Estadística no paramétrica

Regresión no paramétrica III: Modelos aditivos y splines

Jorge de la Vega Góngora

Departamento de Estadística,
Instituto Tecnológico Autónomo de México

14 de abril de 2023

Splines

Introducción I

- Los splines son funciones polinomiales en segmentos que se concatenan para interpolar o aproximar las gráficas de dispersión de puntos generados por pares $(X_1, Y_1), ... (X_n, Y_n)$ de puntos.
- Fueron propuestos por Isaac Jacob Schoenberg (1903–1990) a partir de 1963.

(No confundir con Isaac Schoenberg, ingeniero inventor de la televisión).

- Consideraremos dos tipos de splines en estas notas: splines para interpolar y splines para suavizar funciones.

Splines para interpolación I

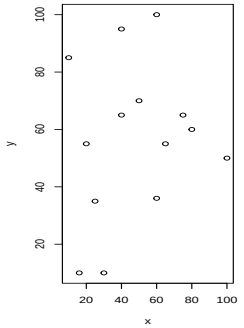
- Supongamos que $X_1 \leq X_2 \leq ... \leq X_n \in [a, b]$ son valores ordenados que se refieren como *nodos*. En cada intervalo $[X_{i-1}, X_i]$ para $i = 1, 2, ..., n + 1$, con $X_0 = a$ y $X_{n+1} = b$, un spline $s(x)$ es un polinomio de grado menor o igual a k.
- Usualmente se considera $k = 3$ porque tienen propiedades que son útiles en los extremos. Las piexas polinomiales se conectan de tal forma que las segundas derivadas son continuas: entonces en los nodos, los polinomios tienen tangente y curvatura común.
- Se dice que $s \in \mathcal{L}^2[1, b]$ que es la clase de funciones en $[a, b]$ con segunda derivada continua. El spline cúbico es *natural* si las piezas en los intervalos $[a, X_1]$ y $[X_n, b]$ son de grado 1. Las siguientes dos propiedades distinguen a los splines cúbicos naturales de otras funciones en $\mathcal{L}^2[1, b]$:
  1. **Interpolación única**: Dados los n pares, $(X_1, Y_1), ..., (X_n, Y_n)$, con distintos nodos X_i, hay un spline natural cúbico *único* s que interpola los puntos, i.e. $s(X_i) = Y_i$.
  2. **Propiedad extremal**: Dados n pares, $(X_1, Y_1), ..., (X_n, Y_n)$, con nodos distintos y ordenados X_i, el spline cúbico natural $s(x)$ que interpola los puntos también minimiza la curvatura en el intervalo $[a, b]$, donde $a < X_1$ y $X_n < b$. Formalmente, para cualquier función $g \in \mathcal{L}^2[1, b]$, se cumple:

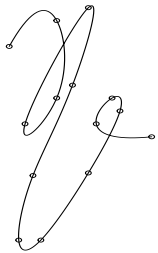
$$\int_a^b (s''(t))^2 \, dt \leq \int_a^b (g''(t))^2 \, dt$$

Splines para interpolación II

Ejemplo. [dibujando una letra v]

```
par(mfrow=c(1,2))
x <- c(10, 40, 40, 20, 60, 50, 25, 16, 30, 60, 80, 75, 65, 100)
y <- c(85, 95, 65, 55, 100, 70, 35, 10, 10, 36, 60, 65, 55, 50)
plot(x,y)
u <- 1:length(x)
uu <- seq(1, length(u), length = 250)
# funciones para ajustar splines interpolantes: splinefun
fit1 <- splinefun(u,x); xx <- fit1(uu)
fit2 <- splinefun(u,y); yy <- fit2(uu)
plot(x,y, axes = F, xlab = "", ylab = "", ylim=c(0, 100), xlim=c(0, 100))
lines(xx, yy, type = "l")
```



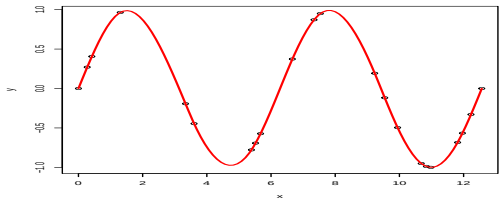


□

Splines para interpolación III

Ejemplo. [Interpolación de una función sinusoidal]

```r
x <- 4*pi*c(0, 1, runif(20))
y <- sin(x)
fit <- splinefun(x,y)
xx <- seq(0, max(x), length = 100) # nodos para ajuste
yy <- fit(xx)
plot(x, y)
lines(xx, yy, lwd = 3, col = "red")
```



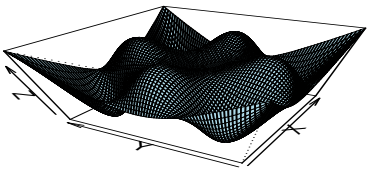
□

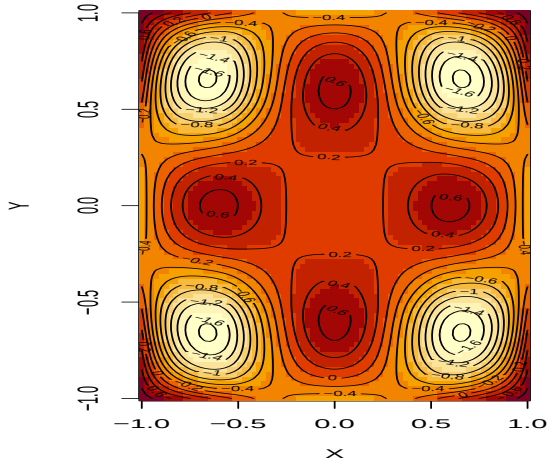
Ejemplo: interpolación en dos dimensiones I

Una superficie de interpolación para splines bidimensionales se muestra a continuación

```
library(akima) # uso de la función bicubic
# Grid y evaluación de la función en el grid
x <- seq(-1, 1, by = 0.50)
y <- seq(-1, 1, by = 0.50)
z <- outer(x, y, function(x,y){ sin(10*(x^2 + y^2)) })

# creación del grid
xy <- expand.grid(seq(-1, 1, length = 80), seq(-1, 1, length = 80))
fit <- bicubic(x, y, z, xy[,1], xy[,2])
xx <- seq(-1, 1, length = 80)
yy <- seq(-1, 1, length = 80)
zz <- matrix(fit$z, nrow = 80)

par(mfrow=c(1,2))

#grafica de superficie:
persp(x = xx, y = yy, z = zz, col = "lightblue", phi = 45, theta = -60,
xlab = "X", ylab = "Y", zlab = "Z");

#gráfica de contorno:
image(x = seq(-1, 1, length = 80), y = seq(-1, 1, length = 80),
z = matrix(fit$z, nrow = 80), xlab = "X", ylab = "Y")
contour(x = seq(-1, 1, length = 80), y = seq(-1, 1, length = 80),
z = matrix(fit$z, nrow = 80), add = TRUE)
```

Ejemplo: interpolación en dos dimensiones II





Splines de suavizamiento I

- Los *splines de suavizamiento* son una forma de regresión no paramétrica, que se utiliza como suavizadores. Se tienen pares de puntos (X_i, Y_i) para $i = 1, 2, \dots, n$. La función continuamente diferenciable $\hat{s}$ en $[a, b]$ que minimiza la funcional:

$$T(s) = \sum_{i=1}^{n} (Y_i - s(X_i))^2 + \lambda \int_a^b (s''(t))^2 \, dt$$

  es exactamente un spline cúbico natural.

- El costo funcional tiene dos partes:
  1. $\sum_{i=1}^{n} (Y_i - s(X_i))^2$ se minimiza por el polinomio de interpolación, y
  2. $\int_a^b (s''(t))^2 \, dt$ se minimiza por una línea recta.

  El parámetro λ intercambia la importancia de estos costos competitivos: si λ es pequeño, la solución es un spline de interpolación y si λ es grande, la minimizante es una línea recta.

Modelos Aditivos

Introducción I

- Un modelo no paramétrico aditivo para regresión está dado por la función

$$Y_i|\mathbf{X}_i = \alpha + f_1(X_{i1}) + \cdots + f_p(X_{ip}) + \epsilon_i$$

donde las funciones f_i se conocen como *funciones parciales de regresión* y son las que se deben estimar a partir de los datos y los errores se asumen con $\mathrm{E}(\epsilon) = \mathbf{0}$ y $\mathrm{Var}(\epsilon) = \sigma^2\mathbf{I}$.

- También se podrían incluir funciones bi o tri-variadas como parte del modelo, por ejemplo:

$$Y_i|\mathbf{X}_i = \alpha + f_1(X_{i1}) + f_2(X_{i2}) + f_3(X_{i3}, X_{i4}) + \cdots + \epsilon_i$$

- Los *modelos semiparamétricos* es una extensión de los modelos aditivos, donde sólo parte de los predictores se modelan de manera no paramétrica:

$$Y = \beta'\mathbf{X} + \sum_{j=1}^{q} f_j(Z_j) + \epsilon$$

Introducción II

- El modelo aditivo es más restrictivo que el modelo de regresión multivariado $Y_i|\mathbf{X}_i = m(\mathbf{X}_i) + \epsilon_i$, que, en teoría, podría estimarse utilizando por ejemplo, el esquema de vecinos cercanos y aplicar loess. Pero el modelo general presenta algunas dificultades:
  - Dimensionalidad: los datos vecinos se vuelven más escasos con el aumento de dimensiones. El número de observaciones vecinas declina.
  - Visualización e interpretación.
- La ventaja de un modelo aditivo es que nos da una forma muy efectiva de ajustar una función no lineal de varias variables y generar las gráficas de cada una y estudiar los efectos de cada variable en la respuesta.

Ajuste de un modelo de regresión aditivo.

- El modelo aditivo da la impresión de que las variables son *independientes* entre sí, y si ese fuera el caso, podríamos ajustar una función separada para cada par Y y X_k.
- Supongamos que las $X's$ están relacionadas, pero que conocemos las funciones parciales de regresión f_k, excepto para $k = 1$. Entonces podríamos formar el residual parcial:

$$Y_{(1)i} \equiv Y_i - [f_2(X_{i2}) + \cdots + f_p(X_{ip})] = \alpha + f_1(X_{i1}) + \epsilon_i$$

- Ignorando el nivel constante α, el suavizamiento de $Y_{(1)}$ contra X_1 da un estimado de f_1.
- En la práctica, no conocemos todas las funciones parciales de regresión, pero podemos aplicar el siguiente algoritmo recursivo conocido como *backfitting* (o algoritmo de Gauss-Seidel en análisis numérico).

Ajuste de un modelo de regresión aditivo: *backfitting* I

1. Hacer $l = 0$. Encontrar estimadores iniciales de as funciones de regresión parcial $\hat{f}_i^{(0)}$. Por ejemplo, podrían ser las funciones de mínimos cuadrados ordinarios de Y en las X's. Típicamente, las funciones de regresión parcial se evalúan en los valores observados:

$$\hat{f}_{ij}^{(0)} \equiv \hat{f}_j^{(0)}(Xij) = B_j X_{ij}$$

Definimos $\hat{\alpha} = \bar{Y}$ y centramos los estimadores iniciales $\hat{f}_{(ij)}^{(0)}$ sustrayendo la media $\bar{\hat{f}}_j^{(0)}$. Este proceso de centrado se repite en cada iteración.

Ajuste de un modelo de regresión aditivo: *backfitting* II

2. En la iteración l, se recorren los índices de los predictores $j = 1, \dots, p$, calculando residuales parciales usando los valores más recientes de las otras funciones de regresión parcial, y suavizando los residuales parciales para actualizar la función de regresión actual. Esto es:

$$\begin{aligned}
\widehat{f}_{i1}^{(l)} &= loess(Y_{(1)i} \sim X_{i1}) = \mathbf{S}_1(Y_i - [f_2^{(l-1)}(X_{i2}) + \cdots + f_p^{(l-1)}(X_{ip})]) \\
\widehat{f}_{i2}^{(l)} &= loess(Y_{(2)i} \sim X_{i2}) = \mathbf{S}_2(Y_i - [f_1^{(l)}(X_{i1}) + f_3^{(l-1)}(X_{i3}) \cdots + f_p^{(l-1)}(X_{ip})]) \\
&\vdots \\
\widehat{f}_{ip}^{(l)} &= loess(Y_{(p)i} \sim X_{ip}) = \mathbf{S}_p(Y_i - [f_1^{(l)}(X_{i1}) + \cdots + f_{p-1}^{(l-1)}(X_{i,p-1})])
\end{aligned}$$

   donde $\mathbf{S}_j$ son las matrices de suavizamiento que vimos para ajustar un `loess`, que dependen sólo de los valores X_{ij} para el predictor j.

3. Repetir el paso 2 hasta que las funciones de regresión parcial $\widehat{f}_{ij}^{(l)}$ converjan.

- Cuando se aplica el algoritmo de *backfitting* utilizando mínimos cuadrados en cada paso, se obtienen los estimadores usuales de minimos cuadrados ordinarios múltiples.

Ajuste de un modelo de regresión aditivo: *backfitting* III

- El algoritmo de *backfitting* resuelve implícitamente el conjunto de ecuaciones de estimación:

$$\underset{(np+1)\times(np+1)}{\mathbf{S}}\ \underset{(np+1)\times 1}{\hat{\mathbf{f}}} = \underset{(np+1)\times n}{\mathbf{Q}}\ \underset{n\times 1}{\mathbf{Y}}$$

donde $\mathbf{S} = \begin{bmatrix} 1 & \mathbf{0}' & \mathbf{0}' & \cdots & \mathbf{0}' \\ \mathbf{0} & \mathbf{I}_n & \mathbf{S}_1 & \cdots & \mathbf{S}_1 \\ \mathbf{0} & \mathbf{S}_2 & \mathbf{I}_n & \cdots & \mathbf{S}_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{S}_p & \mathbf{S}_p & \cdots & \mathbf{I}_n \end{bmatrix}$, $\hat{\mathbf{f}} = \begin{pmatrix} \alpha \\ \hat{\mathbf{f}}_1 \\ \hat{\mathbf{f}}_2 \\ \vdots \\ \hat{\mathbf{f}}_p \end{pmatrix}$, $\mathbf{Q} = \begin{pmatrix} \frac{1}{n}\mathbf{1}' \\ \mathbf{S}_1 \\ \mathbf{S}_2 \\ \vdots \\ \mathbf{S}_p \end{pmatrix}$ y $\mathbf{Y} = \begin{pmatrix} Y_1 \\ Y_2 \\ Y_3 \\ \vdots Y_n \end{pmatrix}$

El vector $\hat{\mathbf{f}}$ tiene subvectores $\hat{\mathbf{f}}_i = [\hat{f}_{(1i)}, \hat{f}_{(2i)}, \ldots, \hat{f}_{(ni)}]'$ para $i = 1, \ldots, p$.

Ajuste de un modelo de regresión aditivo: *backfitting* IV

- La primera ecuación define $\alpha = \bar{Y}$. Las ecuaciones matriciales subsecuentes son de la forma:

$$\hat{\mathbf{f}}_i + \mathbf{S}_i \sum_{k \neq i} \hat{\mathbf{f}}_k = \mathbf{S}_i \mathbf{Y} \quad i = 1, \ldots, p$$

  y reacomodando términos, se obtiene la función de regresión parcial ajustada, como se requiere:

$$\hat{\mathbf{f}}_i = \mathbf{S}_i \left(\mathbf{Y} - \sum_{k \neq i} \hat{\mathbf{f}}_k \right)$$

- En teoría, se podría resolver el sistema de $np + 1$ ecuaciones en $np + 1$ incógnitas, pero la dimensión lo hace impráctico de resolver directamente.

Resolución de modelos aditivos en R

- En R hay principalmente cuatro paquetes que pueden ajustar modelos GAM que tienen o estiman diferentes características: `mgcv`, `gam` `pgam` y `gss`.
- El paquete `gam` es de los más antiguos, de hecho fue realizado por Trevor Hastie y Rob Tibshirani. Su documentación está en parte en el libro: *Statistical Models in S*. Aunque su estructura es antigua, es muy simple y poderosa su estructura.
- El paquete `mgcv` (Generalized Additive (mixed) models) tiene una función `gam` que sirve para ajustar modelos aditivos. Es muy similar al paquete `gam`, que tiene una función de su mismo nombre, permite ajustar modelos generalizados aditivos (por ejemplo, considerando variables de respuesta binarias, binomiales), dentro de los que se encuentran los modelos aditivos lineales. Algunas de las diferencias son:
  i. Como funciones f_i se pueden usar como funciones de suavizamiento `loess` (`lo`) o `splines` cúbicos (`s`). Como veremos splines más adelante, postergamos el uso de `s`.
  ii. `gam::gam` no estima el grado del suavizamiento de manera automática.

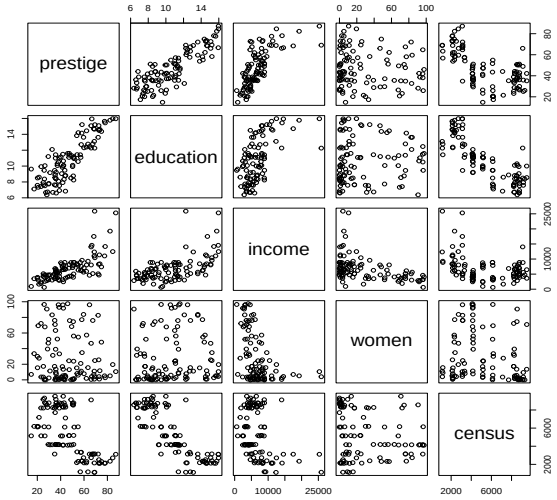
  No se recomienda cargar `mgcv` y `gam` de manera simultánea porque chocarán mucho.
- Otro paquete que tiene funciones relacionadas es el paquete `pgam` que considera respuestas Poisson o Gamma.
- El paquete más avanzado es `gss` que tiene una implementación del enfoque basado en splines para modelar GAM, que está descrito en las monografias de Grace Wahba (1990)[1] y Gu (2013)[2].

[1] Spline Models for observational data. SIAM 1990

[2] Gu, C. Smoothing Spline ANOVA Models (2nd Ed) Springer.

Resolución de modelos aditivos en R

```
plot(Prestige[,c(4,1,2,3,5)])
```



Resolución de modelos aditivos en R

```
library(gam)
gam1 <- gam(prestige ~ lo(income, span=0.3) + lo(women, span=1/10) + lo(education) , data = Prestige)
summary(gam1)
```

```
Call: gam(formula = prestige ~ lo(income, span = 0.3) + lo(women, span = 1/10) +
    lo(education), data = Prestige)
Deviance Residuals:
     Min       1Q   Median       3Q      Max
-11.4148  -3.3730  -0.7938   3.2264  15.9590

(Dispersion Parameter for gaussian family taken to be 43.8199)

    Null Deviance: 29895.43 on 101 degrees of freedom
Residual Deviance: 2923.434 on 66.7148 degrees of freedom
AIC: 704.2991

Number of Local Scoring Iterations: NA

Anova for Parametric Effects
                          Df  Sum Sq Mean Sq F value    Pr(>F)
lo(income, span = 0.3)  1.000 13789.1 13789.1 314.676 < 2.2e-16 ***
lo(women, span = 1/10)  1.000  3147.4  3147.4  71.827 3.459e-12 ***
lo(education)           1.000  5982.9  5982.9 136.533 < 2.2e-16 ***
Residuals              66.715  2923.4    43.8
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Anova for Nonparametric Effects
                       Npar Df Npar F     Pr(F)
(Intercept)
lo(income, span = 0.3)     6.9 6.1682 1.518e-05 ***
lo(women, span = 1/10)    22.0 1.4453  0.126705
lo(education)              2.4 5.3316  0.004619 **
---
```

Resolución de modelos aditivos en R I

- La última parte del resumen da una descomposición de los grados de libertad entre los términos y separa la contribución paramétrica y no paramétrica en los términos. Los resultados se presentan en una tabla ANOVA.
- La columna `Npar F` es un tipo de prueba para evaluar la contribución no lineal de los términos no paramétricos. La que no parece significativa es la de `women` con el `loess` ajustado.
- Para comparar ajustes de modelos anidados, por ejemplo

```
gam2 <- gam(prestige ~ lo(income, span=0.3) + lo(education) , data = Prestige)
anova(gam1, gam1, test = "Chi") # Chi porque estamos probando devianza

Analysis of Deviance Table

Model 1: prestige ~ lo(income, span = 0.3) + lo(women, span = 1/10) +
    lo(education)
Model 2: prestige ~ lo(income, span = 0.3) + lo(women, span = 1/10) +
    lo(education)
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1    66.715     2923.4
2    66.715     2923.4  0        0
```

- Por último, dentro de los componentes del modelo tenemos
  - `smooth`: la matriz de suavizamiento **S**
  - `var`: matriz de varianzas puntuales

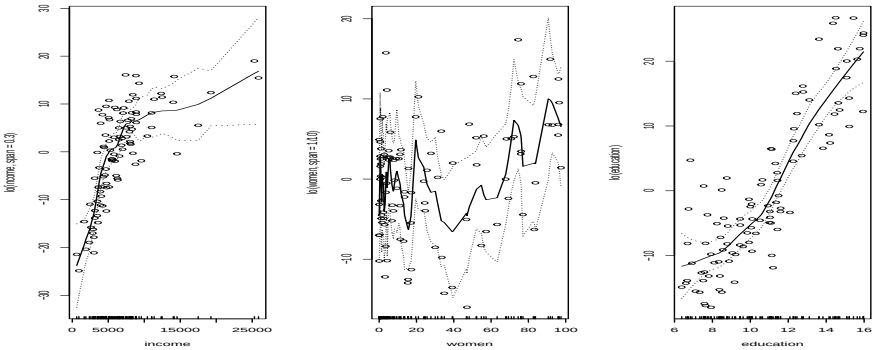
Resolución de modelos aditivos en R II

- `nl.df`: es un vector con los grados efectivos de libertad para las partes no lineales de cada suavizador
- `nl.chisq`: es un vector de estadísticas χ^2 que aproximan el efecto de reemplazar cada curva no paramétrica por su componente paramétrico.

Resolución de modelos aditivos en R I

Podemos graficar el modelo que consiste en las gráficas $\{(f_i(x_i), Y)\}$

```
par(mfrow = c(1,3))
plot(gam1, se = T, residuals=T)  # Calcula las bandas de confianza
```



Los valores ajustados por el modelo los podemos obtener de:

Resolución de modelos aditivos en R II

```
head(fitted(gam1))

 GOV.ADMINISTRATORS    GENERAL.MANAGERS         ACCOUNTANTS PURCHASING.OFFICERS            CHEMISTS
           65.13721            70.49401            55.78753            54.95981            68.08156
         PHYSICISTS
           77.57499
```

Con `predict` podemos obtener tanto el valor de la función de respuesta, como los valores $\widehat{f}_i(x_{ij})$ y el valor de la constante en el punto que estamos evaluando para predicción:

```
predict(gam1,newdata=data.frame(income=1000,education=10,women=20),type = "terms")

  lo(income, span = 0.3) lo(women, span = 1/10) lo(education)
1              -22.18265               5.273322     -5.254956
attr(,"constant")
[1] 46.83333

predict(gam1,newdata=data.frame(income=1000,education=10,women=20),type = "response")

       1
24.66905
```

Pero no se tiene un valor para el error de predicción estimado.

Resolución de modelos aditivos en R III

```
deviance(gam1)

[1] 2923.434

plot(residuals(gam1))
abline(h=0, lwd=2)
```