



FACULTADE DE MATEMÁTICAS

Traballo Fin de Grao

# Estimación de la función densidad en grafos lineales

Pablo Manuel Vilas Gilda

2020/2021

UNIVERSIDADE DE SANTIAGO DE COMPOSTELA



GRAO DE MATEMÁTICAS

Traballo Fin de Grao

# Estimación de la función densidad en grafos lineales

Pablo Manuel Vilas Gilda

06/2021

UNIVERSIDADE DE SANTIAGO DE COMPOSTELA



# Trabajo propuesto

<b>Área de Conocimiento:</b> Estadística e Investigación Operativa
<b>Título:</b> Estimación de la función densidad en grafos lineales
<b>Breve descripción del contenido</b>
Comenzaremos introduciendo y analizando las técnicas tipo kernel de estimación de la función densidad, tanto con datos escalares como vectoriales. A continuación nos centraremos en el caso de estar trabajando con grafos lineales, y por último, trabajar con patrones de puntos en grafos lineales en $\mathbb{R}$ .
<b>Recomendaciones</b>
<b>Otras observaciones</b>



# Índice general

<b>Resumen</b>	<b>VIII</b>
<b>Introducción</b>	<b>XI</b>
<b>1. Estimación escalar tipo núcleo de la densidad</b>	<b>1</b>
1.1. Errores de estimación . . . . .	2
1.2. Aproximaciones asintóticas del ECM y ECIM . . . . .	3
1.3. Teoría del núcleo óptimo . . . . .	5
1.4. Núcleos de mayor orden . . . . .	6
1.5. Cómo medir la dificultad de estimar una densidad . . . . .	7
1.6. Modificaciones del estimador tipo núcleo . . . . .	8
1.6.1. Estimadores tipo núcleo locales . . . . .	8
1.6.2. Estimadores tipo núcleo variables . . . . .	8
1.6.3. Estimadores tipo núcleo de transformación . . . . .	9
1.7. Estimación de las derivadas de la densidad . . . . .	9
1.8. Selección de la escala $h$ . . . . .	10
1.8.1. Método de escala normal . . . . .	10
1.8.2. Método de suavizado máximo . . . . .	11
1.8.3. Validación cruzada por mínimos cuadrados . . . . .	12
1.8.4. Validación cruzada insesgada . . . . .	12
1.8.5. Métodos plug-in . . . . .	13
1.8.6. Método plug-in directo . . . . .	14
1.8.7. Método de resolver la ecuación . . . . .	15
1.8.8. Validación cruzada suavizada . . . . .	15
1.9. Comparación de métodos de selección de escala . . . . .	16
1.9.1. VCMC y VCI . . . . .	16
1.9.2. Plug-in y VCS . . . . .	17
1.9.3. Método raíz $n$ -ésima . . . . .	17

1.9.4. En la práctica . . . . .	18
<b>2. Estimación vectorial tipo núcleo de la densidad</b>	<b>19</b>
2.1. Aproximaciones asintóticas del ECIM . . . . .	21
2.2. Elección del núcleo multivariante . . . . .	23
2.3. Elección de la parametrización del suavizado . . . . .	23
2.4. Selección de la matriz escala $H$ . . . . .	25
2.4.1. Validación cruzada por mínimos cuadrados . . . . .	25
2.4.2. Métodos Plug-in . . . . .	25
<b>3. Estimación de la densidad en Grafos Lineales</b>	<b>27</b>
3.1. Funciones núcleo de forma similar . . . . .	30
3.2. Funciones núcleo de divisiones iguales . . . . .	32
3.3. Funciones núcleo de divisiones iguales continuas . . . . .	33
3.4. Estimación del riesgo relativo . . . . .	36
3.4.1. Función intensidad . . . . .	36
3.4.2. Estimación tipo núcleo de la intensidad . . . . .	36
3.4.3. Selección del suavizado para la estimación de la intensidad . . . . .	39
3.4.4. Función riesgo relativo . . . . .	40
3.4.5. Selección del parámetro de suavizado para el riesgo relativo . . . . .	41
<b>4. Grafos lineales en <math>\mathbf{R}</math></b>	<b>43</b>
4.1. Nociones básicas . . . . .	43
4.2. Intensidad en grafos lineales . . . . .	45
4.3. Covariables . . . . .	46
<b>Bibliografía</b>	<b>57</b>





## Resumen

Introducción a la teoría sobre la estimación tipo núcleo de la densidad en el caso escalar y en el vectorial. En ambos casos se explica como utilizar el método, y se analizan los diferentes núcleos que se pueden utilizar así como los diferentes métodos de selección del parámetro de escala.

También se analiza en profundidad la estimación tipo núcleo de la densidad en el caso de los grafos lineales. Se dan diferentes estimadores tipo núcleo y se aborda el tema de no poder tratar el caso del grafo como un problema bidimensional. También se analiza la estimación de la intensidad, ya que en algunos casos simplifica las cosas, y con ella, la estimación del riesgo relativo.

Por ultimo, se dan unas nociones básicas de como trabajar con patrones de puntos en grafos lineales en  $\mathbb{R}$ , entre las cuales está la estimación de la intensidad.

## Abstract

Introduction to the theory of univariate and multivariate kernel density estimation. In both cases, we explain how to use the method, and we analice diferent kernels and diferent bandwidth selectors.

We also provide an in-depth analysis of the kernel density estimation on a network. Different kernel density estimators are given and the issue of not being able to treat the network case as a two-dimensional problem is addressed. We analice the intensity estimation too, since in some cases simplify things, and the relative risk.

Finally, we give basic notions of how to work with point patterns in a linear network in  $\mathbb{R}$ , among which is the intensity estimation.



# Introducción

Las técnicas de estimación sirven para encontrar alguna estructura en conjuntos de datos que no están sujetos a modelos conocidos. En este estudio nos vamos a centrar en la estimación de la función densidad por medio de los estimadores tipo núcleo, los llamados *kernel smoothers*.

El estimador no paramétrico de la densidad más antiguo es el histograma. Se crea dividiendo la recta real en intervalos del mismo tamaño, y las alturas de cada intervalo son la proporción de la muestra contenida en cada intervalo dividida por el ancho del intervalo. Por lo tanto, siendo  $X_1, \dots, X_n$  una muestra aleatoria simple (m.a.s.), la altura del histograma para cada punto de la recta real viene dada por:

$$f_H(x; b) = \frac{N^{\circ} \text{ de observaciones en el intervalo que contiene a } x}{nb}$$

donde

$n$  : tamaño de la muestra

$b$  : ancho de los intervalos

Por lo tanto, hay que hacer dos importantes elecciones: donde colocar los bordes de los intervalos y escoger el ancho  $b$ , el cual será análogo a la escala en los estimadores tipo núcleo. (*Smoothing parameter*).

Existen ciertos problemas y dificultades con los histogramas que estimulan el uso de los estimadores tipo núcleo, como que la mayoría de densidades no son funciones escalonadas, la extensión a histogramas multivariados es algo complicada, a veces no muestran los datos de forma eficiente, y el principal problema, la colocación de los bordes de los intervalos, el cual podría tener una solución lógica con los *histogramas medios desplazados*. Con todo esto, los histogramas se siguen usando por su simplicidad.

A continuación vamos a introducir y estudiar unos estimadores de curvas no paramétricas que llamaremos estimadores tipo núcleo. Estos estimadores tienen la ventaja de ser muy intuitivos y relativamente fáciles de analizar matemáticamente. Estos estimadores fueron

introducidos por Fix y Hodges en el 1951 y aparecen también en documentos de Rosenblatt(1956) y Parzen (1962). Sin embargo, es un campo donde quedan muchas cuestiones por resolver.

Pese a ello, los diferentes métodos de estimación tipo núcleo han sido un campo en continuo desarrollo, hasta la actualidad, donde aun se siguen estudiando innumerables situaciones. Un conjunto de esas situaciones son las que se pueden analizar como un conjunto de puntos en un grafo, y la estimación de la densidad en estos casos puede ser muy útil para sacar conclusiones en muchos campos diferentes. Aún hoy se sigue investigando en estos casos para poder estimar mejor los patrones de puntos y su densidad. Respecto al aspecto de computación, se ha avanzado mucho con el avance de las tecnologías, el acceso cada vez más fácil a los datos y el desarrollo y mejora de las herramientas computacionales. Aún así, se sigue investigando en la mejora de los tiempos de computación.

# Capítulo 1

## Estimación escalar tipo núcleo de la densidad

La estimación no paramétrica de la densidad es una herramienta muy útil para mostrar la estructura de un conjunto de datos que no se adapta a un modelo paramétrico conocido. Queremos un estimador de la densidad que no asuma que la densidad tiene una particular forma funcional. El estimador escalar tipo núcleo de la densidad es el más sencillo entre otros estimadores tipo núcleo.

**Definición 1.1.** El *Estimador escalar tipo núcleo de la densidad* es:

$$\hat{f}(x; h) = (nh)^{-1} \sum_{i=1}^n K[(x - X_i)/h]$$

siendo

$$\begin{aligned} K(\cdot) &: \text{Función núcleo (kernel), tal que } \int K(x)dx = 1 \\ h &: \text{Parámetro de escala, tal que } h \in \mathbb{R}, h > 0 \end{aligned}$$

No es razonable que todas los  $X_i$  del intervalo  $(x - h, x + h)$  cuenten lo mismo a la hora de estimar la densidad en  $x$ . Este estimador corrige eso asignando un peso a cada dato en función de la proximidad a  $x$ . Otra notación para este estimador es:

$$\hat{f}(x; h) = n^{-1} \sum_{i=1}^n K_h(x - X_i) \tag{1.1}$$

siendo

$$\begin{aligned} K_h(\cdot) &= h^{-1}K(\cdot/h) \\ &(\text{K}_h \text{ densidad de } hx_k \text{ siendo } x_k \text{ variable con densidad } K) \end{aligned}$$

Se suele escoger como núcleo  $K$  una densidad de una distribución unimodal y simétrica respecto al 0. Con esto conseguimos que  $\hat{f}(x; h)$  sea una función densidad también. Por tanto, las elecciones más importantes para este estimador son las de la forma del núcleo y sobre todo, la elección de la escala  $h$ , que analizaremos más adelante.

En resumen, la idea es centrar un núcleo escalado  $K_h$  en cada observación  $X_i$ , y obtenemos el valor del estimador tipo kernel en el punto  $x$ ,  $\hat{f}(x; h)$ , como la media de los  $n$  núcleos escalados en el punto  $x$ . De esta forma se consigue que en los sitios donde haya más observaciones, donde se supone que el valor de la verdadera densidad sea alto, la estimación también tendrá un valor alto. Análogamente, en sitios con pocas observaciones donde se espera un valor bajo de la densidad real, se obtendrá también un valor bajo de la estimación.

## 1.1. Errores de estimación

**Definición 1.2.** Con el *Error cuadrático medio (ECM)* tenemos una herramienta para poder cuantificar la calidad del estimador.

$$ECM(\hat{f}) = Var(\hat{f}) + [Sesgo(\hat{f})]^2$$

Ahora analicemos nuestro caso particular del estimador tipo núcleo de la densidad. Sea  $X$  una variable aleatoria con densidad  $f$ :

$$\begin{aligned} E[\hat{f}(x; h)] &= E[K_h(x - X)] = \int K_h(x - y)f(y)dy = (K_h * f)(x) \Rightarrow \\ &\Rightarrow Sesgo(\hat{f}(x; h)) = E[\hat{f}(x; h)] - f(x) = (K_h * f)(x) - f(x) \end{aligned}$$

$$Var[\hat{f}(x; h)] = E[(\hat{f} - E(\hat{f}))^2] = n^{-1}[(K_h^2 * f)(x) - (K_h * f)^2(x)]$$

Por lo tanto obtenemos el siguiente ECM del estimador tipo núcleo:

$$ECM(\hat{f}) = n^{-1}[(K_h^2 * f)(x) - (K_h * f)^2(x)] + [(K_h * f)(x) - f(x)]^2$$

**Definición 1.3.** Para cuando estimemos  $f$  en todo  $\mathbb{R}$ , tenemos el *Error cuadrático integrado (ECI)*:

$$ECI(\hat{f}(\cdot; h)) = \int [\hat{f}(x; h) - f(x)]^2 dx$$

**Definición 1.4.** A partir del ECI obtenemos el *Error cuadrático integrado medio (ECIM)*:

$$\begin{aligned} ECIM(\hat{f}(\cdot; h)) &= E \left[ \int [\hat{f}(x; h) - f(x)]^2 dx \right] = \int E[[\hat{f}(x; h) - f(x)]^2] dx = \\ \int ECM(\hat{f}(x; h)) dx &= n^{-1} \int [(K_h^2 * f)(x) - (K_h * f)^2(x)] dx + \int [(K_h * f)(x) - f(x)]^2 dx = \\ &= (nh)^{-1} \int K^2(x) dx + (1 - n^{-1}) \int (K_h * f)^2(x) dx - 2 \int (K_h * f)(x) f(x) dx + \int f^2(x) dx \end{aligned}$$

Existen otros criterios como el *Error absoluto integrado medio*,  $EAIM(\hat{f}(\cdot; h)) = E[\int |\hat{f}(x; h) - f(x)| dx]$ . Tiene la ventaja de estar definido siempre que  $\hat{f}(\cdot; h)$  sea una densidad y que no varía respecto a transformaciones monótonas. Sin embargo el análisis de esta cantidad es más complicado que el ECIM, el cual es matemáticamente más simple en comparación.

Observamos que todos estos errores dependen de  $h$  de una forma bastante complicada. Vamos a ver aproximaciones asintóticas de estas expresiones, que dependen de  $h$  de una forma más sencilla y permiten observar el efecto de este parámetro.

## 1.2. Aproximaciones asintóticas del ECM y ECIM

Como ya hemos dicho, las expresiones del ECM y ECIM dependen de la escala  $h$  de una forma complicada. Esto dificulta la interpretación de la influencia de  $h$  en la actuación del estimador tipo núcleo.

Hay una forma que involucra la derivación de aproximaciones que nos llevará a las expresiones de la varianza y el sesgo. Estas aproximaciones tienen expresiones más sencillas que permiten apreciar mejor el papel de  $h$ . También sirven para obtener el índice de convergencia del estimador tipo núcleo de la densidad,  $\hat{f}$ , y para obtener el  $h$  óptimo respecto al ECIM. Para simplificar futuros cálculos, vamos a asumir los siguientes puntos:

1. La densidad real  $f$  tiene segunda derivada  $f''$  continua, integrable al cuadrado y monótona.
2.  $h = h_n$  es una serie de números positivos no aleatoria tal que  $h$  converge a 0 con un índice menor que  $n^{-1}$ , i.e.,  $\lim_{n \rightarrow \infty} h = 0$  y  $\lim_{n \rightarrow \infty} nh = \infty$ .
3. El núcleo  $K$  es una función de densidad acotada, que tiene un cuarto momento finito y simetría respecto al origen.

Por tanto, asumiendo todo esto, tenemos que:

$$E(\hat{f}(x; h)) = \int K(z) f(x - hz) dz$$

$$f(x - hz) = f(x) - hz f'(x) + \frac{1}{2} h^2 z^2 f''(x) + o(h^2)$$

(Desarrollo de Taylor de  $f(x - hz)$ )

Operando, llegamos a que:

$$\begin{aligned} E[\hat{f}(x; h)] &= f(x) + \frac{1}{2}h^2 f''(x) \int K(z)z^2 dz + o(h^2) = f(x) + \frac{1}{2}h^2 f''(x)\mu_2(K) + o(h^2) \Rightarrow \\ \Rightarrow \text{Sesgo}(\hat{f}(x; h)) &= E[\hat{f}(x; h)] - f(x) = \frac{1}{2}h^2 f''(x)\mu_2(K) + o(h^2) \end{aligned} \quad (1.2)$$

Observamos que el Sesgo del estimador es de orden  $h^2$ , por lo que  $\hat{f}(x; h)$  es asintóticamente insesgado. Además, donde la curvatura de  $f$  es grande, el sesgo será mayor (por  $f''(x)$ ). Por otro lado:

$$\begin{aligned} \text{Var}[\hat{f}(x; h)] &= \dots = (nh)^{-1} \int K(z)^2 dz f(x) + o((nh)^{-1}) = \\ &= (nh)^{-1} R(K) f(x) + o((nh)^{-1}) \xrightarrow{n \rightarrow \infty} 0 \end{aligned}$$

Juntando las expresiones del Sesgo y la Varianza del estimador, obtenemos las siguientes fórmulas del ECM:

$$ECM(\hat{f}(x; h)) = (nh)^{-1} R(K) f(x) + \frac{1}{4}h^4 \mu_2(K)^2 f''(x)^2 + o[(nh)^{-1} + h^4]$$

Integrando, obtenemos el ECIM:

$$ECIM(\hat{f}(\cdot; h)) = (nh)^{-1} R(K) + \frac{1}{4}h^4 \mu_2(K)^2 R(f'') + o[(nh)^{-1} + h^4] \quad (1.3)$$

Observando esta última expresión, vemos que cuando  $n$  aumenta, cada componente del ECIM disminuye. Se conoce que existe una compensación entre la varianza y el sesgo. Si  $h$  es muy pequeño, la forma de  $\hat{f}$  será con muchos picos, es decir,  $\hat{f}$  será muy variable pero con un sesgo pequeño. Si aumentamos  $h$ , habrá un sesgo más grande porque los rasgos de  $\hat{f}$  son suavizados, pero una varianza más pequeña ya que los datos de la muestra son más ignorados.

De la expresión (1.3) obtenemos un error que nos llevará a un  $h$  óptimo:

**Definición 1.5.** El *Error cuadrático integrado medio asintótico (ECIMA)* es:

$$ECIMA(\hat{f}(\cdot; h)) = (nh)^{-1} R(K) + \frac{1}{4}h^4 \mu_2(K)^2 R(f'') \quad (1.4)$$

A continuación derivamos (1.4) respecto a  $h$  e igualamos a 0 para obtener el  $h$  óptimo respecto el ECIMA:

$$h_{ECIMA} = \left[ \frac{R(K)}{\mu_2(K)^2 R(f'') n} \right]^{\frac{1}{5}} \quad (1.5)$$

Observamos que esta expresión depende de  $K$  y  $n$ , los cuales son conocidos, y además es inversamente proporcional a  $R(f'')^{\frac{1}{5}}$ , por lo que cuando estemos en el caso de una densidad

$f$  con curvatura pequeña,  $R(f'')$  será pequeña, y por lo tanto se usará un  $h$  grande. En el caso de que la curvatura de la densidad real sea grande, un  $h$  pequeña será más óptima. Sin embargo, la expresión (1.5) no sirve en la práctica cuando  $R(f'')$  es desconocida. Veremos más adelante, diferentes métodos para escoger  $h$ .

Por último en este apartado, analicemos el orden de convergencia del ECIM:

Si sustituimos (1.5) en (1.4) y operamos, llegamos a:

$$\inf_{h>0} ECIMA(\hat{f}(\cdot; h)) = \frac{5}{4} [\mu_2(K)^2 R(f'')]^{\frac{1}{5}} R(K)^{\frac{4}{5}} n^{-\frac{4}{5}}$$

Además,  $\inf_{h>0} ECIMA(\hat{f}(\cdot; h))$  es asintóticamente equivalente a  $\inf_{h>0} ECIM(\hat{f}(\cdot; h))$ , por lo que sabemos que el mejor orden de convergencia del ECIM es de  $n^{-\frac{4}{5}}$ .

Se puede comprobar que el orden de convergencia del ECIM del histograma es  $O(n^{-\frac{2}{3}})$ , lo que muestra la ineficiencia del histograma frente al estimador tipo núcleo.

### 1.3. Teoría del núcleo óptimo

En esta sección vamos a ver el efecto que tiene la forma de la función núcleo  $K$ . Como ya hemos dicho,  $K$  se suele escoger simétrica y unimodal. Pero hay muchas funciones con esas características, entonces, ¿cuál es la mejor forma de  $K$  para elegir?.

Partiendo de la fórmula (??) del  $ECIMA(\hat{f})$ , no es sencillo optimizarla respecto a  $K$  ya que la escala de  $K$  está relacionada con  $h$ . Pero ya que la escala de  $K$  se puede escoger de forma arbitraria, podemos “reescalarlo“ de la siguiente forma:

$$K_\delta(\cdot) = \frac{K(\cdot/\delta)}{\delta}$$

**Definición 1.6.** Se le llama *Núcleo canónico* al núcleo

$$K^c = K_{\delta_0} ; \delta_0 = \left[ \frac{R(K)}{\mu_2(K)^2} \right]^{\frac{1}{5}}$$

$K^c$  es el único núcleo del conjunto de núcleos “reescalados“ que permite separar  $k$  y  $h$  en el ECIMA.

$$ECIMA(\hat{f}) = \frac{\underbrace{C(K_{\delta_0})}_{[R(K)^4 \mu_2(K)^2]^{\frac{1}{5}}}}{[(nh)^{-1} + \frac{1}{4}h^4 R(f'')]}$$

Por lo tanto, hay que escoger  $K$  para minimizar  $C(K_{\delta_0})$ . Pero  $C(K)$  es invariante respecto a los  $\delta$ , por lo que el  $K$  óptimo es el que minimiza  $C(K)$  sujeto a :

$$\int K(x)dx = 1, \int xK(x)dx = 0, \int x^2K(x)dx = a^2 < \infty \text{ y } K(x) \geq 0 \forall x.$$

Se puede demostrar que ese  $K$  es:

$$K^a(x) = \frac{\frac{3}{4}(1 - \frac{x^2}{5a^2})}{(5^{\frac{1}{2}}a)} 1_{\{|x| < 5^{\frac{1}{2}}a\}}$$

Cuya versión más sencilla es el *Núcleo de Epanechnikov* ( $a^2 = \frac{1}{5}$ ):

$$K^*(x) = \frac{3}{4}(1 - x^2)1_{\{|x| < 1\}}$$

**Definición 1.7.** La eficiencia de un núcleo  $K$  respecto a  $K^*$  representa el ratio de tamaños de muestra necesarios para obtener el mismo ECIMA mínimo (dada una  $f$ ) usando  $K^*$  y  $K$ , y viene dada por la cantidad:

$$[C(K^*)/C(K)]^{\frac{5}{4}}$$

Es decir, si  $K$  tiene una eficiencia de 0.95, significa que la estimación óptima de la densidad con el núcleo  $K^*$  puede alcanzar el mismo ECIMA mínimo usando el 95 % de los datos que con el núcleo  $K$ .

Una familia de núcleos con diferentes eficiencias sería:

$$K(x; p) = [2^{2p+1}B(p+1, p+1)]^{-1}(1 - x^2)^p 1_{\{|x| < 1\}}$$

siendo  $B(\cdot, \cdot)$  la función beta.

Cabe destacar que la eficiencia es muy parecida en densidades unimodales. Además la elección del núcleo podría ser por otros motivos como la eficiencia computacional.

## 1.4. Núcleos de mayor orden

Anteriormente hemos visto que el mejor orden de convergencia del estimador tipo kernel es de  $n^{-\frac{4}{5}}$ . Se pueden obtener mejores ordenes quitando la restricción de que el núcleo sea una función de densidad. Si  $K$  no es una densidad, podemos construir un  $K$  con  $\mu_2(K) = 0$ , lo que reduciría el sesgo del estimador por la fórmula (1.2), y conseguiríamos un ECM y ECIM con mejor orden de convergencia de  $n^{-\frac{8}{9}}$ .

**Definición 1.8.**  $K$  es un núcleo de orden  $k$  si:

$$\mu_0(K) = \int K(x)dx = 1, \mu_j(K) = \int x^j K(x)dx = 0 \forall j = 1, \dots, k-1 \text{ y } \mu_k(K) \neq 0$$

Ahora supongamos que  $\hat{f}$  está construido a partir de un núcleo de orden  $k$ ,  $K_{[k]}$ , y que la densidad  $f$  tiene derivada  $k$ -ésima continua e integrable al cuadrado. Entonces se llega a que:

$$\begin{aligned} ECIMA(\hat{f}(\cdot; h)) &= (nh)^{-1}R(K_{[k]}) + h^{2k}[\mu_k(K_{[k]})/k!]^2 R(f^{(k)}) \Rightarrow \\ \Rightarrow h_{ECIMA} &= \left[ \frac{(k!)^2 R(K_{[k]})}{2k\mu_k(K_{[k]})^2 R(f^{(k)})n} \right]^{\frac{1}{2k+1}} \end{aligned}$$

Por tanto:

$$\inf_{h>0} ECIM(\hat{f}(\cdot; h)) \sim \frac{2k+1}{2k} [2k(k!)^{-2} \mu_k(K_{[k]})^2 R(K_{[k]})^{2k} R(f^{(k)}) n^{-2k}]^{\frac{1}{2k+1}}$$

Esto implica que, para densidades suficientemente suaves, el orden de convergencia puede mejorarse tanto como se quiera con los núcleos de mayor orden. Esto no implica que estos núcleos siempre mejoren el error. Además, los núcleos de mayor orden necesariamente toman valores negativos, lo que dificulta mucho promediar y la interpretación.

Destacamos un núcleo de orden infinito ( $\mu_j(K) = 0 \forall j$ ) cuyos ordenes de convergencia del ECIM dependen de la forma de  $f$ :

$$K(x) = \frac{\sin(x)}{\pi x}$$

## 1.5. Cómo medir la dificultad de estimar una densidad

Recordemos que para  $K$  densidad simétrica,  $\inf_{h>0} ECIM(\hat{f}(\cdot; h)) \sim \frac{5}{4} C(K) R(f'')^{\frac{1}{5}} n^{-\frac{4}{5}}$ . La dependencia de  $f$  es a través de  $R(f'') = \int f''(x)^2 dx$ . Esta cantidad nos indica cuan bien puede ser estimada  $f$  cuando  $h$  es escogida en  $\hat{f}$  de forma óptima. Si  $f$  tiene rasgos puntiagudos,  $|f''(x)|$  será más grande, y si  $f$  es más suave,  $R(f'')$  será más pequeño y el problema de estimación será más fácil ( $R(f'') = 0 \Rightarrow f''(x) = 0 \Rightarrow f$  lineal, pero no hay densidades lineales a lo largo de  $\mathbb{R}$ ). La densidad uniforme, por ejemplo, es una buena candidata a estimar, pero hay problemas en los extremos del intervalo. Vamos a asumir que las densidades  $f$  tienen segunda derivada continua e integrable al cuadrado.

**Definición 1.9.** Siendo  $\sigma(f)$  la desviación típica muestral de una densidad  $f$ , definimos la medida de dificultad (invariante respecto a la escala) para la estimación tipo núcleo de una densidad  $f$  como:

$$D(f) = [\sigma(f)^5 R(f'')]^{\frac{1}{4}}$$

$D(f)$  es mínima cuando  $f^*(x) = \frac{35}{32} (1-x^2)^3 1_{|x|<1}$ , es decir, la densidad beta(4,4) es la más fácil de estimar.

**Definición 1.10.** El valor de dificultad de estimar  $f$  respecto a estimar  $f^*$  es  $\frac{D(f^*)}{D(f)}$ . Si este valor es grande ( $\approx 1$ ), es tan difícil estimar  $f$  como estimar la más fácil,  $f^*$ . Si el valor es pequeño, es mucho más difícil estimar  $f$  que  $f^*$ .

Destacamos que el grado de dificultad de estimación aumenta con la asimetría, curtosis y multimodalidad.

## 1.6. Modificaciones del estimador tipo núcleo

### 1.6.1. Estimadores tipo núcleo locales

$$\hat{f}_L(x; h(x)) = [nh(x)]^{-1} \sum_{i=1}^n K((x - X_i)/h(x))$$

La técnica en la que se basa este estimador es en utilizar un estimador tipo núcleo básico en cada punto. Sin embargo, una consecuencia de dejar que  $h$  dependa de  $x$  es que  $\hat{f}_L(\cdot; h(\cdot))$  no es una función de densidad propiamente dicha. Análogo a capítulos anteriores, del ECM sacamos que (siempre que  $f''(x) \neq 0$ ):

$$\begin{aligned} h_{ECMA}(x) &= \left[ \frac{R(x)f(x)}{\mu_2(K)^2 f''(x)^2 n} \right]^{\frac{1}{5}} \Rightarrow \\ &\Rightarrow ECMA(\hat{f}_L(\cdot; h(\cdot))) = \frac{5}{4} [\mu_2(K)^2 R(K)^4]^{\frac{1}{5}} R((f^2 f'')^{\frac{1}{5}}) n^{-\frac{4}{5}} \end{aligned}$$

Observamos varias cosas:

1.  $R((f^2 f'')^{\frac{1}{5}}) \leq R(f'')^{\frac{1}{5}} \forall f$ , por lo que siempre habrá una mejora del error.
2. El orden de convergencia es de  $n^{-\frac{4}{5}}$ , por lo que no hay mejora respecto al estimador básico  $\hat{f}(\cdot; h)$ .
3. En la práctica,  $\hat{f}_L(x; h(x))$  requiere especificar todas las funciones  $h(x)$ , para lo que existen diferentes métodos.

### 1.6.2. Estimadores tipo núcleo variables

$$\hat{f}_V(x; \alpha) = n^{-1} \sum_{i=1}^n [\alpha(X_i)]^{-1} K((x - X_i)/\alpha(X_i))$$

En estos estimadores la  $h$  se sustituye por  $n$  valores  $\alpha(X_i), i = 1, \dots, n$  en lugar de  $h(x)$ . La idea es que el núcleo centrado en  $X_i$  tenga asociado su propio parámetro de escala  $\alpha(X_i)$ . Esto permite usar diferentes grados de suavización dependiendo de donde esté  $X_i$  respecto a los otros datos. La idea con esto es suavizar más donde los datos estén más esparcidos que en los que están más juntos. Además en este caso tenemos que si  $K$  es densidad de probabilidad, entonces  $\hat{f}_V$  también lo es.

Una posible elección de parámetro de escala sería por ejemplo  $\alpha(X_i) = hf(X_i)^{-1/2}$ , con el que se obtiene un sesgo de orden  $h^4$  con  $f$  suficientemente diferenciable.

$$ECMA(\hat{f}_V(x; \alpha)) = (nh)^{-1} f(x)^{3/2} \left[ \frac{6R(K) + R(xK')}{4} \right] + \frac{1}{576} h^8 \mu_4(K)^2 \left( \frac{1}{f} \right)^{(IV)}(x)^2$$

Se obtiene que el  $h$  óptimo tiene orden  $n^{-\frac{1}{9}}$  por lo que el ECM óptimo mejora su orden de  $n^{-\frac{4}{5}}$  a  $n^{-\frac{8}{9}}$ .

### 1.6.3. Estimadores tipo núcleo de transformación

Si una m.a.s.  $X_1, \dots, X_n$  tiene una densidad  $f$  difícil de estimar, una posibilidad es aplicar una transformación a los datos para obtener una muestra  $Y_1, \dots, Y_n$  con una densidad  $g$  mucho más fácil de estimar con el estimador tipo núcleo básico. Después desharamos la transformación en  $\hat{g}$  para obtener  $\hat{f}$ .

Supongamos  $Y_i = t(X_i)$ , con  $t$  función diferenciable, creciente y definida en el soporte de  $f$ . Se sabe que  $f(x) = g(t(x)) \cdot t'(x)$ , por lo que el estimador tipo núcleo de transformación se obtiene reemplazando  $g$  por  $\hat{g}$ , estimador tipo núcleo basado en  $Y_1, \dots, Y_n$ .

$$\begin{aligned} \hat{f}_T(x; h, t) &= n^{-1} \sum_{i=1}^n K_h(t(x) - t(X_i)) t'(x) && \underbrace{\Rightarrow}_{\text{Teorema del Valor Medio}} \\ \Rightarrow \hat{f}_T(x; h, t) &= n^{-1} \sum_{i=1}^n [t'(x)/h] K_h(t'(\xi_i)(x - X_i)/h), \quad \xi_i \in (x, X_i) \text{ ó } \xi_i \in (X_i, x) \end{aligned}$$

La mejor elección de la transformación  $t$  tiene que ver sobre todo con la forma de  $f$ . Por ejemplo, si  $f$  es una densidad unimodal asimétrica, entonces  $t$  debería de ser convexa en el soporte de  $f$  para reducir la asimetría de  $f$ . Si por ejemplo  $f$  es simétrica pero con alta curtosis, entonces  $t$  debería de ser cóncava a la izquierda y convexa a la derecha del centro de simetría de  $f$ . Existen muchas más opciones para escoger  $t$ .

## 1.7. Estimación de las derivadas de la densidad

La primera y segunda derivada tienen interés para el cálculo de la pendiente y curvatura de  $f$ , y también para otras funciones importantes como  $-\frac{f'}{f}$ . Además existen métodos para la selección de la escala  $h$  que requieren estimaciones de las derivadas de la densidad. También se requieren estas estimaciones para el cálculo de extremos y puntos de inflexión.

**Definición 1.11.** Siendo  $K$  suficientemente diferenciable, el estimador tipo núcleo de la derivada  $r$ -ésima de  $f$  es:

$$\hat{f}^{(r)}(x; h) = n^{-1} h^{-r-1} \sum_{i=1}^n K^{(r)}((x - X_i)/h)$$

Por lo que:

$$ECM(\hat{f}^{(r)}(x; h)) = (nh^{2r+1})^{-1} R(K^{(r)})f(x) + \frac{1}{4}h^4 \mu_2(K)^2 f^{(r+2)}(x) + o((nh^{2r+1})^{-1} + h^4)$$

Desarrollando los cálculos se llega a que el  $h$  óptimo es de orden  $n^{\frac{-1}{2r+5}}$ , y que el orden de los ECM y ECIM óptimos es de  $n^{\frac{-4}{2r+5}}$ . Por lo tanto, el orden de convergencia se vuelve más lento con el aumento de  $r$ , reflejando el aumento de la dificultad al estimar derivadas de mayor orden.

## 1.8. Selección de la escala $h$

Este tema es bastante actual y por lo que bastante inestable, con nuevos métodos y dificultades sin resolver. Hay muchas situaciones donde es satisfactorio escoger el parámetro de escala  $h$  subjetivamente a ojo. Por ejemplo, comenzando con un  $h$  grande e ir reduciendo el suavizado hasta que las fluctuaciones que son más “aleatorias” que “estructurales” comiencen a aparecer. Pero esto es viable cuando hay alguna certeza de la estructura de los datos. Suele ser más beneficioso tener la “escala” automáticamente seleccionada de los datos, para ahorrar tiempo y por falta de conocimiento de la estructura de los datos.

**Definición 1.12.** Llamamos método de selección de  $h$  a los métodos que usan los datos  $X_1, \dots, X_n$  para producir un  $\hat{h}$ . Diferenciamos algunos grupos de métodos de selección:

1. **Métodos rápidos y sencillos:** Fórmulas fácilmente computables que buscan un  $h$  razonable para muchas situaciones, pero sin garantías matemáticas de que esté cerca del óptimo. Son métodos motivados por la necesidad de tener estimaciones del núcleo rápidas y automáticamente generadas en algoritmos con muchos pasos de estimación de curvas.
2. **Métodos Hi-tech:** Involucran más argumentos matemáticos y más esfuerzo computacional. Métodos motivados por minimizar el ECIM.
3. Existen muchos otros métodos de aproximación con un análisis más complicado.

Para lo que sigue en esta sección, vamos a asumir que  $\hat{f}(\cdot; h)$  es un estimador tipo núcleo de la densidad basado en una muestra aleatoria  $X_1, \dots, X_n$  con densidad  $f$  suficientemente diferenciable e integrable. Con esto, pasamos a describir dos métodos del primer grupo, y algunos del segundo grupo.

### 1.8.1. Método de escala normal

Este método para la selección de  $h$  implica el uso del parámetro de escala óptimo para el ECIMA de la densidad normal teniendo la misma escala como la estimada para la densidad

subyacente. Anteriormente ya vimos que el  $h$  que minimiza el ECIM asintóticamente es:

$$h_{ECIMA} = \left[ \frac{R(K)}{\mu_2(K)^2 R(f'') n} \right]^{\frac{1}{5}}$$

Si tenemos que  $f$  es la densidad normal con varianza  $\sigma^2$ :

$$h_{ECIMA} = \left[ \frac{8\pi^{1/2} R(K)}{3\mu_2(K)^2 n} \right]^{\frac{1}{5}} \cdot \sigma$$

Por lo tanto:

$$\hat{h}_{EN} = \left[ \frac{8\pi^{1/2} R(K)}{3\mu_2(K)^2 n} \right]^{\frac{1}{5}} \cdot \hat{\sigma}$$

siendo  $\hat{\sigma}$  alguna estimación de  $\sigma$ .

Este método implica una primera suposición de  $h$  que puede ser razonable cuando los datos están cerca de la normal.

### 1.8.2. Método de suavizado máximo

Este método se apoya en el hecho de que existe el límite superior del  $h$  óptimo respecto al ECIMA para la estimación de densidades con un valor fijo de escala. Por ejemplo, se puede demostrar que:

$$h_{ECIMA} = \left[ \frac{243R(K)}{35\mu_2(K)^2 n} \right]^{\frac{1}{5}} \cdot \sigma$$

para todas las densidades con desviación típica  $\sigma$ , y que este límite es alcanzado por la densidad beta(4,4). Por tanto:

$$\hat{h}_{SM} = \left[ \frac{243R(K)}{35\mu_2(K)^2 n} \right]^{\frac{1}{5}} \cdot s$$

siendo  $s$  la desviación típica muestral.

Destacamos que  $\hat{h}_{SM}$  se puede basar en otras escalas óptimas y sus límites superiores.

Aunque  $\hat{h}_{SM}$  da una escala muy grande para una estimación óptima de una densidad  $f$ , también nos da un punto de partida para una elección subjetiva del parámetro  $h$ .

Una buena estrategia gráfica sería representar una estimación de  $f$  con el parámetro  $\hat{h}_{SM}$ , y sucesivamente representarla con varios parámetros que fueran fracciones de  $\hat{h}_{SM}$ , para observar que características hay en los datos para varios niveles de alisado.

Observamos que

$$\hat{h}_{EN}/\hat{h}_{SM} \approx 0,93$$

por lo que la densidad normal está cerca de llegar al límite superior del que partimos.

**1.8.3. Validación cruzada por mínimos cuadrados**

Comenzamos desarrollando la fórmula del ECIM:

$$ECIM(\hat{f}(\cdot; h)) = E \left[ \int \hat{f}(x; h)^2 dx \right] - 2E \left[ \int \hat{f}(x; h)f(x)dx \right] + \int f(x)^2 dx$$

Por lo que minimizar  $ECIM(\hat{f})$  es equivalente a minimizar:

$$ECIM(\hat{f}) - \int f(x)^2 dx = E \left[ \int \hat{f}(x; h)^2 dx - 2 \int \hat{f}(x; h)f(x)dx \right]$$

Pero esta cantidad depende de  $f$ , por lo que será desconocida y utilizaremos un estimador insesgado:

$$VCMC(h) = \int \hat{f}(x; h)^2 dx - 2n^{-1} \sum_{i=1}^n \hat{f}_{-i}(X_i, h)$$

con

$$\hat{f}_{-i}(x; h) = (n-1)^{-1} \sum_{i \neq j}^n K_h(x - X_j)$$

estimador de la densidad basado en la muestra sin  $X_i$ .

Por lo tanto, parece razonable escoger como  $\hat{h}_{VCMC}$  el  $h$  que minimice  $VCMC(h)$ .

Puede haber situaciones donde  $VCMC(h)$  tiene varios mínimos locales. Este método no es muy bueno,  $\hat{h}_{VCMC}$  es muy variable, así que continuemos viendo otros métodos Hi-tech.

**1.8.4. Validación cruzada insesgada**

Empezamos basándonos en la fórmula:

$$ECIMA(\hat{f}(\cdot; h)) = (nh)^{-1}R(K) + \frac{1}{4}h^4\mu_2(K)^2R(f'')$$

ya que  $R(f'')$  es desconocido porque depende de  $f$ , utilizamos el siguiente estimador:

$$R(\hat{f}'') = R(\hat{f}''(\cdot; h)) - (nh^5)^{-1}R(K'') = n^{-2} \sum_{i=1}^n \sum_{i \neq j}^n (K_h'' * K_h'')(X_i - X_j)$$

Por lo tanto, en este método tenemos que:

$$VCI(h) = (nh)^{-1}R(K) + \frac{1}{4}h^4\mu_2(K)^2R(\hat{f}'')$$

De forma análoga al método anterior,  $\hat{h}_{VCI}$  será el  $h$  que minimice  $VCI(h)$ .

Destacamos que  $\hat{h}_{VCI}$  es más estable que  $\hat{h}_{VCMC}$ , ya que su varianza asintótica es menor.

Esto conlleva el precio de que  $\hat{h}_{VCI}$  tenga un sesgo mayor que el  $h$  óptimo para el ECIM.

También existen situaciones, igual que antes, donde  $VCI(h)$  tiene más de un mínimo.

### 1.8.5. Métodos plug-in

Como estamos observando, uno de los motivos de los métodos Hi-tech es la estimación de la integral de las derivadas al cuadrado de la densidad real de los datos, ya que surgen en varias expresiones de diferentes estimaciones óptimas de  $h$ . Por lo que antes de continuar con los métodos Hi-tech, vamos a analizar e introducir algunas notaciones.

$$R(f^{(s)}) = \int f^{(s)}(x)^2 dx = (-1)^s \int f^{(2s)}(x) f(x) dx \quad (\text{Integración por partes})$$

Por tanto, es suficiente el estudio de la estimación de:

$$\psi_r = \int f^{(r)}(x) f(x) dx \quad \forall r \text{ par}$$

Ya que tenemos que  $\psi_r = E[f^{(r)}(X)]$ , un buen estimador podría ser:

$$\hat{\psi}_r(g) = n^{-1} \sum_{i=1}^n \hat{f}^{(r)}(X_i; g) = n^{-2} \sum_{i=1}^n \sum_{j=1}^n L_g^{(r)}(X_i - X_j) \quad (1.6)$$

$$(g \neq h, L \neq K)$$

Las propiedades del ECMA de  $\hat{\psi}_r$  son fundamentales para los siguientes métodos de selección de  $h$ . Asumimos:

1. El núcleo  $L$  es simétrico de orden par  $k$ , con  $r$  derivadas tales que:  
 $(-1)^{\frac{r+k}{2}+1} L^{(r)}(0) \mu_k(L) > 0$ .
2. La densidad  $f$  tiene  $p$  derivadas continuas monótonas con  $p > k$ .
3.  $g = g_n$  es una serie de valores positivos de parámetros de escala (escala piloto) tal que:

$$\lim_{n \rightarrow \infty} g = 0 \text{ y } \lim_{n \rightarrow \infty} n g^{2r+1} = \infty$$

Si desarrollamos la expresión (1.6) tenemos:

$$\hat{\psi}_r(g) = n^{-1} L_g^{(r)}(0) + n^{-1} \sum_{i=1}^n \sum_{i \neq j}^n L_g^{(r)}(X_i - X_j)$$

Si calculamos el sesgo y la varianza de este estimador y operamos, obtenemos que ECM es: ( $ECM = \text{Sesgo}^2 + \text{Var}$ )

$$ECM(\hat{\psi}_r(g)) = [n^{-1} g^{-r-1} L^{(r)}(0) + (k!)^{-1} g^k \mu_k(L) \psi_{r+k}]^2 +$$

$$+ 2n^{-2} g^{-2r-1} R(L^{(r)}) \psi_0 + 4n^{-1} \left[ \int f^{(r)}(x)^2 f(x) dx - \psi_r^2 \right] + O(g^{2k+2}) + o(n^{-2} g^{-2r-1} + n^{-1})$$

Igual que en ocasiones anteriores, obtenemos:

$$g_{ECMA} = \left[ \frac{k!L^{(r)}(0)}{-\mu_k(L)\psi_{r+k}n} \right]^{\frac{1}{r+k+1}}$$

Con todo esto, tenemos que:

$$\begin{aligned} \inf_{g>0} ECM(\hat{\psi}_r(g)) &\sim 2R(L^{-1})\psi_0 \left[ \frac{\mu_k(L)\psi_{r+k}}{-L^{(r)}(0)k!} \right]^{\frac{2r+1}{r+k+1}} \cdot n^{\frac{-2k-1}{r+k+1}}, \quad k < r \\ \inf_{g>0} ECM(\hat{\psi}_r(g)) &\sim 4[Var(f^{(r)}(X))]n^{-1}, \quad k > r \\ ECM(\hat{\psi}_r(g)) &= o(n^{-1}), \quad k = r \end{aligned}$$

Por último, anotamos que computacionalmente,  $\hat{\psi}_r(g)$  puede ser muy caro con un algoritmo directo, ya que se usan operaciones de orden  $n^2$ .

### 1.8.6. Método plug-in directo

Este método se basa en relacionar estimaciones de cantidades desconocidas que aparecen en la fórmula de la escala óptima asintótica. Escala óptima para el ECIMA:

$$h_{ECIMA} = \left[ \frac{R(K)}{\mu_2(K)^2\psi_4n} \right]^{1/5}$$

Por tanto, podemos obtener una estimación de  $h$ :

$$\hat{h}_{PID} = \left[ \frac{R(K)}{\mu_2(K)^2\hat{\psi}_4(g)n} \right]^{1/5}$$

Esta estimación no es automática ya que depende de la elección de la escala piloto  $g$ . Si en  $\hat{\psi}_4(g)$  se usa el núcleo de segundo orden  $K$ , escogemos  $g$  como:  $g_{ECMA} = \left[ \frac{2K^{(4)}(0)}{-\mu_2(K)\psi_6n} \right]^{1/7}$ , pero también depende de algo desconocido,  $\psi_6$ . Podríamos estimar  $\psi_6$ , pero tendríamos el mismo problema ya que la escala piloto óptima para estimar  $\psi_r$  depende de  $\psi_{r+2}$ .

Para solucionar este problema, se suele estimar  $\psi_r$  por uno de los **Métodos rápidos y sencillos**. Así, tenemos una familia de métodos plug-in directos que depende del número de etapas de la estimación funcional antes de usar el Método rápido y sencillo. Supongamos que un método plug-in directo involucra  $l$  estimaciones de núcleos sucesivas, con la escala inicial vía un método rápido y sencillo. Lo llamaremos Método plug-in directo de  $l$  etapas y lo denotamos al estimador  $\hat{h}_{PID,l}$ .

Otro problema del método es escoger el número de etapas,  $l$ . Se suele escoger  $l = 2$ .

Nótese que el método de escala normal puede pensarse como un método de plug-in directo de 0 etapas.

### 1.8.7. Método de resolver la ecuación

Este último método plug-in se basa en la resolución de la siguiente ecuación:

$$h = \left[ \frac{R(K)}{\mu_2(K)^2 \hat{\psi}_4(\gamma(h)) n} \right]^{1/5}$$

En este caso, la escala piloto para la estimación de  $\psi_4$  es una función que depende de  $h$ . La elección de  $\gamma$  está motivada por la relación

$$g_{ECMA} = \left[ \frac{2L^{(4)}(0)\mu_2(K)^2}{R(K)\mu_2(L)} \right]^{\frac{1}{7}} \left( \frac{-\psi_4}{\psi_6} \right)^{\frac{1}{7}} h^{5/7} h_{ECMA}$$

Finalmente escogemos:

$$\gamma(h) = \left[ \frac{2L^{(4)}(0)\mu_2(K)^2}{R(K)\mu_2(L)} \right]^{\frac{1}{7}} \left[ \frac{-\hat{\psi}_4(g_1)}{-\hat{\psi}_6(g_2)} \right]^{\frac{1}{7}} h^{\frac{5}{7}}$$

siendo  $\hat{\psi}_4(g_1)$  y  $\hat{\psi}_6(g_2)$  estimaciones tipo núcleo de  $\psi_4$  y  $\psi_6$  respectivamente. La elección de  $g_1$  y  $g_2$  se puede hacer con la fórmula de  $g_{ECMA}$ , aunque esto lleva al problema de la selección de etapas, igual que en el método plug-in directo.

### 1.8.8. Validación cruzada suavizada

Este método es similar al método plug-in en el que se usa un estimador del núcleo con escala piloto  $g$  para estimar la componente del  $ECIM(\hat{f})$  que se corresponde con el  $sesgo^2$  integrado. La diferencia es que este método se basa en el orden exacto del  $sesgo^2$  integrado, y no en su aproximación asintótica. Por otro lado, es más complicado de implementar que el método plug-in directo.

$$VCS(h) = (nh)^{-1}R(K) + \hat{SCI}(h)$$

donde  $\hat{SCI}(h)$  es una estimación del  $sesgo^2$  integrado. Como siempre, cogemos  $\hat{h}_{VCS}$  como el mínimo local más grande de  $VCS(h)$ .

La VCS es un ajuste de la VCMC que permite el “presuavizado” de las diferencias  $(X_i - X_j)$ , de ahí el nombre del método. Existen otras aproximaciones de la VCS ya estudiadas, y se puede demostrar que algunas variaciones de la VCS son un caso particular de la VCMC.

Existen buenas razones asintóticas para escoger en la VCS la escala piloto como:  $g = Cn^p h^m$  con  $C, p, m$  constantes. La elección óptima de estas constantes va dirigida para mejorar el rendimiento asintótico de  $\hat{h}_{VCS}$ .

También existe  $\hat{h}_{VCS,l}$ , estimación que involucra  $l$  etapas de estimación funcional, con una escala inicial escogida por un método rápido y sencillo.

## 1.9. Comparación de métodos de selección de escala

Podría hacerse un análisis asintótico y comparar órdenes de convergencia entre la escala seleccionada y alguna óptima. El problema es que los resultados teóricos, aunque nos dan alguna indicación de cual puede ser mejor, no siempre reflejan lo que pasa en la práctica, ya que los resultados asintóticos no suelen tener efecto en muestras más pequeñas. Por eso, la simulación computacional es una herramienta fundamental para esta comparación.

**Definición 1.13.** La forma más usual de medir el rendimiento teórico de un método particular es a través de un resultado de distribución asintótica:

$$n^\nu \cdot \underbrace{(\hat{h}/h_0 - 1)}_{\text{Error relativo de } \hat{h}} \longrightarrow_D N(\mu, \sigma^2) \quad (h_0: \text{escala óptima})$$

Esto significaría que  $\hat{h}$  tiene un orden relativo de convergencia a  $h_0$  de  $n^{-\nu}$  con varianza asintótica  $\sigma^2$ . Por tanto, valores altos de  $\nu$  corresponden a escalas que convergen más rápido a la escala óptima. Además, valores más pequeños de  $\sigma^2$  corresponden a escalas más estables.

Veamos que  $h_0$  escogemos. Tenemos que  $h_{ECI}$  es la escala que minimiza el  $ECI(\hat{f}(\cdot; h))$ , y que  $h_{ECIM}$  es la escala que minimiza el  $ECIM(\hat{f}(\cdot; h))$ . Por resultados *minimax* tenemos que el orden de convergencia de cualquier  $\hat{h}$  no es más rápido que  $n^{-1/10}$  para el  $h_{ECI}$ , y no más rápido que  $n^{-1/2}$  para el  $h_{ECIM}$ . Esto implica que será más sencillo minimizar el ECIM que minimizar el ECI, por lo que escogemos  $h_0 = h_{ECIM}$  para la comparación teórica de los métodos de elección.

### 1.9.1. VCMC y VCI

Bajo ciertas condiciones de regularidad, está demostrado que:

$$n^{1/10}(\hat{h}_{VCMC}/h_{ECIM} - 1) \longrightarrow_D N(0, \sigma_{VCMC}^2)$$

$$n^{1/10}(\hat{h}_{VCI}/h_{ECIM} - 1) \longrightarrow_D N(0, \sigma_{VCI}^2)$$

El ratio de estas varianzas asintóticas para el núcleo normal estandar es  $\sigma_{VCMC}^2/\sigma_{VCI}^2 \approx 15,7$ , lo que implica que  $\hat{h}_{VCMC}$  es mucho más variable que  $\hat{h}_{VCI}$ . Ambos tienen un orden de convergencia a  $h_{ECIM}$  de  $n^{-1/10}$ , bastante más lento que  $n^{-1/2}$ . Este rendimiento asintótico tan malo es efecto de la elección de la escala piloto  $g$ .

### 1.9.2. Plug-in y VCS

Tanto para el método de plug-in directo como para el método de resolver la ecuación (ambos de dos etapas), se puede demostrar que:

$$n^{5/14}(\hat{h}_{PI,2}/h_{ECIM} - 1) \longrightarrow_D N(0, \sigma_{PI}^2)$$

Esto significa que el orden de convergencia es de  $n^{-5/14}$ , el cual está más cerca de  $n^{-1/2}$  que de  $n^{-1/10}$ .

En el caso de la VCS, se llega también a que:

$$n^{5/14}(\hat{h}_{VCS}/h_{ECIM} - 1) \longrightarrow_D N(0, \sigma_{VCS}^2)$$

La clave en estos métodos es la elección de la escala piloto  $g$ , la cual representa una compensación entre la varianza y el sesgo de la estimación de  $h$  que se esté utilizando. Por tanto, un buen método para determinar el  $g$  óptimo es combinar términos de la varianza asintótica y del sesgo para obtener un error relativo pequeño. Por ejemplo, para la VCS, la opción asintóticamente óptima de  $g$  es:

$$g = \left[ \frac{(C_5/C_3)^{1/7}}{C_0^m} \right] n^{m/5-1/7} h^m, \quad m \neq -2$$

con  $C_3$  y  $C_5$  dependiendo de  $K$  y  $L$  y de  $f$  a través de  $\psi_r$ , y además

$$C_3 = \frac{1}{10}(m+2) \left[ \frac{-\mu_2(K)^6 \mu_2(L)^5 \psi_6^5}{R(K)^3 \psi_4^2} \right]^{1/5} \quad \text{y} \quad C_0 = \left[ \frac{R(K)}{\mu_2(K)^2 R(f'')} \right]^{1/5}.$$

Si  $m = 0$  obtenemos la escala piloto óptima para el método PID:  $g = (C_5/C_3)^{1/7} n^{-1/7}$ .

### 1.9.3. Método raíz n-ésima

Es posible construir métodos plug-in ó de VCS que logren el orden de convergencia óptimo de  $n^{-1/2}$ , son los llamados *Métodos raíz n-ésima*. Uno de estos métodos se basa en la aproximación de  $h_{ECIM}$ :

$$h_{ECIMA,2} = \left[ \frac{R(K)}{\mu_2(K)^2 \psi_4 n} \right]^{1/5} + \frac{\mu_4(K) \psi_6}{20} \left[ \frac{R(K)^3}{\mu_2(K)^{11} \psi_4^8 n^3} \right]^{1/5}$$

El error relativo de  $h_{ECIMA,2}$  es de orden  $n^{-3/5}$ , mejor que el orden de la aproximación  $h_{ECIMA}$  de  $n^{-2/5}$ .

Si se usan núcleos de mayor orden para estimar  $\psi_4$  y  $\psi_6$  con un ECM de  $O(n^{-1})$ , la estimación de la escala tipo plug-in resultante conseguiría el orden óptimo de  $n^{-1/2}$ .

El orden de convergencia óptimo se puede obtener también escogiendo los parámetros de la escala piloto  $g = C n^p h^m$ . Por ejemplo, para la VCS (análogo para plug-in) se llega a que con  $g = (-C_5/C_4)^{1/9} C_0^2 n^{-23/45} h^{-2}$  se cumple que:

$$n^{1/2}(\hat{h}_{VCS}/h_{ECIM} - 1) \longrightarrow_D N(0, \sigma_{VCS}^2)$$

Observamos que se puede obtener el orden de  $n^{-1/2}$  sin usar núcleos de mayor orden.

#### 1.9.4. En la práctica

Los resultados asintóticos previos, a parte de necesitar que la muestra sea suficientemente grande, tienen oculta la elección de parámetros auxiliares, la estimación de escala por el Método de escala normal ó el número de etapas en un método plug-in, entre otras posibilidades. La principal herramienta para evaluar el rendimiento de un método de selección de escala es la simulación. Así, en un mismo gráfico, podemos comparar diferentes métodos, ó un métodos con diferentes etapas.

Es difícil dar un resumen conciso de los resultados de una simulación en general ya que no todos los métodos son igual de buenos para todas las densidades, aunque aparecen algunos patrones.

Como sugieren los resultados asintóticos, los métodos hi-tech que involucran escalas piloto, como  $\hat{h}_{PID}$  y  $\hat{h}_{VCS}$ , tienen un buen rendimiento para muchos tipos de densidades. Sin embargo, para densidades con rasgos afilados, los resultados tienden a ser menos precisos, y el rendimiento de los métodos puede ser peor que el de un método no asintótico.

La VCMC no depende de argumentos asintóticos, pero su variabilidad muestral suele ser muy alta para ser fiable en la práctica. El rendimiento en las simulaciones del método VCI tampoco es bueno, por lo que en general tampoco se recomienda este método.

En resumen, aunque el reciente progreso en este campo está dirigido a conseguir un buen rendimiento de los métodos de selección de la escala, no hay una regla que garantice que funcionen en todos los casos. Una buena estrategia para el análisis de datos es obtener varias estimaciones de escala, con diferentes métodos de selección y diferentes parámetros auxiliares. En el caso de querer usar una sola escala, se recomienda  $\hat{h}_{PID}$ ,  $\hat{h}_{RLE}$  y  $\hat{h}_{VCS}$  antes que  $\hat{h}_{VCMC}$  y  $\hat{h}_{VCI}$ .

## Capítulo 2

# Estimación vectorial tipo núcleo de la densidad

La necesidad de estimar es mayor que con datos escalares ya que la modelización paramétrica con datos vectoriales es más difícil. La escasez de datos en espacios de mayor dimensión hace que estas técnicas de estimación sean demasiado difíciles de aplicar. Esto es conocido como la “maldición de la dimensionalidad“, que significa que con tamaños de muestra prácticos, la estimación de la densidad no paramétrica es muy difícil a partir de 5 dimensiones. Dicho esto, cabe destacar que hay muchos estudios donde el estimador tipo núcleo de la densidad es una herramienta efectiva para mostrar la estructura de muestras bivariantes. Recientemente también ha sido útil para conjuntos de datos de 3 y 4 dimensiones.

Para este capítulo usaremos la siguiente notación:

Muestra aleatoria vectorial de dimensión  $d$ ,  $\mathbf{X}_1, \dots, \mathbf{X}_n$ , con densidad  $f$ , siendo  $\mathbf{X}_i = (X_{i1}, \dots, X_{id})^T$ .

Además,  $\mathbf{x} \in \mathbb{R}^d \Rightarrow \mathbf{x} = (x_1, \dots, x_d)^T$ . Por tanto  $d\mathbf{x} = dx_1 \cdots dx_d$  y  $\int \equiv \int \cdots \int_{\mathbb{R}^d}$ .

Por último, denotaremos como  $I$  la matriz identidad  $d \times d$ .

**Definición 2.1.** De forma análoga a (1.1), el *Estimador vectorial tipo núcleo de la densidad* es:

$$\hat{f}(\mathbf{x}; H) = n^{-1} \sum_{i=1}^n K_H(\mathbf{x} - \mathbf{X}_i)$$

siendo

$$K_H(\mathbf{x}) = |H|^{-1/2} K(H^{-1/2}\mathbf{x}) \text{ con } K(\cdot) \text{ función núcleo } d\text{-variante, } \int K(\mathbf{x}) d\mathbf{x} = 1$$

$H$  : Matriz escala,  $d \times d$ , simétrica y definida positiva.

**Definición 2.2.** Existen dos técnicas para generar núcleos multivariantes a partir de un núcleo  $K$  univariante y simétrico:

1. Núcleo producto:

$$K(\mathbf{x}) = \prod_{i=1}^n k(x_i)$$

2. Núcleo esférico ó radialmente simétrico:

$$K(\mathbf{x}) = c_{k,d} k((\mathbf{x}^T \mathbf{x})^{1/2}) = \frac{k((\mathbf{x}^T \mathbf{x})^{1/2})}{\int k((\mathbf{x}^T \mathbf{x})^{1/2}) d\mathbf{x}}$$

Una elección de  $K(\mathbf{x})$  suele ser la densidad normal estándar d-variante:

$$K(\mathbf{x}) = (2\pi)^{-d/2} \exp(-\frac{1}{2} \mathbf{x}^T \mathbf{x}) \Rightarrow K_H(\mathbf{x} - \mathbf{X}_i) \text{ es la densidad } N(\mathbf{X}_i, \mathbf{H}) \text{ en el vector } \mathbf{x}.$$

(Se podría construir con cualquiera de las dos técnicas con  $k$  densidad normal estándar univariante)

**Definición 2.3.** Vamos a introducir algunos conjuntos de matrices, que darán lugar a la elección de diferentes parametrizaciones para la estimación.

1.  $\mathcal{F}$ : Matrices simétricas,  $d \times d$  y definidas positivas.
2.  $\mathcal{D}$ : Matrices diagonales,  $d \times d$  y definidas positivas.
3.  $\mathcal{S} = \{h^2 I / h > 0\}$ .

Claramente tenemos que  $\mathcal{S} \subseteq \mathcal{D} \subseteq \mathcal{F}$ .

$H$  suele tener  $\frac{1}{2}d(d+1)$  entradas independientes, que incluso para un  $d$  moderado, serían muchos parámetros a escoger. Por lo que podemos simplificar el estimador tipo núcleo con la restricción  $H \in \mathcal{D}$ ,  $H = \text{diag}(h_1^2, \dots, h_d^2)$ :

$$\hat{f}(\mathbf{x}; \mathbf{h}) = n^{-1} \left( \prod_{l=1}^d h_l \right)^{-1} \sum_{i=1}^n K \left( \frac{x_1 - X_{i1}}{h_1}, \dots, \frac{x_d - X_{id}}{h_d} \right) \quad (\text{Epanechnikov, 1969})$$

Conseguimos una mayor simplificación con  $H \in \mathcal{S}$ :

$$\hat{f}(\mathbf{x}; h) = n^{-1} h^{-d} \sum_{i=1}^n K[(\mathbf{x} - \mathbf{X}_i)/h] \quad (\text{Cacoullos, 1966})$$

Como en el caso escalar (univariante), hay ciertas elecciones importantes que hay que hacer. La extensión a dimensiones mayores significa, sin embargo, que hay más grados de

libertad. Hay que escoger primero el núcleo  $d$ -variante. Después, hay que decidir la parametrización del suavizado, es decir, la matriz de escala  $H$ . Una matriz completa da más flexibilidad, pero tiene más parámetros que hay que escoger.

A continuación vamos a ir introduciendo una metodología para cuantificar matemáticamente el rendimiento del estimador multivariante tipo núcleo de la densidad.

## 2.1. Aproximaciones asintóticas del ECIM

Bajo ciertas suposiciones sobre la densidad  $f$ , se puede obtener una expresión simple de una aproximación asintótica del ECIM. Esas suposiciones nos tienen que permitir usar la versión vectorial del *Teorema de Taylor*. Por tanto, asumimos que :

1. Cada entrada de la Hessiana de  $f$ ,  $\mathcal{H}_f(\cdot)$ , es continua a trozos e integrable al cuadrado.
2.  $H = H_n$  es una serie de matrices de escala tal que  $n^{-1}|H|^{-1/2}$  y todas las entradas de  $H$  tienden a 0 cuando  $n$  tiende a  $\infty$ . Además, el ratio de los autovalores más altos y más bajos está acotado por  $n$ .
3.  $K$  está acotado, con soporte compacto, cumpliendo:

$$\int K(\mathbf{z})d\mathbf{z} = 1, \quad \int \mathbf{z}K(\mathbf{z})d\mathbf{z} = 0, \quad \int \mathbf{z}\mathbf{z}^T K(\mathbf{z})d\mathbf{z} = \mu_2(K)I$$

$$(\text{con } \mu_2(K) = \int z_i^2 K(\mathbf{z})d\mathbf{z})$$

Bajo estas suposiciones, calculando los términos del sesgo y varianza del estimador tipo núcleo de la densidad, obtenemos que:

$$ECIMA(\hat{f}(\cdot; H)) = n^{-1}|H|^{-1/2}R(K) + \frac{1}{4}\mu_2(K)^2 \int \text{tr}(H\mathcal{H}_f(\mathbf{x}))^2 d\mathbf{x}$$

Otra posible notación sería:

$$ECIMA(\hat{f}(\cdot; H)) = n^{-1}|H|^{-1/2}R(K) + \frac{1}{4}\mu_2(K)^2(\text{vech}(H))^T \Psi_F(\text{vech}(H)) \quad (2.1)$$

Notación:

Sea  $A = [C_1 \dots C_d]$  una matriz cuadrada  $d \times d$ .

Vector de  $A$ :  $vec(A)$  vector de dimensión  $d^2 \times 1$  cuyas entradas son todos los elementos por orden de las columnas de  $A$ .

Vector medio de  $A$ :  $vech(A)$  vector de dimensión  $\frac{1}{2}d(d+1) \times 1$  que se obtiene eliminando de  $vec(A)$  los elementos por encima de la diagonal.

$$\Psi_F = \int vech(2\mathcal{H}_f(\mathbf{x}) - dg(\mathcal{H}_f(\mathbf{x}))) \times vech(2\mathcal{H}_f(\mathbf{x}) - dg(\mathcal{H}_f(\mathbf{x})))^T d\mathbf{x}.$$

Cada componente de  $\Psi_F$ :  $\psi_{\mathbf{r}} = \int f^{(\mathbf{r})}(\mathbf{x})f(\mathbf{x})d\mathbf{x}$ .

Al contrario que en el caso escalar, en general las expresiones explícitas de la matriz escala  $H$  óptimas respecto al ECIMA no están disponibles y hay que calcularlas numéricamente. Cuando  $H \in \mathcal{D}$ , es decir,  $H = diag(h_1^2, \dots, h_d^2) = diag(\mathbf{h}^2)$ , la expresión (2.1) se reduce a:

$$ECIMA(\hat{f}(\cdot; H)) = n^{-1}R(K) \left( \prod_{j=1}^d h_j \right)^{-1} + \frac{1}{4}\mu_2(K)^2(\mathbf{h}^2)^T \Psi_D(\mathbf{h}^2)$$

con  $\Psi_D$  matriz  $d \times d$  con la entrada  $(i, j)$  igual a  $\psi_{2\mathbf{e}_i + 2\mathbf{e}_j}$ .

Cuando  $H \in \mathcal{S}$ , es decir,  $H = h^2I$ , la expresión (2.1) se reduce a:

$$ECIMA(\hat{f}(\cdot; H)) = n^{-1}h^{-d}R(K) + \frac{1}{4}h^4\mu_2(K)^2 \int [\nabla^2 f(\mathbf{x})]^2 d\mathbf{x}$$

con  $\nabla^2 f(\mathbf{x}) = \sum_{i=1}^d \frac{\partial^2}{\partial x_i^2} f(\mathbf{x})$ . En este último caso si que podemos obtener una fórmula explícita de la escala óptima:

$$h_{ECIMA} = \left[ \frac{d \cdot R(K)}{\mu_2(K)^2 \int \nabla^2 f(\mathbf{x}) d\mathbf{x} n} \right]^{\frac{1}{d+4}}$$

Con esto, obtenemos que el ECIMA mínimo es:

$$\inf_{h>0} (ECIMA(\hat{f}(\cdot; h))) = \frac{d+4}{4d} \left[ \mu_2(K)^{2d} (dR(K))^4 \left( \int (\nabla^2 f(\mathbf{x}))^2 d\mathbf{x} \right)^d n^{-4} \right]^{\frac{1}{d+4}}$$

Observamos que el orden de convergencia es  $O(n^{\frac{-4}{d+4}})$ , con lo que es más lento cuando la dimensión aumenta. Esto nos dice que el rendimiento de los estimadores en dimensiones altas no es bueno, pero si que es práctica la estimación en dimensiones moderadas.

## 2.2. Elección del núcleo multivariante

Recordamos que las dos formas más populares de crear núcleos multivariantes a partir de núcleos univariantes son el *Núcleo producto*,  $K^P(\mathbf{x})$ , y el *Núcleo esféricamente simétrico*,  $K^S(\mathbf{x})$ . Vamos a comparar ambos núcleos cuando se crean a partir del mismo núcleo univariante. En general son diferentes, excepto cuando el núcleo univariante  $k(\cdot)$  es la densidad normal. Para estudiar esta comparación necesitamos la expresión multivariante análoga a  $C(K) = [R(K)^4 \mu_2(K)^2]^{1/5}$ , la cual determina el rendimiento del ECIMA de núcleos univariantes. Esa expresión es:

$$C_d(K) = [R(K)^4 \mu_2(K)^{2d}]^{\frac{1}{d+4}}$$

**Definición 2.4.** Por extensión de la eficiencia de núcleos univariantes, tenemos que la eficiencia del núcleo  $K^S$  respecto al núcleo  $K^P$  vendrá dada por:

$$[C_d(K^S)/C_d(K^P)]^{\frac{d+4}{4}} = \frac{R(K^S) \mu_2(K^S)^{d/2}}{R(K^P) \mu_2(K^P)^{d/2}}$$

Para cada una de estas dos clases de núcleos multivariantes, es posible calcular el núcleo óptimo respecto al ECIMA. Para los núcleos producto, la cantidad a minimizar es:

$$C_d(K^P) = [R(k)^4 \mu_2(k)^2]^{\frac{d}{d+4}}$$

El núcleo que minimiza esto respecto a  $k$  es el que minimiza  $C(K)$  en el caso univariante, el núcleo de Epanechnikov. Por tanto:

**Definición 2.5.** El núcleo producto óptimo respecto al ECIMA es:

$$K_*^P(\mathbf{x}) = \left(\frac{3}{4}\right)^d \prod_{i=1}^d (1 - x_i^2) 1_{[|x_i| < 1]}$$

**Definición 2.6.** Para el núcleo esféricamente simétrico, el problema de optimización es más complicado. Se obtiene que el núcleo esféricamente simétrico óptimo respecto al ECIMA es:

$$K_*^S = \frac{1}{2} v_d^{-1} (d+2) (1 - \mathbf{x}^T \mathbf{x}) 1_{[\mathbf{x}^T \mathbf{x} \leq 1]}$$

donde  $v_d$  es el volumen de la esfera unitaria de dimensión  $d$ .

## 2.3. Elección de la parametrización del suavizado

Hay varios niveles de complejidad al escoger la matriz  $H$ . Como hemos visto, el más bajo es la restricción  $H \in \mathcal{S}$ , es decir,  $H = h^2 I$ ,  $h > 0$ . Esta restricción tiene la ventaja de

que solo tenemos que trabajar con un único parámetro, pero tiene la gran desventaja de que la cantidad de suavizado es la misma en cada coordenada.

El siguiente nivel sería  $H \in \mathcal{D}$ , es decir,  $H = \text{diag}(h_1^2, \dots, h_d^2)$ . Aquí tenemos  $d$  parámetros diferentes, pero podemos suavizar en diferentes medidas cada coordenada.

En algunos casos, nos interesa suavizar en direcciones diferentes a los ejes de coordenadas.

En este caso usamos una matriz de escala completa,  $H \in \mathcal{F}$ .

Existe una forma sencilla de obtener una matriz de escala con una orientación arbitraria:  $H = h^2 S$ , siendo  $S$  la matriz de covarianzas muestrales. Este planteamiento es equivalente a transformar linealmente los datos para obtener una matriz de covarianzas unitaria (proceso conocido como “esferificar“ los datos), aplicar el estimador tipo núcleo escalar a esos datos transformados, y deshacer la transformación para obtener el estimador de la densidad de los datos originales. Por ejemplo, se sabe que si  $f$  es una  $N(\mu, \Sigma)$  multivariante, entonces la  $H$  óptima respecto al ECIMA es  $H_{ECIMA} = c\Sigma$ , con  $c$  escalar que depende de  $d$  y de  $n$ . Por tanto, en caso de datos normales, la transformación nombrada de los datos es apropiada.

**Definición 2.7.** Una forma de comparar el coste de usar una clase particular de matrices de escala para estimar una densidad dada  $f$ , es con la *Eficiencia Relativa Asintótica* (ERA). Para una clase de matrices de escala  $\mathcal{A}$ , se compara con la clase de matrices de escala completas  $\mathcal{F}$  con:

$$ERA_f(\mathcal{F} : \mathcal{A}) = \left[ \frac{\inf_{H \in \mathcal{F}} ECIMA(\hat{f}(\cdot; H))}{\inf_{H \in \mathcal{A}} ECIMA(\hat{f}(\cdot; H))} \right]^{\frac{d+4}{4}}$$

Ya que para cada clase  $\mathcal{A}$ ,  $\inf_{H \in \mathcal{A}} ECIMA(\hat{f}(\cdot; H)) = O(n^{-\frac{4}{d+4}})$ , esto se puede interpretar como que una muestra de tamaño  $n$  en la que se usa  $H \in \mathcal{A}$  tiene el mismo ECIMA mínimo que una muestra de tamaño  $ERA_f(\mathcal{F} : \mathcal{A}) \cdot n$  en la que se usa  $H \in \mathcal{F}$ . Aunque el estudio de esto es amplio ya que depende de la forma de la densidad, su orientación respecto a los ejes, etc.

No habrá pérdida de eficiencia si comparamos el proceso con  $H \in \mathcal{D}$  y con  $H \in \mathcal{F}$ , si la masa de probabilidad aparece sobre los ejes de coordenadas. En caso de haber alguna rotación, esperamos que la eficiencia caiga por debajo de 1.

Observamos que la elección de la parametrización del suavizado tiene una gran influencia en el rendimiento de la estimación. Concretamente, no se pierde mucho rendimiento usando una matriz de escala diagonal, aunque las matrices completas son necesarias en algunos casos. Las otras parametrizaciones simplificadas tienen demasiadas carencias como para que en general se recomienden usar.

## 2.4. Selección de la matriz escala $H$

Muchas ideas de los métodos de selección de la escala  $h$  en el caso escalar se pueden extender al caso vectorial para seleccionar la matriz escala  $H$ . Veamos algunos de estos métodos:

### 2.4.1. Validación cruzada por mínimos cuadrados

$$\hat{H}_{VCMC} = \arg \left( \min_{H \in \mathcal{F}} VCMC(H) \right)$$

siendo

$$VCMC(H) = \int \hat{f}(\mathbf{x}; H)^2 d\mathbf{x} - 2n^{-1} \sum_{i=1}^n \hat{f}_{-i}(\mathbf{X}_i; H)$$

y  $\hat{f}_{-i}(\cdot; H)$  el estimador vectorial tipo núcleo basado en la muestra sin  $\mathbf{X}_i$ .

Podrían hacerse los cálculos tomando  $H \in \mathcal{D}$  ó  $H \in \mathcal{S}$ , pero hay aspectos teóricos desafortunados que se extienden de la versión escalar de la VCMC. Sin embargo, sorprendentemente, el orden de convergencia de la VCMC mejora con dimensiones mayores.

Aunque no lo analicemos, la VCI también tiene extensión directa para dimensiones mayores.

### 2.4.2. Métodos Plug-in

Usando las aproximaciones asintóticas del ECIMA es posible desarrollar versiones vectoriales de métodos de selección plug-in. De los desarrollos de la sección 2.1 observamos que las únicas cantidades desconocidas son:

$$\psi_r = \int f^{(r)}(\mathbf{x}) f(\mathbf{x}) d\mathbf{x}$$

Este funcional se puede sustituir por un estimador tipo kernel de la forma:

$$\hat{\psi}_r(G) = n^{-1} \sum_{i=1}^n \hat{f}^{(r)}(\mathbf{X}_i; G)$$

siendo  $G$  otra matriz de escala.

Esto permite utilizar esta estimación para minimizar el resultado estimado del ECIMA.

Está claro que existen muchas posibilidades para la selección de escala en el caso vectorial, sin embargo muchas cuestiones prácticas están por resolver, estamos ante un campo bastante actual.



## Capítulo 3

# Estimación de la densidad en Grafos Lineales

Hasta ahora hemos desarrollado toda una teoría sobre la estimación tipo núcleo para casos escalares y casos vectoriales. Es el momento de introducir una nueva metodología para pasar de los espacios euclidianos bidimensionales a los grafos lineales, que es el caso que nos interesa analizar.

Existen muchas situaciones a analizar en muchos campos que motivan la estimación de la densidad en puntos de grafos lineales, como pueden ser: accidentes en las carreteras, atropello de animales, crímenes en las calles, robos en pasillos de centros comerciales u otros lugares, distanciamiento de plantas y árboles en las aceras, localización de pájaros marinos a lo largo de la costa, potencia de la señal WiFi a lo largo de las calles, etc. Otras posibles situaciones son los eventos que se producen al lado en vez de sobre el grafo, como pueden ser las entradas a las instalaciones en una red de calles muy concurridas.

Muchos de los métodos estadísticos para analizar estas situaciones han sido desarrollados recientemente. Los datos no estaban fácilmente disponibles y la computación es mucho más difícil que en el plano. En los años 90 aparecen los primeros estudios teóricos, y gracias a internet y la tecnología, los datos se han podido obtener más fácil.

A continuación introducimos la definición de grafo lineal y toda la notación correspondiente:

### Definición 3.1.

1. Primeramente definimos un segmento en el plano con puntos finales  $u$  y  $v$  como:

$$[u, v] = \{tu + (1 - t)v / 0 \leq t \leq 1\}$$

2. Un grafo lineal sería  $N = \bigcup_{i=1}^m l_i$ , con  $l_1, \dots, l_n$  segmentos del plano con  $1 \leq n \leq \infty$ . Denotamos la distancia total de todos los segmentos por  $|L|$ .
3. También podemos ver un grafo lineal  $N = (V, L)$  como un conjunto de nodos o vértices  $V$  junto a un conjunto de segmentos o aristas  $L$  contenidos en un plano. Asumimos que las aristas de  $L$  no se intersecan entre si excepto en los nodos de  $V$ . Además partimos de que el grafo  $N$  no es directo, es decir, las aristas son de doble sentido.  $N$  está dentro de un plano por simplificar, se pueden extender los cálculos cuando está contenido en un espacio de mayor dimensión.
4. El grado de un vértice  $v$ ,  $deg(v)$ , es el número de segmentos  $l = [v, v']$  con  $v$  como uno de los puntos finales. Se dice vértice terminal si tiene grado 1.
5. Un camino entre dos puntos  $x$  e  $y$  a lo largo de  $N$  es una secuencia  $\pi = (x, v_1, \dots, v_p, y)$  donde  $v_1, \dots, v_p$  son vértices tales que  $l_j = [v_j, v_{j+1}]$  son segmentos de  $L$  para  $j = 1, \dots, p-1$ , cumpliendo que  $x$  y  $v_1$  estén en un segmento común  $l_0$ , e  $y$  y  $v_p$  en un segmento  $l_p$ . La distancia del camino sería  $d(\pi) = \|v_1 - x\| + \sum_{j=1}^{p-1} \|v_{j+1} - v_j\| + \|y - v_p\|$ , siendo  $\|\cdot\|$  la distancia euclidiana. La distancia del camino más corto entre  $u$  y  $v$ ,  $d_N(u, v)$ , es el mínimo de las distancias de todos los caminos entre  $u$  y  $v$ . Si no hay caminos de  $u$  a  $v$  (grafo no conexo) entonces  $d_N(u, v) = \infty$ .
6. Denotamos como  $\tilde{L}$  al conjunto de puntos de la unión de las aristas de  $L$ , incluyendo a los nodos. Sea  $y \in \tilde{L}$ :

$$L_y = \{\text{puntos que distan de } y \text{ por el camino más corto menos ó igual que } h\} \subset \tilde{L}$$

Relacionando todo esto con la estimación tipo núcleo, tenemos que la función núcleo de centro  $y$  y soporte  $L_y$  que se usaría sería:

$$K_y(x) \begin{cases} \geq 0 & \forall x \in L_y \\ = 0 & \forall x \in \tilde{L} \setminus L_y \end{cases}, \quad \int_{x \in L_y} K_y(x) dx = 1$$

La forma de  $K_y(x)$  puede variar ó no entre los distintos puntos de  $L_y$ . El problema es que habrá que diferenciar distintas situaciones ya que  $L_y$  puede ser un segmento, un árbol o incluir ciclos. Si  $y_1, \dots, y_n$  son  $n$  puntos independientes e idénticamente distribuidos en  $\tilde{L}$  bajo una densidad desconocida  $f(x)$ , tenemos que el estimador tipo núcleo de  $f(x)$  es:

$$\hat{f}(x) = \frac{1}{n} \sum_{i=1}^n K_{y_i}(x)$$

Vamos a introducir una notación para analizar los diferentes casos que pueden aparecer en un grafo lineal:

$$V_{(1)} = \{\text{nodos con grado } 1\} \subset V$$

$$V_{(\geq 3)} = \{\text{nodos con grado mayor ó igual a } 3\} \subset V$$

$$L_{(1)i} = \{\text{puntos de } \tilde{L} \text{ que distan del nodo } i\text{-ésimo de } V_{(1)} \text{ menos ó igual que } 2h\} \subset \tilde{L}$$

$$L_{(\geq 3)i} = \{\text{puntos de } \tilde{L} \text{ que distan del nodo } i\text{-ésimo de } V_{(\geq 3)} \text{ menos ó igual que } 2h\} \subset \tilde{L}$$

$$L_S = \tilde{L} \setminus \left\{ \bigcup_i L_{(1)i} \cup \bigcup_i L_{(\geq 3)i} \right\}$$

En cada segmento de  $L_S$  y  $L_{(1)i}$  podríamos aplicar la estimación escalar tipo núcleo de la densidad, aunque en el caso de los puntos de  $L_{(1)i}$  habría problemas en los puntos de los extremos (ya se han propuesto varios métodos para abordar este problema). En el caso de  $L_{(\geq 3)i}$ , la estimación de la densidad tiene que ser diferente al método escalar, y ya hay muchos estudios sobre este caso. Esta parte es particularmente importante ya que muchos de los eventos de los grafos, suelen ocurrir en las intersecciones, y los estimadores insesgados son imprescindibles.

Comenzamos asumiendo que los puntos observados  $y_1, \dots, y_n$  están independiente e idénticamente distribuidos en  $\tilde{L}$  según la densidad de probabilidad uniforme, es decir:

$$f(x) = \frac{1}{|\tilde{L}|}, \quad x \in \tilde{L}, \quad |\tilde{L}| = \text{distancia total de las aristas de } N. \quad (3.1)$$

Ahora queremos definir funciones núcleo que nos produzcan estimadores tipo núcleo insesgados, es decir, estimadores que cumplan que

$$\text{Sesgo}(\hat{f}) = E(\hat{f}) - f(x) = \int_{y \in \tilde{L}} K_y(x) f(y) dy - f(x) = 0$$

Se pueden definir muchas funciones núcleo en un grafo, que pueden ser caracterizadas por todas ó algunas de las siguientes propiedades:

1.  $K_y(x)$  unimodal.
2. El centro de  $K_y(x)$  es en el punto  $y$ .
3.  $K_y(x)$  es continua.
4.  $d(x, y) = d(x', y) \Rightarrow K_y(x) = K_y(x') \quad \forall x, x' \in L_{(\geq 3)i}$ .

5. Invariante respecto al ángulo de los nodos: Si el grafo formado por un nodo y aristas en  $L_y$  es isomorfo al grafo formado por un nodo y aristas en  $L_{y'}$ , y además el tamaño de cada arista en  $L_y$  es el mismo que la correspondiente arista en  $L_{y'}$ , entonces  $K_y(x) = K_{y'}(x) \forall x \in L_y$  y su correspondiente  $x' \in L_{y'}$ .
6.  $K_y(x)$  es simétrica con respecto a  $x$  e  $y$ , es decir,  $K_x(y) = K_y(x) \forall x, y \in \tilde{L}$ .
7.  $K_y(x)$  es insesgado, es decir, tiene sesgo 0 para la función de densidad uniforme  $\forall x \in \tilde{L}$ .

A continuación vamos a formular 3 tipos de funciones núcleo en un grafo en términos de  $k(x)$ , la función núcleo base, definida en  $\mathbb{R}$ , y que satisface (además de cumplir todas las propiedades anteriores menos la 5 y la 6):

1.  $k(x) = k(-x) \forall x \in \mathbb{R}$ .
2.  $\int_{-\infty}^{\infty} k(x)dx = 1$ .
3. Para un  $h$  real y positivo:  $k(x) = 0, |x| \geq h$  y  $k(x) > 0, |x| < h$ .
4.  $k(x)$  es no creciente  $\forall x \in [0, \infty)$ .
5.  $k(x)$  es continua.

Un primer paso natural para analizar los datos en un grafo lineal sería crear la estimación tipo núcleo de la tasa de eventos, que varía en el espacio. Pero en un grafo, esto no es sencillo. Una falacia muy común es la de coger la estimación tipo núcleo de la densidad para el caso escalar,  $\hat{f}(x) = \frac{1}{n} \sum_{i=1}^n k(x - y_i)$ , y trasladarla al grafo como  $\hat{f}(x) = \frac{1}{n} \sum_{i=1}^n k(d_N(x, y_i))$ , siendo  $x$  cualquier localización en el grafo,  $y_1, \dots, y_n$  las localizaciones de los datos observados,  $d_N(x, y_i)$  la distancia del camino más corto en el grafo desde  $x$  a  $y_i$ , y  $k$  es la función núcleo base definida en  $\mathbb{R}$ . Sin embargo, esta estimación es errónea ya que la aportación de cada  $y_i$  sería  $\frac{1}{n} \int_N k(d_N(x, y_i))dx$ , que es diferente a  $\frac{1}{n}$ , por lo que no se cumpliría que  $\int_N \hat{f}(x)dx = 1$ , por tanto  $\hat{f}$  no es una función de densidad válida. Esto provocará que la densidad real se sobreestimaré en las partes más densas del grafo.

A continuación damos unas opciones para la función núcleo con el fin de intentar solventar este problema.

### 3.1. Funciones núcleo de forma similar

Es cierto que para los puntos de  $L_S$  se podría utilizar una función núcleo  $K_y(x)$  unimodal, simétrica y continua. Pero para los puntos de  $L_{(\geq 3)_i}$  necesitamos una función núcleo

similar, pero con otra forma, ya que el soporte de esta cambia según donde esté situado el centro  $y$ . Definimos entonces la función núcleo como:

$$K_y(x) = \begin{cases} k(x-y), & |y| \geq h \quad (l_1) \\ c(y)k(x-y), & 0 \leq |y| \leq h \quad (l_2, \dots, l_n) \end{cases}$$

con

$$c(y) = \begin{cases} \frac{1}{1+(n-2) \int_{y-h}^0 k(x-y)dx}, & 0 \leq y \leq h \quad (l_1) \\ \frac{1}{1+(n-2) \int_0^{y+h} k(x-y)dx}, & -h \leq y \leq 0 \quad (l_2, \dots, l_n) \end{cases}$$

que cumple

$$0 < c(y) < 1 \quad \forall |y| < h \quad \text{y} \quad \frac{dc(y)}{dy} > 0, 0 < y < h; \quad \frac{dc(y)}{dy} < 0, -h < y < 0$$

Nos quedaría una función núcleo como muestra la siguiente figura:

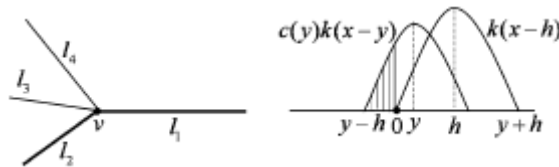


Figura 3.1: Construcción de la función núcleo

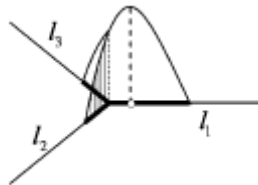


Figura 3.2: Función núcleo final alrededor de un nodo de grado  $\geq 3$

Esta construcción de la función núcleo es una extensión natural de la función núcleo escalar, sin embargo, vamos a ver que este núcleo es sesgado. Tenemos que para  $h < x < 2h$ :

$$E(K_y(x)) = \int_{x-h}^h c(y)k(x-y)dy + \int_h^{x+h} k(x-y)dy$$

Por las propiedades de  $c(y)$  tenemos que  $E(K_y(x)) < 1$  y que  $\frac{dE(K_y(x))}{dy} > 0$ , por lo que  $K_y(x)$  crea un estimador tipo núcleo sesgado cuando  $h < x < 2h$ . Por otro lado, tenemos que para  $0 < x < h$ :

$$E(K_y(x)) = \int_h^{x+h} k(x-y)dy + \int_0^h c(y)k(x-y)dy + (n-1) \int_{x-h}^0 c(y)k(x-y)dy$$

Por las propiedades de  $c(y)$  se cumple que  $E(K_y(0)) > 1$ , por lo que  $K_y(x)$  también crea un estimador sesgado cuando  $0 < x < h$ .

Todo esto nos avisa de que si usamos este núcleo en un grafo, el estimador que calculemos nos lleva a la falsa conclusión de que los puntos tienden a acumularse en los nodos, cuando los puntos se supone que están uniformemente distribuidos por el grafo.

### 3.2. Funciones núcleo de divisiones iguales

Para esta sección, asumimos que la distancia de cualquier ciclo en el grafo es más grande que  $2h$ , siendo  $h$  el parámetro de suavizado de la función núcleo básica  $k(x)$ . Sean  $x, y \in \tilde{L}$  y  $p$  el camino más corto de  $y$  a  $x$ . Sean  $v_1, \dots, v_s$  los nodos que se recorren en el camino  $p$ ,  $n_i$  el grado del nodo  $v_i \forall i = 1, \dots, s$  y  $d(x, y) = d(y, x)$  la distancia del camino más corto  $p$ . Si consideramos primero el caso en el que  $y$  no coincide con un nodo del grafo, podemos definir una función núcleo así:

$$K_y(x) = \begin{cases} \frac{k(d(y,x))}{(n_1-1)(n_2-1)\dots(n_s-1)}, & 0 \leq d(y, x) \leq h \\ 0, & d(y, x) \geq h \end{cases}$$

Esta función cumple las propiedades 1, 2 y 5 anteriormente nombradas. Pero no es continua respecto a  $x$  aunque  $k(d(x, y))$  lo sea,  $K_y(x)$  es discontinua en los nodos del grafo. Se puede comprobar fácilmente que es una función núcleo ya que por la propiedad 3 de  $k(d(x, y))$  se cumple que  $K_y(x) \geq 0 \forall x \in L_y$  y  $K_y(x) = 0 \forall x \in \tilde{L} \setminus L_y$ , y además  $\int_{x \in L_y} K_y(x) dx = 1$ . También cumple la propiedad 6,  $K_y(x) = \frac{k(d(y,x))}{(n_1-1)(n_2-1)\dots(n_s-1)} = K_x(y)$ .

Ahora consideremos el caso donde  $y$  está en el nodo  $v_0$  con grado  $n_0$ . Entonces:

$$K_y(x) = \begin{cases} \frac{k(d(y,x))}{(n_1-1)(n_2-1)\dots(n_s-1)} \cdot \frac{2}{n_0}, & 0 \leq d(y, x) \leq h \\ 0, & d(y, x) \geq h \end{cases}$$

En general nos quedaría como en la siguiente figura:

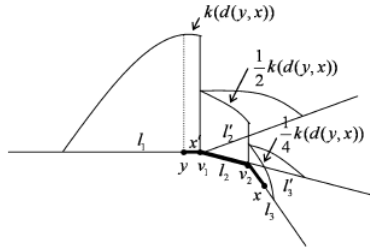


Figura 3.3: Función núcleo de divisiones iguales

El caso en el que  $y$  sea un nodo es raro, pero sería un conjunto de medida 0, por lo que

el valor de  $k(d(x, y))$  se asigna igualmente a cualquier nodo. Además, como  $\int_{-\infty}^{\infty} k(x)dx = 1$ , también se mantiene para  $k(d(x, y))$ , por lo que  $\int_{x \in \tilde{L}} K_y(x)dx = 1$  se mantiene para cualquier caso de  $y$ . Por todo esto, el núcleo dado por  $K_y(x)$  crea un estimador tipo núcleo insesgado.

Cabe destacar que esto se cumple para cualquier función núcleo básica  $k(x)$  de  $K_y(x)$ , por lo que hay un amplio grupo de funciones núcleo de divisiones iguales.

### 3.3. Funciones núcleo de divisiones iguales continuas

La función definida en la sección anterior es discontinua claramente, y parece más deseable construir una función núcleo continua. Seguimos suponiendo que el soporte de la función serán  $n$  aristas  $l_1, \dots, l_n$ , juntándose en un nodo  $v$ , y que la distancia de cada arista no sea menor que  $h$ . El caso general donde las aristas se juntan en varios nodos es similar pero con cálculos más largos.

Por lo tanto, para cada par de aristas  $l_1$  y  $l_i$ ,  $i \neq 1$  juntándose en el nodo  $v$ , donde  $y \in l_1$ , definimos:

$$K_y(x) = \begin{cases} k(x), & -h \leq x \leq 2d - h \\ k(x) - \frac{n-2}{n}k(2d-x), & 2d - h \leq x \leq d \\ \frac{2}{n}k(x), & d \leq x \leq h \end{cases}$$

Siendo  $d$  la distancia de  $y$  al nodo  $v$ . Nos quedaría algo como:

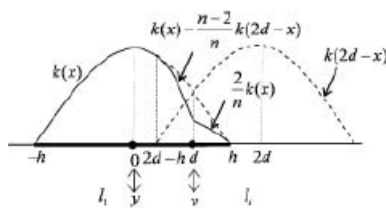


Figura 3.4: Construcción de la función núcleo

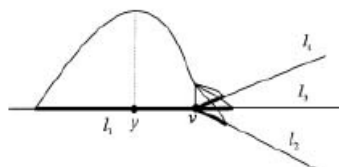


Figura 3.5: Función núcleo de divisiones iguales continua sobre aristas que se juntan en un nodo

Veamos que es una función núcleo que además es continua en el nodo:

$$\int_{-h}^{2d-h} k(x)dx + \int_{2d-h}^d [k(x) - \frac{n-2}{n}k(2d-x)]dx + (n-1)\frac{2}{n} \int_d^h k(x)dx = 1 \implies$$

$\implies K_y(x)$  es una función núcleo.

$$k(d) - \frac{n-2}{n}k(2d-d) = \frac{2}{n}k(d) \implies K_y(x) \text{ es continua en el nodo.}$$

Como en la función de la sección anterior, la forma de la función núcleo es la misma para las aristas  $l_i$ ,  $i \neq 1$ . Ahora veamos que  $K_y(x)$  da un estimador insesgado (seguimos en el caso simple donde  $n$  aristas se unen en un nodo  $v$ ). Sean un punto  $x$  de  $l_1$  y el resto de las  $n-1$  aristas  $l_i$ ,  $i = 2, \dots, n$ . La densidad en el punto  $x$  viene dada por:

$$\begin{aligned} D(0) &= \int_{-h}^{2d-h} k(-y)dy + \int_{2d-h}^d [k(-y) - \frac{n-2}{n}k(-2d+y)]dy + \\ &\quad \int_d^h (n-1)\frac{2}{n}k(-y)dy \quad \underbrace{\implies}_{\int_{2d-h}^d k(-2d+y)dy = \int_d^h k(-y)dy} \quad D(0) = \\ &= \int_{-h}^{2d-h} k(-y)dy + \int_{2d-h}^d k(-y)dy + \int_d^h k(-y)dy = \int_{-h}^h k(y)dy = 1 \end{aligned}$$

Ahora vamos a considerar un caso más general, un soporte donde haya dos nodos de donde salen dos ó mas aristas.

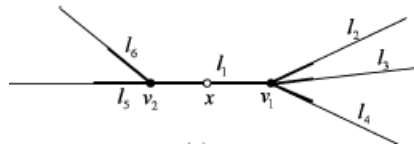


Figura 3.6: Soporte más general para la función núcleo

Pero la integración en este soporte de la función núcleo básica  $k(x)$  es equivalente a su integración en el siguiente soporte:

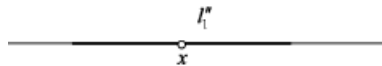


Figura 3.7: Soporte equivalente para integrar  $k(x)$

Pero como  $\int_{-h}^h k(x)dx = 1$ , también conseguiríamos un estimador tipo núcleo insesgado. Ahora vamos a considerar el caso más general, en el que el soporte del núcleo incluye 3 ó más nodos con grados de 3 ó más.

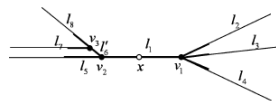


Figura 3.8: Soporte más general

Siguiendo la notación de la figura 3.8, primero analizamos la integral a lo largo de las aristas  $l_7$ ,  $l_8$  y  $l_6$ . Esta parte es equivalente al caso anterior, por lo que, como ya vimos, la integral de la función núcleo en  $x$  a través de  $y$  en este soporte es la unidad, por lo que el estimador tipo núcleo será insesgado. Por otro lado, el análisis del resto de las aristas es exactamente igual al caso anterior. Por lo que queda totalmente demostrado que la función núcleo de divisiones iguales continua crea un estimador tipo núcleo insesgado. Esto se mantiene para cualquier  $k(x)$ , por lo que hay un amplio conjunto de funciones  $K_y(x)$  de este tipo. Cabe destacar que se puede demostrar que los estimadores núcleo con las dos funciones núcleo anteriores son insesgados también en el caso de que el soporte de las funciones núcleo contenga un ciclo, aunque la propiedad 1 de que la función núcleo sea unimodal no se cumpla.

Recordemos que estamos consiguiendo estimadores de  $f(x)$  insesgados en el caso donde  $f(x)$  es la densidad de la probabilidad uniforme. Si los datos no siguen esa distribución uniforme, pero nos dan  $f(x)$  explícitamente, podemos obtener los estimadores insesgados a través de la siguiente transformación:

$f_i(x)$  : parte de  $f(x)$  con dominio la arista  $l_i \in L$ .

$t$  : distancia más corta desde el punto final de  $l_i$  a un punto de  $l_i$ .

$f_i(t)$  : densidad de los puntos en  $t$  de  $l_i$ .

$l'_i = f_i(l_i)$ .

$z(t) = \int_0^t f_i(t) dt$ .

De esta forma, la densidad de  $z$ ,  $g_i(z)$ , sigue la distribución uniforme en  $l'_i$ . Por tanto, si  $L' = \{l'_1, \dots, l'_n\}$ ,  $N' = (V, L')$ ,  $y'_j = f_i(y_j)$ ,  $y_j \in l_i$ , entonces las funciones núcleo de divisiones iguales generadas en  $y'_1, \dots, y'_n$  en el grafo  $N'$  darán estimadores de la densidad insesgados.

### 3.4. Estimación del riesgo relativo

Una vez que hemos introducido la teoría de la estimación tipo núcleo para los grafos lineales, vamos a hablar de la estimación del riesgo relativo, que es el ratio de las intensidades de dos eventos diferentes en un mismo grafo lineal. Para ello, primero tenemos que analizar la estimación tipo núcleo de las funciones intensidad.

#### 3.4.1. Función intensidad

La intensidad de podría interpretar como el número esperado de puntos aleatorios por unidad de longitud del grafo. Formalmente, para un conjunto de eventos  $X = \{x_1, \dots, x_n\}$  en un grafo  $N$ , la función tasa ó intensidad  $\lambda(u)$ ,  $u \in N$ , se define de la siguiente forma:

**Definición 3.2.** El número de puntos de  $X$  que caen en  $B \subset N$ ,  $n(X \cap B)$ , tiene una esperanza

$$E[n(X \cap B)] = \int_B \lambda(u) du, \forall B \subset N$$

Las estimaciones de  $f$  y  $\lambda$  estarán muy relacionadas ya que si fijamos  $n$ , tenemos que  $\lambda(x) = nf(x)$ ,  $\forall x \in N$

Si  $X$  es un proceso de Poisson con función intensidad  $\lambda(u)$ , entonces la localización de los puntos en el grafo está independiente e idénticamente distribuida con densidad de probabilidad:

$$f(u) = \frac{\lambda(u)}{\int_N \lambda(u) du}$$

A veces usar la intensidad en vez de la densidad simplifica las cosas.

#### 3.4.2. Estimación tipo núcleo de la intensidad

Tenemos que en general, siendo  $x_1, \dots, x_n$  los datos:

$$\hat{\lambda}(u) = \sum_{i=1}^n K(u|x_i), \quad u \in N, \quad K(\cdot|x) \text{ función núcleo con origen en } x.$$

Hay muchas proposiciones para este estimador, pero todavía no hay ningún consenso de cual es la mejor elección, de hecho hay algunos que no cumplen los requerimientos básicos, como el de que la función núcleo,  $K(\cdot|x)$ , sea una densidad de probabilidad  $\forall x \in N$ , es decir,  $K(\cdot|x) \geq 0$  y  $\int_N K(u|x) du = 1 \forall x \in N$ . Si esto se cumple, entonces tendríamos que  $\int_N \hat{f}(u) du = 1$  y  $\int_N \hat{\lambda}(u) du = n$ .

Para una intensidad uniforme,  $\lambda(u) \equiv \lambda$ , se tiene que  $\hat{\lambda}(u)$  es insesgado si y solo si  $\int_N K(u|x)dx = 1 \forall u \in N$ , y una condición suficiente (no necesaria) para esto sería la simetría de  $K$ ,  $K(x|u) = K(u|x)$ . El problema es que la intensidad no suele ser constante.

### Estimador de difusión

La estimación de la densidad en un grafo lineal está muy relacionada con la difusión en el grafo. La densidad de probabilidad del movimiento Browniano en un grafo lineal en un tiempo  $t$ ,  $f_t(x)$ ,  $x \in N$ , satisface la ecuación clásica del calor si  $x$  no es un vértice:

$$\frac{\partial f_t}{\partial t} = a \frac{\partial^2 f_t}{\partial x^2} \quad (3.2)$$

Además, en cualquier vértice  $v$ ,  $f_t$  es continua y cumple que

$$\sum_{v' \sim v} \frac{\partial f}{\partial x_{[v,v']}} \Big|_v = 0$$

donde el sumatorio es sobre todos los segmentos  $[v, v']$  que inciden en el vértice  $v$ , y  $\frac{\partial f}{\partial x_{[v,v']}} \Big|_v = \lim_{h \rightarrow 0} \frac{f(v+h(v'-v))}{h||v-v'||}$  es la primera derivada direccional de  $f$  en  $v$  con dirección hacia  $v'$ . Esto último es la conservación del flujo de calor, es decir, el calor no escapa del grafo, incluido en los vértices terminales.

La solución de (3.2) con la condición inicial  $f_0(x) = p(x)$ , se puede expresar como

$$f_t(x) = \int_N p(u)k_t(x|u)du$$

siendo  $k_t(x|u)$  el núcleo de calor en el grafo. Podemos pensar  $k_t(u|x)$  como la función de transferencia de temperatura desde el lugar  $u$  en el tiempo 0 hasta el lugar  $x$  en el tiempo  $t$ .

**Definición 3.3.** El estimador de difusión de la intensidad  $\lambda(u)$  en un grafo  $N$  con parámetro de suavizado  $h = \sqrt{t}$  es:

$$\hat{\lambda}_h(u) = \hat{\lambda}(u|X, h) = \sum_{x_i \in X} k_t(u|x_i), \quad u \in N$$

donde  $k_t(u|x_i)$  es el núcleo de calor en el grafo  $N$ .

**Teorema 3.4.** El núcleo de calor en un grafo lineal es:

$$k_t(u|x) = \sum_{\pi} a(\pi)\varphi_t(l(\pi))$$

donde  $\varphi_t$  es la densidad Gaussiana con varianza  $t = h^2$ , el sumatorio es a lo largo de todos los caminos de  $u$  a  $x$ , y  $a(\pi) = \prod_{v_i \in \pi} (\frac{2}{\deg(v_i)} - \delta_i)$  con  $\delta_i = 1_{\{l_i=l_{i-1}\}}$ ,  $0 < i < p$ .  $\varphi_t(l(\pi))$  se puede pensar como la densidad de probabilidad para que la difusión haya recorrido un desplazamiento total  $l(\pi)$  en el tiempo  $t$ , y  $a(\pi)$  es un peso combinatorio para el camino  $\pi$  (puede ser negativo o positivo).

El estimador de difusión tiene unas propiedades muy deseables:

1. Si se aplica el método de divisiones iguales continuo al núcleo normal, y se permite que se ejecute durante tiempo infinito, el resultado convergerá al núcleo de calor. Esto relaciona la estimación por difusión con otros métodos. En la práctica es útil cuando el parámetro de suavizado  $h = \sqrt{t}$  es muy pequeño, ya que el método de divisiones iguales continuo convergerá rápidamente.
2. Sea  $X$  un conjunto de puntos en el grafo  $N$ , con intensidad real  $\lambda(u)$ ,  $u \in N$ , y sea  $\hat{\lambda}(u)$  el estimador de difusión de la intensidad. Entonces,  $E[\hat{\lambda}(u)]$  es la solución en el tiempo  $t = h^2$  de la ecuación del calor en  $N$  con condición inicial  $\lambda_0(x) = \lambda(x)$ . Propiedad útil para los cálculos cuando la intensidad es conocida, y para la validación de los modelos.
3. Si la intensidad es constante,  $\lambda(u) \equiv \lambda$ , entonces el estimador de difusión  $\hat{\lambda}(u)$  es insesgado.
4. El núcleo de calor  $k_t(u|x)$  satisface que  $\int_N k_t(u|x) du = 1$ ,  $\forall x \in N$  y  $\int_N k_t(u|x) dx = 1$ ,  $\forall u \in N$ .
5. Si  $N$  es conexo, entonces:  $k_t(u|x)$  converge, cuando  $t$  tiende a infinito, a  $\frac{1}{|L|} = cte$ .
6. El núcleo de calor tiene la propiedad de semigrupo:  $\int_N k_s(y|u) k_t(u|x) du = k_{t+s}(y|x)$ ,  $\forall s, t > 0$ .
7. Propiedades asintóticas:  
Supongamos que hay  $n$  observaciones de una densidad  $f(u)$ , la cual se asume dos veces continuamente diferenciable. Ahora,  $n \rightarrow \infty$  y consideramos el estimador tipo núcleo  $\hat{f}$  con el núcleo de calor, con parámetro de suavizado  $h = h_n \rightarrow 0$  tal que  $nh_n \rightarrow \infty$ . Entonces, para cualquier localización  $u$  que no sea un nodo, el comportamiento de  $\hat{\lambda}(u)$  es asintóticamente equivalente al estimador tipo núcleo con núcleo Gaussiano sobre la recta real, por lo que  $\hat{\lambda}(u)$  es asintóticamente normal con:

$$Sesgo(\hat{\lambda}(u)) = E[\hat{f}(u) - f(u)] = \frac{h^2}{2} \frac{\partial^2 f(u)}{\partial u^2} + O(h^4) \quad (3.3)$$

$$\text{Var}(\hat{\lambda}(u)) = \text{Var}(\hat{f}(u)) = \frac{f(u)}{2\sqrt{\pi N}h} + o(1) \quad (3.4)$$

Además, si  $h = O(N^{-1/5})$ , entonces  $ECM(\hat{\lambda}) = O(N^{-4/5})$  y el estimador será puntualmente consistente.

### 3.4.3. Selección del suavizado para la estimación de la intensidad

Los métodos para la selección del parámetro de suavizado en el caso escalar se pueden extender a los grafos lineales, los comportamientos asintóticos para grandes muestras y suavizados pequeños son los mismos en un grafo lineal y en la recta real. Sin embargo, la complejidad computacional de los grafos lineales no permitirá usar algunas técnicas. Teóricamente, basándonos en el ECIMA y usando las fórmulas (3.3) y (3.4), el parámetro óptimo sería:

$$h_{ECIMA} = \left[ 2\sqrt{\pi}N \int_L \left( \frac{\partial^2 f(u)}{\partial u^2} \right)^2 du \right]^{-1/5}$$

Con una estimación “piloto“ de  $f$  se podría calcular, pero en la práctica, el método es muy sensible a esta elección.

### Validación cruzada para la estimación de la intensidad

No solo es el método de diferencias finitas más rápido para computar la estimación tipo núcleo de la intensidad para un suavizado dado, también computa las estimaciones para suavizados intermedios  $h_j = \sqrt{t_j}$  sin coste adicional.

La Validación Cruzada que deja un dato fuera busca un  $h$  que maximice la función:

$$A(h) = \sum_i \log(\hat{\lambda}_h^{-i}(x_i))$$

siendo  $\hat{\lambda}_h^{-i}(x_i)$  la estimación de  $\lambda(x_i)$  basada en todos los datos excepto en el  $x_i$ :

$$\hat{\lambda}_h^{-i}(x_i) = \hat{\lambda}(x_i | \mathbf{x} \setminus \{x_i\}, h) = \sum_{i \neq j} k_t(x_i | x_j) = \hat{\lambda}_h(x_i) - k_t(x_i | x_i)$$

La evaluación de  $A(h)$  para una sucesión de valores de  $h$  es factible en un grafo lineal, pero es más difícil computacionalmente que en la recta real, donde para un estimador tipo núcleo con un núcleo  $k$  y un suavizado fijado,  $\hat{\lambda}_{-i}(x_i) = \hat{\lambda}(x_i) - k(0)$  es más fácil de calcular. Por lo que para calcular  $A(h)$  en un grafo, hay que aplicar el algoritmo de diferencias finitas para cada punto de los datos por separado.

En general, el coste computacional de este método en los grafos lineales es especialmente alto.

### Métodos del pulgar

Sea un conjunto de datos de dimensión  $d$ , el **Método del pulgar de Scott** dice que el parámetro de suavizado para la coordenada  $i$  debería ser:

$$h_i = n^{\frac{-1}{d+1}} \cdot s_i$$

siendo  $s_i$  la desviación típica muestral de los valores de las coordenadas  $i$  de los datos.

Por otro lado, el **Método del pulgar de Silverman** dice que:

$$h_i = \left( \frac{4}{d+2} \right)^{\frac{1}{d+4}} n^{\frac{-1}{d+4}} \cdot s_i$$

En el caso de los grafos lineales utilizaríamos  $d = 2$  para ambos casos.

Otra alternativa sería aplicar la versión escalar de estos dos métodos a una proyección ortogonal de los puntos a un eje unidimensional, escogido para maximizar la desviación típica de las coordenadas proyectadas. La desviación típica maximizada sería  $s = \sqrt{a}$ , donde  $a$  es el autovalor más alto de la matriz de varianzas-covarianzas de la muestra. Nos quedarían:

$$\text{Scott : } h = n^{-1/5} \sqrt{a}; \text{ Silverman : } h = (4/3)^{1/5} n^{-1/5} \sqrt{a}$$

#### 3.4.4. Función riesgo relativo

Una vez hemos visto como estimar la intensidad, estamos listos para abarcar la estimación del riesgo relativo. Supongamos que hay dos patrones de puntos,  $X = \{x_1, \dots, x_m\}$  e  $Y = \{y_1, \dots, y_n\}$ , en el mismo grafo lineal  $N$ . Queremos estimar el *Riesgo Relativo Logarítmico*:

$$\rho(u) = \log \left( \frac{\lambda_X(u)}{\lambda_Y(u)} \right), \quad u \in N$$

donde  $\lambda_X$  y  $\lambda_Y$  son las intensidades de los eventos de  $X$  y de  $Y$  respectivamente.

Tenemos que el estimador plug-in de esta función sería:

$$\hat{\rho}(u) = \hat{\rho}_{h_1, h_2}(u) = \log \left( \frac{\hat{\lambda}_X(u)}{\hat{\lambda}_Y(u)} \right)$$

siendo  $\hat{\lambda}_X(u) = \hat{\lambda}(u|X, h_1)$  y  $\hat{\lambda}_Y(u) = \hat{\lambda}(u|Y, h_2)$  los estimadores de difusión de  $\lambda_X$  y  $\lambda_Y$  computados con parámetros de suavizado  $h_1$  y  $h_2$  respectivamente. Teniendo todo esto, ahora habría que atacar el problema de la selección del parámetro de suavizado para esta estimación.

### 3.4.5. Selección del parámetro de suavizado para el riesgo relativo

Observando la definición del riesgo relativo anterior, tenemos tres opciones:

1. Escoger por separado,  $h_1$  basado solo en los datos de  $X$ , y  $h_2$  basado solo en los datos de  $Y$ .
2. Hacer una selección conjunta de  $(h_1, h_2)$  basada en  $X$  e  $Y$ .
3. Selección simétrica de un parámetro común  $h = h_1 = h_2$  basado en  $X$  e  $Y$ .

Para el primer método se puede utilizar cualquier método de selección del suavizado descrito en capítulos anteriores. Para los otros dos métodos requerimos técnicas nuevas.

#### Validación cruzada de Kelsall y Diggle

Kelsall y Diggle proponen un método para seleccionar los parámetros  $(h_1, h_2)$  en eventos bidimensionales, con ó sin restricción  $h_1 = h_2$ , por validación cruzada basado en el ECI. Adaptándolo al grafo lineal, les queda que el par  $(h_1, h_2)$  ó  $(h, h)$  que se escoge, será el que minimice la función:

$$\tilde{C}_{KD}(h_1, h_2) = - \int_L \left[ \log \frac{\hat{\lambda}_X(u)}{\hat{\lambda}_Y(u)} \right]^2 du - 2 \sum_{i=1}^m \frac{1}{\hat{\lambda}_X^{-i}(x_i)} \log \left( \frac{\hat{\lambda}_X^{-i}(x_i)}{\hat{\lambda}_Y(x_i)} \right) - 2 \sum_{j=1}^n \frac{1}{\hat{\lambda}_Y^{-j}(y_j)} \log \left( \frac{\hat{\lambda}_Y^{-j}(y_j)}{\hat{\lambda}_X(y_j)} \right)$$

con  $\hat{\lambda}_X(u)$  y  $\hat{\lambda}_Y(u)$  las estimaciones de  $\lambda_X(u)$  usando  $h_1$  y de  $\lambda_Y(u)$  usando  $h_2$  respectivamente, y  $\hat{\lambda}_X^{-i}$  y  $\hat{\lambda}_Y^{-j}$  las estimaciones de  $\lambda_X$  y  $\lambda_Y$  sin los datos  $x_i$  e  $y_j$  respectivamente.

#### Validación cruzada de probabilidad y por mínimos cuadrados

Sean  $X$  e  $Y$  eventos de Poisson independientes en  $\mathbb{R}^2$  con intensidades  $\lambda_X$  y  $\lambda_Y$  respectivamente.  $Z = X \cup Y$  también será un evento con distribución Poisson con intensidad  $\lambda_Z(u) = \lambda_X(u) + \lambda_Y(u)$ . Un punto aleatorio de  $Z$  en  $u$  tiene probabilidad  $p(u) = \frac{\lambda_X(u)}{\lambda_Z(u)}$  de ser originado en  $X$  en vez de en  $Y$ . Entonces, para cada  $x_i \in X$ ,  $i = 1, \dots, m$ :

$$\hat{p}_i = \frac{\hat{\lambda}_X^{-i}(x_i)}{\hat{\lambda}_X^{-i}(x_i) + \hat{\lambda}_Y(x_i)}$$

es la probabilidad estimada de que un punto de  $X \cup Y$  localizado en  $x_i$  pertenezca a  $X$  en vez de a  $Y$ . De forma análoga, tenemos que para cada  $y_j \in Y$ ,  $j = 1, \dots, n$ :

$$\hat{q}_j = \frac{\hat{\lambda}_Y^{-j}(y_j)}{\hat{\lambda}_X(y_j) + \hat{\lambda}_Y^{-j}(y_j)}$$

es la probabilidad estimada de que un punto de  $X \cup Y$  en  $y_j$  pertenezca a  $Y$  en lugar de a  $X$ . Por tanto, el método de validación cruzada de probabilidad consistirá en minimizar:

$$\tilde{C}_{VCP}(h_1, h_2) = - \left[ \sum_{i=1}^m \log(\hat{p}_i) + \sum_{j=1}^n \log(\hat{q}_j) \right]$$

Por otro lado, el método de validación cruzada por mínimos cuadrados consistirá en minimizar:

$$\tilde{C}_{VCMC}(h_1, h_2) = \sum_{i=1}^m (1 - \hat{p}_i)^2 + \sum_{j=1}^n (1 - \hat{q}_j)^2$$

Se sabe que este último método funciona en muchos contextos.

## Capítulo 4

# Grafos lineales en R

Una vez hayamos visto e interiorizado toda esta teoría sobre la estimación tipo núcleo en grafos lineales, es el momento de mostrar algunos ejemplos de como se trabaja este tema en el programa R (ó RStudio). Como ya hemos dicho, este campo ha empezado a avanzar mucho desde que los datos son más accesibles gracias a las herramientas informáticas desarrolladas e internet, y también son más fáciles de estudiar gracias a programas como R. Una de las herramientas que se utiliza mucho para conseguir y tratar los datos son los GIS (Geographical Information System), que son un conjunto de herramientas que permiten la organización, almacenamiento, manipulación, análisis y modelización de grandes cantidades de datos procedentes del mundo real, vinculados a una referencia espacial. Esta herramienta es usada no solo por científicos, sino por gente de campos completamente diferentes, a quienes les permite tomar decisiones de forma más eficaz. En este trabajo no se van a tomar datos de ningún GIS, este capítulo se centra en presentar las herramientas básicas de R para tratar los grafos lineales.

### 4.1. Nociones básicas

Para empezar, vamos a ver como construir un grafo lineal. Creamos el conjunto de vértices como un “ppp” (*Planar Point Patern*), y el conjunto de aristas como un “im” (*Pixel Image*). A continuación creamos el grafo lineal con esos dos conjuntos como un “linnet” (*Linear Network*).

```
v <- ppp(x=(-2):2, y=3*c(0,1,2,1,0), c(-3,3), c(-1,7))
a <- matrix(c(1,2,3,4,2,2,3,4,5,4), ncol=2)
letraA <- linnet(v, edges=a)
```

Una vez lo hemos creado, podemos obtener un pequeño resumen del grafo de la siguiente manera:

```
summary(letraA)
```

Como vemos en 4.1, este resumen nos da información como el número de vértices y aristas del grafo, la distancia total de las aristas del grafo, el grado máximo de los vértices, si el grafo es conexo o no, las medidas del rectángulo donde está contenido el grafo, etc. También podemos representar el grafo con:

```
plot(letraA); text(vertices(letraA), vertexdegree(letraA))
```

Así, vemos en 4.2 el grafo lineal representado y el grado de cada vértice. A parte de esta forma de construir grafos lineales, podemos transformar otros objetos a la clase “linnet” con el comando *as.linnet()*. En la tabla 4.3 tenemos todas las funciones para manipular y obtener información sobre los objetos de la clase “linnet”.

Una vez tenemos construido el grafo, podemos crear sobre él diferentes patrones de puntos:

```
xx <- list(x=c(-1.5,0,0.5,1.5), y=c(1.5,3,4.5,1.5))
X <- lpp(xx, letraA)
Y <- rpoislpp(1, letraA)
#puntos aleatorios con distribucion Poisson(1)
Z <- runiflpp(10, letraA)
#10 puntos aleatorios de forma uniforme
```

Y obtenemos la representación de cada uno como se observa en 4.4 con el siguiente código:

```
par(mfrow=c(1,3))
plot(X); text(vertices(letraA), vertexdegree(letraA))
plot(Y)
plot(Z)
```

Una vez tenemos construidos los patrones de puntos en grafos lineales, veamos como podemos estudiarlos y manipularlos. Ahora vamos a trabajar sobre tres conjuntos de patrones de puntos en tres grafos lineales diferentes (datos propios de R). El primero se corresponde con siete acciones criminales diferentes en torno a una zona cercana a la Universidad de Chicago. El segundo es la situación de un conjunto de telarañas sobre una pared de ladrillos. El tercero se corresponde con tres tipos de protuberancias a lo largo de una red neuronal. Como observamos en 4.5 estamos ya ante tres “lpp” (*Point Patern on a Linear Network*). En las figuras 4.6, 4.7 y 4.8 vemos representados estos tres patrones en sus grafos correspondientes. Además añadimos para el lector la tabla 4.9 correspondiente a operacio-

nes básicas que se pueden aplicar a un “lpp“, y las tablas 4.10 y 4.11 correspondientes a funciones para la manipulación y análisis de los patrones de puntos en grafos lineales.

## 4.2. Intensidad en grafos lineales

Como ya hemos observado en la sección 3.4.1., la estimación de la intensidad está muy relacionada con la estimación de la función densidad, ya que si fijamos el número de eventos  $n$ , tenemos que la función intensidad cumple:  $\lambda(x) = nf(x)$  para todo punto del grafo. A veces estimar la intensidad facilitará las cosas, por lo que en este capítulo nos centraremos en estimar la intensidad de un patrón de puntos en un grafo lineal (“lpp“).

Para el caso en el que se quiera estimar una intensidad homogénea, tenemos el comando `intensity()`, que sería equivalente a dividir el número de puntos en el grafo entre la distancia total del grafo.

```
n=npoints(spiders)
d=volume(as.linnet(spiders))
intensity(spiders) # =n/d
```

También podemos cambiar de escala el “lpp“ para poder dar la intensidad en unidades más cómodas, ya que recordemos que la intensidad es el número de eventos por unidad de distancia.

```
spidersm <- rescale(spiders,1000,c("metros","metros"))
intensity(spidersm)
```

Recordemos también que este valor es uno de los datos que nos da el comando `summary()`. Aquí ya podemos comprobar la importancia de no tratar el patrón de puntos en el grafo como un patrón de puntos en el plano. Si ignoramos la estructura de grafo con el siguiente código:

```
intensity(as.ppp(spidersm)) # =n/(1.125^2)
```

pasamos de tener 2.37 telarañas por metro a tener 37.93 telarañas por metro cuadrado, lo cual es una sobreestimación.

Sin embargo, la intensidad no suele ser uniforme, y para ello R lleva a cabo una estimación tipo núcleo con el comando `density()`, el cual utiliza por defecto el núcleo Gaussiano, pero se puede escoger otro.

```
d60 <- density(unmark(chicago), 60)
#parametro de escala: 60 pies
plot(d60)
```

```

    plot(d60, style = "width")
d10 <- density(dendrite,2) #parametro de escala: 2 micras
    plot(d10)

```

Aquí hemos estimado la intensidad en cada caso sin tener en cuenta los diferentes tipos de eventos que hay en cada grafo y con núcleo Gaussiano, escogiendo al azar el parámetro de escala. De esta forma obtenemos las representaciones de las intensidades 4.12 y 4.13.

Por último, también podemos estimar la intensidad teniendo en cuenta los diferentes tipos de eventos en un mismo grafo. Si utilizamos el comando *intensity(chicago)* obtendremos las estimaciones de las intensidades homogéneas de cada tipo de crimen en el grafo. Esta información también la obtenemos con *summary(chicago)*. También podemos cambiar la escala de la longitud en el grafo para obtener datos más sencillos de cada tipo de evento en el grafo:

```

chicagokm <- rescale(chicago, 3280.84, c("km", "km"))
summary(chicagokm)

```

También podemos usar la estimación tipo núcleo para cada tipo de evento:

```

plot(density(split(chicago), sigma = 20))

```

de esta forma observamos en 4.14 las estimaciones de las densidades de los diferentes tipos de eventos en un mismo grafo.

### 4.3. Covariables

En ocasiones nos interesará analizar el patrón de puntos en el grafo introduciendo lo que llamamos una *Covariable espacial*, la cual se entiende por una cantidad que se puede medir en cualquier lugar del grafo. Por ejemplo, covariables en un patrón de accidentes en una red de carreteras podrían ser propiedades de la carretera como su ancho, el límite de velocidad, la pendiente, la distancia al accidente más cercano, etc.

En R, las covariables están representadas por objetos de las clases “*lfnun*” (*Function in a Linear Network*) y “*linim*” (*Pixel Image on a Linear Network*).

A modo de ejemplo, vamos a representar como un “*linim*” los puntos correspondientes a *assault* en el grafo de “*chicago*”. A continuación modificamos ese objeto y lo representamos también.

```

dna <- distfun(split(chicago)$assault)
Dna <- as.linim(dna)
plot(Dna)

```

```
a <- sqrt(Dna) + 3
plot(a, style="color")
```

El resultado es la figura 4.15.

Como ejemplo de una “linfun“ vamos a crear una función en el grafo lineal que indique si la arista en la que se encuentra cada punto del grafo es horizontal o vertical. Para ello utilizamos los siguientes comandos:

```
ladrillos <- domain(spiders)
alpha <- angles.psp(as.psp(ladrillos)) * 180/pi
mortarvert <- (round(alpha/90) == 1)
f <- function(x, y, seg, tp) { mortarvert[seg] }
vertical <- linfun(f, ladrillos)
```

El resultado es la función *vertical*, la cual se puede aplicar a los patrones de puntos en un grafo:

```
vertical(spiders[5:8])
[1] FALSE FALSE TRUE TRUE
which(vertical(spiders))
[1] 7 8
table(vertical(spiders))
FALSE TRUE
46      2
```

Para terminar, vamos a utilizar ese último ejemplo para ver como estimar la intensidad dependiendo de una covariable:

```
mortlen <- lengths_psp(as.psp(ladrillos))
totlen <- tapply(mortlen, mortarvert, sum)/1000
totlen
FALSE      TRUE
14.62500   5.59375
#Segmentos verticales: 5.59 metros
#Segmentos horizontales: 14.62 metros
totpts <- table(vertical(spiders))
totpts
FALSE TRUE
46     2
#Segmentos verticales: 2 telas
```

```
#Segmentos horizontales: 46 telas  
totpts/totlen  
FALSE      TRUE  
3.1452991 0.3575419  
#Segmentos verticales: 0.3575419 telas por metro  
#Segmentos horizontales: 3.1452991 telas por metro
```

## Anexos para el capítulo 4

```
Linear network with 5 vertices and 5 lines
Total length 14.64911 units
Maximum vertex degree: 3
[Non-sparse matrix representation]
Network is connected
Diameter: 8.324555 units
Bounding radius: 5.162278 units
Numerical tolerance: 0.002 units
Enclosing window: rectangle = [-3, 3] x [-1, 7] units
```

Figura 4.1: Resumen de las características del grafo lineal “letraA”.

---

<code>L; print(L)</code>	print basic information
<code>summary(L)</code>	print summary information
<code>plot(L)</code>	plot network
<code>iplot(L)</code>	interactively plot network
<code>vertices(L)</code>	extract vertices (nodes) of network
<code>as.psp(L)</code>	extract line segments of network
<code>as.owin(L)</code>	extract window containing network
<code>L[W]</code>	sub-network inside window W
<code>unitname(L)</code>	extract unit of length
<code>unitname(L) &lt;- u</code>	change unit of length
<code>rescale(L, s)</code>	convert to different unit of length
<code>volume(L)</code>	total length of network
<code>diameter(L)</code>	maximum network distance between two locations
<code>circumradius(L)</code>	maximum network distance from central location
<code>vertexdegree(L)</code>	degree of each vertex
<code>rotate(L, angle)</code>	rotate entire network
<code>shift(L, vec)</code>	shift (translate) entire network
<code>scalardilate(L, f)</code>	apply scale change to entire network
<code>affine(L, mat, vec)</code>	apply affine transformation to network

---

Figura 4.3: Tabla de funciones para manipular los “linnet”.

### letraA

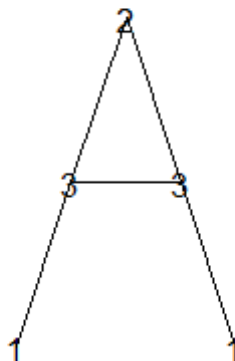


Figura 4.2: Representación del grafo lineal “letraA”.

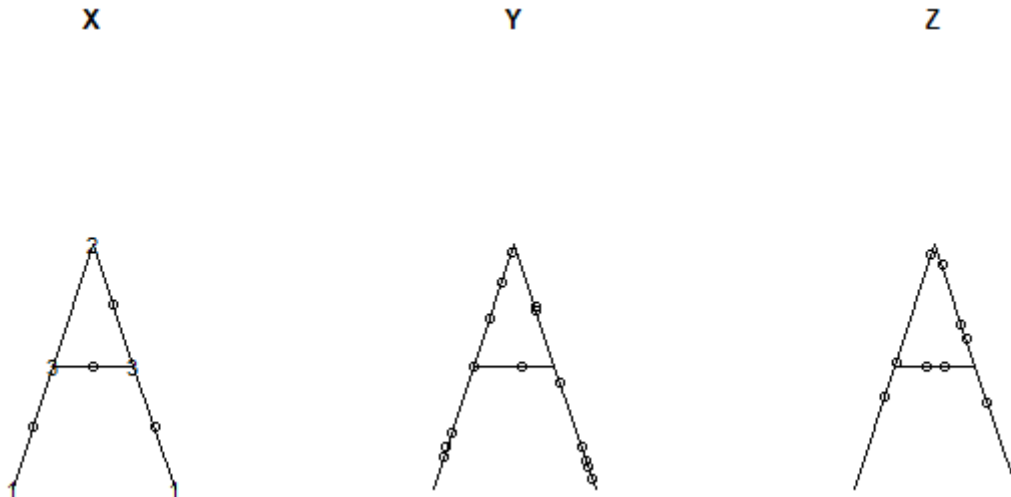


Figura 4.4: Representación de diferentes patrones puntuales en “letraA”.

```

> chicago
Point pattern on linear network
116 points
Multitype, with possible types:
  assault burglary cartheft damage robbery theft trespass
Linear network with 338 vertices and 503 lines
Enclosing window: rectangle = [0.3894, 1281.9863] x [153.1035, 1276.5602] feet
> spiders
Point pattern on linear network
48 points
Linear network with 156 vertices and 203 lines
Enclosing window: rectangle = [0, 1125] x [0, 1125] mm
> dendrite
Point pattern on linear network
566 points
Multitype, with possible types: mushroom, stubby, thin
Linear network with 640 vertices and 639 lines
Enclosing window: rectangle = [0, 243.4] x [130, 317] microns

```

Figura 4.5: Tres patrones de puntos sobre grafos lineales.

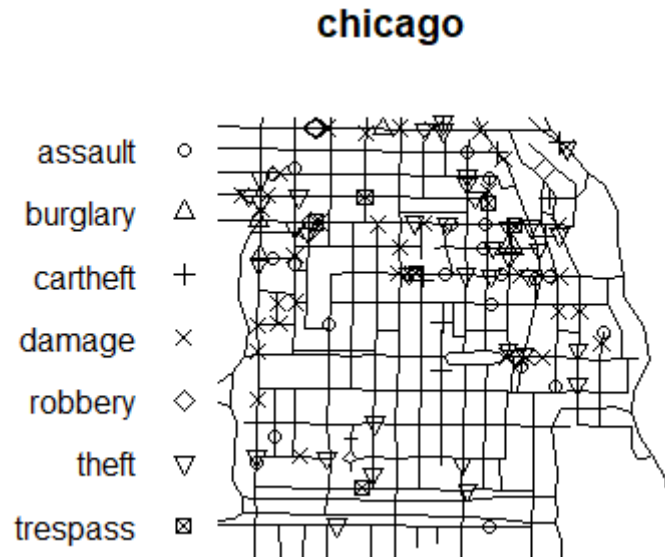


Figura 4.6: Siete tipos de crímenes en Chicago.

### spiders

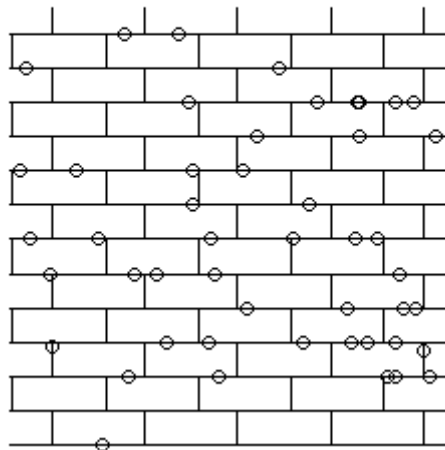


Figura 4.7: Telarañas sobre una pared de ladrillos.

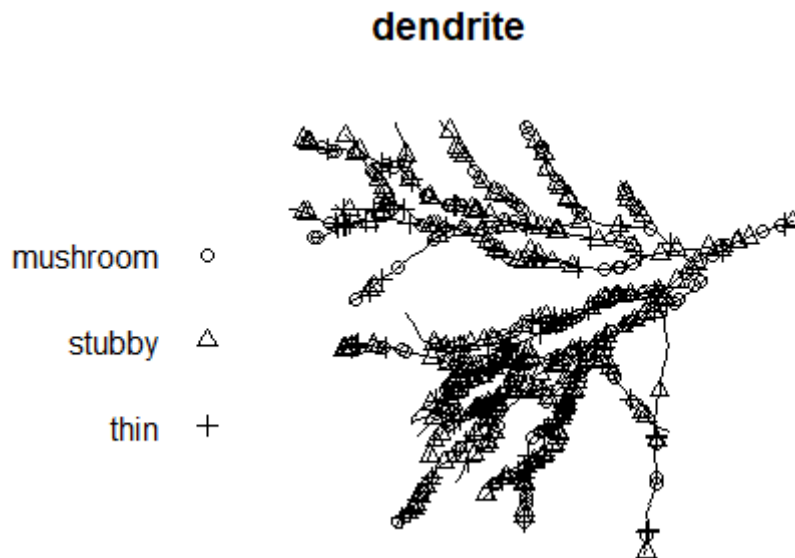


Figura 4.8: Tres tipos de protuberancias en una red neuronal.

---

<code>X; print(X)</code>	print basic information
<code>summary(X)</code>	print summary information
<code>plot(X)</code>	plot point pattern and network
<code>iplot(X)</code>	interactively plot point pattern and network
<code>as.linnet(X); domain(X)</code>	extract linear network
<code>as.owin(X); Window(X)</code>	extract containing window
<code>as.ppp(X)</code>	extract points in two dimensions
<code>as.psp(X)</code>	extract line segments
<code>coords(X)</code>	extract spatial and local coordinates
<code>coords(X) &lt;- value</code>	assign new coordinates
<code>marks(X)</code>	extract marks
<code>marks(X) &lt;- value</code>	assign new marks
<code>unmark(X)</code>	remove marks
<code>is.multitype(X)</code>	check whether pattern is multitype
<code>as.hyperframe(X)</code>	extract coordinates and marks
<code>as.matrix(X)</code>	extract coordinates and numerical marks
<code>as.data.frame(X)</code>	extract coordinates and marks
<code>X[W]</code>	subset inside window W
<code>X[i]</code>	subset of points selected by index i

---

Figura 4.9: Operaciones básicas para un “lpp”.

<code>anyDuplicated(X)</code>	determine whether any points are identical
<code>duplicated(X)</code>	determine which points are duplicated
<code>multiplicity(X)</code>	count number of duplicates of each point
<code>unique(X)</code>	remove duplicated points
<code>subset(X, subset, select)</code>	subset specified by conditions
<code>unitname(X)</code>	extract name of unit of length
<code>unitname(X) &lt;- value</code>	change name of unit of length
<code>superimpose(X,Y,...)</code>	combine several point patterns
<code>split(X, f)</code>	divide point pattern into several point patterns
<code>rescale(X, s)</code>	convert to another unit of length
<code>rotate(X, angle)</code>	rotate the entire spatial dataset
<code>shift(X, vec)</code>	shift the entire spatial dataset
<code>affine(X, mat, vec)</code>	apply affine transformation
<code>scalardilate(X, f)</code>	multiply spatial coordinates by a factor
<code>round(X, digits)</code>	applying rounding to spatial coordinates
<code>rlabel(X)</code>	permute the marks

Figura 4.10: Funciones útiles para manipular un “lpp”.

<code>npoints(X)</code>	number of points
<code>intensity(X)</code>	number of points per unit length
<code>pairdist(X)</code>	shortest-path distances between pairs of points
<code>nndist(X)</code>	shortest-path distance to nearest neighbour
<code>crossdist(X,Y)</code>	shortest-path distances between pairs of points
<code>nncross(X, Y)</code>	shortest-path distance to nearest neighbour
<code>distfun(X)</code>	distance function (shortest-path distance)
<code>nnwhich(X)</code>	identify which point is nearest neighbour
<code>nnfun(X)</code>	nearest-neighbour index function
<code>rounding(X)</code>	guess the amount of rounding in spatial coordinates
<code>envelope(X, Fun)</code>	simulation envelopes of summary function

Figura 4.11: Funciones básicas para el análisis de un “lpp”.

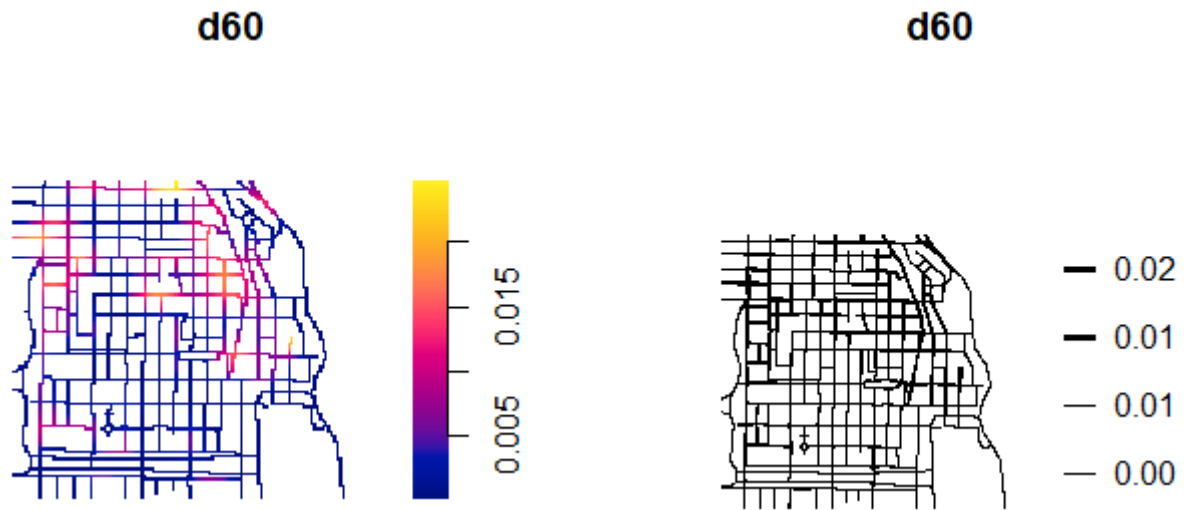


Figura 4.12: Dos tipos de representación de la intensidad en el objeto “chicago”.

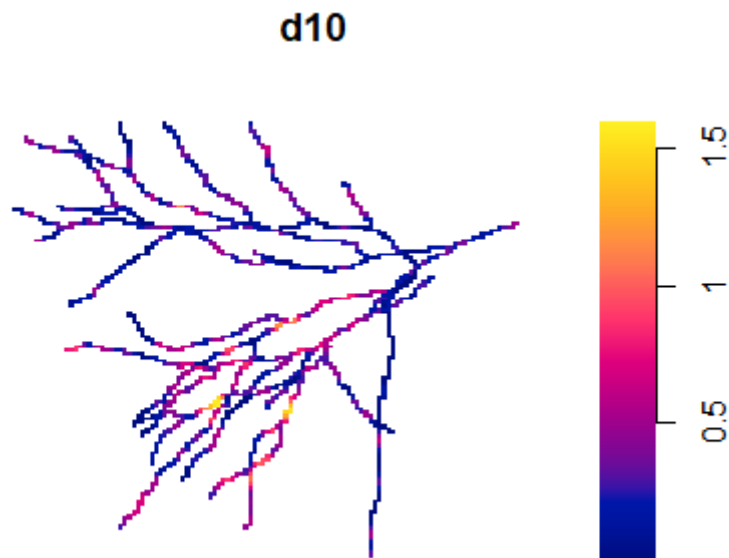


Figura 4.13: Intensidad en el objeto “dendrite”.

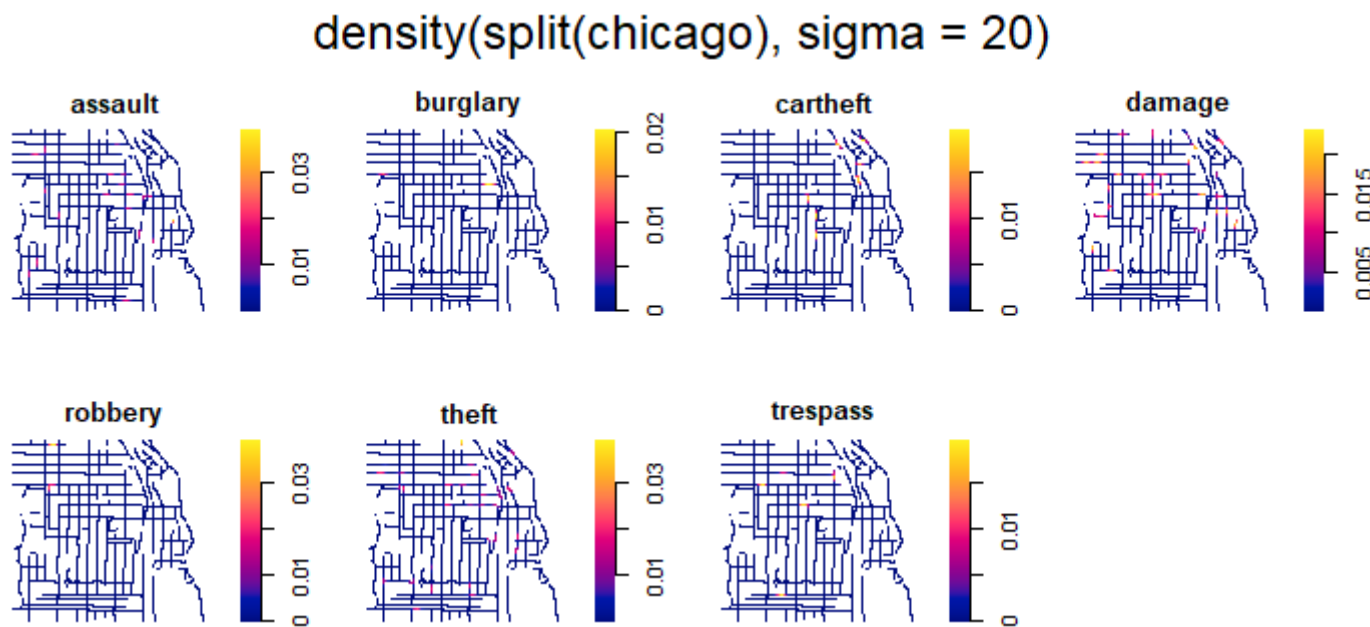


Figura 4.14: Intensidades de diferentes eventos en un mismo grafo.

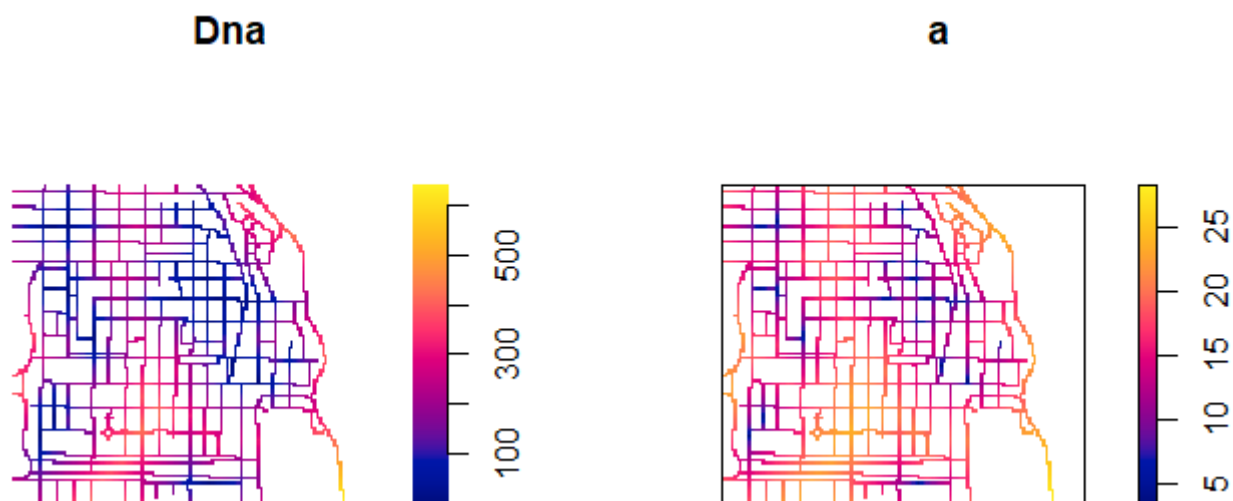


Figura 4.15: Representación de una “linim” y una modificación de la misma.

# Bibliografía

- [1] Wand, M. P. and Jones M. C., *Kernel Smoothing*, 1st ed., Monographs on Statistics and Applied Probability, 60, Springer-Science+Business Media, B. V., New York, 1995.
- [2] Zhixiao Xie and Jun Yan, *Kernel Density Estimation of traffic accidents in a network space*, Computers, Environment and Urban Systems, 32 (2008), 396–406.
- [3] Atsuyuki Okabe, Toshiaki Satoh and Kokichi Sugihara *A kernel density estimation method for networks, its computational method and a GIS-based tool*, International Journal of Geographical Information Science, Vol. 23 No. 1 (January 2009), 7–32.
- [4] Qi Wei Ang, Adrian Baddeley and Gopalan Nair, *Geometrically Corrected Second Order Analysis of Events on a Linear Network, with Applications to Ecology and Criminology*, Scandinavian Journal of Statistics, 39 (2012), 591-617.
- [5] Greg McSwiggan, Adrian Baddeley and Gopalan Nair, *Kernel Density Estimation on a Linear Network*, Scandinavian Journal of Statistics, 44 (2017), 324–345.
- [6] Greg McSwiggan, Adrian Baddeley and Gopalan Nair, *Estimation of relative risk for events on a linear network*, Statistics and Computing, 30 (2020), 469–484.
- [7] Adrian Baddeley, Ege Rubak and Rolf Turner, *Spatial Point Patterns. Methodology and Applications with R*, Chapman and Hall/CRC, Interdisciplinary Statistics Series.