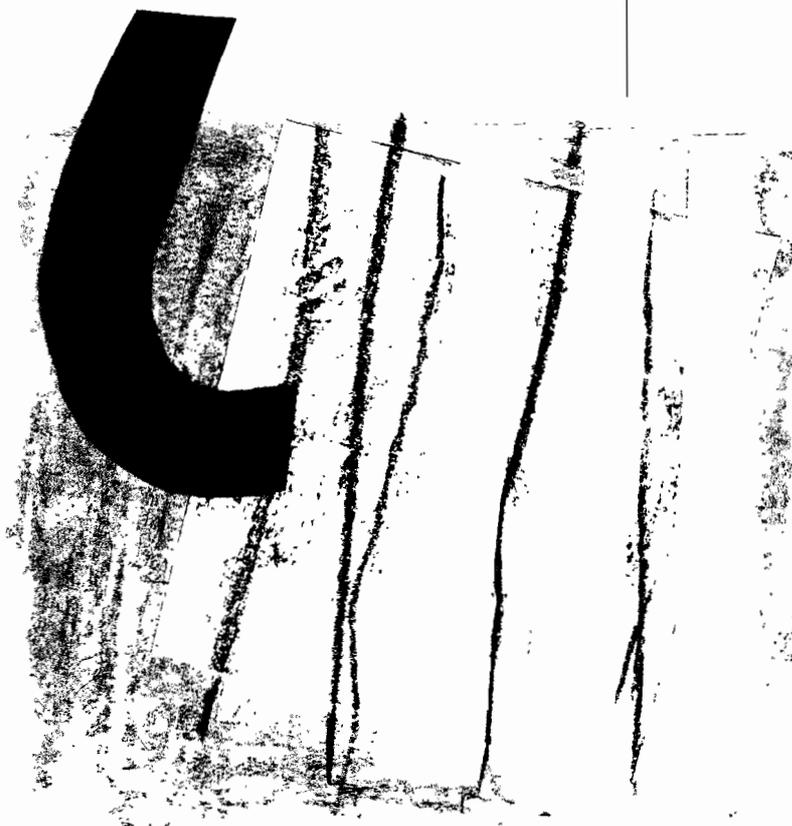


**NOPARAM:  
ESTIMACION FUNCIONAL  
NO-PARAMETRICA.  
UN ACERCAMIENTO DEL SOFTWARE  
CURVDAT AL USUARIO**

Pedro F. Delicado Useros

92 - 06



DOCUMENTOS DE TRABAJO

Documento de Trabajo 92-06  
Marzo 1992

División de Economía  
Universidad Carlos III de Madrid  
Calle Madrid, 126  
28903 Getafe (Madrid)  
Fax: 91-6249849

**NOPARAM**  
Estimación Funcional No-Paramétrica.  
Un acercamiento del Software CURVDAT al usuario

Pedro F. Delicado Useros\*

**Resumen**

---

En los últimos tiempos las técnicas de estimación no-paramétrica de funciones han experimentado un gran desarrollo teórico. Sin embargo, el software asociado a ellas es escaso. La colección CURVDAT de subrutinas FORTRAN es una de las herramientas que permiten hacer estimaciones no-paramétricas unidimensionales. En este documento se presenta NOPARAM, consistente en varios programas FORTRAN que permiten el uso de las subrutinas CURVDAT sin necesidad de programar y ofrecen directamente gráficos de las funciones estimadas.

---

**Palabras clave:**

Estimación no-paramétrica. Densidad. Regresión. Subrutinas y programas FORTRAN. Gráficos.

\* Departamento de Estadística y Econometría, Universidad Carlos III de Madrid.

# 1 Introducción

<sup>1</sup> En los últimos tiempos hemos asistido a un gran desarrollo teórico de las técnicas de estimación no paramétrica de funciones. Estas nuevas técnicas, sin embargo, no han entrado a formar parte del uso común de los investigadores empíricos debido, entre otras razones, a que el Software asociado a ellas es escaso.

Cabe destacar el paquete **XPLORE** como el más completo de los existentes, aunque presenta algunas limitaciones. Mencionaremos también que el paquete **TIMESLAB**, a pesar de no estar especializado en el tema, tiene una buena subrutina de estimación de la densidad unidimensional (no de la regresión). La colección de subrutinas **CURVDAT** <sup>2</sup> constituye una alternativa a estos paquetes. Son ficheros **FORTRAN** que permiten hacer estimaciones no-paramétricas unidimensionales. Las ventajas que presenta **CURVDAT** frente a **XPLORE** son la mayor potencia de cálculo y la capacidad de manejar conjuntos de datos de cualquier tamaño (las limitaciones las presentará el ordenador, no los programas, mientras que algunas versiones del **XPLORE** no manejan más de 300 datos). Por contra, el **XPLORE** abarca un campo mucho más amplio que **CURVDAT** (más técnicas, dimensión mayor que 1, etc.), sus gráficos son envidiables y su uso no requiere ningún conocimiento de programación.

Movidos por el propósito de hacer más accesible el uso de las técnicas no-paramétricas, hemos desarrollado **NOPARAM**: varios programas **FORTRAN** que sirven de apoyo a las subrutinas **CURVDAT**. Evitamos con ellos el que haya que programar para poder estimar no-paraméricamente. Además hemos incluido subrutinas gráficas que permiten representar las estimaciones obtenidas, sin necesidad de usar ningún paquete informático adicional. Hemos respetado al máximo las posibilidades que ofrece **CURVDAT**, principalmente la posibilidad de trabajar con muestras de gran tamaño.

En la sección 2 están descritas sucintamente las subrutinas **CURVDAT**: en la sección 3 presentamos los programas que constituyen **NOPARAM**; la última sección se reserva para exponer detalles técnicos de la instalación de los programas y dar consejos y ejemplos que sirvan de guía al usuario. La teoría necesaria para poder usar estos programas puede encontrarse en [Silverman,86] y [Härdle,90].

## 2 CURVDAT: Versión 90.1

**CURVDAT** es una colección de subrutinas escritas en **FORTRAN77** que realizan estimación no paramétrica de la función de densidad, de la curva de regresión y de las derivadas de ambas. La estimación se lleva a cabo por el "método kernel" (ó método núcleo). Estas subrutinas se centran en la estimación univariante.

Hay dos subrutinas principales: **SMODEN** y **SMOOTH**. La primera se ocupa de estimar la densidad, y la segunda la regresión. Aquí sólo las describiremos brevemente. Una infor-

---

<sup>1</sup>Agradezco al Profesor Antonio Cuevas las facilidades que me dio para realizar este trabajo

<sup>2</sup>**CURVDAT** es propiedad de **STATCOM** (Institut für Statistik und Computing, Walter Köhler, Am Mühlrain 24B, D-6903 Neckargemünd)

mación más detallada puede hallarse en [Curvdat.90.1]. Aspectos comunes a ambas son los siguientes:

**\* Tipo de núcleo**

Los núcleos utilizables son de dos tipos: núcleo óptimo (para el criterio  $L_2$ ) y núcleo de mínima varianza. Recordemos que el núcleo óptimo (núcleo de Epanechnikov, si es de orden 2) se obtiene minimizando la integral del cuadrado del núcleo bajo ciertas restricciones. Para obtener el núcleo de mínima varianza que aquí se propone, el funcional que se minimiza es la integral del cuadrado de la diferencia del núcleo a una constante, bajo las mismas restricciones anteriores. Así, el núcleo óptimo de mínima varianza de orden 2, se obtiene haciendo el núcleo constante en un intervalo centrado en 0: es el núcleo Uniforme.

Dentro de cada uno de estos dos tipos, se pueden elegir núcleos de alto orden. Los órdenes entre los que se puede elegir dependen del orden de la derivada que se quiera estimar y de si el parámetro de suavizado se elige a partir de los datos. En las secciones siguientes explicaremos esa dependencia.

Obsérvese que no es posible usar el núcleo Normal.

**\* Parámetro de suavizado**

La ventana (nombre con el que también se designa el parámetro de suavizado) puede ser fija en todo el soporte de la variable  $X$ , o variar a lo largo de éste. Hay una diferencia entre la forma en que varía la ventana en estas subrutinas y cómo lo hace en los núcleos Adaptativos (ver [Silverman,86]). Mientras que en los núcleos adaptativos se fijaba una ventana distinta para cada una de las observaciones de la muestra, lo que se hace en CURVDAT es fijar una ventana en cada punto en que se estima la densidad o la regresión.

Estas ventanas pueden ser dadas de antemano por el usuario o elegidas automáticamente a partir de los datos. En este último caso, las subrutinas eligen las ventanas que minimizan el Error Cuadrático Medio Integrado (MISE), usando una fórmula asintótica con refinamientos iterativos. En caso de elección automática y ventanas variables, éstas son proporcionales a  $f(t)^{-\alpha}$ , donde  $f(t)$  es la densidad estimada en el punto  $t$ . Esta expresión es lo suficientemente general como para englobar métodos equivalentes al de "K-Puntos más Próximos" (si  $\alpha = 1$ ) o al "método de los splines" en regresión (si  $\alpha = .25$ ).

**\* Tratamiento de los extremos del soporte**

Hay cuatro tratamientos distintos que se pueden aplicar cuando la densidad que se trata de estimar (o la del regresor, en estimación de la regresión) tiene soporte compacto. Surgen de combinar las dos siguientes medidas para corregir la estimación en puntos cercanos a los extremos del soporte:

- reducir o no la ventana en los extremos
- truncar adecuadamente el núcleo cerca de los extremos, o sólo hacer una aproximación que consiste en ponderar los núcleos de forma distinta en esos puntos.

Señalemos una diferencia importante entre SMODEN y SMOOTH: es posible estimar la regresión en cualquier vector cuyo rango esté contenido en el de la variable independiente de las observaciones, pero sólo se puede estimar la densidad en un vector cuyos puntos sean equidistantes.

### 3 Programas de apoyo

A continuación pasamos a describir los programas que hemos desarrollado para complementar las subrutinas mencionadas anteriormente.

#### 3.1 NOPARAM.BAT

Es un fichero por lotes que, en combinación con otros ficheros BAT, ofrece al usuario un "menú" con una serie de opciones que facilitan el uso de los programas de estimación. Se activa tecleando NOPARAM.

La opción **5. Información** muestra al usuario una lista de los ficheros que se suministran, y una breve explicación de cada uno de ellos. Al elegir esta opción se ofrece por pantalla el contenido del fichero LEEME.TXT. En el *APENDICE 3* se facilita un listado de este fichero.

La opción **6. Editor de textos** permite el uso de un editor sin necesidad de salir al sistema operativo. Es posible editar directamente un fichero desde el menú, sin más que escribir el nombre de este fichero a continuación del número correspondiente a la opción del editor de textos (i.e., tecleando: **6 fichero.txt**). Para que el editor usado sea el preferido por el usuario, éste debe editar el fichero **6.BAT** y escribir el nombre de ese editor en la línea que allí se le indica.

#### 3.2 ESTDEN.FOR

Este programa realiza la estimación de la función de densidad, por lo que se relaciona de forma directa con la subrutina SMODEN. El programa realiza tres tareas fundamentales:

- i) Lee el fichero ESTDEN.CTR, donde están registrados los parámetros (o controles) con los que se llamará después a SMODEN, así como los nombres de los ficheros de entrada y salida de datos.
- ii) Llama a SMODEN y escribe las estimaciones obtenidas en los ficheros correspondientes.
- iii) Representa gráficamente la función de densidad estimada.

El hecho de que los parámetros se lean de un fichero de texto permite que, si sólo se quiere modificar un número mínimo de parámetros, esto se pueda hacer de forma más rápida que si la lectura se hiciese por pantalla (basta con editar el fichero ESTDEN.CTR, modificar lo

imprescindible y ejecutar de nuevo el programa). Además, no hay que compilar el programa cada vez que se desea realizar una nueva estimación.

Este programa permite hacer varias estimaciones de densidades en una sola ejecución, pudiendo usar datos y parámetros distintos en distintas estimaciones.

### 3.2.1 Fichero ESTDEN.CTR

El fichero de controles ESTDEN.CTR debe estar estructurado de la misma forma que lo está el fichero que aparece en el *APENDICE 1*. Recomendamos que el usuario cree en el futuro los ficheros ESTDEN.CTR a partir de uno anterior, respetando cuidadosamente los formatos. La estructura de uno de estos ficheros es la siguiente:

Línea 1 Formato: (i3)

Variable: **numest**

Es el número de estimaciones distintas que se van a realizar. Es un entero menor que 1000.

A partir de esa línea hay un grupo de 21 líneas que se repiten **numest** veces, una por cada estimación que se va a realizar:

Línea 2 Línea de comentarios.

Línea 3 Formato: (a10)

Variable: **finput**

Es el nombre del fichero donde se encuentra el vector de observaciones. En 10 espacios debe estar escrito el nombre y la extensión del fichero, por lo que se recomienda que el nombre ocupe 6 espacios, la extensión 3 y vayan separados por un punto.

Línea 4 Formato: (a10)

Variable: **foutput**

Es el nombre del fichero de salida general.

Línea 5 Formato: (i1,1x,a10,1x,a8)

Variables: **isalzf**, **fzf**, **formzf**

**isalzf** debe valer 0 ó 1. Si vale 1, se escribirán las estimaciones de la densidad en un fichero de datos (además de en **foutput**). Este fichero será **fzf**, y tendrá formato **formzf** en cada línea. El formato debe permitir escribir dos números reales en doble precisión en cada línea: el primero de ellos es el punto en el que se estima la densidad y el segundo es la estimación. Si **isalzf** es 0, no se crea el fichero **fzf**.

Línea 6 Formato: (i5)

Variable: **m**

Es el número de puntos equidistantes en los que se estimará la densidad. A estos puntos les llamaremos *mallá de salida*.

Línea 7 Formato: (g14.0)

Variable: **z1**

Es el extremo inferior de la mallá de salida.

Línea 8 Formato: (g14.0)

Variable: **zm**

Es el extremo superior de la mallá de salida.

Línea 9 Formato: (i1)

Variable: **NKE**

Indica el tipo de núcleo que se utiliza. Si **NKE** vale 1, se usa el núcleo de mínima varianzá, y si vale 2, el núcleo óptimo.

Línea 10 Formato: (i1)

Variable: **ISMO**

Si la ventána es elegida automáticamente a partir de los datos, **ISMO** vale 0, y si es el usuario el que la fija **ISMO** vale 1.

Línea 11 Formato: (i1)

Variable: **NY**

Es el parámetro que dice si la ventána es fija en todo el soporte de la densidad (**NY**= 0) o si es variable de un punto de la mallá de salida a otro (**NY**= 1).

Línea 12 Formato: (g14.0)

Variable: **B**

Si la ventána es fija y dada por el usuario, **B** es este parámetro de suavizado. Si es variable y dada por el usuario, entonces **B** es la media de las ventánas que se van a usar. Si **ISMO** es 0, entonces **B** es un parámetro de salida, que será la ventána óptima elegida por el programa (si la ventána es fija) o la media de las ventánas elegidas (si son variables).

Línea 13 Formato: (g14.0)

Variable: **EPS**

Si la ventána es elegida por el programa, se necesita un criterio de convergencia para el algoritmo que la busca. Este criterio viene dado por **EPS**. Como orientación, digamos que un valor apropiado para **EPS** es  $0.005(zm-z1)$ .

Línea 14 Formato: (g14.0)

Variable: ALPHA

Si las ventanas son variables y elegidas automáticamente, estas ventanas son proporcionales a  $f(z)^{-\alpha}$ , donde  $z$  son los puntos en los que se estima la densidad y  $f$  es una estimación piloto de esa densidad. ALPHA es el valor de  $\alpha$ .

Línea 15 Formato: (a10)

Variable: fg

Si las ventanas son variables, estas deben ser leídas de un fichero (respectivamente, escritas en un fichero) si son dadas por el usuario (resp., si son calculadas por el programa). Si  $B > 0$ , este fichero debe contener  $m$  números con media 1: las ventanas en cada uno de los puntos de la malla de salida serán  $B$  veces cada uno de esos números. Si  $B = 0$ , las ventanas son justamente los valores escritos en ese fichero. El nombre de este fichero viene dado por la variable fg.

Línea 16 Formato: (i1)

Variable: NUE

Indica qué derivada de la función de densidad se estima. Debe estar comprendido entre 0 y 4. Si la ventana se elige a partir de los datos, sólo es posible estimar hasta la segunda derivada. Por tanto, hay que elegir NUE entre 0, 1 y 2.

Línea 17 Formato: (i1)

Variable: KORD

Es el orden del núcleo que se va a usar. Debe ser menor o igual que 6. Si el parámetro de suavizado es un dato, debe ser NUE + 2, 4 ó 6. Si por el contrario se elige automáticamente la ventana, sólo puede ser NUE + 2 ó 4.

Línea 18 Formato: (i1)

Variable: nsoporte

Si el soporte de la densidad es toda la recta real, nsoporte vale 0, y en caso contrario vale 1. Si el soporte no es acotado ni inferior ni superiormente, las cuatro líneas siguientes no se leen.

Línea 19 Formato: (g14.0)

Variable: B0(1)

Es el extremo inferior del soporte de la densidad. Si está acotado superiormente pero no inferiormente, B0(1) debe ser un valor negativo y suficientemente grande en valor absoluto como para que pueda ser considerado como  $-\infty$ .

Línea 20 Formato: (g14.0)

Variable: B0(2)

Es el extremo superior del soporte de la densidad. Si está acotado superiormente pero no inferiormente,  $B0(2)$  debe ser un valor positivo y suficientemente grande en valor absoluto como para que pueda ser considerado como  $\infty$ .

Línea 21 Formato: (i1)

Variable: `nbovent`

Esta variable indica si en los extremos del soporte de la densidad la ventana se toma igual que en el interior (`nbovent=0`), o se toma una ventana reducida (`nbovent=1`).

Línea 22 Formato: (i1)

Variable: `nbokern`

Si en los extremos del soporte se toman núcleos truncados adecuadamente, entonces la variable `nbokern` vale 0. En el caso de que sólo se haga una ponderación adecuada en los puntos extremos, `nbokern` vale 1.

Como se puede ver, hay muchas variables que tienen sentido sólo si alguna de las anteriores toma determinados valores. Aunque en algún caso no sea necesario usar una variable, se debe mantener un valor cualquiera en el fichero `ESTDEN.CTR` para evitar problemas en la lectura de este fichero. Así, por ejemplo, aunque hayamos especificado que la densidad tiene como soporte toda la recta real (`nsoporte=0`), deben aparecer las cuatro líneas siguientes con el mismo formato de siempre y con valores arbitrarios (que serán totalmente ignorados por el programa).

### 3.2.2 ESTDEN1.FOR

Su misión es crear un fichero que tenga la estructura de `ESTDEN.CTR`, descrita anteriormente. Para ello, realiza una serie de preguntas al usuario a través de la pantalla. Recomendamos ejecutar este programa al menos una vez, cuando todavía no se esté familiarizado con `ESTDEN.FOR`. Así se tiene una visión de todas las posibilidades de las que se dispone y se crea un modelo de fichero `ESTDEN.CTR` que puede servir para futuras estimaciones sin más que cambiar lo necesario en cualquier editor.

### 3.2.3 Fichero por lotes ESTDENB.BAT

Es un fichero por lotes que necesita como parámetro el nombre (sin extensión) de un fichero con la estructura de `ESTDEN.CTR`, y que como él tenga extensión `CTR`. Su misión es copiar este fichero en `ESTDEN.CTR`, para que los parámetros de una estimación concreta puedan ser leídos por `ESTDEN.FOR`, y ejecutar este programa.

## 3.3 ESTREG.FOR

Es un programa análogo a `ESTDEN.FOR`. Por ello, muchas de las cosas que a continuación diremos son una repetición casi exacta de lo que ya explicamos en la sección anterior. El

fichero del que lee los parámetros este programa es ESTREG.CTR. Su relación con SMOOTH es paralela a la que existe entre ESTDEN.FOR y SMODEN. El programa realiza las siguientes tareas :

- i) Lee el fichero ESTREG.CTR, donde están registrados los parámetros (o controles) con los que se llamará después a SMOOTH, así como los nombres de los ficheros de entrada y salida de datos.
- ii) Llama a SMOOTH y escribe las estimaciones obtenidas en los ficheros correspondientes.
- iii) Representa gráficamente la función de regresión estimada y, opcionalmente, la nube de puntos original.

Este programa permite hacer varias estimaciones de la regresión en una sola ejecución, pudiendo usar datos y parámetros distintos en distintas estimaciones.

### 3.3.1 Fichero ESTREG.CTR

El fichero de controles ESTREG.CTR debe estar estructurado de la misma forma que lo está el fichero que aparece en el *APENDICE 2*. Recomendamos que el usuario cree en el futuro los ficheros ESTREG.CTR a partir de uno anterior, respetando cuidadosamente los formatos. La estructura de uno de estos ficheros es la siguiente:

Línea 1 Formato: (i3)

Variable: **numest**

Es el número de estimaciones distintas que se van a realizar. Es un entero menor que 1000.

A partir de esa línea hay un grupo de 21 líneas que se repiten **numest** veces, una por cada estimación que se va a realizar:

Línea 2 Línea de comentarios.

Línea 3 Formato: (a10)

Variable: **finput**

Es el nombre del fichero donde se encuentran las observaciones. En 10 espacios debe estar escrito el nombre y la extensión del fichero, por lo que se recomienda que el nombre ocupe 6 espacios, la extensión 3 y vayan separados por un punto. Este fichero debe contener los datos del regresor de la variable dependiente, y opcionalmente los datos de la malla de salida (puntos en los que se estimará la curva de regresión).

Línea 4 Formato: (a10)

Variable: **foutput**

Es el nombre del fichero de salida general.

Línea 5 Formato: (i1,1x,a10,1x,a8)

Variables: `isaltty`, `ftty`, `formtty`

`isaltty` debe valer 0 ó 1. Si vale 1, se escribirán las estimaciones de la regresión en un fichero de datos (además de en `foutput`). Este fichero será `ftty`, y tendrá formato `formtty` en cada línea. El formato debe permitir escribir dos números reales en doble precisión en cada línea: el primero de ellos es el punto en el que se estima la regresión y el segundo es la estimación. Si `isaltty` es 0, no se crea el fichero `ftty`.

Línea 6 Formato: (i1)

Variable: `ied`

Esta variable vale 1 si los puntos en los que se estima la regresión son equidistantes, y 0 en otro caso. Las cuatro líneas siguientes son leídas sólo si `ied`= 1. Si `ied`= 0, entonces el fichero `finput` debe contener los puntos en los que se estima el valor de la función de regresión, a continuación de los casos observados.

Línea 7 Formato: (i1)

Variable: `ieq`

Si la malla de salida está formada por puntos equidistantes y además éstos son una parte de la malla de entrada (variable regresora), entonces `ieq` vale 0; en cualquier otro caso `ieq` valdrá 1.

Línea 8 Formato: (i5)

Variable: `m`

Es el número de puntos equidistantes en los que se estimará la regresión. A estos puntos les llamamos *malla de salida*.

Línea 9 Formato: (g14.0)

Variable: `tt1`

Es el extremo inferior de la malla de salida.

Línea 10 Formato: (g14.0)

Variable: `ttm`

Es el extremo superior de la malla de salida.

Línea 11 Formato: (i1)

Variable: `NKE`

Indica el tipo de núcleo que se utiliza. Si `NKE` vale 1, se usa el núcleo de mínima varianza, y si vale 2, el núcleo óptimo.

Línea 12 Formato: (i1)

Variable: `ISMO`

Si la ventana es elegida automáticamente a partir de los datos `ISMO` vale 0, y si es el usuario el que la fija `ISMO` vale 1.

Línea 13 Formato: (i1)

Variable: NY

Es el parámetro que dice si la ventana es fija en todo el soporte del regresor (NY= 0) o si es variable de un punto otro (NY= 1).

Línea 14 Formato: (g14.0)

Variable: B

Si la ventana es fija y dada por el usuario, B es este parámetro de suavizado. Si es variable y dada por el usuario, entonces B es la media de las ventanas que se van a usar. Si ISMO es 0, entonces B es un parámetro de salida, que será la ventana óptima elegida por el programa (si la ventana es fija) o la media de las ventanas elegidas (si son variables).

Línea 15 Formato: (g14.0)

Variable: EPS

Si la ventana es elegida por el programa, se necesita un criterio de convergencia para el algoritmo que la busca. Este criterio viene dado por EPS. Como orientación, digamos que un valor apropiado para EPS es 0.005(ttm-tt1).

Línea 16 Formato: (g14.0)

Variable: ALPHA

Si las ventanas son variables y elegidas automáticamente, éstas ventanas son proporcionales a  $f(tt)^{-\alpha}$ , donde  $tt$  son los puntos donde se estima la regresión y  $f$  es una estimación de la función de densidad del regresor. ALPHA es el valor de  $\alpha$ .

Línea 17 Formato: (a10)

Variable: fg

Si las ventanas son variables, estas deben ser leídas de un fichero (respectivamente, escritas en un fichero) si son dadas por el usuario (resp., si son calculadas por el programa). Si  $B > 0$ , este fichero debe contener  $m$  números con media 1; las ventanas en cada uno de los puntos de la malla de salida serán B veces cada uno de esos números. Si  $B = 0$ , las ventanas son justamente los valores escritos en ese fichero. El nombre de este fichero viene dado por la variable fg.

Línea 18 Formato: (i1)

Variable: NUE

Indica qué derivada de la función de densidad se estima. Debe estar comprendido entre 0 y 4. Si la ventana se elige a partir de los datos, sólo es posible estimar hasta la segunda derivada. Por tanto, hay que elegir NUE entre 0, 1 y 2.

Línea 19 Formato: (i1)

Variable: KORD

Es el orden del núcleo que se va a usar. Debe ser menor o igual que 6. Si el parámetro de suavizado es un dato, debe ser  $NUE + 2$ , 4 ó 6. Si por el contrario la ventana se elige automáticamente,  $KORD$  sólo puede ser  $NUE + 2$  ó 4.

Línea 20 Formato: (i1)

· Variable: **nbovent**

Esta variable indica si en los extremos del soporte del regresor, la ventana se toma igual que en el interior (**nbovent**= 0), o se toma una ventana reducida (**nbovent**= 1).

Línea 21 Formato: (i1)

Variable: **nbokern**

Si en los extremos del soporte se toman núcleos truncados adecuadamente, entonces la variable **nbokern** vale 0. En el caso de que sólo se haga una ponderación adecuada en los puntos extremos, **nbokern** vale 1.

Línea 22 Formato: (i1)

Variable: **igraf**

Esta variable indica si se representará gráficamente solamente la curva de regresión estimada (**igraf**= 1), o si se acompañará el gráfico con la nube de puntos inicial (**igraf**= 2).

Como ocurría en **ESTDEN.CTR**, hay muchas variables que tienen sentido sólo si alguna de las anteriores toma determinados valores. Aunque en algún caso no sea necesario usar una variable, se debe mantener un valor cualquiera en **ESTREG.CTR** para evitar problemas en la lectura de este fichero.

### 3.3.2 ESTREG1.FOR

Su misión es crear un fichero que tenga la estructura de **ESTREG.CTR**, descrita antes. Para ello, realiza una serie de preguntas al usuario a través de la pantalla. Recomendamos ejecutar este programa al menos una vez, cuando todavía no se esté familiarizado con **ESTREG.FOR**. Así se tiene una visión de todas las posibilidades de las que se dispone y se crea un modelo de fichero **ESTREG.CTR** que puede servir para futuras estimaciones sin más que cambiar lo necesario en cualquier editor.

### 3.3.3 Fichero por lotes ESTREGB.BAT

Es un fichero por lotes que necesita como parámetro el nombre (sin extensión) de un fichero con la estructura de **ESTREG.CTR**, y que como él tenga extensión **CTR**. Su misión es copiar este fichero en **ESTREG.CTR**, para que los parámetros de una estimación concreta puedan ser leídos por **ESTREG.FOR**, y ejecutar este programa.

## 4 Algunos detalles técnicos del Software

A continuación vamos a enumerar algunos puntos que ayudarán al lector a instalar adecuadamente los programas que hemos desarrollado, así como a modificarlos levemente para conseguir alguna mejora.

### 4.1 Ficheros de datos

La estructura de los ficheros de datos utilizados por estos programas es siempre la misma, tanto para los ficheros de entrada como los de salida: tienen una primera línea en la que aparece una etiqueta identificativa (que debe ocupar a lo sumo 40 espacios), la segunda línea contiene el número de datos del fichero (escrito en 5 espacios con formato (I5) ) y el formato FORTRAN con el que está escrita cada línea (hay 8 espacios reservados para escribirlo); en las siguientes líneas están escritos los datos (uno en cada línea). Aunque es posible leer varios datos por línea, recomendamos que (si se puede elegir) los ficheros tengan sólo un dato por línea.

El fichero de datos del programa ESTREG.FOR es distinto, ya que en él hay dos bloques de datos (con la estructura anterior): uno para el regresor y otro para la variable independiente. Puede aparecer incluso un tercer bloque: el de los puntos donde se estima la curva de regresión, en el caso de que estos puntos no sean equidistantes.

Los ficheros de salida exclusiva de datos (**fzf** y **ftty**) también son especiales, pues guardan dos datos por línea.

### 4.2 Limitaciones de tamaño de los conjuntos de datos

En todos los programas FORTRAN que trabajan con vectores es necesario dimensionar éstos con una dimensión máxima que no puede ser superada. En los programas ESTDEN.FOR y ESTREG.FOR, se ha necesitado fijar una cota máxima a las dimensiones tanto de los vectores de datos como de aquellos en los que se estiman las funciones. Estos límites no son en absoluto restrictivos, ya que si en algún momento se necesita trabajar con un número de datos superior al permitido, basta con editar el fichero fuente correspondiente, modificar en la sentencia **PARAMETER** los valores de los parámetros-cotas y volver a compilarlo.

A continuación se especifican qué parámetros marcan las dimensiones máximas posibles y en qué valores se encuentran fijados en la actualidad:

**ESTDEN** : **Nmax** Es la dimensión máxima del vector de observaciones. *Valor actual:* 5000

**Mmax** Es el máximo número de puntos en los que se puede estimar la densidad. *Valor actual:* 1001

**ESTREG** : **Nmax** Es la dimensión máxima de los vectores de observaciones. *Valor actual:* 2000

**Nmax3** Es un valor que utiliza el programa internamente. Su valor debe ser siempre mayor o igual que tres veces **Nmax**. *Valor actual:* 6000

**Mmax** Es el máximo número de puntos en los que se puede estimar la curva de regresión.  
*Valor actual:* 1001

### 4.3 Gráficos

Las subrutinas que se emplean para representar gráficamente los resultados de las estimaciones están contenidas en el fichero GXYSCA.FOR. Están desarrolladas en MICROSOFT FORTRAN, V 5.0. Utilizan conexiones con subrutinas escritas en lenguaje C. En general, no pueden ser compiladas con otros compiladores de FORTRAN distintos de éste.

Además de las subrutinas contenidas en GXYSCA.FOR, la subrutina CLS (que aparece en los programas que interactúan con el usuario por medio de la pantalla) también utiliza las librerías gráficas.

Para compilar cualquier programa de los presentados aquí hace falta llamar a la librería GRAPHICS.LIB, una de las que incorpora el compilador MICROSOFT FORTRAN, V 5.0. También es necesario tener los ficheros FGRAPH.FI, FGRAPH.FD y GXYSCA.FOR colocados, bien en el directorio activo en el momento en que se compile algún programa, bien en el directorio que por defecto esté asociado con el comando **INCLUDE** del compilador FORTRAN.

Si se desea compilar algún programa FORTRAN y las opciones gráficas plantean problemas, lo que se ha de hacer es lo siguiente:

- Eliminar las líneas en las que aparezca el comando **INCLUDE** acompañado de alguno de los tres ficheros mencionados en el párrafo anterior: FGRAPH.FI, FGRAPH.FD y GXYSCA.FOR
- Eliminar las líneas en las que se haga una llamada a las subrutina GXY o GXYSCA.
- Borrar la marca **c** de comentarios que acompaña a la primera línea del programa principal, de modo que la sentencia **program** quede activada.
- Hay que modificar la subrutina CLS, si ésta está contenida en el fichero que se quiere compilar: hay que quitar la marca **c** de comentarios a las tres primeras líneas y eliminar las tres siguientes. La presentación por pantalla perderá calidad pero el programa se podrá compilar.

Una vez que los programas se han compilado con las subrutinas gráficas, en cada ejecución éstos necesitan leer el fichero TMSRB.FON. Por defecto, los programas buscan ese fichero en el directorio activo en el momento en que se ejecutan; si no lo encuentran preguntan el directorio en el que lo pueden encontrar. Aunque no se encuentre ese fichero, el programa sigue ejecutándose. TMSRB.FON contiene fuentes de texto necesarias para escribir texto en una pantalla gráfica. Si no se dispone de ese fichero los gráficos son realizados, pero ningún número aparecerá acompañando a los ejes.

## 4.4 Algunos consejos y ejemplos

\* Obsérvese que ESTDEN.FOR lee los controles única y exclusivamente del fichero ESTDEN.CTR. Por lo tanto recomendamos que se guarde ese nombre como comodín, es decir, que se nombre a los ficheros de control de una estimación particular con cualquier otro nombre y, justo en el momento en que se vaya a ejecutar ESTDEN.FOR, se copie ese fichero en ESTDEN.CTR. Recomendación análoga a ésta hacemos para el caso de la estimación de la regresión. Los ficheros BAT que se adjuntan hacen precisamente esta operación.

\* CURVDAT dispone de una buena subrutina de detección de errores en los datos y falta de coherencia en los parámetros; también está diseñado para que, en un gran número de casos, tome valores de parámetros por defecto si los que se le introducen no son los correctos. Al principio de cada fichero de salida se da una lista de los errores que pudo solventar por sí mismo, así como de los que le obligaron a cortar la ejecución. Recomendamos que se mire siempre ese fichero para comprobar que todo a ido bien en nuestra estimación.

\* Se pueden hacer varias estimaciones de los mismos datos en una sola ejecución de los programas: si en dos estimaciones sucesivas se usan los mismos datos, éstos sólo son leídos la primera vez, con lo que se ahorra algo de tiempo. Los ficheros de salida pueden ser los mismos para distintas ejecuciones: se guardan las salidas secuencialmente.

\* El fichero LEEME.TXT tiene una lista de los ficheros que se ofrecen y una rápida descripción de ellos.

\* Si se desea imprimir un gráfico, la forma de hacerlo es mediante un "volcado" de la pantalla a la impresora, pulsando la tecla **Ptr Scr** cuando el gráfico deseado aparezca en pantalla. Este procedimiento puede presentar problemas al usar una impresora *Laser*. Recomendamos usar programas como el **HPSCREEN**, que facilitan el volcado de pantalla en estas impresoras.

\* Recomendamos que la instalación de *NOPARAM* se haga como sigue:

- Crear un subdirectorio exclusivo para *NOPARAM*. Por ejemplo: `c:\[path] noparam`
- Añadir al **path** y al **append** del autoexec el nuevo directorio.
- Guardar los ficheros de datos, de controles y de salidas en otro directorio distinto, en el que nos situaremos cada vez que queramos trabajar con *NOPARAM*.

Así estará siempre clara la separación entre los ficheros del usuario y los que pertenecen a *NOPARAM*.

### \* Un ejemplo

Vamos a reproducir los pasos que deberíamos seguir si quisiésemos hacer una estimación de una curva de regresión.

Tenemos los siguientes datos: estaturas de 33 niños con distintas edades (ver el fichero *EDADES.DAT*). Sea *T* la variable *EDAD* y *X* la variable *ESTATURA*. Queremos estimar no-paramétricamente la curva de regresión de *X* sobre *T*, y queremos hacerlo en los mismos puntos *T* observados.

Debemos empezar por construir un fichero de datos con tres bloques como el que se describió en el apartado *Ficheros de datos*: uno que contenga T, otro X y el tercero que de nuevo contenga a T. Edite el fichero EDADES.DAT y compruebe que su estructura es efectivamente la correcta.

El siguiente paso es ejecutar el programa ESTREG1. A la pregunta de cómo se llamará el fichero de controles respondemos, por ejemplo, EDADES.CTR. Vamos contestando a las preguntas de modo coherente, respetando las relaciones necesarias entre las variables, llamamos EDADES.OUT al fichero de salida, pedimos que se nos dibuje la función estimada y también la nube de puntos (T,X), etc.

Una vez acabada la ejecución de ESTREG1, tenemos el fichero de controles EDADES.CTR con la estructura de ESTREG.CTR. Ahora debemos copiar el primero de estos ficheros en el segundo (comando *copy* del sistema operativo) o llamar al fichero ESTREGB.BAT de la forma siguiente:

#### ESTREGB EDADES

Si no llamamos al fichero BAT, debemos aún ejecutar el programa ESTREG.

Una vez hecho todo esto, tras unos instantes aparecerá en pantalla la gráfica pedida. Para volver al *PROMPT* del sistema operativo, basta con pulsar la tecla **ENTER**.

Recomendamos, no obstante, poner en marcha el fichero por lotes NOPARAM.BAT, que nos ofrece un menú en el que están contempladas todas las operaciones que aquí hemos hecho. Debemos elegir en este caso la opción **4 EDADES**.

Otro consejo: estime la velocidad del crecimiento humano, mediante la primera derivada de esta función de regresión.

#### \* Los ejemplos que acompañan a los programas

En los ficheros DENSID.DAT y REGRES.DAT hay datos preparados para ser procesados con los parámetros que hay en los ficheros DENSID.CTR y REGRES.CTR, respectivamente. Los primeros datos son edades de pacientes en una prueba clínica. El segundo fichero contiene datos para una regresión: el regresor son números pseudoaleatorios Uniforme(0,1), y la variable dependiente es una transformación de la independiente (tendencia lineal más una función sinusoidal) más un ruido Normal.

## Referencias

- [Härdle.90] HÄRDLE. W. (1990) *Applied nonparametric regression*. Oxford Univ. Press., Boston
- [Silverman.86] SILVERMAN. B. W. (1986) *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, London
- [Curvdat.90.1] STATCOM (1990) *Nonparametric smoothing, differentiation and density estimation. Curvdat 90.1 User's manual*. Institut für Statistik und Computing. Walter Köhler. Am Mühlrain 24B. D-6903 Neckargemünd

**APENDICE 1**  
**Contenido del fichero ESTDEN.CTR**

```

1
*** Estimación 1 de 1 ***
densid.dat          numest = num. de est. que se harán
densid.out          finput = fichero de datos
1 mallas.dat (2g14.8) foutput = fichero de salida
100                 isalzf,fzf,formzf= resultados numéricos
13.000000          m = tamaño de la red de salida
63.000000          z(1) = ext. inf. red de salida
2                  z(m) = ext. sup. red de salida
1                  NKE = tipo de kernel
0                  ISMO = ventana automática(0) o no(1)
5.0000000          NY = ventana fija(0) o variable(1)
.100000000E-01     B = ventana
.00000000          EPS = criterio de convergencia
*****.***        ALPHA = par. elecc. vent. autom.
0                  fg = fich. de anchuras de banda
2                  NUE = derivada de f(z) que se estima
0                  KORD = orden del KERNEL
13.000000          nsoporte = toda la recta(0) o no(1)
63.000000          BO(1) = ext. inf. soporte
1                  BO(2) = ext. sup. soporte
0                  nbovent=vent.en ext.igual(0) o reduc(1)
                   nbokern=KNL en ext.acot(0) o ponder(1)

```

**APENDICE 2**  
**Contenido del fichero ESTREG.CTR**

```

1
*** Estimación 1 de 1 ***
regres.dat
regres.out
0 ***** (*****)
1
1
40
.00000000
1.00000000
2
1
1
.30000000
.10000000E-02
.25000000
ventre.dat
0
4
1
0
2
numest = num. de est. que se harán
finput = fichero de datos
foutput = fichero de salida
isaltty,ftty,formtty= resultados numér
ied = ptos. equidist.[1], no equid.[0]
ieq=malla sal.parte malla ent.Sí 0 no
m = tamaño de la red de salida
tt(1) = ext. inf. red de salida
tt(m) = ext. sup. red de salida
NKE=tipo de kernel:min.var (1),ópt.(2)
ISMO = ventana automática(0) o no(1)
NY = ventana fija(0) o variable(1)
B = ventana
EPS = criterio de convergencia
ALPHA = par. elecc. vent. autom.
fg = fich. de anchuras de banda
NUE = derivada de f(z) que se estima
KORD = orden del KERNEL
nbovent=vent.en ext.igual(0) o reduc(1)
nbokern=KNL en ext.acot(0) o ponder(1)
igraf = curva estimada[1], tb. nube[2]

```

**APENDICE 3**  
**Contenido del fichero LEEME.TXT**

NOPARAM : Estimación Funcional No-Paramétrica

LEEME.TXT : Fichero de información acerca de los ficheros  
que debe contener este disco (o subdirectorio)

A continuación se da una relación de los ficheros disponibles y  
se da una breve explicación de ellos.  
Se recomienda leer el manual:

"NOPARAM: Estimación Funcional  
No-Paramétrica. Un acercamiento del software  
CURVDAT usuario"  
Pedro F. Delicado Useros

Tras esta lectura, conviene comenzar usando los programas que ayudan a  
crear los ficheros de control: ESTDEN1 ó ESTREG1 .

LISTA DE FICHEROS

Ficheros \*.BAT

\* Ficheros de los que depende el MENU INICIAL:

NOPARAM.BAT (Activa Menú inicial)  
1.BAT  
2.BAT  
3.BAT  
4.BAT  
5.BAT  
6.BAT

\* Ficheros que llaman a ESTDEN.EXE y ESTREG.EXE y tienen como parámetro el  
nombre (sin extensión) de un fichero de controles:

ESTDENB.BAT  
ESTREG.BAT

\* Ficheros de ayuda para compilar programas FORTRAN que usan las subrutinas  
SMODEN ó SMOOTH y necesitan las subrutinas gráficas:

DENCOMPI.BAT  
REGCOMPI.BAT

### Ficheros \*.CTR

- \* Fichero de control para el ejemplo de estimación de la densidad:

DENSID.CTR  
ESTREG.CTR (fichero comodín)

- \* Fichero de control para el ejemplo de estimación de la regresión:

REGRES.CTR  
ESTDEN.CTR (fichero comodín)

### Ficheros \*.DAT

- \* Fichero de datos para el ejemplo de estimación de la densidad:

DENSID.DAT

- \* Ficheros de datos para el ejemplo de estimación de la regresión:

REGRES.DAT  
EDADES.DAT

### Ficheros \*.OUT

- \* Fichero de salida del ejemplo de estimación de la densidad:

DENSID.OUT

- \* Fichero de datos para el ejemplo de estimación de la regresión:

REGRES.OUT

### Ficheros \*.EXE

- \* Hay 4 ficheros ejecutables:

ESTDEN1.EXE : Crea un fichero de controles para ejecutar ESTDEN

ESTDEN.EXE : Estima la función de densidad, con los parámetros registrados en el fichero ESTDEN.CTR

ESTREG1.EXE : Crea un fichero de controles para ejecutar ESTREG

ESTREG.EXE : Estima la función de regresión, con los parámetros registrados en el fichero ESTREG.CTR

## Ficheros \*.FOR

- \* Ficheros fuente de los programas 4 ejecutables antes mencionados:

ESTDEN1.FOR  
ESTDEN.FOR  
ESTREG1.FOR  
ESTREG.FOR

- \* Ficheros con el listado FORTRAN de las subrutinas SMODEN y SMOOTH originales de CURVDAT 90.1

SMODEN.FOR  
SMOOTH.FOR

- \* Fichero que contiene las subrutinas gráficas

GXYSCA.FOR

## Ficheros \*.OBJ

- \* Ficheros OBJETO creados en la compilación de los ficheros \*.FOR

ESTDEN1.OBJ  
ESTDEN.OBJ  
ESTREG1.OBJ  
ESTREG.OBJ  
SMODEN.OBJ  
SMOOTH.OBJ

## Fichero TMSRB.FON

Es un fichero que contiene las fuentes de texto necesarias para poder escribir cualquier texto en las pantallas gráficas de los programas ESTDEN.EXE y ESTREG.EXE .

Este fichero no es estrictamente necesario: si no existe, los gráficos se siguen creando, pero se pierden las marcas contiguas a los ejes.

## Fichero MENU.PIC

Es un fichero necesario para disponer del MENU inicial.

Otros ficheros:

\* Ficheros del editor de texto

PE2.EXE ; PE2.PRO ; PE2.HLP

\* Fichero de ayuda a la impresión

HPSCREEN.EXE

\* Fichero de ayuda a la edición de comandos del DOS

CED.COM

---