conditional mean:

$$E(y|X) = g(X)$$

- Where the mean function $g(\cdot)$ is unknown

Mean Function

- Some parametric functional form assumptions.
  - regression: $E(Y|X) = X\beta$
  - probit: $E(Y|X) = \Phi(X\beta)$
  - Poisson: $E(Y|X) = \exp(X\beta)$
- The relationship of interest is also a conditional mean:

$$E(y|X) = g(X)$$

- Where the mean function $g(\cdot)$ is unknown

Mean Function

- Some parametric functional form assumptions.
  - regression: $E(Y|X) = X\beta$
  - probit: $E(Y|X) = \Phi(X\beta)$
  - Poisson: $E(Y|X) = \exp(X\beta)$
- The relationship of interest is also a conditional mean:

$$E(y|X) = g(X)$$

- Where the mean function $g(\cdot)$ is unknown

Mean Function

- Some parametric functional form assumptions.
  - regression: $E(Y|X) = X\beta$
  - probit: $E(Y|X) = \Phi(X\beta)$
  - Poisson: $E(Y|X) = \exp(X\beta)$
- The relationship of interest is also a conditional mean:

$$E(y|X) = g(X)$$

- Where the mean function $g(\cdot)$ is unknown

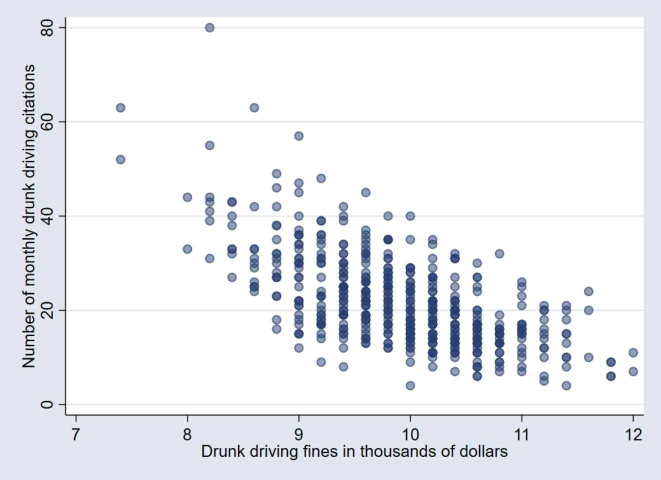
Traditional Approach to Nonparametric Estimation

- A cross section of counties
- `citations`: Number of monthly drunk driving citations
- `fines`: The value of fines imposed in a county in thousands of dollars if caught drinking and driving.

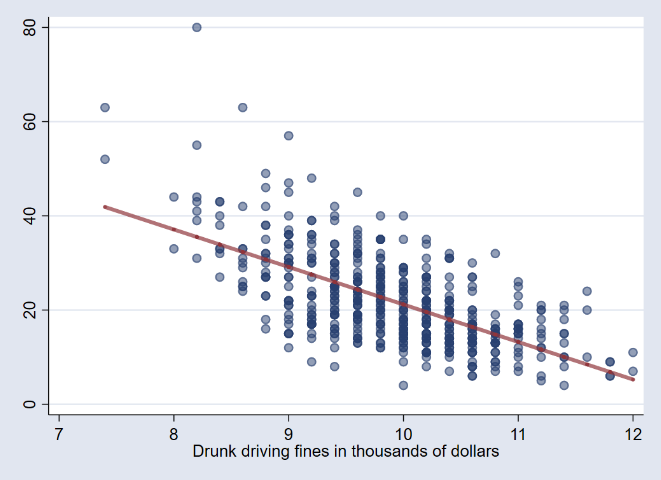
Traditional Approach to Nonparametric Estimation

- A cross section of counties
- `citations`: Number of monthly drunk driving citations
- `fines`: The value of fines imposed in a county in thousands of dollars if caught drinking and driving.

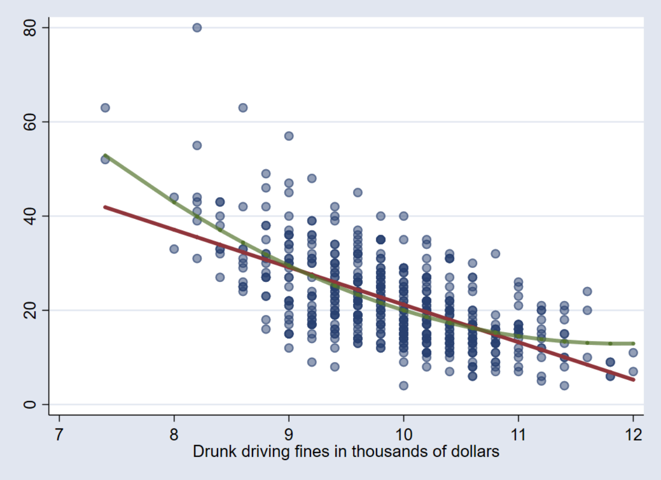
Implicit Relation



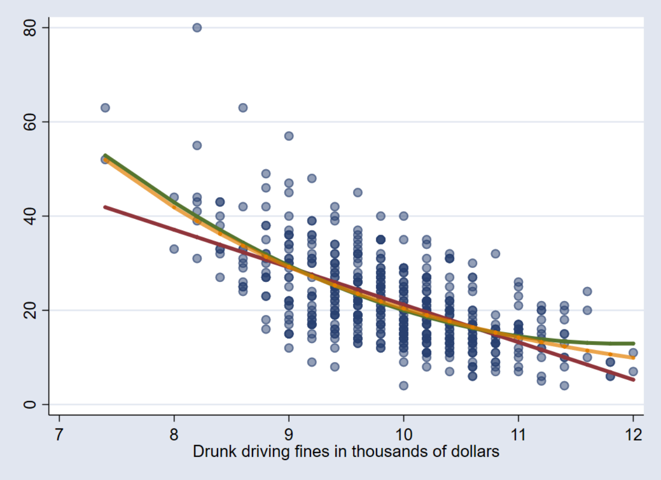
Simple linear regression



Regression with nonlinearities

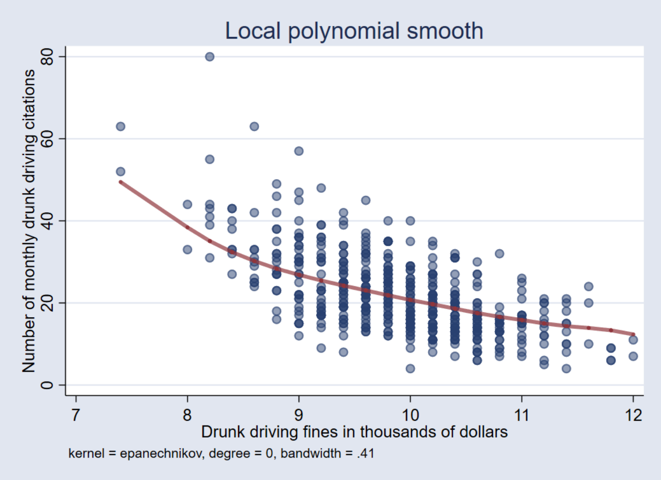


Poisson regression



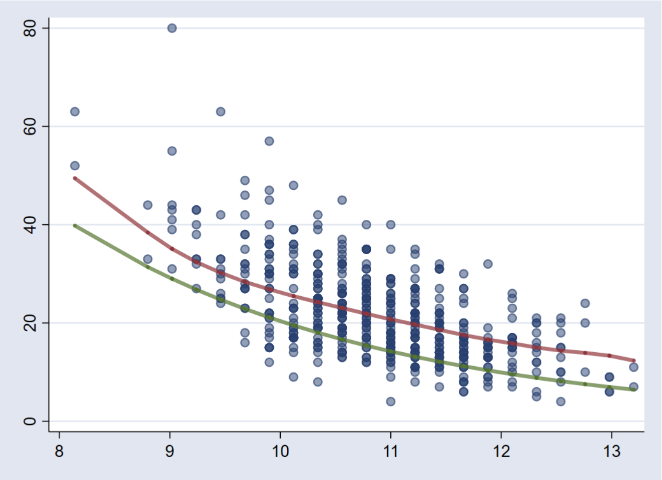
Nonparametric Estimation of Mean Function

```
.  lpoly citations fines
```

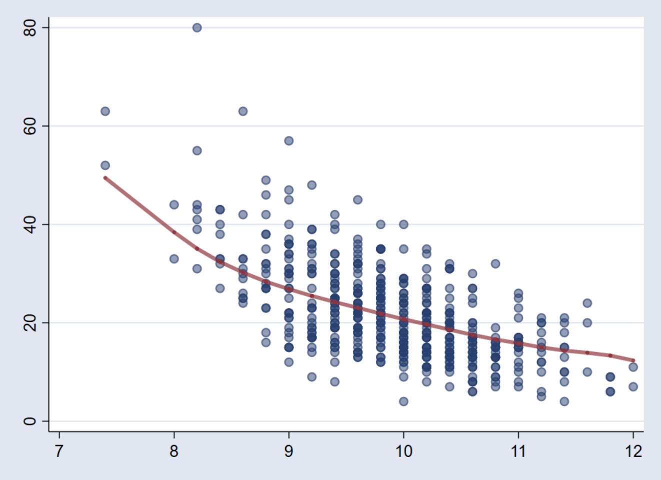


Now That We have the Mean Function

- What is the effect on the mean of `citations` of increasing `fines` by 10% ?



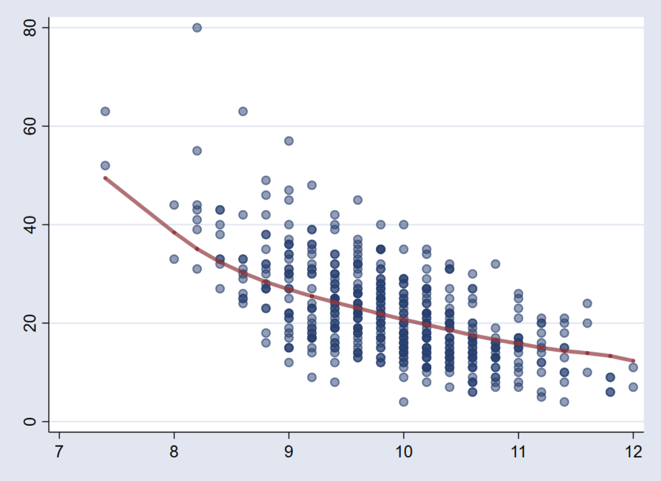
Traditional Approach Gives Us



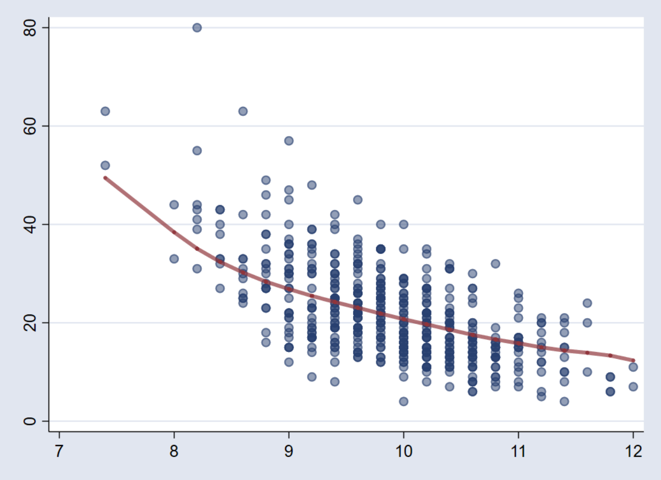
Additional Variables

- I would like to add controls
  - Whether county has a college town `college`
  - Number of highway patrol `patrols` units per capita in the county
- With those controls I can ask some new questions

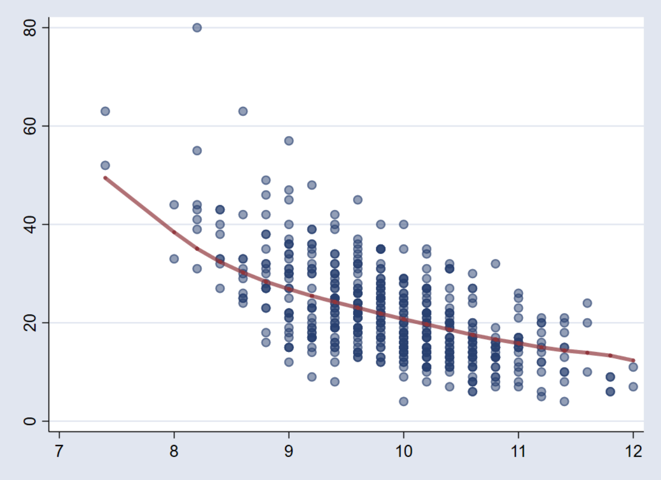
- What is the mean of `citations` if I increase `patrols` and `fines` ?



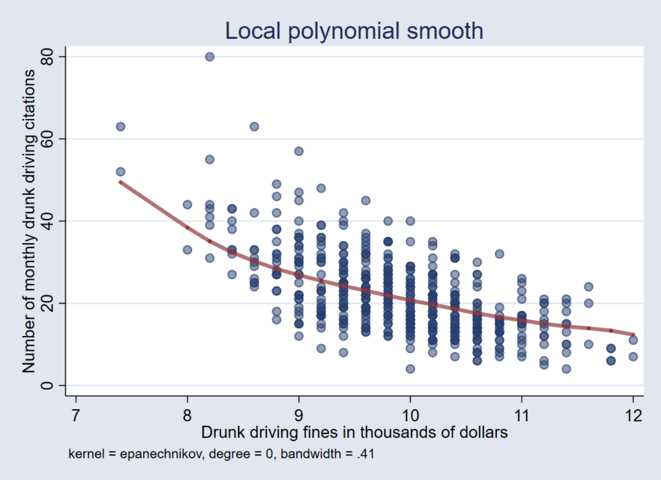
- How does the mean of `citations` differ for counties where there is a college town, averaging out the effect of `patrols` and `fines`?



- What policy has a bigger effect on the mean of `citations`, an increase in `fines`, an increase in `patrols`, or a combination of both?



What We Have Is



What We Have

- I have a mean function. That makes no functional form assumptions.
- I cannot answer the previous questions.
- My analysis was graphical not statistical
- My analysis is limited to one covariate
- This is true even if I give you the true mean function, $g(X)$

What We Have

- I have a mean function. That makes no functional form assumptions.
- I cannot answer the previous questions.
- My analysis was graphical not statistical
- My analysis is limited to one covariate
- This is true even if I give you the true mean function, $g(X)$

What We Have

- I have a mean function. That makes no functional form assumptions.
- I cannot answer the previous questions.
- My analysis was graphical not statistical
- My analysis is limited to one covariate
- This is true even if I give you the true mean function, $g(X)$

What We Have

- I have a mean function. That makes no functional form assumptions.
- I cannot answer the previous questions.
- My analysis was graphical not statistical
- My analysis is limited to one covariate
- This is true even if I give you the true mean function, $g(X)$

What We Have

- I have a mean function. That makes no functional form assumptions.
- I cannot answer the previous questions.
- My analysis was graphical not statistical
- My analysis is limited to one covariate
- This is true even if I give you the true mean function, $g(X)$

Nonparametric regression: discrete covariates

Mean function for a discrete covariate

- Mean (probability) of low birthweight (`lbweight`) conditional on smoking 1 to 5 cigarettes (`msmoke`=1) during pregnancy

```
. mean lbweight if msmoke==1
Mean estimation                    Number of obs   =       480

              |       Mean   Std. Err.     [95% Conf. Interval]
--------------+------------------------------------------------
     lbweight |      .1125   .0144375      .0841313    .1408687
```

- `regress lbweight 1.msmoke, noconstant`
- $E(lbweigth|msmoke = 1)$, nonparametric estimate

Nonparametric regression: discrete covariates

Mean function for a discrete covariate

- Mean (probability) of low birthweight (`lbweight`) conditional on smoking 1 to 5 cigarettes (`msmoke`=1) during pregnancy

```
. mean lbweight if msmoke==1
Mean estimation                    Number of obs   =        480
```

	Mean	Std. Err.	[95% Conf.	Interval]
lbweight	.1125	.0144375	.0841313	.1408687

- `regress lbweight 1.msmoke, noconstant`
- $E(lbweigth|msmoke = 1)$, nonparametric estimate

Nonparametric regression: discrete covariates

Mean function for a discrete covariate

- Mean (probability) of low birthweight (`lbweight`) conditional on smoking 1 to 5 cigarettes (`msmoke`=1) during pregnancy

```
. mean lbweight if msmoke==1
Mean estimation                    Number of obs   =        480

--------------------------------------------------------------
             |       Mean   Std. Err.     [95% Conf. Interval]
-------------+------------------------------------------------
    lbweight |      .1125   .0144375      .0841313    .1408687
--------------------------------------------------------------
```

- `regress lbweight 1.msmoke, noconstant`
- $E(lbweigth|msmoke = 1)$, nonparametric estimate

Nonparametric regression: continuous covariates

Conditional mean for a continuous covariate

- low birthweight conditional on log of family income `fincome`
- $E(lbweight|fincome = 10.819)$
- Take observations **near** the value of 10.819 and then take an average
- $|fincome_i - 10.819| \leq h$
- h is a small number referred to as the bandwidth

Nonparametric regression: continuous covariates

Conditional mean for a continuous covariate

- low birthweight conditional on log of family income `fincome`
- $E(lbweight|fincome = 10.819)$
- Take observations **near** the value of 10.819 and then take an average
- $|fincome_i - 10.819| \leq h$
- h is a small number referred to as the bandwidth

Nonparametric regression: continuous covariates

Conditional mean for a continuous covariate

- low birthweight conditional on log of family income `fincome`
- $E(lbweight|fincome = 10.819)$
- Take observations **near** the value of 10.819 and then take an average
- $|fincome_i - 10.819| \leq h$
- h is a small number referred to as the bandwidth

Nonparametric regression: continuous covariates

Conditional mean for a continuous covariate

- low birthweight conditional on log of family income `fincome`
- $E(lbweight|fincome = 10.819)$
- Take observations **near** the value of 10.819 and then take an average
- $|fincome_i - 10.819| \leq h$
- h is a small number referred to as the bandwidth

Nonparametric regression: continuous covariates

Conditional mean for a continuous covariate

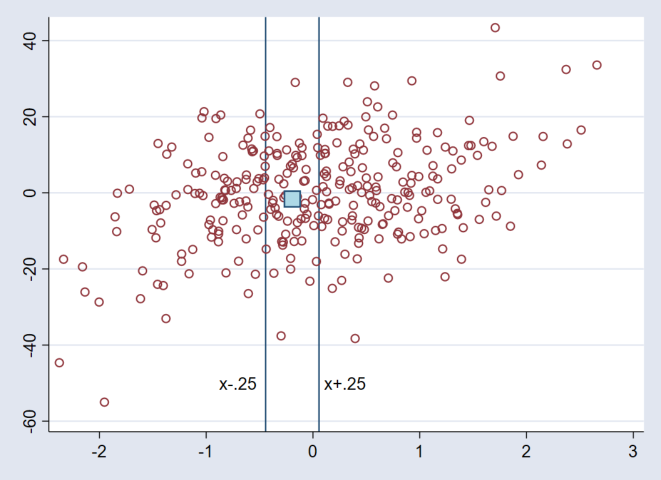
- low birthweight conditional on log of family income `fincome`
- $E(lbweight|fincome = 10.819)$
- Take observations **near** the value of 10.819 and then take an average
- $|fincome_i - 10.819| \leq h$
- h is a small number referred to as the bandwidth

Nonparametric regression: continuous covariates

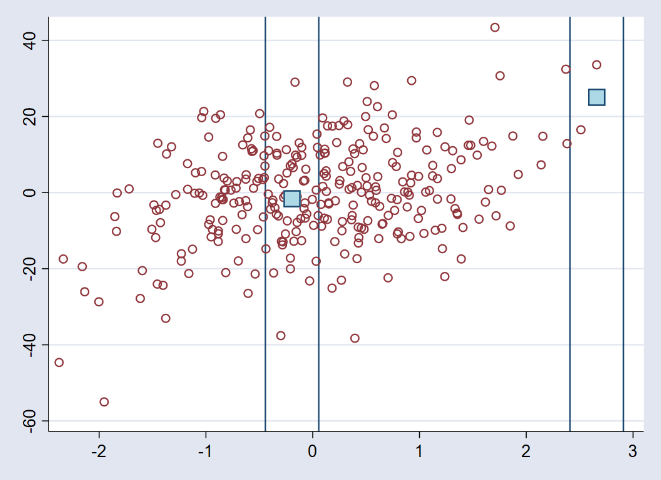
Conditional mean for a continuous covariate

- low birthweight conditional on log of family income `fincome`
- $E(lbweight|fincome = 10.819)$
- Take observations **near** the value of 10.819 and then take an average
- $|fincome_i - 10.819| \leq h$
- h is a small number referred to as the bandwidth

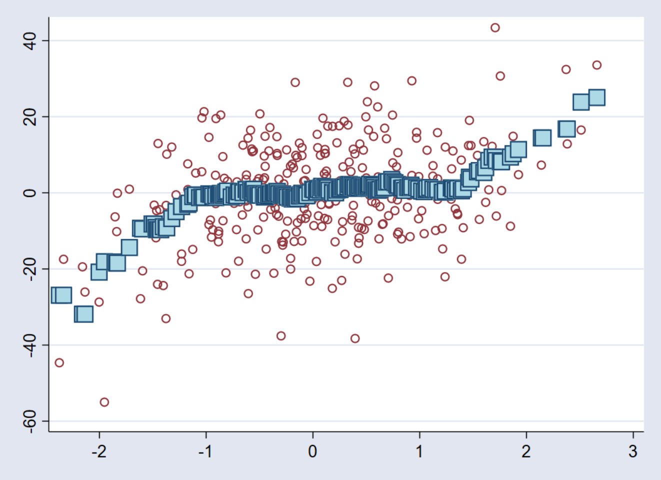
Graphical representation



Graphical example



Graphical example continued



Two concepts

1. h !!!!
2. Definition of distance between points, $|x_i - x| \leq h$

Kernel weights

- Epanechnikov
- Gaussian
- Epanechnikov2
- Rectangular(Uniform)
- Triangular
- Biweight
- Triweight
- Cosine
- Parzen

Kernel weights

- Epanechnikov
- Gaussian
- Epanechnikov2
- Rectangular(Uniform)
- Triangular
- Biweight
- Triweight
- Cosine
- Parzen

Discrete bandwidths

- Li–Racine Kernel

$$k(\cdot) = \begin{cases} 1 & \text{if} \quad x_i = x \\ h & \text{otherwise} \end{cases}$$

- Cell mean

$$k(\cdot) = \begin{cases} 1 & \text{if} \quad x_i = x \\ 0 & \text{otherwise} \end{cases}$$

- Cell mean was used in the example of discrete covariate estimate $E(lbweigth|msmoke = 1)$

Discrete bandwidths

- Li–Racine Kernel

$$k(\cdot) = \begin{cases} 1 & \text{if} \quad x_i = x \\ h & \text{otherwise} \end{cases}$$

- Cell mean

$$k(\cdot) = \begin{cases} 1 & \text{if} \quad x_i = x \\ 0 & \text{otherwise} \end{cases}$$

- Cell mean was used in the example of discrete covariate estimate $E(lbweigth|msmoke = 1)$

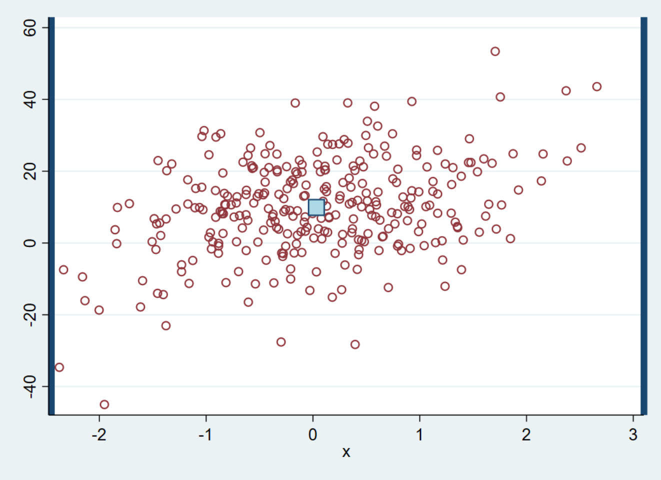
Selecting The Bandwidth

- A very large bandwidth will give you a biased estimate of the mean function with a small variance
- A very small bandwidth will give you an estimate with small bias and large variance

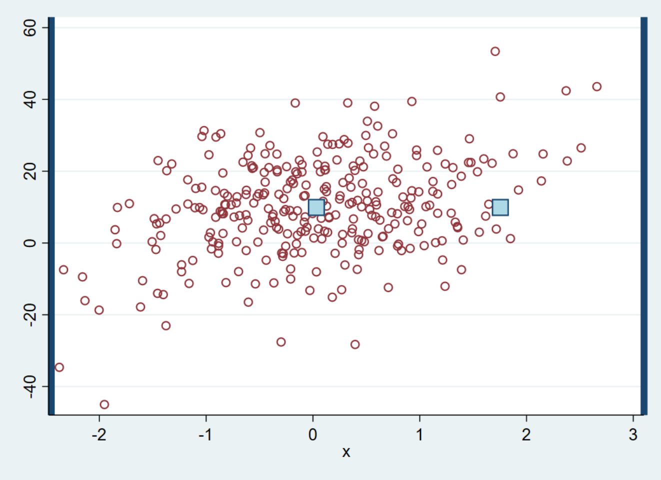
Selecting The Bandwidth

- A very large bandwidth will give you a biased estimate of the mean function with a small variance
- A very small bandwidth will give you an estimate with small bias and large variance

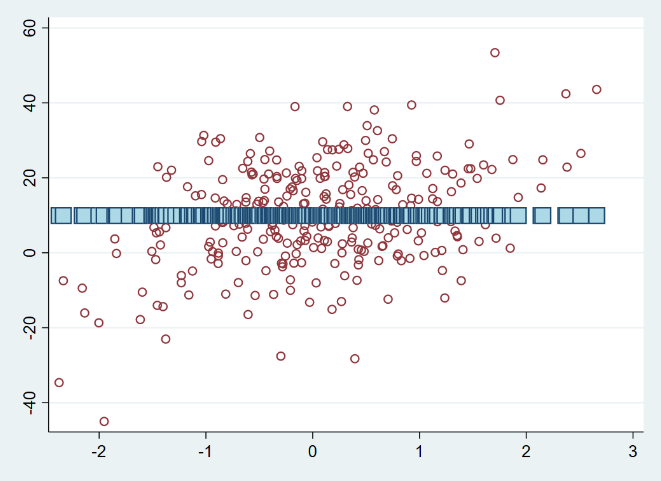
A Large Bandwidth At One Point



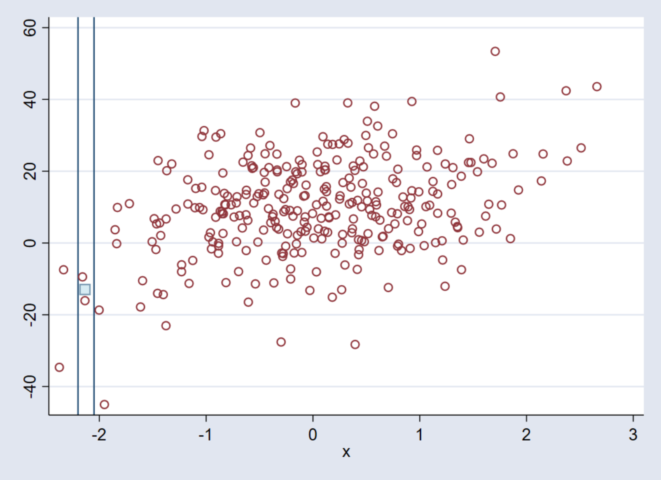
A Large Bandwidth At Two Points



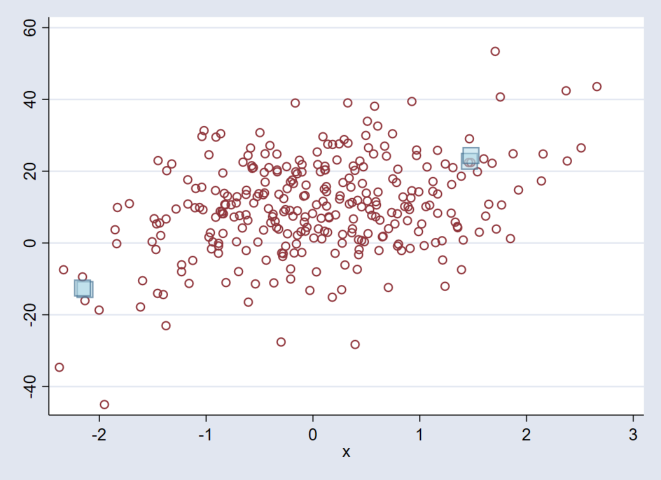
No Variance but Huge Bias



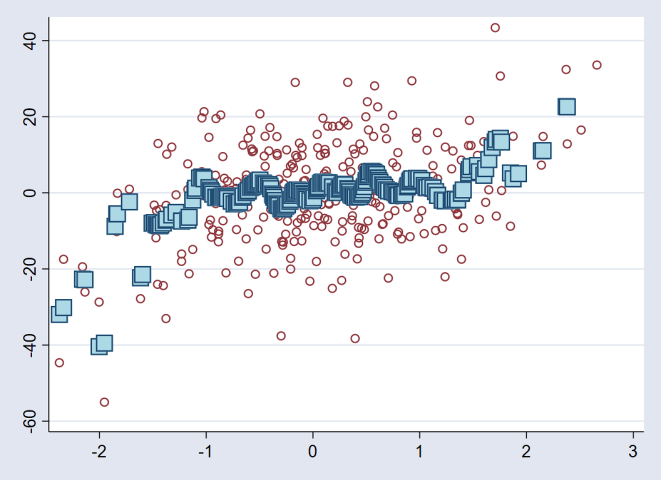
A Very Small Bandwidth at a Point



A Very Small Bandwidth at 4 Points



Small Bias Large Variance



Estimation

- Choose bandwidth optimally. Minimize bias–variance trade–off
  - Cross-validation (default)
  - Improved AIC (IMAIC)
- Compute a mean for every point in data (local-constant)
- Compute a regression for every point in data (local linear)
  - Computes constant (mean) and slope (effects)
  - Mean function and derivatives and effects of mean function
  - There is a bandwidth for the mean computation and another for the effects.
- Local-linear regression is the default

Estimation

- Choose bandwidth optimally. Minimize bias–variance trade–off
  - Cross-validation (default)
  - Improved AIC (IMAIC)
- Compute a mean for every point in data (local-constant)
- Compute a regression for every point in data (local linear)
  - Computes constant (mean) and slope (effects)
  - Mean function and derivatives and effects of mean function
  - There is a bandwidth for the mean computation and another for the effects.
- Local-linear regression is the default

Estimation

- Choose bandwidth optimally. Minimize bias–variance trade–off
  - Cross-validation (default)
  - Improved AIC (IMAIC)
- Compute a mean for every point in data (local-constant)
- Compute a regression for every point in data (local linear)
  - Computes constant (mean) and slope (effects)
  - Mean function and derivatives and effects of mean function
  - There is a bandwidth for the mean computation and another for the effects.
- Local-linear regression is the default

Estimation

- Choose bandwidth optimally. Minimize bias–variance trade–off
  - Cross-validation (default)
  - Improved AIC (IMAIC)
- Compute a mean for every point in data (local-constant)
- Compute a regression for every point in data (local linear)
  - Computes constant (mean) and slope (effects)
  - Mean function and derivatives and effects of mean function
  - There is a bandwidth for the mean computation and another for the effects.
- Local-linear regression is the default

Estimation

- Choose bandwidth optimally. Minimize bias–variance trade–off
  - Cross-validation (default)
  - Improved AIC (IMAIC)
- Compute a mean for every point in data (local-constant)
- Compute a regression for every point in data (local linear)
  - Computes constant (mean) and slope (effects)
  - Mean function and derivatives and effects of mean function
  - There is a bandwidth for the mean computation and another for the effects.
- Local-linear regression is the default

Estimation

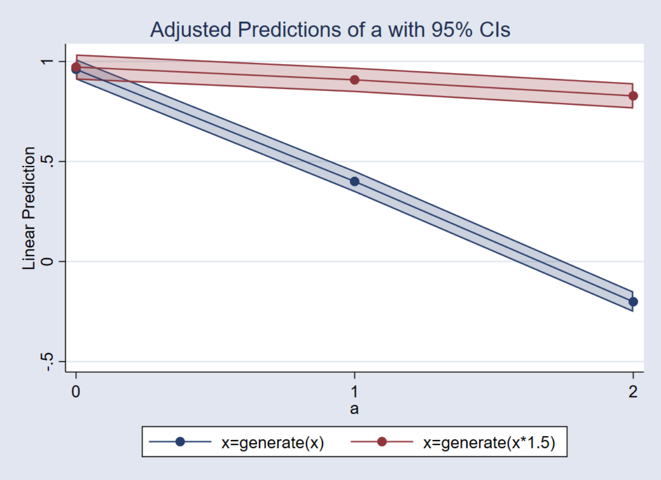
- Choose bandwidth optimally. Minimize bias–variance trade–off
  - Cross-validation (default)
  - Improved AIC (IMAIC)
- Compute a mean for every point in data (local-constant)
- Compute a regression for every point in data (local linear)
  - Computes constant (mean) and slope (effects)
  - Mean function and derivatives and effects of mean function
  - There is a bandwidth for the mean computation and another for the effects.
- Local-linear regression is the default

Simulated data example for continuous covariate

```
 . clear
. set obs 1000
number of observations (_N) was 0, now 1,000
. set seed 111
. generate x = (rchi2(5)-5)/10
. generate a = int(runiform()*3)
. generate e = rnormal(0, .5)
. generate y = 1 - x -a + 4*x^2*a + e
```

True model unknown to researchers

```
quietly regress y (c.x##c.x)##i.a margins a, ///
at(x=generate(x)) at(x=generate(x*1.5))
marginsplot, recastci(rarea) ciopts(fcolor(%30))
```



`npregress` Syntax

```
. npregress kernel y x i.a
```

- `kernel` refers to the kind of nonparametric estimation
- By default Stata assumes variables in my model are continuous
- `i.a` States the variable is categorical
- Interactions between continuous variables and between continuous and discrete variables are implicit

`npregress` Syntax

```
. npregress kernel y x i.a
```

- `kernel` refers to the kind of nonparametric estimation
- By default Stata assumes variables in my model are continuous
- `i.a` States the variable is categorical
- Interactions between continuous variables and between continuous and discrete variables are implicit

`npregress` Syntax

```
.  npregress kernel y x i.a
```

- `kernel` refers to the kind of nonparametric estimation
- By default Stata assumes variables in my model are continuous
- `i.a` States the variable is categorical
- Interactions between continuous variables and between continuous and discrete variables are implicit

`npregress` Syntax

```
. npregress kernel y x i.a
```

- `kernel` refers to the kind of nonparametric estimation
- By default Stata assumes variables in my model are continuous
- `i.a` States the variable is categorical
- Interactions between continuous variables and between continuous and discrete variables are implicit

`npregress` Syntax

```
. npregress kernel y x i.a
```

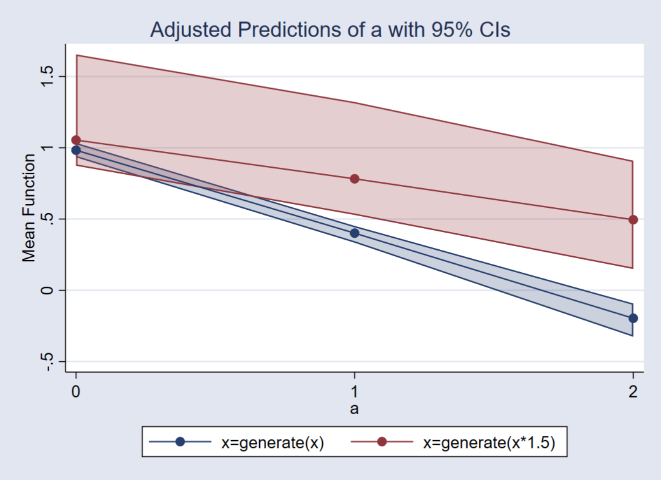
- `kernel` refers to the kind of nonparametric estimation
- By default Stata assumes variables in my model are continuous
- `i.a` States the variable is categorical
- Interactions between continuous variables and between continuous and discrete variables are implicit

Fitting the model with `npregress`

```
. npregress kernel y x i.a, nolog
Bandwidth
─────────────────────────────────────
             │      Mean     Effect
─────────────┼───────────────────────
           x │  .0616294   .0891705
           a │   .490625    .490625
─────────────────────────────────────

Local-linear regression                    Number of obs      =      1,000
Continuous kernel : epanechnikov           E(Kernel obs)      =         62
Discrete kernel   : liracine               R-squared          =     0.8409
Bandwidth         : cross validation
──────────────────────────────────────────────────────────────────────────
           y │   Estimate
─────────────┼────────────────────────────────────────────────────────────
Mean         │
           y │   .4071379
─────────────┼────────────────────────────────────────────────────────────
Effect       │
           x │  -.8212713
             │
           a │
   (1 vs 0)  │  -.5820049
   (2 vs 0)  │  -1.179375
──────────────────────────────────────────────────────────────────────────
Note: Effect estimates are averages of derivatives for continuous covariates and averages of
      contrasts for factor covariates.
Note: You may compute standard errors using vce(bootstrap) or reps().
```

The same effect

```
quietly regress y (c.x##c.x)##i.a margins a, ///
at(x=generate(x)) at(x=generate(x*1.5))
marginsplot, recastci(rarea) ciopts(fcolor(%30))
```



Longitudinal/Panel Data

- Under large N and fixed asymptotics behaves like cross-sectional models
- The difficulties arise because of time-invariant unobservables, i.e. α_i in

$$y_{it} = G(X_{it}\beta + \alpha_i + \varepsilon_{it})$$

- Our framework still works but we need to be careful with what it means to average over the sample.

Averaging

- Our model gives us:

$$E\left(y_{it}|X_{it},\alpha_i\right) = g\left(X_{it}\beta + \alpha_i\right)$$

- We cannot consistently estimate α_i so we integrate it out

$$\begin{aligned} E_\alpha E\left(y_{it}|X_{it},\alpha_i\right) &= E_\alpha g\left(X_{it}\beta + \alpha_i\right) \\ E_\alpha E\left(y_{it}|X_{it},\alpha_i\right) &= h\left(X_{it}\theta\right) \end{aligned}$$

- Sometimes we know the functional form $h(.)$. Sometimes we do not.

Averaging

- Our model gives us:

$$E\left(y_{it}|X_{it},\alpha_i\right)=g\left(X_{it}\beta+\alpha_i\right)$$

- We cannot consistently estimate α_i so we integrate it out

$$\begin{aligned} E_\alpha E\left(y_{it}|X_{it},\alpha_i\right) &= E_\alpha g\left(X_{it}\beta+\alpha_i\right) \\ E_\alpha E\left(y_{it}|X_{it},\alpha_i\right) &= h\left(X_{it}\theta\right) \end{aligned}$$

- Sometimes we know the functional form $h(.)$. Sometimes we do not.

Averaging

- Our model gives us:

$$E\left(y_{it}|X_{it},\alpha_i\right) = g\left(X_{it}\beta+\alpha_i\right)$$

- We cannot consistently estimate α_i so we integrate it out

$$\begin{aligned} E_\alpha E\left(y_{it}|X_{it},\alpha_i\right) &= E_\alpha g\left(X_{it}\beta+\alpha_i\right) \\ E_\alpha E\left(y_{it}|X_{it},\alpha_i\right) &= h\left(X_{it}\theta\right) \end{aligned}$$

- Sometimes we know the functional form $h(.)$. Sometimes we do not.

Averaging

- Our model gives us:

$$E\left(y_{it}|X_{it},\alpha_i\right)=g\left(X_{it}\beta+\alpha_i\right)$$

- We cannot consistently estimate α_i so we integrate it out

$$\begin{aligned}E_{\alpha}E\left(y_{it}|X_{it},\alpha_i\right) &= E_{\alpha}g\left(X_{it}\beta+\alpha_i\right)\\ E_{\alpha}E\left(y_{it}|X_{it},\alpha_i\right) &= h\left(X_{it}\theta\right)\end{aligned}$$

- Sometimes we know the functional form $h(.)$. Sometimes we do not.

A probit example

```
 . clear
. set seed 111
. set obs 5000
number of observations (_N) was 0, now 5,000
. generate id = _n
. generate a  = rnormal()
. expand 10
(45,000 observations created)
. bysort id: generate year = _n
. generate x    = (rchi2(5)-5)/10
. generate b    = int(runiform()*3)
. generate e    = rnormal()
. generate xb   = .5*(-1-x + b - x*b) + a
. generate dydx = normalden(.5*(-1-x + b - x*b)/(sqrt(2)))*((-.5-.5*b)/sqrt(2))
. generate y  = xb + e > 0
```

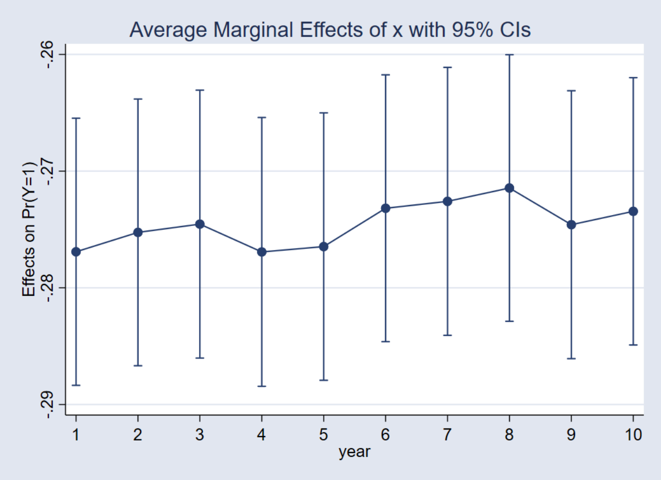
Panel data estimation

```
. xtset id year
       panel variable:  id (strongly balanced)
        time variable:  year, 1 to 10
                delta:  1 unit
. xtprobit y c.x##i.b, nolog
Random-effects probit regression                Number of obs     =     50,000
Group variable: id                              Number of groups  =      5,000
Random effects u_i ~ Gaussian                   Obs per group:
                                                              min =         10
                                                              avg =       10.0
                                                              max =         10
Integration method: mvaghermite                 Integration pts.  =         12
                                                Wald chi2(5)      =    5035.63
Log likelihood  = -27522.587                    Prob > chi2       =     0.0000
------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
           x |  -.5212161   .0393606   -13.24   0.000    -.5983614   -.4440708
             |
           b |
          1  |   .4859038   .0170101    28.57   0.000     .4525647     .519243
          2  |    1.00774   .0179167    56.25   0.000     .9726241    1.042856
             |
       b#c.x |
          1  |  -.5454211   .0557341    -9.79   0.000    -.6546579   -.4361843
          2  |  -1.059613   .0568466   -18.64   0.000     -1.17103   -.9481958
             |
       _cons |   -.506777   .0187516   -27.03   0.000    -.5435294   -.4700246
-------------+----------------------------------------------------------------
    /lnsig2u |   .0004287   .0298177                      -.058013    .0588704
-------------+----------------------------------------------------------------
     sigma_u |   1.000214   .0149121                      .9714102    1.029873
         rho |   .5001072   .0074544                      .4855008    .5147133
------------------------------------------------------------------------------
LR test of rho=0: chibar2(01) = 9819.64                Prob >= chibar2 = 0.000
```

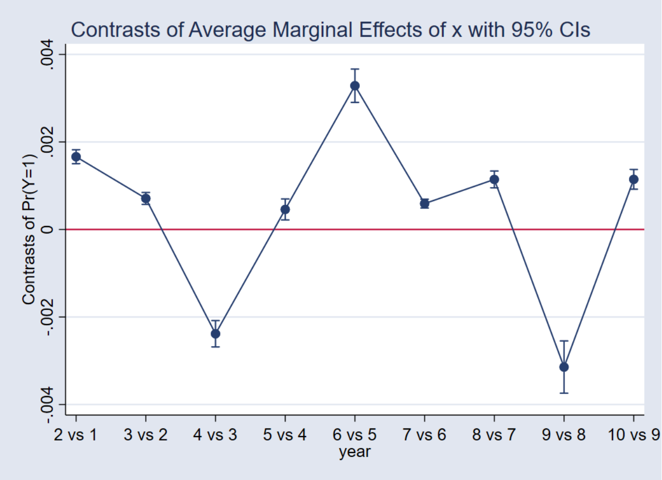
Effect estimation

```
. margins, dydx(x) over(year)
Average marginal effects                        Number of obs     =     50,000
Model VCE     : OIM
Expression    : Pr(y=1), predict(pr)
dy/dx w.r.t. : x
over          : year

------------------------------------------------------------------------------
             |            Delta-method
             |      dy/dx   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
x            |
        year |
          1  |  -.2769118   .0058397   -47.42   0.000    -.2883573   -.2654662
          2  |  -.2752501   .0058296   -47.22   0.000    -.2866759   -.2638242
          3  |  -.2745409    .005857   -46.87   0.000    -.2860204   -.2630613
          4  |  -.2769241   .0058773   -47.12   0.000    -.2884433   -.2654049
          5  |  -.2764666   .0058452   -47.30   0.000     -.287923   -.2650102
          6  |  -.2731819    .005833   -46.83   0.000    -.2846145   -.2617493
          7  |  -.2725905   .0058577   -46.54   0.000    -.2840714   -.2611096
          8  |   -.271447   .0058275   -46.58   0.000    -.2828686   -.2600253
          9  |  -.2745909   .0058566   -46.89   0.000    -.2860697   -.2631122
         10  |  -.2734455   .0058435   -46.79   0.000    -.2848985   -.2619924
------------------------------------------------------------------------------

. summarize dydx
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
        dydx |     50,000   -.2609633    .1032875  -.4231422  -.0394023
```

Effect estimation



Effect estimation



Beware of `pu0` or any $\alpha_i = 0$

- The coefficients of population averaged models are useful to compute ATE:

$$
\begin{aligned}
ATE &= E\left[F\left(X_{it}\delta + \delta_{treat} + \alpha_i\right) - F\left(X_{it}\delta + \alpha_i\right)\right] \\
&= E_x\left[E_\alpha\left[F\left(X_{it}\delta + \delta_{treat} + \alpha_i\right)\right]\right] - E_x\left[E_\alpha\left[F\left(X_{it}\delta + \alpha_i\right)\right]\right]
\end{aligned}
$$

- When we use $\alpha_i = 0$ we get it wrong
- The reason is that $E(g(x)) \neq g(E(x))$ when g is not a linear function:

$$
\begin{aligned}
E_x\left[F\left(X_{it}\delta + \delta_{treat} + 0\right)\right] - E_x\left[F\left(X_{it}\delta + 0\right)\right] &= \\
E_x\left[F\left(X_{it}\delta + \delta_{treat} + E\left(\alpha_i\right)\right)\right] - E_x\left[F\left(X_{it}\delta + E\left(\alpha_i\right)\right)\right] &\neq \\
E_x\left[E_\alpha\left[F\left(X_{it}\delta + \delta_{treat} + \alpha_i\right)\right]\right] - E_x\left[E_\alpha\left[F\left(X_{it}\delta + \alpha_i\right)\right]\right] &= ATE
\end{aligned}
$$

Beware of `pu0` or any $\alpha_i = 0$

- The coefficients of population averaged models are useful to compute ATE:

$$\begin{aligned} ATE &= E\left[F\left(X_{it}\delta + \delta_{treat} + \alpha_i\right) - F\left(X_{it}\delta + \alpha_i\right)\right] \\ &= E_x\left[E_\alpha\left[F\left(X_{it}\delta + \delta_{treat} + \alpha_i\right)\right]\right] - E_x\left[E_\alpha\left[F\left(X_{it}\delta + \alpha_i\right)\right]\right] \end{aligned}$$

- When we use $\alpha_i = 0$ we get it wrong
- The reason is that $E(g(x)) \neq g(E(x))$ when g is not a linear function:

$$\begin{aligned} E_x\left[F\left(X_{it}\delta + \delta_{treat} + 0\right)\right] - E_x\left[F\left(X_{it}\delta + 0\right)\right] &= \\ E_x\left[F\left(X_{it}\delta + \delta_{treat} + E\left(\alpha_i\right)\right)\right] - E_x\left[F\left(X_{it}\delta + E\left(\alpha_i\right)\right)\right] &\neq \\ E_x\left[E_\alpha\left[F\left(X_{it}\delta + \delta_{treat} + \alpha_i\right)\right]\right] - E_x\left[E_\alpha\left[F\left(X_{it}\delta + \alpha_i\right)\right]\right] &= ATE \end{aligned}$$

Beware of `pu0` or any $\alpha_i = 0$

- The coefficients of population averaged models are useful to compute ATE:

$$\begin{aligned} ATE &= E\left[F\left(X_{it}\delta + \delta_{treat} + \alpha_i\right) - F\left(X_{it}\delta + \alpha_i\right)\right] \\ &= E_x\left[E_\alpha\left[F\left(X_{it}\delta + \delta_{treat} + \alpha_i\right)\right]\right] - E_x\left[E_\alpha\left[F\left(X_{it}\delta + \alpha_i\right)\right]\right] \end{aligned}$$

- When we use $\alpha_i = 0$ we get it wrong
- The reason is that $E(g(x)) \neq g(E(x))$ when g is not a linear function:

$$\begin{aligned} E_x\left[F\left(X_{it}\delta + \delta_{treat} + 0\right)\right] - E_x\left[F\left(X_{it}\delta + 0\right)\right] &= \\ E_x\left[F\left(X_{it}\delta + \delta_{treat} + E\left(\alpha_i\right)\right)\right] - E_x\left[F\left(X_{it}\delta + E\left(\alpha_i\right)\right)\right] &\neq \\ E_x\left[E_\alpha\left[F\left(X_{it}\delta + \delta_{treat} + \alpha_i\right)\right]\right] - E_x\left[E_\alpha\left[F\left(X_{it}\delta + \alpha_i\right)\right]\right] &= ATE \end{aligned}$$

Concluding Remarks

- Our work is not done after we get the parameters of our model
- After we get the parameters is when our work starts. We can ask interesting questions
- The questions we ask can be placed in a general framework:
  - Define an object of interest $E(y|X)$ or $E(y|X, \alpha)$
  - Explore the multidemensional function
- Use `margins` and `marginsplot`