

# Métodos no paramétricos en Stata

Eduardo García Echeverri

Webinar de Stata, 2023

# Agenda

- 1 El atractivo de los métodos no paramétricos
- 2 Estimando densidades de probabilidad
- 3 Estimando esperanzas condicionales / funciones de regresión
  - Regresión kernel
  - Aproximación con series
- 4 Ventajas y desventajas de los métodos no paramétricos

## ¿Qué es un método no paramétrico?

Método de estimación que no presupone:

- **Densidad de probabilidad** para el outcome y los regresores.
- **Forma funcional** relacionando el outcome y los regresores.

Un **método paramétrico**, requiere de estos dos supuestos.

## Ejemplo

¿Cuál es el efecto de **fumar durante el embarazo** (msmoke) en el **peso del bebé** al nacer (bweight)?

- **Paramétrico:**

$$\text{bweight} = \beta_0 + \beta_1 \text{msmoke} + \gamma \text{Controles} + \varepsilon$$

$$\varepsilon \sim N(0, \sigma^2)$$

- **No Paramétrico:**

$$\text{bweight} = g(\text{msmoke}, \text{Controles}) + \varepsilon$$

$$\mathbb{E}[\varepsilon | \text{msmoke}, \text{Controles}] = 0$$

# El Atractivo de los Métodos No Paramétricos

## Ventajas:

1. Evitan problemas causados por **mala especificación**.
2. Mejoran predicciones.
3. Conocer la forma funcional no es necesario para responder nuestras preguntas de investigación.
4. Fáciles de implementar en Stata.

## Desventajas:

1. Intensivos en datos (esp. con muchos regresores)
2. Computacionalmente costosos.

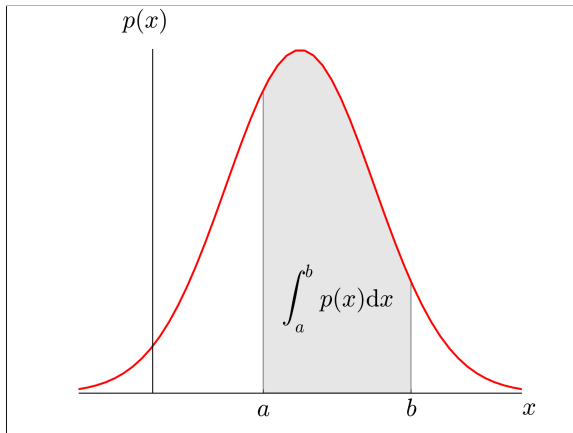
- 1 El atractivo de los métodos no paramétricos
- 2 Estimando densidades de probabilidad
- 3 Estimando esperanzas condicionales / funciones de regresión
  - Regresión kernel
  - Aproximación con series
- 4 Ventajas y desventajas de los métodos no paramétricos

# Agenda

- 1 El atractivo de los métodos no paramétricos
- 2 Estimando densidades de probabilidad
- 3 Estimando esperanzas condicionales / funciones de regresión
  - Regresión kernel
  - Aproximación con series
- 4 Ventajas y desventajas de los métodos no paramétricos

# ¿Qué es una densidad de probabilidad?

Nos muestra la **distribución de una variable aleatoria**.





## ¿Por qué son útiles?

La **densidad de probabilidad** nos muestra:

1. Qué valores son **probables/improbables**.
2. La **moda(s)** de la distribución (**picos** de la densidad)
3. El **rango** de valores que toma la variable
4. Probabilidad de **eventos extremos** (colas de la distribución)

# Estimación paramétrica de densidades

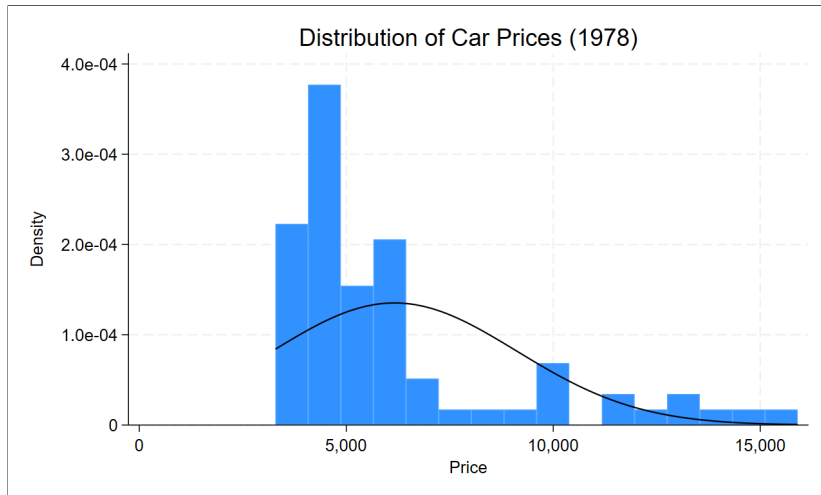
Primero, se especifica un tipo de densidad de probabilidad:

1. **Normal** (más común)
2. Log-normal
3. ...

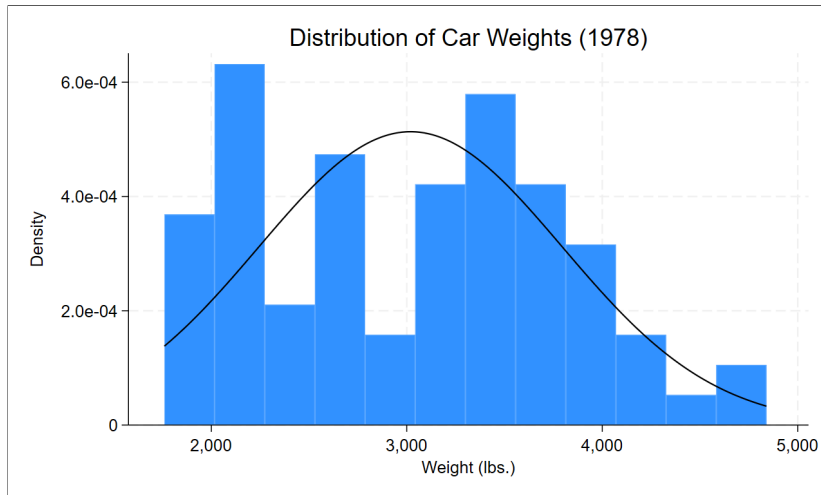
Segundo, basados en la **media y varianza muestral**, se **estima** la densidad que **mejor encaje en los datos**.

**Problema:** mala especificación. La normal puede ser una mala aproximación.

# Normalidad puede no cumplirse – larga cola derecha



# Normalidad puede no cumplirse – múltiples modas



# Estimadores Kernel para densidades

Estos estimadores **no presuponen un tipo de distribución**.

Para estimar la **densidad** en el punto  $x$ :

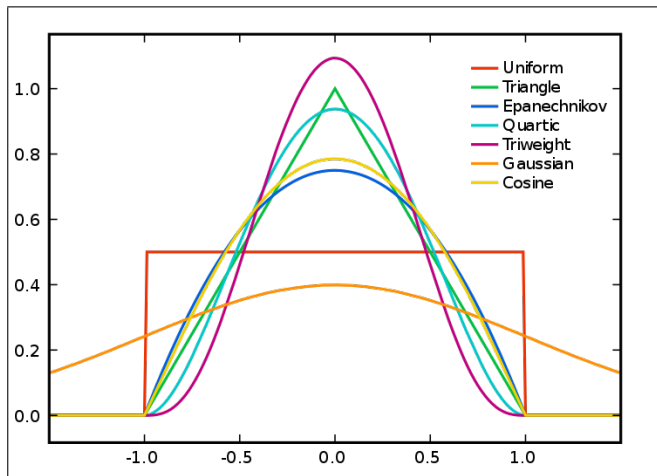
- **“Contamos”** cuántas observaciones son cercanas a  $x$ ,

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - X_i}{h}\right)$$

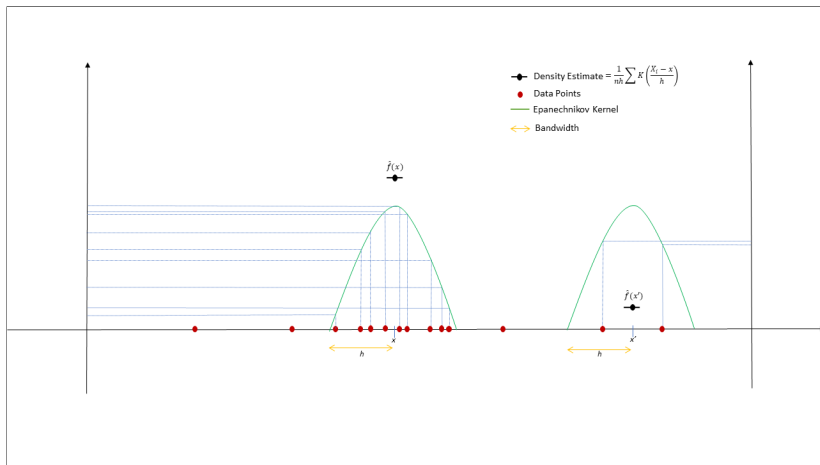
$h$ : **Bandwidth** – número positivo (muy importante)

$K(\cdot)$ : **Kernel** – una función ( $K : \mathbb{R} \rightarrow \mathbb{R}$ )

# Tipos de kernel



# Ilustración



# Implementación en Stata

```
kdensity varname [if] [in] [weight] [, options]
```

## Quick Start:

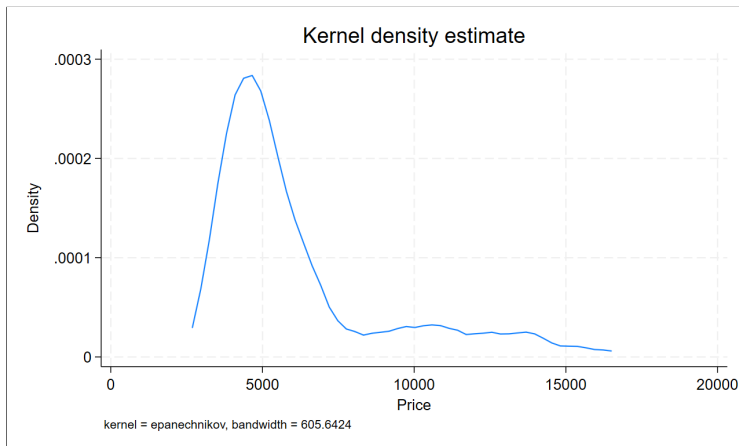
Graficar la densidad de la variable  $x_1$  usando **estimadores kernel**

- `kdensity x1`



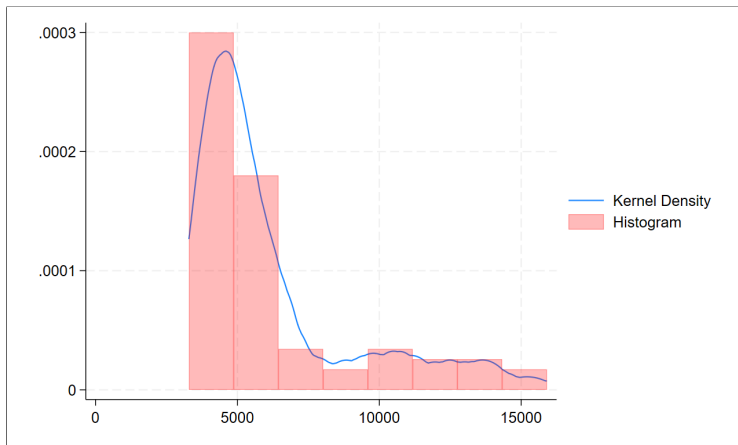
## Ejemplo 1: Graficando una densidad de probabilidad

- `sysuse auto, clear`
- `kdensity price`



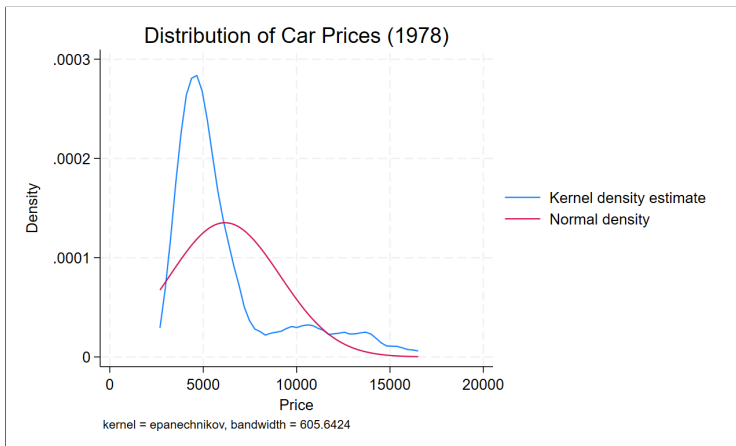
## Ejemplo 1: Un mejor ajuste a los datos de precios

- `twoway kdensity price || histogram price, legend(order(1 "Kernel Density" 2 "Histogram")) color(red%30)`



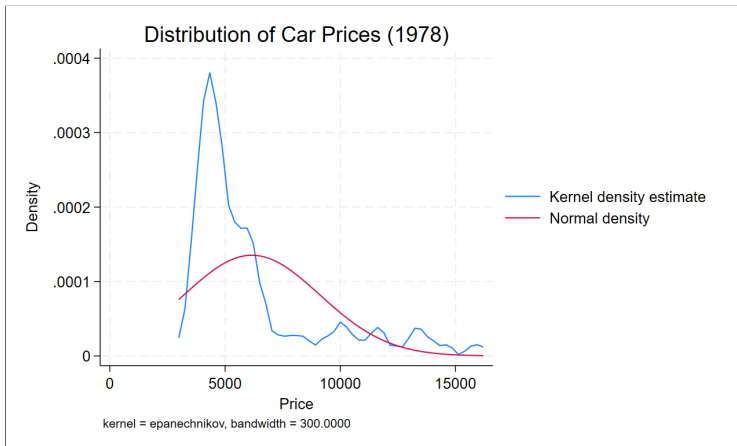
## Ejemplo 2: Comparando con la distribución normal

- `kdensity price, normal title("Distribution of Car Prices")`



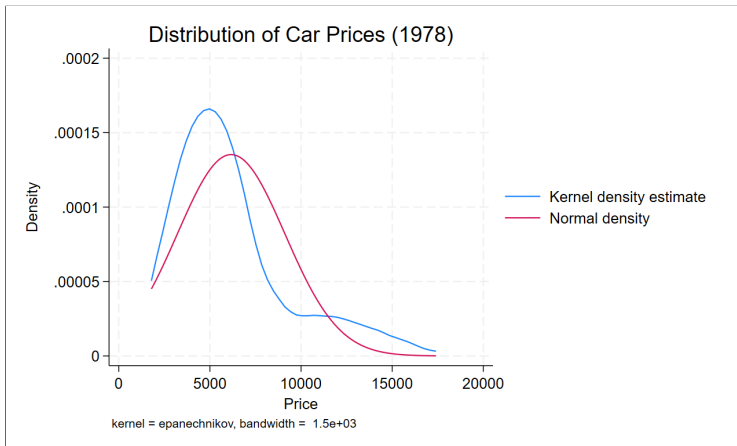
## Ejemplo 3: Un bandwidth más pequeño (overfitting)

- `kdensity price, normal bwidth(300) title("Distribution of Car Weights")`



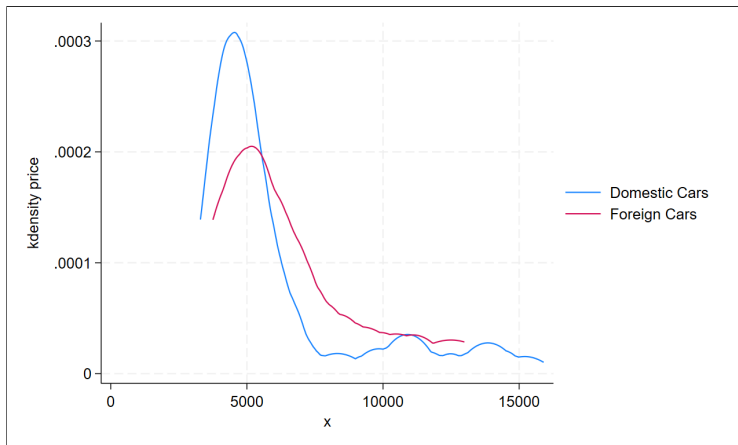
## Ejemplo 4: Un bandwidth más grande (oversmoothing)

- `kdensity price, normal bwidth(1500) title("Distribution of Car Prices")`



## Ejemplo 5: Comparando distribuciones de precios

- `twoway kdensity price if foreign==0 || kdensity price if foreign==1`



## Opciones útiles

`kernel`: Especificar la función Kernel:

- `epanechnikov`; por defecto
- `gaussian`
- `triangle...`

`generate(newvar1, newvar2)`: guardar los valores estimados en `newvar1` y sus respectivas densidades en `newvar2`

`nograph`: Suprimir la gráfica.

# Opción generate

- `kdensity price, generate(pricepoint density) nograph`
- browse pricepoint density

The screenshot shows the Stata Data Editor window titled "2 - Data Editor (Browse) - [auto]". The main window displays a dataset with two columns: `density` and `pricepoint`. The `density` column contains values generated by the `kdensity` command, and the `pricepoint` column contains the corresponding price values. The right-hand pane shows the "Variables" list, where `make` is selected, and its properties are displayed.

| density   | pricepoint |
|-----------|------------|
| 0.0002919 | 2685.3578  |
| 0.0000687 | 2967.5267  |
| 0.0011843 | 3249.6958  |
| 0.0017546 | 3531.8648  |
| 0.0022515 | 3814.0339  |
| 0.0028398 | 4096.203   |
| 0.0034082 | 4378.3721  |
| 0.0039363 | 4660.5412  |
| 0.0045799 | 4942.7102  |
| 0.0052435 | 5224.8793  |
| 0.0059179 | 5507.0484  |
| 0.0066067 | 5789.2175  |
| 0.0073028 | 6071.3865  |
| 0.0080114 | 6353.5556  |
| 0.0087391 | 6635.7247  |
| 0.0094827 | 6917.8938  |
| 0.0102462 | 7200.0628  |
| 0.0110251 | 7482.2319  |
| 0.0118143 | 7764.401   |
| 0.0126115 | 8046.5701  |
| 0.0134222 | 8328.7392  |
| 0.0142438 | 8610.9082  |
| 0.0150715 | 8893.0773  |
| 0.0159115 | 9175.2464  |
| 0.0167615 | 9457.4155  |
| 0.0176215 | 9739.5845  |
| 0.0184915 | 10021.754  |
| 0.0193715 | 10303.923  |
| 0.0202615 | 10586.092  |
| 0.0211615 | 10868.261  |
| 0.0220715 | 11150.43   |
| 0.0229915 | 11432.599  |
| 0.0239215 | 11714.768  |

The right-hand pane shows the "Variables" list, where `make` is selected. The properties for `make` are displayed:

| Name              | Label          | Type  | Format | Value l |
|-------------------|----------------|-------|--------|---------|
| <code>make</code> | Make and model | str18 | %-18s  |         |

The bottom status bar shows: "Vars: 2 of 14. Order: Dataset Obs: 74 Filter: Off Mode: Browse CAP: NUM".



# Agenda

- 1 El atractivo de los métodos no paramétricos
- 2 Estimando densidades de probabilidad
- 3 Estimando esperanzas condicionales / funciones de regresión
  - Regresión kernel
  - Aproximación con series
- 4 Ventajas y desventajas de los métodos no paramétricos

# Esperanza condicional – Función de regresión

## Objetivo:

Estimar la **esperanza condicional**/función de regresión  $g(\cdot)$ :

$$y = g(X) + \varepsilon$$

$$\mathbb{E}[\varepsilon|X] = 0$$

## Enfoques paramétricos:

- Regresión lineal:  $g(X) = x\beta$
- Probit:  $g(X) = \Phi(x\beta)$
- Poisson:  $g(X) = \exp(x\beta)$

## Problemas por mala especificación.

**Regresión kernel** y **aproximación con series** no presuponen una forma funcional para  $g(\cdot)$

# Agenda

- 1 El atractivo de los métodos no paramétricos
- 2 Estimando densidades de probabilidad
- 3 Estimando esperanzas condicionales / funciones de regresión
  - Regresión kernel
  - Aproximación con series
- 4 Ventajas y desventajas de los métodos no paramétricos

## Regresión kernel – Local linear

Para estimar la **función de regresión** en el punto  $x$ :

- “**Regresión ponderada**” de  $y$  en  $X$  usando obs. cerca a  $x$ ,

$$\min_{\gamma_0, \gamma_1} \sum_{i=1}^n \left( y_i - \gamma_0 - \gamma_1(x_i - x) \right)^2 \cdot K\left(\frac{x - X_i}{h}\right)$$

$\gamma_0$  : **Función de regresión** estimada  $\hat{g}(x)$

$\gamma_1$ : **Derivada** de  $g(\cdot)$  estimada en el punto  $x$ .

$h$ : **Bandwidth** – número positivo (muy importante!)

$K(\cdot)$ : **Kernel** – una función ( $K : \mathbb{R} \rightarrow \mathbb{R}$ )

## Regresión kernel – Local constant

Para estimar la **función de regresión** en el punto  $x$ :

- “**Media ponderada**” de  $y$  usando puntos cerca de  $x$ ,

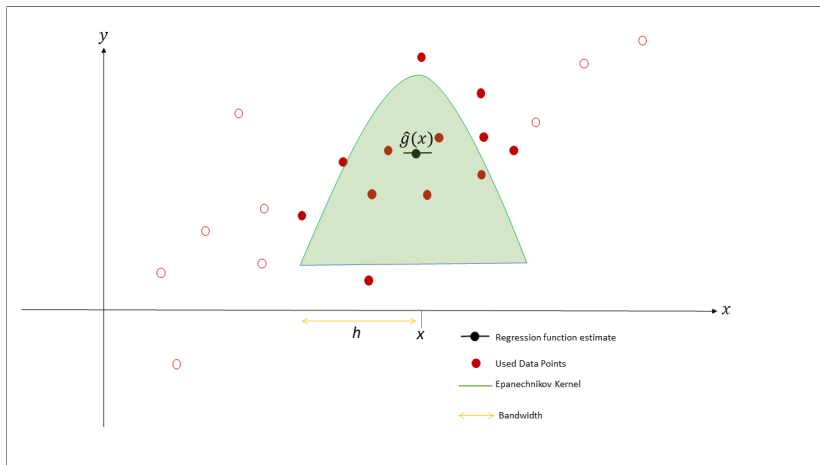
$$\min_{\gamma_0, \gamma_1} \sum_{i=1}^n \left( y_i - \gamma_0 \right)^2 \cdot K\left( \frac{x - X_i}{h} \right)$$

$\gamma_0$  : **Función de regresión** estimada  $\hat{g}(x)$

$h$ : **Bandwidth** – número positivo (muy importante!)

$K(\cdot)$ : **Kernel** – una función ( $K : \mathbb{R} \rightarrow \mathbb{R}$ )

# Ilustración – Local Constant



# Implementación en Stata

```
npregress kernel depvar indepvars [if] [in] [, options]
```

## Quick Start:

Regresión kernel local-linear de  $y$  con  $x$  y la variable discreta  $a$ :

- `npregress kernel y x i.a`

Regresión kernel local-const. de  $y$  con  $x$  y la variable discreta  $a$ :

- `npregress kernel y x i.a, estimator(constant)  
noderivatives`

## Ejemplo 7: El efecto de fumar sobre el peso de los bebés

### Base de datos:

- `lbwsim2.dta`: Extracto de Cattaneo (2010) Journal of Econometrics

### Outcome:

- `bweight`: **Peso del bebé** en gramos.

### Treatment:

- `msmoke`: **Cigarrillos** fumados durante el embarazo

### Controls:

- `mage`: Edad de la madre.
- `medu`: Años de educación de la madre.
- `alcohol`: 1 si se consumió alcohol durante el embarazo.
- `prenatal`: trimestre de la primera visita prenatal.



## Ejemplo 7: Regresión kernel Local-Linear

```
. npregress kernel bweight mage medu i.msmoke i.alcohol i.prenatal, nolog
```

Bandwidth

|          | Mean     | Effect   |
|----------|----------|----------|
| mage     | 2.178108 | 3.09538  |
| medu     | .9284651 | 1.319472 |
| msmoke   | .5       | .5       |
| alcohol  | .5       | .5       |
| prenatal | .5       | .5       |

Local-linear regression

Continuous kernel : epanechnikov

Discrete kernel : liracine

Bandwidth : cross-validation

Number of obs

E(Kernel obs)

R-squared

= 4,605

= 4,605

= 0.0893

|                         | bweight  | Estimate  |
|-------------------------|----------|-----------|
| Mean                    | bweight  | 3357.97   |
| Effect                  | mage     | 9.550276  |
|                         | medu     | 4.95285   |
|                         | msmoke   |           |
| (1-5 daily vs 0 daily)  |          | -114.3037 |
| (6-10 daily vs 0 daily) |          | -214.6567 |
| (11+ daily vs 0 daily)  |          | -306.8657 |
|                         | alcohol  |           |
| (1 vs 0)                |          | -45.48752 |
|                         | prenatal |           |
| (1 vs 0)                |          | 31.2373   |
| (2 vs 0)                |          | 10.41102  |
| (3 vs 0)                |          | -30.41237 |

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().

# Ejemplo 7: Predicciones generadas

2 - Data Editor (Browse) - [lbw\_sim2]

File Edit View Data Tools

bweight[1] 3459

|    | bweight | alcohol | mage | medu         | msmoke | prenatal | _Mean_bw... |
|----|---------|---------|------|--------------|--------|----------|-------------|
| 1  | 3459    | 0       | 24   | 14 0 daly    |        | 1        | 3432.9464   |
| 2  | 3260    | 0       | 20   | 10 0 daly    |        | 1        | 3371.1014   |
| 3  | 3572    | 0       | 22   | 9 0 daly     |        | 1        | 3454.9649   |
| 4  | 2948    | 0       | 26   | 12 0 daly    |        | 1        | 3431.5499   |
| 5  | 2410    | 0       | 20   | 12 0 daly    |        | 1        | 3335.7757   |
| 6  | 3147    | 0       | 27   | 12 0 daly    |        | 1        | 3427.7775   |
| 7  | 3799    | 0       | 27   | 12 0 daly    |        | 1        | 3427.7775   |
| 8  | 3629    | 0       | 24   | 12 0 daly    |        | 1        | 3428.6979   |
| 9  | 2835    | 0       | 21   | 12 0 daly    |        | 1        | 3370.4283   |
| 10 | 3880    | 0       | 30   | 15 0 daly    |        | 1        | 3498.2042   |
| 11 | 3090    | 0       | 26   | 12 11+ daly  |        | 1        | 3063.5346   |
| 12 | 3345    | 0       | 20   | 12 0 daly    |        | 1        | 3335.7757   |
| 13 | 4013    | 0       | 34   | 14 0 daly    |        | 1        | 3496.9393   |
| 14 | 3771    | 0       | 21   | 8 0 daly     |        | 1        | 3399.5816   |
| 15 | 662     | 0       | 23   | 12 0 daly    |        | 2        | 3378.9015   |
| 16 | 3657    | 0       | 22   | 12 0 daly    |        | 1        | 3397.0289   |
| 17 | 3572    | 0       | 26   | 12 0 daly    |        | 1        | 3431.5499   |
| 18 | 3430    | 0       | 40   | 16 0 daly    |        | 1        | 3375.6734   |
| 19 | 4479    | 0       | 34   | 12 0 daly    |        | 1        | 3454.5931   |
| 20 | 3166    | 0       | 27   | 12 6-10 daly |        | 2        | 3188.7895   |
| 21 | 4253    | 0       | 33   | 12 0 daly    |        | 1        | 3442.9508   |
| 22 | 4054    | 0       | 27   | 14 0 daly    |        | 1        | 3457.9927   |
| 23 | 3160    | 0       | 25   | 12 0 daly    |        | 1        | 3434.6629   |
| 24 | 2466    | 0       | 22   | 12 0 daly    |        | 1        | 3397.0289   |
| 25 | 3147    | 0       | 19   | 10 11+ daly  |        | 2        | 3093.1773   |
| 26 | 3232    | 0       | 33   | 14 0 daly    |        | 1        | 3491.6733   |
| 27 | 3005    | 0       | 19   | 12 0 daly    |        | 1        | 3290.9059   |
| 28 | 1899    | 0       | 36   | 17 0 daly    |        | 1        | 3470.78     |
| 29 | 4026    | 0       | 33   | 14 0 daly    |        | 1        | 3491.6733   |
| 30 | 3969    | 0       | 28   | 12 0 daly    |        | 2        | 3411.2734   |
| 31 | 4167    | 0       | 19   | 9 0 daly     |        | 1        | 3320.6182   |
| 32 | 4054    | 1       | 32   | 16 0 daly    |        | 1        | 3491.0059   |
| 33 | 3660    | 0       | 28   | 15 0 daly    |        | 1        | 3481.2317   |

Variables

Filter variables here

| <input checked="" type="checkbox"/> | Name     | Label                       | Type | Format | Value L  |
|-------------------------------------|----------|-----------------------------|------|--------|----------|
| <input checked="" type="checkbox"/> | bweight  | Infant birthweight (gra...  | int  | %9.0g  |          |
| <input type="checkbox"/>            | mmarried | 1 if mother married         | byte | %11.0g | mmarried |
| <input type="checkbox"/>            | mhispc   | 1 if mother hispanic        | byte | %9.0g  |          |
| <input type="checkbox"/>            | fhisp    | 1 if father hispanic        | byte | %9.0g  |          |
| <input type="checkbox"/>            | foreign  | 1 if mother born abroad     | byte | %9.0g  |          |
| <input checked="" type="checkbox"/> | alcohol  | 1 if alcohol consumed d...  | byte | %9.0g  |          |
| <input type="checkbox"/>            | deadkids | Previous births where n...  | byte | %9.0g  |          |
| <input checked="" type="checkbox"/> | mage     | Mother's age                | byte | %9.0g  |          |
| <input checked="" type="checkbox"/> | medu     | Mother's education attal... | byte | %9.0g  |          |

Variables Snapshots

Properties

Variables

|             |                            |
|-------------|----------------------------|
| Name        | bweight                    |
| Label       | Infant birthweight (grams) |
| Type        | int                        |
| Format      | %9.0g                      |
| Value label |                            |
| Notes       |                            |

Data

|              |                                      |
|--------------|--------------------------------------|
| Frame        | default                              |
| Filename     | lbw_sim2.dta                         |
| Label        | Excerpt from Cattaneo (2010) Journal |
| Notes        |                                      |
| Variables    | 34                                   |
| Observations | 4,605                                |
| Size         | 512.67K                              |

Ready Vars: 7 of 34 Order: Dataset Obs: 4,605 Filter: Off Mode: Browse CAP NJM

# Ejemplo 8: Errores estándar usando un Bootstrap

```
npregress kernel bweight mage medu i.smoke i.alcohol i.prenatal, nolog vce(bootstrap, reps(40))
```

Bootstrap replications (40): .....10.....20.....30.....40 done

Bandwidth

|          | Mean     | Effect   |
|----------|----------|----------|
| mage     | 2.178108 | 3.09538  |
| medu     | .9284651 | 1.319472 |
| msmoke   | .5       | .5       |
| alcohol  | .5       | .5       |
| prenatal | .5       | .5       |

Local-linear regression                      Number of obs        =        4,605  
 Continuous kernel : epanechnikov        E(Kernel obs)        =        4,605  
 Discrete kernel : lracine                R-squared            =        0.0893  
 Bandwidth        : cross-validation

|        | bweight                 | Observed estimate | Bootstrap std. err. | z      | P> z  | Percentile [95% conf. interval] |           |
|--------|-------------------------|-------------------|---------------------|--------|-------|---------------------------------|-----------|
| Mean   | bweight                 | 3357.97           | 10.23615            | 328.05 | 0.000 | 3341.884                        | 3379.431  |
| Effect |                         |                   |                     |        |       |                                 |           |
|        | mage                    | 9.550276          | 1.77653             | 5.38   | 0.000 | 6.437531                        | 12.45403  |
|        | medu                    | 4.95285           | 7.087289            | 0.70   | 0.485 | -9.00919                        | 15.87284  |
|        | msmoke                  |                   |                     |        |       |                                 |           |
|        | (1-5 daily vs 0 daily)  | -114.3037         | 10.67519            | -10.71 | 0.000 | -135.9948                       | -95.59864 |
|        | (6-10 daily vs 0 daily) | -214.6567         | 19.9158             | -10.78 | 0.000 | -257.2621                       | -179.2783 |
|        | (11+ daily vs 0 daily)  | -306.8657         | 29.30562            | -10.47 | 0.000 | -367.6013                       | -261.8348 |
|        | alcohol                 |                   |                     |        |       |                                 |           |
|        | (1 vs 0)                | -45.48752         | 42.34277            | -1.07  | 0.283 | -119.0151                       | 39.6552   |
|        | prenatal                |                   |                     |        |       |                                 |           |
|        | (1 vs 0)                | 31.2373           | 23.23159            | 1.34   | 0.179 | -16.08415                       | 73.16318  |
|        | (2 vs 0)                | 10.41102          | 41.17059            | 0.25   | 0.800 | -79.54673                       | 84.2395   |
|        | (3 vs 0)                | -30.41237         | 61.28031            | -0.50  | 0.620 | -158.3454                       | 86.92535  |

Note: Effect estimates are averages of derivatives for continuous covariates and averages of contrasts for factor covariates.

## Ejemplo 9: Presentar los resultados gráficamente

```
. npregress kernel bweight mage, nolog
```

Bandwidth

|      | Mean     | Effect   |
|------|----------|----------|
| mage | 5.795516 | 3.695218 |

Local-linear regression

Number of obs = 4,605

Kernel : epanechnikov

E(Kernel obs) = 4,605

Bandwidth: cross-validation

R-squared = 0.0132

| bweight | Estimate |
|---------|----------|
| Mean    |          |
| bweight | 3354.357 |
| Effect  |          |
| mage    | 11.41642 |

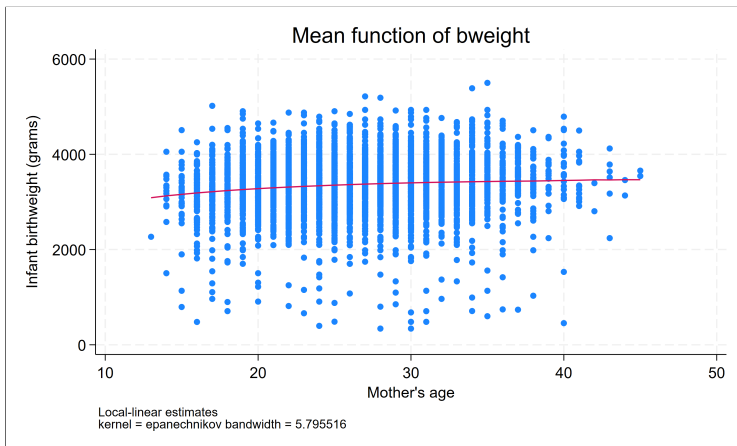
Note: Effect estimates are averages of derivatives.

Note: You may compute standard errors using vce(bootstrap) or reps().

```
.
```

```
. npgraph
```

## Ejemplo 9: Presentar los resultados gráficamente



# Agenda

- 1 El atractivo de los métodos no paramétricos
- 2 Estimando densidades de probabilidad
- 3 Estimando esperanzas condicionales / funciones de regresión
  - Regresión kernel
  - Aproximación con series
- 4 Ventajas y desventajas de los métodos no paramétricos

## Aproximación con series

Podemos **aproximar la función**  $g(\cdot)$  **usando series**. Por ejemplo, podríamos usar una **expansión de Taylor**:

$$g(x) = g(0) + \frac{g'(0)}{1!} \cdot x + \frac{g''(0)}{2!} \cdot x^2 + \frac{g^{(3)}(0)}{3!} \cdot x^3 + \text{Residuo}$$

Así, podemos **estimar la función**  $g(\cdot)$  como:

$$\hat{g}(x_i) = z(x_i)\hat{\beta}$$

Donde  $z(x_i) = (1, x_i, x_i^2, x_i^3)$  y  $\hat{\beta} = (Z^\top Z)^{-1} Z^\top y$ .

## Bases disponibles en Stata

### **Polynomial basis:**

La función  $g(\cdot)$  se aproxima con un polinomio.

### **Piecewise polynomial spline basis:**

La función  $g(\cdot)$  se aproxima con un polinomio por partes.

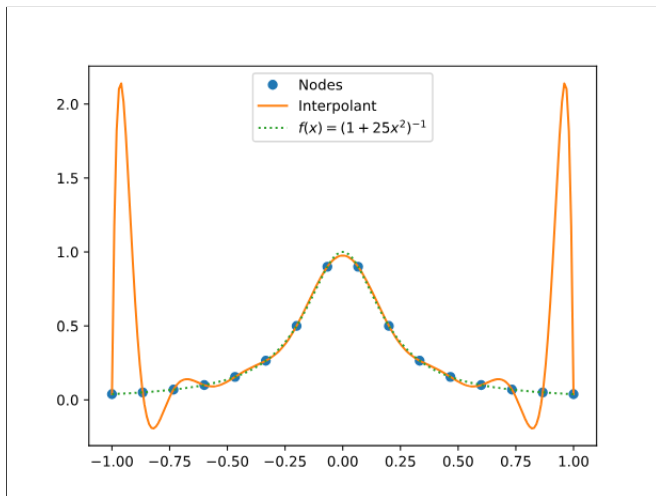
### **B-Spline basis (default):**

La función  $g(\cdot)$  se aproxima con una función spline. Una función spline es un polinomio por partes “suavizado”.

Los polinomios por partes y las splines **mitigan el fenómeno de Runge**.



# Fenómeno de Runge



# Implementación en Stata

```
npregress series depvar indepvars [if] [in] [weight] [, options]
```

## Quick Start:

Regresión no paramétrica de  $y$  con  $x$  y la variable discreta  $a$  (usando una base B-spline):

- `npregress series y x i.a`

Igual que arriba pero usando una base polinomial:

- `npregress series y x i.a, polynomial`

## Ejemplo 10: Efecto de las multas en las citaciones de DUI

### Base de datos:

- <https://www.stata-press.com/data/r18/dui> (fictional data)

### Outcome:

- citations: **Citaciones por DUI anuales** en un condado.

### Tratamiento:

- fines: **Multa** por manejar bajo los efectos del alcohol en un condado

### Controles:

- csize: Tamaño del condado (3 categorías)
- college: 1 si hay un campus universitario en el condado.

# Regresión no paramétrica en Stata usando series

```
. npregress series citations fines i.csize i.college
```

Computing approximating function

Minimizing cross-validation criterion

Iteration 0: Cross-validation criterion = 30.26251

Computing average derivatives

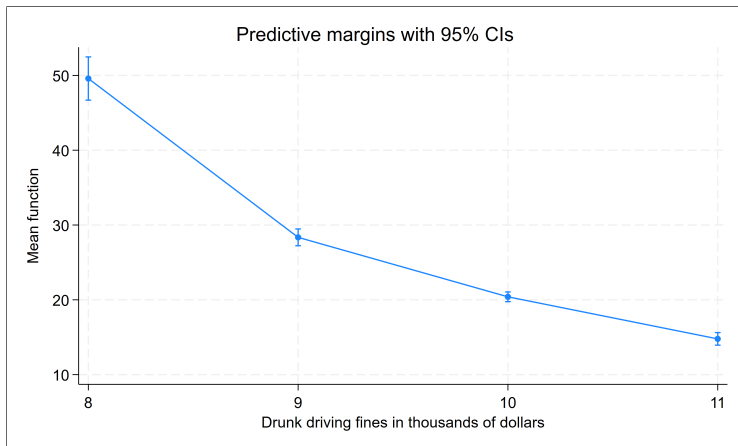
Cubic B-spline estimation                      Number of obs        =        500  
 Criterion: cross-validation                  Number of knots       =        1

| citations                | Effect    | Robust<br>std. err. | z      | P> z  | [95% conf. interval] |           |
|--------------------------|-----------|---------------------|--------|-------|----------------------|-----------|
| fines                    | -7.787386 | .2917941            | -26.69 | 0.000 | -8.359292            | -7.215481 |
| cszize                   |           |                     |        |       |                      |           |
| (Medium vs Small)        | 4.732592  | .5087968            | 9.30   | 0.000 | 3.735368             | 5.729815  |
| (Large vs Small)         | 10.91757  | .5350892            | 20.40  | 0.000 | 9.868813             | 11.96632  |
| college                  |           |                     |        |       |                      |           |
| (College vs Not college) | 6.514286  | .5958949            | 10.93  | 0.000 | 5.346353             | 7.682218  |

Note: Effect estimates are averages of derivatives for continuous covariates and averages of contrasts for factor covariates.

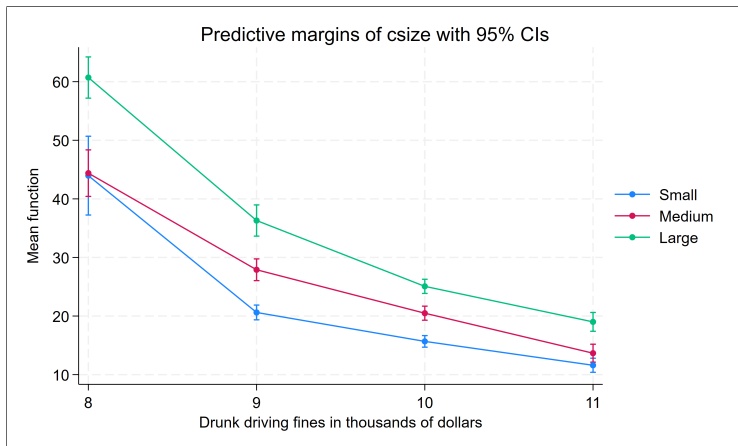
# Citaciones predichas para distintos niveles de multas

```
margins, at(fines=(8 9 10 11))
```



# El efecto de las multas según tamaño del condado

```
margins csize, at(fines=(8 9 10 11))
```



# makespline

Genera un conjunto de variables que constituyen una **base predeterminada**.

- Piecewise polynomial spline
- B-spline

Puedes usar estas variables **directamente en tus regresiones**.  
Útil para implementar **métodos semiparamétricos**.

## Quick Start:

Generar una base B-spline de tercer order a partir de las variables  $x_1$  y  $x_2$ , con nodos en las medianas de cada variable

```
makespline bspline x1 x2
```

# Ejemplo 11: Creando una base B-spline

```
sysuse auto
```

```
makespline bspline price
```

2 - Data Editor (Browse) - [auto]

File Edit View Data Tools

price[1] 4099

|    | price  | _bsp_1_1 | _bsp_1_2 | _bsp_1_3 | _bsp_1_4  | _bsp_1_5  |
|----|--------|----------|----------|----------|-----------|-----------|
| 1  | 4,099  | 11905163 | 79122659 | 08954835 | 00267342  | 0         |
| 2  | 4,749  | 00273345 | 78369636 | 20053231 | 01303788  | 0         |
| 3  | 3,799  | 28186433 | 67394655 | 04339485 | 00003636  | 0         |
| 4  | 4,816  | 00110678 | 77168007 | 21247004 | 01474831  | 0         |
| 5  | 7,627  | 0        | 30260696 | 48527033 | 2153729   | 0167404   |
| 6  | 5,788  | 0        | 58808973 | 35410686 | 0507073   | 0003561   |
| 7  | 4,453  | 02714761 | 81877214 | 14707026 | 00709999  | 0         |
| 8  | 5,189  | 0        | 89637284 | 27445574 | 02716889  | 4.535e-06 |
| 9  | 10,372 | 0        | 09933557 | 36289385 | 42255667  | 11524301  |
| 10 | 4,082  | 1285926  | 78721528 | 08375201 | 00253011  | 0         |
| 11 | 11,385 | 0        | 054977   | 28250557 | 46884961  | 19361652  |
| 12 | 14,500 | 0        | 00187027 | 04340015 | 31626842  | 63361116  |
| 13 | 15,906 | 0        | 1100e-06 | 00033303 | 03359915  | 96060672  |
| 14 | 3,289  | 76700463 | 20078032 | 00220713 | 7.918e-06 | 0         |
| 15 | 5,705  | 0        | 60334109 | 34442986 | 05197568  | 00025427  |
| 16 | 4,504  | 0203136  | 81555088 | 15625071 | 007878    | 0         |
| 17 | 5,104  | 0        | 71402566 | 26116447 | 02390017  | 6.915e-07 |
| 18 | 3,667  | 38477501 | 58861932 | 02819004 | 00041526  | 0         |
| 19 | 3,955  | 18612578 | 74775617 | 06450017 | 00161788  | 0         |
| 20 | 3,984  | 17114675 | 75830252 | 06874809 | 00180264  | 0         |
| 21 | 4,010  | 15842022 | 70680583 | 07263415 | 00197879  | 0         |
| 22 | 5,886  | 0        | 5721696  | 3647883  | 06253453  | 00060757  |
| 23 | 6,342  | 0        | 48843849 | 40702036 | 08276302  | 00177713  |
| 24 | 4,389  | 03769551 | 82064232 | 13564586 | 0080182   | 0         |
| 25 | 4,187  | 08811008 | 80759439 | 10079316 | 00350237  | 0         |
| 26 | 11,497 | 0        | 05109706 | 27309748 | 47189037  | 2039961   |
| 27 | 13,594 | 0        | 00793967 | 10555921 | 41901854  | 47248459  |
| 28 | 13,466 | 0        | 00925693 | 10993251 | 42914032  | 45167024  |
| 29 | 3,329  | 26137344 | 68545826 | 04720754 | 00098076  | 0         |
| 30 | 5,379  | 0        | 68230481 | 30233127 | 03329661  | 00003056  |
| 31 | 6,165  | 0        | 52625488 | 38212022 | 08045665  | 00116005  |
| 32 | 4,516  | 01888277 | 81459871 | 15841788 | 00080863  | 0         |
| 33 | 6,303  | 0        | 50448198 | 40369889 | 06969341  | 00162594  |

Ready

Vars: 6 of 17 Order Modified Obs: 74 Filter Off Mode: Browse CAP NUM

**Variables**

Filter variables here

| <input checked="" type="checkbox"/> Name     | Label                       | Type   | Format | Value L |
|--|-----------------------------|--------|--------|---------|
| <input checked="" type="checkbox"/> price    | Price                       | int    | %8.0gc |         |
| <input checked="" type="checkbox"/> _bsp_1_1 | B-spline basis term 1 fo... | double | %10.0g |         |
| <input checked="" type="checkbox"/> _bsp_1_2 | B-spline basis term 2 fo... | double | %10.0g |         |
| <input checked="" type="checkbox"/> _bsp_1_3 | B-spline basis term 3 fo... | double | %10.0g |         |
| <input checked="" type="checkbox"/> _bsp_1_4 | B-spline basis term 4 fo... | double | %10.0g |         |
| <input checked="" type="checkbox"/> _bsp_1_5 | B-spline basis term 5 fo... | double | %10.0g |         |
| <input type="checkbox"/> make                | Make and model              | str18  | %-18s  |         |
| <input type="checkbox"/> mpg                 | Mileage (mpg)               | int    | %8.0g  |         |
| <input type="checkbox"/> rep78               | Repair record 1978          | int    | %8.0g  |         |

**Variables Snapshots**

**Properties**

**Variables**

| Name        | price  |
|-------------|--------|
| Label       | Price  |
| Type        | int    |
| Format      | %8.0gc |
| Value label |        |
| Notes       |        |

**Data**

| Frame        | default              |
|--------------|----------------------|
| Filename     | auto.dta             |
| Label        | 1978 automobile data |
| Notes        |                      |
| Variables    | 17                   |
| Observations | 74                   |
| Size         | 6.00K                |



# Métodos semiparamétricos usando `makespline`

Usaremos **datos simulados** para este ejemplo:

$$y = 3x_1 + 3 \sin(3(x_2 - x_3)) + \varepsilon$$

Como investigadores, **no conocemos** la forma funcional específica. No obstante, suponemos un **modelo semiparamétrico**:

$$y = \beta_1 x_1 + g(x_2, x_3) + \varepsilon$$

Nuestro **parámetro de interés** es  $\beta_1$ . Aquí,  $g(\cdot)$  es un **parámetro secundario**.

# Paso 1: Crear la base B-Spline

- use <https://www.stata-press.com/data/r18/splines>
- `makespline bspline x2 x3, knots(8)`

Data Editor (Browse) - [splines]

File Edit View Data Tools

└─ 66608318

|    | x1        | x2        | x3        | gx        | e         | y         | _bsp_1_1   | _bsp_1_2   | _bsp_1_3   | _bsp_1_4   | _bsp_1_5  | _bsp_1_6   |           |
|----|-----------|-----------|-----------|-----------|-----------|-----------|------------|------------|------------|------------|-----------|------------|-----------|
| 1  | -6098636  | -6303378  | 2221987   | -1.563013 | -4785608  | -1.071334 | 0          | 0          | 0.03202160 | 39538540   | 58951465  | 05127823   |           |
| 2  | 20345555  | 1590177   | 1.035121  | -1.449541 | -1022404  | 1.968577  | 0          | 0          | 0          | 0          | 0         | 0.03216641 |           |
| 3  | -5158365  | -2234488  | -0.044974 | -1.68813  | -4402962  | -8739348  | 0          | 0          | 0          | 0          | 0         | 0.03943067 | 5851080   |
| 4  | 9774078   | 1382372   | 3455514   | 0.003785  | 11.32719  | 11.48834  | 0          | 0          | 0          | 0          | 0         | 0          |           |
| 5  | -1.255500 | -1.209291 | -5.081133 | -2.537384 | -1.763282 | -5.067212 | 0.0980011  | 42528647   | 42834281   | 08947062   | 0         | 0          |           |
| 6  | 4983057   | -604555   | 229545    | 5.597749  | 2.407091  | 7.514023  | 0          | 0.0335681  | 24994128   | 885213     | 05128614  | 0          |           |
| 7  | -4853438  | -8628554  | 1591382   | -4.015542 | -2.85422  | 1.711396  | 0          | 0.00553943 | 1748667    | 72938522   | 09944885  | 0          |           |
| 8  | -8410513  | -1.303768 | 5544983   | 1.972259  | -3893099  | 2.056165  | 0.845715   | 46405432   | 38298594   | 06780623   | 0         | 0          |           |
| 9  | 0.899468  | 1.207115  | -4.323741 | -2.809523 | 1.542085  | 1.875412  | 0          | 0          | 0          | 0          | 0         | 0          |           |
| 10 | -2.182855 | 1.982164  | 0.034829  | -1.430589 | -2.185232 | -1.184862 | 0          | 0          | 0          | 0          | 0         | 0          |           |
| 11 | 1.872006  | -3577864  | -6.003853 | 2.017756  | -9.061822 | 9.725593  | 0          | 0          | 0          | 0.0055526  | 2582525   | 04414740   |           |
| 12 | 3993892   | -1.007385 | -4.867571 | -2.988879 | 7.728785  | 8.91699   | 0          | 0.0388686  | 48001864   | 49528611   | 00098209  | 0          |           |
| 13 | 1.239301  | -7547539  | -6.005128 | -3.899028 | -3.541662 | 4.774028  | 0          | 0.03915668 | 61772811   | 33994181   | 00317286  | 0          |           |
| 14 | 1089767   | 1.019357  | -1.088197 | -6.70389  | 6.171591  | 8.803824  | 0          | 0          | 0          | 0          | 0         | 0          |           |
| 15 | 1127381   | 1.122267  | 1.67718   | -2.566079 | 2.066117  | 2.840102  | 0.03244405 | 39296275   | 55283127   | 25282192   | 0         | 0          |           |
| 16 | -6031965  | 1.193125  | 878285    | 1.652332  | 6.522084  | 8.503025  | 0          | 0          | 0          | 0          | 0         | 0          |           |
| 17 | 1.58686   | 3262983   | -1.048639 | -2.488534 | -3.210885 | 2.012272  | 0          | 0          | 0          | 0          | 0         | 0          |           |
| 18 | -1.231102 | 1170859   | -2.89257  | 2.818295  | -2.822703 | -8.966025 | 0          | 0          | 0          | 0          | 0         | 0          | 0.0002885 |
| 19 | 7148838   | -3047527  | 0.870202  | -2.835122 | 5.190594  | 7.496523  | 0          | 0          | 0          | 3.888607   | 18083439  | 06952104   |           |
| 20 | -3828277  | 1477628   | -4.02393  | -2.808803 | 3.542481  | 6.295881  | 0          | 0          | 0          | 0          | 0         | 0          | 0.0371285 |
| 21 | 5487259   | -0.048841 | 1.267137  | 2.462713  | 3.528167  | 10.14887  | 0          | 0          | 0          | 0          | 0         | 0          | 0.0523591 |
| 22 | -9480666  | -0.943417 | 2.159828  | -1.774    | -2.119032 | -1.236502 | 0          | 0.02738176 | 4382182    | 55545680   | 00828052  | 0          |           |
| 23 | -1.38896  | -7018036  | -8.688141 | 8478326   | -2.065478 | 7.228716  | 0          | 0          | 0.06162548 | 67142788   | 20935014  | 00098649   |           |
| 24 | 1.1355448 | -4485596  | 8.049546  | 0.947988  | 3.228181  | 8.822362  | 0          | 0          | 0          | 0.02811652 | 48510136  | 48325747   |           |
| 25 | 4182784   | 885101    | 1.923957  | -1.648838 | -2.313863 | 1.768677  | 0          | 0          | 0          | 0          | 0         | 0          |           |
| 26 | -7074443  | 3.654576  | -0.404403 | -1.130014 | 2.864551  | 2.845889  | 0          | 0          | 0          | 0          | 0         | 0          |           |
| 27 | 1.888755  | -3029553  | -6.847763 | 2.868931  | 2.781032  | 14.4404   | 0          | 0          | 0          | 0.0002605  | 27153657  | 37545555   |           |
| 28 | -3231472  | -1.49148  | -1.087834 | 2.523082  | 3.138788  | 3.384822  | 45182113   | 4493196    | 00382404   | 00483322   | 0         | 0          |           |
| 29 | 0.078384  | 0.025480  | -20.01218 | -2.63829  | 2.286891  | 2.431711  | 0          | 0.121440   | 3351729    | 63188643   | 02086567  | 0          |           |
| 30 | 8755918   | 2093304   | -3.77574  | 2.722489  | -1.353158 | 1.742536  | 0          | 0          | 0          | 0          | 0         | 0          |           |
| 31 | 3.047812  | 6.09892   | 32.00772  | 2.274573  | 7.104568  | 21.52258  | 0          | 0          | 0          | 0          | 0         | 0          |           |
| 32 | -1.18735  | -18.104   | -1.158448 | 6111985   | -3.778881 | -0.730725 | 0          | 0          | 0          | 0          | 0.0482443 | 48824888   |           |
| 33 | -3218362  | 5309096   | -4.026172 | 8579376   | -4.216258 | -1.224457 | 0          | 0          | 0          | 0          | 0         | 0          |           |

Variables

Filter variables here

| Name     | Label                       | Type   | Format | Value l |
|----------|-----------------------------|--------|--------|---------|
| x1       |                             | float  | %9.0g  |         |
| x2       |                             | float  | %9.0g  |         |
| x3       |                             | float  | %9.0g  |         |
| gx       |                             | float  | %9.0g  |         |
| e        |                             | float  | %9.0g  |         |
| y        |                             | float  | %9.0g  |         |
| _bsp_1_1 | B-spline basis term 1 fo... | double | %10.0g |         |
| _bsp_1_2 | B-spline basis term 2 fo... | double | %10.0g |         |
| _bsp_1_3 | B-spline basis term 3 fo... | double | %10.0g |         |

Variables Snapshots

Properties

Variables

| Name | Label | Type  | Format | Value label | Notes |
|------|-------|-------|--------|-------------|-------|
| x1   |       | float | %9.0g  |             |       |

Data

Filename

Label

Format

Variables

Observations

Size

Mode Browse

Los elementos de la base se guardan en r(regressors)

## Paso 2: Usar LASSO para escoger los elementos de la base

Con todas las interacciones tendríamos **168 regresores**. Usamos **poregress** para seleccionar de entre estos usando **LASSO**.

- `poregress y x1, controls('r(regressors)')`

```
. poregress y x1, controls('r(regressors)')
```

```
Estimating lasso for y using plugin
```

```
Estimating lasso for x1 using plugin
```

```
Partialing-out linear model      Number of obs      =      5,000
                                Number of controls      =      168
                                Number of selected controls =      19
                                Wald chi2(1)              =    3535.78
                                Prob > chi2                =      0.0000
```

|  |    | Robust      |           |       |       |                      |
|--|----|-------------|-----------|-------|-------|----------------------|
|  | y  | Coefficient | std. err. | z     | P> z  | [95% conf. interval] |
|  | x1 | 2.951242    | .049632   | 59.46 | 0.000 | 2.853965 3.048519    |

Note: Chi-squared test is a Wald test of the coefficients of the variables of interest jointly equal to zero. Lassos `select controls` for model estimation. Type `lassoinfo` to see number of selected variables in each lasso.

# Agenda

- 1 El atractivo de los métodos no paramétricos
- 2 Estimando densidades de probabilidad
- 3 Estimando esperanzas condicionales / funciones de regresión
  - Regresión kernel
  - Aproximación con series
- 4 Ventajas y desventajas de los métodos no paramétricos

# Limitaciones de regresión kernel y aproximación con series

**1. Costo computacional:** Tiempo CPU incrementa rápidamente con  $n$  y  $k$ . Problema es más agudo en aproximación con series.

- Regresión kernel
- Opción `nointeract`.

**2. ‘Maldición de la dimensionalidad’:** Cuando  $k$  es grande,  $n$  debe ser muy grande para que los IC sean informativos.

- Intentar tener bastantes observaciones por regresor.
- Métodos semiparamétricos.

## Conclusión: Métodos no paramétricos en Stata

1. Herramienta poderosa para explorar las **relaciones entre variables**.
  - No requieren **supuestos de forma funcional**.
  - No hay riesgo de **mala especificación**.
2. Fáciles de implementar en Stata
  - `npregress`: Estimar  $g(\cdot)$ , efectos, y hacer predicciones.
  - `margins`: explorar  $\hat{g}(\cdot)$  y responder preguntas interesantes.
3. Cuidado con las limitaciones:
  - **Costo computacional**.
  - **'Maldición de la dimensionalidad'**.

## ¿Dónde aprender más?

1. Documentación de Stata:  
<https://www.stata.com/features/documentation/>
2. Canal de YouTube: <https://www.youtube.com/user/statacorp>
3. Envía un correo a nuestro soporte técnico:  
[tech-support@stata.com](mailto:tech-support@stata.com).
4. El comando `help`.

¡Gracias!