23rd London Stata Users Group Meeting
7-8 September 2017
Cass Business School, London, UK

Nonparametric Synthetic Control Method for program evaluation: Model and Stata implementation

Giovanni Cerulli

CNR-IRCrES,
National Research Council of Italy
Research Institute on Sustainable Economic Growth

The Synthetic Control Method (SCM)

- In some cases, treatment and potential control groups do not follow **parallel trends**. Standard DID method would lead to biased estimates.

- The basic idea behind synthetic controls is that a **combination of units** often provides a better comparison for the unit exposed to the intervention than any single unit alone.

- Abadie and Gardeazabal (2003) pioneered a synthetic control method when estimating the effects of the terrorist conflict in the Basque Country using other **Spanish regions as a comparison group**.

- They want to evaluate whether Terrorism in the Basque Country had a negative effect on growth. They cannot use a standard DID method because none of the other Spanish regions followed the *same time trend* as the Basque Country.

- They therefore take a **weighted average of other Spanish regions** as a synthetic control group.

METHOD

They have *J* available control regions (i.e., the 16 Spanish regions other than the Basque Country).

They want to assign weights $\boldsymbol{\omega} = (\omega_1, ..., \omega_J)'$ – which is a $(J \times 1)$ vector – to each region:

$$\omega_j \geq 0 \quad \text{with} \quad \sum_{j=1}^{J} \omega_j = 1$$

The weights are chosen so that the **synthetic Basque country** most closely resembles the **actual one** ***before*** terrorism.

Let $\mathbf{x}_1$ be a $(K \times 1)$ vector of pre-terrorism economic growth predictors in the Basque Country.

Let $\mathbf{X}_0$ be a $(K \times J)$ matrix which contains the values of the same variables for the J possible control regions.

Let $\mathbf{V}$ be a diagonal matrix with non-negative components reflecting the **relative importance** of the different growth predictors. The vector of weights $\boldsymbol{\omega}^*$ is then chosen to *minimize*:

$$D(\boldsymbol{\omega}) = (\mathbf{x}_1 - \mathbf{X}_0\, \boldsymbol{\omega})'\, \mathbf{V}\, (\mathbf{x}_1 - \mathbf{X}_0\, \boldsymbol{\omega})$$

They choose the matrix $\mathbf{V}$ such that the real per capita GDP path for the Basque Country during the 1960s (pre terrorism) is best reproduced by the resulting synthetic Basque Country.

Alternatively, they could have just chosen the weights to reproduce ***only*** **the pre-terrorism growth path for the Basque country**. In that case, the vector of weights $\boldsymbol{\omega}^*$ is then chosen to *minimize*:

$$G(\boldsymbol{\omega}) = (\mathbf{z}_1 - \mathbf{Z}_0\, \boldsymbol{\omega})' \; (\mathbf{z}_1 - \mathbf{Z}_0\, \boldsymbol{\omega})$$

where:

$\mathbf{z}_1$ is a (10×1) vector of pre-terrorism (1960-1969) GDP values for the Basque Country

$\mathbf{Z}_0$ is a $(10 \times J)$ matrix of pre-terrorism (1960-1969) GDP values for the J potential control regions.

Constructing the counterfactual using the weights

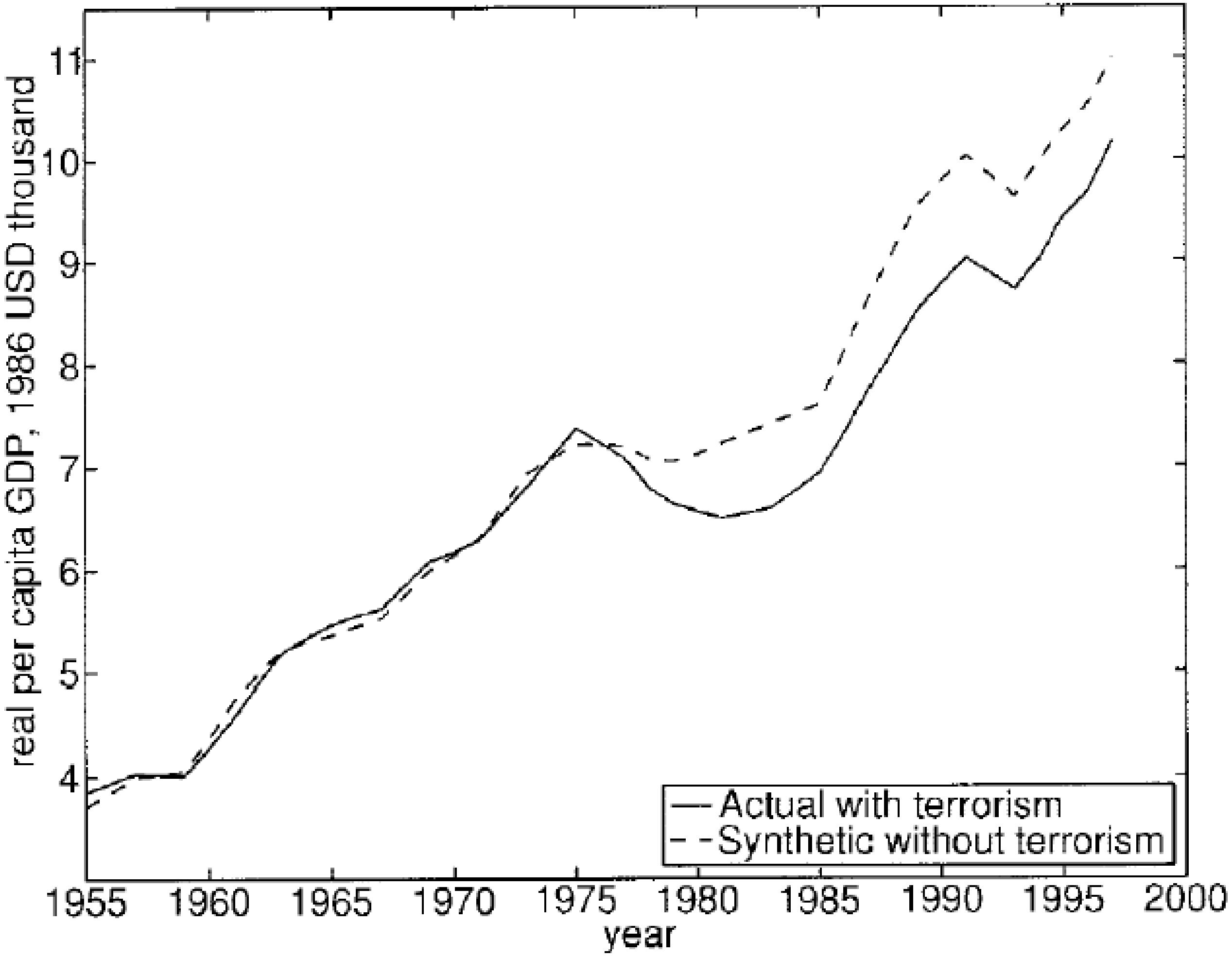
$\mathbf{y_1}$ is a (T x 1) vector whose elements are the values of real per capita GDP values for T years in the Basque country.

$\mathbf{y_0}$ is a (T x J) matrix whose elements are the values of real per capital GDP values for T years in the control regions.

They then constructed the **counterfactual GDP pattern** (i.e. in the absence of terrorism) as:

$$\underset{T\times 1}{\mathbf{y}_1^*} = \underset{T\times J}{\mathbf{y}_0} \cdot \underset{J\times 1}{\boldsymbol{\omega}^*}$$

Growth in the Basque Country *with* and *without* terrorism



Nonparametric Synthetic Control Methods (NPSCM)

- I propose an extension to the previous approach.
- The idea is that of computing the **weights** using a **kernel-vector-distance** approach.
- Given a certain **bandwidth**, this method allows to estimate a **matrix of weights** proportional to the **distance** between the treated unit and all the rest of untreated units.
- Therefore, instead of relying on one single vector of weights common to all the years, we get a vector of weights for each year.

An instructional example of the NSCM

- Suppose the treated country is UK, and treatment starts at 1973.

- Assume that the pre-treatment period is {1970, 1971, 1972}, and the post-treatment period is {1973, 1974, 1975}.

- Three countries used as controls: FRA, ITA, and GER.

- We have an available set of M covariates: $\mathbf{x} = \{x_1, x_{1, \ldots,} x_M\}$ for each country.

- We define a distance metric based on $\mathbf{x}$ between each pair of countries in each year. For instance: with only one covariate x (i.e. M=1), the distance between – let's say – UK and ITA in terms of x in 1970 may be:

$$d_{1970}(UK, ITA) = | x_{1970,UK} - x_{1970,ITA} |$$

- Given such *distance definition*, the **pre-treatment weight** for ITA will be:

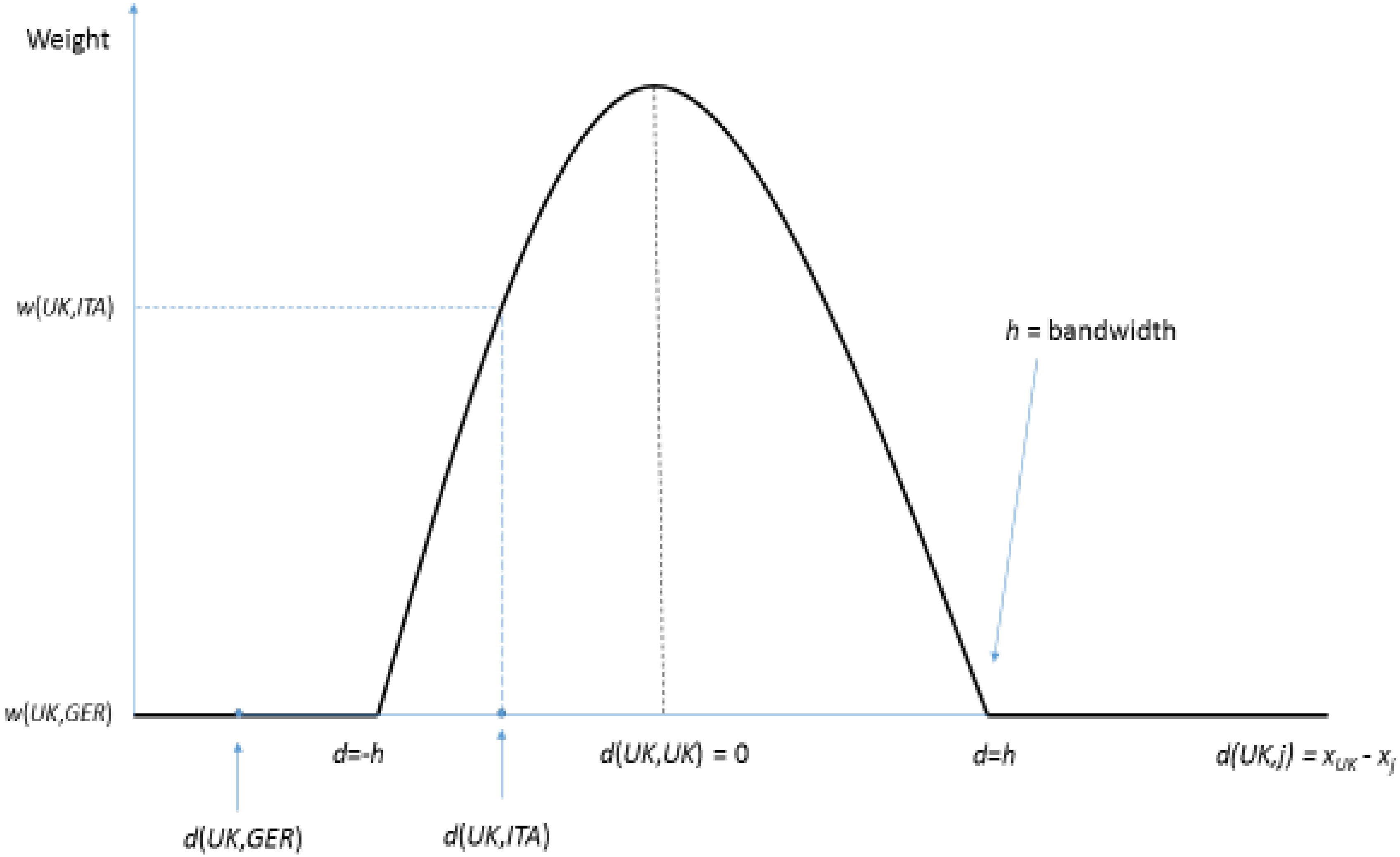
$$\omega_{1970,ITA}^{\text{UK}}(h) = K\left(\frac{|x_{1970,UK} - x_{1970,ITA}|}{h}\right)$$

where $K(\cdot)$ is one specific kernel function, and h is the **bandwidth** chosen by the analyst.

The Kernel function defines a **weighting scheme** penalizing countries that are far away from UK and giving more relevance to countries closer to UK.

Important: ***closeness*** is measured in terms of a pre-defined **x**-distance such as the Mahalanobis, Euclidean (L2), Modular, etc.

Understanding kernel distance weighting



Based on the vector-distance over the covariates: $\mathbf{x} = \{x_1, x_{1,} \ldots, x_M\}$, we can derive the **matrix of weights** $\mathbf{W}$, whose generic element is:

$$\omega_{t,s}^{\text{UK}}(h) = K\left(\frac{|\mathbf{x}_{t,s} - \mathbf{x}_{t,s}|}{h}\right)$$

In the previous example, we have:

$$\mathbf{W} = \begin{pmatrix} & 1970 & 1971 & 1972 \\ \text{FRA} & \omega_{11}^{\text{UK}} & \omega_{12}^{\text{UK}} & \omega_{13}^{\text{UK}} \\ \text{ITA} & \omega_{21}^{\text{UK}} & \omega_{22}^{\text{UK}} & \omega_{23}^{\text{UK}} \\ \text{GER} & \omega_{31}^{\text{UK}} & \omega_{32}^{\text{UK}} & \omega_{33}^{\text{UK}} \end{pmatrix}$$

Now, we define the matrix of data $\mathbf{Y}$ as follows, where y is the target variable:

$$\mathbf{Y} = \begin{pmatrix} & FRA & ITA & GER \\ 1970 & y_{11} & y_{12} & y_{13} \\ 1971 & y_{21} & y_{22} & y_{23} \\ 1972 & y_{31} & y_{32} & y_{33} \\ 1973 & y_{41} & y_{42} & y_{43} \\ 1974 & y_{51} & y_{52} & y_{53} \\ 1975 & y_{61} & y_{62} & y_{63} \end{pmatrix}$$

We define the unit *weight* as an ***average*** over the years:

$$\bar{\omega}_s^{UK} = \frac{1}{3} \sum_{t=1970}^{1972} \omega_{t,s}^{UK}$$

We also define an augmented weighting matrix we call $\mathbf{W}^*$:

$$\mathbf{W}^* = \begin{pmatrix} & 1970 & 1971 & 1972 & 1973 & 1974 & 1975 \\ \text{FRA} & \bar{\omega}_{FRA}^{UK} & \bar{\omega}_{FRA}^{UK} & \bar{\omega}_{FRA}^{UK} & \bar{\omega}_{FRA}^{UK} & \bar{\omega}_{FRA}^{UK} & \bar{\omega}_{FRA}^{UK} \\ \text{ITA} & \bar{\omega}_{ITA}^{UK} & \bar{\omega}_{ITA}^{UK} & \bar{\omega}_{ITA}^{UK} & \bar{\omega}_{ITA}^{UK} & \bar{\omega}_{ITA}^{UK} & \bar{\omega}_{ITA}^{UK} \\ \text{GER} & \bar{\omega}_{GER}^{UK} & \bar{\omega}_{GER}^{UK} & \bar{\omega}_{GER}^{UK} & \bar{\omega}_{GER}^{UK} & \bar{\omega}_{GER}^{UK} & \bar{\omega}_{GER}^{UK} \end{pmatrix}$$

Once computed an imputation of the post-treatment weights, we can define a matrix **C** as follows:

$$\underset{T\times T}{\mathbf{C}} = \underset{T\times J}{\mathbf{Y}} \cdot \underset{J\times T}{\mathbf{W}^{*}}$$

The **diagonal** of matrix **C** contains the **"UK synthetic time series $\mathbf{Y}_0$"**:

$$\mathbf{Y}_{0,\mathrm{UK}} = \mathrm{diag}(\mathbf{C})$$

This vector is an **estimation** of the ***unknown*** **counterfactual** behavior of UK.

The generic element of the diagonal of **C** is:

$$c_t = \underset{1\times J}{y_t} \cdot \underset{J\times 1}{\overline{w}^*}$$

In the previous example:

$$c_{75}^{UK} = \left[y_{75,FRA}, y_{75,ITA}, y_{75,GER} \right] \cdot \begin{bmatrix} \overline{\omega}_{FRA}^{UK} \\ \overline{\omega}_{ITA}^{UK} \\ \overline{\omega}_{GER}^{UK} \end{bmatrix} = \sum_{s=ITA,FRA,GER} y_{75,s} \overline{\omega}_s$$

Therefore, it is now clearer that c_t is a **weighted mean** of controls' y at time t, with weights provided by the previous procedure.

Previous estimation of the synthetic counterfactual is based on a specific choice of the bandwidth h. Thus, one question is how to select properly such bandwidth. As usual with non-parametric estimators, a *cross-validation* approach can be used. In this context, it reduces to select the *optimal* bandwidth as the one minimizing as loss objective function the pre–intervention *Root Mean Squared Prediction Error* (RMSPE) defined as:

$$\text{RMSPE}_j(h) = \sqrt{\frac{1}{T_{-0}} \sum_{t=1}^{T_{-0}} [y_{j,t} - y^*_{j,t}(h)]^2}$$

where T_{-0} is the last pre–treatment time. We can estimate the optimal bandwidth computationally, by forming a grid of possible values for h and then finding h^*, the value of the bandwidth minimizing the RMSPE over the grid. We provide an example of such a procedure in the next section.

The Stata command npsynth

Title

npsynth -   Nonparametric Synthetic Control Method

Syntax

npsynth *outcome* [*varlist*], t_0(#) bandw(#) panel_var(*varname*) time_var(*varname*) trunit(#) kern(*kerneltype*) [w_median gr_y_name(*name*) gr_tick(#) gr1 gr2 gr3 save_gr1(*graphname1*) save_gr2(*graphname2*) save_gr3(*graphname3*)

Description

npsynth extends the Synthetic Control Method (SCM) for program evaluation proposed by Abadie and Gardeazabal (2003) and Abadie, Diamond, and Hainmueller (2010) to the case of a nonparametric identification of the synthetic (or counterfactual) time pattern of a treated unit. The model assumes that the treated unit - such as a country, a region, a city, etc. - underwent a specific intervention in a given year, and estimates its counterfactual time pattern, the one without intervention, as a weighted linear combination of control units based on the predictors of the outcome. The nonparamentric imputation of the counterfactual is computed using weights proportional to the vector-distance between the treated unit's and the controls' predictors, using a kernel function with pre-fixed bandwidth. The routine provides a graphical representation of the results for validation purposes.

According to the npsynth syntax:

outcome: is the target variable over which measuring the impact of the treatment

varlist: is the set of covariates (or observable confounding) predicting the outcome in the pre-treatment period

Options

kern(*kerneltype*) specifies the type of kernel function to use for builfing synthetic weights.

t_0(#) specifies the time in which treatment starts.

bandw(#) specifies the bandwidth of the kernel weighting function.

panel_var(*varname*) specifies the panel variable.

time_var(*varname*) specifies the time variable.

w_median specifies that the unique vector of synthetic weights is calculated by the yearly weights' median (the default uses the mean).

gr_y_name(*name*) allows to give a convenient name to the outcome variable to appear in the graphs.

gr_tick(#) allows to set the tick of the time in the time axis of the graphs.

gr1: allows to plot the the pre-treatment balancing and parallel trend graph.

gr2: allows to plot the overall treated/synthetic pattern comparison graph.

gr3: allows to plot the overall pattern of the difference between the treated and synthetic pattern graph.

save_gr1(*graphname1*) allows to save graph 1, i.e. the pre-treatment balancing and parallel trend.

save_gr2(*graphname2*) allows to save graph 2, i.e. the overall treated/synthetic pattern comparison.

save_gr3(*graphname3*) allows to save graph 3, i.e. the overall pattern of the difference between the treated and synthetic pattern.

kerneltype_options	Description
kern	
epan	uses a Epanechnikov kernel
normal	uses a Normal kernel
biweight	uses a Biweight (or Quartic) kernel
uniform	uses a Uniform kernel
triangular	uses a Triangular kernel
tricube	uses a Tricube kernel

npsynth returns the following objects:

e(bandh) is the bandwidth used within the selected kernel function.

e(RMSPE) is the Root Mean Squared Prediction Error of the estimated model.

e(W) is the vector of (kernel) weights.

Application

Aim: comparison between parametric and nonparametric approaches

Policy: effects of adopting the Euro as national currency on exports

Treated: Italy

Outcome: Domestic Direct Value Added Exports

Covariates: countries' distance, sum of GDP, common language, contiguity

Goodness-of-fit: pre-intervention Root Mean Squared Prediction Error (RMSPE) for Italy

Donors pool: 18 countries worldwide, experiencing no change in currency

Years: 1995 - 2011

PARAMETRIC vs. NONPARAMETRIC: synth vs. npsynth

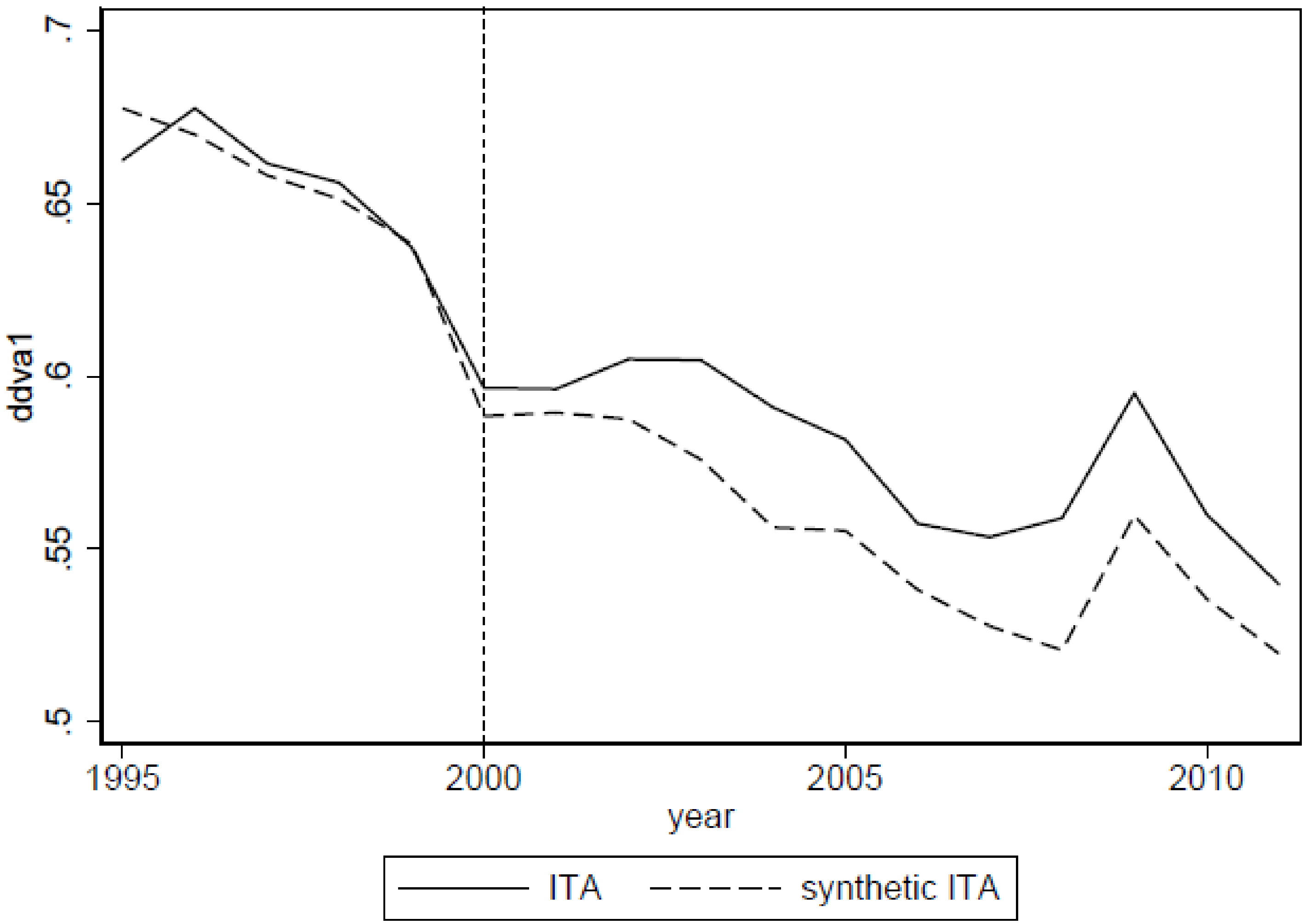
```
. use Ita_exp_euro , clear
. tsset reporter year
. global xvars "ddva1 log_distw sum_rgdpna comlang contig"

* PARAMETRIC
. synth ddva1  $xvars , trunit(11) trperiod(2000) figure  // ITA
```

```
-------------------------------------------------------------------------------------------
Loss: Root Mean Squared Prediction Error
-------------------------------------------------------------------------------------------
RMSPE | .0079342
---------------------
Unit Weights:
-----------------------
Co_No      | Unit_Weight
-----------+------------
AUS        | 0
BRA        | 0
CAN        | 0
CHN        | 0
CZE        | 0
DNK        | 0
GBR        | .122
HUN        | 0
IDN        | 0
IND        | 0
JPN        | .18
KOR        | 0
MEX        | 0
POL        | .599
ROM        | 0
SWE        | .099
TUR        | 0
USA        | 0
-----------------------
```

```
Predictor Balance:
------------------------------------------------------
            | Treated         Synthetic
-------------------------------+----------------------
ddva1       | .6587541        .6587987
log_distw   | 7.708661         7.839853
sum_rgdpna  | 27.20794        26.33796
comlang     |        0        .0234725
contig      | .0824561        .088393
------------------------------------------------------
```

Parametric model

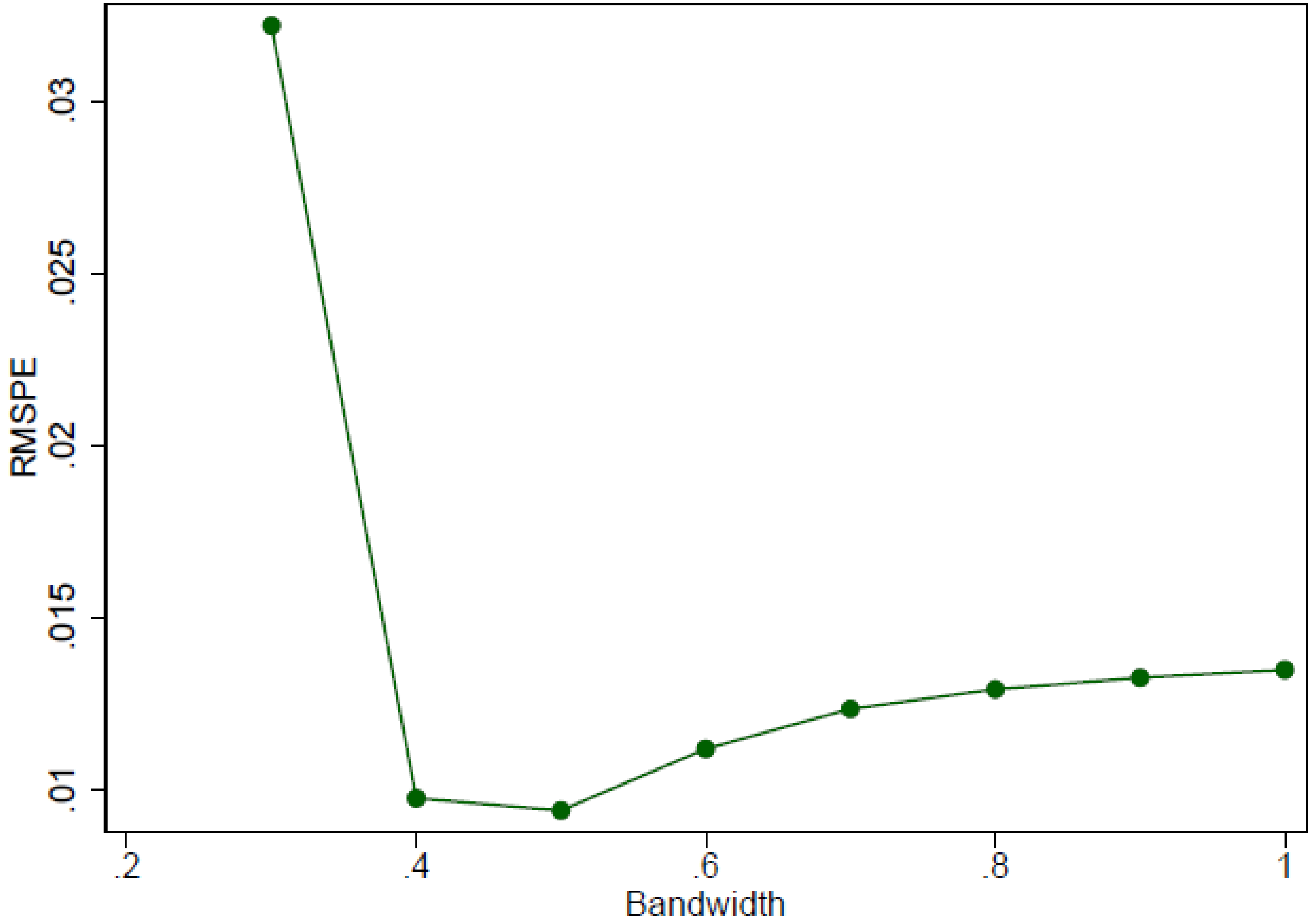
Treated and synthetic pattern of the outcome variable DDVA.

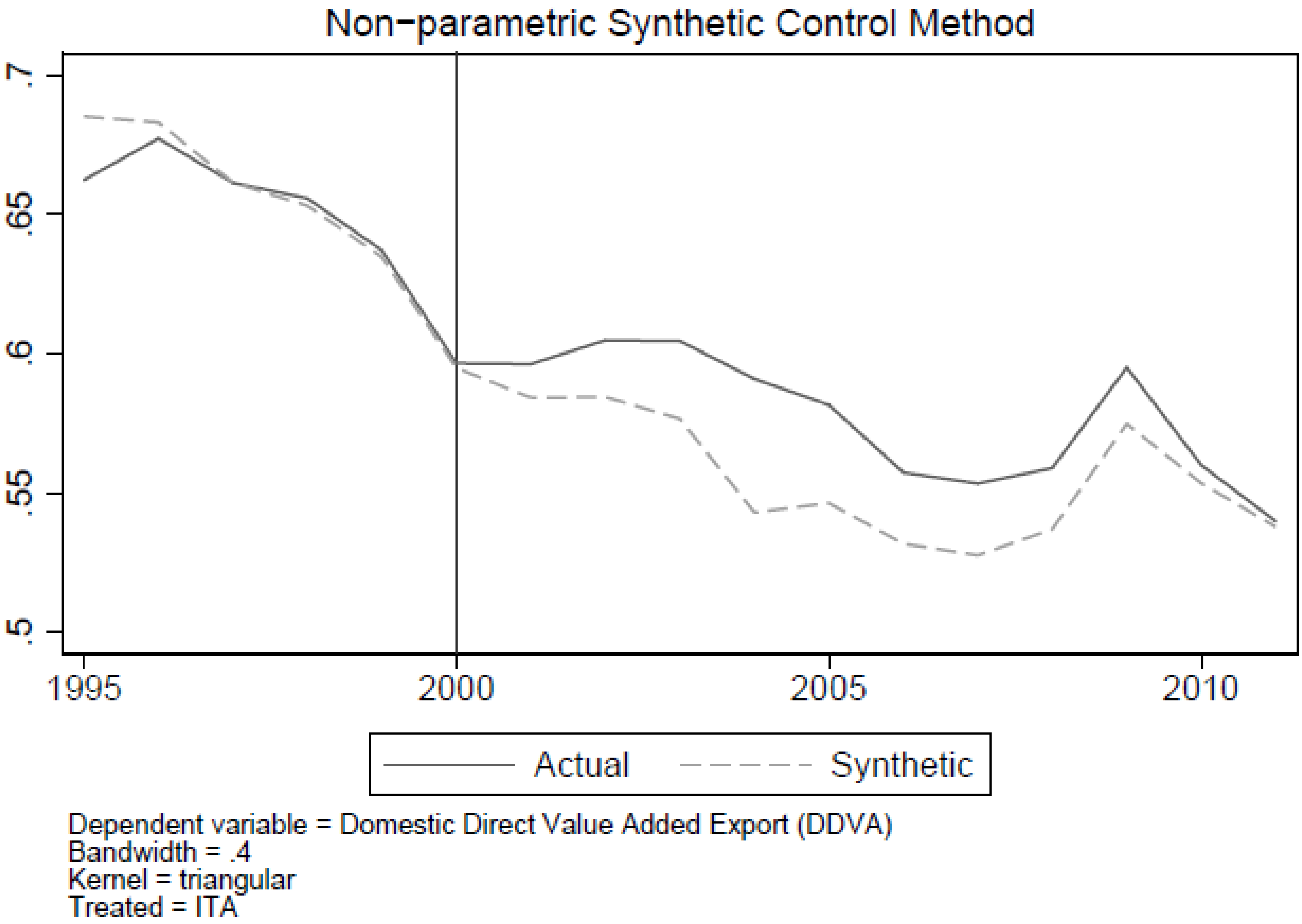


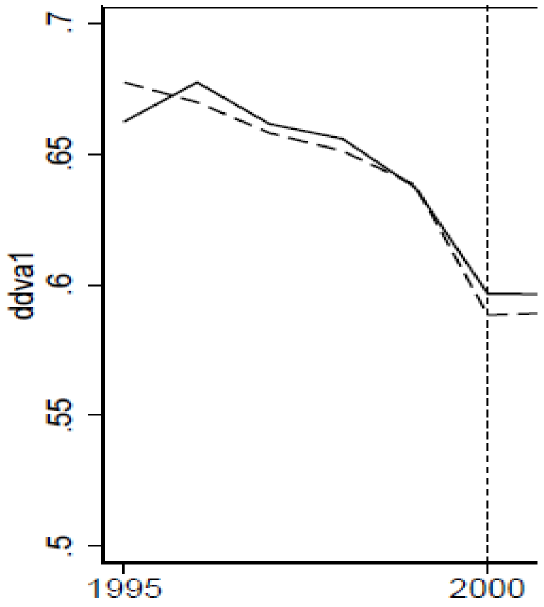
```
* NON-PARAMETRIC
. npsynth ddva1 $xvars , panel_var(reporter) time_var(year)  t0(2000)  ///
  trunit(11) bandw(0.4) kern(triangular) gr1 gr2 gr3 ///
  save_gr1(gr1) save_gr2(gr2) save_gr3(gr3) ///
  gr_y_name("Domestic Direct Value Added Export (DDVA)") gr_tick(5)
```

```
Root Mean Squared Prediction Error (RMSPE)
-------------------------------------------
RMSPE = .01
-------------------------------------------
AVERAGE UNIT WEIGHTS
-------------------------------------------
-------------------------------------------
UNIT |   WEIGHT
-------------------------------------------
AUS  |        0
BRA  |        0
CAN  |        0
CHN  | .3569087
CZE  | .1244664
DNK  |        0
GBR  | .0133546
HUN  |        0
IDN  | .035076
IND  |        0
JPN  | .1021579
KOR  |        0
MEX  | .0083542
POL  | .0563253
ROM  | .0733575
SWE  | .0837784
TUR  | .1410372
USA  | .0051846
-------------------------------------------
```

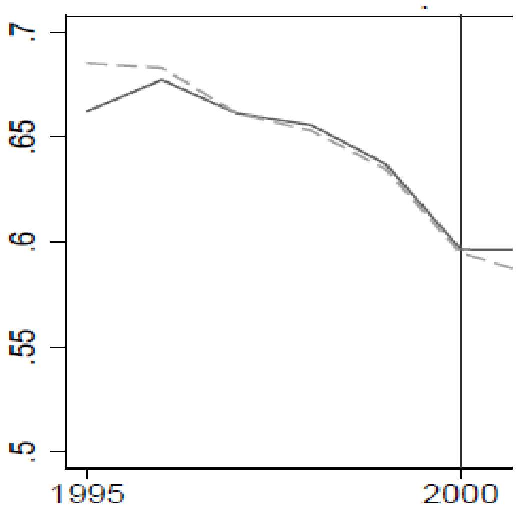
Optimal bandwidth using cross-validation




Non-parametric Synthetic Control Method
.7
.65
.6
.55
.5
1995
2000
2005
2010
Actual
Synthetic
Dependent variable = Domestic Direct Value Added Export (DDVA)
Bandwidth = .4
Kernel = triangular
Treated = ITA



PARAMENTRIC



NON-PARAMENTRIC

Conclusion

- Results show that both methods provide a small pre-treatment prediction error.

- When departing from the beginning of the pre-treatment period, the nonparametric SCM seems to outperform slightly the parametric one.

- I have briefly presented **`npsynth`**, the Stata routine I developed for estimating the nonparametric SCM as proposed in this presentation.