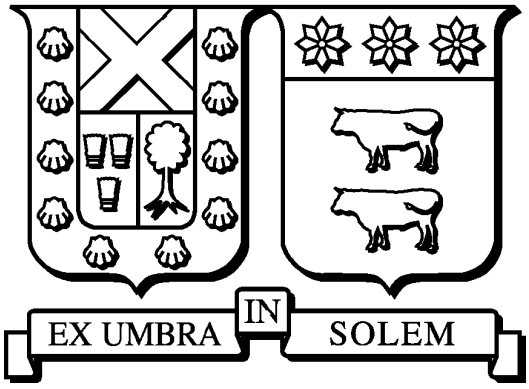
UNIVERSIDAD TÉCNICA FEDERICO
SANTA MARÍA

Departamento de Matemática
Valparaíso-Chile



Estimación Robusta del Parámetro de Suavizamiento en P-splines

Memoria presentada por:
Carlos Alejandro Schwarzenberg Millar

Como requisito parcial

para optar al título profesional Ingeniero Civil Matemático

Profesores Guías:

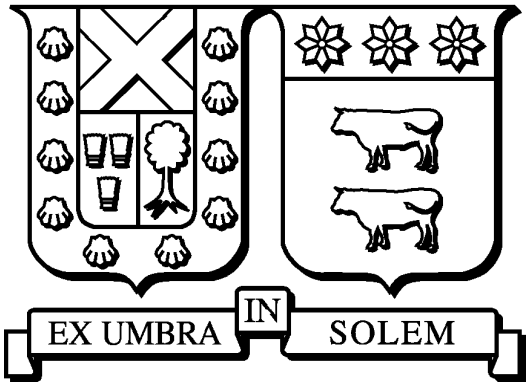
Felipe Osorio Salgado

Ronny Vallejos Arriagada

Noviembre, 2016

Universidad Técnica Federico Santa María

Departamento de Matemática
Valparaíso-Chile



Estimación Robusta del Parámetro de Suavizamiento en P-splines

Memoria presentada por:
Carlos Alejandro Schwarzenberg Millar

Como requisito parcial
para optar al título profesional Ingeniero Civil Matemático

Profesores Guías:
Felipe Osorio Salgado
Ronny Vallejos Arriagada

Examinadores:
Alberto Mercado Saucedo

Noviembre, 2016

TÍTULO DE LA MEMORIA:
Estimación Robusta del Parámetro de Suavizamiento en P-splines.

AUTOR: Carlos Alejandro Schwarzenberg Millar.

TRABAJO DE MEMORIA, presentado como requisito parcial para optar al título profesional Ingeniero Civil Matemático de la Universidad Técnica Federico Santa María.

COMISIÓN EVALUADORA:

Integrantes	Firma
Felipe Osorio Salgado Pontificia Universidad Católica de Valparaíso, Chile.	_____
Ronny Vallejos Arriagada Universidad Técnica Federico Santa María, Chile.	_____
Alberto Mercado Saucedo Universidad Técnica Federico Santa María, Chile.	_____

Valparaíso, Noviembre 2016.

Agradecimientos

En primer lugar quiero agradecer a mi familia por el cariño y apoyo incondicional durante todos estos años. En especial a mi madre Olga Millar y a mi padre Juan Carlos Schwarzenberg (Q.E.P.D). También a mi tía Gloria (Lola) Millar y a mi abuelita Olga Menares.

Durante estos casi 8 años que he estado en la región de Valparaíso he tenido la oportunidad de conocer un montón de gente, entre compañeros de carrera, compañeros de universidad, y profesores, me causa nostalgia pensar en el el año 2008 cuando ingresé a la universidad y empezó esta etapa que ahora está por concluir. Muchas gracias a los los chicos de mate, con quienes compartí por muchos años, con los cuales pasamos por muchas horas de estudio en la sala F-265 y con quienes compartí varios cafes en la sala del café del departamento de matemáticas. Gracias a los profesores por su dedicación, en especial a mi profesor guía Felipe Osorio.

Un agradecimiento especial al Coro de Cámara de la Universidad Técnica Federico Santa María y a su director Felipe Molina Lavandera y pos su puesto a sus integrantes, esta agrupación me permitió desarrollarme como músico y cantante a la par que me desarrollaba como ingeniero, además aquí conocí a varios amigos.

También quiero agradecer a la tía Leontina y a su familia, ella es la dueña de la pensión en donde estuve viviendo por casi 6 años y quien cuidó de mi y mis compañeros de pensión con mucho cariño y dedicación.

Valparaíso, Noviembre 2016.

A mi familia y amigos.

Índice general

Índice de figuras

Índice de cuadros

Estimación Robusta del Parámetro de Suavizamiento en P-Splines.

por Carlos Schwarzenberg Millar

Resumen

Para seleccionar el parámetro de suavizamiento en la regresión P-spline se han propuesto varios métodos en la literatura, algunos de los más populares son validación cruzada (CV) y el criterio de información de Akaike (AIC). En este trabajo se ha desarrollado una herramienta para seleccionar el parámetro de suavizamiento en la regresión P-spline [Eilers y Marx, 1996] a la par que se realiza el ajuste de la curva a los datos, este método utiliza un modelo de mezcla de escala de normales [Andrews y Mallows, 1974] por lo que es robusto ante la presencia de observaciones escapadas (outliers). Los cálculos se llevan a cabo mediante el algoritmo EM anidado [van Dyk, 2000], el cual es una variante computacionalmente eficiente del algoritmo EM usual. Luego se realizan estimaciones del error estándar de los parámetros mediante el trabajo de Oakes [1999], en el cual se presenta una ecuación para calcular la matriz Hessiana en el algoritmo EM. Finalmente se han hecho pruebas de rendimiento al algoritmo para medir su desempeño en términos de tiempo de ejecución y calidad de las respuestas.

Capítulo 1

Introducción

En muchos trabajos de ingeniería es usual encontrarse con procesos a los cuales se les busca una ecuación (modelo) que los pueda representar con el fin de entender el comportamiento de los datos o de realizar predicciones. En algunas ocasiones se suele llevar a cabo esta tarea mediante una interpolación por splines, es decir, un ajuste por funciones base ponderadas que permiten trazar la curva deseada. En estos casos el proceso de ajuste puede verse afectado por la presencia de datos atípicos o outliers, los cuales pueden alterar el resultado del ajuste haciendo que la curva trazada no represente de buena manera el comportamiento del proceso en estudio. Ante este tipo de situaciones, se requiere de métodos robustos que permitan realizar la estimación de los parámetros y que estos no se vean alterados por estas observaciones atípicas.

En este trabajo se propone un método de estimación robusto para el parámetro de suavizamiento (denotado por λ) en P-splines (Splines Penalizados) [Durbán, 2009], en el cual se consideran modelos con mezcla de escala de normales [Andrews y Mallows, 1974] con el objetivo de atenuar el efecto de los datos atípicos, en donde también se recurre a estimaciones de parámetros mediante el algoritmo EM anidado [van Dyk, 2000], el cual es una variante computacionalmente eficiente del algoritmo EM [Dempster *et al.*, 1977]. La importancia del parámetro de suavizamiento λ tiene que ver con que este permite controlar la suavidad de la curva ajustada, para valores grandes de este parámetro la curva ajustada será muy suave (en casos extremos, similar a una recta) y para valores pequeños de este la curva será más rugosa (en casos extremos, similar a una interpolación). En este punto la presencia de observaciones escapadas puede afectar fuertemente la elección del valor de λ, ver por ejemplo, Osorio [2016b],

Shi y Wang [1999], Thomas [1991] ya que según el criterio de ajuste que se esté utilizando se pueden obtener valores que no sean satisfactorios al momento de realizar el ajuste de la curva.

Cuando el proceso de estimación está completo es de interés tener una cuantificación del error asociado a los parámetros estimados, para esto se consideran los trabajos de Oakes [1999] y de Lee y Pawitan [2014] quienes han abordado el tema del cálculo de la matriz de información cuando usamos el algoritmo EM y también del cálculo de la matriz de covarianza de los estimadores máximo verosímiles penalizados respectivamente, en base a estos trabajos es posible aprovechar la estructura del algoritmo EM anidado para tener una expresión de la matriz de información observada.

Finalmente, en este trabajo, se llevan a cabo experimentos numéricos para evaluar el comportamiento y desempeño del algoritmo propuesto en términos de tiempo de ejecución, error estándar en el caso de simulaciones y la calidad de los resultados entregados. En esa sección se analiza la base de datos de esperanza de vida que se ha estudiado, por ejemplo, en el trabajo de Leinhardt y Wasserman [1979]. Estos datos representan el PIB per cápita versus la esperanza de vida de ciertos países. Se escogen estos datos por la presencia de observaciones atípicas que dificultan los ajustes tradicionales [Thomas, 1991]. Luego se presenta un estudio de simulación donde se estudian bases de datos generadas a partir de algunas funciones de prueba y cuyos resultados permiten apreciar la bondad del procedimiento propuesto.

1.1. Organización

Este trabajo se ha organizado de la siguiente forma:

Capítulo 2, en este capítulo se describen algunas de las herramientas necesarias para el desarrollo del algoritmo propuesto y su posterior formulación, estas herramientas corresponden al suavizamiento mediante P-spline, algoritmo EM y algoritmo EM anidado.

Capítulo 3, este es el capítulo donde se presentan los resultados principales de este trabajo, en el que se desarrolla la formulación del algoritmo propuesto utilizando las herramientas introducidas en el capítulo de preliminares. Junto con la estimación del parámetro de suavizamiento también se desarrolla un método para estimar el error estándar de la estimación.

Capítulo 4, finalmente en este capítulo se realizan distintas pruebas para evaluar el desempeño del algoritmo, estas pruebas consisten en experimentos numéricos y análisis de datos.

Además, este trabajo cuenta con los siguientes apéndices:

Apéndice A, en este apéndice se muestran en detalle los principales cálculos del Capítulo 3. Se decide dejar estos cálculos en un Apéndice con el objetivo de que la lectura del mismo sea más expedita.

Apéndice B, en este apéndice final se presentan los códigos usados para realizar los experimentos del Capítulo 4, estos cálculos se llevaron a cabo con el programa R Core Team [2016].

Capítulo 2

Preliminares

En este capítulo se detallan algunas de las herramientas que se utilizan para el planteo e implementación del algoritmo propuesto. En la Sección 2.1 se presenta el método de suavizamiento usando splines penalizados (P-spline) [Eilers y Marx, 1996], sobre cómo se plantea este y sobre la notación usada. Además de algunos criterios existentes para seleccionar el parámetro de suavizamiento. En la Sección 2.2 se introduce el algoritmo EM [Dempster *et al.*, 1977] y posteriormente se presenta el algoritmo EM anidado [van Dyk, 2000], este último de vital importancia, ya que de este se obtiene la estructura del algoritmo propuesto.

2.1. Suavizamiento usando P-splines

Cuando se tiene la necesidad de ajustar curvas o buscar modelos para estudiar alguna base de datos, esto con el fin de poder realizar predicciones o para tener un mejor entendimiento de los fenómenos en estudio, son muchas las opciones existentes con las que se puede llevar a cabo esta tarea. Las alternativas para desarrollar este tipo de análisis se dividen principalmente en modelos de regresión paramétricos y no paramétricos. La diferencia de estas dos ideas es que en los modelos paramétricos se asume una estructura preestablecida, la cual depende de la elección de parámetros, los que adquieren una interpretación de interés en el modelo. Por su parte, lo modelos no paramétricos se enfocan en estimar directamente la función deseada, esto mediante, por ejemplo, el uso de funciones bases ponderadas. Teniendo en cuenta lo anterior, a continuación se desarrolla una descripción más detallada acerca de los

modelos de regresión no paramétricos, para luego entrar en detalle al suavizamiento usando P-splines:

Se considera que se tiene una serie de datos ordenados (x_i, y_i) para $i = 1, \ldots, n$. Se asume que estos se encuentran relacionados mediante una función $f(x)$ suave y definida en un intervalo I de la siguiente forma

$$y_i = f(x_i) + \epsilon_i, \quad i = 1, \ldots, n \tag{2.1.1}$$

para $\{\epsilon_i\}$ variables aleatorias con media cero y varianza constante.

El objetivo es estimar $f(x)$ directamente usando la información desde los datos. Para llevar a cabo este paso existen varios métodos que se dividen en dos grupos principales, los ajustes tipo Kernel y tipo splines [Durbán, 2009]. El método de interés para nosotros y sobre el cual se desarrolla este trabajo es el ajuste por splines penalizados o P-splines [Eilers y Marx, 1996], el cual nace de la idea de buscar la función que minimiza la suma de cuadrados penalizada, definida como:

$$S(f) = \sum_{i=1}^{n} \{y_i - f(x_i)\}^2 + \lambda \int_I \{f''(x)\}^2 \, dx. \tag{2.1.2}$$

A simple vista vemos que el primer término de la Ecuación (2.1.2) corresponde al ajuste de mínimos cuadrados, mientras que el segundo término es una penalización que se encarga de controlar la suavidad de la curva ajustada principalmente mediante el valor de λ el cual recibe el nombre de parámetro de suavizamiento, este término ha sido añadido con el fin de regular la fexibilidad del ajuste (dependiendo de su elección) [O'Sullivan, 1986, 1988]. Para el caso en que $\lambda = 0$ la Ecuación (2.1.2) solo consideraría la suma de cuadrados, por lo que el ajuste pasaría a ser una interpolación de los pares (x_i, y_i). Si por el contrario consideramos un $\lambda \to \infty$ el castigo a la suavidad de la curva sería tan grande que esta se convertiría en casi una recta.

Para estimar $f(x)$ se considera que ésta se puede escribir como una combinación lineal de funciones base $\{B_1, B_2, \ldots, B_p\}$, también llamada base spline, en este trabajo se ha optado por usar funciones B-spline, las cuales consisten en piezas de polinomios conectadas y que cumplen con características bastante específicas, ver por ejemplo [Eilers y Marx, 1996]. En particular, la característica que es de nuestro interés es que las derivadas son fáciles y rápidas de calcular ya que estas utilizan B-splines de menor grado y diferencias de los coeficientes que los acompañan. En particular, la fórmula

para la segunda derivada de una combinación lineal de funciones B-splines cuando los nodos son equidistantes es:

$$h^2 \sum_j a_j B_j''(x;q) = \sum_j \Delta^2 a_j B_j(x;q-2), \tag{2.1.3}$$

con Δ el operador de diferencias, en donde $\Delta^2 a_j = \Delta(\Delta a_j) = a_j - 2a_{j-1} + a_{j-2}$ y h es la distancia entre nodos. En este trabajo se consideran nodos equidistantes y B-splines de grado 3.

Cuando consideramos una combinación lineal de B-splines para aproximar $f(x) = \sum_{j=1}^{n} a_j B_j(x)$ la Ecuación (2.1.2) puede ser escrita como:

$$S(f) = \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} a_j B_j(x_i) \right)^2 + \lambda \int_{x_{\text{mín}}}^{x_{\text{máx}}} \left(\sum_{j=1}^{p} a_j B_j''(x) \right)^2 dx, \tag{2.1.4}$$

en esta última, los valores a_j deben ser estimados de manera de minimizar $S(f)$. Para llevar a cabo este paso se introduce la siguiente notación vectorial:

- $\boldsymbol{Y} = (y_1, y_2, \ldots, y_n)^T$.
- $\boldsymbol{B} = (B_j(x_i)) = (\boldsymbol{b}_{ij})$ matriz de orden $n \times p$.
- $\boldsymbol{a} = (a_1, a_2, \ldots, a_p)^T$.
- $\boldsymbol{P} = (p_{rs})$ con $p_{rs} = \int B_r'' B_s'' dX$.

Las propiedades de las derivadas de los B-spline [Eilers y Marx, 1996] permiten que el cálculo de la matriz $\boldsymbol{P}$ sea sencillo, ya que como se mencionó anteriormente, esta depende de las diferencias de los coeficientes que acompañan a los B-splines, por lo que se puede reescribir $\boldsymbol{P} = \boldsymbol{K}^T \boldsymbol{K}$, en donde $\boldsymbol{K}$ es una representación matricial del operador de diferencias Δ^2, en este caso, la matriz $\boldsymbol{K}$ será tridiagonal. Con esta notación la función $S(f)$ se puede reescribir como:

$$S(f) = (\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a})^T (\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a}) + \lambda \boldsymbol{a}^T \boldsymbol{K}^T \boldsymbol{K} \boldsymbol{a}. \tag{2.1.5}$$

Ahora, sólo queda por calcular el mínimo de esta función para $\boldsymbol{a}$, esto se hace derivando con respecto al vector $\boldsymbol{a}$ e igualando a cero para encontrar el estimador de mínimos

cuadrados penalizados (PLS).

$$\frac{\partial}{\partial \boldsymbol{a}} S(f) = -2\boldsymbol{B}^T(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a}) + 2\lambda \boldsymbol{K}^T\boldsymbol{K}\boldsymbol{a} = 0, \tag{2.1.6}$$

de esta forma se obtiene que el estimador PLS de $\boldsymbol{a}$ esta dado por:

$$\widehat{\boldsymbol{a}}(\lambda) = (\boldsymbol{B}^T\boldsymbol{B} + \lambda \boldsymbol{K}^T\boldsymbol{K})^{-1}\boldsymbol{B}^T\boldsymbol{Y}. \tag{2.1.7}$$

Se escribe esta ecuación dependiente del valor de λ a propósito, ya que a priori no se conoce cual es el valor correcto de este parámetro. A partir de este punto existen criterios para seleccionar el valor de λ como validación cruzada (CV), validación cruzada generalizada (GCV), C_p de Mallows, del cual se habla en el libro de Ruppert *et al.* [2003], o el criterio de Akaike (AIC) por ejemplo. Algunos de estos ejemplos se han calculado para la base de datos `life.rda` y se muestran en los siguientes gráficos a modo de ejemplo.

Para el cálculo del GCV se utiliza la siguiente ecuación:

$$GCV(\lambda) = \frac{1}{n}\frac{\|(\boldsymbol{I} - \boldsymbol{H}(\lambda))\boldsymbol{Y}\|^2}{\{1 - \mathrm{tr}(\boldsymbol{H}(\lambda))/n\}^2}, \tag{2.1.8}$$

que es un caso especial de la ecuación que se encuentra en el trabajo de Osorio [2016b],en donde hemos considerado $\boldsymbol{W} = \boldsymbol{I}$. De esta forma se obtiene el siguiente par de gráficos en la Figura 2.1, en el primero se traza el valor del GCV para valores de λ entre $(0, 10]$ y en el segundo gráfico se toma un intervalo menor de lambda entre $(0, 0{,}05]$ con el fin de hacer un zoom del primer gráfico para apreciar donde se alcanza el mínimo de esta función. Para este criterio se obtiene un valor de λ cercano a 0,003.

Un algoritmo alternado, el cual es descrito en Osorio [2016b] se encuentra implementado en el paquete heavy [Osorio, 2016a] para el software estadístico R, con esto se obtiene el siguiente gráfico en la Figura 2.2, para este caso se obtiene un valor de λ cercano a 1.356.

Finalmente, el criterio de Akaike es definido mediante la ecuación:

$$AIC = \sum_{i=1}^{m} \frac{(Y_i - \hat{\mu}_i)^2}{\hat{\sigma}_0} + 2\mathrm{tr}(H) - 2m\ln\hat{\sigma}_0 - m\ln 2\pi, \tag{2.1.9}$$

en donde $\hat{\sigma}_0 = \text{var}(\boldsymbol{Y} - \boldsymbol{Y}_{\text{est}})$ cuando $\lambda = 0$ y $\hat{\mu} = \boldsymbol{Ba}$. Esta ecuación se encuentra en el trabajo de Eilers y Marx [1996], para este caso se obtiene un valor de λ cercano a 0.0120.

Se destaca la forma de los gráficos del GCV y del AIC ya que en los primeros gráficos de las Figuras 2.1 y 2.3 no se percibe realmente cuál es el valor correcto de λ y es necesario realizar un segundo gráfico para captar de mejor manera la forma de las curvas, es más, en el caso del AIC si se hubiese escogido un intervalo para λ del estilo $(0{,}2, 10)$ se podría haber escogido un valor de λ mayor al obtenido. Esto da muestras de que la selección del parámetro de suavizamiento para la base de datos `life.rda` es relativamente compleja.

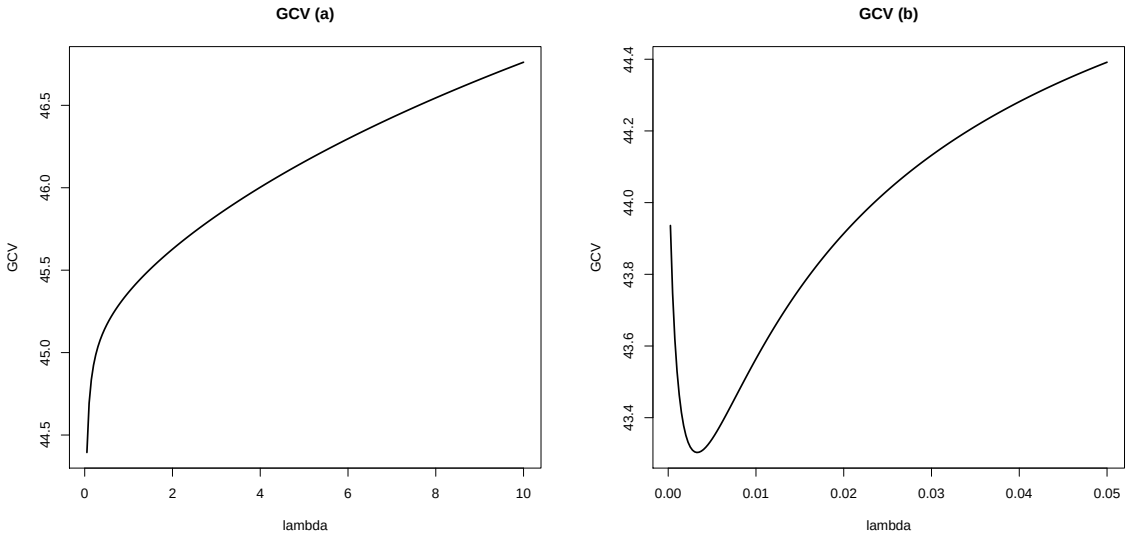


FIGURA 2.1: Gráficos del GCV para distintos valores de λ

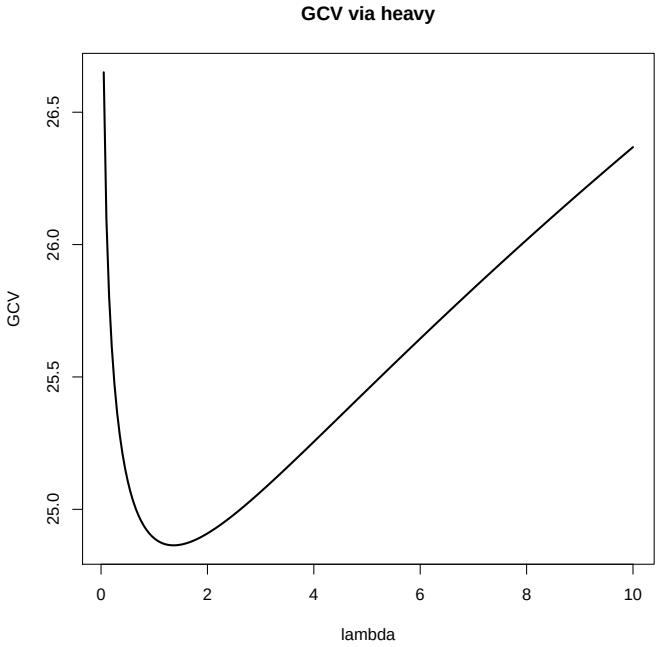


FIGURA 2.2: Gráfico del GCV calculado via el paquete "heavy"para distintos valores de λ

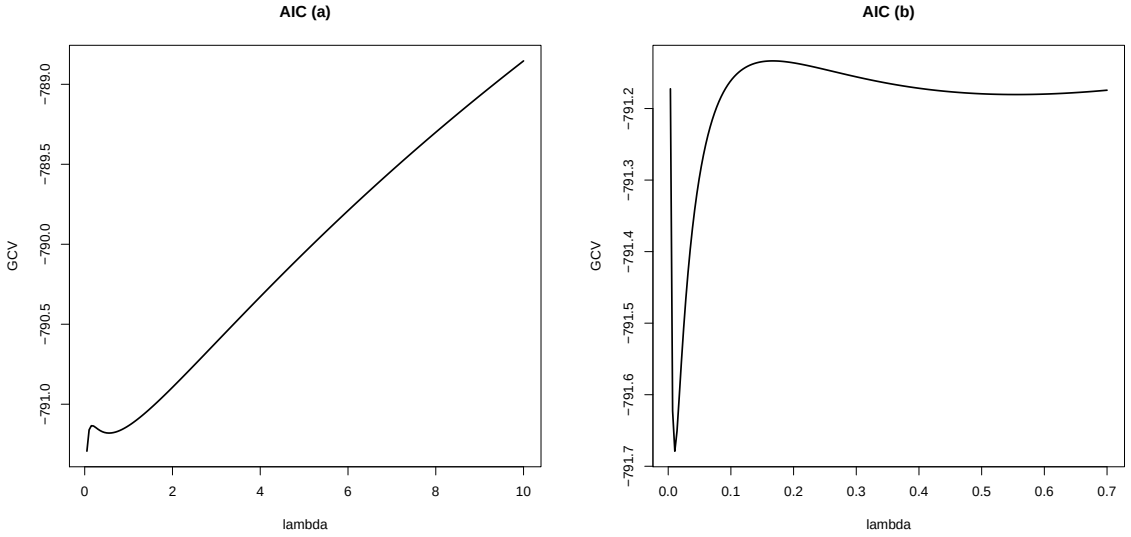


FIGURA 2.3: Gráfico del criterio de Akaike para distintos valores de λ. Similar a la Figura 2.1 el segundo gráfico es un "zoom"del primero.

2.2. Algoritmo EM

El Algoritmo EM es un método iterativo que se usa para calcular estimaciones máximo-verosímiles en problemas de estimación donde se tienen datos perdidos, o en aquellos que se puedan formular de esta forma. Este algoritmo fue introducido por Dempster *et al.* [1977] y desde entonces ha ganado bastante popularidad en el área de Estadística por su versatilidad y simplicidad de implementación. Esta popularidad se ve reflejada en el uso de este algoritmo en numerosos trabajos de investigación y en la vasta bibliografía que se ha escrito al respecto.

El principal problema de la estimación usando el método de máxima verosímilitud, cuando se tienen datos perdidos, es que la función de verosimilitud no puede ser evaluada directamente a causa de los datos faltantes. Para llevar a cabo la estimación en un problema con estas características, el algoritmo EM considera la existencia de un vector aleatorio de datos observados $\boldsymbol{Y}_{\text{obs}}$ y un vector de datos completos $\boldsymbol{Y}_{\text{aug}}$ que contiene tanto los datos observados como los perdidos y que están relacionados mediante un mapeo $\mathcal{M}$ sobreyectivo de modo que $\boldsymbol{Y}_{\text{obs}} = \mathcal{M}\left(\boldsymbol{Y}_{\text{aug}}\right)$, luego se da inicio a un proceso que consta de dos etapas iterativas en las cuales primero se calcula una esperanza condicional (etapa E) y luego se buscan los valores de los parámetros que maximizan la esperanza condicional obtenida en la etapa previa (etapa M), de esta forma comienza una serie de iteraciones en las cuales se buscan las estimaciones máximo-verosímiles. Las etapas antes descritas son las que de dan el nombre al algoritmo EM, etapa E por Esperanza y etapa M por Maximización.

A continuación se enuncia la forma general del algoritmo EM:

Sea $\boldsymbol{Y}$ un vector aleatorio con distribución $p(\boldsymbol{Y};\boldsymbol{\theta})$, en donde $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_d)^T$ es el vector de parámetros que deseamos estimar. Como se menciona al principio, se llama $\boldsymbol{Y}_{\text{obs}}$ a los datos observados y $\boldsymbol{Y}_{\text{aug}} = (\boldsymbol{Y}_{\text{obs}}, \boldsymbol{Y}_{\text{mis}})$ al vector aleatorio de datos completos (observados y perdidos), lo cuales están relacionados por un mapeo sobreyectivo $\mathcal{M}$ de manera que $\boldsymbol{Y}_{\text{obs}} = \mathcal{M}(\boldsymbol{Y}_{\text{aug}})$. Además se anota como $\ell(\boldsymbol{\theta}; \boldsymbol{Y}_{\text{aug}}) = \log p(\boldsymbol{Y}_{\text{aug}}; \boldsymbol{\theta})$ la función de log-verosimilitud de los datos aumentados. Con esto se construye las siguientes etapas que definen al algoritmo EM.

▷ **Etapa E**: Calcular

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)}) = \mathrm{E}\left[\ell(\boldsymbol{\theta}; \boldsymbol{Y}_{\mathrm{aug}})|\boldsymbol{Y}_{\mathrm{obs}}, \boldsymbol{\theta}^{(k)}\right]. \tag{2.2.1}$$

▷ **Etapa M**: Actualizar el valor de $\boldsymbol{\theta}^{(k+1)}$ maximizando $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})$ con respecto de $\boldsymbol{\theta}$, esto es, se debe obtener $\theta^{(k+1)}$ tal que:

$$Q(\boldsymbol{\theta}^{(k+1)}|\boldsymbol{\theta}^{(k)}) \geq Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{k}), \quad \forall \boldsymbol{\theta} \in \boldsymbol{\Theta}. \tag{2.2.2}$$

Se puede demostrar que este proceso incrementa la log-verosimilitud en cada iteración y converge a un punto crítico de $\ell(\boldsymbol{\theta}; \boldsymbol{Y}_{\mathrm{obs}})$ [Dempster *et al.*, 1977]. Para detener el algoritmo EM se debe considerar algún criterio de parada, ya que el algoritmo itera entre estas dos etapas descritas refinando cada vez más el resultado. Usualmente se considera que la diferencia entre las verosimilitudes de los datos observados de dos pasos consecutivos sea menor que alguna tolerancia ε.

$$\|\ell(\boldsymbol{\theta}^{(k+1)}; \boldsymbol{Y}_{\mathrm{obs}}) - \ell(\boldsymbol{\theta}^{(k)}; \boldsymbol{Y}_{\mathrm{obs}})\| \leq \varepsilon. \tag{2.2.3}$$

En casos donde la función de verosimilitud no pueda ser calculada directamente se puede utilizar como criterio de parada la diferencia entre dos estimaciones consecutivas.

$$\|\boldsymbol{\theta}^{(k+1)} - \boldsymbol{\theta}^{(k)}\| \leq \varepsilon \cdot \|\boldsymbol{\theta}^{(k)}\|. \tag{2.2.4}$$

La monotonicidad y convergencia del algoritmo han sido demostradas en el trabajo de Dempster *et al.* [1977] .

2.3. Algoritmo EM anidado

El algoritmo EM anidado nace de la idea de anidar un algoritmo EM dentro de otro con el fin de reducir la fracción de datos perdidos en comparación al algoritmo EM usual, lo cual resulta útil en algoritmos EM con etapas E que que tengan estructuras complejas. En el trabajo de van Dyk [2000] se muestra cómo el anidar dos (o más) algoritmos EM puede generar un algoritmo que tiene una mayor velocidad de convergencia, es fácil de implementar y que posee propiedades de convergencia estables. La desventaja es que debido al anidado cada iteración requerirá más tiempo, en otras palabras, se

obtiene un algoritmo que converge más rápido, pero que sus pasos requerirán mayor esfuerzo computacional.

El anidado se efectúa de la siguiente forma, se considera que el vector de datos aumentados se puede dividir en dos (o más) partes $\boldsymbol{Y}_{\text{aug}_1}$ y $\boldsymbol{Y}_{\text{aug}_2}$ tales que estas puedan ser escritas de la siguiente forma $\boldsymbol{Y}_{\text{obs}} = \mathcal{M}_1(\boldsymbol{Y}_{\text{aug}_1})$ y $\boldsymbol{Y}_{\text{aug}_1} = \mathcal{M}_2(\boldsymbol{Y}_{\text{aug}_2})$ para $\mathcal{M}_1$ y $\mathcal{M}_2$ dos mapeos sobreyectivos asociados al modelo del vector de datos aumentados.

De forma similar que en la sección previa, se definen dos funciones de esperanzas condicionales $Q_1(\boldsymbol{\theta}|\boldsymbol{\theta}_0) = \text{E}\left[\ell(\boldsymbol{\theta}; \boldsymbol{Y}_{\text{aug}_1})|\boldsymbol{Y}_{\text{obs}}, \boldsymbol{\theta}_0\right]$ y $Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}_0) = \text{E}\left[\ell(\boldsymbol{\theta}; \boldsymbol{Y}_{\text{aug}_2})|\boldsymbol{Y}_{\text{obs}}, \boldsymbol{\theta}_0\right]$, en donde $\ell(\boldsymbol{\theta}, \cdot)$ son funciones de log-verosimilitud de los vectores asociados, con esto se define la función:

$$Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}_{01}, \boldsymbol{\theta}_{02}) = \text{E}\left[\text{E}\left[\ell(\boldsymbol{\theta}; \boldsymbol{Y}_{\text{aug}_2})|\boldsymbol{Y}_{\text{aug}_1}, \boldsymbol{\theta}_{01}\right]|\boldsymbol{Y}_{\text{obs}}, \boldsymbol{\theta}_{02}\right], \tag{2.3.1}$$

esta función considera a $\boldsymbol{Y}_{\text{aug}_1}$ como un vector de datos observados para el vector de datos aumentados $\boldsymbol{Y}_{\text{aug}_2}$. Luego, al igual que el algoritmo EM estándar, se formula el proceso iterativo que consta de las etapas E y M considerando el anidado antes descrito de la siguiente forma:

▷ **Etapa E**: Calcular

$$Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k+\frac{t}{T})}, \boldsymbol{\theta}^{(k)}) = \text{E}\left[\text{E}\left[\ell(\boldsymbol{\theta}; \boldsymbol{Y}_{\text{aug}_2})|\boldsymbol{Y}_{\text{aug}_1}, \boldsymbol{\theta}^{(k+\frac{t}{T})}\right]|\boldsymbol{Y}_{\text{obs}}, \boldsymbol{\theta}^{(k)}\right] \tag{2.3.2}$$

▷ **Etapa M**: Actualizar el valor de $\boldsymbol{\theta}^{(k+\frac{t+1}{T})}$ al maximizar $Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k+\frac{t}{T})}, \boldsymbol{\theta}^{(k)})$ con respecto de $\boldsymbol{\theta}$, esto es

$$Q_{21}(\boldsymbol{\theta}^{(k+\frac{t+1}{T})}|\boldsymbol{\theta}^{(k+\frac{t}{T})}, \boldsymbol{\theta}^{(k)}) \geq Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k+\frac{t}{T})}, \boldsymbol{\theta}^{(k)}), \quad \forall \boldsymbol{\theta} \in \boldsymbol{\Theta} \tag{2.3.3}$$

Contrariamente al algoritmo EM, la interna del algoritmo EM anidado se ejecuta con un número fijo T de ciclos, estos ciclos se encuentran indexados por el término t con $1 \leq t \leq T$, esto quiere decir que se llevarán a cabo T ciclos internos antes de pasar a una nueva iteración $(k+1)$ del ciclo externo, dicho de otra forma, cuando se completan T iteraciones del ciclo interno se toma $\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k+\frac{T}{T})}$, se inicia una nueva iteración y se vuelven a iniciar los ciclos internos.

Capítulo 3

Estimación del parámetro de suavizamiento

En en la Sección 2.1 se presentó el suavizamiento P-spline y se dieron a conocer algunos de los criterios que se usan para escoger el parámetro de suavizamiento.Aún cuando el criterio de validación cruzada generalizada ha sido criticado por subestimar el verdadero valor de λ existe en la literatura de problemas inversos una serie de procedimientos para la selección del parámetro de suavizamiento (o de regulación) (ver, por ejemplo, Hansen [2010], Cáp. 5). En este trabajo abordamos un procedimiento para estimar el parámetro de suavizamiento en P-splines mediante el uso del algoritmo EM anidado.

Considere el siguiente modelo de regresión:

$$Y_i|\boldsymbol{a},\tau_i \overset{ind}{\sim} \mathcal{N}(\boldsymbol{b_i}^T\boldsymbol{a},\tfrac{\phi}{\tau_i}),\quad \tau_i \overset{ind}{\sim} \mathcal{H}(\tau_i;\nu),\quad \boldsymbol{a}\sim\mathcal{N}_p(\boldsymbol{0},\tfrac{\phi}{\lambda}(\boldsymbol{K}^T\boldsymbol{K})^-),\quad i=1,\ldots,n, \tag{3.0.1}$$

en donde τ_i es una variable aleatoria positiva con función de distribución acumulada $\mathcal{H}$. En este modelo la variable $\boldsymbol{Y}$ asume una mezcla de escala de normal [Andrews y Mallows, 1974]. La elección de este modelo se debe principalmente debido a sus características de robustez, así como de simpleza computacional.

Para plantear el algoritmo EM anidado usando el modelo anterior se consideran los vectores de datos aumentados $\boldsymbol{Y}_{\text{aug}_2} = (\boldsymbol{Y}^T,\boldsymbol{\tau}^T,\boldsymbol{a}^T)^T$ y $\boldsymbol{Y}_{\text{aug}_1} = (\boldsymbol{Y}^T,\boldsymbol{\tau}^T)^T$ en donde $\boldsymbol{\tau} = (\tau_1,\ldots,\tau_n)^T$ y $\boldsymbol{a}$ son consideradas variables perdidas mientras que $\boldsymbol{Y}$ son los

datos observados. A continuación se escriben las etapas del algoritmo EM anidado en función del modelo (3.0.1), las cuales se mencionan en el Capítulo de Preliminares.

Dado que el vector a puede ser visto como un efecto aleatorio, en este caso tenemos que el vector de parámetros de interés es $\boldsymbol{\theta} = (\phi, \lambda)^T$. Para poder escribir explícitamente el valor de Q_{21} es necesario calcular paso a paso las funciones necesarias que componen su estructura, partiendo por la función de log-verosimilitud $\ell_{\mathrm{a}}(\boldsymbol{\theta}; \boldsymbol{Y}_{\mathrm{aug}_2})$ para luego ser reemplazada en las esperanzas condicionales de Q_{21}. Estos cálculos se han anexado el Apéndice A en donde se presentan las derivaciones en una forma más exhaustiva. A continuación, se muestran los resultados principales que se necesitan para planter el algoritmo.

La función de esperanzas condicionales $Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k+\frac{t}{T})}, \boldsymbol{\theta}^{(k)})$ queda escrita de la siguiente forma:

$$
\begin{aligned}
&Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k+\frac{t}{T})}, \boldsymbol{\theta}^{(k)}) = -\frac{n+p}{2}\log\phi + \frac{p}{2}\log\lambda \\
&- \frac{1}{2\phi}\mathrm{E}\Big[S_W\left(\mathbf{a}_W(\lambda^{(k+\frac{t}{T})})\right) + \lambda \boldsymbol{a}_W^T(\lambda^{(k+\frac{t}{T})})\boldsymbol{K}^T\boldsymbol{K}\boldsymbol{a}_W(\lambda^{(k+\frac{t}{T})})\Big|\mathbf{Y}, \boldsymbol{\theta}^{(k)}\Big] \\
&+ \frac{\phi^{(k+\frac{t}{T})}}{2\phi}\mathrm{tr}\left\{\mathrm{E}\Big[(\mathbf{B}^T\mathbf{W}\mathbf{B} + \lambda\boldsymbol{K}^T\boldsymbol{K})(\mathbf{B}^T\mathbf{W}\mathbf{B} + \lambda^{(k+\frac{t}{T})}\boldsymbol{K}^T\boldsymbol{K})^{-1}|\mathbf{Y}, \boldsymbol{\theta}^{(k)}\Big]\right\},
\end{aligned}
\tag{3.0.2}
$$

con $S_W(\boldsymbol{a}) = (\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a})^T\boldsymbol{W}(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a})$ y $\boldsymbol{a}_W(\lambda) = (\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda\boldsymbol{K}^T\boldsymbol{K})^{-1}\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{Y}$.

Ahora se procede a buscar aquellos valores de λ y ϕ que maximizan la ecuación (3.0.2) (ver Apéndice A). Los valores de $\phi^{(k+t/T)}$ y $\lambda^{(k+t/T)}$ asumen las siguientes expresiones:

$$
\begin{aligned}
\phi^{(k+\frac{t+1}{T})} &= \frac{1}{n+p}\left(p\phi^{(k+\frac{t}{T})} + E_1 + \lambda^{(k+\frac{t}{T})}E_2\right) \\
\lambda^{(k+\frac{t+1}{T})} &= \frac{p\phi^{(k+\frac{t+1}{T})}}{E_2 + \phi^{(k+\frac{t+1}{T})}\cdot\mathrm{tr}\big(\boldsymbol{K}^T\boldsymbol{K}\cdot E_3\big)},
\end{aligned}
$$

en donde:

$$E_1 = \mathrm{E}\left[S_W(\mathbf{a}_W(\lambda^{(k+\frac{t}{T})}))\middle|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\right]$$
$$E_2 = \mathrm{E}\left[\left\|\boldsymbol{K}\boldsymbol{a}_W(\lambda^{(k+\frac{t}{T})})\right\|^2\middle|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\right]$$
$$E_3 = \mathrm{E}\left[(\mathbf{B}^T\mathbf{W}\mathbf{B} + \lambda^{(k+\frac{t}{T})}\boldsymbol{K}^T)^{-1}\middle|\mathbf{Y}, \boldsymbol{\theta}^{(k)}\right].$$

Además $\boldsymbol{K}$ es una matriz que cumple con $\boldsymbol{P} = \boldsymbol{K}^T\boldsymbol{K}$ y tr() es el operador traza.

Las esperanzas condicionales E_1, E_2 y E_3 que se utilizan en el cálculo de λ y ϕ lamentablemente no tienen una forma explícita, por lo que estas deben ser estimadas usando algún método estocástico, en este caso se ha decidido aproximarlas usando un algoritmo Monte Carlo. Este método permite aproximar esperanzas condicionales mediante el cálculo de promedios muestrales, obtenidos a partir de datos simulados. De este modo, mientras más términos se consideren mayor será la precisión de la aproximación.

3.1. Esquema del algoritmo

Manteniendo la notación introducida en los capítulos previos se presenta a continuación un esquema del algoritmo, el cual se divide en dos partes, en la primera se muestran algunos cálculos preliminares así como nuestras elecciones para estimaciones iniciales, y en la segunda parte se muestra en detalle el proceso de estimación:

Algorithm 1 Cálculos previos al inicio del algoritmo de estimación

1: Definir $\boldsymbol{X}$ y $\boldsymbol{Y}$, seleccionar los valores del grado del B-Spline (`deg`), el número de nodos (`nknots`), `tolerancia` y ν
2: $n = \texttt{length}(\mathbf{X})$, $\texttt{nseg} = \texttt{nknots} - 2 \cdot \texttt{deg} - 1$, $p = \texttt{nseg} + 3$
3: Calcular las matrices $\boldsymbol{B}, \boldsymbol{K}$ y $\boldsymbol{P}$
4: Seleccionar los valores iniciales de $\lambda^{(0)}$ y $\phi^{(0)}$
5: Calcular $\boldsymbol{a} = (\boldsymbol{B}^T\boldsymbol{B} + \lambda\boldsymbol{p})^{-1}\boldsymbol{B}^T\boldsymbol{Y}$ y $D = (\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a})^2/\phi$

Algorithm 2 Inicio del Algoritmo de estimación

while (`ecm` $>$ `tolerancia`) **do**
  for (i=1:T) **do**
    $$N = \begin{cases} 5, & \text{si} \quad \texttt{it} \leq 5 \\ 20, & \text{si} \quad 5 < \texttt{it} \leq 10 \\ 100, & \text{si} \quad 10 < \texttt{it} \leq 20 \\ 200, & \text{si} \quad 20 < \texttt{it} \end{cases}$$
    for (l=1:N) **do**
      for (j in 1:n) **do**
        $\boldsymbol{W}[j,j] = \texttt{rgamma}(1, (\nu+1)/2, (\nu + \boldsymbol{D}[j])/2)$
      end for
      $\boldsymbol{a}_W = (\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda\boldsymbol{P})^{-1}\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{Y}$
      $S_W = (\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a}_W)^T\boldsymbol{W}(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a}_W)$
      $E1 = E1 + S_W/N$
      $E2 = E2 + (\boldsymbol{K}\boldsymbol{a}_W)^T\boldsymbol{K}\boldsymbol{a}_W/N$
      $E3 = E3 + (\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda \cdot \boldsymbol{P})^{-1}/N$
      $\boldsymbol{a}\_\text{est} = \boldsymbol{a}\_\text{est} + \boldsymbol{a}_W/N$
    end for
    $\phi = \frac{1}{(n+p)} \cdot (p \cdot \phi + E1[1] + \lambda \cdot E2)$
    $\lambda = (p \cdot \phi)/(E2 + \phi \cdot \text{tr}P \cdot E3)$
    $\boldsymbol{a}\text{Mc} = \boldsymbol{a}\text{Mc} + \boldsymbol{a}\_\text{est}/T$
  end for
  $a = \boldsymbol{a}\text{Mc}$
  $D = ((\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a})^2)/\phi$
  $\texttt{it} = \texttt{it} + 1$
end while

Observaciones:

1- El cálculo de las matrices $\boldsymbol{B}$ y $\boldsymbol{K}$ se ha hecho mediante el uso de funciones para construir bases B-Splines que se describen en el trabajo de Eilers y Marx [2010], estas se encuentran en el Apéndice B.

2- El ciclo interno por lo general avanza rápido paso a paso por lo que se recomienda usar pequeños valores para T, en nuestro caso se ha tomado $T = 7$.

3- Las cadenas Monte Carlo de las estimaciones de las esperanzas condicionales se han escogido de distintos tamaños conforme avanza el algoritmo, esto con el fin de agilizar las estimaciones en los pasos iniciales y dejando el refinamiento de estas para los últimos pasos.

3.2. Cálculo del error estándar

Una de las críticas que se le hace tanto al algoritmo EM como a su variante anidada es que, a diferencia de los métodos tipo Newton-Raphson, estos carecen del cálculo sistemático de la matriz de covarianza de las estimaciones máximo verosímiles, la cual entrega una noción de la precisión de los estimadores máximo verosímiles. Ante esta problemática existen trabajos como los de Lee y Pawitan [2014] y Oakes [1999] los cuales dan alternativas para calcular la matriz Hessiana en el Algoritmo EM estándar y a partir de esta, obtener una estimación de la matriz de covarianza, la cual permite tener una estimación del error estándar mediante la raíz cuadrada de los elementos de su diagonal. Para poder usar estos métodos en el caso del algoritmo EM anidado se aprovecha la estructura de este y se utiliza la propiedad $Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)}, \boldsymbol{\theta}^{(k)}) = Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})$ [van Dyk, 2000].

Del trabajo de Oakes [1999], la ecuación que permite calcular la matriz Hessiana es:

$$\frac{\partial^2 \ell(\boldsymbol{\theta}; \boldsymbol{Y})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} = \left\{ \frac{\partial^2 Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} + \frac{\partial^2 Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^{(k)^T}} \right\} \Bigg|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}}. \tag{3.2.1}$$

Notando que:

$$Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)}) = Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)}, \boldsymbol{\theta}^{(k)}) = \mathrm{E}\left[\mathrm{E}\left[\ell_{\mathrm{a}}(\boldsymbol{\theta}; \boldsymbol{Y}_{\mathrm{aug}_2})|\boldsymbol{Y}_{\mathrm{aug}_1}, \boldsymbol{\theta}^{(k)}\right]|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\right], \tag{3.2.2}$$

en donde $Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})$ queda escrito como:

$$\begin{aligned} Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)}) &= -\frac{n+p}{2}\log\phi + \frac{p}{2}\log\lambda \\ &- \frac{1}{2\phi}\mathrm{E}\left[S_W\left(\mathbf{a}_W(\lambda^{(k)})\right) + \lambda\mathbf{a}_W^T(\lambda^{(k)})\mathbf{P}\mathbf{a}_W(\lambda^{(k)})|\mathbf{Y},\boldsymbol{\theta}^{(k)}\right] \\ &+ \frac{\phi^{(k)}}{2\phi}\cdot\mathrm{tr}\left\{\mathrm{E}\left[(\mathbf{B}^T\mathbf{W}\mathbf{B}+\lambda\mathbf{P})(\mathbf{B}^T\mathbf{W}\mathbf{B}+\lambda^{(k)}\mathbf{P})^{-1}|\mathbf{Y},\boldsymbol{\theta}^{(k)}\right]\right\}. \end{aligned} \tag{3.2.3}$$

De esta forma, es posible calcular las derivadas parciales del lado derecho de la ecuación (3.2.1), las cuales son matrices de 2×2

$$\frac{\partial^2 Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^T}\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}} = \left(\begin{array}{cc} \frac{\partial^2 Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})}{\partial\phi^2} & \frac{\partial^2 Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})}{\partial\phi\partial\lambda} \\ \frac{\partial^2 Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})}{\partial\lambda\partial\phi} & \frac{\partial^2 Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})}{\partial\lambda^2} \end{array}\right)\Bigg|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}}, \tag{3.2.4}$$

$$\frac{\partial^2 Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^{(k)^T}}\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}} = \left(\begin{array}{cc} \frac{\partial^2 Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})}{\partial\phi\partial\phi^{(k)}} & \frac{\partial^2 Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})}{\partial\phi\partial\lambda^{(k)}} \\ \frac{\partial^2 Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})}{\partial\lambda\partial\phi^{(k)}} & \frac{\partial^2 Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})}{\partial\lambda\partial\lambda^{(k)}} \end{array}\right)\Bigg|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}}. \tag{3.2.5}$$

Los resultados de estas ecuaciones se adjuntan en el Apéndice A.

Capítulo 4

Aplicaciones

En el capítulo anterior se desarrolló la metodología propuesta dando su marco teórico e indicando sus pasos para el proceso de estimación así como el cálculo para el error estandar. En este capítulo se realizan experimentos numéricos para evaluar el desempeño del algoritmo. Es de interés conocer características de éste como por ejemplo, el tiempo de ejecución o el error de las estimaciones obtenidas. Para estudiar las propiedades del algoritmo se decide realizar dos tipos de análisis estadísticos. Primero, se realiza un ajuste a una base de datos que presenta observaciones que pueden ser catalogadas como atípicas, el conjunto de datos de esperanza de vida (`life.rda`) [Leinhardt y Wasserman, 1979] posee 101 datos que contienen información sobre la esperanza de vida y el ingreso per cápita de 101 países en el año 1979. Esta base de datos fue introducida originalmente por Leinhardt y Wasserman [1979] y también ha sido estudiada en los trabajos de Thomas [1991] y Osorio [2016b]. En segundo lugar se realiza un estudio de simulación usando una función de prueba y agregando errores con contaminación en la varianza, esto con el fin de simular datos con observaciones atípicas que requieran un método robusto para la estimación, de esta forma se generaron 500 bases de datos con 100 observaciones cada una para distintos niveles de contaminación, luego se realiza un ajuste a estas simulaciones y se calcula el error cuadrático medio, esta información es resumida mediante gráficos de cajón con bigotes o box-plot. En último lugar, se analiza la base de datos `balloon.rda` la cual contiene 4984 observaciones correspondientes a mediciones realizadas por un globo meteorológico, en este experimento se observa cómo se comporta el algoritmo al analizar bases de datos de tamaño medio. Todos los cálculos y gráficos se realizaron con el software estadístico R [R Core Team, 2016].

Previo a los experimentos descritos se describen algunos parámetros del algoritmo, puesto que éstos valores pueden tener un impacto sobre el desempeño del mismo, estos parámetros son, por ejemplo, el número de nodos usados en el ajuste, el largo de las cadenas Monte Carlo utilizado para el cálculo de las esperanzas condicionales o los valores iniciales de λ y ϕ escogidos para iniciar el algoritmo, entre otros.

4.1. Acerca de los ajustes realizados

El algoritmo es controlado por ciertos parámetros que se deben escoger antes de iniciar el proceso de estimación, en esta sección se describen algunos aspectos relacionados con la selección de estos parámetros. En paréntesis se muestra el nombre que se usa en el código para estos parámetros, el código usado se encuentra en el Apéndice B.

- El grado de los splines (`deg`), este es el grado de las funciones base splines. En este trabajo se consideran splines de grado 3, es decir, splines cúbicos. Se escogen de esta forma debido a que sus segundas derivadas son fáciles de calcular según lo visto en la Sección 2.1, además, se considera que son un buen consenso entre suavidad del spline y el esfuerzo computacional que agrega, ya que al ser de grado 3 la matriz $\boldsymbol{P} = \boldsymbol{K}^T\boldsymbol{K}$ será una matriz banda de ancho 7.

- El número de nodos (`nknots`) y el número de segmentos (`nseg`), estos se encuentran relacionados de la siguiente forma `nseg=nknots-2*deg-1` en donde `deg` es el grado de los splines antes mencionado. El número de nodos escogido tiene que ver con la partición del dominio de la base de datos a ajustar. Instintivamente mientras más nodos se escogen el ajuste será más moldeable, aunque esto también le puede aportar mayor rugosidad a la curva ajustada. Además un mayor número de nodos requerirá un mayor gasto computacional al momento de hacer los cálculos. En este trabajo se consideran distintos números de nodos según el experimento que se realiza, en el trabajo de Ruppert [2002] se discuten algunos criterios para escoger la cantidad de nodos.

- Grados de libertad del modelo (`nu`), los grados de libertad del modelo escogido guardan relación con la distribución de las variables aleatorias τ, en este caso se considera que las variables τ siguen una distribución t de Student o slash.

- Largo de las cadenas Monte Carlo (`N`), la aproximación Monte Carlo es un proceso iterativo que aumenta su calidad a medida que se escoge un mayor número

de iteraciones. Teniendo esto en cuenta se decide tomar pocas iteraciones para las estimaciones iniciales y se toma un mayor número de iteraciones para los cálculos finales, esto con el fin de agilizar el algoritmo en los primeros pasos al considerar estimaciones rápidas y al refinar sólo los resultados de las estimaciones finales para tener resultados más exactos, de esta forma para las primeras se considera el siguiente esquema para el largo de las cadenas:

$$N = \begin{cases} 5, & \text{si} \quad \texttt{it} \leq 5 \\ 20, & \text{si} \quad 5 < \texttt{it} \leq 10 \\ 100, & \text{si} \quad 10 < \texttt{it} \leq 20 \\ 200, & \text{si} \quad 20 < \texttt{it} \end{cases}$$

en donde `it` es el número de la iteración.

- Tolerancia (`tolerancia`). La tolerancia es el valor escogido para el criterio de parada, el cual en este caso es medir el error cuadrático medio en estimaciones consecutivas. Se ha considerado que cuando esta diferencia sea menor que 0.01 se dará por finalizado el algoritmo.

- Valores iniciales de lambda (`lambda`) y phi (`phi`). Estos corresponden a los valores iniciales que se escogen para dar inicio al algoritmo. En esta ocasión se utilizó $\lambda^{(0)} = 0$ y $\phi^{(0)} = 1$.

4.2. Análisis base de datos: Life Expectancy v/s PIB per cápita

Los primeros datos analizados corresponden a la base de datos `life.rda`, la cual contiene 101 observaciones con información sobre el ingreso per capita (PIB per cápita) de ciertos países en el año 1979 versus la esperanza de vida de estos (medida en años), esta base de datos fue introducida y analizada originalmente en el trabajo de Leinhardt y Wasserman [1979]. Actualmente en la página `http://www.gapminder.org/world/` se encuentra una actualización más reciente de esta base de datos y con más países incluidos.

Lo interesante de estos datos es que hay observaciones que pueden ser catalogadas como atípicas y que dificultan los ajustes tradicionales. En la práctica ante problemas de este tipo se puede optar por buscar criterios de identificación y clasificación de

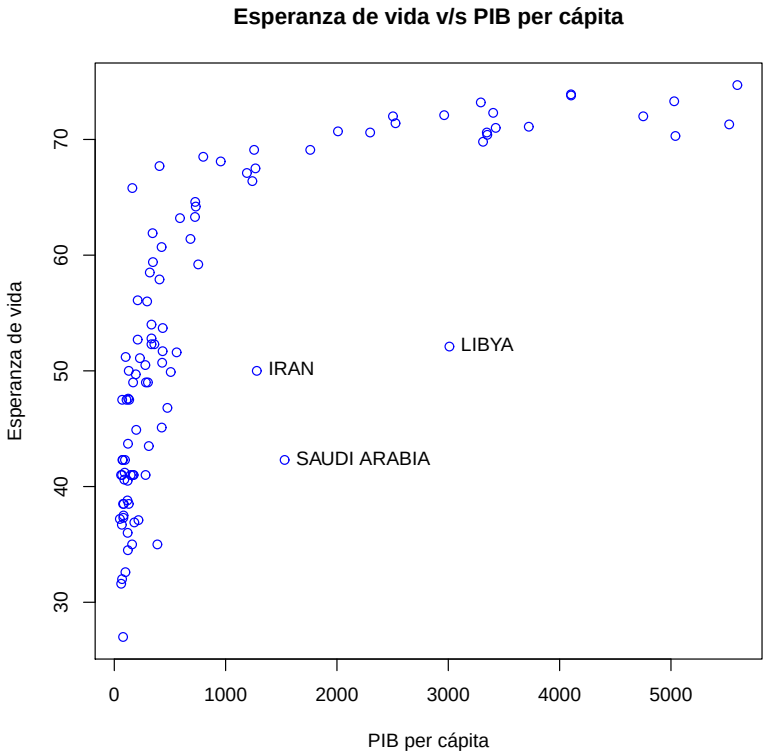


FIGURA 4.1: Base de datos life donde se identifican 3 posibles outliers.

datos atípicos para quitarlos y de esta forma mejorar el ajuste, o también, se puede optar por usar métodos robustos de estimación (como el propuesto en este trabajo).

La información de la base de datos `life.rda` se presenta en el gráfico de la Figura 4.1. Los países Iran, Libia y Arabia Saudita corresponden a observaciones que podrían ser catalogadas como escapadas, ya que estos se encuentran alejados del resto de las observaciones, esto genera que el suavizamiento mediante P-splines sea fuertemente afectado, tal como se muestra en la Figura 4.2.

En la Figura 4.2, en la curva roja se traza un ajuste a la base de datos usando el método propuesto pero con errores normales, como se puede observar esta curva se ve influenciada por las tres observaciones marcadas al ser esta desplazada hacia abajo. Para tener una noción de cuanto afectan en realidad estas observaciones atípicas, se dibuja en la curva verde un ajuste a la base de datos pero quitando las tres observaciones antes mencionadas. De esta forma se puede observar que la curva verde sigue de mejor manera la media de las observaciones en comparación a la curva roja.

En la curva naranja se observa el ajuste realizado con la función `heavyPS` disponible en el paquete `heavy` [Osorio, 2016a] del programa estadístico R [R Core Team, 2016]

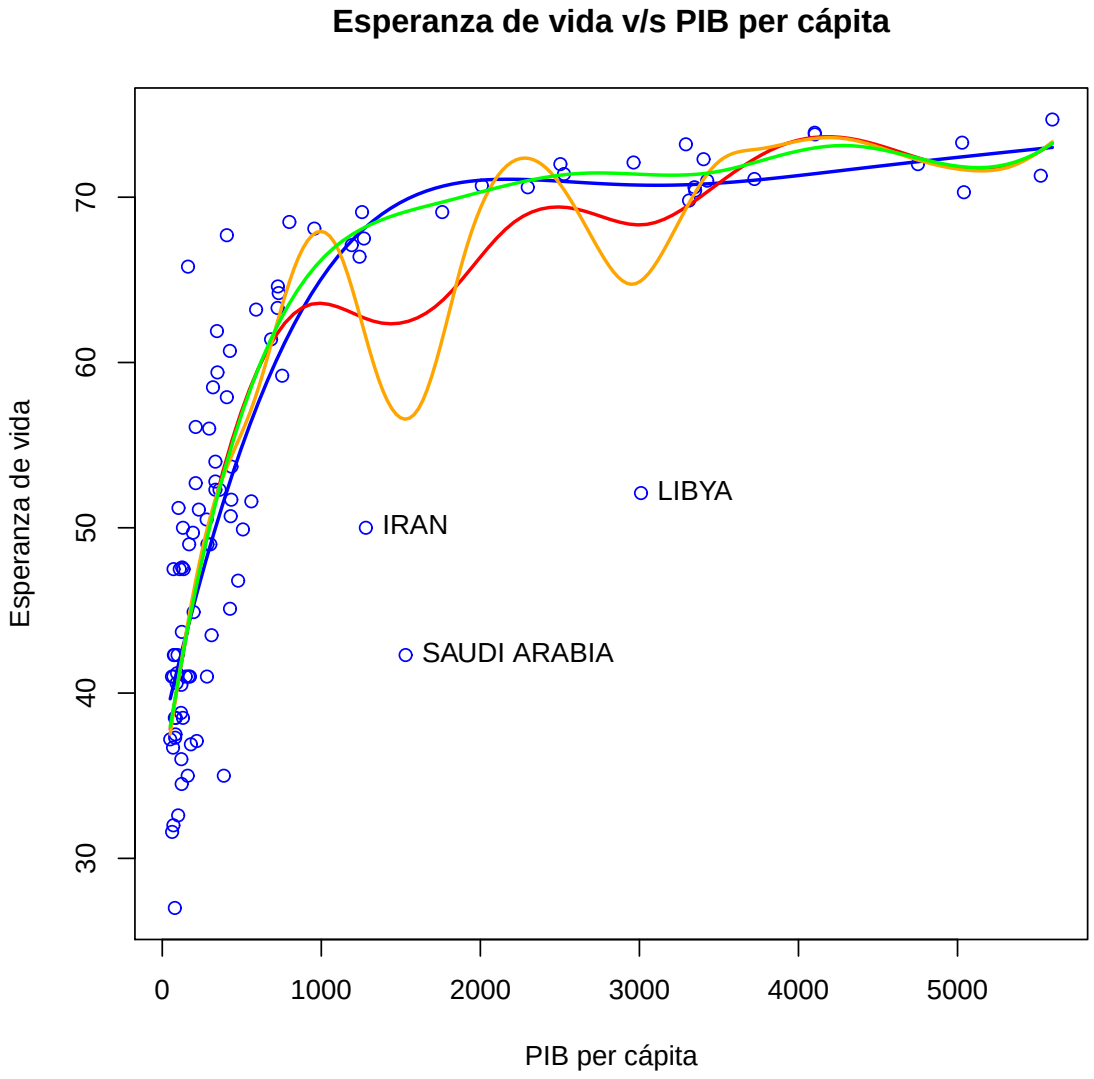


FIGURA 4.2: Ajuste P-Splines con errores normales (curva roja), ajuste P-spline robusto (curva azul) y ajuste P-Splines con errores normales quitando las 3 observaciones catalogadas como outliers (curva verde).

usando una errores normales. Se puede ver que este ajuste es el más afectado por los datos outliers.

Finalmente, en la curva azul se ha trazado el ajuste usando el método propuesto a toda la base de datos (es decir, sin quitar ninguna observación), se aprecia que esta tiene un comportamiento similar al de la curva verde, con la diferencia de que en este caso no fue necesario retirar ningún dato.

Al inicio del Capítulo mencionan brevemente ciertos parámetros fijos que se escogen al iniciar el algoritmo. Para ver cómo cambia el desempeño de este se presenta una

tabla comparando los valores obtenidos para distintos valores de los grados de libertad y distinto número de nodos. Principalmente vemos como estos dos parámetros afectan a los valores estimados y al tiempo de ejecución. Además, acompañando a las estimaciones, se encuentra el error estándar estimado [Oakes, 1999] calculado en conjunto con el algoritmo y en paréntesis acompañando al número de nodos se encuentra el número de segmentos.

CUADRO 4.1: Tabla comparativa de ajustes usando distintos valores para los grados de libertad y para el número de nodos

ν	nodos	valor de $\hat{\lambda}$	Error Estándar	valor de $\hat{\phi}$	Error Estándar	tiempo [seg]
2	10 (3)	0.003	2.165	14.962	0.002	8.42
	20 (13)	7.741	2.157	15.351	2.287	5.76
	28 (21)	89.072	2.167	15.473	20.206	18.55
4	10 (3)	0.005	3.362	23.161	0.004	7.86
	20 (13)	14.728	3.334	23.368	4.319	14.04
	28 (21)	163.466	3.347	23.632	36.890	9.27
8	10 (3)	0.009	4.573	31.289	0.007	11.38
	20 (13)	22.969	4.473	31.651	6.723	4.97
	28 (21)	256.999	4.510	31.791	57.946	14.11

La información resumida en la Tabla 4.1 muestra cómo influye la cantidad de nodos escogidos en el tiempo de ejecución del algoritmo, al tener una mayor partición del dominio, las matrices involucradas tienen mayor dimensión, por lo que se incurre en un mayor gasto numérico. Similar con los grados de libertad escogidos, al aumentarlos también se observa un aumento del tiempo de ejecución.

4.3. Experimentos numéricos: estudio de simulación

En esta sección se considera el siguiente experimento: Usando la función de prueba $f(x) = \sin(2\pi(1-x)^2)$ con dominio en el intervalo $(0, 1)$ se simulan bases de datos que contienen un error aleatorio con contaminación, es decir, un error aleatorio que

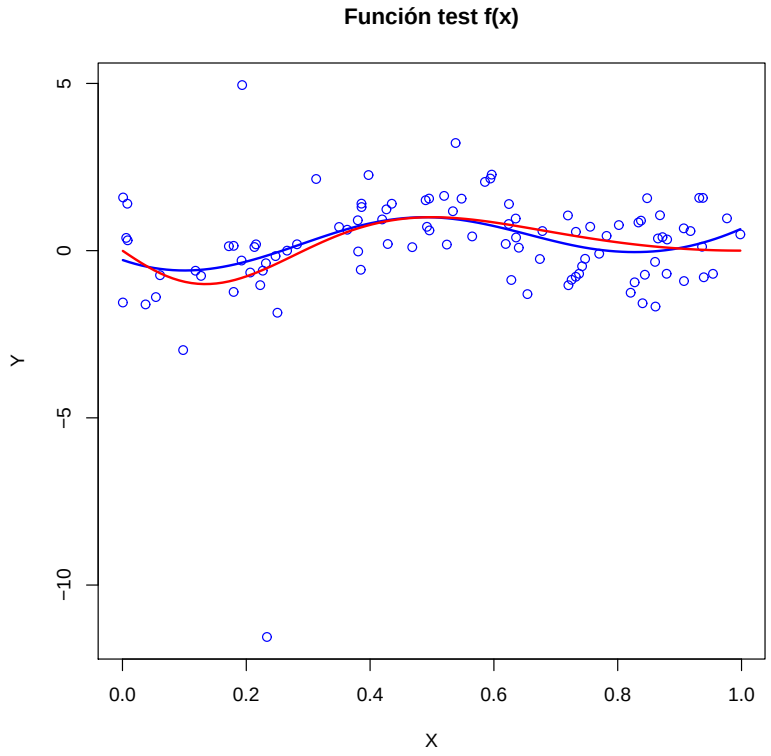


FIGURA 4.3: Ejemplo de ajuste a la base de datos generada a partir de $f(x)$. En rojo se muestra la función original y en azul la curva ajustada.

tiene la siguiente distribución:

$$(1-\delta)\mathcal{N}(0,1)+\delta\mathcal{N}(0,\gamma) \tag{4.3.1}$$

Así, con una probabilidad de $(1-\delta)$ los errores provienen de una distribución $\mathcal{N}(0,1)$ y con una probabilidad de δ estos vienen de una distribución $\mathcal{N}(0,\gamma)$, a este valor γ lo llamamos nivel de inflación de varianza. Esto con el fin de poder reconstruir la función original a partir de la base de datos contaminados y luego calcular el error cuadrático al comparar el ajuste con la función original. A modo de ejemplo, en la Figura 4.3 se muestra una simulación con su respectivo ajuste en la curva azul, y en la curva roja se representa la función $f(x)$, en esta simulación se usó una contaminación de varianza de $10\,\%$, y un nivel de inflación de varianza de $\gamma = 4$.

Entonces, el experimento consistió en generar 500 bases de datos de 100 observaciones para distintos niveles inflación de varianza $\gamma = 2, 4, 10$ y distintos porcentajes de contaminación $\delta = 0\,\%, 5\,\%, 10\,\%, 25\,\%, 40\,\%$, y en cada caso, se calculó el error

cuadrático medio de cada ajuste y resumir esta información mediante gráficos boxplot. Los ajustes se realizaron usando 15 nodos, 4 grados de libertad y un criterio de parada de 0.05.

Como se observa en la Figuras 4.4, en el primer gráfico boxplot que corresponde al experimento realizado con una inflación de varianza de $\gamma = 2$. Se puede observar que el comportamiento de los boxplot es bastante similar a medida que se aumentan los porcentajes de contaminación. Este mismo comportamiento se ve en el segundo gráfico en donde el nivel de inflación de varianza es de $\gamma = 4$, esto nos permite observar la robustez del método aún en altos niveles de contaminación. Finalmente en el tercer gráfico se aprecia que los últimos porcentajes de contaminación logran afectar a un mayor número de ajustes, pero aún así la media de estos se encuentra cerca a las otras.

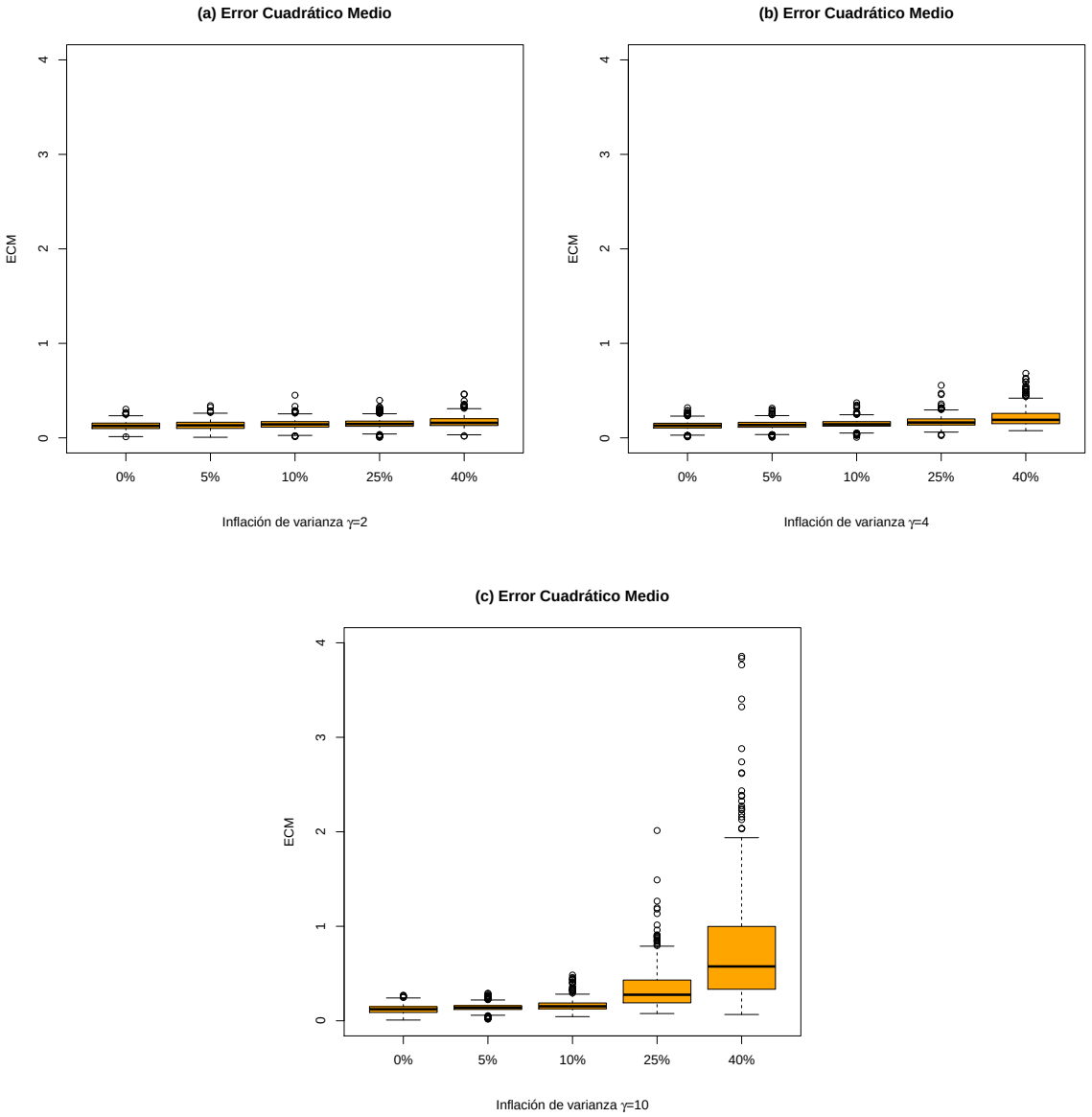
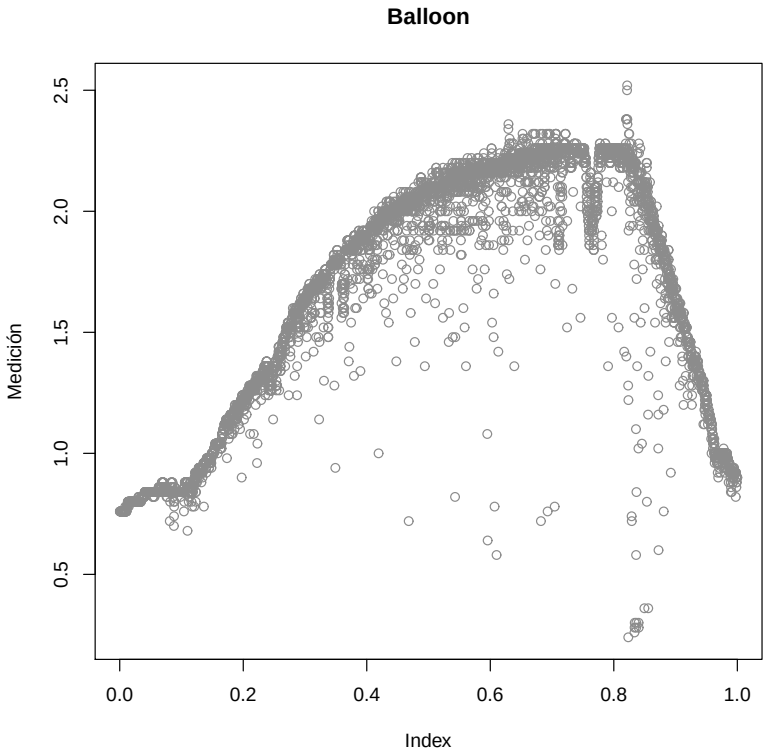


FIGURA 4.4: Gráficos de boxplot para distintos porcentajes de contaminación y distintos niveles de contaminación de varianza

4.4. Análisis base de datos: Balloon



Como una última prueba se ha realizado un ajuste a la base de datos `balloon.rda` con el fin de testear el tiempo de ejecución cuando se tienen muchos datos, la base de datos `balloon.rda` posee 4984 observaciones que corresponden a mediciones realizadas por un globo meteorológico, por lo que estas se encuentran ordenadas según el orden en que fueron medidas, en la imagen se puede observar que en una de las secciones del gráfico hay una caída de las observaciones, una de las explicaciones de esto es que el globo fue rotado por el viento y su propia sombra obstruyó las mediciones.

Los resultados del ajuste se resumen en la siguiente tabla similar a la anterior,

Además, se muestra el ajuste realizado para distinto número de nodos (10 y 20) y 8 grados de libertad. En las Figuras 4.5 y 4.6 se aprecia que la principal diferencia entres los ajuste se encuentra en el principio y el final de la curva, el ajuste de 20 nodos es capaz de seguir de mejor manera el comportamiento de los datos. Se resalta en ambos gráficos el ajustre realizado no se ve afectado por las observaciones atípicas en la zona de caída de los datos.

CUADRO 4.2: Tabla comparativa de ajustes usando distintos valores para los grados de libertad y para el número de nodos

ν	nodos	valor de $\hat{\lambda}$	Error Estándar	valor de $\hat{\phi}$	Error Estándar	tiempo [seg]
2	20 (13)	0.003	0.000	0.001	0.001	890.83
	28 (21)	0.006	0.000	0.001	0.002	1357.45
4	20 (13)	0.005	0.000	0.002	0.002	815.25
	28 (21)	0.010	0.000	0.002	0.003	1377.55

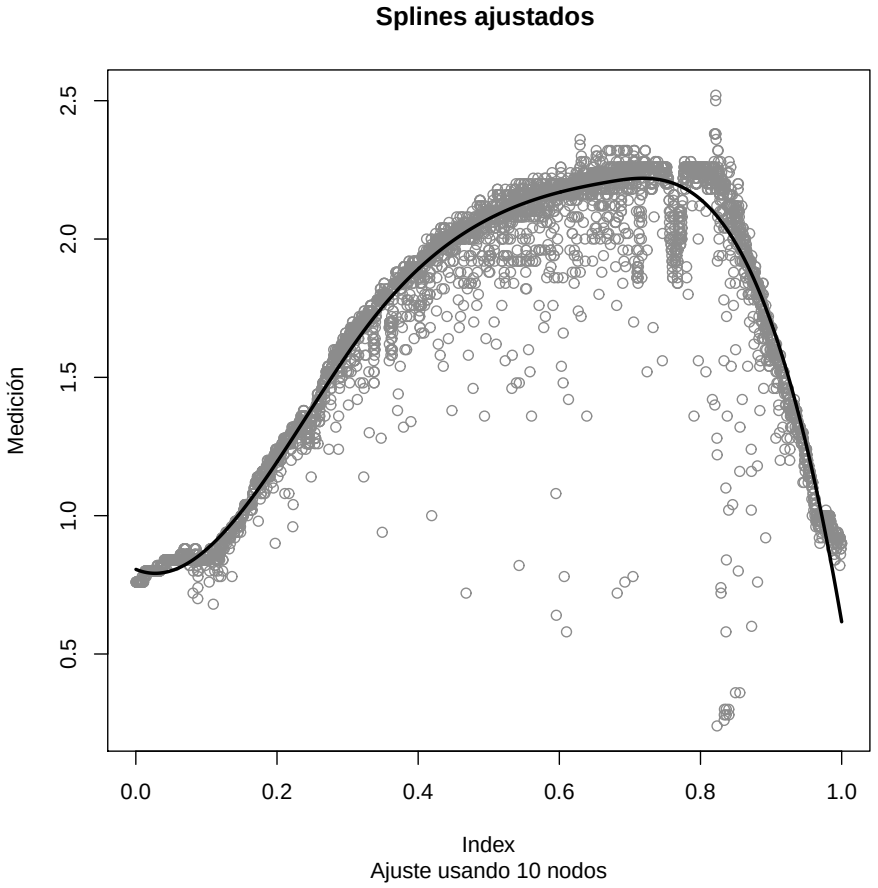


FIGURA 4.5: Ajuste de la base de datos balloon usando 8 grados de libertad y 10 nodos.

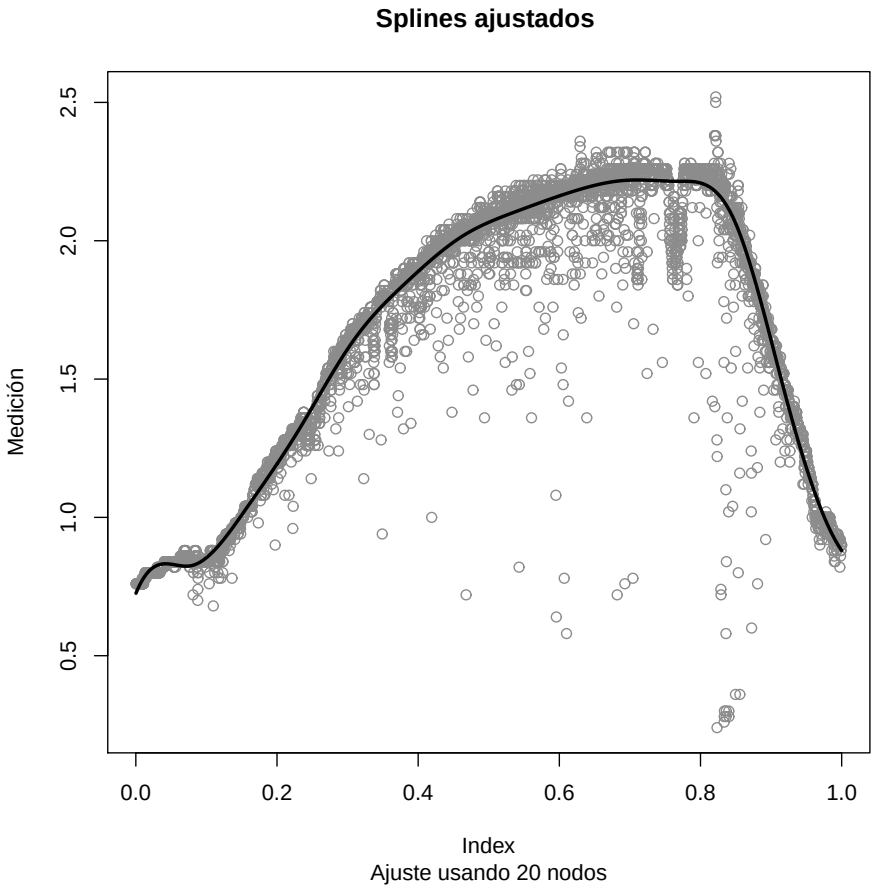


FIGURA 4.6: Ajuste de la base de datos balloon usando 8 grados de libertad y 20 nodos.

Capítulo 5

Conclusiones y Trabajos Futuros

Este trabajo se ha desarrollado una herramienta que estima el valor del parámetro de suavizamiento en la regresión P-spline a la par que se lleva a cabo el ajuste, lo cual es una diferencia sustancial con respecto a los criterios usuales que calculan el valor de λ, esta herramienta cuenta con buenas propiedades en términos computacionales y en términos la calidad de los resultados que entrega, los cuales cuentan con una estimación del error estándar, lo que eventualmente puede servir para generar intervalos de confianza de estos parámetros y también para trazar bandas de confianza para la curva ajustada. Las metodologías usadas en su elaboración le otorgan las características de robustez para la estimación y la flexibilidad necesaria para ajustar distintas bases de datos, por lo que los escenarios en donde puede ser utilizado son bastante variados.

Algunos detalles que pueden ser mejorados en trabajos futuros son, entre otras, la elección del vector $\boldsymbol{a}$, este vector depende del valor de λ de la forma: $\boldsymbol{a}_W = (\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda\boldsymbol{P})^{-1}\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{Y}$, por lo que para cada valor estimado de λ se tendrá un valor de $\boldsymbol{a}$ asociado. En este trabajo se ha optado por utilizar todos los valores calculados de $\boldsymbol{a}$ y estimar un $\boldsymbol{a}$ final mediante el método Monte Carlo, esta metodología entrega resultados que son aceptables en términos de tiempo y de valores finales pero tienen el problema de que las primeras estimaciones de este vector están calculadas con valores de λ muy distintos a los λ finales debido al proceso de estimación, por lo que se podría argumentar que estos valores iniciales de λ generan un ruido en la estimación de $\boldsymbol{a}$. Algunas de las opciones que se barajan para solucionar este problema son usar sólo las últimas estimaciones de λ para estimar $\boldsymbol{a}$ mediante Monte Carlo o, cuando ya se han estabilizado los valores λ se pueden generar algunos más con el fin de estimar el valor de $\boldsymbol{a}$ mediante Monte Carlo.

Con respecto a el código del algoritmo, este es costoso computacionalmente debido a las multiples estimaciones realizadas dentro de los ciclos internos del algoritmo, estimaciones que son de la forma de un `for` dentro de otro `for`. Queda pendiente para el código una escritura que sea más expedita y que aproveche de mejor manera las variables aleatorias generadas y los ciclos del algoritmo para reciclar algunos del los cálculos realizados. También se podría utilizar algún otro lenguaje de programación de bajo nivel para disminuir los tiempos de ejecución.

Apéndice A

Apéndice de Cálculos.

A.1. Cálculo de la función de log-verosimilitud del vector de datos aumentados

A continuación se muestran los cálculos realizados para la obtención del valor de la función $\ell_{\mathrm{a}}(\boldsymbol{\theta}; \boldsymbol{Y}_{\mathrm{aug}_2})$, la cual interviene en el algoritmo EM anidado directamente en la función de esperanzas condicionales $Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k+\frac{t}{T})}, \boldsymbol{\theta}^{(k)})$ descrita en la Sección 2.3. Para esto, primero se busca la función de log-verosimilitud del vector de datos aumentados $\boldsymbol{Y}_{\mathrm{aug}_2}$ para poder desarrollar el resto de los cálculos.

Se escribe la función de verosimilitud del vector de datos aumentados como la multiplicación de las funciones de verosimilitud de las variables aleatorias que lo componen:

$$L(\boldsymbol{\theta}; \boldsymbol{Y}_{\mathrm{aug}_2}) = L(\boldsymbol{\theta}; \boldsymbol{Y}^T, \boldsymbol{\tau}^T, \boldsymbol{a}^T) = \prod_{i=1}^{n} L(\boldsymbol{\theta}; Y_i) \cdot \prod_{i=1}^{p} L(\boldsymbol{\theta}; \tau_i) \cdot L(\boldsymbol{\theta}; \boldsymbol{a}), \qquad \text{(A.1.1)}$$

para los pasos que siguen se tiene en cuenta la siguiente consideración, como la etapa M del algoritmo EM se encarga de maximizar con respecto al vector $\boldsymbol{\theta} = (\phi, \lambda)^T$, se dejarán fuera de las ecuaciones los términos que no dependan de $\boldsymbol{\theta}$, por ejemplo, el término $\prod_{i=1}^{p} L(\boldsymbol{\theta}; \tau_i)$ en la ecuación (A.1.1), ya que la distribución de $\boldsymbol{\tau}$ no depende de $\boldsymbol{\theta}$ por lo que este se comporta como una constante al momento de maximizar. Este criterio se usa a lo largo de esta sección. Además, se utiliza la notación vectorial introducida en la Sección 2.1.

De esta forma la ecuación (A.1.1) se reescribe como:

$$L(\boldsymbol{\theta}; \boldsymbol{Y}_{\text{aug}_2}) = \prod_{i=1}^{n} L(\boldsymbol{\theta}; Y_i) \cdot L(\boldsymbol{\theta}; \boldsymbol{a}). \tag{A.1.2}$$

A continuación se presentan los cálculos por separado para cada función de verosimilitud de la ecuación anterior empezando por la función de verosimilitud de $\boldsymbol{Y}$:

$$\begin{aligned} L(\boldsymbol{\theta}; \boldsymbol{Y}) &= \prod_{i=1}^{n} L(\boldsymbol{\theta}; Y_i) \\ &= \prod_{i=1}^{n} \frac{1}{\left(\frac{\phi}{\tau_i}\right)^{1/2} \sqrt{2\pi}} \exp\left\{-\frac{1}{2}\frac{\tau_i}{\phi}(Y_i - \boldsymbol{b}_i^T \boldsymbol{a})^2\right\} \end{aligned} \tag{A.1.3}$$

Aplicando ln en ambos lados de la ecuación y despreciando los términos que no dependen de $\boldsymbol{\theta}$

$$\begin{aligned} \ell(\boldsymbol{\theta}; \boldsymbol{Y}) &= \sum_{i=1}^{n} \ln\left(\frac{1}{\sqrt{2\pi}}\right) - \frac{1}{2}\ln(\phi) + \frac{1}{2}\ln(\tau_i) + \left\{-\frac{1}{2}\frac{\tau_i}{\phi}(Y_i - \boldsymbol{b}_i^T \boldsymbol{a})^2\right\} \\ &= -\frac{n}{2}\ln(\phi) - \frac{1}{2\phi}(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a})^T \boldsymbol{W}(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a}), \end{aligned} \tag{A.1.4}$$

en donde $\boldsymbol{W} = \text{diag}(\tau_1, \ldots, \tau_n)$ es una matriz diagonal que contiene los componentes del vector $\boldsymbol{\tau}$.

De forma similar, a continuación se calcula la función de verosimilitud de $\boldsymbol{a}$:

$$L(\boldsymbol{\theta}; \boldsymbol{a}) \propto \frac{1}{(2\pi)^{p/2} \left|\frac{\phi}{\lambda}\boldsymbol{P}^-\right|^{1/2}} \exp\left\{-\frac{1}{2}\boldsymbol{a}^T \frac{\lambda}{\phi}\boldsymbol{P}\boldsymbol{a}\right\} \tag{A.1.5}$$

$$\tag{A.1.6}$$

Aplicando ln en ambos lados de la ecuación y despreciando los términos que no dependen de $\boldsymbol{\theta}$

$$\begin{aligned} \ell(\boldsymbol{\theta}; \boldsymbol{a}) &= -\frac{p}{2}\ln\left(\frac{\phi}{\lambda}\right) - \frac{1}{2}\boldsymbol{a}^T \left(\frac{\phi}{\lambda}\boldsymbol{P}^-\right)^{-1} \boldsymbol{a} \\ &= -\frac{p}{2}\ln(\phi) + \frac{p}{2}\ln(\lambda) - \frac{\lambda}{2\phi}\boldsymbol{a}^T \boldsymbol{P}\boldsymbol{a}. \end{aligned} \tag{A.1.7}$$

Usando los resultados anteriores se escribe de forma explícita la función de log-verosimilitud del vector de datos aumentados:

$$\begin{aligned}\ell_a(\theta; \boldsymbol{Y}_{\text{aug}_2}) &= \ell(\boldsymbol{\theta}; \boldsymbol{Y}) + \ell(\boldsymbol{\theta}; \boldsymbol{a}) \\ &= -\frac{n}{2}\ln(\phi) + \frac{1}{2\phi}(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a})^T \boldsymbol{W}(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a}) - \frac{p}{2}\ln(\phi) + \frac{p}{2}\ln(\lambda) - \frac{\lambda}{2\phi}\boldsymbol{a}^T\boldsymbol{P}\boldsymbol{a} \\ &= -\frac{n+p}{2}\ln(\phi) + \frac{p}{2}\ln(\lambda) + \frac{1}{2\phi}\left((\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a})^T \boldsymbol{W}(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a}) - \lambda\boldsymbol{a}^T\boldsymbol{P}\boldsymbol{a}\right).\end{aligned} \tag{A.1.8}$$

A.2. Cálculo de la función de esperanzas condicionales del algoritmo EM anidado

Usando la expresión de $\ell_{\text{a}}(\theta; \boldsymbol{Y}_{\text{aug}_2})$ descrita en la sección anterior, en la ecuación (A.1.8), se calcula la expresión para la función $Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k+\frac{t}{T})}, \boldsymbol{\theta}^{(k)})$ del Algoritmo EM anidado, la cual tiene la siguiente estructura.

$$Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k+\frac{t}{T})}, \boldsymbol{\theta}^{(k)}) = \text{E}\left[\text{E}\left[\ell_{\text{a}}(\boldsymbol{\theta}; \boldsymbol{Y}_{\text{aug}_2})|\boldsymbol{Y}_{\text{aug}_1}, \boldsymbol{\theta}^{(k+\frac{t}{T})}\right]|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\right], \tag{A.2.1}$$

para llevar esto a cabo, se desarrolla primero la esperanza condicional interna, es decir, el término $\text{E}\left[\ell_{\text{a}}(\boldsymbol{\theta}; \boldsymbol{Y}_{\text{aug}_2})|\boldsymbol{Y}_{\text{aug}_1}, \boldsymbol{\theta}^{(k+\frac{t}{T})}\right]$ para luego ser reemplazada en la esperanza condicional externa. Para simplificar la notación se llama $\boldsymbol{\theta}^{(k+\frac{t}{T})} = \boldsymbol{\theta}^*$. De esta forma se tiene:

$$\begin{aligned}\text{E}\left[\ell_{\text{a}}(\boldsymbol{\theta}; \boldsymbol{Y}_{\text{aug}_2})|\boldsymbol{Y}_{\text{aug}_1}, \boldsymbol{\theta}^*\right] &= -\frac{n+p}{2}\ln(\phi) + \frac{p}{2}\ln(\lambda) \\ &+ \frac{1}{2\phi}\text{E}\left[(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a})^T \boldsymbol{W}(\boldsymbol{Y} - \boldsymbol{B}\boldsymbol{a})|\boldsymbol{Y}, \boldsymbol{\tau}, \boldsymbol{\theta}\right] \\ &- \frac{\lambda}{2\phi}\text{E}\left[\boldsymbol{a}^T\boldsymbol{P}\boldsymbol{a}|\boldsymbol{Y}, \boldsymbol{\tau}, \boldsymbol{\theta}\right].\end{aligned} \tag{A.2.2}$$

Las esperanzas condicionales de esta última ecuación se calculan por separado para luego reemplazarlas en la ecuación. Para esto el trabajo de West [1984] entrega el siguiente resultado que permite realizar el cálculo de las esperanzas condicionales:

$$\boldsymbol{a}|\boldsymbol{Y}, \boldsymbol{\tau} \sim \mathcal{N}_p(\boldsymbol{a}_W(\lambda), \phi(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda\boldsymbol{P})^{-1}) \tag{A.2.3}$$

con $\boldsymbol{a}_W(\lambda) = (\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda\boldsymbol{P})^{-1}\boldsymbol{B}^T$.

Teniendo esto en cuenta y considerando que la esperanza de formas cuadráticas tiene la fórmula explícita $\mathrm{E}[\boldsymbol{X}^T\boldsymbol{A}\boldsymbol{X}] = \mathrm{tr}(\boldsymbol{A}\boldsymbol{\Sigma}) + \boldsymbol{\mu}^T\boldsymbol{A}\boldsymbol{\mu}$, en donde $\boldsymbol{\Sigma}$ es la matriz de covarianzas de $\boldsymbol{X}$ y $\boldsymbol{\mu}$ es su media, se pueden calcular de forma sencilla y explícita las esperanzas condicionales de la ecuación (A.2.2).

$$\mathrm{E}\left[\boldsymbol{a}^T\boldsymbol{P}\boldsymbol{a}|\boldsymbol{Y},\boldsymbol{\tau},\boldsymbol{\theta}^*\right] = \phi^*\mathrm{tr}\left(\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}+\lambda^*\boldsymbol{P})^{-1}\right) + \boldsymbol{a}_W(\lambda^*)^T\boldsymbol{P}\boldsymbol{a}_W(\lambda^*), \quad \text{(A.2.4)}$$

$$\begin{aligned}\mathrm{E}\left[(\boldsymbol{Y}-\boldsymbol{B}\boldsymbol{a})^T\boldsymbol{W}(\boldsymbol{Y}-\boldsymbol{B}\boldsymbol{a})|\boldsymbol{Y},\boldsymbol{\tau},\boldsymbol{\theta}^*\right] &= \phi^*\mathrm{tr}\left(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}+\lambda^*\boldsymbol{P})^{-1}\right)\\ &\quad + S_W(\boldsymbol{a}_W(\lambda^*)), \qquad \text{(A.2.5)}\end{aligned}$$

en donde $S_W(\boldsymbol{a}) = (\boldsymbol{Y}-\boldsymbol{B}\boldsymbol{a})^T\boldsymbol{W}(\boldsymbol{Y}-\boldsymbol{B}\boldsymbol{a})$.

Luego, se toman estos términos para reescribir la ecuación (A.2.2), la cual queda de la siguiente forma:

$$\begin{aligned}&\mathrm{E}\left[\ell_a(\boldsymbol{\theta},\boldsymbol{Y}_{\mathrm{aug}_2})|\boldsymbol{Y}_{\mathrm{aug}_1},\boldsymbol{\theta}^*\right] = -\frac{n+p}{2}\log\phi + \frac{p}{2}\log\lambda\\ &-\frac{\phi^*}{2\phi}\mathrm{tr}\left(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}+\lambda^*\boldsymbol{P})^{-1}\boldsymbol{B}\right) + \frac{1}{2\phi}(\boldsymbol{Y}-\boldsymbol{B}\boldsymbol{a}_W(\lambda^*))\boldsymbol{W}(\boldsymbol{Y}-\boldsymbol{B}\boldsymbol{a}_{\boldsymbol{W}}(\lambda^*))\\ &-\frac{\lambda\phi^*}{2\phi}\mathrm{tr}\left(\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}+\lambda^*\boldsymbol{P})^{-1}\right) - \frac{\lambda}{2\phi}(\boldsymbol{a}_W(\lambda^*)\boldsymbol{P}\boldsymbol{a}_W(\lambda^*)), \qquad \text{(A.2.6)}\end{aligned}$$

en esta última ecuación se considera lo siguiente:

$$\begin{aligned}&\mathrm{tr}\left(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}+\lambda^*\boldsymbol{P})^{-1}\right) + \mathrm{tr}\left(\lambda\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}+\lambda^*\boldsymbol{P})^{-1}\right)\\ &\qquad = \mathrm{tr}\left((\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}+\lambda\boldsymbol{P})(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}+\lambda^*\boldsymbol{P})^{-1}\right). \qquad \text{(A.2.7)}\end{aligned}$$

Por lo que la ecuación (A.2.2) queda finalmente escrita como:

$$\begin{aligned}\mathrm{E}\left[\ell_a(\boldsymbol{\theta},\boldsymbol{Y}_{\mathrm{aug}_2})|\boldsymbol{Y}_{\mathrm{aug}_1},\boldsymbol{\theta}^*\right] &= -\frac{n+p}{2}\log\phi + \frac{p}{2}\log\lambda\\ &\quad - \frac{1}{2\phi}\left(S_W(a_W(\lambda^*)) + \lambda\boldsymbol{a}_W^T(\lambda^*)\boldsymbol{P}\boldsymbol{a}_W(\lambda^*)\right)\\ &\quad + \frac{\phi^*}{2\phi}\mathrm{tr}\left((\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}+\lambda\boldsymbol{P})(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}+\lambda^*\boldsymbol{P})^{-1}\right).\end{aligned} \qquad \text{(A.2.8)}$$

Haciendo uso de la expresión para la esperanza interna de la ecuación anterior se procede a calcular la esperanza condicional externa de $Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k+\frac{t}{T})}, \boldsymbol{\theta}^{(k)})$.

$$\begin{aligned} Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k+\frac{t}{T})}, \boldsymbol{\theta}^{(k)}) &= -\frac{n+p}{2}\log\phi + \frac{p}{2}\log\lambda \\ &- \frac{1}{2\phi}\mathrm{E}\left[S_W(\boldsymbol{a}_W(\lambda^{(k+\frac{t}{T})})) + \lambda \boldsymbol{a}_W^T(\lambda^{(k+\frac{t}{T})})\boldsymbol{P}a_W(\lambda^{(k+\frac{t}{T})})|\mathbf{Y}, \boldsymbol{\theta}^k\right] \\ &+ \frac{\phi^{(k+\frac{t}{T})}}{2\phi}\cdot \mathrm{tr}\left(\mathrm{E}\left[(\mathbf{B}^T\mathbf{W}\mathbf{B} + \lambda\mathbf{P})(\mathbf{B}^T\mathbf{W}\mathbf{B} + \lambda^{(k+\frac{t}{T})}\boldsymbol{P})^{-1}|\mathbf{Y}, \boldsymbol{\theta}^{(k)}\right]\right). \end{aligned} \quad \text{(A.2.9)}$$

A.3. Cálculo de los valores óptimos del vector de parámetros

Teniendo la expresión de $Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k+\frac{t}{T})}, \boldsymbol{\theta}^{(k)})$ descrita en la sección anterior en la ecuación (A.2.9), se procede a buscar los valores de λ y ϕ que maximizan esta ecuación. Para llevar esto a cabo primero se fija el valor de $\lambda = \lambda^{(k+\frac{t}{T})}$ y se deriva con respecto a ϕ para buscar su valor óptimo.

$$\begin{aligned} \frac{\partial Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k+\frac{t}{T})}, \boldsymbol{\theta}^{(k)})}{\partial\phi} &= -\frac{n+p}{2\phi} + \frac{2}{\phi^2}\mathrm{E}\left[[S_W(\boldsymbol{a}_W(\lambda^{(k+\frac{t}{T})})) \right. \\ &\left. + \lambda^{(k+\frac{t}{T})}\boldsymbol{a}_W^T(\lambda^{(k+\frac{t}{T})})\boldsymbol{P}a_W(\lambda^{(k+\frac{t}{T})})]|\mathbf{Y}, \boldsymbol{\theta}^k\right] \\ &- \frac{\phi^{(k+\frac{t}{T})}}{2\phi^2}\cdot \mathrm{tr}\left(\mathrm{E}\left[(\mathbf{B}^T\mathbf{W}\mathbf{B}\right.\right. \\ &\left.\left. + \lambda^{(k+\frac{t}{T})}\mathbf{P})(\mathbf{B}^T\mathbf{W}\mathbf{B} + \lambda^{(k+\frac{t}{T})}\boldsymbol{P})^{-1}|\mathbf{Y}, \boldsymbol{\theta}^{(k)}\right]\right), \end{aligned} \quad \text{(A.3.1)}$$

al fijar el valor de λ el último término de la ecuación anterior se reduce de la siguiente forma:

$$\begin{aligned} \mathrm{tr}\left(\mathrm{E}\left[(\mathbf{B}^T\mathbf{W}\mathbf{B} + \lambda^{(k+\frac{t}{T})}\mathbf{P})(\mathbf{B}^T\mathbf{W}\mathbf{B} + \lambda^{(k+\frac{t}{T})}\boldsymbol{P})^{-1}|\mathbf{Y}, \boldsymbol{\theta}^{(k)}\right]\right) = \\ \mathrm{tr}\left(\mathrm{E}\left[I|\mathbf{Y}, \boldsymbol{\theta}^{(k)}\right]\right) = p, \end{aligned} \quad \text{(A.3.2)}$$

por lo que la derivada parcial de Q_{21} con respecto a ϕ queda escrita como

$$\begin{aligned} \frac{\partial Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k+\frac{t}{T})}, \boldsymbol{\theta}^{(k)})}{\partial\phi} &= -\frac{n+p}{2\phi} - \frac{p\cdot\phi^{(k+\frac{t}{T})}}{2\phi^2} + \frac{2}{\phi^2}\mathrm{E}\left[[S_W(\boldsymbol{a}_W(\lambda^{(k+\frac{t}{T})})) \right. \\ &\left. + \lambda^{(k+\frac{t}{T})}\boldsymbol{a}_W^T(\lambda^{k+\frac{t}{T}})\boldsymbol{P}a_W(\lambda^{(k+\frac{t}{T})})]|\mathbf{Y}, \boldsymbol{\theta}^k\right], \end{aligned} \quad \text{(A.3.3)}$$

Luego, se separa la suma de la esperanza condicional y se iguala la ecuación a cero para buscar el máximo:

$$0 = -\frac{n+p}{2\phi} + \frac{2}{\phi^2}\mathrm{E}\left[S_W(\boldsymbol{a}_W(\lambda^{(k+\frac{t}{T})}))|\mathbf{Y},\boldsymbol{\theta}^k\right] + \frac{2\lambda^{(k+\frac{t}{T})}}{\phi^2}\mathrm{E}\left[\boldsymbol{a}_W^T(\lambda^{k+\frac{t}{T}})\boldsymbol{P}a_W(\lambda^{(k+\frac{t}{T})})|\mathbf{Y},\boldsymbol{\theta}^k\right] - \frac{p\phi^{(k+\frac{t}{T})}}{2\phi^2}, \tag{A.3.4}$$

De esta última ecuación se obtiene al despejar que el valor óptimo de ϕ es:

$$\phi^{(k+\frac{t+1}{T})} = \frac{1}{n+p}\left[p\phi^{(k+\frac{t}{T})} + \mathrm{E}\left[S_W(\boldsymbol{a}_W(\lambda^{(k+\frac{t}{T})}))\middle|\boldsymbol{Y},\boldsymbol{\theta}^{(k)}\right] + \lambda^{(k+\frac{t+1}{T})}\mathrm{E}\left[\left\|\boldsymbol{D}\boldsymbol{a}_W(\lambda^{(k+\frac{t}{T})})\right\|^2\middle|\boldsymbol{Y},\boldsymbol{\theta}^{(k)}\right]\right]. \tag{A.3.5}$$

Se procede de forma similar para buscar el valor óptimo de λ pero utilizando el valor actualizado de $\phi^{(k+\frac{t+1}{T})}$ obtenido en la ecuación (A.3.5). Entonces, primero se deriva $Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k+\frac{t}{T})},\boldsymbol{\theta}^{(k)})$ con respecto a λ

$$\frac{\partial Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k+\frac{t}{T})},\boldsymbol{\theta}^{(k)})}{\partial\lambda} = \frac{p}{2\lambda} - \frac{1}{2\phi^{(k+\frac{t+1}{T})}}\mathrm{E}\left[\boldsymbol{a}_W^T(\lambda^{(k+\frac{t}{T})})\boldsymbol{P}a_W(\lambda^{(k+\frac{t}{T})})|\mathbf{Y},\boldsymbol{\theta}^k\right] + \frac{1}{2}\cdot\mathrm{tr}\left(\mathrm{E}\left[\boldsymbol{P}(\mathbf{B}^T\mathbf{W}\mathbf{B}+\lambda^{(k+\frac{t}{T})}\boldsymbol{P})^{-1}|\mathbf{Y},\boldsymbol{\theta}^{(k)}\right]\right). \tag{A.3.6}$$

Al igualar a 0 y despejar la ecuación anterior se obtiene que el valor óptimo de λ es:

$$\left(\lambda^{(k+\frac{t+1}{T})}\right)^{-1} = \frac{\mathrm{E}\left[\left\|\boldsymbol{D}\boldsymbol{a}_W(\lambda^{(k+\frac{t}{T})})\right\|^2\middle|\boldsymbol{Y},\boldsymbol{\theta}^{(k)}\right]}{p\phi^{(k+\frac{t+1}{T})}} + \frac{\phi^{(k+\frac{t+1}{T})}\cdot\mathrm{tr}\left(\boldsymbol{P}\mathrm{E}\left[\left(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}+\lambda^{(k+\frac{t+1}{T})}\boldsymbol{P}\right)^{-1}\middle|\boldsymbol{Y},\boldsymbol{\theta}^{(k)}\right]\right)}{p\phi^{(k+\frac{t+1}{T})}}. \tag{A.3.7}$$

Las esperanzas condicionales de las ecuaciones (A.3.5) y (A.3.7) no tienen una forma explícita, por lo que al momento de buscar sus valores numéricos estas serán estimadas mediante el método Monte Carlo.

A.4. Cálculo de las derivadas parciales de Q_2

Como se explica en la Sección 3.2, para tener una estimación del error estándar de las estimaciones se puede utilizar la ecuación propuesta por Oakes [1999] para obtener la matriz Hessiana asociada a los parámetros. Esta ecuación tiene la siguiente forma:

$$\frac{\partial^2 L(\boldsymbol{\theta}, \boldsymbol{Y})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} = \left\{ \frac{\partial^2 Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} + \frac{\partial^2 Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^{(k)}} \right\} \Bigg|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}}. \tag{A.4.1}$$

Para hacer uso de esta fórmula se debe usar la función $Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})$ descrita en la Sección 3.2, la cual representa a la función Q_{21} cuando esta es evaluada al finalizar los ciclos internos de la forma $Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)}) = Q_{21}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)}, \boldsymbol{\theta}^{(k)})$. Esta función tiene la siguiente expresión

$$\begin{aligned} Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)}) &= -\frac{n+p}{2}\log\phi + \frac{p}{2}\log\lambda \\ &- \frac{1}{2\phi}\mathrm{E}\left[S_W\left(\boldsymbol{a}_W(\lambda^{(k)})\right) + \lambda \boldsymbol{a}_W^T(\lambda^{(k)})\boldsymbol{P}\boldsymbol{a}_W(\lambda^{(k)})|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\right] \\ &+ \frac{\phi^{(k)}}{2\phi} \cdot \mathrm{tr}\left\{\mathrm{E}\left[(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda\boldsymbol{P})(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\right]\right\}. \end{aligned} \tag{A.4.2}$$

Los cálculos de las derivadas parciales de $Q_2(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})$ son bastante tediosos dada la cantidad de términos y variables que intervienen, por lo que en los cálculos siguientes se dan resultados explícitos para no entrar en cálculos engorrosos.

Se empieza por el primer término del lado derecho de la ecuación (A.4.1), es decir la matriz

$$\frac{\partial^2 Q_2}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} = \begin{pmatrix} \frac{\partial^2 Q_2}{\partial \phi^2} & \frac{\partial^2 Q_2}{\partial \phi \partial \lambda} \\ \frac{\partial^2 Q_2}{\partial \lambda \partial \phi} & \frac{\partial^2 Q_2}{\partial \lambda^2} \end{pmatrix}, \tag{A.4.3}$$

a continuación se muestras las primeras derivadas parciales necesarias para calcular la matriz anterior.

$$\begin{aligned} \frac{\partial Q_2}{\partial \phi} &= -\frac{n+p}{2\phi} + \frac{1}{2\phi^2}\mathrm{E}\left[S_W\left(\boldsymbol{a}_W(\lambda^{(k)})\right) + \lambda \boldsymbol{a}_W^T(\lambda^{(k)})\boldsymbol{P}\boldsymbol{a}_W(\lambda^{(k)})|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\right] \\ &- \frac{\phi^{(k)}}{2\phi^2} \cdot \mathrm{tr}\left\{\mathrm{E}\left[(\boldsymbol{B}^T\mathbf{W}\mathbf{B} + \lambda\boldsymbol{P})(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\right]\right\}. \end{aligned} \tag{A.4.4}$$

$$\frac{\partial Q_2}{\partial \lambda} = \frac{p}{2\lambda} + \frac{\phi^{(k)}}{2\phi} \cdot \mathrm{tr}\left\{ \mathrm{E}\left[\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1} | \boldsymbol{Y}, \boldsymbol{\theta}^{(k)} \right] \right\}. \tag{A.4.5}$$

Luego, los términos de las segundas derivadas cruzadas son:

$$\begin{aligned} \frac{\partial^2 Q_2}{\partial \phi^2} &= \frac{\partial}{\partial \phi}\left(\frac{\partial Q_2}{\partial \phi}\right) = \frac{n+p}{2\phi^2} \\ &- \frac{1}{\phi^3}\mathrm{E}\left[S_W\left(\boldsymbol{a}_W(\lambda^{(k)})\right) + \lambda \boldsymbol{a}_W^T(\lambda^{(k)})\boldsymbol{P} a_W(\lambda^{(k)}) | \boldsymbol{Y}, \boldsymbol{\theta}^{(k)} \right] \\ &+ \frac{\phi^{(k)}}{\phi^3} \cdot \mathrm{tr}\left\{ \mathrm{E}\left[(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda\boldsymbol{P})(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1} | \boldsymbol{Y}, \boldsymbol{\theta}^{(k)} \right] \right\} \end{aligned} \tag{A.4.6}$$

$$\begin{aligned} \frac{\partial^2 Q_2}{\partial \phi \partial \lambda} = \frac{\partial^2 Q_2}{\partial \lambda \partial \phi} &= \frac{\partial}{\partial \lambda}\left(\frac{\partial Q_2}{\partial \phi}\right) = \\ &- \frac{\phi^{(k)}}{2\phi^2} \cdot \mathrm{tr}\left\{ \mathrm{E}\left[\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1} | \boldsymbol{Y}, \boldsymbol{\theta}^{(k)} \right] \right\} \end{aligned} \tag{A.4.7}$$

$$\frac{\partial^2 Q_2}{\partial \lambda^2} = \frac{\partial}{\partial \lambda}\left(\frac{\partial Q_2}{\partial \lambda}\right) = -\frac{p}{2\lambda^2} \tag{A.4.8}$$

Ahora los cálculos del segundo término del lado derecho de la ecuación (A.4.1), es decir, la matriz

$$\frac{\partial^2 Q_2}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^{(k)}} = \begin{pmatrix} \frac{\partial^2 Q_2}{\partial \phi \partial \phi^{(k)}} & \frac{\partial^2 Q_2}{\partial \phi \partial \lambda^{(k)}} \\ \frac{\partial^2 Q_2}{\partial \lambda \partial \phi^{(k)}} & \frac{\partial^2 Q_2}{\partial \lambda \partial \lambda^{(k)}} \end{pmatrix} \tag{A.4.9}$$

Las primeras derivadas parciales son:

$$\frac{\partial Q_2}{\partial \phi^{(k)}} = \frac{1}{2\phi} \cdot \mathrm{tr}\left\{ \mathrm{E}\left[(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda\boldsymbol{P})(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1} | \boldsymbol{Y}, \boldsymbol{\theta}^{(k)} \right] \right\} \tag{A.4.10}$$

$$\frac{\partial Q_2}{\partial \lambda^{(k)}} = -\frac{1}{2\phi}\mathrm{E}\left[-2\boldsymbol{a}_W(\lambda^{(k)})\boldsymbol{P}\boldsymbol{a}_W(\lambda^{(k)})\right.$$
$$+ 2(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}\boldsymbol{P}\boldsymbol{a}_W(\lambda^{(k)})$$
$$\left. - 2\lambda\boldsymbol{a}_W(\lambda^{(k)}\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}\boldsymbol{P}\boldsymbol{a}_W(\lambda^{(k)})\middle|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\right]$$
$$- \frac{\phi^{(k)}}{2\phi}\mathrm{tr}\Big\{\mathrm{E}\Big[- (\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda\boldsymbol{P})(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}$$
$$+ \lambda^{(k)}\boldsymbol{P})^{-1}\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}\Big|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\Big]\}\Big) \quad \text{(A.4.11)}$$

Luego, los términos de las segundas derivadas cruzadas son:

$$\frac{\partial^2 Q_2}{\partial\phi\partial\phi^{(k)}} = \frac{\partial}{\partial\phi}\left(\frac{\partial Q_2}{\partial\phi^{(k)}}\right) =$$
$$- \frac{1}{2\phi^2}\cdot\mathrm{tr}\left\{\mathrm{E}\left[(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda\boldsymbol{P})(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\right]\right\} \quad \text{(A.4.12)}$$

$$\frac{\partial^2 Q_2}{\partial\lambda\partial\phi^{(k)}} = \frac{\partial}{\partial\lambda}\left(\frac{\partial Q_2}{\partial\phi^{(k)}}\right) = -\frac{1}{2\phi^2}\cdot\mathrm{tr}\left\{\mathrm{E}\left[\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\right]\right\} \quad \text{(A.4.13)}$$

$$\frac{\partial^2 Q_2}{\partial\phi\partial\lambda^{(k)}} = \frac{\partial}{\partial\phi}\left(\frac{\partial Q_2}{\partial\lambda^{(k)}}\right) = \frac{1}{2\phi^2}\mathrm{E}\left[-2\boldsymbol{a}_W(\lambda^{(k)})\boldsymbol{P}\boldsymbol{a}_W(\lambda^{(k)})\right.$$
$$+ 2(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}\boldsymbol{P}\boldsymbol{a}_W(\lambda^{(k)})$$
$$\left. - 2\lambda\boldsymbol{a}_W(\lambda^{(k)})\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}\boldsymbol{P}\boldsymbol{a}_W(\lambda^{(k)})\middle|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\right]$$
$$+ \frac{\phi^{(k)}}{2\phi^2}\mathrm{tr}\Big\{\mathrm{E}\Big[- (\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda\boldsymbol{P})(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}$$
$$+ \lambda^{(k)}\boldsymbol{P})\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})\Big|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\Big]\Big\} \quad \text{(A.4.14)}$$

$$\frac{\partial^2 Q_2}{\partial\lambda\partial\lambda^{(k)}} = \frac{\partial}{\partial\lambda}\left(\frac{\partial Q_2}{\partial\lambda^{(k)}} =\right)$$
$$- \frac{\phi^{(k)}}{2\phi}\mathrm{tr}\left\{\mathrm{E}\left[-\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}\middle|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\right]\right\}$$
$$\text{(A.4.15)}$$

Por último, queda evaluar todas las ecuaciones anteriores en $\boldsymbol{\theta} = \boldsymbol{\theta}^{(k)}$, empezando por $\frac{\partial^2 Q_2}{\partial \boldsymbol{\theta}^2}\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}}$

$$\frac{\partial^2 Q_2}{\partial \phi^2}\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}} = \frac{n+p}{2(\phi^{(k)})^2} - \frac{1}{(\phi^{(k)})^3}\mathrm{E}\Big[S_W\big(\boldsymbol{a}_W(\lambda^{(k)})\big) + \lambda \boldsymbol{a}_W^T(\lambda^{(k)})\boldsymbol{P}a_W(\lambda^{(k)})|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\Big] + \frac{n}{(\phi^{(k)})^2} \quad \text{(A.4.16)}$$

$$\frac{\partial^2 Q_2}{\partial \phi \partial \lambda}\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}} = -\frac{1}{2(\phi^{(k)})^2} \cdot \mathrm{tr}\left\{\mathrm{E}\left[\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\right]\right\} \quad \text{(A.4.17)}$$

$$\frac{\partial^2 Q_2}{\partial \lambda^2}\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}} = -\frac{p}{2(\lambda^{(k)})^2}. \quad \text{(A.4.18)}$$

Ahora los otros términos de $\frac{\partial^2 Q_2}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^{(k)}}\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}}$

$$\frac{\partial^2 Q_2}{\partial \phi \partial \phi^{(k)}}\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}} = -\frac{n}{2(\phi^{(k)})^2} \quad \text{(A.4.19)}$$

$$\frac{\partial^2 Q_2}{\partial \lambda \partial \phi^{(k)}}\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}} = -\frac{1}{2(\phi^{(k)})^2} \cdot \mathrm{tr}\left\{\mathrm{E}\left[\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\right]\right\} \quad \text{(A.4.20)}$$

$$\begin{aligned}\frac{\partial^2 Q_2}{\partial \phi \partial \lambda^{(k)}}\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}} &= \frac{1}{2(\phi^{(k)})^2}\mathrm{E}\big[-2\boldsymbol{a}_W(\lambda^{(k)})\boldsymbol{P}\boldsymbol{a}_W(\lambda^{(k)}) \\ &+ 2(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}\boldsymbol{P}\boldsymbol{a}_W(\lambda^{(k)}) \\ &- 2\lambda \boldsymbol{a}_W(\lambda^{(k)})\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}\boldsymbol{P}\boldsymbol{a}_W(\lambda^{(k)})\big|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\Big] \\ &+ \frac{1}{2(\phi^{(k)})^2}\mathrm{tr}\Big\{\mathrm{E}\Big[-(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} \\ &\qquad + \lambda^{(k)}\boldsymbol{P})\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})\big|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\Big]\Big\} \quad \text{(A.4.21)}\end{aligned}$$

$$\frac{\partial^2 Q_2}{\partial \lambda \partial \lambda^{(k)}}\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(k)}} = -\frac{1}{2}\mathrm{tr}\Big\{\mathrm{E}\Big[-\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}\boldsymbol{P}(\boldsymbol{B}^T\boldsymbol{W}\boldsymbol{B} + \lambda^{(k)}\boldsymbol{P})^{-1}\big|\boldsymbol{Y}, \boldsymbol{\theta}^{(k)}\Big]\Big\} \quad \text{(A.4.22)}$$

Apéndice B

Apéndice de Códigos

```
tpower <-
function(x, s, p) {
    # truncated p-th power function
    (x - s) ^ p * (x > s)
}

bbase <-
function(x, xmin = min(x), xmax = max(x), deg = deg, nknots=nknots) {
# construct a B-spline basis of degree 'deg'
    nseg=nknots-2*deg-1
    dx <- (xmax - xmin) / nseg
    knots <- seq(xmin - deg * dx, xmax + deg * dx, by = dx)
    P <- outer(x, knots, tpower, deg)
    n <- dim(P)[2]
    K <- diff(diag(n), diff = deg + 1) / (gamma(deg + 1) * dx ^ deg)
    B <- (-1) ^ (deg + 1) * P %*% t(K)
    list(B = B, knots = knots, K=K)
}

penalty <-
function(p, ord) {
    # construct a penalty matrix of order 'ord'
    K <- diff(diag(p), diff = ord)
    K
```

```
}


P_ajust <-
function(X, Y, deg=deg,  nknots=nknots, tolerancia=tolerancia, nu=nu) {
# Ajuste P-spline y estimación del parámetro de suavizamiento
n=length(X)
nseg=nknots-2*deg-1
p=nseg+3
fit<-bbase(X,deg=deg,nknots=nknots)

B<-fit$B; K<-penalty(p,3); P<-t(K)%*%K
lambda=0; phi=1;
lambda_vec<-c(lambda); phi_vec<-c(phi)
error_lambda<-c(0); error_phi<-c(0)

a=solve(t(B)%*%B + lambda*P)%*%t(B)%*%Y
D=((Y - B%*%a)^2)/phi
 W<- matrix(0, length(X), length(X))

T=7; it=1
ECM<-c()
ecm=sum((Y-B%*%a)^2)
est=B%*%a

v1=c(0);v2=c(0);v3=c(0);v4=c(0);v5=c(0);v6=c(0);v7=c(0)

ptm <- proc.time()
while(ecm > tolerancia){
    aMc=matrix(0,p,1)
    for(i in 1:T){

        if(it<5){N=5}
        if(5<= it && it <10){N=20}
        if(10<= it && it <20){N=100}
        if(20<= it){N=200}
```

```
        E1=0; E2=0; E3=matrix(0, p, p)
        a_est=matrix(0,p,1)

        E_1=0; E_2=0; E_3=0; E_4=0; E_5=0; E_6=0
            for(l in 1:N){
                for(j in 1:length(X)){W[j,j]=rgamma(1,(nu+1)/2,
                                                    (nu+D[j])/2)}
                    inv=solve( t(B)%*%W%*%B + lambda*P )
                    aw= inv %*% t(B) %*% W %*% Y
                    Sw=(t(Y-B%*%aw))%*%W%*%(Y-B%*%aw)

                    E1=E1+(Sw/N)
                    E2=E2+(t(K%*%aw)%*%K%*%aw)/N
                    E3=E3+solve(t(B)%*%W%*%B + lambda*P)/N
                    a_est=a_est+(aw/N)
                }
                phi=((1/(n+p))*( p*phi + E1[1] + lambda*E2[1] ))
                lambda=(p*phi)/(E2[1] + phi*sum(diag(P%*%E3)))
                aMc=aMc+a_est/T

                lambda_vec[it]=lambda
                phi_vec[it]=phi
}
        a=aMc
        D=((Y - B%*%a)^2)/phi
        ecm=sum((est-B%*%a)^2)
        ECM[it]<-c(ecm)
        est=B%*%a
        it=it+1
    }
    N=200
    for(j in 1:N){

        for(i in 1:length(X)){W[i,i]=rgamma(1,(nu+1)/2,(nu+D[i])/2)}
        inv=solve( t(B)%*%W%*%B + lambda*P )
        aw= inv %*% t(B) %*% W %*% Y
```

```
        Sw=(t(Y-B%*%aw))%*%W%*%(Y-B%*%aw)

        E_1 = E_1 + ( Sw[1]/N )
        E_2 = E_2 + ( t(aw)%*%P%*%aw )[1]/N
        E_3 = E_3 + inv/N
        E_4 = E_4 + (2*t(aw) %*% P %*% inv %*% (t(B) %*% W %*% (Y-B%*%aw)
                                              - lambda * P %*% aw ))[1]/N
        E_5 = E_5 + (2*t(aw)%*%P %*% inv%*%P %*% aw)[1]/N
        E_6 = E_6 + inv %*% P %*% inv/N
    }

    d2Q_dphi2 = as.numeric( (n+p)/(2*phi^2) - (1/(phi^3))*(E_
                                              1 + lambda*E_2))
    d2Q_dlambda_dphi =as.numeric( 1/(2*phi^2)*E_2 - 1/(2*phi)
                                          * sum(diag(P%*%E_3)))
    d2Q_dlambda2 = as.numeric( -p/(2*lambda^2))
    d2Q_dphi_dphik = as.numeric( -p/(2*phi^2))
    d2Q_dlambda_dphik =as.numeric(  1/(2*phi) * sum(diag( P%*%E_3 )) )
    d2Q_dphi_dlambdak =  as.numeric( 1/(2*phi^2) * E_4 + 1/(2*phi)
                                          * sum(diag( P%*%E_3  )) )
    d2Q_dlambda_dlambdak = as.numeric( 1/(2*phi)*( E_5 )
                                - 0.5*sum(diag(P%*%E_6)) )
    Q21=cbind(c( d2Q_dphi2 , d2Q_dlambda_dphi ),
                    c(d2Q_dlambda_dphi ,d2Q_dlambda2))
    Q22=cbind(c( d2Q_dphi_dphik , d2Q_dlambda_dphik),
                    c(d2Q_dphi_dlambdak ,d2Q_dlambda_dlambdak))

    error_lambda=sqrt(solve(-Q21-Q22)[1,1])
    error_phi=sqrt(solve(-Q21-Q22)[2,2])

    tiempo =proc.time() - ptm

    list(lambda=lambda, phi=phi,a=a, ECM=ECM, tiempo=tiempo,
            Q=solve(-Q21-Q22),error_lambda=error_lambda,
            error_phi=error_phi,N=N)
}
```

Bibliografía

Andrews, D. y Mallows, C. (1974). Scale mixtures of normal distributions. *Journal of the Royal Statistical Society*, 36:99–102.

Cantoni, E. y Ronchetti, E. (2001). Resistant selection of the smoothing parameter for smoothing splines. *Statistics and Computing*, 11:141–146.

Dempster, A., Laird, N., y Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorith. *Journal of the Royal Statistic Society, Series B.*, (39):1–38.

Durbán, M. (2009). An introduction to smoothing whit penalties: P-splines. *Boletín de Estadística e Investigación Operativa*, 25:195–205.

Eilers, P. y Marx, B. (1996). Flexible smoothing whit B-splines and penalties. *Statistical Science*, 11:89–121.

Eilers, P. y Marx, B. (2010). Splines, knots, and penalties. *WIREs Computational Statistics*, 2:637–653.

Hansen, C. (2010). *Discrete Inverse Problems: Insight and Algorithms*. Fundamentals of Algorithms. SIAM, Philadelphia.

Lee, W. y Pawitan, Y. (2014). Direct calculation of the variance of maximum penalized likelihood estimates via EM algorithm. *The American Statistician.*, 68:93–97.

Leinhardt, S. y Wasserman, S. S. (1979). Teaching regression: An exploratory approach. *The American Statistician*, 33:196–203.

McLachlan, G. y Krishnan, T. (2008). *The EM Algorithm and Extensions*. Wiley, New York. Wiley-Interscience; 2 edition.

Meza, C., Osorio, F., y De la Cruz, R. (2010). Estimation in nonlinear mixed-effects models using heavy-tailed distributions. *Statistics and Computing.*, 22:121–139.

Oakes, D. (1999). Direct calculation of the information matrix via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 61:479–482.

Osorio, F. (2016a). *heavy: Robust estimation using heavy-tailed distributions. R package version 0.38.* URL: CRAN.R-project.org/package=heavy.

Osorio, F. (2016b). Influence diagnostics for robust p-splines using scale mixture of normal distributions. *Annals of the Institute of Statistical Mathematics.*, 68:589–619.

O'Sullivan, F. (1986). A statistical perspective in ill-posed inverse problems. *Statistical Science*, 1:502–527.

O'Sullivan, F. (1988). Fast computation of fully automated log-density and log-hazard estimators. *SIAM Journal on Scientific and Statistical Computing*, 9:363 – 379.

Philippe, A. (1997). Simulation of right and left trucated gamma distributions by mixtures. *Statistics and Computing*, 7:173–181.

R Core Team (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Ruppert, D. (2002). Selecting the number of knots for penalized splines. *Journal of Computational and Graphical Statistics*, 11:738 – 757.

Ruppert, D., Wand, M. P., y Carroll, R. J. (2003). *Semiparametric Regression*. Cambridge University Press.

Shi, L. y Wang, X. (1999). Local influence in ridge regression. *Computational Statistics & Data Analysis.*, 31:341–353.

Thomas, W. (1991). Influence diagnostics for the cross-validated smoothing parameter in spline smoothing. *Journal of the American Statistical Association*, 86:693–698.

van Dyk, D. (2000). Nesting EM algorithms for computational efficiency. *Statistica Sinica*, 10:203–225.

West, M. (1984). Outliers models y prior distributions in bayesian linear regression. *Journal of the Royal Statistical Society, Series B*, 46:431–439.