Smoothing Splines

Vito Simone Lacatena 747810

3/7/2021

Introduzione

Prima di entrare nel cuore dell'algoritmo occorre introdurre dei concetti fondamentali, partendo dal definire il problema della regressione.

Il problema della regressione

In generale un problema di regressione ha l'obiettivo di stimare un modello che descriva uan relazione tra una **variabile di risposta** Y dipendente è un insieme di **variabili predittive** indipententi $X_1, ..., X_p$.

Il più semplice modello di regressione è il modello di **regressione lineare** in cui si assume una relazione lineare tra una singola variabile predittiva X e la variabile di risposta, tale funzione lineare viene approssimata come segue: Y

$$Y = \beta_0 + \beta_1 X + \epsilon$$

I parametri β_0 e β_1 sono sconosciuti e vanno stimati utilizzando i dati. In altri termini occorre determinare le stime $\hat{\beta}_0$ e $\hat{\beta}_1$, dove $\hat{\beta}_0$ è l'intercetta e $\hat{\beta}_1$ è la pendenza della retta che dovrebbe passare il più vicino possibile alle osservazioni. Per determinare il valore ottimale di questi parametri è possibile utilizzare diversi metodi che misurano la vicinanza delle stime ai valori delle osservazioni, il metodo più comune è il criterio dei **minimi quadtrati**, ovvero scegliere le stime dei parametri $\hat{\beta}_0$ e $\hat{\beta}_1$ in modo da minimizzare la **somma dei residui al quadrato**, definita come :

$$RSS = \sum_{i=1}^{N}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{N}(y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i))^2$$

La regressione lineare standard ha limiti in termini di capacità predittiva, dato che necessità dell'assunzione di linearità dei dati, che di solito risulta una assunzione molto debole. Per questo occorre esaminare delle estensioni del semplice modello lineare, ovvero la regressione polinomiale, le funzioni a gradino e le spline.

Regressione Polinomiale

Il modello di regressione polinomiale estende il modello di regressione lineare attraverso l'aggiunta di ulteriori variabili ottenuti dalla variabile originale elevandola ad una potenza. Ovvero si estendo il modello lineare classico

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

con la seguente funzione polinomiale:

$$y_i = \beta_0 + \beta_1 x_1 + \beta_2 x_i^2 + ... + \beta_d x_i^d + \epsilon_i$$

In questo modo è possibile ottenere curve non lineari, non è consigliabile utilizzare valori trppo grandi di d poichè la curva diventerebbe troppo flessibile,è preferibile utilizzare polinomi di grado 3, o al massimo 4.

Funzioni a gradino

L'uso di funzioni polinomiale definisce una struttura globale della funzioe non lineare della variabile predittrice X, per evitare ciò è possibile utilizzare delle **funzioni a gradino**. Tali funzioni dividono l'intervallo dei valori in K bins distinti creando un insieme di valori di soglia (detti anche **cutpoints**) $c_1, c_2, ..., c_k$ nell'intervallo e per ogni bin si va ad adattare una funzione costante, Questo equivale a trasformare una variabile continua in una variabile categorica ordinale, avremo quindi:

$$y_i = \beta_0 + \beta_1 C_1(x_i) + \beta_2 C_2(x_i) + ... + \beta_k C_k(x_i) + \epsilon_i$$

dove C_j con $j = 1...k$ è una funzione che assume valore 1 se $c_{j-1} < x_i < c_j$ altrimenti 0.

Funzioni base

I modelli di regressione polinomiali e funzoni a scalini sono casi particolari di un framework generico di funzioni base, in cui si va ad adattare il seguente modello:

$$y_i = \beta_0 + \beta_1 b_1(x_i) + \beta_2 b_2(x_i) + ... + \beta_k b_k(x_i) + \epsilon_i$$

dove $b_j(.)$ sono funzioni base, quindi possiamo rivedere i casi precedente come istanze di questo modello,ovvero:

- regressione polinomiale : $b_j(x_i) = x_i^j$;
- funzioni a scalini : $b_j(x_i) = C_j(x_i)$;

Splines

Invece di adattare un polinomio di grado elevato su l'intero intervallo , l'idea base dell'approccio Spline è quello di utilizzare la **regressione polinomiale a tratti** che comporta l'inserimento di polinomi di grado inferiore su diversi sottointervalli. Esempio se si considerano polinomi di terzo grado e si divide l'intervallo originale X sul punto c, avremmo due cubiche con differenti coefficenti:

$$y_i = \begin{cases} \beta_{01} + \beta_{11} x_i + \beta_{21} x_i^2 + \beta_{31} x_i^3 + \varepsilon_i & \text{se } x_i < c \\ \beta_{02} + \beta_{12} x_i + \beta_{22} x_i^2 + \beta_{32} x_i^3 + \varepsilon_i & \text{se } x_i \geq c \end{cases}$$

In questo caso si adatterano due diversi polinomi con i propri coefficenti, il punto c dell'intervallo in cui si passa da un polinomio ad un altro è chiamatato **nodo** (o **knot**).

Osservazione: con K nodi si avranno $K + 1$ polinomi.

I **gradi di libertà** per questo esempio sono 8, per gradi di libertà si intende il numero di parametri indipendenti da stimare che vengono utilizzati dal modello, nell'esempio precedente il numero di gradi di libertà è quattro per ogni polinomio (poichè di grado 3), quindi in tutto otto.

Per avere una curva che non sia troppo flessibile e non appaia discontinua nel passaggio da un intervallo all'altro è importante aggiungere tre vincoli: la **funzione deve essere continua** ,e la **derivata prima e seconda dei polinomi a tratti devono essere continue**, in questo modo non solo il polinomio a tratti sarà continuo ma sarà anche smussato. Ad ogni vincolo il numero di gradi di libertà si riduce di 1 , in questo modo i gradi libertà del polinomio di grado 3 visto prima si riducono a 5.

Quindi in generale, una **spline di ordine** d con nodi c_j, $j = 1, ..., K$ è un polinomio a tratti di ordine d che ha derivate continue sino all' ordine $d - 1$, una **spline cubica** è quindi due volte differenziabile nell'intero intervallo. Le **spline cubiche** sono ritenute popolari poichè difficile individuare la discontinuità ai nodi, una

spline cubica con K nodi utilizza $K+4$ gradi di libertà. Le Spline di grado superiore ad secondo mostrano un elevata variabilità agli estremi dell'intervallo, una **spline naturale** è una spline con vincoli di linearità agli estremi.

La scelta del numero e della posizione dei nodi

Un modo per poter definire un posizionamento uniforme dei nodi è quello di speficificare i gradi di libertà desiderati, provando diversi valori, Un metodo più obiettivo consiste nell'utilizzare la cross-validation in cui si effetuano più iterazioni utilizzando una parte dei dati per adattare la spline con un determinato numero K di nodi e i dati non visti rimanenti sono utilizzati per fare la previsioni,si calcola una somma dei quadtrati dei residui complessiva per avere una misura di valutazione.La procedura può essere effettuata più volte per divers valori di K e scegliere il modello che ha fornito un risultato RSS più piccolo.

Basi della Spline

Una regressione spline può essere rappresentata in termini di basi di funzioni.

Caso semplice: regressione spline lineare con grado $d=1$ e $K=1$ nodi

$$y_i = \beta_0 + \beta_1 b_1(x_i) + \beta_2 b_2(x_i) + \epsilon_i$$

dove $b_1(x) = x$ e successivamente si usano funzioni di base troncate

$$h(x,c) = (x-c)_+ = \begin{cases} (x-c) & \text{se } x > c \\ 0 & \text{se } x \leq c \end{cases}$$

Se $x_i \leq c$ allora $(x_i - c)_+ = 0$ e $y_i = \beta_0 + \beta_1 x_i + \epsilon_i$

Se $x_i > c$ allora $(x_i - c)_+ = (x_i - c)$ e $y_i = \beta_0 + \beta_1 x_i + \beta_1(x_i - c) + \epsilon_i$

In generale: regressione spline di grado d e K nodi

$y_i = \beta_0 + \beta_1 b_1(x_i) + \beta_2 b_2(x_i) + \beta_K b_K(x_i) + \beta_{K+1} b_{K+1}(x_i) + ... + \beta_{K+d} b_{K+d}(x_i) + \epsilon_i$

le funzioni di base sono

$$x, x^2, ..., x^d, h(x,c_1), ..., h(x,c_K)$$

dove in questo caso

$h(x,c) = (x-c)_+^d = (x-c)^d$ se $x > c$ altrimenti 0.

Struttura dell'algoritmo

Smoothing Splines:Panoramica

L'obiettivo della regressione è adattare una funzione $f(x)$ ad un set di dati tale che minimizzi la **somma dei quadtrati del residui**

$$RSS = \sum_{i=1}^{n} (y_i - f(x_i))^2$$

In questo modo si potrebbe semplicemente ottenere un RSS pari a zero scegliendo la funzione f che interpoli perfettamente tutti i puntii y_i, ma tale risultato sarebbe troppo soggetto ad overfitting poichè la funzione sarebbe troppo flessibile. Occorre quindi trovare il modo di smussare la funzione, ridefinendo la RSS nel seguente modo:

$$RSS(f,\lambda) = \sum_{i=1}^{n}(y_i - f(x_i))^2 + \lambda \int f''(t)^2 dt$$

La funzione f è definita **smoothing spline**.

La prima parte di RSS definisce una misura di distanza dai dati mentre la seconda parte definisce una penalità che penalizza la variabilità della funzione f , la quantità $\lambda \int f''(t)^2 dt$ è quindi una misura del cambiamento globale nella funzione $f'(t)$

- se f è smussata $\int f'(t)$ sarà costante e $\int f''(t)$ avrà un valore molto piccolo;
- se f è irregolare $\int f'(t)$ sarà molto variabile e $\int f''(t)$ avrà un valore molto grande;

Il parametro λ è un **parametro di smoothing** che porterà f ad essere smussata, il parametro assume valori compresi in $(0, \infty)$:

- Con valore $\lambda = 0$ i suoi effetti sono nulli e quindi la funzione assumerà un comportamento molto irregolare;
- con $\lambda = \infty$ la funzione sarà una linea retta che passa il più vicino possibile ai dati.

Quidi grandi valori di λ producono curve smussata, mentre piccoli valori di questo parametro producono curve più irregolari.

Si consideri il vettore n-dimensionale $\hat{f}_\lambda$ contenenti i valori adattati ai dati di training $x_1, ..., x_n$, nonchè soluzione di $RSS(f,\lambda)$ per un determinato valore di λ. Questo vettore può essere scritto come:

$$\hat{f}_\lambda = S_\lambda y$$

Dove la matrice di dimensione $n \times n$ S_λ è nota come **smoother matrix**, e il vettore y e il **vettore di risposta**, il numero dei gradi di libertà corrisponde alla traccia della matrice S_λ.

$$df_\lambda = trace(S_\lambda)$$

Selezione Automatica dei parametri di Smoothing

In generale i parametri di smoothing da stimare per la regressione spline sono:

- il grado delle spline;
- il numero e la posizione dei nodi.

In pratica si può però dimostrare che la funzione $f(x)$ che minimizza la RSS è una **spline cubica naturale** avente nodi ad ogni punto $x_1, ...x_n$, questo significa che per quanto riguarda lo smoothing splines si ha un solo parametro λ da stimare poichè i nodi corrispondono ai punti di training e si utilizzano polinomi di grado 3.

Occorre notare che all'aumentare del valore di λ da 0 a ∞ il numero di gradi di libertà descresce da n a 2.

Uno dei metodi per valutare la scelta del parametro λ è la cross-validation, in particolare l'errore di cross-validation Leave One Out (LOOCV) viene calcolato con la seguente formula:

$$LOOCV(\hat{f}_\lambda) = \frac{1}{N}\sum_{i=1}^{N}(\frac{y_i - \hat{f}_\lambda(x_i)}{1 - S_\lambda ii})^2$$

In questa formula $\hat{f}_\lambda$ indica la funzione smoothing spline adattata su tutti i dati di training tranne x_i e $S_\lambda ii$ l'elemento diagonale i-esimo della matrice S_λ.

Caso Multidimensionale

Sino ad ora si è considerato il caso unidimensionale dell'aproccio smoothing splines,ma tale metodo può essere generalizzato anche per un numero maggiore di dimensioni. Data la coppia y_i, x_i , con $y_i \in \mathbb{R}$ e $x_i \in \mathbb{R}^d$, e una funzione di regressione d-dimensionale $f(x)$, si consideri il seguente problema così impostato:

$$\min_f \sum_{i=1}^{N} \{y_i - f(x_i)\}^2 + \lambda J[f]$$

dove J è una funzione di penalità appropriata per stabilizzare una funzione f in $\mathbb{R}^d$. Esempio in $\mathbb{R}^2$:

$$J[f] = \int\int_{\mathbb{R}^2} \left[\left(\frac{\partial^2 f(x)}{\partial x_1^2} \right)^2 + 2 \left(\frac{\partial^2 f(x)}{\partial x_1 \partial x_2} \right)^2 + \left(\frac{\partial^2 f(x)}{\partial x_2^2} \right)^2 \right] dx_1 dx_2.$$

Utilizzando per l'otimizzazione questa penalità si ottiene una superficie bidimensionale smussata, nota come spline a superficie sottile. Condivide molte proprietà con la smoothing spline unidimensionale.

- Per $\lambda \to 0$ si ottiene come soluzione una funzione di interpolazione;
- Per $\lambda \to \infty$ la soluzione converge al piano dei minimi quadtrati;

Implementazioni in R

Il linguaggio di programmazione R trova applicazione in ambiti scientifici e statistici. Sono disponibili due implementazioni della smoothing spline in R, in due paakgage distinti: la funzione **smooth.spline** nel package **stats** e la funzione **ss** nel package **npreg**.

Funzione smooth.spline

Gli argomenti in input alla funzione smooth.spline sono i seguenti:

Argomento	Descrizione
x	Vettore di valori della variabile predittiva
y	Vettore di valori della variabile di risposta
w	Vettore di pesi della stessa dimensione di x (opzionale), di default è un vettore di 1
df	Argomento che permette di settare numero dei gradi di libertà desiderati. Più alto è il numero, più ondulata è la curva adattata e più da vicino segue i dati. Dovrebbe essere compreso tra 1 e il numero di punti distinti in x
spar	Parametro di smoothing,se specificato il parametro λ dell'integrale della derivata seconda quadrata nel criterio di fitting (log likelihood penalizzato) è una funzione monotona di spar
lambda	Al posto di scar può essere specificato questo parametro di smoothing
cv	Argomento booleana, se impostato a TRUE viene utilizzata la cross-validation di Tipo Leave-One-Out (LOOCV) altriemnti se FALSE la Generalized Cross-Validation (GCV). Viene usato per il calcolo dei parametri di smoothing solo quando sia spar che df non sono specificati è comunque usatoper determinare cv.crit nel risultato. Impstando questo argoemnto su NA per la velocizzazione si salta la valutazione.
all.knots	Se TRUE utilizza tutii i punti distinti come nodi, se FALSE usa un sottoinsieme di questi punti
nknots	Può essere settato come il numero o un criterio che resituisce il numero di nodi (funziona solo se all.nots = FALSE)
keep.data	Argomento boolenao che specifica se i dati di input devono essere mantenuti nel risultato. Se TRUE (come per default), i valori montati e i residui sono disponibili dal risultato
df.offset	Permette di aumentare i gradi di libertà di df.offset nel criterio GCV.

Argomento	Descrizione
penality	Coefficiente della penalità per i gradi di libertà nel criterio GCV.
control.spar	Lista opzionale con i componenti nominati che controllano la ricerca della radice quando il parametro di smoothing spar è calcolato.
tol	Soglia di tolleranza per l'uguaglianza o l'unicità dei valori x. I valori sono suddivisi in intervalli di dimensione tol e i valori che cadono nello stesso intervallo sono considerati uguali. Deve essere strettamente positivo (e finito)
keep.stuff	Argomento booleano sperimentale che indica se il risultato deve tenere extra dai calcoli interni. Dovrebbe permettere di ricostruire la matrice X e altro.

Funzione ss

La funzione *ss* è inspirata alla funzione *smooth.spline* del package stats,in aggiunta a *smooth.splin*:

- Invece dell'argomento cv disponde di *method* permettendo di scegliere tra otti diversi metodi (GCV, OCV, GACV, ACV, REML, ML, AIC) per selezionare il parametro di smoothing;
- Permette di definire tre tipi di spline(linear, cubic,quintic);
- permette di definire un vincolo di periodicità
- permette di specificare i valori dei nodi

GLi argomenti x,y,w,df,spar,lambda,all.knots,nknots,keep.data,df.offset,penality,control.spar sono gli stessi ritrovati in smooth.spline. Di seguito verranno descritti gli argomenti presenti in *ss* e non presenti in *smooth.spline*.

Argomento	Descrizione
method	Metodo per selezionare il parametro di smoothing. Ignorato se viene fornito spar o lambda.
m	Il valore predefinito è m = 2, che è una spline di smoothing cubica. Imposta m = 1 per uno spline lineare o m = 3 per uno spline quintico
periodic	Se TRUE, la funzione stimata f(x) è costretta ad essere periodica
knots	Vettore dei valori dei nodi per la spline. I valori dovrebbero essere singoli e all'interno dell'intervallo dei valori x (per evitare un warning).
bernoulli	Se TRUE, vengono utilizzati polinomi di Bernoulli scalati per le funzioni di base e di penalizzazione. Se FALSE, produce la definizione "classica" di una spline di smoothing.

Esempio:Heart Failure Clinical Records

Vediamo ora un esempio pratico utilizzando come riferimento il dataset **Heart Failure Clinical Records**, il dataset contiene 13 feature, che riportano informazioni cliniche, corporee e sullo stile di vita di pazienti tra i 40 e 95 anni di cui 105 donne e 194 uomini . Alcune feature sono binarie: anemia, pressione alta, diabete, sesso e fumo, altre sono di tipo numerico

features :

- **age**: età del paziente (anni)
- **anaemia**: diminuzione dei globuli rossi o dell'emoglobina (booleano)
- **high blood pressure**: se il paziente ha l'ipertensione (booleano)
- **creatinine phosphokinase** (CPK): livello dell'enzima CPK nel sangue (mcg/L)
- **diabetes**: se il paziente ha il diabete (booleano)
- **ejection fraction**: percentuale di sangue che lascia il cuore ad ogni contrazione (percentuale)
- **platelets**: piastrine nel sangue (kiloplatelets/mL)
- **sex**: donna o uomo (binario)
- **serum creatinine**: livello di creatinina sierica nel sangue (mg/dl)
- **serum sodium**: livello di sodio sierico nel sangue (mEq/L)

- **smoking**: se il paziente fuma o no (booleano)
- **time**: periodo di follow-up (giorni)
- **death event**: se il paziente è deceduto durante il periodo di follow-up (booleano)

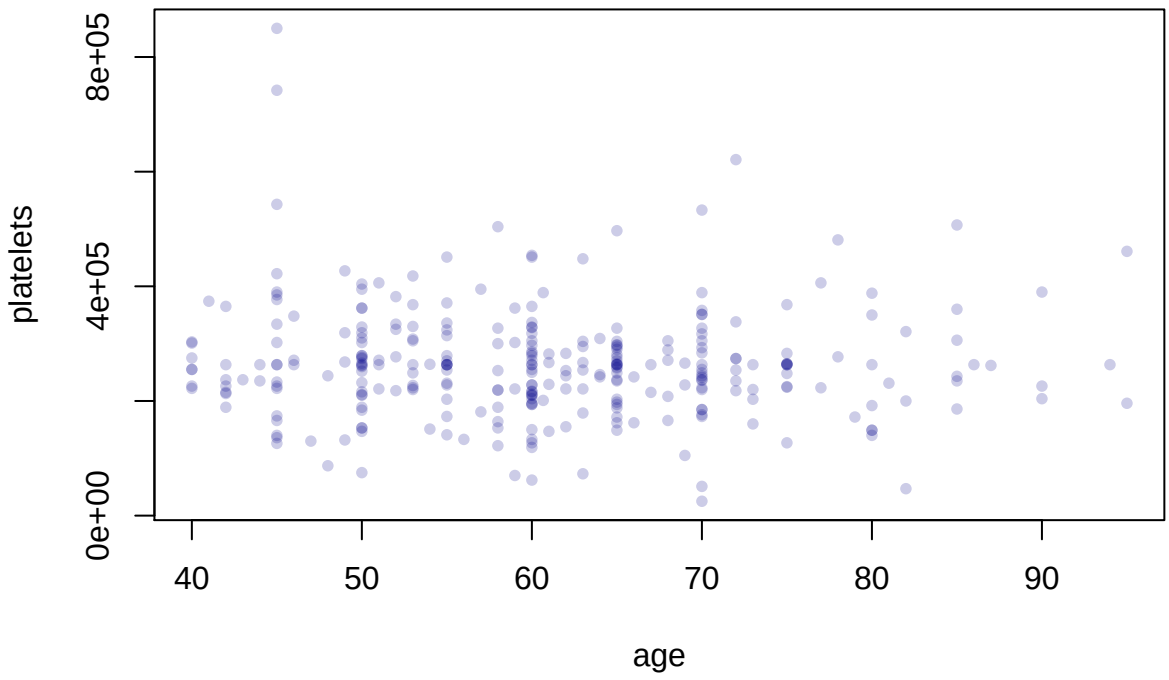
```
library(DT)
data <- read.csv(file = './data/hfcr.csv',sep=',')


head(data,5)
```

```
##   age anaemia creatinine_phosphokinase diabetes ejection_fraction
## 1  75       0                     582        0                20
## 2  55       0                    7861        0                38
## 3  65       0                     146        0                20
## 4  50       1                     111        0                20
## 5  65       1                     160        1                20
##   high_blood_pressure platelets serum_creatinine serum_sodium sex smoking time
## 1                   1    265000              1.9          130   1       0    4
## 2                   0    263358              1.1          136   1       0    6
## 3                   0    162000              1.3          129   1       1    7
## 4                   0    210000              1.9          137   1       0    7
## 5                   0    327000              2.7          116   0       0    8
##   DEATH_EVENT
## 1           1
## 2           1
## 3           1
## 4           1
## 5           1
```

Considero come variabile predittiva **age** e come variabile di risposta **platelets**

```
library(scales)
xlabel = 'age'
ylabel = 'platelets'
x <- data$age
y <- data$platelets
plot(x,y,xlab = xlabel,ylab = ylabel,type = "p",col=alpha('darkblue', 0.2),pch =16,cex=0.8)
```



Smoothing Spline con smooth.spline

Si osservi che settando il parametro **df** dei gradi di libertà si ottengono i casi estremi di df = n (numero di osservazioni uniche) e df = 2, si ricorda che al crescere del valore di λ i gradi di libertà diminuiscono.

```
n = length(unique(x))

spline_A <- smooth.spline(x,y,df = 2)
spline_B <- smooth.spline(x,y,df = n)

print(spline_A)
```

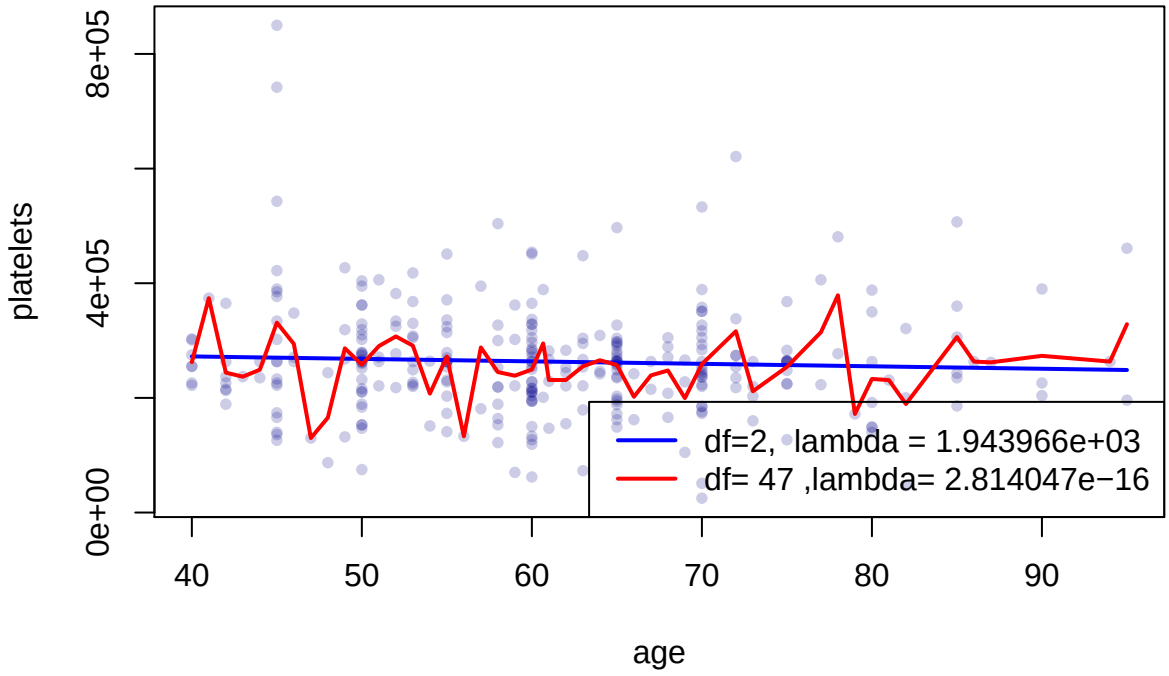
```
## Call:
## smooth.spline(x = x, y = y, df = 2)
##
## Smoothing Parameter  spar= 1.499937  lambda= 1943.966 (34 iterations)
## Equivalent Degrees of Freedom (Df): 2.000253
## Penalized Criterion (RSS): 347330735935
## GCV: 9635993712
```

```
print(spline_B)
```

```
## Call:
## smooth.spline(x = x, y = y, df = n)
##
## Smoothing Parameter  spar= -1.107721  lambda= 2.814047e-16 (30 iterations)
## Equivalent Degrees of Freedom (Df): 47
```

```
## Penalized Criterion (RSS): 1.130914e-07
## GCV: 11749307576
```

```
plot(x,y,xlab = xlabel,ylab = ylabel,type = "p",col=alpha('darkblue', 0.2),pch =16,cex=0.8)
lines(spline_A,lwd=2,col="blue")
lines(spline_B,lwd=2,col="red")
legend("bottomright", c(paste("df=2, ",'lambda',"=",format(spline_A$lambda,scientific = TRUE)),paste("d
```



Dal grafico si può notare infatti che come previsto che con un numero di gradi di libertà di 2 il parametro labmda assume un elevato valore smussando la curva in modo che la smoothing spline è una retta (curva blu), con un numero di gradi di libertà pari al numero di osservazioni il parametro labda tendo al valore 0 e la curva risulta molto irregolare (curva rossa).

Vediamo di adattare la smoothing spline stimando i parametri di λ e di ds in modo da ottenere un risultato ottimale. Richiamando la funzione smooth.spline non specificando i parametri di smoothing la funzione stimerà tali parametro automaticamente mediante una Leave-One Out o una Cross Validation Generalizzata, in questo caso con cv = TRUE verrà utilizza una cross-validation Leave-One-Out

```
spline <- smooth.spline(x, y,cv = TRUE)
spline
```

```
## Call:
## smooth.spline(x = x, y = y, cv = TRUE)
##
## Smoothing Parameter  spar= 0.964789  lambda= 0.2643345 (12 iterations)
## Equivalent Degrees of Freedom (Df): 2.903834
## Penalized Criterion (RSS): 318629731212
## PRESS(l.o.o. CV): 9619340327
```

```
plot(x,y,xlab = xlabel,ylab = ylabel,type = "p",col=alpha('darkblue', 0.2),pch =16,cex=0.8)

lines(spline,lwd=2,col="darkblue")
legend("bottomright",paste("df=",spline$df ,'lambda',"=",format(spline$lambda,scientific = TRUE)),col="
```