



03061

2
2ey

**UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO**

**UNIDAD ACADEMICA DE LOS CICLOS
PROFESIONAL Y DE POSGRADO.**

U. A. C. P. Y P.

I. I. M. A. S.

**REGRESION BAYESIANA: ANALISIS Y
COMPARACION DE MODELOS LINEALES
GENERALIZADOS.**

T E S I S

QUE PARA OBTENER EL GRADO DE:

MAESTRO EN CIENCIAS

P R E S E N T A:

E L A C T U A R I O

GABRIEL NUÑEZ ANTONIO

MEXICO, D.F.

JULIO 1998

264148

**TESIS CON
FALLA DE ORIGEN**



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

A Dios.

Por todo cuanto soy.

A mis padres.

A mi madre que es el ser más maravilloso del mundo. Gracias por su apoyo moral, cariño y comprensión que desde pequeño me ha brindado, por guiar mi camino y estar siempre junto a mi en los momentos más difíciles.

A mi padre porque desde pequeño ha sido para mi un hombre grande y maravilloso al que siempre he admirado.

A quienes ni con todas las riquezas del mundo podré pagar sus desvelos y sacrificios.

A mis hermanos.

Tania, Alejandro, Mauricio, Leonel e Ivan. Porque siempre continuemos formando un gran equipo.

A Angélica.

Por su paciencia para convivir todo este tiempo conmigo.

Quisiera externar mis agradecimientos:

- A la Universidad Nacional Autónoma de México, en particular al Departamento de Estadística y Probabilidad del IIMAS y a la UACP y P, por todo el apoyo recibido durante el tiempo en que realice mis estudios de Posgrado.
- A mis profesores y compañeros de la Maestría, de quienes aprendí un poco de lo que algunos llaman *estadística*.
- A mis sinodales
 - Dr. Eduardo Gutiérrez-Peña.
 - Dr. Federico O'Reilly.
 - Dr. Manuel Mendoza.
 - Dr. Raúl Rueda.
 - M. en C. Alejandro Alegría.

Por sus valiosos y desinteresados comentarios en la realización de este trabajo.

- *De manera muy especial quiero agradecer a mi director de tesis, el Dr. Eduardo Gutiérrez-Peña, por su apoyo, tiempo, dedicación y entusiasmo en la dirección de esta tesis, y cuyo ejemplo como persona y profesionalista fueron siempre un estímulo para continuar.*

NO DESISTAS

*Cuando vayan mal las cosas,
como a veces suelen ir,
cuando ofrezca tu camino,
solo cuestas por subir,
cuando tengas poco haber,
pero mucho qué pagar y
precises un reír,
aún teniendo que llorar,
cuando ya el dolor te agobie y
no puedas más sufrir,
descansar acaso debes*

¡ pero nunca desistir !

*Tras las sombras de la duda,
ya plateadas, ya sombrías,
puede bien surgir el triunfo,
no el fracaso que temías,
y no es dable a tu ignorancia,
figurarte cuán cercano,
puede estar el bien que anhelas
y que juzgas tan lejano.
Lucha pues por más que tengas
en la brega que sufrir,
cuando todo esté peor,
más debemos insistir.*

Rudyard Kipling

PARABOLA DE LOS DOS MARES

*Hay dos mares en Palestina.
Uno es fresco y lleno de peces,
hermosas plantas adornan sus orillas;
los arboles extienden sus ramas sobre el
y alargan sus sedientas raíces para
beber sus saludables aguas y,
en sus playas los niños juegan.*

*El río Jordán hace este mar con burbujeantes aguas
de las colinas, que rien en el atardecer.
Los hombres construyen sus casas en la cercanía
y los pájaros, sus nidos y,
toda clase de vida es feliz por estar allí.*

*El río Jordán corre hacia al sur a otro mar.
Aquí no hay trazas de vida, ni murmullos de hojas,
ni canto de pájaros, ni risas de niños.
Los viajeros escogen otra ruta, solamente
por urgencia lo cruzan.
El aire es espeso sobre sus aguas y ningún hombre
ni bestias, ni aves la beben.*

Que hace esta diferencia entre mares vecinos?

*No es el río Jordán. El lleva la misma agua a los dos.
No es el suelo sobre el que están,
Ni el campo que los rodea,
La diferencia es esta:
El mar de Galilea recibe al río pero no lo retiene.
Por cada gota que a él llega, otra sale.*

*El dar y recibir son en igual manera.
El otro mar es un avaro... guarda su ingreso celosamente.
No tiene un generoso impulso. Cada gota que llega, allí queda.
El mar de Galilea da y vive.
El otro mar no da nada, le llaman
El Mar Muerto.*

Indice

Introducción	3
1 Preliminares	6
1.1 El Modelo de Regresión	6
1.1.1 Antecedentes	6
1.1.2 Regresión Lineal Normal	7
1.2 Comparación de Modelos	10
1.2.1 Comparación de Modelos como un Problema de Decisión.	10
1.2.2 Un enfoque predictivo semiparamétrico para la selección de modelos.	18
1.3 Aproximaciones Numéricas y Técnicas de Simulación	23
1.3.1 Aproximación Normal Asintótica	24
1.3.2 Aproximación de Laplace	26
1.3.3 Métodos de Monte Carlo	29
1.3.4 Monte Carlo vía Cadenas de Markov	31
2 Regresión Lineal Generalizada	38
2.1 Modelos Lineales Generalizados.	38
2.2 Modelo de Regresión Generalizado Semiparamétrico	40
2.2.1 Análisis del Modelo	43
2.2.2 Estimación de la Superficie de Regresión	47
2.3 Comparación de Modelos Lineales Generalizados	48

3	Aplicaciones	52
3.1	Ejemplo 1	52
3.1.1	Análisis del Modelo Logístico Cuadrático vía M-H.	54
3.1.2	El modelo Semi-paramétrico. Comparación de Modelos.	56
3.2	Ejemplo 2	62
3.2.1	Análisis del Modelo	62
3.2.2	El modelo Semiparamétrico	64
3.2.3	Comparación de Modelos	65
3.3	Ejemplo 3	67
4	Conclusiones	69
	Apéndice A	72
	Apéndice B	74
	Bibliografía	124

Introducción

De entre los modelos estadísticos, una clase importante, de gran diversidad, la constituyen los *modelos lineales generalizados*, que tienen como casos especiales a los modelos de regresión, de análisis de varianza, los modelos logit, probit, loglineales, modelos de respuesta multinomial y algunos modelos para datos de supervivencia, entre otros. La importancia de estos modelos se debe, en parte, a su gran aplicación en los diferentes campos del conocimiento como la medicina, la economía, el control de calidad, etc.

Las inferencias clásicas para modelos lineales generalizados se basan en la estimación de los parámetros y las propiedades distribucionales de los estimadores. Los modelos de la familia exponencial poseen ciertas propiedades de convexidad que en muchos casos garantiza la existencia y unicidad de los estimadores de máxima verosimilitud. Sin embargo, en general la maximización de la verosimilitud requiere de métodos numéricos. Por otro lado, los métodos de Monte Carlo vía Cadenas de Markov proporcionan una forma relativamente sencilla de hacer inferencias Bayesianas para una clase amplia de modelos, incluyendo los lineales generalizados.

Un problema de interés cuando se trabaja con modelos estadísticos es la selección de modelos. En la literatura Bayesiana existen técnicas basadas en los llamados factores de Bayes para tratar los problemas de selección de modelos y pruebas de hipótesis. Sin embargo, el problema de los factores de Bayes es que éstos dependen fuertemente de la especificación de las distribuciones iniciales de los parámetros. Concretamente, los factores de Bayes no están bien definidos si se utilizan distribuciones iniciales impropias. Otras propuestas que existen en la literatura consideran criterios predictivos para solucionar el problema

de Comparación de Modelos y atacan el problema considerando el llamado enfoque \mathcal{M} -cerrado, según el cual el modelo 'verdadero' está contenido en la colección de modelos a comparar. En Bernardo y Smith (1994) se propone una solución aproximada, que involucra lo que se conoce como *validación cruzada*, para aproximar la utilidad esperada considerando el enfoque \mathcal{M} -abierto. En este enfoque no se considera que alguno de los modelos a ser comparados sea el 'verdadero'. En la práctica, asumir el enfoque \mathcal{M} -cerrado es poco realista.

Las técnicas de regresión usuales proponen alguna forma paramétrica para el predictor lineal y proceden a analizar el modelo como si éste fuera el verdadero, sin tomar en cuenta la discrepancia entre la forma real y el predictor paramétrico considerado. En el presente trabajo se analiza un modelo de regresión generalizado semiparamétrico que toma en cuenta la discrepancia mencionada anteriormente y se propone un enfoque predictivo para la selección de modelos lineales generalizados desde un punto de vista Bayesiano.

El material de esta tesis se encuentra organizado de la siguiente manera. En el Capítulo 1 se presenta el modelo de regresión usual y se define un modelo alternativo (semiparamétrico) propuesto originalmente por Bligh y Ott (1975). También se describe el problema general de Comparación de Modelos, y se analiza como un problema de decisión. Además, se presenta el enfoque predictivo semiparamétrico para la selección de modelos.

Salvo en ciertos casos donde los procedimientos analíticos son posibles, la aplicación de técnicas Bayesianas para la solución de problemas involucra el uso de aproximaciones numéricas y/o técnicas de simulación. Al final del Capítulo 1 se discuten algunas de estas técnicas.

En el Capítulo 2 se describen los modelos lineales generalizados, y se define y analiza la extensión del modelo de regresión semiparamétrico presentado en el Capítulo 1. En este capítulo también se discute el problema de Predicción y el problema de Comparación de Modelos en el contexto de modelos lineales generalizados.

Como una forma de ilustrar los conceptos discutidos en los dos primeros capítulos, en el Capítulo 3 se presentan algunas aplicaciones.

Finalmente, en el Capítulo 4 se presentan las conclusiones de este trabajo y se esbozan algunas líneas de investigación.

Capítulo 1

Preliminares

1.1 El Modelo de Regresión

1.1.1 Antecedentes

En estadística aplicada las técnicas de regresión están entre algunas de las más ampliamente utilizadas. En términos generales, el problema puede plantearse de la siguiente manera: dada una variable de respuesta Y y un conjunto de covariables $X = (x_1, \dots, x_r)'$ se tiene interés, por un lado, en estimar la posible relación funcional entre Y y X , y por otro lado, en predecir valores de la variable de respuesta asociados a nuevos valores de las covariables.

Una forma de modelar la relación entre Y y X es representar el valor esperado de Y como

$$E(Y | X) = \mu(X)$$

donde $\mu(\cdot)$ es una función desconocida de las covariables. Sin embargo, en la práctica $\mu(\cdot)$ es usualmente aproximada por una función paramétrica simple, $\psi(\cdot; \beta)$, donde $\beta = (\beta_0, \dots, \beta_k)'$ denota un vector de parámetros desconocidos. Es común suponer que $\psi(\cdot; \beta)$ es lineal en β sobre alguna escala, es decir,

$$\psi(x; \beta) = G(\beta_0 + \beta_1 h_1(X) + \dots + \beta_k h_k(X)),$$

donde $G(\cdot)$ es una transformación conocida, uno a uno, y las $h_j(\cdot)$ son funciones

suaves conocidas, con $j = 1, \dots, k$.

1.1.2 Regresión Lineal Normal

Un caso particular importante es el modelo lineal normal, el cual se describe a continuación.

Supóngase que el conjunto de datos $\{(y_i, x_i), i = 1, \dots, n\}$ sigue el modelo

$$y_i = \eta(x_i) + \varepsilon_i, \quad (1.1)$$

donde $\eta(x)$ es una función desconocida o su forma funcional es extremadamente complicada. En la práctica es frecuente aproximar $\eta(x)$ a través de una forma funcional $p_k(x) = \beta_0 + \beta_1 h_1(x) + \dots + \beta_k h_k(x)$. Las técnicas de regresión usuales suponen que los $\{\varepsilon_i\}$ són variables aleatorias independientes, con $\varepsilon_i \sim N(0, \sigma^2)$ ($i = 1, \dots, n$), y analizan el modelo (1.1) considerando que $\eta(x)$ es igual a $p_k(x)$. Así, el modelo usual clásico ignora la discrepancia entre $\eta(\cdot)$ y $p_k(\cdot)$. En su lugar Blight & Ott (1975) proponen un modelo alternativo de la forma

$$y_i = p_k(x_i) + \delta_i + \varepsilon_i, \quad (1.2)$$

donde $\delta_i = \delta(x_i) = \eta(x_i) - p_k(x_i)$ es llamado el *error determinístico* del modelo. Los errores $\{\delta_i\}$ son determinísticos y difieren del componente de error aleatorio ε_i en el sentido de que $\delta_i = \delta_j$ si $x_i = x_j$, mientras que dos errores aleatorios para observaciones en el mismo punto de diseño son independientes. Hay que notar que dada la diferente naturaleza de los errores δ_i y ε_i es claro que no pueden ser absorbidos o considerados en un mismo componente de error.

El modelo (1.2) se puede reescribir como:

$$\mathbf{y} = \mathbf{H}\boldsymbol{\beta} + \boldsymbol{\delta} + \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim N(\mathbf{0}, \boldsymbol{\Sigma}) \quad (1.3)$$

con $\mathbf{y} = (y_1, \dots, y_n)'$, $\boldsymbol{\beta} = (\beta_0, \dots, \beta_k)'$, $\boldsymbol{\delta} = (\delta_1, \dots, \delta_n)'$, $\boldsymbol{\varepsilon} = (\varepsilon_1, \dots, \varepsilon_n)'$, \mathbf{H} la matriz con renglones $h_i' = (1, h_1(x_i), \dots, h_k(x_i))$ y $\boldsymbol{\Sigma} = \sigma^2 \mathbf{W}$. Aquí \mathbf{W} es una matriz diagonal de pesos con elemento j -ésimo m_j^{-1} . Donde los m_j^{-1} están

relacionados, por ejemplo, con los tamaños de muestra en el caso de repeticiones.

Para analizar el modelo (1.3) desde un punto de vista Bayesiano, es necesario especificar distribuciones iniciales sobre los parámetros desconocidos del modelo. En este caso, se supone que el error determinístico δ sigue una distribución normal multivariada con media cero y matriz de varianza-covarianza $\mathbf{R} = \rho^2 \mathbf{\Lambda}_\lambda$, donde $\mathbf{\Lambda}_\lambda$ es una matriz de $n \times n$ con elementos

$$\text{Cov}(\delta_i, \delta_j) = K_\lambda(x_i, x_j) = \lambda^{|x_i - x_j|^\alpha}.$$

En la mayoría de las aplicaciones $\alpha = 1$ ó $\alpha = 2$.

En forma más general, se asume que $\delta(\cdot)$ sigue un proceso Gaussiano con media cero y función de correlación $K_\lambda(\cdot, \cdot)$, para alguna constante λ ($0 < \lambda < 1$).

Considerando σ^2, ρ^2 y λ conocidos, la especificación de distribuciones iniciales queda completa suponiendo que β , δ y ϵ son independientes y que β se distribuye normal con media b_0 y matriz de varianza-covarianza B_0 . Este planteamiento es más general que el del modelo de regresión usual, ya que los estimadores correspondientes aparecen como un caso especial cuando ρ^2 tiende a cero.

Una vez que se ha definido completamente el modelo (1.2), y considerando que la distribución de \mathbf{y} es $N(A\gamma, \Sigma)$ donde $A = (\mathbf{H}, I)$ y $\gamma' = (\beta', \delta')$, se puede mostrar (Blight & Ott, 1975) que la distribución final conjunta de β y δ es normal, con

$$\begin{aligned} \mathbf{b} &= E(\beta|\mathbf{y}) = \mathbf{B} \left\{ \mathbf{H}'(\mathbf{R} + \Sigma)^{-1} \mathbf{y} + B_0^{-1} b_0 \right\}, \\ \mathbf{B} &= \text{Var}(\beta|\mathbf{y}) = \left\{ \mathbf{H}'(\mathbf{R} + \Sigma)^{-1} \mathbf{H} + B_0^{-1} \right\}^{-1}, \\ E(\delta|\mathbf{y}) &= \mathbf{R}(\mathbf{R} + \Sigma)^{-1} \{ \mathbf{y} - \mathbf{Hb} \}, \\ \text{Var}(\delta|\mathbf{y}) &= \mathbf{R}(\mathbf{R} + \Sigma)^{-1} \mathbf{H} \mathbf{B} \mathbf{H}' (\mathbf{R} + \Sigma)^{-1} \mathbf{R} + \mathbf{R}(\mathbf{R} + \Sigma)^{-1} \Sigma, \quad \mathbf{y} \\ \text{Cov}(\beta, \delta) &= -\mathbf{B} \mathbf{H}' (\mathbf{R} + \Sigma)^{-1} \mathbf{R}. \end{aligned} \tag{1.4}$$

Como se mencionó anteriormente, una aplicación importante de las técnicas de regresión es la predicción de la variable de respuesta Y asociada a un punto

x , el cual puede o no ser un punto de diseño. En otras palabras, se desea hacer inferencia sobre el valor de $\eta_x = \eta(x)$ en un punto dado x . Para resolver este problema se requiere la distribución condicional del nuevo η_x dado las respuestas observadas, es decir, se necesita la distribución final de η_x . De (1.2), η_x se puede representar como $\eta_x = p_k(x) + \delta_x$. Aquí $p_k(x)$ tiene una distribución inducida por la β ; en lo que respecta a δ_x se puede notar que si x es un punto de diseño entonces las ecuaciones (1.4) producen un estimador para δ_x . Si x no es un punto de diseño entonces se necesita relacionar δ_x con δ . De las propiedades de un proceso Gaussiano se sigue que $\delta_x \sim N(0, \rho^2)$ y $\text{Cov}(\delta_x, \delta_i) = \rho^2 K_\lambda(\delta_x, \delta_i)$ ($i = 1, \dots, n$). Considerando ahora las propiedades de normalidad del modelo, se sigue que la distribución condicional de δ_x dado δ es normal con

$$E(\delta_x|\delta) = \mathbf{c}'\mathbf{R}^{-1}\delta$$

y

$$\text{Var}(\delta_x|\delta) = \rho^2 - \mathbf{c}'\mathbf{R}^{-1}\mathbf{c},$$

donde $\mathbf{c} = \mathbf{c}(x) = \rho^2(K_\lambda(x, x_1), \dots, K_\lambda(x, x_n))'$.

Finalmente, se puede mostrar (Gutiérrez-Peña & Smith, 1998) que la distribución de η_x es normal con

$$E(\eta_x|\mathbf{y}) = \mathbf{h}'\mathbf{b} + \mathbf{c}'(\mathbf{R} + \mathbf{\Sigma})^{-1}(\mathbf{y} - \mathbf{H}\mathbf{b})$$

y

$$\text{Var}(\eta_x|\mathbf{y}) = \rho^2 - \mathbf{c}'\mathbf{R}^{-1}\mathbf{c} + (\mathbf{h} - \mathbf{H}'\mathbf{R}^{-1}\mathbf{c})'\mathbf{B}(\mathbf{h} - \mathbf{H}'\mathbf{R}^{-1}\mathbf{c}) + \mathbf{c}'(\mathbf{R} + \mathbf{\Sigma})^{-1}\mathbf{\Sigma}\mathbf{R}^{-1}\mathbf{c},$$

donde $\mathbf{h} = (1, h_1(x), \dots, h_k(x))'$.

Una vez obtenida la distribución final de η_x , se tiene resumida toda la información sobre $\eta(x)$ dado los datos observados y se está en condiciones de obtener un estimador $\hat{\eta}(x)$. Las propiedades de $\hat{\eta}(\cdot)$ dependerán fuertemente de la selección de la función de correlación $K_\lambda(\cdot, \cdot)$ y de los valores de ρ^2 y λ . En particular $\hat{\eta}(\cdot)$ será diferenciable sólo si $K_\lambda(\cdot, \cdot)$ lo es.

1.2 Comparación de Modelos

Sea $\{p_i(x|\theta_i), p_i(\theta_i); i \in I\}$ el conjunto de modelos que se desean comparar. Dada la especificación de las densidades iniciales, $p_i(\theta_i)$, las distribuciones predictivas para los modelos alternativos están dadas por

$$p_i(x) = p(x|M_i) = \int p_i(x|\theta_i) p_i(\theta_i) d\theta_i, \quad i \in I \quad (1.5)$$

Por conveniencia se denotarán los modelos alternativos por

$$M_i = \{p_i(x|\theta_i), p_i(\theta_i)\},$$

y al conjunto de estos modelos por $\mathcal{M} = \{M_i, i \in I\}$.

Dentro del contexto de selección o comparación de modelos, existen en la literatura varias formas de plantear el problema. En Bernardo & Smith (1994) se discuten los llamados enfoques \mathcal{M} -cerrado y \mathcal{M} -abierto, y en forma más reciente en Gutiérrez-Peña & Walker (1996) se introduce el enfoque \mathcal{M} -mezcla.

En la primera alternativa, la perspectiva \mathcal{M} -cerrada, se considera que uno de los modelos $\{M_i, i \in I\}$ es el verdadero, sin el conocimiento explícito de cual de ellos lo es.

Por otro lado, el enfoque \mathcal{M} -abierto considera que \mathcal{M} es simplemente un conjunto de modelos a comparar. Sin embargo, en este caso no se considera que alguno de los modelos a ser comparados es el verdadero, lo cual resulta más realista que la perspectiva \mathcal{M} -cerrada.

1.2.1 Comparación de Modelos como un Problema de Decisión.

El problema de comparación o selección de modelos, al igual que cualquier problema estadístico, se puede analizar como un problema de decisión. A continuación se discuten algunos problemas de decisión que involucran la selección o comparación de los modelos en $\mathcal{M} = \{M_i, i \in I\}$. (ver Bernardo & Smith, 1994, Cap. 6)

Un primer problema a considerar es aquel que sólo involucra la selección de

un M_i , sin ninguna acción subsecuente, por lo que las funciones de utilidad son de la forma $u(m_i, w)$, donde w es algún aspecto desconocido de interés. Si los datos observados sobre los cuales se basarán las decisiones se denotan por x , y la decisión de seleccionar un modelo M_i se denota por m_i , $i \in I$, entonces la estructura de decisión relacionada con este problema se puede ver esquemáticamente de la siguiente manera:

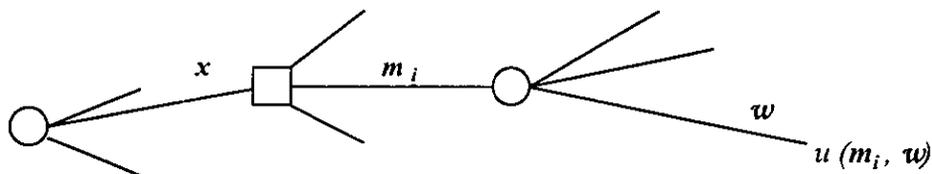


Figura 1.1. Problema de decisión: selección de un modelo.

Desde la perspectiva \mathcal{M} -cerrada, un ejemplo de un w de interés podría ser precisamente el modelo M_i para el cual, si se tuviera una muestra grande de observaciones futuras $y = (y_1, \dots, y_s)$, entonces $P(M_i|y) \rightarrow 1$ cuando $s \rightarrow \infty$. En este caso w etiqueta al "modelo verdadero".

Independientemente de las formas de w y $u(m_i, w)$, el problema de decisión planteado en la Figura 1.1, considerando el *criterio de Bayes*, se resuelve maximizando la utilidad esperada, lo que implica que el modelo 'óptimo' m^* satisface

$$\bar{u}(m^*|x) = \sup_{i \in I} \bar{u}(m_i|x)$$

con

$$\bar{u}(m_i|x) = \int u(m_i, w) p(w|x) dw, \quad i \in I,$$

donde $p(w|x)$ representa el conocimiento actual sobre w habiéndose observado x .

En el caso del enfoque \mathcal{M} -cerrado,

$$p(w|x) = \sum_{i \in I} p_i(w|x) P(M_i|x) \quad (1.6)$$

donde

$$P(M_i|x) = \frac{P(M_i) p(x|M_i)}{\sum_{j \in I} P(M_j) p(x|M_j)}$$

y $p_i(w|x) = p(w|M_i, x)$. Hay que notar que las cantidades $\{P(M_i|x), i \in I\}$ (las cuales juegan un papel importante dentro del contexto \mathcal{M} -cerrado) son las probabilidades finales, dado x , de que el modelo $M_i, i \in I$, sea el verdadero.

Desde la perspectiva \mathcal{M} -abierto, no se puede decir nada en general acerca de la forma explícita de $p(w|x)$. Sin embargo, es posible llevar a cabo un análisis aproximado y comparar los modelos en \mathcal{M} en base a sus utilidades esperadas sin tener realmente especificado un modelo alternativo "verdadero".

Otro problema de decisión que se puede considerar es aquel en el que primero se requiere tomar una decisión m_i , correspondiente a la selección del modelo M_i de \mathcal{M} y, posteriormente, tomar una decisión $d_j, j \in J_i$, relacionada con un evento desconocido w de interés. Por ejemplo, se puede estar interesado en predecir una observación futura, o estimar un parámetro común a todos los modelos en \mathcal{M} . Si w es el actual parámetro desconocido y $u(m_i, d_j, w)$ denota la utilidad que resulta de las selecciones sucesivas m_i , y $d_j, j \in J_i$, (la clase de decisiones posibles dado M_i), entonces este problema de decisión se puede ver esquemáticamente en la Figura 1.2.

Aplicando sistemáticamente el criterio de la utilidad esperada máxima, se tiene que la selección óptima de algún modelo es aquella m_x^* para la cual

$$\bar{u}(m_x^*|x) = \sup_{i \in I} \bar{u}(m_i|x)$$

donde

$$\bar{u}(m_i|x) = \int u(m_i, d_x^*, w) p(w|x) dw,$$

y d_x^* se obtiene de maximizar

$$\int u(m_i, d_j, w) p_i(w|x) dw.$$

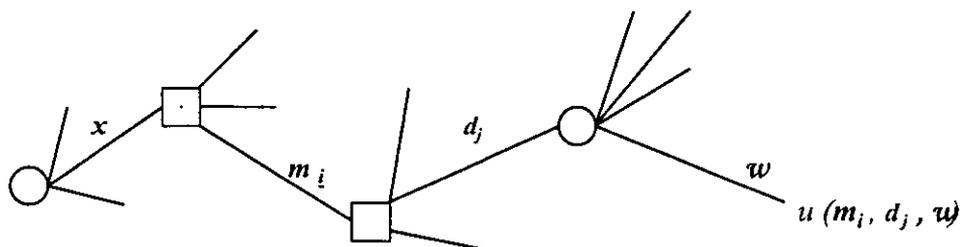


Figura 1.2. *Un problema de decisión que involucra la selección de un modelo y una decisión subsecuente.*

En este momento es importante reconocer que diversas selecciones tanto de w como de funciones de utilidad implica diferentes formas de resolver el problema de selección de modelos. A continuación se presentan algunos casos para ilustrar lo anterior.

Considerando la perspectiva \mathcal{M} -cerrada y como único problema de decisión el de seleccionar un modelo de \mathcal{M} (sin ninguna decisión subsecuente), el problema, hablando coloquialmente, es la *selección del verdadero modelo*.

En este caso, una función de utilidad que resulta natural puede ser de la forma cero-uno, es decir

$$u(m_i, w) = \begin{cases} 1 & \text{si } w = M_i \\ 0 & \text{si } w \neq M_i \end{cases}$$

Se puede ver, del análisis relacionado a la Figura 1.1, que

$$p_i(w|x) = \begin{cases} 1 & \text{si } w = M_i \\ 0 & \text{si } w \neq M_i \end{cases}$$

y

$$p(w|x) = \begin{cases} P(M_i|x), & \text{si } w = M_i \\ 0 & \text{si } w \neq M_i \end{cases}$$

Así, la utilidad esperada de la decisión m_i (selección de M_i), dado x , está dada por

$$\begin{aligned} \bar{u}(m_i|x) &= \int u(m_i, w) p(w|x) dw \\ &= P(M_i|x), \quad i \in I. \end{aligned}$$

Por lo tanto la decisión óptima se obtiene al, “seleccionar el modelo que tenga la probabilidad final mayor”.

Lo anterior sugiere que M_i y M_j pueden ser comparados usando los *cocientes de momios finales*

$$\frac{P(M_i|x)}{P(M_j|x)} = \frac{p(x|M_i)}{p(x|M_j)} \times \frac{P(M_i)}{P(M_j)}$$

donde

$$p(x|M_i) = \int p_i(x|\theta_i) p_i(\theta_i) d\theta_i.$$

Es decir, la relación que define el cociente de momios finales puede ser descrita de la siguiente manera: “*cociente de momios finales = cociente de verosimilitudes integradas × cociente de momios iniciales*”. Así, el *cociente de verosimilitudes integradas* proporciona el mecanismo por el cual los datos transforman el conocimiento inicial relativo en conocimiento final relativo, en el contexto de los modelos paramétricos.

La importancia fundamental de esta transformación justifica lo que en la literatura se conoce como *factor de Bayes* y que se define a continuación.

Definición 1. (Factor de Bayes). Sean H_i, H_j dos hipótesis que establecen los modelos alternativos M_i y M_j , respectivamente. Dados los datos x , el

factor de Bayes en favor de H_i (y en contra de H_j) está dado por

$$B_{ij}(x) = \frac{p(x|M_i)}{p(x|M_j)} = \left\{ \frac{P(M_i|x)}{P(M_j|x)} \right\} / \left\{ \frac{P(M_i)}{P(M_j)} \right\}$$

Así, si $B_{ij}(x) > 1$ significa que H_i es más plausible que H_j a la luz de los datos x .

En la literatura existen (ver, por ejemplo, Bernardo & Smith, 1994) técnicas basadas en factores de Bayes para tratar los problemas de selección de modelos y pruebas de hipótesis. Sin embargo, el problema de los factores de Bayes es que éstos dependen fuertemente de la especificación de las distribuciones iniciales de los parámetros de cada modelo. Concretamente, los factores de Bayes no están bien definidos si se utilizan distribuciones iniciales impropias. Berger & Perichi (1996) proponen un criterio llamado *factor de Bayes intrínseco*; por su parte O'Hagan (1995) utiliza los llamados *factores de Bayes fraccionales*. En ambos casos se pretende resolver algunas de las desventajas de los factores de Bayes. Cabe mencionar que estos dos métodos incorporan algunos elementos no deseables en el análisis.

Como se mencionó anteriormente, se pueden considerar diferentes selecciones de la función de utilidad para el problema de selección de modelos. Las llamadas funciones cero-uno dan el mismo peso a cualquier decisión errónea. Sin embargo, las consecuencias de la selección incorrecta de un modelo pueden ser menos serias si los modelos alternativos son "cercaños" en algún sentido, en cuyo caso las funciones de utilidad del tipo cero-uno pueden ser inapropiadas ya que no toman en cuenta esa "cercanía".

Si se considera el problema de comparación o selección de modelos para reportar inferencias sobre algún 'evento incierto' w , éste se puede ver como un problema de decisión, donde el espacio de decisiones consiste en la clase de distribuciones de probabilidad para w compatibles con los datos obtenidos. En este caso las funciones de utilidad apropiadas son las llamadas *funciones puntajes logarítmicas* (ver Sec. 2.7 y 3.4 de Bernardo & Smith, 1994). A continuación se presentan algunas definiciones relevantes.

Definición 2. (Función de Puntajes). Sea Ω el conjunto de posibles valores de w , D la clase de todas las distribuciones de probabilidad para $w \in \Omega$, compatibles con los datos x . Una función de puntajes para el problema de reportar inferencias definido por D, Ω , es una función de utilidad $u: D \times \Omega \rightarrow \mathbf{R}$. Se dice que una función de puntajes es suave si, para cada w , es continuamente diferenciable como una función de $q(w|x)$, $\forall d_q \in D$. Donde $q(w|x)$ es la función de densidad que un individuo reporta como base, para describir sus creencias sobre w condicional a los datos x .

Definición 3. (Función de puntajes Propia). Una función de puntajes u es propia si

$$\max_{d_q \in D} \int u(d_q, w) p(w|x) dw = \int u(d_p, w) p(w|x) dw,$$

donde el máximo, tomado sobre x , es alcanzado si y sólo si $d_q = d_p$, excepto tal vez sobre un conjunto de medida cero. Donde $d_p = p(\cdot|x)$ es la densidad que describe verdaderamente el comportamiento de w .

Es razonable pedir que una función de puntajes sea propia ya que esto asegura que un individuo coherente también sea honesto. El que la utilidad esperada se maximice si, y sólo si, $d_q = d_p$, garantiza que la mejor forma de describir las creencias sobre w , correspondientes a $d_q \in D$, es que el individuo describa los juicios que verdaderamente tiene sobre w .

Definición 4. (Función de puntajes Local). Una función de puntajes u es local si para cada $d_q \in D$ existen funciones $u_w, w \in \Omega$, definidas sobre \mathbf{R}^+ tales que

$$u(d_q, w) = u_w(q(w|x)).$$

Una función de puntajes local le da mayor peso a los juicios asociados con el 'verdadero estado de la naturaleza'. Así, si el verdadero resultado es w la función de utilidad sólo evaluará los juicios de un individuo en términos de lo que éste reporte acerca de w .

Proposición 1. (Caracterización de funciones de puntajes locales, propias y suaves). Si $u: D \times \Omega \rightarrow \mathbf{R}$ es una función de puntajes local, propia

y suave, para el problema de reportar inferencias, entonces debe ser de la forma

$$u(d_q, w) = A \log q(w|x) + B(w), \quad (1.7)$$

donde $A > 0$, es una constante arbitraria y $B(\cdot)$ una función arbitraria de w , sujeta a la existencia de $\bar{u}(d_q)$ para toda $d_q \in D$.

Una función de puntajes de la forma (1.7) es llamada una *función de puntajes logarítmica*.

El problema de reportar inferencias es un caso especial del problema representado por la Figura 1.2, donde, dados los datos x , m_i representa la selección del modelo M_i , la acción subsecuente d_j , $j \in J_i$ es algún reporte de los juicios sobre w , considerando M_i , y la función de utilidad está definida por

$$u(m_i, d_j, w) = u_i(q_i(\cdot|x), w),$$

para alguna función u_i . Aquí $q_i(\cdot|x)$ denota alguna distribución que describe las creencias sobre w correspondientes a d_j , $j \in J_i$.

Si $p_i(\cdot|x)$ es la distribución que realmente describe los juicios sobre w dado el modelo M_i y si u_i es una función de puntajes propia, entonces la d_j , $j \in J_i$, óptima debe de ser $d_j^* = p_i(\cdot|x)$ y

$$u(m_i, d_j^*, w) = u_i(p_i(\cdot|x), w), \quad i \in I.$$

Más aún, si la función de puntajes es local, se tiene la siguiente forma

$$u(m_i, d_j^*, w) = a \log p_i(w|x) + b(w), \quad i \in I,$$

para $a > 0$ y $b(w)$ arbitraria, de acuerdo con la Proposición 1.

La utilidad esperada de m_i es por tanto

$$\bar{u}(m_i, x) = \int \{a \log p_i(w|x) + b(w)\} p(w|x) dw,$$

y se prefiere el modelo para el cual esta utilidad esperada se maximice sobre

$i \in I$.

Considerando la perspectiva \mathcal{M} -cerrada, se tiene la siguiente forma explícita de $\bar{u}(m_i, x)$

$$\bar{u}(m_i, x) = u \sum_{j \in I} P(M_j | x) \int p_j(w | x) \log p_i(w | x) dw + \int b(w) p_j(w | x) dw.$$

Así, el modelo óptimo se obtiene al minimizar, sobre $i \in I$,

$$\sum_{j \in I} P(M_j | x) \int p_j(w | x) \log \left\{ \frac{p_j(w | x)}{p_i(w | x)} \right\} dw,$$

i.e., el promedio ponderado final, sobre los modelos, de la divergencia logarítmica (o discrepancia) entre $p_i(w | x)$ y cada $p_j(w | x)$, $j \neq i \in I$.

Desde la perspectiva \mathcal{M} -abierto el problema es comparar las utilidades esperadas

$$\int u(m_i, d_i^*, w) p(w | x), \quad i \in I,$$

con d_i^* la decisión óptima subsecuente dado M_i ; y donde $p(w | x)$ depende del modelo verdadero, el cual es desconocido.

Bernardo & Smith (1994) proponen una solución aproximada que involucra lo que se conoce como *validación cruzada* (o *muestreo predictivo de re-uso*) para el caso donde $w = y$, es decir, cuando se requiere predecir una observación futura.

1.2.2 Un enfoque predictivo semiparamétrico para la selección de modelos.

Sea $\mathcal{M} = \{M_1, \dots, M_k\}$ un conjunto de modelos de regresión paramétricos. El problema es seleccionar uno de los modelos en \mathcal{M} con fines predictivos. En este caso se tiene que

$$M_i = \{p_i(y | \beta_i), p_i(\beta_i)\} \quad y \in \mathbf{R}, \quad \beta_i \in \mathbf{R}^{q_i}.$$

donde

$$p_i(y|\beta_i) = N(y|\mathbf{h}_i(x)'\beta_i, \sigma_i^2), \quad x \in \mathbf{R}^r, \quad \sigma_i^2 \in \mathbf{R}^+$$

y

$$p_i(\beta_i) = N(\beta_i|b_{0i}, \Sigma_{0i}).$$

Para facilitar la exposición supondremos que $\{\sigma_i^2\}$ son conocidas. Es posible realizar un análisis similar para el caso de varianzas desconocidas, ver, por ejemplo, Gutiérrez-Peña (1997b).

Así, el modelo M_i se define por la verosimilitud, $p_i(y|\beta_i)$, y la correspondiente distribución inicial, $p_i(\beta_i)$, la cual no necesariamente es propia.

Denotemos por M_t al modelo verdadero y sea $(y_1, x_1), \dots, (y_n, x_n)$ una muestra de observaciones independientes de M_t . Finalmente, sea $\mathbf{y}' = (y_1, \dots, y_n)$. De acuerdo con lo visto en la Sección 1.2.1, si M_t fuera conocido, un criterio predictivo adecuado para la selección de modelos se podría basar en la utilidad (función de puntajes logarítmica) esperada final

$$\bar{u}_t(M_i) = \int \log p_i(y_*|\mathbf{y}) p_t(y_*|\mathbf{y}) dy_*,$$

donde $p_i(y_*|\mathbf{y})$ y $p_t(y_*|\mathbf{y})$ denotan la densidad predictiva final de una observación futura y_* , bajo el i -ésimo y el modelo verdadero, respectivamente. Maximizar $\bar{u}_t(M_i)$ es equivalente a minimizar la divergencia logarítmica de Kullback-Leibler de $p_i(y_*|\mathbf{y})$ con respecto a $p_t(y_*|\mathbf{y})$.

Se puede mostrar que, para cada $i = 1, \dots, k$,

$$p_i(y_*|\mathbf{y}) = N(y_*|\hat{f}_i(x), \hat{\sigma}_i^2(x))$$

donde

$$\hat{f}_i(x) = \mathbf{h}_i(x)'\mathbf{b}_{1i}$$

y

$$\hat{\sigma}_i^2(x) = \hat{\sigma}_i^2 + \mathbf{h}_i(x)'(H_i'\Sigma_i^{-1}H_i + \Sigma_{0i}^{-1})^{-1}\mathbf{h}_i(x)$$

con

$$\mathbf{b}_{1i} = (H_i' \Sigma_i^{-1} H_i + \Sigma_{0i}^{-1})^{-1} (H_i' \Sigma_i^{-1} \mathbf{y} + \Sigma_{0i}^{-1} \mathbf{b}_{0i}),$$

y

$$H_i = \begin{bmatrix} \mathbf{h}_i(x_1)' \\ \vdots \\ \mathbf{h}_i(x_n)' \end{bmatrix}, \quad \Sigma_i = \sigma_i^2 W,$$

donde W una matriz diagonal de pesos $\{1/m_i\}$.

Así, si se tienen los modelos M_i y M_j en \mathcal{M} , M_i será preferido a M_j si y sólo si

$$\bar{u}_t(M_i) > \bar{u}_t(M_j).$$

El criterio anterior considera que el modelo verdadero M_t es conocido y como consecuencia también $p_t(y_*|\mathbf{y})$, lo cual en general no es cierto. San Martini & Spezzaferri (1984) atacan este problema considerando el llamado enfoque \mathcal{M} -cerrado (es decir, suponen que $M_t \in \mathcal{M}$) y consideran una función de utilidad de puntajes logarítmica (ver sección 1.2.1). Ellos asignan una distribución de probabilidad sobre \mathcal{M} , $\{P(M_1), \dots, P(M_k)\}$, que describe las creencias iniciales sobre los distintos modelos en \mathcal{M} . Entonces estiman $p_t(y_*|\mathbf{y})$ a través de

$$\hat{p}_t(y_*|\mathbf{y}) = \sum_{i=1}^k p_i(y_*|\mathbf{y}) P(M_i|\mathbf{y}).$$

La justificación para hacer esto se presentó en la sección anterior (ver ecuaciones 1.5 y 1.6). En este caso

$$\begin{aligned} P(M_i|\mathbf{y}) &= P(M_i)p(\mathbf{y}|M_i) / \sum_{j=1}^k P(M_j)p(\mathbf{y}|M_j) \\ &\propto P(M_i) \int p_i(\mathbf{y}|\beta_i)p_i(\beta_i)d\beta_i. \end{aligned}$$

En la práctica, asumir el enfoque \mathcal{M} -cerrado es poco realista. Como se mencionó al final de la sección anterior, sólo en Bernardo & Smith (1994) se propone una técnica para aproximar la utilidad esperada considerando el enfoque \mathcal{M} -abierto. En esta tesis, se presenta un enfoque alternativo (semiparamétrico) para el problema de selección de modelos, considerando éste como un problema

de decisión.

Elementos del Problema de Decisión

Espacio de decisiones:

$$\mathbf{D} = \mathcal{M}$$

Espacio de 'sucesos inciertos':

$$\mathbf{F} = \{f \text{ es una función suave sobre } \mathbf{R}^T\}$$

Distribución inicial sobre F :

(a). Condicional en β

$$f(x) \sim \mathcal{N}(\mu_{\beta}^*(x), \Sigma^*(x, x)) \quad (\text{Proceso Gaussiano})$$

con

$$\begin{aligned}\mu_{\beta}^*(x) &= \mathbf{h}(x)' \beta \\ \Sigma^*(x, x) &= \rho^2 \Lambda(x, \tilde{x})\end{aligned}$$

es decir, dado β

$$f(x) = \mathbf{h}(x)' \beta + \delta(x), \quad \text{donde } \delta(x) \sim \mathcal{N}(0, \rho^2 \Lambda(x, \tilde{x})).$$

(b).

$$\beta \sim N_q(b_0, \Sigma_0)$$

Función de utilidad sobre $\mathbf{D} \times \mathbf{F}$:

$$u(M_i, f) = \int \log p_i(y_* | y) N(y_* | f(x_*)) dy_*$$

donde x_* es un valor en la región de interés. La elección de dicho valor será discutida en la sección 2.3.

La solución al problema de decisión es por lo tanto seleccionar el modelo que maximice la utilidad esperada final.

Con el enfoque anterior lo que se propone es la siguiente clase más general

de modelos para y .

Nivel I.

$$p(y|f(x)) = N(y|f(x), \sigma^2 W),$$

es decir, dado $f(x)$,

$$y = f(x) + \varepsilon, \quad \text{donde } \varepsilon \sim N(0, \sigma^2).$$

Nivel II.

$$p(f(x)|\beta) = \mathcal{N}(f(x) | \mathbf{h}(x)' \beta, \rho^2 \Lambda(x, \tilde{x}))$$

es decir, dado β ,

$$f(x) = \mathbf{h}(x)' \beta + \delta(x), \quad \text{donde } \delta(x) \sim \mathcal{N}(0, \rho^2 \Lambda(x, \tilde{x})).$$

Nivel III.

$$p(\beta) = N_q(\beta | b_0, \Sigma_0)$$

Dada la información $(y_1, x_1), \dots, (y_n, x_n)$, con $\mathbf{y}' = (y_1, \dots, y_n)$, se tiene que

$$p(f(x)|\beta, \mathbf{y}) \sim \mathcal{N}(f(x) | \mu_{\beta}^{**}(x), \Sigma^{**}(x, x))$$

con

$$\begin{aligned} \mu_{\beta}^{**}(x) &= \mathbf{h}(x)' \beta + \mathbf{s}(x)' (\Sigma^{**} + \Sigma)^{-1} (\mathbf{y} - H) \beta, \\ \Sigma^{**}(x, x) &= \Sigma^*(x, x) - \mathbf{s}(x)' (\Sigma^* + \Sigma)^{-1} \mathbf{s}(x) \end{aligned}$$

y

$$\mathbf{s}(x) = (\Lambda(x, x_1), \dots, \Lambda(x, x_n))'.$$

Por otra parte,

$$p(\beta | \mathbf{y}) = N_q(b_1, \Sigma_1)$$

donde

$$\begin{aligned} b_1 &= \left[H' (\Sigma^* + \Sigma)^{-1} H + \Sigma_0^{-1} \right]^{-1} \left[H' (\Sigma^* + \Sigma)^{-1} \mathbf{y} + \Sigma_0^{-1} \mathbf{b}_0 \right] \quad \text{y} \\ \Sigma_1 &= \left[H' (\Sigma^* + \Sigma)^{-1} H + \Sigma_0^{-1} \right]^{-1}, \quad \Sigma = \sigma^2 W. \end{aligned}$$

Se puede mostrar que

$$p(f(x)|\mathbf{y}) \sim \mathcal{N}(\mu_0^{**}(x), \Sigma_0^{**}(x, x))$$

donde

$$\begin{aligned} \mu_0^{**}(x) &= \mathbf{h}(x)' \beta + \mathbf{s}(x)' (\Sigma^* + \Sigma)^{-1} (\mathbf{y} - H) \beta & y \\ \Sigma_0^{**}(x, x) &= \Sigma^{**}(x, x) + \\ & \quad [h(x) - H'(\Sigma^* + \Sigma)^{-1} \mathbf{s}(x)]' \Sigma^* [h(x) - H'(\Sigma^* + \Sigma)^{-1} \mathbf{s}(x)]. \end{aligned}$$

De lo anterior se tiene que

$$\begin{aligned} \bar{u}(M_i) &= E_{f|\mathbf{y}} [u(M_i, f)] \\ &= \int \log p_i(y_* | \mathbf{y}) \{ \int N(y_* | f(x), \sigma^2) \mathcal{N}(f(x) | \mu_0^2(x), \Sigma_0^2(x, x)) df \} dy_* \\ &= \int \log p_i(y_* | \mathbf{y}) \hat{p}(y_* | \mathbf{y}) dy_* \end{aligned}$$

donde

$$\hat{p}(y_* | \mathbf{y}) = N(y | \hat{f}(x), \hat{\sigma}^2(x))$$

con

$$\hat{f}(x) = \mu_0^{**}(x)$$

y

$$\hat{\sigma}^2(x) = \Sigma + \Sigma_0^{**}(x, x).$$

Así, el modelo M_i será preferido al modelo M_j si y sólo si

$$\bar{u}(M_i) > \bar{u}(M_j).$$

1.3 Aproximaciones Numéricas y Técnicas de Simulación

Dada una verosimilitud $p(x|\theta)$ y una distribución inicial $p(\theta)$, con $\theta = (\theta_1, \theta_2, \dots, \theta_k)'$, se puede obtener, vía el teorema de Bayes, la distribución final $p(\theta|x)$ y a partir de ésta realizar inferencias sobre θ . Por ejemplo, se puede tener interés

en obtener las densidades marginales de algunos componentes de θ o bien, las densidades finales de alguna función de esos componentes, tales como cocientes o productos; o en general explorar y resumir distribuciones finales.

Salvo en ciertos casos donde los procedimientos analíticos son posibles, la aplicación de las técnicas Bayesianas para la solución de problemas involucra el uso de aproximaciones numéricas y/o técnicas de simulación.

La clave para la implementación de la solución, en la mayoría de los problemas es, por un lado, la habilidad para obtener o evaluar cierto número de integraciones y, por otro lado, la posibilidad de obtener muestras de alguna distribución final. Por lo que a continuación se discuten algunas técnicas de aproximación numérica y de simulación.

1.3.1 Aproximación Normal Asintótica

En estadística Bayesiana la distribución final $p(\theta|x)$, dada por

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{\int p(x|\theta)p(\theta)d\theta},$$

juega un papel fundamental, ya que a partir de ésta es posible hacer inferencias sobre θ . A continuación se presenta la aproximación normal asintótica de $p(\theta|x)$ para el caso univariado. El caso multivariado es totalmente análogo.

Sea $p_x(\theta) = p(x|\theta)p(\theta)$ y $\hat{\theta}$ la moda de $p_x(\theta)$. Desarrollando $\log p_x(\theta)$ en serie de Taylor alrededor de $\hat{\theta}$, se tiene

$$\log p_x(\theta) = \log p_x(\hat{\theta}) + \sum_{k=1}^{\infty} \frac{1}{k!} \frac{\partial^k \log p_x(\hat{\theta})}{\partial \theta^k} (\theta - \hat{\theta})^k.$$

Por otra parte, se puede notar que

$$\log p_x(\theta) = \log p(\theta) + \sum_{i=1}^n \log p(x_i|\theta).$$

Si denotamos

$$V(\hat{\theta}) = - \left\{ \frac{\partial^2 \log p_x(\hat{\theta})}{\partial \theta^2} \right\}^{-1}$$

y teniendo en cuenta que

$$\frac{\partial \log p_x(\hat{\theta})}{\partial \theta} = 0,$$

entonces

$$\log p_x(\theta) \approx \log p_x(\hat{\theta}) - \frac{1}{2V(\hat{\theta})}(\theta - \hat{\theta})^2.$$

Por lo tanto

$$p_x(\theta) \approx p_x(\hat{\theta}) \exp \left\{ -\frac{1}{2V(\hat{\theta})}(\theta - \hat{\theta})^2 \right\}$$

y

$$\int p_x(\theta) d\theta \approx p_x(\hat{\theta})(2\pi V(\hat{\theta}))^{1/2},$$

de donde

$$p(\theta|x) \approx N(\theta | \hat{\theta}, V(\hat{\theta}))$$

es la *aproximación normal asintótica* de $p(\theta|x)$.

En el caso en el que $\theta \in \mathbf{R}^d$ la aproximación resulta ser

$$p(\theta|x) \approx N_d(\theta | \hat{\theta}, \mathbf{V}(\hat{\theta}))$$

donde $\hat{\theta}$ es la moda de $p(\theta|x)$ y

$$\mathbf{V}(\hat{\theta}) = - \left\{ \frac{\partial^2 \log p_x(\hat{\theta})}{\partial \theta' \partial \theta} \right\}^{-1}.$$

Se puede mostrar (ver, por ejemplo, O'Hagan 1994, Capítulo 8) que

$$p(\theta|x) = N_d(\theta | \hat{\theta}, \mathbf{V}(\hat{\theta})) \left\{ 1 + O(n^{-1/2}) \right\}$$

para el caso univariado y multivariado, respectivamente.

En particular, se tiene

$$E(\theta|x) = \hat{\theta} \left\{ 1 + O(n^{-1}) \right\} \quad \text{y} \quad \text{Var}(\theta|x) = \mathbf{V}(\hat{\theta}|x) \left\{ 1 + O(n^{-1}) \right\}.$$

La calidad de esta aproximación dependerá del tamaño de muestra n y de la parametrización empleada.

1.3.2 Aproximación de Laplace

En muchos casos, el problema de calcular ciertas cantidades que se requieren para hacer inferencias desde el punto de vista Bayesiano, se reduce a evaluar integrales de la forma

$$S(g(\theta)) = \int g(\theta)p(\theta|x)d\theta,$$

donde $g(\theta)$ es alguna función de valor real, $x = (x_1, x_2, \dots, x_n)'$ es un conjunto de observaciones, y $p(\theta|x)$ es la distribución final de θ , dada por

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{\int p(x|\theta)p(\theta)d\theta}.$$

Si $g(\theta)$ es positiva (excepto en un conjunto de medida cero), $S(g(\theta))$ se puede reescribir de la siguiente manera

$$S(g(\theta)) = \frac{\int \exp \{-nh^*(\theta)\} d\theta}{\int \exp \{-nh(\theta)\} d\theta}$$

con

$$-nh^*(\theta) = \log g(\theta) + \log p(\theta) + \log p(x|\theta) \quad y$$

$$-nh(\theta) = \log p(\theta) + \log p(x|\theta).$$

Considerando el caso donde $\theta \in \mathbf{R}$, si se supone que $h(\cdot)$ y $h^*(\cdot)$ son funciones suaves, se pueden definir $\hat{\theta}, \theta^*$, y $\hat{\sigma}, \sigma^*$ de manera que

$$-h(\hat{\theta}) = \sup_{\theta} \{-h(\theta)\}, \quad \hat{\sigma} = [h''(\theta)]^{-1/2} \Big|_{\theta=\hat{\theta}}$$

$$-h^*(\theta^*) = \sup_{\theta} \{-h^*(\theta)\}, \quad \sigma^* = [h''(\theta)]^{-1/2} \Big|_{\theta=\theta^*}.$$

Entonces la *aproximación de Laplace* para cada una de las dos integrales que aparecen en el numerador y el denominador de $S(g(\theta))$ es respectivamente,

$$\sqrt{2\pi}\sigma^* n^{-1/2} \exp \{-nh^*(\theta^*)\}$$

y

$$\sqrt{2\pi\hat{\sigma}} n^{-1/2} \exp \left\{ -nh(\hat{\theta}) \right\}.$$

Esencialmente, la aproximación se basa en expansiones de Taylor de segundo orden para $h(\cdot)$ y $h^*(\cdot)$.

Tierney & Kadane (1986) muestran que la aproximación que resulta para $S(g(\theta))$ es de la forma

$$\hat{S}(g(\theta)) = \left(\frac{\sigma^*}{\hat{\sigma}} \right) \exp \left\{ -n \left[h^*(\theta^*) - h(\hat{\theta}) \right] \right\}$$

y que

$$S(g(\theta)) = \hat{S}(g(\theta)) \left[1 + O(n^{-2}) \right].$$

El argumento anterior explota el hecho de que los resúmenes inferenciales generalmente involucran un cociente de integrales. La aproximación de Laplace proporciona una técnica de aproximación general, potencialmente útil en el contexto Bayesiano. Para mayores detalles o extensiones de esta técnica, se pueden revisar los trabajos de Tierney, Kass & Kadane (1987, 1989a, 1989b) y Kass, Tierney & Kadane (1988, 1989a, 1989b).

En el caso donde $\theta \in \mathbf{R}^k$ la aproximación de Laplace para el denominador de $S(g(\theta))$ es

$$\int \exp \{ -nh(\theta) \} d\theta = (2\pi)^{k/2} \left| n \nabla^2 h(\hat{\theta}) \right|^{-1/2} \exp \{ -nh(\hat{\theta}) \}$$

donde $\hat{\theta}$ es tal que

$$-h(\hat{\theta}) = \sup_{\theta} \{ -h(\theta) \}$$

y

$$\left[\nabla^2 h(\hat{\theta}) \right]_{ij} = \left. \frac{\partial^2 h(\theta)}{\partial \theta_i \partial \theta_j} \right|_{\theta=\hat{\theta}}.$$

Una expresión totalmente análoga se puede obtener para el numerador de $S(g(\theta))$ en términos de $h^*(\cdot)$ y θ^* . En este caso la aproximación de Laplace

toma la forma

$$\hat{S}(g(\theta)) = \left(\frac{\nabla^2 h^*(\theta^*)}{\nabla^2 h(\hat{\theta})} \right) \exp \left\{ -n \left[h^*(\theta^*) - h(\hat{\theta}) \right] \right\}.$$

Si $\theta = (\phi, \omega)$ y se tiene interés en la densidad marginal de ω , se puede aplicar la aproximación de Laplace, resultando

$$\hat{p}(\omega|x) \propto \left| \nabla^2 h_\phi(\hat{\omega}_\phi) \right|^{-1/2} \exp \{ -n h_\phi(\hat{\omega}_\phi) \},$$

donde

$$-n h_\phi(\omega) = \log p(\phi, \omega) + \log p(x|\phi, \omega)$$

se considera como una función de ω para ϕ fijo, y

$$-h_\phi(\hat{\omega}_\phi) = \sup_{\omega} \{ -h_\phi(\omega) \}.$$

Si $g(\theta)$ no es positiva, es posible utilizar la aproximación de Laplace sobre $g(\theta) + k$, donde k es una constante apropiada. En forma alternativa se puede aproximar la función generadora de momentos.

Hay que notar que la aproximación de Laplace se deriva esencialmente al considerar aproximaciones normales a la integrales que aparecen en el numerador y en el denominador de $S(g(\theta))$. Si se tienen muestras pequeñas o si los componentes de θ están restringidos a otros rangos diferentes a la recta real, las aproximaciones de Taylor de segundo orden para las formas asociadas a los términos de los exponentes pueden resultar no muy adecuadas. En este caso la aproximación de Laplace se puede mejorar, considerando aproximaciones de las integrales por otras formas diferentes a la normal. Una alternativa, al menos para el caso donde $\theta \in \mathbf{R}$, es propuesta por Morris (1988), quien desarrolla una técnica de aproximación general basada en la familia de densidades de Pearson.

La aproximación de Laplace, propuesta por Tierney & Kadane (1986), no es invariante ante reparametrizaciones, por lo que se pueden considerar parametrizaciones alternativas. Algunas de estas parametrizaciones pueden mejorar sustancialmente las aproximaciones obtenidas; ver por ejemplo Achcar

& Smith (1989).

1.3.3 Métodos de Monte Carlo

La idea básica de los métodos de Monte Carlo consiste en escribir la integral que se requiere como el valor esperado de alguna función con respecto a alguna distribución de probabilidad.

Muestreo por Importancia

La aproximación del muestreo por importancia se basa en la idea de que si f es una función y g es una función de densidad de probabilidad, entonces

$$\begin{aligned} I &= \int f(\theta) d\theta = \int \left[\frac{f(\theta)}{g(\theta)} \right] g(\theta) d\theta \\ &= \int \left[\frac{f(\theta)}{g(\theta)} \right] dG(\theta) \\ &= E_G \left[\frac{f(\theta)}{g(\theta)} \right]. \end{aligned}$$

La distribución $G(\cdot)$ se conoce como la *distribución de muestreo por importancia* y generalmente se elige de manera que sea fácil de simular. De esta manera, si se obtiene una muestra $\theta_1, \theta_2, \dots, \theta_N$ de $g(\theta)$, entonces se puede aproximar la integral I a través del estimador insesgado

$$\hat{I} = \frac{1}{N} \sum_{i=1}^N \frac{f(\theta_i)}{g(\theta_i)}.$$

Hay que notar que la varianza de este estimador depende de la selección de $G(\cdot)$. En general, $g(\cdot)$ debe satisfacer los siguientes requisitos:

- a) Debe ser fácil de simular;
- b) Debe tener una forma similar a $f(\theta)$, la función que se desea integrar;
- c) Debe tener colas más pesadas que $f(\theta)$, de otra forma la varianza del estimador \hat{I} podría ser infinita.

Para una discusión e ilustraciones del uso de los métodos de Monte Carlo en la estadística Bayesiana, se pueden revisar los trabajos de Stewart (1979, 1983, 1985, 1987) y Stewart & Davis (1986).

Muestreo-Remuestreo por Importancia

En lugar de utilizar el método de muestreo por importancia para estimar integrales solamente, es posible aprovechar esta idea para producir muestras simuladas de distribuciones finales o predictivas. La técnica asociada, en este caso, es llamada por Rubin (1988) *muestreo-remuestreo por importancia*. A continuación se presenta una breve descripción de la técnica.

Supóngase que se tiene una muestra generada de una densidad $g(\theta)$, pero que lo que se desea es una muestra de la densidad

$$h(\theta) = \frac{f(\theta)}{\int f(\theta)d\theta},$$

donde sólo se conoce la forma funcional de $f(\theta)$. El problema consiste, entonces, en derivar una muestra de $h(\theta)$, dado $f(\theta)$ y una muestra de $g(\theta)$.

En los casos en los que exista una constante $M > 0$ tal que

$$f(\theta)/g(\theta) \leq M, \quad \text{para todo } \theta,$$

es posible utilizar el método de rechazo para generar variables aleatorias (ver, por ejemplo, Ripley 1987). El algoritmo es el siguiente:

- (i) Generar una observación θ de $g(\theta)$;
- (ii) Generar una variable $u \sim U(0, 1)$;
- (iii) Si $u \leq f(\theta)/Mg(\theta)$ aceptar θ ; en caso contrario, repetir (i)-(iii).

Cada valor aceptado de θ es entonces una observación de $h(\theta)$, y dado un tamaño de muestra N para la muestra original, el tamaño de muestra esperado para $h(\theta)$ es $M^{-1}N \int f(\theta)d\theta$.

En los casos en que no exista la cota M , o no se pueda encontrar fácilmente, es posible obtener muestras aproximadamente de $h(\theta)$ de la siguiente manera. Dada una muestra $\{\theta_1, \theta_2, \dots, \theta_N\}$ de $g(\theta)$, se puede calcular

$$q_i = \frac{w_i}{\sum_{i=1}^N w_i}, \quad \text{donde } w_i = \frac{f(\theta_i)}{g(\theta_i)}.$$

lo cual induce una distribución sobre $\{\theta_1, \theta_2, \dots, \theta_N\}$ con masa q_i sobre θ_i . Si pos-

teriormente se genera una observación θ^* de esta distribución discreta, entonces θ^* se distribuye aproximadamente como $h(\theta)$; ver Smith & Gelfand (1992).

1.3.4 Monte Carlo vía Cadenas de Markov

Dada una muestra es posible aproximar esencialmente cualquier característica de la distribución que la generó (e.g. densidades o momentos marginales). Sin embargo, en general no es posible generar muestras directamente de una función de distribución arbitraria y más cuando se tienen dimensiones altas.

Las técnicas numéricas revisadas hasta el momento (aproximación de Laplace y técnicas analíticas relacionadas así como distintas versiones de Monte Carlo) han contribuido a extender la aplicabilidad de los métodos de inferencia Bayesianos. No obstante, todos sufren de ciertas limitaciones en cuanto a su alcance e implementación. Ver, por ejemplo, Smith (1991).

Los métodos de Monte Carlo vía Cadenas de Markov (MCCM) solucionan los problemas anteriores a través de una aproximación indirecta en la que se requiere simular cadenas de Markov. Por lo anterior, antes de describir los métodos relacionados con MCCM, se presentan algunos conceptos relevantes.

·Un *proceso estocástico* es una familia de variables aleatorias X_t , donde t es un índice que corre sobre un conjunto apropiado T .

·Los procesos estocásticos son clasificados de acuerdo a su *espacio de estados* o rango de posibles valores que pueden tomar las variables aleatorias X_t , a su conjunto de índices T , y la relación de dependencia entre las variables X_t .

·Un proceso estocástico es *discreto*, si su espacio de estados es un conjunto finito o contable.

·Sea X_0, X_1, X_2, \dots , una serie de variables aleatorias, denotada por $\{X_t\}$, que puede tomar valores en el conjunto $\{S_1, \dots, S_k\}$. Se dice que $\{X_t\}$ es una *cadena de Markov discreta*, si cumple con la siguiente propiedad

$$P[X_{t+1} = S_j \mid X_1 = S_{j_1}, X_2 = S_{j_2}, \dots, X_t = S_i] = P[X_{t+1} = S_j \mid X_t = S_i]$$

para todo $t = 0, 1, 2, \dots$ y para todo $i, j = 1, 2, \dots, k$

·A la probabilidad $P[X_{t+1} = S_j \mid X_t = S_i] = P_{ij}^{t,t+1}$, se le llama *probabilidad*

de transición a un paso.

Cuando $P_{ij}^{t,t+1} = P_{ij}$, es decir, las probabilidades de transición de un paso son independientes de la variable t , se dice que la cadena de Markov tiene *probabilidades de transición estacionarias*.

Dada la distribución de probabilidad de X_0 , $\{p_j^{(0)}\}$, con

$$p_j^{(0)} = P\{X_0 = S_j\} \quad \forall j = 0, 1, 2, \dots, k.$$

la cadena de Markov queda completamente determinada por la *matriz de probabilidades de transición*

$$\mathbf{P} = [P_{ij}].$$

Una cadena de Markov es *homogénea* si

$$P\{X_{t+m} = S_j \mid X_t = S_i\} = P\{X_m = S_j \mid X_0 = S_i\} \quad \forall m = 1, 2, \dots$$

Sea $P_{ij}^{(m)} = P\{X_m = S_j \mid X_0 = S_i\}$ la probabilidad de transición a m pasos, entonces

$$P_{ij}^{(m)} = \sum_{t=1}^k P_{it} P_{tj}^{(m-1)} \quad (i, j = 1, 2, \dots, k)$$

donde

$$P_{ij}^{(0)} = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{si } i \neq j \end{cases}$$

Lo anterior implica, por un lado, que $\mathbf{P}^{(m)} = \mathbf{P} \times \mathbf{P}^{(m-1)}$ y, por otro lado, que las probabilidades de transición $P_{ij}^{(m)}$ (que corresponden a las entradas de la matriz $\mathbf{P}^{(m)}$) se pueden obtener de la siguiente manera

$$\mathbf{P}^m = \underbrace{\mathbf{P} \times \mathbf{P} \times \dots \times \mathbf{P}}_{m \text{ factores}} = [P_{ij}^{(m)}].$$

Una cadena de Markov es *irreducible* si todos los estados están comunicados entre sí, es decir, si $P_{ij}^{(m)} > 0$ para algún $m = 1, 2, \dots$ y $\forall i, j = 1, 2, \dots, k$.

El periodo de un estado S_i , denotado por $d(i)$, es el divisor común más

grande de todos los enteros $m \geq 1$ para los cuales $P_{ii}^{(m)} > 0$. Si $P_{ii} > 0$ para algún estado S_i , entonces el estado tiene periodo 1.

Una cadena de Markov en la que cada estado tiene periodo 1 es llamada *aperiódica*.

Toda matriz de probabilidad de transición \mathbf{P} sobre los estados $1, \dots, k$ que satisface las dos condiciones siguientes es *regular*:

(a) Para cada par de estados i, j existe una ruta k_1, \dots, k_q para la cual $P_{i k_1}, P_{k_1 k_2}, \dots, P_{k_q j} > 0$.

(b) Hay al menos un estado i para el cual $P_{ii} > 0$.

Si una matriz de probabilidad de transición \mathbf{P} sobre los k estados es *regular*, entonces \mathbf{P}^{k^2} no tendrá ningún elemento cero. Inversamente, si \mathbf{P}^{k^2} no es estrictamente positiva, entonces la cadena de Markov no es regular.

Sea \mathbf{P} una matriz de transición regular y suponga que

$$\lim_{m \rightarrow \infty} P_{ij}^{(m)} = \pi_j.$$

A $\pi = (\pi_1, \dots, \pi_k)$ se le conoce como la *Distribución Límite* del proceso o *Distribución de Equilibrio*.

Teorema. Sea \mathbf{P} una matriz de probabilidad de transición regular, homogénea sobre los estados $1, \dots, k$. Entonces la distribución de equilibrio $\pi = (\pi_1, \dots, \pi_k)$ es la única solución de las ecuaciones

$$\begin{aligned} \pi_j &= \sum_{k=1}^N \pi_k P_{k j}, & j &= 1, \dots, k \\ \sum_{i=1}^k \pi_i &= 1 \end{aligned} \tag{1.8}$$

(ver Howard & Karlin, 1984).

Cabe señalar que si una cadena es irreducible y aperiódica entonces también se garantiza la existencia de la distribución de equilibrio (Smith & Roberts, 1993). De hecho si $\pi = (\pi_1, \dots, \pi_k)$ es la distribución de equilibrio de una cadena de Markov irreducible, aperiódica entonces las cantidades $\pi_i, i = 1, \dots, k$ son las únicas soluciones a las ecuaciones (1.8). Ver, por ejemplo, Ross (1990).

Una interpretación de la distribución de equilibrio $\pi = (\pi_1, \dots, \pi_k)$ es la siguiente: después de que el proceso ha operado durante un periodo de tiempo

suficientemente largo, la probabilidad de encontrar el proceso en el estado j es π_j , sin importar el estado de inicio.

En este momento se está en condiciones de revisar los métodos de MCCM. Las ideas básicas se presentan a continuación.

Suponga que se desea generar una muestra de una distribución $\pi(x)$ para $x \in X \subseteq \mathbf{R}^k$, pero que esto no se puede hacer directamente. Sin embargo, suponga que se puede construir una cadena de Markov con espacio de estados X , matriz de probabilidad de transición \mathbf{P} que sea fácil de simular, y cuya función de equilibrio sea $\pi(x)$. Si se corre la cadena, es decir, si se simulan observaciones x_1, x_2, \dots , por un tiempo suficientemente largo; entonces la observación x_n , con n suficientemente grande, tiene una distribución aproximadamente $\pi(x)$.

Una vez que se tiene una cadena de Markov con distribución de equilibrio $\pi(x)$, ésta se puede usar para simular muestras aleatorias de $\pi(x)$ o estimar el valor esperado de alguna función $f(x)$ con respecto a $\pi(x)$. A continuación se muestran dos resultados relevantes.

Si $x_1, x_2, \dots, x_t, \dots$, es una realización de una cadena de Markov homogénea, irreducible y aperiódica, con espacio de estados X y distribución de equilibrio $\pi(x)$ entonces:

- (1) $x_t \xrightarrow{t \rightarrow \infty} x$ donde $x \sim \pi(x)$;
- (2) $\frac{1}{t} \sum_{k=1}^t f(x_k) \xrightarrow{t \rightarrow \infty} E_{\pi} \{f(x_i)\}$ casi seguramente.

La implementación de la técnica anterior sólo necesita algoritmos para construir cadenas que tengan distribuciones de equilibrio específicas. Claramente, diferentes formas de la cadena de Markov dan lugar a diferentes métodos de simulación. A continuación se presentan algunos de estos métodos, los cuales han resultado ser particularmente convenientes en la estadística Bayesiana.

Muestreo de Gibbs

Sea $\pi(x) = \pi(x_1, \dots, x_k)$, $x \in \mathbf{R}^k$, una densidad conjunta, y sea $\pi(x_i | x_{-i})$ la densidad condicional completa para cada componente x_i , dado los valores de las otras componentes $x_{-i} = (x_j, j \neq i)$, $i = 1, \dots, k$.

El algoritmo llamado *muestreo de Gibbs* (Geman & Geman, 1984) permite

simular una cadena de Markov con distribución de equilibrio $\pi(x)$, de la siguiente manera.

Dado un valor inicial $x^0 = (x_1^0, \dots, x_k^0)$, generar observaciones aleatorias

$$\begin{aligned} x_1^1 &\text{ de } \pi(x_1|x_{-1}^0); \\ x_2^1 &\text{ de } \pi(x_2|x_1^1, x_3^0, \dots, x_k^0); \\ x_3^1 &\text{ de } \pi(x_3|x_1^1, x_2^1, x_4^0, \dots, x_k^0); \\ &\vdots \\ x_k^1 &\text{ de } \pi(x_k|x_{-k}^1). \end{aligned}$$

El procedimiento anterior completa una transición de $x^0 = (x_1^0, \dots, x_k^0)$ a $x^1 = (x_1^1, \dots, x_k^1)$. Iteraciones sucesivas de este ciclo producen la serie $x^0, x^1, \dots, x^t, \dots$, la cual es una realización de una cadena de Markov con probabilidades de transición

$$P[x^{t+1}|x^t] = \prod_{r=1}^k \pi(x_r^{t+1}|x_1^{t+1}, \dots, x_{r-1}^{t+1}, x_{r+1}^t, \dots, x_k^t)$$

La ventaja de este algoritmo es que sólo se necesita muestrear de las distribuciones condicionales completas $\pi(x_i|x_{-i})$, las cuales típicamente pueden identificarse fácilmente al inspeccionar la forma de la distribución $\pi(x)$.

Si el procedimiento descrito arriba se repite para cada uno de N valores iniciales x_1^0, \dots, x_N^0 , entonces al final de un número grande de iteraciones, T , los valores x_1^T, \dots, x_N^T pueden considerarse como una muestra de tamaño N de $\pi(x)$. Alternativamente, se puede generar una sola cadena y, a partir de un cierto número de iteraciones h , tomar los valores $x^{h+k}, x^{h+2k}, \dots, x^{h+Nk}$, donde k se elige de manera que la correlación entre las observaciones sea despreciable.

Algoritmo de Metropolis-Hasting (M-H)

Este algoritmo también construye una cadena de Markov x^1, \dots, x^t, \dots , con espacio de estados X y distribución de equilibrio $\pi(x)$, definiendo la probabilidad de transición de $x^t = x$ a x^{t+1} de la siguiente manera.

Si $q(x^*, x)$ es una función de probabilidad de transición, de manera que, si

$x^t = x$, x^* obtenida de $q(x^*, x)$ se considera como un valor candidato para x^{t+1} . Con cierta probabilidad $\alpha(x^*, x)$ se acepta $x^{t+1} = x^*$. Si se rechaza el valor candidato de $q(x^*, x)$, entonces $x^{t+1} = x$.

Así, el algoritmo M-H se puede describir como sigue:

- (1) Generar $x^* \sim q(x^*, x)$;
- (2) Generar $u \sim U(0, 1)$
 - si $u \leq \alpha(x^*, x)$, entonces $x^{t+1} = x^*$,
 - si $u > \alpha(x^*, x)$, entonces $x^{t+1} = x$.

La construcción anterior define una cadena de Markov con probabilidades de transición dadas por

$$p(x^*, x) = \begin{cases} q(x^*, x)\alpha(x^*, x) & \text{si } x^* \neq x \\ 1 - \sum_{x^{**}} q(x^{**}, x)\alpha(x^{**}, x) & \text{si } x^* = x \end{cases}$$

Si se define

$$\alpha(x^*, x) = \begin{cases} \min \left[\frac{\pi(x^*)q(x, x^*)}{\pi(x)q(x^*, x)}, 1 \right] & \text{si } \pi(x)q(x^*, x) > 0 \\ 1 & \text{si } \pi(x)q(x^*, x) = 0 \end{cases}$$

se puede verificar que $\pi(x)p(x, x^*) = \pi(x^*)p(x^*, x)$, lo cual junto con la condición de que $q(x^*, x)$ sea irreducible y aperiódica, es una condición suficiente para que $\pi(x)$ sea la distribución de equilibrio de la cadena contruida (ver, por ejemplo, el apéndice A de Smith & Roberts, 1993).

El algoritmo anterior es una versión general debida a Hasting (1970). Es importante notar que $\alpha(\cdot, \cdot)$ sólo depende de $\pi(x)$ a través del cociente $\pi(x^*)/\pi(x)$. Lo anterior es crucial en contextos donde $\pi(x)$ es una distribución final, ya que esto significa que la constante de normalización no es necesaria para la implementación del algoritmo.

Claramente, diferentes selecciones específicas de $q(x^*, x)$ dan como resultado diferentes algoritmos. Tierney (1994) ofreció una clasificación de los posibles tipos de selecciones.

Si $q(x^*, x) = q(x^*, x)$, entonces

$$\alpha(x^*, x) = \min \{ \pi(x^*) / \pi(x), 1 \}$$

lo cual es llamado *algoritmo de Metropolis* (Metropolis *et al.*, 1953).

Si $q(x^*, x) = q(x^* - x)$, la cadena es conducida por un proceso estocástico de caminata aleatoria. En este caso, distribuciones normales multivariadas o t-Student son posibles candidatos para $q(x^* - x)$; ver Muller (1991).

Si $q(x^*, x) = q(x^*)$, entonces

$$\alpha(x^*, x) = \min \{ w(x^*) / w(x), 1 \} \quad \text{con } w(x) = \pi(x) / q(x)$$

en este caso distribuciones t-Student con pocos grados de libertad, son posibles candidatos para $q(x^*, x)$.

Hay que notar que se pueden crear diferentes estrategias al combinar diferentes cadenas en varias formas, por ejemplo, a través de ciclos, mezclando o usando otro método de MCMC para simular las condicionales del muestreo de Gibbs, etc; ver por ejemplo Tierney (1994) y la Sección 7 de Smith & Roberts (1993).

Capítulo 2

Regresión Lineal Generalizada

2.1 Modelos Lineales Generalizados.

Una generalización de los modelos lineales clásicos son los llamados *modelos lineales generalizados*, los cuales tienen como casos especiales a los modelos de regresión lineal, de análisis de varianza, los modelos logit, probit, loglineales, modelos de respuesta multinomial y algunos modelos para datos de supervivencia.

Retomando la discusión de la Sección 1.1, se presenta una breve descripción de los elementos que conforman los modelos lineales generalizados; para mayores detalles, ver McCullagh & Nelder (1989).

En el caso de los modelos lineales clásicos se tiene un vector $Y = (y_1, y_2, \dots, y_n)'$ de n componentes que se distribuyen normal independientemente, con media μ . La parte sistemática de los modelos es una especificación de μ en términos de ciertos parámetros desconocidos β_0, \dots, β_k . En los modelos lineales ordinarios, la parte sistemática toma la forma

$$\mu = X\beta$$

donde X es la matriz del modelo y β es el vector de parámetros desconocidos.

Así, para cada observación i la parte sistemática del modelo se puede reescribir como

$$E(Y_i) = \mu_i = \sum_{j=0}^p x_{ij}\beta_j \quad (i = 1, \dots, n)$$

Donde x_{ij} es el valor de la j -ésima covariable para la observación i . De lo anterior los modelos lineales clásicos se pueden resumir en la siguiente forma: los componentes de Y son variables independientes que se distribuyen Normal con varianza constante σ^2 y

$$E(Y) = \mu \quad \text{donde } \mu = \mathbf{X}\beta. \quad (2.1)$$

Para simplificar la transición hacia los modelos lineales generalizados, (2.1) se puede especificar de la siguiente manera.

1. *Una componente aleatoria:* las componentes de Y se distribuyen Normal con $E(Y) = \mu$ y varianza constante σ^2 ;

2. *Una componente sistemática:* las covariables x_0, x_1, \dots, x_p producen un predictor lineal η dado por

$$\eta_i = \sum_{j=0}^p x_{ij}\beta_j;$$

3. La *Liga* entre la componente aleatoria y la componente sistemática: $\mu = \eta$.

En la generalización las componentes aleatoria y sistemática están relacionadas a través de

$$\eta_i = g(\mu_i)$$

y $g(\cdot)$ es llamada la *función liga*.

En la especificación anterior, los modelos lineales clásicos tienen asociada una distribución Normal (o Gaussiana) en la componente aleatoria y la función identidad como función liga. Los modelos lineales generalizados permiten dos extensiones; la primera, que la distribución en la componente aleatoria puede ser cualquier otra distribución que pertenezca a una familia exponencial y la segunda, que la función liga puede ser cualquier función monótona diferenciable.

2.2 Modelo de Regresión Generalizado Semiparamétrico

A partir de la discusión anterior el modelo (1.3) del Capítulo 1 se puede generalizar de la siguiente manera (ver también McCullagh & Nelder, 1989).

Componente Aleatoria. Y_1, \dots, Y_n son variables aleatorias independientes cuya distribución es un miembro de la familia exponencial, con función de densidad de probabilidad dada por

$$p(y_i | \theta_i, \sigma^2) = b(y_i, \sigma^2/m_i) \exp(m_i[y_i\theta_i - a(\theta_i)]/\sigma^2) \quad (2.2)$$

con $a(\cdot)$ y $b(\cdot, \cdot)$ ciertas funciones específicas, donde los $\{m_i\}$ son pesos conocidos, asociados a cada observación. Si σ^2 se conoce, entonces (2.2) es un modelo que pertenece a la *familia exponencial natural* (ver, por ejemplo, Morris 1982) con *parámetro canónico* θ_i . En este caso

$$\mu_i = E(y_i | \theta_i, \sigma^2) = a'(\theta_i) \quad \text{y} \quad \text{Var}(y_i | \theta_i, \sigma^2) = \frac{\sigma^2}{m_i} a''(\theta_i)$$

El parámetro σ^2 es conocido como el *parámetro de dispersión*.

Componente sistemática. Para cada respuesta Y_i , se tiene asociado un vector de covariables $x_i = (x_{i1}, \dots, x_{ir})'$, con el cual se obtiene un predictor de la forma

$$\eta_i = \eta(x_i) = p_k(x_i) + \delta_i$$

donde $p_k(x_i) = \beta_0 + \beta_1 h_1(x_i) + \dots + \beta_k h_k(x_i) = \mathbf{h}(x_i)' \boldsymbol{\beta}$ es un término paramétrico (lineal) y $\delta_i = \delta(x_i)$ representa el componente no paramétrico del modelo.

La diferencia de este modelo y el modelo usual descrito por McCullagh & Nelder es la introducción del error determinístico, δ_i , en la componente sistemática.

Liga. Las componentes aleatoria y sistemática se relacionan vía la función liga, de tal manera que

$$\eta_i = g(\mu_i)$$

Una liga particularmente importante se obtiene cuando $g^{-1}(\cdot) = a'(\cdot)$. En este caso $\theta_i = \eta_i$ y $g(\cdot)$ se denomina la *liga canónica*.

En los modelos lineales generalizados ordinarios, es importante la selección de $g(\cdot)$ ya que la función liga define la escala sobre la cual los efectos sistemáticos *lineales* son aditivos. Por el contrario, el énfasis del modelo semiparamétrico es sobre la estimación semiparamétrica de la respuesta media $\mu(\cdot)$, por lo que se puede esperar que el modelo sea bastante robusto ante cambios moderados en la especificación de la función liga.

Por otra parte, la función liga juega un papel importante, ya que si se elige convenientemente evita tener que imponer restricciones sobre los valores de β y δ en casos donde Θ sea un subconjunto propio de \mathbf{R} . Un problema de este tipo ocurre, por ejemplo, si se utiliza la liga canónica en el caso de la distribución gamma, pues entonces $\Theta = (-\infty, 0)$, lo que llevaría a imponer la restricción $\eta_i < 0$. En situaciones como esta, es posible modificar la liga canónica introduciendo una transformación (conveniente) $\theta_i = t(\eta_i)$, donde $t(\cdot)$ mapea la recta real sobre el espacio parametral canónico Θ . Dicha transformación, $t(\cdot)$, induce la liga $\eta_i = g(\mu_i)$ donde $g^{-1}(\cdot) = a'(t(\cdot))$. Se puede notar que si $t(\cdot)$ es el mapeo identidad se tiene la liga canónica en los modelos lineales generalizados ordinarios. La transformación $t(\cdot)$ puede considerarse como una función liga entre el predictor η_i y el parámetro canónico θ_i .

Las inferencias clásicas para modelos lineales generalizados se basan en la estimación de los parámetros y las propiedades distribucionales asintóticas de los estimadores. Los modelos de la familia exponencial poseen ciertas propiedades de convexidad que en muchos casos garantizan la existencia y unicidad de los estimadores de máxima verosimilitud (Wedderburn, 1976). Sin embargo, en general la maximización de la verosimilitud requiere de métodos numéricos. Por otro lado, los métodos de Monte Carlo vía Cadenas de Markov producen una forma relativamente sencilla para hacer inferencias Bayesianas para una clase amplia de modelos lineales generalizados (ver, por ejemplo, Dellaportas & Smith, 1993). Gutiérrez-Peña & Smith (1998) presentan un análisis Bayesiano del modelo semiparamétrico en esta sección, basado en el muestreo de Gibbs.

Antes de analizar el modelo general semiparamétrico presentado hasta ahora, es necesario especificar distribuciones iniciales sobre los parámetros desconocidos del modelo. En este caso se asume, a priori, que $\delta(\cdot)$ es un proceso Gaussiano con media cero y función de covarianza $\rho^2 K_\lambda(\cdot, \cdot)$. Sack *et. al.* (1989) discuten una familia de funciones de correlaciones de la forma

$$K_\lambda(x, x^*) = \prod_{i=1}^r \lambda_i^{|x_i - x_i^*|^\alpha} \quad (2.3)$$

donde $0 < \alpha \leq 2$. Así, $K_\lambda(\cdot, \cdot)$ se puede ver como el producto de las correlaciones unidimensionales. Si $r = 1$ y $\alpha = 1$ entonces (2.3) se reduce a la correlación usada por Blight & Ott (1975), y en el caso de que $\alpha = 2$ se obtiene la función de correlación usada por O'Hagan (1992). Se puede ver que la familia (2.3) ofrece gran flexibilidad en varios casos. De lo anterior se sigue que δ se distribuye normal con media $\mathbf{0}$ y matriz de varianza-covarianza $\rho^2 \Lambda_\lambda$, donde Λ_λ es una matriz de $n \times n$ con elementos $K_\lambda(x_i, x_j)$ ($i, j = 1, 2, \dots, n$). Por otro lado, el uso de distribuciones iniciales normales para los coeficientes de un modelo lineal generalizado es comúnmente adecuado para describir la información inicial sobre estos parámetros. Así, se tiene la siguiente especificación jerárquica para el modelo semiparamétrico.

Nivel 1. Condicionalmente sobre β y δ , Y_1, \dots, Y_n son variables aleatorias independientes con $y_i \sim p(y_i | \theta_i, \sigma^2)$ y $\theta_i = t(x_i' \beta + \delta_i)$ ($i = 1, \dots, n$). Lo cual produce la siguiente verosimilitud aproximada

$$l(y | \beta, \delta) \propto \exp \left\{ \frac{1}{\sigma^2} \sum_{i=1}^n m_i [y_i t(x_i' \beta + \delta_i) - a(t(x_i' \beta + \delta_i))] \right\} \quad (2.4)$$

Nivel 2. Condicionalmente sobre ρ^2 y λ , los parámetros β y δ son independientes y

$$\begin{aligned} \beta &\sim N(b_0, B_0^{-1}) \\ \delta &\sim N(\mathbf{0}, \rho^2 \Lambda) \end{aligned} \quad (2.5)$$

La densidad final correspondiente a esta especificación inicial es de la forma

$$\begin{aligned}
 p(\beta, \delta | \mathbf{y}) &\propto \exp \left\{ \frac{-1}{2} (\gamma - t_0)' T_0 (\gamma - t_0) \right\} \\
 &\times \exp \left\{ \frac{1}{\sigma^2} \sum_{i=1}^n m_i [y_i t(x_i' \beta + \delta_i) - a(t(x_i' \beta + \delta_i))] \right\} \\
 &= \exp \left\{ \frac{-1}{2} (\gamma - t_0)' T_0 (\gamma - t_0) + \frac{1}{\sigma^2} \sum_{i=1}^n m_i [y_i t(D_i \gamma) - a(t(D_i \gamma))] \right\}
 \end{aligned} \tag{2.6}$$

donde $\gamma = (\beta', \delta')'$ es un vector de $(p+n) \times 1$, con $\delta = (\delta_1, \dots, \delta_n)$ un vector de $n \times 1$. Por otro lado, $t_0 = (b_0, \mathbf{0})'$ es un vector de $(p+n) \times 1$ y T_0 una matriz de $(p+n) \times (p+n)$ que tiene la siguiente forma

$$T_0 = \begin{bmatrix} B_0 & \mathbf{0} \\ \mathbf{0} & (\rho^2 \Lambda)^{-1} \end{bmatrix}.$$

Finalmente $D_i = (x_i', \mathbf{1}_i')'$ un vector de $(p+n) \times 1$, con $\mathbf{1}_i$ un vector de n componentes todos iguales a cero excepto en el lugar i -ésimo donde tiene un valor de uno.

No en todos los casos se pueden hacer inferencias analíticas exactas basadas en (2.6), por lo que se requiere de aproximaciones alternativas como las discutidas en la Sección 1.3.

2.2.1 Análisis del Modelo

Para hacer inferencias sobre la densidad final (2.6), las versiones del algoritmo de Metropolis-Hastings requieren de la especificación de funciones de probabilidad de transición $q(\cdot | \cdot)$ (ver Sección 1.3.4).

En este trabajo se consideran dos elecciones particulares de funciones $q(\cdot | \cdot)$; las cuales en la práctica resultan muy comunes.

La primera es

$$q(\gamma^* | \gamma) = N(\gamma^* | \gamma, \mathbf{V}(\hat{\gamma})),$$

donde $\gamma = (\beta, \delta)$.

En este caso $q(\gamma^* | \gamma) = q(\gamma^* - \gamma)$ y el procesos de Markov se conoce como

un proceso de *Caminata Aleatoria*. En este caso

$$\alpha(\gamma^* | \gamma) = \min \left\{ \frac{p(\gamma^* | \mathbf{y})}{p(\gamma | \mathbf{y})}, 1 \right\}.$$

La segunda elección es

$$q(\gamma^* | \gamma) = N(\gamma^* | \hat{\gamma}, \mathbf{V}(\hat{\gamma})),$$

por lo que $q(\gamma^* | \gamma) = q_0(\gamma^*)$. Este algoritmo se conoce como el algoritmo de *Independencia* con

$$\alpha(\gamma^* | \gamma) = \min \left\{ \frac{w(\gamma^*)}{w(\gamma)}, 1 \right\} \quad \text{y}$$

$$w(\gamma) = p(\gamma | \mathbf{y}) / q(\gamma).$$

En los dos casos anteriores $\hat{\gamma}$ y $\mathbf{V}(\hat{\gamma})$ denotan una aproximación a la media y a la matriz de varianzas-covarianzas de $p(\gamma | \mathbf{y})$, respectivamente.

Es común en estas situaciones utilizar la aproximación asintótica normal para $p(\gamma | \mathbf{y})$. A continuación se analiza este problema con base en lo presentado por West (1985, Sección 4).

Si $\gamma \sim N(t_0, \mathbf{T}_0^{-1})$, donde t_0 y \mathbf{T}_0^{-1} son conocidos, la distribución final (2.6) será unimodal cuando la verosimilitud (2.4) sea unimodal, lo cual ocurre la gran mayoría de los casos de interés (ver Wedderburn 1976). En este caso las funciones de score y de información quedan definidas de la siguiente forma:

$$g(\gamma | \mathbf{y}, \sigma^2) = \frac{d}{d\gamma} \log p(\gamma | \mathbf{y}, \sigma^2)$$

$$= \frac{1}{\sigma^2} [\sum_{i=1}^n m_i \{y_i - a'(t(D'_i \gamma))\} t'(D'_i \gamma) D'_i] - \mathbf{T}_0(\gamma - t_0)$$

y

$$G(\gamma | \mathbf{y}, \sigma^2) = -\frac{d}{d\gamma} g(\gamma | \mathbf{y}, \sigma^2) =$$

$$\frac{1}{\sigma^2} \{ \sum_{i=1}^n m_i [a''(t(D'_i \gamma))(t'(D'_i \gamma))^2 - [y_i - a'(t(D'_i \gamma))] t''(D'_i \gamma)] D_i D'_i \} + \mathbf{T}_0$$

En el caso en que $t(\cdot)$ sea el mapeo identidad

$$g(\gamma | y, \sigma^2) = \frac{1}{\sigma^2} \left[\sum_{i=1}^n m_i [y_i - a'(D'_i \gamma)] D_i \right] - \mathbf{T}_0(\gamma - t_0) \quad (2.7)$$

$$\begin{aligned} G(\gamma | y, \sigma^2) &= \frac{1}{\sigma^2} \left\{ \sum_{i=1}^n m_i a''(D'_i \gamma) D_i D'_i \right\} + \mathbf{T}_0 \\ &= \frac{1}{\sigma^2} D' A(\gamma) D + \mathbf{T}_0 \end{aligned} \quad (2.8)$$

donde $A(\gamma)$ es una matriz diagonal con i -ésimo elemento $m_i a''(D'_i \gamma)$, $i = 1, \dots, n$, y D una matriz con i -ésimo renglón igual a D_i .

La moda de $p(\gamma | y, \sigma^2)$, t_1 , se puede calcular iterativamente como el límite del algoritmo gradiente de segundo orden, dado por

$$\gamma_{k+1} = \gamma_k + G^{-1}(\gamma_k | y, \sigma^2) g(\gamma_k | y, \sigma^2). \quad (2.9)$$

Usando (2.7-2.9) se puede ver que

$$\gamma_{k+1} = \left[\frac{1}{\sigma^2} D' A(\gamma_k) D + \mathbf{T}_0 \right]^{-1} \left[\frac{1}{\sigma^2} D' A(\gamma) Z(\gamma_k) + \mathbf{T}_0 t_0 \right] \quad (2.10)$$

donde $Z(\gamma_k)$ es un vector $(n \times 1)$ con elementos

$$Z_i(\gamma_k) = D'_i \gamma + [y_i - a'(D'_i \gamma)] / a''(D'_i \gamma) \quad i = 1, \dots, n.$$

Una vez obtenida la moda de $p(\gamma | y, \sigma^2)$, t_1 , ésta se puede usar como una aproximación a la $E(\gamma | y)$ y $G(t_1 | y, \sigma^2)^{-1}$, como una aproximación a la $\text{Var}(\gamma | y)$.

Para emplear el algoritmo anterior y obtener la moda t_1 se necesita partir de un valor inicial γ_0 . Debe notarse que las recursiones (2.7-2.10) dependen de γ sólo a través de $\theta_i = D'_i \gamma$, por lo que basta encontrar valores iniciales $\hat{\theta}_{i0}$ para cada $i = 1, \dots, n$ (los cuales definen implícitamente un valor inicial γ_0). Dichos valores iniciales se pueden elegir como aquellos valores que maximizan, para cada i , la distribución final obtenida al considerar como inicial la distribución no informativa de Jeffreys para θ_i ($i = 1, \dots, n$). A continuación se presentan

algunos de los desarrollos correspondientes para obtener este valor inicial.

· Si y tiene una función de distribución que pertenece a una familia exponencial con θ_i y se considera una inicial de Jeffreys para θ_i , es decir,

$$\begin{aligned}\pi(\theta_i) &\propto \left| -E \left(\frac{\partial^2 \ln p(y|\theta_i)}{\partial \theta_i^2} \right) \right|^{1/2} \\ &= |a''(\theta_i)|^{1/2}.\end{aligned}$$

· Entonces, se puede ver que para los casos Normal, Gamma, Poisson y Binomial (ver Gutiérrez-Peña & Smith, 1997)

$$\pi(\theta_i) \propto \exp \{k_1 \theta_i - k_2 a(\theta_i)\}$$

con $k_1 = k_2 = 0$ para el caso Normal; $k_1 = 0$, $k_2 = -1$ para el caso Gamma; $k_1 = 1/2$, $k_2 = 0$ para el caso Poisson y $k_1 = 1/2$, $k_2 = 1$ para el caso Binomial.

De lo anterior se obtiene que

$$\begin{aligned}\pi(\theta_i | y_i) &\propto \exp \left\{ \frac{1}{\sigma^2} m_i [y_i \theta_i - a(\theta_i)] \right\} \exp \{k_1 \theta_i - k_2 a(\theta_i)\} \\ &= \exp \left\{ \theta_i \left(\frac{m_i}{\sigma^2} y_i + k_1 \right) - \left(\frac{m_i}{\sigma^2} + k_2 \right) a(\theta_i) \right\}.\end{aligned}\quad (2.11)$$

Se puede mostrar que la moda ($\hat{\theta}_i$) de $\pi(\theta_i | y_i)$ cumple con que

$$a'(\hat{\theta}_i) = \frac{y_i + a_i}{1 + c_i}$$

con $a_i = \frac{\sigma^2 k_1}{m_i}$ y $c_i = \frac{\sigma^2 k_2}{m_i}$.

Resumiendo lo expuesto hasta ahora se tiene lo siguiente.

1. Una vez obtenida la moda ($\hat{\theta}_i$) de (2.11) y recordando que $\theta_i = D_i' \gamma$, se puede tomar como valor inicial para el algoritmo gradiente de segundo orden un γ_0 definido implícitamente por $\hat{\theta}_i = D_i' \gamma_0$ ($i = 1, \dots, n$). Sin embargo, como ya se mencionó basta con encontrar los $\hat{\theta}_i$'s para inicializar el algoritmo.

2. Una vez elegido un valor inicial para el algoritmo gradiente se puede encontrar la moda t_1 de $p(\gamma | y, \sigma^2)$ y consecuentemente se puede obtener una aproximación de $E(\gamma | y)$ y $\text{Var}(\gamma | y)$.

3. Con la aproximación de $E(\gamma | y)$ y $\text{Var}(\gamma | y)$ se esta en condiciones de

correr los algoritmos de *Caminata Aleatoria* o de *Independencia*.

2.2.2 Estimación de la Superficie de Regresión

Una aplicación importante de las técnicas de regresión es la estimación de la relación funcional entre la variable de respuesta Y y las variables explicativas. En otras palabras, se desea hacer inferencia sobre el valor $\mu_x = \mu(x) = E(Y | X = x)$ en un punto dado x .

De la Sección 1.1.2, η_x se puede ver como $\eta_x = p_k(x) + \delta_x$, donde $p_k(x)$ es la función simple que se utiliza para aproximar a η_x . Se puede notar que si x no es un punto de diseño, que es el caso de interés, lo que se necesita es hacer inferencias acerca de δ_x .

Por lo visto en la Sección 1.1.2, se sigue que la distribución condicional de δ_x dado δ es normal con

$$E(\delta_x | \delta) = \mathbf{c}' \mathbf{R}^{-1} \delta$$

y

$$\text{Var}(\delta_x | \delta) = \rho^2 - \mathbf{c}' \mathbf{R}^{-1} \mathbf{c},$$

donde $\mathbf{c} = \mathbf{c}(x) = \rho^2 (K_\lambda(x, x_1), \dots, K_\lambda(x, x_n))'$.

Con lo anterior se pueden obtener muestras de la distribución de δ_x y consecuentemente, muestras de la distribución de η_x , y a partir de éstas últimas se pueden hacer inferencias sobre η_x . En particular se puede proponer un estimador de η_x , $\hat{\eta}_x$. Las propiedades de $\hat{\eta}_x$ dependerán fuertemente de la selección de la función de covarianza $\Sigma^* = \rho^2 \mathbf{\Lambda}_\lambda$, donde $\mathbf{\Lambda}_\lambda$ es una matriz con elementos $K_\lambda(x, x_i)$ (ver Sección 1.1.2. y Sección 2.2.1.), y de los valores de ρ^2 y λ .

La función de covarianza $\mathbf{\Lambda}_\lambda$ controla el grado de suavizamiento del estimador $\hat{\eta}_x$. La siguiente forma Gaussiana estacionaria

$$\begin{aligned} \Lambda_\lambda(x, \tilde{x}) &= \exp\{-\tau(x, \tilde{x})' \mathbf{W}^{-1}(x, \tilde{x})\}, \quad x, \tilde{x} \in \mathbf{R}^r, \theta \in \mathbf{R}^+ \\ &= \lambda^{\{(x, \tilde{x})' \mathbf{W}^{-1}(x, \tilde{x})\}}, \end{aligned}$$

con \mathbf{W} una matriz de pesos, simétrica de $r \times r$, ha sido usada por O'Hagan (1978, 1992), entre otros. Esta forma Gaussiana produce estimadores suaves de

η_x infinitamente diferenciables. En las aplicaciones presentadas en este trabajo se empleará la siguiente forma particular de \mathbf{W} , usada por Gutiérrez-Peña y Smith (1998),

$$\mathbf{W} = \text{Diag}(z_1^2, \dots, z_r^2),$$

donde

$$z_j = (n-1)^{-1} \sum_{l=1}^{n-1} |x_{(l+1)j} - x_{(l)j}|$$

y $x_{(1)} \leq \dots \leq x_{(n)}$, denotan las estadísticas de orden de la j -ésima covariable ($j = 1, \dots, r$). Este escalamiento resulta adecuado en muchos casos y produce inferencias que son invariantes bajo cambios de escala en las covariables.

2.3 Comparación de Modelos Lineales Generalizados

En esta sección se extiende el enfoque semiparamétrico presentado en la Sección 1.2.2, al caso de los modelos lineales generalizados.

Sea $\mathcal{M} = \{M_1, \dots, M_k\}$ un conjunto de modelos de regresión paramétricos. El problema es seleccionar uno de los modelos en \mathcal{M} con fines predictivos, donde

$$M_i = \{p_i(y|\beta_i), p_i(\beta_i)\} \quad y \in \mathbf{R}, \quad \beta_i \in \mathbf{R}^{q_i}.$$

En este caso

$$p_i(y|\beta_i) = p(y | \theta(\beta_i), \sigma^2)$$

y

$$p_i(\beta_i) = N_{q_i}(\beta_i | b_{0i}, \mathbf{B}_{0i}),$$

donde $p(y | \theta(\beta_i), \sigma^2) = b(y, \sigma^2/m) \exp(m|y\theta - a(\theta)|/\sigma^2)$ denota una familia exponencial y $\theta(\beta_i) = t(\mathbf{h}_i(x)' \beta_i)$ define tanto al predictor lineal como a la función liga.

Algunas ventajas de trabajar con el parámetro canónico θ_i en vez del parámetro medio μ , es que tanto la verosimilitud como la distribución inicial se pueden parametrizar en términos de θ_i . Además, al asumir la liga canónica, bajo la cual

$\theta_i = \eta_i$, la función de verosimilitud posee ciertas propiedades de log-concavidad deseadas.

El enfoque semiparamétrico propuesto en esta tesis para el problema de selección de modelos, considera éste como un problema de decisión con los siguientes elementos.

Espacio de decisiones:

$$D = \mathcal{M}$$

Espacio de 'sucesos inciertos':

$$F = \{\eta : \eta \text{ es una función suave sobre } \mathbf{R}^r\}.$$

Distribución inicial sobre F :

(a). Condicional sobre β

$$\eta(x) \sim \mathbf{N}(\mu_{\beta}^*(x), \Sigma^*(x, x)) \quad (\text{Proceso Gaussiano})$$

con

$$\begin{aligned} \mu_{\beta}^*(x) &= \mathbf{h}(x)' \beta \\ \Sigma^*(x, x) &= \rho^2 \Lambda_{\lambda} \end{aligned}$$

es decir, dado β

$$\eta(x) = \mathbf{h}(x)' \beta + \delta(x), \quad \text{donde } \delta(x) \sim \mathcal{N}(0, \rho^2 \Lambda_{\lambda}).$$

(b).

$$\beta \sim N_q(b_0, \Sigma_0) \quad \Sigma_0 = \mathbf{B}_0^{-1}$$

Función de utilidad sobre $D \times F$:

$$u(M_i, \eta) = \int \log p_i(y_* | \mathbf{y}) p(y_* | \eta(x_*)) dy_*$$

con

$$p_i(y_* | \mathbf{y}) = \int p_i(y_* | \beta_i) p_i(\beta_i | \mathbf{y}) d\beta,$$

la distribución predictiva final de una observación futura y_* , considerando el i -ésimo modelo. Aquí $p(y_*|\eta(x_*))$ representa un modelo de la familia exponencial natural.

Por tanto la solución al problema de decisión es la selección del modelo que maximice la utilidad esperada final.

$$\begin{aligned}
 u(M_i) &= E_{\eta|\mathbf{y}} [u(M_i, \eta)] \\
 &= E_{\eta|\mathbf{y}} [\int \log p_i(y_*|\mathbf{y}) p(y_*|\eta(x_*)) dy_*] \\
 &= \int \log p_i(y_*|\mathbf{y}) E_{\eta|\mathbf{y}} p(y_*|\eta(x_*)) dy_* \\
 &\quad - \int \log p_i(y_*|\mathbf{y}) p_{x_*}(y_*|\mathbf{y}) dy_*
 \end{aligned} \tag{2.12}$$

donde $p_i(y_*|\mathbf{y})$ es la distribución predictiva del i -ésimo modelo y $p_{x_*}(y_*|\mathbf{y})$ es la distribución predictiva del modelo semi-paramétrico, asociado al punto x_* . Así, el modelo M_i será preferido al modelo M_j si y sólo si

$$\bar{u}(M_i) > \bar{u}(M_j).$$

Se puede observar que la utilidad esperada final (2.12) depende del valor particular de x_* para el cual se realice la predicción. La pregunta en este punto es ¿qué valor de x_* se debe usar? Gutiérrez-Peña (1997b) propone las siguientes opciones:

- (a) Seleccionar un valor de x_* en la región de interés (por ejemplo, seleccionar $x_* = \bar{x}$, donde \bar{x} es el centroide de los puntos observados $\{x_1, \dots, x_n\}$), y usar entonces la correspondiente utilidad esperada.
- (b) Calcular la $u(M_i)$ para cada uno de los valores observados x_l ($l = 1, \dots, n$) y usar el valor promedio de estas utilidades como un criterio para la selección de modelos.

Hay que notar que los cálculos necesarios para obtener las utilidades esperadas (2.12) no se pueden realizar analíticamente, por lo que es necesario emplear métodos como los descritos en la Sección 1.3.

Para el cálculo de las utilidades esperadas finales (2.12), es posible simu-

lar observaciones $y_*^{(1)}, y_*^{(2)}, \dots, y_*^{(IM)}$ de la distribución predictiva final $p_{x_*}(y_*|\eta(x_*))$ de la siguiente manera:

Dada una muestra $\{(\beta^{(1)}, \delta_{x_*}^{(1)}), \dots, (\beta^{(IM)}, \delta_{x_*}^{(IM)})\}$ de la distribución final $p(\beta, \delta_x)$, $\beta^{(j)} \in \mathbf{R}^3$, $j = 1, \dots, IM$, $\delta_x \in \mathbf{R}$; se pueden generar IM observaciones $y_*^{(j)} \sim p(y_*|\eta^{(j)}(x_*))$, $j = 1, \dots, IM$, donde $p(y_*|\eta^{(j)}(x_*))$ es un modelo de la familia exponencial con

$$\eta^{(j)}(x_*) = h(x_*)' \beta^{(j)} + \delta_{x_*}^{(j)}.$$

Se pueden entonces aproximar las utilidades esperadas finales a través del estimador de Monte Carlo

$$\bar{u}(M_i) = (IM)^{-1} \sum_{j=1}^{IM} \log p_i(y_*^{(j)}|\mathbf{y}).$$

El problema es que no se tienen expresiones analíticas para $p_i(y_*|\mathbf{y})$. Por otra parte, no es factible, en términos prácticos, utilizar el argumento anterior para aproximar $p_i(y_*|\mathbf{y})$ vía Monte Carlo. Una propuesta para soslayar este problema es utilizar una aproximación analítica del tipo

$$\hat{p}_i(y_*|\mathbf{y}) = p_i(y_*|\hat{\beta}_i),$$

donde $\hat{\beta}_i$ es la moda de la distribución final correspondiente o el estimador de máxima verosimilitud. Dicha aproximación será más adecuada en la medida en que la dispersión de la distribución final de β_i sea menor, lo cual ocurre conforme el tamaño de muestra aumenta.

En el siguiente capítulo se ilustra la aplicación del enfoque semiparamétrico para la comparación de modelos de regresión lineal generalizados y se detalla el cálculo de las utilidades esperadas (2.12).

Capítulo 3

Aplicaciones

3.1 Ejemplo 1

Los datos de la Tabla 3.1, tomados de Knuiman y Speed (1988), reflejan la relación entre la duración de la diabetes (medida en años) y la retinopatía (una enfermedad de los ojos). La Tabla 3.1 también presenta los datos de un estudio previo.

Tabla 3.1. *Datos de diabetes-retinopatía, Knuiman y Speed (1988).*

Duración de la Diabetes (en años)	Estudio previo.		Estudio actual.	
	Presencia de Retinopatía Si	No	Presencia de Retinopatía Si	No
0-2	17	215	46	290
3-5	26	218	52	211
6-8	39	137	44	134
9-11	27	62	54	91
12-14	35	36	38	53
15-17	37	16	39	42
18-20	26	13	23	23
21+	23	15	52	32

Knuiman y Speed sugieren el uso de un modelo logístico definido por

$$\log \left(\frac{\pi_{1j}}{\pi_{2j}} \right) = \beta_1 + \beta_2 X_j + \beta_3 X_j^2 = \eta_j \quad (3.1)$$

donde (π_{1j}, π_{2j}) son las probabilidades de tener o no retinopatía para personas con tiempo de duración de diabetes en la j -ésima categoría, y X_j es el vector de duraciones medias para los rangos especificados, dado por (1, 4, 7, 10, 13, 16, 19, 24).

Knuiman y Speed incorporan la información inicial a través de una distribución normal multivariada, con vector de medias y matriz de covarianza obtenidos de estimadores máximo verosímiles basados en los datos del estudio previo. Dichos estimadores están dados por

$$\hat{E}(\beta) = \begin{pmatrix} -3.17 \\ 0.33 \\ -0.007 \end{pmatrix}, \quad \hat{\text{Var}}(\beta) = 10^{-4} \begin{pmatrix} 638 & & \\ -111 & 24.1 & \\ 3.9 & -0.9 & 0.04 \end{pmatrix} \quad (3.2)$$

El análisis de la distribución final obtenida a partir del modelo (3.1) y de la distribución inicial $N(\beta_0, D_0)$ requiere de integración numérica en tres dimensiones. En su lugar Knuiman y Speed optan por un análisis aproximado basado en la aproximación normal asintótica para la distribución final (ver Subsección 1.3.1), y reportan las siguientes aproximaciones a la media final y a la matriz de covarianzas, respectivamente:

$$\hat{E}(\beta|y) = \begin{pmatrix} -2.37 \\ 0.21 \\ -0.004 \end{pmatrix}, \quad \hat{\text{Var}}(\beta|y) = 10^{-4} \begin{pmatrix} 207 & & \\ -36 & 8.1 & \\ 1.2 & -0.3 & 0.01 \end{pmatrix}.$$

Dellaportas y Smith (1993) realizan un análisis numérico Bayesiano del modelo (3.1), usando el muestreo de Gibbs, con lo que se evitan tanto la integración numérica directa como el suponer normalidad aproximada para la distribución final de los parámetros del modelo. Después de correr el muestreo de Gibbs con 150 iteraciones, donde en las iteraciones finales usaron una muestra de tamaño 500, estos autores reportan las siguientes aproximaciones asociadas al vector de

medias y a la matriz de covarianzas.

$$\tilde{E}(\beta|y) = \begin{pmatrix} -2.36 \\ 0.21 \\ -0.004 \end{pmatrix}, \quad \tilde{\text{Var}}(\beta|y) = 10^{-4} \begin{pmatrix} 201 & & \\ -35.7 & 7.9 & \\ 1.2 & -0.3 & 0.01 \end{pmatrix}.$$

El análisis realizado por Dellaportas y Smith da soporte a la hipótesis utilizada por Knuiman y Speed sobre la normalidad aproximada final, aunque las aproximaciones de estos últimos autores no ofrecen una validación interna. Sin embargo, en muchas aplicaciones de modelos lineales generalizados, los tamaños de muestra son pequeños y no siempre se cuenta con información inicial. En tales casos la hipótesis de normalidad asintótica puede ser errónea. Ver, por ejemplo, Dellaportas y Smith (1993, Sección 7).

3.1.1 Análisis del Modelo Logístico Cuadrático vía M-H.

La información del estudio previo (Tabla 3.1) se modeló a través de una distribución normal multivariada $N(b_0, B_0^{-1})$, para el vector $\beta' = (\beta_1, \beta_2, \beta_3)$ de parámetros desconocidos.

Retomando los desarrollos presentados en la Sección 2.2, los valores de $b_0 = E(\beta)$ y $B_0^{-1} = \text{Var}(\beta)$ se obtuvieron de la siguiente forma:

El programa M ϕ .F (ver Apéndice B) considera la distribución final del vector $\gamma' = (\beta', \delta')$, y obtiene una aproximación de $E(\gamma|y)$ y la $\text{Var}(\gamma|y)$.

Para obtener resultados comparables con los de la sección anterior, se consideró una inicial para el vector δ , con $\rho^2 = 0$ y una inicial para el vector β con media cero y matriz de varianza-covarianza dada por $10000 \times I$, donde I es una matriz identidad. Con la especificación anterior, y usando los datos del estudio previo se corrió el programa M ϕ .F, con lo cual se obtuvo una aproximación a $E(\gamma)$ y $\text{Var}(\gamma)$ y, consecuentemente, una aproximación de $E(\beta)$ y de $\text{Var}(\beta)$.

Después de 3 iteraciones del programa la convergencia era clara, lo cual era de esperarse por la propiedad de log-concavidad de la

función de verosimilitud.

Las aproximaciones para el vector de medias y la matriz de covarianzas que se obtuvieron fueron:

$$b_0 = \begin{pmatrix} -3.17 \\ 0.327 \\ -0.0068 \end{pmatrix}, \quad B_0^{-1} = 10^{-4} \begin{pmatrix} 639.14 & & \\ -110.99 & 24.19 & \\ 3.89 & -0.943 & 0.0399 \end{pmatrix}.$$

Tal como se esperaba, estos estimadores esencialmente coinciden con los estimadores de máxima verosimilitud reportados por Knuiman y Speed, ecuación (3.2).

Una vez especificada completamente la distribución inicial del vector β como una normal multivariada $N(\beta|b_0, B_0^{-1})$, haciendo $\rho^2 = 0$ y considerando la información del estudio actual (Tabla 3.1), se corrió el programa M ϕ .F nuevamente, obteniéndose los correspondientes valores iniciales para $E(\gamma|\mathbf{y})$ y $\text{Var}(\gamma|\mathbf{y})$.

En particular se obtuvo

$$E(\beta|\mathbf{y}) \approx \begin{pmatrix} -2.3651 \\ 0.2077 \\ -0.00367 \end{pmatrix}, \quad \text{Var}(\beta|\mathbf{y}) \approx 10^{-4} \begin{pmatrix} 206.859 & & \\ -35.672 & 8.0899 & \\ 1.2228 & -0.3082 & 0.01294 \end{pmatrix}. \quad (3.3)$$

Los valores de $E(\gamma|\mathbf{y})$ y $\text{Var}(\gamma|\mathbf{y})$ se utilizaron como base para construir las funciones de probabilidad de transición tanto para la versión de caminata aleatoria como para la de independencia, del algoritmo de Metropolis-Hastings (M-H) (ver Sección 2.2.1). Para esto se crearon los programas M1.F y M2.F, los cuales implementan los algoritmos de caminata aleatoria e independencia, respectivamente (ver Apéndice B).

Hay que aclarar que, para poder hacer inferencias, sólo es necesario correr alguna de las versiones del algoritmo de M-H, ya que, como se discutió en la Sección 1.3.4, cada uno de los algoritmos contruye una cadena $\beta^{(1)}, \dots, \beta^{(t)}, \dots$, con espacio de estados \mathbf{R}^3 y distribución de equilibrio $p(\beta|\mathbf{y})$. En esta aplicación se presentan algunos resultados obtenidos con ambos algoritmos, sólo

con fines ilustrativos. En lo que resta de esta tesis sólo se utilizará al algoritmo de caminata aleatoria y a éste se aludirá cuando se hable del algoritmo de M-H.

Ambos algoritmos se corrieron considerando muestras simuladas de tamaño $IM = 500$ por un periodo de 850 iteraciones y se obtuvieron como resultados, en particular, dos muestras, $\{\beta^{(1)}, \dots, \beta^{(IM)}\}_{CA}$ y $\{\beta^{(1)}, \dots, \beta^{(IM)}\}_{IND}$, de la distribución final $p(\beta | \mathbf{y})$, con $\beta^{(i)} \in \mathbf{R}^3, \forall i = 1, 2, \dots, IM$.

En la Tabla 3.2 se muestran los ajustes obtenidos con estas dos muestras. También se presenta el ajuste realizado con S-plus contemplando toda la información de la Tabla 3.1.

Se puede notar que los resultados de los dos algoritmos de M-H son parecidos y, como se esperaba, ambos son similares a los resultados obtenidos en S-plus ya que el ajuste en S-plus se realizó considerando toda la información de la Tabla 3.1 (es decir, tomando en cuenta tanto la información del estudio previo como la del estudio actual). Este último análisis arroja resultados similares a los del análisis Bayesiano basado en una distribución inicial localmente uniforme para β y considerando todos los datos de la Tabla 3.1.

En la Figura 3.1 se muestran las medias ergódicas de cada componente del vector $\beta' = (\beta_1, \beta_2, \beta_3)$ de parametros de regresión, obtenidos del algoritmo de caminata aleatoria (M-H).

3.1.2 El modelo Semi-paramétrico. Comparación de Modelos.

En esta sección se aplicará el enfoque semiparamétrico propuesto para el problema de selección de modelos lineales generalizados (ver Sección 2.3).

Para el estudio de diabetes-retinopatía supóngase que se desean comparar los siguientes modelos:

$$\begin{aligned}
 M_0 : \log \left(\frac{\pi_{1i}}{\pi_{2i}} \right) &= \beta_{00} \\
 M_1 : \log \left(\frac{\pi_{1i}}{\pi_{2i}} \right) &= \beta_{10} + \beta_{11}x \\
 M_2 : \log \left(\frac{\pi_{1i}}{\pi_{2i}} \right) &= \beta_{21} + \beta_{22}x + \beta_{23}x^2 \\
 M_3 : \log \left(\frac{\pi_{1i}}{\pi_{2i}} \right) &= \beta_{31} + \beta_{32}x + \beta_{33}x^2 + \beta_{34}x^3 \\
 M_4 : \log \left(\frac{\pi_{1i}}{\pi_{2i}} \right) &= \beta_{41} + \beta_{42}x + \beta_{43}x^2 + \beta_{44}x^3 + \beta_{45}x^4
 \end{aligned}$$

Tabla 3.2. π_{1i} observadas y estimadas

π_{1i}	Algoritmo de Caminata Aleatoria	Algoritmo de Independencia	Ajuste en S-plus
0.13690	0.10221	0.10390	0.09869
0.19772	0.16793	0.16935	0.16601
0.24720	0.25045	0.25141	0.25215
0.37241	0.34126	0.34140	0.34730
0.41759	0.42894	0.42860	0.43894
0.48148	0.50452	0.50360	0.51731
0.50000	0.56340	0.56263	0.57768
0.61905	0.62395	0.62327	0.63707
Devianza	2.73714E-02	2.54892E-02	3.60394E-02

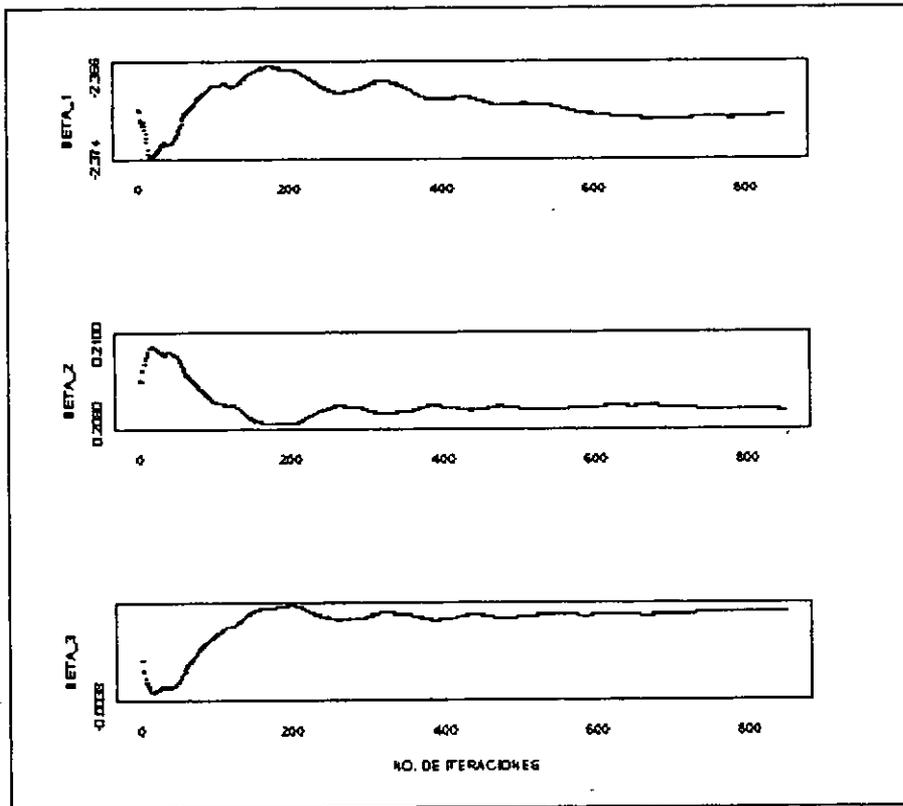


Figura 3.1: Medias ergódicas de β_1 , β_2 y β_3 .

De acuerdo con lo presentado en la Sección 2.3, la solución al problema de decisión es la selección del modelo que maximice la utilidad esperada final:

$$\begin{aligned}\bar{u}(M_i) &= \int \log p_i(y_*|y) E_{\eta|y} p(y_*|\eta(x_*)) dy_* \\ &= \int \log p_i(y_*|y) p_{x_*}(y_*|\eta(x_*)) dy_*\end{aligned}\quad (3.4)$$

donde $p_i(y_*|y)$ es la distribución predictiva del i -ésimo modelo y $p_{x_*}(y_*|\eta(x_*))$ es la distribución predictiva del modelo semi-paramétrico, asociado al punto x_* .

Para efectos del ejemplo se consideraron las siguientes especificaciones:

$$\begin{aligned}x_* = \bar{x} &= 11.75, \rho^2 = 0.7, \theta = 1.0 \quad y \\ \mu_{\beta}^* &= \mathbf{h}(x_*)' \beta = \beta_1 + \beta_2 \bar{x} + \beta_3 \bar{x}^2\end{aligned}$$

De acuerdo a lo discutido en la Sección 2.3, para el cálculo de las utilidades esperadas finales (3.4), se simularon $IM = 1000$ observaciones de $p(y_*|\eta(x_*))$ y las utilidades esperadas finales se pueden aproximar a través de la aproximación de Monte Carlo

$$\bar{u}(M_i) \hat{=} (IM)^{-1} \sum_{j=1}^{IM} \log p_i(y_*^{(j)}|\hat{\beta}_i),$$

donde $\hat{\beta}_i$ es el estimador de máxima verosimilitud asociado al modelo i -ésimo.

Los cálculos de las utilidades esperadas finales para cada modelo se realizaron con rutinas en S-plus (ver Apéndice B). La Tabla 3.3 muestra las utilidades esperadas finales para cada uno de los modelos considerados.

Tabla 3.3. Utilidades esperadas finales $\bar{u}(M_i)$.

Modelo	M_0	M_1	M_2	M_3	M_4	Semiparamétrico
C. Aleatoria	-9.20	-5.484	-4.989	-4.984	-4.994	-4.984

Se puede notar que el modelo M_3 es el mejor modelo y que en términos del poder predictivo los modelos M_2 y M_3 son equivalentes.

El modelo semiparamétrico ofrece un punto de referencia a partir del cual se puede no sólo juzgar qué modelo es el mejor, sino también qué tan lejos se está

del modelo 'verdadero', comparando las utilidades de los otros modelos con la utilidad asociada al modelo semiparamétrico.

La Tabla 3.4 y la Figura 3.2 muestran el ajuste obtenido con el modelo semiparamétrico a los datos de diabetes-retinopatía.

Estimación de la curva de Regresión.

Como se mencionó anteriormente, una aplicación importante de las técnicas de regresión es la estimación de la relación funcional entre la variable de respuesta Y y la variable explicativa X . En otras palabras, se desea hacer inferencia sobre el valor de $\pi_{1x} = \pi_1(x) = E(Y | X = x)$ en un punto dado x .

Para ilustrar todos los conceptos asociados con la estimación de las π_{1i} , y de acuerdo con lo discutido en la Sección 2.2.2, se consideraron 20 puntos equidistantes x_j , $j = 1, \dots, 20$ en el intervalo $[0.5, 26]$ y se obtuvieron las correspondientes π_{1i} de la siguiente manera:

Asumiendo valores de $\rho^2 = 0.7$ y $\lambda = 0.37$, se obtuvo una muestra de la distribución final

$p(\gamma|\mathbf{y}), \{\gamma^{(1)}, \gamma^{(2)}, \dots, \gamma^{(IM)}\}$ con $\gamma^{(i)} \in \mathbf{R}^{3+8}, \forall i = 1, 2, \dots, IM$, correspondiente a las corridas del algoritmo de M-H (caminata aleatoria).

Se creó el programa Predic.F (ver Apéndice B) con el cual se generaron los correspondientes estimadores de $\delta_x, \hat{\delta}_x$, asociados a cada uno de los vectores de las muestras obtenidas.

Para cada vector $\gamma^{(i)}, i = 1, \dots, IM$ en la muestra, se calculó

$$\eta_j^{(i)} = x_j' \beta^{(i)} + \hat{\delta}^{(i)} = C_j' \gamma^{(i)}, \quad \forall j = 1, \dots, 20$$

donde $x_j' = (1, x_j, x_j^2)$, $\beta^{(i)} \in \mathbf{R}^3$, $\hat{\delta}^{(i)} \in \mathbf{R}$, $\gamma^{(i)} = (\beta^{(i)}, \hat{\delta}^{(i)})'$ y $C_j' = (x_j', 1)$ un vector de $(3+1) \times 1$

El estimador de η_x , para cada punto x , se obtuvo a través de la

Tabla 3.4. Ajuste obtenido a los datos de diabetes-retinopatía

π_{1i} observadas	Modelo Semiparamétrico	Ajuste en S-plus
0.13690	0.13239	0.09869
0.19772	0.19609	0.16601
0.24720	0.24367	0.25215
0.37241	0.37248	0.34730
0.41759	0.42155	0.43894
0.48148	0.48257	0.51731
0.50000	0.50553	0.57768
0.61905	0.63752	0.63707
<i>Devianza</i>	0.001054	3.60394E-02

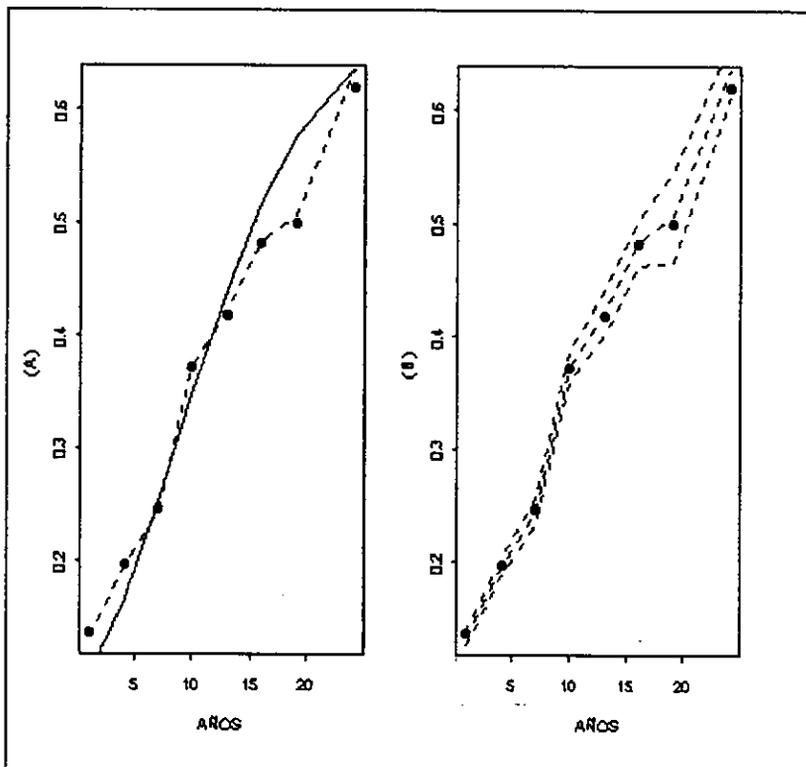


Figura 3.2. Figura (A). Línea punteada.- ajuste obtenidos con el modelo Semiparamétrico. Línea continua.- ajuste obtenido con S-plus, Figura (B). Bandas obtenidas con el modelo semiparamétrico

aproximación de Monte Carlo:

$$\hat{\eta}_x \doteq E(\eta_x|y) \approx (IM)^{-1} \sum_{i=1}^{IM} \eta_x^{(i)}.$$

Finalmente, las predicciones de π_{1i} se obtuvieron como:

$$\tilde{\pi}_{1x} = \frac{\exp(\hat{\eta}_x)}{1 + \exp(\hat{\eta}_x)}.$$

La Tabla 3.5 presenta las predicciones de las π_{1i} asociadas a los 20 puntos en el intervalo [0.5, 26], considerando Metropolis-Hastings y las correspondientes predicciones obtenidas en S-plus.

Tabla 3.5. *Predicciones de 20 puntos en el intervalo [0.5 26]*

Modelo Semiparamétrico	S-plus
0.112719057	0.08960991
0.163071385	0.11537476
0.188230745	0.14557843
0.199441333	0.17996259
0.221385093	0.21795015
0.255267216	0.25867566
0.309912529	0.30106442
0.366796643	0.34394445
0.396836143	0.38616622
0.412732774	0.42670507
0.440755789	0.46472916
0.469815078	0.49962777
0.495567411	0.53100524
0.504744883	0.55865113
0.523423257	0.58249859
0.590447055	0.60258033
0.647911456	0.61898864
0.646476911	0.63184222
0.637330879	0.64125994
0.633744215	0.64734433

3.2 Ejemplo 2

Los datos para este ejemplo se tomaron de Lee (1980). A 51 pacientes adultos con cierto tipo de leucemia se les sometió a un nuevo tratamiento. Al final del tratamiento a cada paciente se le evaluó de la siguiente manera: 1.- si el paciente había respondido al tratamiento, 0.- en otro caso. Las siguientes seis variables pre-tratamiento fueron evaluadas para cada paciente:

1. Edad en años, al inicio del tratamiento, (x_1) .
2. Porcentaje diferencial de mancha de blastocitos, (x_2) .
3. Porcentaje de leucemia medular absoluta infiltrada, (x_3) .
4. Índice etiquetado de células de leucemia medular ósea, (x_4) .
5. Blastocitos absolutos, (x_5) .
6. Temperatura más alta presentada al inicio del tratamiento, (x_6) .

Los datos se presentan en la Tabla 1 del Apéndice A. El objetivo principal es establecer el pronóstico significativo de las variables pre-tratamiento, es decir, cuáles (si hay alguna) de estas variables son predictores de la respuesta de los pacientes al tratamiento. Un análisis tradicional se presenta en Everitt (1996).

3.2.1 Análisis del Modelo

Dado que en este ejemplo no se dispone de información inicial, se especificó la distribución inicial del vector $\beta' = (\beta_1, \beta_2, \beta_3, \beta_4, \beta_5, \beta_6, \beta_7)$ como una normal multivariada $N(\beta|\hat{b}_0, \hat{B}_0^{-1})$, con $\hat{b}_0 = \mathbf{0}$ y $\hat{B}_0^{-1} = 10000 \times \mathbf{I}$, donde $\mathbf{0}$ es un vector en \mathbf{R}^7 de ceros y \mathbf{I} es la matriz identidad de 7×7 .

Para obtener resultados comparables con los expuestos en Everitt (1996) (los cuales fueron obtenidos usando S-plus), se consideró una distribución inicial para el vector δ con $\rho^2 = 0$. Con la información de la Tabla 1 (ver Apéndice A), se corrió el programa M ϕ .F, obteniéndose los correspondientes valores iniciales para $E(\gamma|\mathbf{y})$ y $\text{Var}(\gamma|\mathbf{y})$. Estos valores se utilizaron como base para correr el algoritmo de M-H.

El algoritmo de M-H se corrió considerando muestras simuladas de tamaño $IM = 1000$ por un periodo de 1000 iteraciones y se obtuvo como resultados, en particular, una muestra de la distribución final $p(\beta | \mathbf{y})$, $\{\beta^{(1)}, \beta^{(2)}, \dots, \beta^{(IM)}\}$, con $\beta^{(i)} \in \mathbf{R}^7, \forall i = 1, 2, \dots, IM$.

En la Tabla 3.6 se muestran los ajustes obtenidos con esta muestra. También se presenta el ajuste realizado con S-plus.

Tabla 3.6. π_{1i} estimadas. Dev. de S-plus=39.28, Dev. de M-H=39.39

Resp.	M-H	S-plus	Resp.	M-H	S-plus	Resp.	M-H	S-plus
1	0.5318	0.5132	0	0.2115	0.2134	0	0.0853	0.0989
1	0.6804	0.6235	1	0.9309	0.9300	0**	0.6680	0.6803
1	0.9669	0.9622	1**	0.3446	0.3501	0	0.0043	0.0050
1	0.9747	0.9685	0	0.0619	0.0675	0	0.0404	0.0513
1	0.9186	0.9042	0	0.0034	0.0029	0	0.0477	0.0518
0	0.2188	0.1860	0	0.0198	0.0236	1	0.9705	0.9751
1	0.9989	0.9984	0	0.2214	0.2124	0	0.0016	0.0019
1	0.8197	0.7753	1	0.9646	0.9594	0	0.1879	0.1931
1*	0.5154	0.4883	1	0.6603	0.6631	0	0.0154	0.0203
0**	0.6454	0.6317	0**	0.5748	0.5743	1**	0.4568	0.4838
1	0.8018	0.7788	0**	0.5737	0.5672	0	0.0011	0.0014
1	0.8851	0.8783	1	0.9143	0.91049	0	0.0138	0.0119
0**	0.5135	0.5307	0	0.3528	0.3637	1**	0.0535	0.0578
1	0.9515	0.9450	1	0.6936	0.6823	0**	0.5626	0.5992
0**	0.5167	0.4447	0**	0.7467	0.7073	1	0.7853	0.7923
1	0.9383	0.9356	0	0.0618	0.0659	1**	0.1942	0.2135
1	0.8098	0.7982	0	0.0002	0.0002	0	0.0926	0.1061

Se puede ver que los resultados obtenidos con M-H son parecidos a los obtenidos con S-plus, esto debido a que el análisis Bayesiano (realizado hasta el momento) no incorpora más información inicial que la contenida en los datos de la Tabla 1 (Apéndice A). También hay que notar que con el ajuste anterior se tienen 12 pacientes mal clasificados, los cuales son marcados con (**) en la Tabla 3.6.

3.2.2 El modelo Semiparamétrico

La especificación del modelo semiparamétrico se realizó asumiendo valores de

$$\rho^2 = 3.0, \quad \lambda = 0.37 \quad y$$

$$\mu_\beta^* = \mathbf{h}(x_*)' \beta = \beta_1 + \beta_2 x_1 + \beta_3 x_2 + \beta_4 x_3 + \beta_5 x_4 + \beta_6 x_5 + \beta_7 x_6.$$

Con la especificación anterior se corrió el programa M1.F considerando muestras simuladas de tamaño $IM = 1000$ por un periodo de 1000 iteraciones y se obtuvo una muestra de la distribución final $p(\gamma|y), \{\gamma^{(1)}, \gamma^{(2)}, \dots, \gamma^{(IM)}\}$ con $\gamma^{(i)} \in \mathbf{R}^{7+51}, \forall i = 1, 2, \dots, IM$, correspondiente a las corridas del algoritmo de M-H (caminata aleatoria).

En la Tabla 3.7 se muestra el ajuste obtenido con el modelo semiparamétrico.

Tabla 3.7. *Ajuste del modelo semiparamétrico. Devianza=18.4952*

Resp.	M-H	Resp.	M-H	Resp.	M-H
1	0.707203362	0	0.120002898	0	0.065511244
1	0.846114716	1	0.9476202	0	0.371197208
1	0.974881546	1	0.621563626	0	0.003294286
1	0.982000092	0	0.043846909	0	0.033101981
1	0.936443307	0	0.00291224	0	0.032778988
0	0.143497487	0	0.012556306	1	0.98637761
1	0.99935228	0	0.153951557	0	0.001198876
1	0.8893024	1	0.977025649	0	0.139225737
1	0.709591155	1	0.828283559	0	0.012064559
0	0.344583312	0	0.298549508	1	0.724256734
1	0.862679103	0	0.319672427	0	0.000987743
1	0.923279188	1	0.935604892	0	0.009009874
0	0.282644602	0	0.210552434	1*	0.353179251
1	0.968782406	1	0.828361983	0	0.329454974
0	0.323306006	0	0.482480736	1	0.880785054
1	0.956650425	0	0.052901616	1	0.523495339
1	0.886659601	0	0.000185707	0	0.070322167

Hay que notar que el ajuste obtenido con el modelo semiparamétrico es, por mucho, mejor que el obtenido con S-plus. Mientras que con el ajuste de S-plus se tienen 13 malas clasificaciones, con el ajuste del modelo semiparamétrico se puede lograr tener tan sólo una mala clasificación, correspondiente al paciente

50 (observación marcada con *, en la Tabla 3.7). Sin embargo, si se realiza un análisis usual clásico se puede ver que la observación 50 está asociada a un posible dato atípico (ver Everitt, 1996).

3.2.3 Comparación de Modelos

Para el estudio de leucemia supóngase que se desean comparar los 64 modelos obtenidos al considerar todas las posibles regresiones.

Para el cálculo de las utilidades esperadas finales de cada modelo se consideraron las siguientes especificaciones:

$$x_* = \bar{x} = (1, 49.86275, 65.9418, 58.19608, 9.803922, 9.368627, 996.1373)'$$

$$\rho^2 = 3.0, \quad \lambda = 0.37 \quad y$$

$$\mu_{\beta}^* = \mathbf{h}(x_*)'\beta = \bar{x}'\beta$$

La Tabla 3.8 muestra las utilidades esperadas finales para cada uno de los posibles modelos. Se puede observar que el modelo que contiene las covariables x_1, x_4 , y x_6 es el mejor modelo. Además, el resto de los modelos se pueden comparar entre sí, de acuerdo a su utilidad esperada final.

Tabla 3.8. *Utilidades esperadas finales, $\bar{u}(M_i)$.*

Modelo	$\bar{u}(M_i)$	Modelo	$\bar{u}(M_i)$	Modelo	$\bar{u}(M_i)$
x_1	-22.98*	$x_1x_2x_3$	-22.94	$x_1x_2x_3x_5$	-23.00
x_2	-23.03	$x_1x_2x_4$	-22.94	$x_1x_2x_3x_6$	-22.87
x_3	-23.01	$x_1x_2x_5$	-22.97	$x_1x_3x_4x_5$	-22.89
x_4	-23.13	$x_1x_2x_6$	-22.86	$x_1x_3x_4x_6$	-22.57
x_5	-23.23	$x_1x_3x_4$	-22.94	$x_1x_3x_5x_6$	-22.81
x_6	-22.94*	$x_1x_3x_5$	-22.94	$x_1x_4x_5x_2$	-22.92
x_1x_2	-22.94	$x_1x_3x_6$	-22.86	$x_1x_4x_5x_6$	-22.35*
x_1x_3	-22.92	$x_1x_4x_5$	-23.03	$x_2x_3x_4x_5$	-23.00
x_1x_4	-22.99	$x_1x_4x_6$	-22.34*	$x_2x_3x_4x_6$	-22.37
x_1x_5	-23.04	$x_1x_5x_6$	-22.86	$x_2x_3x_5x_6$	-22.98
x_1x_6	-22.90	$x_2x_3x_4$	-23.09	$x_2x_4x_5x_6$	-22.42
x_2x_3	-23.01	$x_2x_3x_5$	-23.10	$x_3x_4x_5x_6$	-22.39
x_2x_4	-23.10	$x_2x_3x_6$	-22.92	$x_4x_5x_2x_6$	-22.42
x_2x_5	-23.13	$x_2x_4x_5$	-23.04	$x_1x_2x_3x_4x_5$	-22.92
x_2x_6	-22.92	$x_2x_4x_6$	-22.4	$x_1x_2x_3x_4x_6$	-22.58
x_3x_4	-23.09	$x_2x_5x_6$	-22.9	$x_1x_2x_3x_5x_6$	-22.96
x_3x_5	-23.10	$x_3x_4x_5$	-23.00	$x_1x_2x_4x_5x_6$	-22.46
x_3x_6	-22.92	$x_3x_4x_6$	-22.36	$x_1x_3x_4x_5x_6$	-22.53
x_4x_5	-23.16	$x_3x_5x_6$	-22.94	$x_2x_3x_4x_5x_6$	-22.40
x_4x_6	-22.54*	$x_4x_5x_6$	-22.53	$x_1x_2x_3x_4x_5x_6$	-22.53
x_5x_6	-22.95	$x_1x_2x_3x_4$	-22.97		

3.3 Ejemplo 3

Considere el siguiente modelo lineal generalizado

$$\log(\mu) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 \quad (3.5)$$

donde la variable de respuesta Y tiene una distribución Poisson con media μ . Las observaciones de las covariables X se presentan en la Tabla 3.9, y fueron tomadas de Qian, *et. al.* (1996). Los valores de Y se simularon de una distribución Poisson con media la especificada por el modelo (3.5) y parámetro $\beta \in \mathbf{R}^4$ definido por $\beta' = (\beta_0, \beta_1, \beta_2, \beta_3) = (2, 1, 0.5, 0.35)$.

Tabla 3.9. *Tabla de covariables*

Y	x_1	x_2	x_3	Y	x_1	x_2	x_3
12	0.412	0.284	0.97	352	1.15	1.045	0.906
23	0.805	0.296	1.082	41	0.982	0.948	0.221
18	0.485	0.23	0.743	37	0.734	1.042	0.711
18	0.235	0.173	1.038	26	0.628	0.924	0.833
14	0.224	0.16	0.796	39	0.944	0.985	0.731
6	0.31	0.136	0.832	26	0.562	0.938	0.36
5	0.262	0.285	0.387	28	0.993	0.51	0.469
12	0.51	0.103	0.436	19	0.784	0.577	0.404
22	0.614	0.208	0.88	32	0.754	0.559	1.031
10	0.453	0.204	0.224	28	0.964	0.53	0.742
11	0.095	0.197	0.67	22	0.729	0.502	0.711
28	0.841	0.533	0.531	37	1.13	0.559	0.92
27	0.484	0.885	0.83	34	0.896	0.515	1.169
11	0.249	0.969	0.108	30	0.672	0.54	1.032
17	0.443	0.949	0.363	44	1.484	0.51	0.358
28	0.594	0.948	1.195	34	0.931	0.533	0.355
37	0.541	0.959	1.147	10	0.566	0.5	0.653
23	0.464	1.003	0.564	36	0.841	0.533	0.531

Supóngase que se desean comparar los 7 modelos asociados. Para el cálculo de las utilidades esperadas finales de cada modelo se consideraron las siguientes

especificaciones:

$$x_* = \bar{x} = (1, 0.6601389, 0.5770, 0.69258330)'$$

$$\rho^2 = 2.8, \quad \lambda = 0.37 \quad y$$

$$\mu_\beta^* = \mathbf{h}(x_*)'\beta = \bar{x}'\beta$$

Procediendo de manera similar que en los ejemplos anteriores, se corrieron los programas M ϕ .1, M1.F y Predic.F. Con ayuda de S-plus se calcularon las utilidades esperadas finales para cada modelo las cuales se presentan en la Tabla 3.10.

A partir de la Tabla 3.10 se puede decir que la mejor selección, en términos predictivos, es elegir el modelo que contiene las tres covariables x_1 , x_2 , y x_3 . Esto era de esperarse ya que los datos se simularon directamente del modelo (3.5) con $\beta \in \mathbf{R}^4$ definido por $\beta' = (\beta_0, \beta_1, \beta_2, \beta_3) = (2, 1, 0.5, 0.35)$.

Tabla 3.10. *Utilidades esperadas finales.*

Modelo	$\bar{u}(M_i)$
x_1	-157.03
x_2	-155.33
x_3	-153.59
x_1x_2	-158.59
x_1x_3	-157.67
x_2x_3	-155.72
$x_1x_2x_3$	-153.58

Capítulo 4

Conclusiones

Las técnicas de regresión usuales proponen alguna forma paramétrica para el predictor y proceden a analizar el modelo resultante como si éste fuera el verdadero, sin tomar en cuenta la discrepancia entre la forma real y el predictor paramétrico que consideran. Uno de los objetivos de este trabajo era describir las técnicas de regresión usuales en el contexto de los llamados modelos lineales generalizados y analizar la discrepancia mencionada anteriormente. A este respecto, en el Capítulo 1 se especificó un modelo alternativo semiparamétrico en el caso de regresión normal y en el Capítulo 2 se extendió su análisis al caso de los modelos lineales generalizados.

Las inferencias clásicas para modelos lineales generalizados se basan en la estimación de los parámetros y las propiedades distribucionales de los estimadores. Los modelos de la familia exponencial poseen ciertas propiedades de convexidad que en muchos casos garantiza la existencia y unicidad de los estimadores de máxima verosimilitud. Sin embargo, en general la maximización de la verosimilitud requiere de métodos numéricos. Por otro lado, los métodos de Monte Carlo vía Cadenas de Markov proporcionan una forma relativamente sencilla de hacer inferencias Bayesianas para una clase amplia de modelos, incluyendo los lineales generalizados. En el Capítulo 1 de este trabajo se revisaron dichos métodos y, en particular, en el Capítulo 2 se analizó la implementación del algoritmo de Metropolis-Hastings.

En el Capítulo 3 se ilustró el análisis del modelo semiparamétrico y se pudo

observar que los ajustes fueron, por mucho, mejores que los obtenidos en S-plus. En este punto hay que valorar y notar que para el análisis del modelo semiparamétrico fue necesario implementar el algoritmo de Metropolis-Hastings, en el que se simularon varias cadenas de Markov de parámetros de altas dimensiones. En términos prácticos esto significa el uso de grandes recursos computacionales, lo que en principio no debería de presentar ningún problema debido al avance actual de estos recursos.

Otro de los objetivos de este trabajo era proponer un enfoque predictivo, basado en el modelo semiparamétrico, para la selección de modelos lineales generalizados. Para esto, en el Capítulo 1 se presentó el problema de comparación de modelos y se discutieron algunas soluciones que existen en la literatura, esencialmente basadas en el análisis de los Factores de Bayes y considerando el llamado enfoque \mathcal{M} -cerrado. También se mencionaron las limitaciones de tales soluciones. En este trabajo se analizó el problema de comparación de modelos desde un punto de vista Bayesiano usando la teoría de decisiones, considerando funciones de utilidad basadas en las distribuciones predictivas correspondientes. Con respecto a este punto, se pudo observar que el considerar un enfoque predictivo semiparamétrico usando la teoría de decisiones permite, por un lado, proponer una clase más amplia de modelos en la comparación, y por otro, evitar las restricciones de los Factores de Bayes y/o el asumir un enfoque \mathcal{M} -cerrado. En particular, el enfoque propuesto permite el uso de distribuciones iniciales impropias para los parámetros de los modelos bajo comparación.

Como puede observarse, en este trabajo solamente se consideró la especificación del modelo semiparamétrico con valores conocidos de los parámetros de 'suavizamiento', ρ^2 y θ , y del parámetro de dispersión, σ^2 . Resta abordar el caso más general en el cual estos parámetros son parcial o totalmente desconocidos. A este respecto, cabe mencionar que en Gutiérrez-Peña y Smith (1998) se hace un análisis del modelo de regresión generalizado, usando el muestreo de Gibbs, donde se consideran desconocidos los parámetros antes mencionados. La importancia de desarrollar un trabajo de esta naturaleza radica en el hecho de que en la práctica usualmente se desconocen estos parámetros. En este sentido,

la combinación de otras técnicas de Monte Carlo vía Cadenas de Markov con el algoritmo básico de Metropolis-Hastings utilizado en este trabajo, permitiría una simulación más eficiente.

Otras alternativas para seguir investigando, relacionadas con el tema de esta tesis, incluyen el análisis y comparación de modelos para datos categóricos.

Los desarrollos presentados en este trabajo sólo consideran modelos lineales generalizados. Por lo tanto, surge de manera natural la posibilidad de trabajar con modelos no lineales, los cuales son aplicados, por ejemplo, en el ámbito actuarial o en la investigación farmaco-cinética.

Estos son solamente algunos temas de investigación que se podrían abordar a partir del análisis Bayesiano de modelos lineales generalizados presentado en este trabajo. Sin embargo, en el análisis de cualquier modelo estadístico siempre existirán nuevas líneas de investigación, dependiendo del área en la que se trabaje.

Apéndice A

Tabla 1. *Datos de 51 pacientes con cierto tipo de leucemia*

Respuesta	x_1	x_2	x_3	x_4	x_5	x_6
1.0	20.0	78.0	39.0	7.0	0.6	990.0
1.0	25.0	64.0	61.0	16.0	35.0	1030.0
1.0	26.0	61.0	55.0	12.0	7.5	982.0
1.0	26.0	64.0	64.0	16.0	21.0	1000.0
1.0	27.0	95.0	95.0	6.0	7.5	980.0
0.0	27.0	80.0	64.0	8.0	0.6	1010.0
1.0	28.0	88.0	88.0	20.0	4.8	986.0
1.0	28.0	70.0	70.0	14.0	10.0	1010.0
1.0	31.0	72.0	72.0	5.0	2.3	988.0
0.0	33.0	58.0	58.0	7.0	5.7	986.0
1.0	33.0	92.0	92.0	5.0	2.6	980.0
1.0	33.0	42.0	38.0	12.0	2.5	984.0
0.0	34.0	26.0	26.0	7.0	7.0	982.0
1.0	36.0	55.0	55.0	14.0	4.5	986.0
0.0	37.0	71.0	71.0	15.0	4.4	1020.0
1.0	40.0	91.0	91.0	9.0	35.0	986.0
1.0	40.0	52.0	49.0	12.0	2.1	988.0
0.0	43.0	74.0	63.0	4.0	0.1	986.0
1.0	45.0	78.0	47.0	14.0	4.2	980.0
1.0	45.0	60.0	36.0	10.0	0.6	992.0
0.0	45.0	82.0	32.0	10.0	28.1	1016.0
0.0	45.0	79.0	79.0	4.0	1.1	1030.0
0.0	47.0	56.0	28.0	2.0	0.9	990.0
0.0	48.0	60.0	54.0	10.0	2.2	1002.0

Respuesta	x_1	x_2	x_3	x_4	x_5	x_6
1.0	50.0	83.0	66.0	19.0	11.6	996.0
1.0	50.0	36.0	32.0	14.0	4.5	992.0
0.0	51.0	88.0	70.0	8.0	0.5	982.0
0.0	52.0	87.0	87.0	7.0	10.3	986.0
1.0	53.0	75.0	68.0	13.0	2.3	980.0
0.0	53.0	65.0	65.0	6.0	2.3	982.0
1.0	56.0	97.0	92.0	10.0	16.0	992.0
0.0	57.0	87.0	83.0	19.0	21.6	1020.0
0.0	59.0	45.0	45.0	8.0	1.1	999.0
0.0	59.0	36.0	34.0	5.0	0.0	1038.0
0.0	60.0	39.0	33.0	7.0	0.9	988.0
0.0	60.0	76.0	53.0	12.0	0.4	982.0
0.0	61.0	46.0	37.0	4.0	1.4	1006.0
0.0	61.0	39.0	8.0	8.0	0.3	990.0
0.0	61.0	90.0	90.0	1.0	9.9	990.0
1.0	62.0	84.0	84.0	19.0	115.0	1020.0
0.0	63.0	42.0	27.0	5.0	0.3	1014.0
0.0	65.0	75.0	75.0	10.0	20.0	1004.0
0.0	71.0	44.0	22.0	6.0	0.3	990.0
1.0	71.0	63.0	63.0	11.0	10.0	986.0
0.0	73.0	33.0	33.0	4.0	0.5	1010.0
0.0	73.0	93.0	84.0	6.0	38.00	1020.0
1.0	74.0	58.0	58.0	10.0	2.4	1002.0
0.0	74.0	32.0	30.0	16.0	6.7	988.0
1.0	75.0	60.0	60.0	17.0	8.2	990.0
1.0	77.0	69.0	69.0	9.0	1.5	986.0
0.0	80.0	73.0	73.0	7.0	1.5	986.0

Tabla 1. *Continuación ...*

Apéndice B

Programas en S-plus.

Como una forma de ejemplificar los programas en S-plus, usados para el cálculo de las utilidades esperadas, se presentan los correspondientes al ejemplo 1.

```
.....  
#Programa ejemplo1.plu  
# Ajuste de un modelo logístico para los datos de diabetes-retinopatía  
ejemplo1.dat_read.table("a:/base1.prn",col.names=c("res","ps","inter","x_1","x_2"))  
ejemplo1.fit_glm(res~ x_1+x_2,data=ejemplo1.dat,family=binomial)  
.....  
#Programa modelo0.txt  
#Este programa calcula la utilidad esperada final asociada al modelo 0  
IM_1000  
# tamaño de muestra simulado  
cov0_c(1,1,1,1,1,1,1)  
ajuste0_glim(cov0, exitos, ensayos, error="binomial", link="logit", intercept=F)  
#ajuste en S-plus del modelo 0  
mu.modelo_exp(ajuste0$coef)/(1.0+exp(ajuste0$coef) )  
ynui_scan("d:/yui.ca")  
mui_exp(ynui.ca)/(1.0 + exp(ynui.ca) )  
# obtencion de las IM mui simuladas asociadas al punto xbarra  
yi.star_rbinom(1000,91,mui)  
storage.mode(yi.star)_"integer"  
predictiva_dbinom(yi.star,91,mu.modelo)
```

```

u_log(predictiva)
u0_mean(u)
.....

#Programa modelo1.txt
#Este programa calcula la utilidad esperada final asociada al modelo 1
#IM_1000
#tamaño de muestra simulado
ajuste1_glm(res~ x_1,data=ejemplo1.dat,family=binomial)
#ajuste en S-plus del modelo 1
xbarra_c(1.0, 11.75)
nu.modelo_xbarra%*%ajuste1$coefficients
mu.modelo_exp(nu.modelo)/(1.0+exp(nu.modelo))
#ynui_scan("d:/yui.ca")
#mui_exp(ynui.ca)/(1.0 + exp(ynui.ca) )
# obtencion de las IM mui simuladas asociadas al punto xbarra
#yi.star_rbinom(1000,91,mui)
#storage.mode(yi.star)-"integer"
predictiva_dbinom(yi.star,91,mu.modelo)
u_log(predictiva)
u1_mean(u)
.....

#Programa modelo2.txt
#Este programa calcula la utilidad esperada final asociada al modelo 2
#IM_1000
#tamaño de muestra simulado
ajuste2_glm(res~x_1+ x_2,data=ejemplo1.dat,family=binomial)
#ajuste en S-plus del modelo 2
xbarra_c(1.0, 11.75,138.0625)
nu.modelo_xbarra%*%ajuste2$coefficients
mu.modelo_exp(nu.modelo)/(1.0+exp(nu.modelo))
#ynui_scan("d:/yui.ca")

```

```

#mui_exp(ynui.ca)/(1.0 + exp(ynui.ca) )
# obtencion de las IM mui simuladas asociadas al punto xbarra
#yi.star_rbinom(1000,91,mui)
#storage.mode(yi.star)_"integer"
predictiva_dbinom(yi.star,91,mu.modelo)
u_log(predictiva)
u2_mean(u)
.....

#Programa modelo3.txt
#Este programa calcula la utilidad esperada final asociada al modelo 3
#IM_1000
#tamaño de muestra simulado
ajuste3_glm(res~ x_1+x_2+x_3,data=ejemplo1.dat,family=binomial)
#ajuste en S-plus del modelo 3
xbarra_c(1.0, 11.75,138.0625,1622.234375)
nu.modelo_xbarra%*%ajuste3$coefficients
mu.modelo_exp(nu.modelo)/(1.0+exp(nu.modelo))
#ynui_scan("d:/yui.ca")
#mui_exp(ynui.ca)/(1.0 + exp(ynui.ca) )
# obtencion de las IM mui simuladas asociadas al punto xbarra
#yi.star_rbinom(1000,91,mui)
#storage.mode(yi.star)_"integer"
predictiva_dbinom(yi.star,91,mu.modelo)
u_log(predictiva)
u3_mean(u)
.....

#Programa modelo4.txt
#Este programa calcula la utilidad esperada final asociada al modelo 4
#IM_1000
#tamaño de muestra simulado
ajuste4_glm(res~ x_1+x_2+x_3+x_4,data=ejemplo1.dat,family=binomial)

```

```

#ajuste en S-plus del modelo 4
xbarra_c(1.0, 11.75,138.0625,1622.234375,19061.25391)
nu.modelo_xbarra%%ajuste4$coefficients
mu.modelo_exp(nu.modelo)/(1.0+exp(nu.modelo))
#ynui_scan("d:/yui.ca")
#mui_exp(ynui.ca)/(1.0 + exp(ynui.ca) )
# obtencion de las IM mui simuladas asociadas al punto xbarra
#yi.star_rbinom(1000,91,mui)
#storage.mode(yi.star)_"integer"
predictiva_dbinom(yi.star,91,mu.modelo)
u_log(predictiva)
u4_mean(u)
.....

```

Programas en Fortran.

Breve descripción (en cuanto al manejo de archivos)

de los programas M ϕ .F, M1.F, M2.F y Predic.F

El programas M ϕ .F genera los archivos val.0, predic.dat y FSAL(val.ini), para esto requiere de los archivos ins.0, parametros y FDAT(base.dat)

El programa M1.F (M2.F) genera los archivos pre_dev.ca (pre_dev.ind) y bbeta.ca (bbeta.ind), y FSAL(medias.erg) para esto necesita de los archivos val.0, FDAT(base.dat), FVALINI(val.ini), parametros y ca.ins (ind.ins).

El programa Predic.F produce los archivos ban.ca (ban.ind) y ynui.ca (ynui.ind), para lo cual necesita de los archivos predic.dat y bbeta.ca (bbeta.ind). Este programa requiere de algunas especificaciones iniciales que se deben de hacer dentro del mismo.

Breve descripción de archivos

·El archivo val.0 contiene el vector VB0, de orden $p+n$, y la matriz cuadrada de B0INV de $p+n$. Estos elementos especifican, al vector de medias y a la matriz de varianza-covarianza, respectivamente, de la distribución inicial del vector γ , asociado al modelo semiparamétrico.

·El archivo predic.dat contiene la matriz V (de $n \times n$), la matriz de covariables X (de $n \times p$) y el vector WINV (de $p \times 1$).

·El archivo FSAL(val.ini) contiene al vector VBETAF, de orden $p+n$, y a la matriz cuadrada B1INV de $p+n$. Estos elementos determinan, respectivamente, al vector de medias y a la matriz de varianza-covarianza los cuales se usan como valores iniciales en el algoritmo de Metropolis-Hastings.

·El archivo FDAT(base.dat) es un archivo de 'lectura de información' que contiene los siguientes datos: el vector de respuesta $Y(n \times 1)$, vector de pesos $VM(n \times 1)$ y matriz de covariables X.

·El archivo parametros, también es un archivo de 'lectura de información' y contiene los valores de ρ^2 y θ .

·El archivo ins.0 es otro archivo de 'lectura de información' donde se especifica lo siguiente: nombre del archivo que contiene a los vectores Y, VM y a la matriz X (en este caso base.dat), nombre del archivo FSAL (en este caso val.ini), dimensión (p) del vector de parámetros de regresión β , dimensión (n) del vector de respuesta Y, valor de la variable s2 que determina al parámetro de dispersión σ^2 , valor de la variable kk que determina el modelo de la familia exponencial con el que se trabaja, valor de la variable ite que especifica el número de iteraciones deseables del programa M ϕ .F, un vector delxp y una matriz de p \times p que son el vector de medias y la matriz de varianza-covarianza de la distribución inicial del vector β .

·El archivo pre_dev.ca es un archivo de salida que contiene el vector de respuesta Y, el vector de valores ajustados \hat{Y} y el valor de la devianza obtenida.

·El archivo bbeta.ca es otro archivo de salida que contiene la muestra final, de tamaño im, de vectores γ , obtenida a través de Metropolis-Hastings.

·El archivo ca.ins es un archivo de 'lectura de información' que contiene: nombre del archivo FDAT, nombre del archivo FVALINI(val.ini), nombre del archivo FSAL (en este caso medias.erg) donde se guardan las medias ergódicas, valor de los parámetro p, n, s2, kk, im y k. Aquí, el valor de k representa el número de iteraciones que se desee correr el algoritmo de M-H (programa M1.F).

·El archivo ban.ca contiene es un archivo de salida que contiene las predicciones de la variable de respuesta y las bandas correspondientes, asociadas con nuevos valores de las covariables.

·El archivo ynui.ca contiene los valores de $\eta(x_*)$, asociados a cada vector γ generado por M-H (archivo bbeta.ca).

```

C PROGRAM M01.F
C
C PROGRAMA QUE OBTIENE: EL VECTOR MODA Y LA MATRIZ DE PRECISION,
C INICIALES, PARA EL MODELO SEMIPARAMETRICO.
C
C ESTE PROGRAMA CALCULA PARA LAS FAMILIAS NORMAL, GAMMA, POISSON
C Y BINOMIAL, UNA APROXIMACION DE LA E(BETA/Y) Y DE LA V(BETA/Y)
C RESPECTIVAMENTE, PARA PODER USARSE COMO VALORES INICIALES EN EL
C ALGORITMO DE METROPOLIS-HASTINGS.

C ULTIMA ACTUALIZACION JUNIO DE 1998
C
C NOTA: ESTE PROGRAMA TIENE INDEPENDENCIA DE LAS SUBROUTINAS DE IMSL

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
INTEGER P
CHARACTER*12 FDAT, FSAL
C
C PARAMETER(ND=100,NP=150)
C
C DIMENSION Y(ND),VTETAG(ND),VAP(ND),VAIIP(ND),A(ND),VM(ND)
C DIMENSION B0(NP,NP),VB0(NP),B0VB0(NP),X(ND,NP),XT(NP,ND)
C DIMENSION XTA(NP,ND),XTAZ(NP),Z(ND),B2(NP),XTAX(NP,NP)
C DIMENSION B1(NP,NP),B1INV(NP,NP),VBETA F(NP)
C DIMENSION VAP2(ND),VAIIP2(ND),A2(ND),B0INV(NP,NP)
C DIMENSION XTA2(NP,ND),XTA2Z(NP),XTA2X(NP,NP)

C DIMENSION XOR(ND,NP),RA(ND),XDIS(ND,NP),ZZ(NP)
C DIMENSION WINV(NP),V(ND,ND),VINV(ND,ND)
C
C DIMENSION INDX(NP),IINDX(ND)
C
C***** BANDERA PARA EL INTERCEPTO *****
INTER=1
C*****
OPEN(11,FILE='parametros',STATUS='OLD')
READ(11,*) RHO2
READ(11,*) TETHA
CLOSE(11)
C***** LIMPIANDO MATRICES *****
C
DO 12 I=1,NP
DO 13 J=1,NP
B0INV(I,J)=0.0D0
13 CONTINUE
12 CONTINUE
DO 14 I=1,ND
DO 15 J=1,NP
X(I,J)=0.0D0
15 CONTINUE
14 CONTINUE
C
C*****
OPEN(9,FILE='ins.0',STATUS='OLD')
READ(9,(A)) FDAT
READ(9,(A)) FSAL
READ(9,*) P
READ(9,*) N
READ(9,*) S2
READ(9,*) KK
READ(9,*) ITE
WRITE(*,*) P, N, S2, KK, ITE
C
READ(9,*) (VB0(J), J=1,P)
C WRITE(*,*) (VB0(j), J=1,P)

```

```

DO 1100 I=1,P
READ(9,*) (B0INV(I,J), J=1,P)
WRITE(*,*) (B0INV(I,J), J=1,P)
1100 CONTINUE
CLOSE(9)
write(*,*)'*****'
call pprint(b0inv,(n+p),(n+p),np,np)

OPEN(8,FILE=FDAT,STATUS='OLD')
C
C CALL DLINDS(P,B0INV,NP,B0,NP)
C
if(kk.ne.4) then
DO 1200 I=1,N
READ(8,*) Y(I), VM(I), (X(I,J), J=1,P)
1200 CONTINUE
else
DO 1300 I=1,N
READ(8,*) Y(I), VM(I), (X(I,J), J=1,P)
Y(I) = Y(I) / VM(I)
C WRITE(*,*) Y(I), VM(I), (X(I,J), J=1,P)
1300 CONTINUE
endif
C
CLOSE(8)

C*****
IR=P-INTER
C*****
C***** ORDENA LA MATRIZ X Y LA DEPOSITA EN *****
C EN LA MATRIZ XOR

DO 16 J=(INTER+1),P

DO 17 I=1,N
RA(I)=X(I,J)
17 CONTINUE

C CALL DSVRGN(N,RA,RB)
CALL SHELL(N,RA)

DO 18 I=1,N
XOR(I,J)=RA(I)
18 CONTINUE

16 CONTINUE
C***** OBTIENE LA MATRIX DE DISTANCIAS XDIS *****
CC DE N-1 X IR
DO 19 J=(1+INTER),P
DO 20 I=1,(N-1)
XDIS(I,J)= XOR(I+1,J) - XOR(I,J)
20 CONTINUE
19 CONTINUE
C***** CALCULO DE LA MATRIZ V(X,X) *****

DO 21 J=(1+INTER),P
AUX=0.0D0
DO 22 I=1,(N-1)
AUX= XDIS(I,J) + AUX
22 CONTINUE
ZZ(J)=AUX/DBLE(N-1)
WINV(J) = 1.0D0/(ZZ(J)**2)
21 CONTINUE

```

```

DO 101 J=(1+INTER),P
WRITE(*,*) 'ZZ(:,J)='&ZZ(J),'WINV(:,J)='&WINV(J)
101 CONTINUE

CALL MATRXX_V(V,TETHA,X,WINV,P,N,NP,ND,INTER)

write(*,*) ''

C***** SALIDAS PARA EL PROGRAMA PREDIC.F *****

OPEN(5,FILE='predic.dat',STATUS='UNKNOWN')
DO 32 I=1,N
WRITE(5,*)X(V(I,J), J=1,N)
32 CONTINUE
DO 33 I=1,N
WRITE(5,*)X(X(I,J), J=1,P)
33 CONTINUE
WRITE(5,*)X(WINV(J), J=1,P)
CLOSE(5)
C
C***** INCLUYENDO RHO2 *****
C call dwrrm(matriz v,n,n,v,nd,0)

DO 30 I=1,N
DO 31 J=1,N
V(I,J)=RHO2*V(I,J)
31 CONTINUE
30 CONTINUE

C*****
C CALL DWRRRN('MATRIZ V',N,N,V,ND,0)
C CALL DLINDS(N,V,ND,VINV,ND)
CALL INVERSA(VINV,V,IINDX,N,ND)
write(*,*)V
call pprint(v,n,n,nd,nd)
C CALL DWRRRN('MATRIZ VINV',N,N,VINV,ND,0)
write(*,*)B0INV
call pprint(b0inv,n+p,n+p,np,np)

C***** ESPECIFICANDO ELEMENTOS VB0, B0INV Y MATRIZ X *****
C***** DEL MODELO SEMIPARAMETRICO *****
DO 23 I=(P+1),(N+P)
VB0(I)=0.0D0
23 CONTINUE

DO 24 I=(P+1),(P+N)
II=I-P
DO 25 J=(P+1),(P+N)
JJ=J-P
B0INV(I,J)=V(II,JJ)
25 CONTINUE
24 CONTINUE

WRITE(*,*) 'MODELO SEMIPARAMETRICO'
WRITE(*,*) ''
WRITE(*,*) 'VECTOR VB0'
WRITE(*,*)X(VB0(J), J=1,N+P)

write(*,*)B0INV TOTAL'
call pprint(b0inv,n+p,n+p,np,np)

C CALL DWRRRN('MATRIZ B0INV',N+P,N+P,B0INV,NP,0)

```

```

DO 26 I=(P+1),(P+N)
  J=I-P
  X(J,I)=1.0D0
26 CONTINUE

C   CALL DWRRRN('NUEVA MATRIZ X',N,(P+N),X,ND,0)

C   CALL DLINDS(P+N,B0INV,NP,B0,NP)
CALL INVERSA(B0,B0INV,INDX,P+N,NP)
C   CALL DWRRRN('MATRIZ B0',(P+N),(P+N),B0,NP,0)
WRITE(*,*)'MATRIZ B0'
CALL PPRINT(B0,(P+N),(P+N),NP,NP)
C*****
IF(KK.EQ.1) THEN
  CALL GORRO_N(VTETAG,VAP,VAIP,A,Y,S2,N,ND,VM)
ENDIF
IF(KK.EQ.2) THEN
  CALL GORRO_G(VTETAG,VAP,VAIP,A,Y,S2,N,ND,VM)
ENDIF
IF(KK.EQ.3) THEN
  CALL GORRO_P(VTETAG,VAP,VAIP,A,Y,S2,N,ND,VM)
ENDIF
IF(KK.EQ.4) THEN
  CALL GORRO_B(VTETAG,VAP,VAIP,A,Y,S2,N,ND,VM)
ENDIF

C   WRITE(*,*)'VECTOR VTETAG'
WRITE(*,*)VTETAG(I), I=1,N

DO 1 I=1,N
  Z(I) = VTETAG(I) + (Y(I) - VAP(I))/VAIP(I)
1 CONTINUE

C   CALL MULTMV(B0VB0,B0,VB0,(P+N),(P+N),NP,NP)
write(*,*)'aqui0'
CALL TRANSP(XT,X,N,(P+N),ND,NP)
write(*,*)'aqui1'
CALL MULMMDIA(XTA,XT,A,(P+N),N,NP,ND)
write(*,*)'aqui2'
CALL MULTMV(XTAZ,XTA,Z,(P+N),N,NP,ND)
write(*,*)'aqui3'

C   DO 2 J=1,(P+N)
B2(J) = (XTAZ(J)/S2) + B0VB0(J)
2 CONTINUE
write(*,*)'aqui4'
CALL MULTMM(XTAX,XTA,X,(P+N),N,(P+N),NP,ND,NP)

c   call pprint(xtax,(p+n),(p+n),np,np)

DO 3 I=1,(P+N)
DO 4 J=1,(P+N)
  B1(I,J) = (XTAX(I,J)/S2) + B0(I,J)
4 CONTINUE
3 CONTINUE

write(*,*)'aqui5'
call pprint(b1,(n+p),(n+p),np,np)

C
C   CALL DLINDS((P+N),B1,NP,B1INV,NP)
write(*,*)'aqui6'
CALL INVERSA(B1INV,B1,INDX,P+N,NP)
write(*,*)'matriz b1inv'
call pprint(b1inv,(p+n),(p+n),np,np)
CALL MULTMV(VBETAF,B1INV,B2,(P+N),(P+N),NP,NP)

```

```

write(*,*)''
write(*,*)VECTOR BETA FINAL EN LA ITERACION 0'
do 28 j=1,(n+p)
  write(*,*) vbetaf(j)
28 continue
C call dwwrm('MATRIZ B1INV EN LA ITE 0',(n+p),(n+p),b1inv,np,0)
DO 50 K=1,ITE
C
  AUX1=0.0D0
  DO 5 I=1,N
    DO 6 J=1,(P+N)
      AUX1 = X(L,J)*VBETAF(J) + AUX1
  6 CONTINUE
  VTETAG(I) = AUX1
  AUX1=0.0D0
  5 CONTINUE
C
  IF(KK.EQ.1) THEN
    CALL GORRO2_N(VAP2,VAIP2,A2,N,ND,VM,VTETAG)
  ENDIF
  IF(KK.EQ.2) THEN
    CALL GORRO2_G(VAP2,VAIP2,A2,N,ND,VM,VTETAG)
  ENDIF
  IF(KK.EQ.3) THEN
    CALL GORRO2_P(VAP2,VAIP2,A2,N,ND,VM,VTETAG)
  ENDIF
  IF(KK.EQ.4) THEN
    CALL GORRO2_B(VAP2,VAIP2,A2,N,ND,VM,VTETAG)
  ENDIF
C
  DO 7 I=1,N
    Z(I) = VTETAG(I) + (Y(I) - VAP2(I))/VAIP2(I)
  7 CONTINUE
C
  CALL MULMMDIA(XTA2,XT,A2,(P+N),N,NP,ND)
  CALL MULTMV(XTA2Z,XTA2,Z,(P+N),N,NP,ND)
C
  DO 8 J=1,(P+N)
    B2(J) = (XTA2Z(J)/S2) + B0VB0(J)
  8 CONTINUE
C
  CALL MULTMM(XTA2X,XTA2,X,(P+N),N,(P+N),NP,ND,NP)
C
  DO 9 I=1,(P+N)
    DO 10 J=1,(P+N)
      B1(L,J) = (XTA2X(L,J)/S2) + B0(L,J)
  10 CONTINUE
  9 CONTINUE
C
  CALL DLINRG((P+N),B1,NP,B1INV,NP)
  CALL INVERSA(B1INV,B1,INDX,P+N,NP)
  CALL MULTMV(VBETAF,B1INV,B2,(P+N),(P+N),NP,NP)
C
  write(*,*)''
  write(*,*)VECTOR BETA FINAL EN LA ITERACION,K
  do 27 j=1,(n+p)
    write(*,*) vbetaf(j)
  27 continue
C call dwwrm('MATRIZ B1INV',(n+p),(n+p),b1inv,np,0)

50 CONTINUE

```

```

c*****
OPEN(7,FILE=FSAL,STATUS='UNKNOWN')
WRITE(7,*) (VBETA(J), J=1,(P+N))

DO 11 I=1,(P+N)
WRITE(7,*) (BINV(I,J), J=1,(P+N))
11 CONTINUE
C
CLOSE(7)

OPEN(6,FILE='val.0',STATUS='UNKNOWN')
WRITE(6,*)(VB0(J),J=1,(P+N))
DO 29 I=1,(P+N)
WRITE(6,*)(BINV(I,J), J=1,(P+N))
29 CONTINUE
CLOSE(6)
C
STOP
END

```

C PROGRAMA M1.F

C ESTE PROGRAMA REPRESENTA EL ALGORITMO 1 (CAMINATA ALEATORIA).
C PARA EL MODELO SEMIPARAMETRICO CONSIDERANDO UNA VEROSIMILITUD
C ASOCIADA A LA FAMILIA EXPONENCIAL.
C
C ULTIMA ACTUALIZACION JUNIO DE 1998

C MODIFICACIONES QUE INCLUYEN PROMEDIOS ERGODICOS Y
C UNA ETAPA DE CALENTAMIENTO DEL 10% DEL TOTAL DE ITERACIONES

C MODIFICACIONES QUE INCLUYEN EL CALCULO DE DOS ESTADISTICAS
C QUE CONFORMAN UNA ETAPA(1) DE MONITOREO.
C Y UNA ETAPA(2) DE AJUSTE QUE INCLUYE EL CALCULO DE LAS
C MU (PARA CADA OBSERVACION),
C Y LA DEVIANZA EN BASE A LA MUESTRA FINAL.
C
C NOTA: ESTE PROGRAMA NO DEPENDE YA, DE SUBROUTINAS DE IMSL

C234567

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
INTEGER P

CHARACTER*12 FDAT, FVALINI, FSAL

C NOTA: EL VALOR DEL PARAMETRO ND DEBE SER MAYOR AL TAMAÑO DE
C MUESTRA GENERADO, IM.
C
C NOTA: ND->DIMENSION DE LOS ARREGLOS PARA EL TAMAÑO DE MUESTRA N
C NM->DIMENSION PARA LOS ARRWGLOS CON TAMAÑOS DE MUESTRA
C GENERADOS IM.
C NP->DIMENSION DE LOS VECTORES DE PARAMETROS DE (P+N)
C

PARAMETER(ND=101,NP=151,NM=5001)
PARAMETER(INTER=1)

DIMENSION Y(ND),VM(ND),X(ND,NP),VBETAF(NP),BETAI(NP),AE(NP)
DIMENSION BBETA0(NM,NP),BETA0(NP),BETAC(NP),BETAF(NP),BETA(NP)
DIMENSION SUM1(NP),PROM(NP)

DIMENSION B1(NP,NP),B1INV(NP,NP),BBETAF(NM,NP)
DIMENSION VB0(NP),B0(NP,NP),B0INV(NP,NP),AEAUX(NP)

DIMENSION AANU(NM),SSDELTA(NM)

DIMENSION BBNU(ND),BBNNU(NM,ND),SUM2(ND),PROMNU(ND),HMU(ND)

DIMENSION INDX(NP)

DIMENSION SUM22(ND),PROMNU2(ND),HMUTNF(ND),HMUSUP(ND)
DIMENSION VARNU(ND)

C.....
C OPEN(15,FILE='parametros',STATUS='OLD')
C READ(15,*) RHO2
C READ(15,*) TETHA
C CLOSE(15)

C.....
C OPEN(1,FILE='ca.ins',STATUS='OLD')

READ(1,(A)) FDAT
READ(1,(A)) FVALINI
READ(1,(A)) FSAL
READ(1,*) P

```

READ(1,*) N
READ(1,*) S2
READ(1,*) KK
READ(1,*) IM
READ(1,*) K

CLOSE(1)
OPEN(5,FILE='val.0',STATUS='OLD')
  READ(5,*)XVB0(J), J=1,(P+N)
  DO 1 I=1,(P+N)
    READ(5,*)X(B0INV(I,J), J=1,(P+N))
1  CONTINUE
CLOSE(5)

c call dwrmm('matriz b0inv',p+n,p+n,b0inv,np,0)
C CALL DLINDS((P+N),B0INV,NP,B0,NP)
CALL INVERSA(B0,B0INV,INDX,P+N,NP)
c call dwrmm('matriz b0',p+n,p+n,b0,np,0)

OPEN(2,FILE=FDAT,STATUS='OLD')

  if(kk.ne.4) then
    DO 1200 I=1,N
      READ(2,*) Y(I), VM(I), (X(I,J), J=1,P)
1200 CONTINUE
    else
      DO 1300 I=1,N
        READ(2,*) Y(I), VM(I), (X(I,J), J=1,P)
        Y(I) = Y(I) / VM(I)
C WRITE(*,*) Y(I), VM(I), (X(I,J), J=1,P)
1300 CONTINUE
      endif

    CLOSE(2)
C
C***** CONSTRUCCION DE LA NUEVA MATRIZ X *****
C
  DO 2 I=(P+1),(P+N)
    J=I-P
    X(J,I)=1.0D0
2 CONTINUE

C*****
c call dwrmm('matriz x',n,n+p,x.nd,0)

OPEN(3,FILE='val.ini',STATUS='OLD')

  READ(3,*) (VBETAF(J), J=1,(P+N))
C WRITE(*,*) (VBETAF(J), J=1,(P+N))
C
  DO 1101 I=1,(P+N)
    READ(3,*) (B1INV(I,J), J=1,(P+N))
C WRITE(*,*) (B1INV(I,J), J=1,(P+N))
1101 CONTINUE

  CLOSE(3)

c call dwrmm('matriz b1inv',n+p,n+p,b1inv,np,0)
C CALL DLINDS((P+N),B1INV,NP,B1,NP)
CALL INVERSA(B1,B1INV,INDX,P+N,NP)
c CALL DWRRRN('MATRIZ DE PRECISION',(P+N),(P+N),B1,NP,0)

```

```

C WRITE(*,*)'ENTER SEEDS -> IX,IY,IZ (0<IX,IY,IZ<30000)'
C READ(*,*) IX,IY,IZ
  IX=265
  IY=65
  IZ=5

OPEN(4,FILE=FSAL,STATUS='UNKNOWN')
c OPEN(8,file='/tmpu/mbp/ca/sdelta.mon',status='unknown')
c OPEN(7,file='/tmpu/mbp/ca/nu.mon',status='unknown')

C***** APROXIMACION A LA MEDIA Y VARIANZA DE BETA/Y *****

  DO 10 J=1,(P+N)
    BETAI(J)=VBETAF(J)
  10 CONTINUE

C MATRIZ DE VAR-COV = INVERSA DE B1 = B1INV
C MATRIZ DE PRECISION = B1

C***** ERIODO DE CALENTAMIENTO (10% DEL NUMERO DE ITERACIONES) *****

  L=NINT(0.10*K)
  L0=L+1

C** SIMULA IM OBSERVACIONES BETA0 DE UNA NORMAL (P+N) VARIADA CON ***
C MEDIA = BETAI Y MATRIZ DE VAR-COV = INVERSA DE B1 = B1INV

  DO 11 M=1,IM
    CALL MULTNORM(BETA0,(P+N),NP,BETAI,B1INV,IX,IY,IZ)
    DO 12 J=1,(P+N)
      BBETA0(M,J)=BETA0(J)
    12 CONTINUE
  11 CONTINUE

  K=K+L

  DO 400 ITE=1,K

C***** CICLO QUE GENERA LAS IM OBSERVACIONES 'FINALES'
C***** (EN CADA ITERACION)

  DO 600 II=1,IM

C***** ASIGNA EL VECTOR INICIAL

  DO 13 J=1,(P+N)
    BETA0(J)=BBETA0(II,J)
  13 CONTINUE

C*** IMULA UNA VARIABLE BETAC DE UNA NORMAL (P+N) VARIADA(BETA0,B1INV)

  CALL MULTNORM(BETAC,(P+N),NP,BETA0,B1INV,IX,IY,IZ)

C***** CALCULO DE LA PROBABILIDAD DE ACEPTACION

  IF(KK.EQ.1) THEN
    CALL UH_N(UU,HH,BETAC,BETA0,VB0,B0,S2,VMLX,Y,(P+N),N,NP,ND)
  ENDIF

```

```

IF(KK.EQ.2) THEN
  CALL UH_G(UU,HH,BETAC,BETA0,VB0,B0,S2,VM,X,Y,(P+N),N,NP,ND)
ENDIF
IF(KK.EQ.3) THEN
  CALL UH_P(UU,HH,BETAC,BETA0,VB0,B0,S2,VM,X,Y,(P+N),N,NP,ND)
ENDIF
IF(KK.EQ.4) THEN
  CALL UH_B(UU,HH,BETAC,BETA0,VB0,B0,S2,VM,X,Y,(P+N),N,NP,ND)
ENDIF

```

```
RATIO=DEXP(UU-HH)
```

```
C***** OBTENCION DE CANDIDATO FINAL
```

```

IF(RATIO.GE.1) THEN
  DO 20 J=1,(P+N)
    BETAF(J)=BETAC(J)
20 CONTINUE
ENDIF

```

```

IF(RATIO.LT.1) THEN

  CALL UNIF(U,IX,IY,IJ)
  IF(U.LE.RATIO) THEN
    DO 21 J=1,(P+N)
      BETAF(J)=BETAC(J)
21 CONTINUE
  ENDIF
  IF(U.GT.RATIO) THEN
    DO 22 J=1,(P+N)
      BETAF(J)=BETA0(J)
22 CONTINUE
  ENDIF
C   WRITE(*,*)U=,U
ENDIF

```

```
C**** GUARDA LAS IM OBSERVACION FINALES (UNA POR UNA EN CADA CICLO)
```

```

DO 30 J=1,(P+N)
  BBETAF(IJ)=BETAF(J)
30 CONTINUE

```

```
C***** ETAPA DE MONITOREO *****
```

```

C
c   ANU=0.0D0
c   DO 32 J=1,(P+N)
c     ANU=X(7,J)*BETAF(J)+ ANU
c32 CONTINUE
c   AANU(IJ)=ANU
c
C
c   SDELTA=0.0D0
c   DO 33 J=(P+1),(P+N)
c     SDELTA=BETAF(J)+ SDELTA
c33 CONTINUE
c   SSDELTA(IJ)=SDELTA

```

```
C***** ETAPA DE AJUSTE *****
```

```

C
  CALL MULTMV(BBNU,X,BETAF,N,(P+N),ND,NP)
  DO 34 I=1,N
    BBNU(I,I)=BBNU(I)
34 CONTINUE

```

```

C
C*****
C
C**** SE CIERRA EL CICLO DE LA GENERACION DE LAS IM OBSERVACIONES
C**** FINALES, EN CADA ITERACION

600 CONTINUE

C***** CONDICION DE CALENTAMIENTO

IF(ITE.GE.L0) THEN

C***** OBTENCION DE LA MUESTRA DE IM OBSERVACIONES BETA/Y,
C***** EN CADA ITERACION, Y LA RESPECTIVA MEDIA ARITMETICA

DO 410 J=1,(P+N)
SUM1(J)=0.0D0
410 CONTINUE

DO 420 J=1,(P+N)
DO 430 II=1,IM
SUM1(J)=BBETA0(AF(II,J)+SUM1(J))
430 CONTINUE
420 CONTINUE

DO 431 J=1,(P+N)
PROM(J)=SUM1(J)/DBLE(IM)
431 CONTINUE

C***** OBTENCION DE LAS 'MEDIAS ERGODICAS'

IF(ITE.EQ.L0) THEN
DO 433 J=1,(P+N)
AE(J)=PROM(J)
AEAUX(J)=AE(J)
433 CONTINUE
WRITE(4,*) (AE(J), J=1,(P+N))
ENDIF

IF(ITE.GT.L0) THEN
JJ=ITE-L0+1
DO 434 J=1,(P+N)
AE(J)=((DBLE(JJ-1)*AEAUX(J))+PROM(J))/DBLE(JJ)
AEAUX(J)=AE(J)
434 CONTINUE
WRITE(4,*) (AE(J), J=1,(P+N))
ENDIF

C*****

c WRITE(7,*)(AANU(I), I=1,IM)
c WRITE(8,*)(SSDELTA(I), I=1,IM)

C**** ENDIF DE LA CONDICION DE CALENTAMIENTO *****

ENDIF

C***** HACE INICIALES LOS VALORES FINALES DE CADA ITERACION

DO 450 II=1,IM
DO 451 J=1,(P+N)
BBETA0(II,J)=BBETA0(AF(II,J))
451 CONTINUE
450 CONTINUE

```

```

C*****
400 CONTINUE
C***** ETAPA DE AJUSTE *****
DO 411 I=1,N
SUM2(I)=0.0D0
HMU(I)=0.0D0
SUM22(N)=0.0D0
PROMNU2(I)=0.0D0
HMUNF(I)=0.0D0
HMUSUP(I)=0.0D0
VARNU(I)=0.0D0
411 CONTINUE
C
DO 412 I=1,N
DO 413 II=1,IM
SUM2(I)=BBNNU(II,I)+SUM2(I)
SUM22(I)=(BBNNU(II,I)**2.0D0)+SUM22(I)
413 CONTINUE
412 CONTINUE
DO 414 I=1,N
PROMNU(I)=SUM2(I)/DBLE(IM)
PROMNU2(I)=SUM22(I)/DBLE(IM)
VARNU(I)=PROMNU2(I)-(PROMNU(I)**2.0D0)
414 CONTINUE

CON=0.0D0
CON1=0.0D0

IF(KK.EQ.1) THEN
DO 415 I=1,N
HMU(I)=0.0
CON=0.0
415 CONTINUE
ENDIF
IF(KK.EQ.2) THEN
DO 416 I=1,N
HMU(I)=0.0
CON=0.0
416 CONTINUE
ENDIF
IF(KK.EQ.3) THEN
DO 417 I=1,N
c write(*,*)promnu(i),con,con1,y(i),hmu(i)
HMU(I)=DEXP(PROMNU(I))
IF(Y(I).EQ.0.0D0) THEN
CON1=CON1+HMU(I)
ENDIF
IF(Y(I).NE.0.0D0) THEN
CON=(Y(I)*DLOG(Y(I)/HMU(I)))-(Y(I)-HMU(I))+CON
ENDIF
417 CONTINUE
ENDIF
IF(KK.EQ.4) THEN
DO 418 I=1,N
HMUNF(I)=DEXP(PROMNU(I)-2.0D0*VARNU(I))/(1.0 +
+ DEXP(PROMNU(I)-2.0D0*VARNU(I)) )
HMUSUP(I)=DEXP(PROMNU(I)+2.0D0*VARNU(I))/(1.0 +
+ DEXP(PROMNU(I)+2.0D0*VARNU(I)) )

HMU(I)=DEXP(PROMNU(I))/(1.0D0+DEXP(PROMNU(I)))
CON=Y(I)*DLOG(Y(I)/HMU(I)) + ((VM(I)-Y(I))*DLOG((
+ VM(I)-Y(I))/(VM(I)-HMU(I)))) + CON
418 CONTINUE
ENDIF

```

```

DEVIANZA= 2.0D0 * (CON+CON1)
OPEN(9,FILE='pre_dev.ca',STATUS='unknown')
WRITE(9,*)'_____ITERACION____',ITE
DO 419 I=1,N
  WRITE(9,*)Y(I),HMMUINF(I),HMMU(I),HMMUSUP(I)
419 CONTINUE
  WRITE(9,*)DEVIANZA=',DEVIANZA
CLOSE(9)
C***** IMPRESION DE LA ULTIMA MUESTRA DE TAMAÑO IM *****
OPEN(10,FILE='beta.ca',STATUS='UNKNOWN')
DO 1102 I=1,IM
  WRITE(10,*) (BBETA(I,J), J=1,(P+N))
1102 CONTINUE
CLOSE(10)
CLOSE(4)
CLOSE(7)
CLOSE(8)
STOP
END
C

```

C PROGRAMA M2.F

C ESTE PROGRAMA REPRESENTA EL ALGORITMO 2 (INDEPENDENCIA).

C PARA EL MODELO SEMIPARAMETRICO CONSIDERANDO UNA VEROSIMILITUD

C ASOCIADA A LA FAMILIA EXPONENCIAL.

C ULTIMA ACTUALIZACION : JUNIO DE 1998

C MODIFICACIONES QUE INCLUYEN PROMEDIOS ERGODICOS Y

C UNA ETAPA DE CALENTAMIENTO DEL 10% DEL TOTAL DE ITERACIONES

C MODIFICACIONES QUE INCLUYEN EL CALCULO DE DOS ESTADISTICAS

C QUE CONFORMAN UNA ETAPA(1) DE MONITOREO.

C Y UNA ETAPA(2) DE AJUSTE QUE INCLUYE EL CALCULO DE LAS

C MU (PARA CADA OBSERVACION),

C Y LA DEVIANZA EN BASE A LA MUESTRA FINAL.

C

C NOTA: ESTE PROGRAMA NO DEPENDE YA, DE SUBROUTINAS DE IMSL

C234567

IMPLICIT DOUBLE PRECISION (A-H,O-Z)

INTEGER P

CHARACTER*12 FDAT,FVALINI,FSAL

C NOTA: EL VALOR DEL PARAMETRO ND DEBE SER MAYOR AL TAMAÑO DE

C MUESTRA GENERADO, IM.

C

C NOTA: ND->DIMENSION DE LOS ARREGLOS PARA EL TAMAÑO DE MUESTRA N

C NM->DIMENSION PARA LOS ARRWGLOS CON TAMAÑOS DE MUESTRA

C

C GENERADOS IM.

C NP->DIMENSION DE LOS VECTORES DE PARAMETROS DE (P+N)

PARAMETER(ND=101,NP=151,NM=5001)

DIMENSION Y(ND),VM(ND),X(ND,NP),VBETA F(NP),BETA I(NP),AE(NP)

DIMENSION BBETA0(NM,NP),BETA0(NP),BETAC(NP),BETA F(NP),BETA(NP)

DIMENSION SUM1(NP),PROM(NP)

DIMENSION B1(NP,NP),B1INV(NP,NP),BBETA F(NM,NP)

DIMENSION VB0(NP),B0(NP,NP),B0INV(NP,NP),AEAUX(NP)

DIMENSION A_NU(NM),S_DELTA(NM)

DIMENSION B_NU(ND),B_NNU(NM,ND),SUM2(ND),PROMNU(ND),HMU(ND)

DIMENSION INDX(NP)

DIMENSION SUM22(ND),PROMNU2(ND),HMUINF(ND),HMUSUP(ND)

DIMENSION VARNU(ND)

C*****

OPEN(11,FILE='/usr/inv/mbp/gab/parametros',STATUS='OLD')

READ(11,*) RHO2

READ(11,*) TETHA

CLOSE(11)

C*****

OPEN(1,FILE='ind.ins',STATUS='OLD')

READ(1,(A)) FDAT

READ(1,(A)) FVALINI

READ(1,(A)) FSAL

READ(1,*) P

READ(1,*) N

```

READ(1,*) S2
READ(1,*) KK
READ(1,*) IM
READ(1,*) K

CLOSE(1)

OPEN(5,FILE='/usr/inv/mbp/gab/val.0',STATUS='OLD')
READ(5,*)(VB0(J), J=1,(P+N))
DO 1 I=1,(P+N)
  READ(5,*)(B0INV(I,J), J=1,(P+N))
1 CONTINUE
CLOSE(5)

c call dwrmm('matriz b0inv',p+n,p+n,b0inv,np,0)
C CALL DLINDS((P+N),B0INV,NP,B0,NP)
CALL INVERSA(B0,B0INV,INDX,P+N,NP)
c call dwrmm('matriz b0',p+n,p+n,b0,np,0)
C
OPEN(2,FILE='FDAT',STATUS='OLD')

if(kk.ne.4) then
DO 1200 I=1,N
  READ(2,*) Y(I), VM(I), (X(I,J), J=1,P)
1200 CONTINUE
else
DO 1300 I=1,N
  READ(2,*) Y(I), VM(I), (X(I,J), J=1,P)
  Y(I) = Y(I) / VM(I)
C WRITE(*,*) Y(I), VM(I), (X(I,J), J=1,P)
1300 CONTINUE
endif

CLOSE(2)
C*****
INTER=1
C***** CONSTRUCCION DE LA NUEVA MATRIZ X *****

DO 2 I=(P+1),(P+N)
  J=I-P
  X(J,I)=1.0D0
2 CONTINUE

c call dwrmm('matriz x',n, n+p,x,nd,0)
C*****
OPEN(3,FILE='/usr/inv/mbp/gab/val.ini',STATUS='OLD')

READ(3,*)(VBETAF(J), J=1,(P+N))
C WRITE(*,*)(VBETAF(J), J=1,(P+N))
C
DO 1101 I=1,(P+N)
  READ(3,*)(B1INV(I,J), J=1,(P+N))
C WRITE(*,*)(B1INV(I,J), J=1,(P+N))
1101 CONTINUE

CLOSE(3)

c do 1500 I=1,(P+N)
c do 1501 J=1,(P+N)
c B1INV(I,J)=1.2D0*B1INV(I,J)
c 1501 continue
c 1500 continue

c call dwrmm('matriz b1inv',p+n,p+n,b1inv,np,0)
C CALL DLINDS((P+N),B1INV,NP,B1,NP)

```

```

CALL INVERSA(B1,B1INV,INDX,P+N,NP)
c CALL DWRRRN('MATRIZ DE PRECISION',(P+N),(P+N),B1,NP,0)

C WRITE(*,*) 'ENTER SEEDS --> IX,IY,IZ (0<IX,IY,IZ<30000)'
C READ(*,*) IX,IY,IZ
IX=265
IY=65
IZ=5

OPEN(4,FILE=FSAL,STATUS='UNKNOWN')
c OPEN(7,FILE='/tmpu/mbp/ind/nu.mon',STATUS='UNKNOWN')
c OPEN(8,FILE='/tmpu/mbp/ind/sdelta.mon',STATUS='UNKNOWN')

C***** APROXIMACION A LA MEDIA Y VARIANZA DE BETA/Y

DO 10 J=1,(P+N)
  BETAI(J)=VBETA(J)
10 CONTINUE

C MATRIZ DE VAR-COV = INVERSA DE B1 = B1INV
C MATRIZ DE PRECISION = B1

C***** PERIODO DE CALENTAMIENTO (10% DEL NUMERO DE ITERACIONES)

L=NINT(0.10*K)
L0=L+1

C***** SIMULA 500 OBSERVACIONES BETA0 DE UNA NORMAL (P+N)_VARIADA CON
C***** MEDIA = BETAI Y MATRIZ DE VAR-COV = INVERSA DE B1 = B1INV

DO 11 M=1,IM
  CALL MULTNORM(BETA0,(P+N),NP,BETALB1INV,IX,IY,IZ)
C CALL DWRRRN('BETA0',(P+N),1,BETA0,NP,0)
  DO 12 J=1,(P+N)
    BBETA0(M,J)=BETA0(J)
  12 CONTINUE
11 CONTINUE

K=K+L

DO 400 ITE=1,K

C***** CICLO QUE GENERA LAS IM OBSERVACIONES 'FINALES'
C***** (EN CADA ITERACION)

DO 600 II=1,IM

C***** ASIGNA EL VECTOR INICIAL

DO 13 J=1,(P+N)
  BETA0(J)=BBETA0(II,J)
13 CONTINUE

C***** SIMULA UNA VARIABLE BETAC DE UNA NORMAL (P+N) VARIADA(BETALB1INV)

CALL MULTNORM(BETAC,(P+N),NP,BETALB1INV,IX,IY,IZ)

C***** CALCULO DE LA PROBABILIDAD DE ACEPTACION

IF(KK.EQ.1) THEN
CALL UH_N(UU,HH,BETAC,BETA0,VB0,B0,BETAL,S2,VM,X,Y,(P+N),N,NP,ND)
ENDIF
IF(KK.EQ.2) THEN
CALL UH_G(UU,HH,BETAC,BETA0,VB0,B0,BETAL,S2,VM,X,Y,(P+N),N,NP,ND)
ENDIF

```

```

IF(KK.EQ.3) THEN
CALL UH_P(UU,HH,BETAC,BETA0,VB0,B0,BETA1,S2,VM,X,Y,(P+N),N,NP,ND)
ENDIF
IF(KK.EQ.4) THEN
CALL UH_B(UU,HH,BETAC,BETA0,VB0,B0,BETA1,S2,VM,X,Y,(P+N),N,NP,ND)
ENDIF

```

```
RATIO=DEXP(UU-HH)
```

```
C***** OBTENCION DE CANDIDATO FINAL
```

```

IF(RATIO.GE.1) THEN
DO 20 J=1,(P+N)
  BETAF(J)=BETAC(J)
20 CONTINUE
ENDIF

```

```
IF(RATIO.LT.1) THEN
```

```

CALL UNIF(U,IX,IY,IZ)
IF(U.LE.RATIO) THEN
DO 21 J=1,(P+N)
  BETAF(J)=BETAC(J)
21 CONTINUE
ENDIF
IF(U.GT.RATIO) THEN
DO 22 J=1,(P+N)
  BETAF(J)=BETA0(J)
22 CONTINUE
ENDIF
C WRITE(*,*) U=,U
ENDIF

```

```
C**** GUARDA LAS IM OBSERVACION FINALES (UNA POR UNA EN CADA CICLO)
```

```

DO 30 J=1,(P+N)
  BBETAF(I,J)=BETAF(J)
30 CONTINUE

```

```
C***** ETAPA DE MONITOREO *****
```

```

C
c ANU=0.0D0
c DO 32 J=1,(P+N)
c ANU=X(7,J)*BETAF(J)+ ANU
c32 CONTINUE
c A_NU(I)=ANU
C
c SDELTA=0.0D0
c DO 33 J=(P+1),(P+N)
c SDELTA=BETAF(J)+ SDELTA
c33 CONTINUE
c S_DELTA(I)=SDELTA
C

```

```
C***** ETAPA DE AJUSTE *****
```

```

C
CALL MULTMV(B_NU,X,BETAF,N,(P+N),ND,NP)
DO 34 I=1,N
  B_NNU(I,I)=B_NU(I)
34 CONTINUE

```

C**** SE CIERRA EL CICLO DE LA GENERACION DE LAS M OBSERVACIONES
C**** FINALES, EN CADA ITERACION

600 CONTINUE

C***** CONDICION DE CALENTAMIENTO

IF(ITE.GE.L0) THEN

C***** OBTENCION DE LA MUESTRA DE IM OBSERVACIONES BETA/Y,
C***** EN CADA ITERACION, Y LA RESPECTIVA MEDIA ARITMETICA

DO 410 J=1,(P+N)
SUM1(J)=0.0D0
410 CONTINUE

DO 420 J=1,(P+N)
DO 430 II=1,IM
SUM1(J)=BBETAF(II,J)+SUM1(J)
430 CONTINUE
420 CONTINUE

DO 431 J=1,(P+N)

PROM(J)=SUM1(J)/DBLE(IM)
431 CONTINUE

C***** OBTENCION DE LAS 'MEDIAS ERGODICAS'

IF(ITE.EQ.L0) THEN
DO 433 J=1,(P+N)
AE(J)=PROM(J)
AEAUX(J)=AE(J)
433 CONTINUE
WRITE(4,*) (AE(J), J=1,(P+N))
ENDIF

IF(ITE.GT.L0) THEN
JJ=ITE-L0+1
DO 434 J=1,(P+N)
AE(J)=((DBLE(JJ-1)*AEAUX(J))+PROM(J))/DBLE(JJ)
434 CONTINUE
WRITE(4,*) (AE(J), J=1,(P+N))
ENDIF

C

c WRITE(7,*) (A_NU(I), I=1,IM)

c WRITE(8,*) (S_DELTA(I), I=1,IM)

C

C**** ENDIF DE LA CONDICION DE CALENTAMIENTO

ENDIF

C***** HACE INICIALES LOS VALORES FINALES DE CADA ITERACION

DO 450 II=1,IM
DO 451 J=1,(P+N)
BBETA0(II,J)=BBETAF(II,J)
451 CONTINUE
450 CONTINUE

```

C*****
400 CONTINUE
C***** ETAPA DE AJUSTE *****
DO 411 I=1,N
  SUM2(I)=0.0D0
  HMU(I)=0.0D0
  SUM22(N)=0.0D0
  PROMNU2(I)=0.0D0
  HMUINF(I)=0.0D0
  HMUSUP(I)=0.0D0
  VARNU(I)=0.0D0

411 CONTINUE
DO 412 I=1,N
  DO 413 II=1,IM
    SUM2(I)=B_NNU(II,I)+SUM2(I)
    SUM22(I)=(B_NNU(II,I)**2.0D0)+SUM22(I)
413 CONTINUE
412 CONTINUE
DO 414 I=1,N
  PROMNU(I)=SUM2(I)/DBLE(IM)
  PROMNU2(I)=SUM22(I)/DBLE(IM)
  VARNU(I)=PROMNU2(I)-( PROMNU(I)**2.0D0 )
414 CONTINUE

CON=0.0D0

IF(KK.EQ.1) THEN
  DO 415 I=1,N
    HMU(I)= 0.0
    CON= 0.0
415 CONTINUE
  ENDIF
IF(KK.EQ.2) THEN
  DO 416 I=1,N
    HMU(I)= 0.0
    CON= 0.0
416 CONTINUE
  ENDIF
IF(KK.EQ.3) THEN
  DO 417 I=1,N
    HMU(I)= 0.0
    CON= 0.0
417 CONTINUE
  ENDIF
IF(KK.EQ.4) THEN
  DO 418 I=1,N
    HMUINF(I)=DEXP(PROMNU(I)-2.0D0*VARNU(I))/( 1.0 +
+ DEXP(PROMNU(I)-2.0D0*VARNU(I)) )
    HMUSUP(I)=DEXP(PROMNU(I)+2.0D0*VARNU(I))/( 1.0 +
+ DEXP(PROMNU(I)+2.0D0*VARNU(I)) )

    HMU(I)=DEXP(PROMNU(I))/( 1.0D0+DEXP(PROMNU(I)))
    CON=Y(I)*DLOG( Y(I)/HMU(I) ) + ( (VM(I)-Y(I))*DLOG( (
+ VM(I)-Y(I))/(VM(I)-HMU(I))) ) + CON
418 CONTINUE
  ENDIF

DEVIANZA= 2.0D0 * CON

OPEN(9,FILE='pre_dev.ind',STATUS='unknown')

WRITE(9,*) '_____ ITERACION ____',ITE

```

```
DO 419 I=1,N
  WRITE(9,*)Y(I),HMUIF(I),HMU(I),HMUSUP(I)
419 CONTINUE
  WRITE(9,*)DEVIANZA=',DEVIANZA

  CLOSE(9)

C***** IMPRESION DE LA ULTIMA MUESTRA DE TAMAÑO IM *****
  OPEN(10,FILE='/tmpu/mbp/ind/bbeta.ind',STATUS='UNKNOWN')

  DO 1102 II=1,IM
    WRITE(10,*)(BBETAF(II,J), J=1,(P+N))
1102 CONTINUE

  CLOSE(10)

  CLOSE(4)
  CLOSE(7)
  CLOSE(8)

  STOP
  END
```

```

C  PROGRAMA: Predic.F
C  MODULO: PREDICION.
C
C  ESTE MODULO OBTIENE EL PREDICTOR DE UN NUMERO (MAX)
C  DE OBSERVACIONES,ASI COMO LA CORRESPONDIENTE BANDA DE
C  PREDICION, USANDO EL MODELO SEMIPARAMETRICO.
C
C  ULTIMA ACTUALIZACION: JUNIO DE 1998.
C234567
C
C  IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C  INTEGER P
C
C  PARAMETER(ND=100,NP=100,NM=1001)
C
C  DIMENSION BBETA(NM,NP),BETA(NP),X(ND,NP),WTV(NP)
C  DIMENSION XP(NP),V(ND,ND),R(ND,ND),RINV(ND,ND)
C  DIMENSION XFF(ND,NP),XF(NP),CC(ND),C(ND),RINVC(ND),DXM(ND)
C  DIMENSION DXV(ND),DDX(ND),DELTA(ND),XFFB(ND),VB(NP),YNU(ND,NM)
C  DIMENSION SUM1(ND),PROM(ND),VARNU(ND),HMU(ND),HMUSUP(ND)
C  DIMENSION HMUINF(ND),YNU(ND),YNUINF(ND)
C  DIMENSION SUM2(ND),ENU2(ND)
C
C  DIMENSION INDX(ND)
C*****
C
C  kk=3
C  IM=1000
C  N=34
C  P=4
C  INTER=1
C  RHO2=2.8D0
C  TETHA=1.0D0
C  IX=265
C  IY=65
C  IZ=5
C  MAX=1
C*****
C
C  OPEN(3,FILE='ban.ca',STATUS='UNKNOWN')
C  OPEN(4,FILE='ban.ind',STATUS='UNKNOWN')
C  OPEN(5,FILE='ynui.ca',STATUS='UNKNOWN')
C  OPEN(6,FILE='ynui.ind',STATUS='UNKNOWN')
C
C  OPEN(1,FILE='predic.dat',STATUS='OLD')
C  DO 950 I=1,N
C    READ(1,*)(V(I,J), J=1,N)
C  950 CONTINUE
C  DO 951 I=1,N
C    READ(1,*)(X(I,J), J=1,P)
C  951 CONTINUE
C  READ(1,*)(WTV(J), J=1,P)
C  CLOSE(1)
C
C
C  call dwrmm('matriz x',n,p,x.nd,0)
C  call dwrmm('matriz v',n,n,v.nd,0)
C  call dwrmm('vector w',p,1,winv,np,0)
C***** BANDERA PARA INDICAR CON QUE BETAF SE ESTA TRABAJANDO *****
C***** INDEP=0 --> C.A.F , INDEP=1 --> IND.F *****

```

```

INDEP=0

IF(INDEP.EQ.1) THEN
  OPEN(2,FILE='/export/home/pruebas/gab/ca/
+ bbeta.ind',STATUS='OLD')
  DO 1 II=1,IM
    READ(2,*)(BBETA(II,J), J=1,(P+N))
1  CONTINUE
  CLOSE(2)
ENDIF

IF(INDEP.EQ.0) THEN
C
  OPEN(2,FILE='bbeta.ca',STATUS='OLD')
  DO 2 II=1,IM
    READ(2,*)(BBETA(II,J), J=1,(P+N))
2  CONTINUE
  CLOSE(2)
ENDIF
C
C*****
IF(MAX.GT.1) THEN

C*****  GENERACION DE (MAX) PUNTOS DE DISEÑO *****
C
  XP(1)=.50D0
  XP(MAX)=26.0D0
  PAR=( XP(MAX)-XP(1) )/( DBLE(MAX-1) )
C
  DO 900 IP=2,(MAX-1)
    XP(IP)=XP(1)+( DBLE(IP-1))*PAR
900 CONTINUE
C
c  call dwrmm('vector xp',max,1,xp,nd,0)

  DO 901 IP=1,MAX
    DO 902 J=1,P
      XFF(IP,J)=XP(IP)**(J-1)
902 CONTINUE
      write(*,*) (xff(ip,j),j=1,p)
901 CONTINUE
      write(*,*)*****
C
  ENDIF

IF(MAX.EQ.1) THEN
  XFF(MAX,1)=1.0D0
  XFF(MAX,2)=0.6576D0
  XFF(MAX,3)=0.5806D0
  XFF(MAX,4)=0.6985D0
ENDIF

C***** SE CIERRA EL CICLO DE GENERACION DE MAX PUNTOS DE DISEÑO *****
c  call dwrmm('matriz xff',max,p,xff,nd,0)

  DO 904 I=1,N
    DO 905 J=1,N
      R(I,J)=RHO2*V(I,J)
905 CONTINUE
904 CONTINUE
C
C  CALL DLENDS(N,R,ND,RINV,ND)
  CALL INVERSA(RINV,R,INDX,N,ND)

```

C** CICLO QUE GENERA LAS NUI PARA CADA VECTOR DE LA MUESTRA FINAL

DO 4 II=1,IM

C*****
C

DO 3 J=1,(P+N)
BETAF(J)=BBETAF(II,J)
3 CONTINUE

C***** CICLO QUE GENERA LA MEDIA Y LA VARIANZA PARA *****
C***** UNA MUESTRA DE TAMAÑO (MAX) DE DELTAX *****

C
DO 906 IIF=1,MAX

C*****
C

DO 907 J=1,P
XF(J)=XFF(IIF,J)
907 CONTINUE

C
CALL VECTOR_C(CC,TETHA,X,WINV,XF,P,N,NP,ND,INTER)

c do 1000 i=1,n
c write(*,*) cc(i)
c1000 continue

C
DO 908 I=1,N
C(I)=RHO2*CC(I)
908 CONTINUE

C
CALL MULTMV(RINVC,RINV,C,N,N,ND,ND)
c call dwrmm('vector rinvc',n,1,rinvc,nd,0)

C
CRINVC=0.0D0
DXMEDIA=0.0D0
DO 909 I=1,N
DXMEDIA=(RINVC(I)*BETAF(P+I))+DXMEDIA
CRINVC=(RINVC(I)*C(I))+CRINVC
909 CONTINUE
DXVAR=RHO2-CRINVC

C
DXM(IIF)=DXMEDIA
DXV(IIF)=DXVAR

c do 1001 i=1,max
c write(*,*) dxm(i),dxv(i)
c1001 continue

C
C***** SE CIERRA EL CICLO DE GENERACION DE *****
C***** MEDIAS Y VARIANZAS *****

C
906 CONTINUE

C
C*****

C
c call dwrmm('medias',max,1,dxm,nd,0)

CALL NORMAL(DDX,MAX,ND,IX,IY,IZ)

```

C
  DO 910 IIF=1,MAX
    DELTAX(IIF)=( DDX(IIF)*DSQRT( DXV(IIF) ) )-DXM(IIF)
910 CONTINUE
C
  DO 911 J=1,P
    VB(J)=BETAF(J)
911 CONTINUE
C
  CALL MULTMV(XFFB,XFF,VB,MAX,P,ND,NP)
C
  DO 912 IIF=1,MAX
    YNU(IIF,II)=XFFB(IIF)+DELTAX(IIF)
912 CONTINUE
C***** SE CIERRA CICLO DE OBTENCION DE NUI *****

4 CONTINUE

c call dwrtn('matriz nu',max,im,ynu,nd,0)
*****
IF(INDEP.EQ.1) THEN
  DO 922 IIF=1,MAX
    WRITE(6,*)YNU(IIF,J), J=1,IM)
922 CONTINUE
  ENDIF
IF(INDEP.EQ.0) THEN
  DO 923 IIF=1,MAX
    WRITE(5,*)YNU(IIF,J), J=1,IM)
923 CONTINUE
  ENDIF
C***** OBTENCION DE LA MEDIA Y 'VARIANZA' DEL PREDICTOR XB *****

  DO 913 IIF=1,MAX
    SUM1(IIF)=0.0D0
    SUM2(IIF)=0.0D0
    ENU2(IIF)=0.0D0
913 CONTINUE

  DO 914 IIF=1,MAX
    DO 915 II=1,IM
      SUM1(IIF)=YNU(IIF,II)+SUM1(IIF)
      SUM2(IIF)=(YNU(IIF,II)**2.0D0) + SUM2(IIF)
915 CONTINUE
914 CONTINUE

  DO 916 IIF=1,MAX
    PROM(IIF)=SUM1(IIF)/(DBLE(IM))
    ENU2(IIF)=SUM2(IIF)/(DBLE(IM))
916 CONTINUE

  DO 917 IIF=1,MAX
    DO 918 II=1,IM
      VARNU(IIF)=ENU2(IIF) - (PROM(IIF)**2.0D0)
918 CONTINUE
    YNUSUP(IIF)=PROM(IIF) + DSQRT( VARNU(IIF) )
    YNUINF(IIF)=PROM(IIF) - DSQRT( VARNU(IIF) )
917 CONTINUE

  do 1002 i=1,max
    write(*,*) ynuinf(i),prom(i),ynusup(i),varnu(i)
1002 continue

```

C***** OBTENCION DE 'BANDAS DE PREDICION' *****

DO 919 IIF=1,MAX

```
IF(KK.EQ.1) THEN
  HMU(IIF)=0.0D0
  HMUSUP(IIF)=0.0D0
  HMUINF(IIF)=0.0D0
ENDIF
IF(KK.EQ.2) THEN
  HMU(IIF)=0.0D0
  HMUSUP(IIF)=0.0D0
  HMUINF(IIF)=0.0D0
ENDIF
IF(KK.EQ.3) THEN
  HMU(IIF)=DEXP(PROM(IIF))
  HMUSUP(IIF)=DEXP(YNUSUP(IIF))
  HMUINF(IIF)=DEXP(YNUINF(IIF))
ENDIF
IF(KK.EQ.4) THEN
  HMU(IIF)=DEXP(PROM(IIF))/( 1.0D0+DEXP(PROM(IIF)) )
  HMUSUP(IIF)=DEXP(YNUSUP(IIF))/( 1.0D0+DEXP(YNUSUP(IIF)) )
  HMUINF(IIF)=DEXP(YNUINF(IIF))/( 1.0D0+DEXP(YNUINF(IIF)) )
ENDIF
```

919 CONTINUE

```
IF(INDEP.EQ.1) THEN
  DO 920 IIF=1,MAX
  WRITE(4,*)HMUINF(IIF),HMU(IIF),HMUSUP(IIF)
```

920 CONTINUE

```
ENDIF
IF(INDEP.EQ.0) THEN
  DO 921 IIF=1,MAX
  WRITE(3,*)HMUINF(IIF),HMU(IIF),HMUSUP(IIF)
```

921 CONTINUE

ENDIF

CLOSE(3)

C CLOSE(4)

CLOSE(5)

C CLOSE(6)

STOP

END

```

C*****
C      SUBROUTINAS DEL PROGRAMA M0.F
C*****
SUBROUTINE SHELL(N,ARR)
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
PARAMETER (ALN2I=1.4426950D0, TINY=1.0D-5)
DIMENSION ARR(N)
LOGNB2=INT(ALOG(FLOAT(N))*ALN2I+TINY)
M=N
DO 12 NN=1,LOGNB2
  M=M/2
  K=N-M
  DO 11 J=1,K
    I=J
3    CONTINUE
    L=I+M
    IF(ARR(L).LT.ARR(I)) THEN
      T=ARR(I)
      ARR(I)=ARR(L)
      ARR(L)=T
      I=I-M
      IF(I.GE.1)GO TO 3
    ENDF
11  CONTINUE
12  CONTINUE
RETURN
END
C
SUBROUTINE GORRO_N(VG,V1,V2,H,ZZ,S,KN,KD,V3)

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
INTEGER P
DIMENSION VG(KD),V1(KD),V2(KD),H(KD),ZZ(KD),V3(KD)

C1=0.0D0
C2=0.0D0

DO 20 I=1,KN

  VG(I) = ( ZZ(I) + C1*S/V3(I) )/( 1.0D0 + C2*S/V3(I) )
  V1(I) = VG(I)
  V2(I) = 1.0D0
  H(I) = V3(I)
20 CONTINUE
RETURN
END
C
SUBROUTINE GORRO_G(VG,V1,V2,H,ZZ,S,KN,KD,V3)

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
INTEGER P
DIMENSION VG(KD),V1(KD),V2(KD),H(KD),ZZ(KD),V3(KD)

C1=0.0D0
C2=-1.0D0

DO 20 I=1,KN
  VG(I) = -( 1.0D0 + C2*S/V3(I) )/( ZZ(I) + C1*S/V3(I) )
  V1(I) = -1.0D0/VG(I)
  V2(I) = 1.0D0/(VG(I)**2.0D0)
  H(I) = V3(I)*V2(I)
20 CONTINUE
RETURN
END

```

SUBROUTINE GORRO_P(VG,V1,V2,H,ZZ,S,KN,KD,V3)

IMPLICIT DOUBLE PRECISION (A-H,O-Z)

INTEGER P

DIMENSION VG(KD),V1(KD),V2(KD),H(KD),ZZ(KD),V3(KD)

C1=0.5D0

C2=0.0D0

DO 20 I=1,KN

VG(I) = DLOG(ZZ(I) + C1*S/V3(I))/(1.0D0 + C2*S/V3(I))

V1(I) = DEXP(VG(I))

V2(I) = V1(I)

H(I) = V3(I)*V2(I)

20 CONTINUE

RETURN

END

C

SUBROUTINE GORRO_B(VG,V1,V2,H,ZZ,S,KN,KD,V3)

IMPLICIT DOUBLE PRECISION (A-H,O-Z)

INTEGER P

PARAMETER(ND=5001)

DIMENSION AUX(ND),VG(KD),V1(KD),V2(KD),H(KD),ZZ(KD),V3(KD)

C1=0.5D0

C2=1.0D0

DO 20 I=1,KN

AUX(I) = (ZZ(I) + C1*S/V3(I))/(1.0D0 + C2*S/V3(I))

VG(I) = DLOG(AUX(I) / (1 - AUX(I)))

V1(I) = DEXP(VG(I)) / (1.0D0 + DEXP(VG(I)))

V2(I) = V1(I)*(1.0D0 - V1(I))

H(I) = V3(I)*V2(I)

20 CONTINUE

RETURN

END

C

SUBROUTINE GORRO2_N(V1,V2,H,KN,KD,V3,VG)

IMPLICIT DOUBLE PRECISION (A-H,O-Z)

INTEGER P

DIMENSION VG(KD),V1(KD),V2(KD),H(KD),V3(KD)

DO 20 I=1,KN

V1(I) = VG(I)

V2(I) = 1.0D0

H(I) = V3(I)

20 CONTINUE

RETURN

END

C

SUBROUTINE GORRO2_G(V1,V2,H,KN,KD,V3,VG)

IMPLICIT DOUBLE PRECISION (A-H,O-Z)

INTEGER P

DIMENSION VG(KD),V1(KD),V2(KD),H(KD),V3(KD)

DO 20 I=1,KN

```

V1(I) = -1.0D0/VG(I)
V2(I) = 1.0D0/(VG(I)**2.0D0)
H(I) = V3(I)*V2(I)
20 CONTINUE
RETURN
END
C
SUBROUTINE GORRO2_P(V1,V2,H,KN,KD,V3,VG)

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
INTEGER P
DIMENSION VG(KD),V1(KD),V2(KD),H(KD),V3(KD)

DO 20 I=1,KN
V1(I) = DEXP(VG(I))
V2(I) = V1(I)
H(I) = V3(I)*V2(I)
20 CONTINUE

RETURN
END
C
SUBROUTINE GORRO2_B(V1,V2,H,KN,KD,V3,VG)

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
INTEGER P
DIMENSION VG(KD),V1(KD),V2(KD),H(KD),V3(KD)

DO 20 I=1,KN

V1(I) = DEXP(VG(I)) / ( 1.0D0 + DEXP(VG(I)) )
V2(I) = V1(I)*( 1.0D0 - V1(I) )
H(I) = V3(I)*V2(I)
20 CONTINUE

RETURN
END
C CALL MATRIX_V(V,TETHA,X,WINV,P,N,NP,ND,INTER)

SUBROUTINE MATRIX_V(A,TETA,XX,PM,KP,KN,LP,LN,IB)

C SALIDA
C
C A <----- MATRIZ DE KNxKN
C
C ENTRADAS
C
C TETA <----- ESCALAR POSITIVO
C XX <----- MATRIZ DE DATOS DE KNxKP
C PM <----- VECTOR (MATRIX DIAG) DE (P-1)x(P-1)
C KR,KN <----- DIMENSIONES REALES
C LR,LN <----- DIMENSIONES ESPECIFICADAS
C <----- COMO EN EL PRINCIPAL
C IB <----- BANDERA PARA INDICAR SI HAY O NO
C <----- INTERCEPTO

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
DIMENSION A(LN,LN),XX(LN,LP),PM(LP)
PARAMETER(NI=50)
DIMENSION A1(NI)

WRITE(*,*) 'TETA=',TETA,' KP=',KP,' KN=',KN,' LP=',LP,' LN=',LN
C CALL DWRRRN('MATRIZ DE DATOS',KN,KP,XX,KN,0)
C CALL DWRRRN('VECTOR WINV',1,KP,PM,1,0)

```

```

DO 101 I=1,KN
DO 102 J=1,KN
DO 103 K=(1+IB),KP
A1(K)=XX(I,K)-XX(J,K)
103 CONTINUE

```

```

A2=0.0D0
DO 104 KK=(1+ib),KP
A2=PM(KK)*A1(KK)*A1(KK)+A2
104 CONTINUE

```

A(I,J) = DEXP(-TETA*A2)

```

102 CONTINUE
101 CONTINUE

```

```

RETURN
END

```

```

C*****
C SUBROUTINAS DEL PROGRAMA M1.F
C*****

```

SUBROUTINE UH_N(A,B,BC,B0,BI,PM,S,VP,XX,YY,KP,KN,IP,IN)

```

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
PARAMETER(NP=151)
INTEGER P
DIMENSION BC(IP),B0(IP),BI(IP),PM(IP,IP),VP(IN),XX(IN,IP)
DIMENSION A1(NP),A11(NP),YY(IN)

```

```

DO 101 J=1,KP
A1(J)=BC(J)-BI(J)
101 CONTINUE

```

```

DO 102 J=1,KP
A11(J)=B0(J)-BI(J)
102 CONTINUE

```

```

A2=0.0D0
DO 103 J=1,KP
DO 104 JJ=1,KP
A2=PM(JJ,J)*A1(JJ)*A1(J)+A2
104 CONTINUE
103 CONTINUE

```

```

A22=0.0D0
DO 105 J=1,KP
DO 106 JJ=1,KP
A22=PM(JJ,J)*A11(JJ)*A11(J)+A22
106 CONTINUE
105 CONTINUE

```

```

A3=0.0D0
A4=0.0D0
DO 107 I=1,KN
A3=0.0D0
DO 108 J=1,KP
A3=XX(I,J)*BC(J)+A3
108 CONTINUE
A4=VP(I)*(YY(I)*A3 - (A3**2.0D0)/2.0D0) + A4
107 CONTINUE

```

```

A = -(A2/(2.0D0)) + (A4/S)
  A33=0.0D0
A44=0.0D0
DO 109 I=1,KN
  A33=0.0D0
  DO 110 J=1,KP
    A33=XX(L,J)*B0(J)+A33
110 CONTINUE
  A44=VP(I)*(YY(I)*A33 - (A33**2.0D0)/2.0D0) + A44
109 CONTINUE

  B = -(A22/(2.0D0)) + (A44/S)

RETURN
END
C
SUBROUTINE UH_G(A,B,BC,B0,BI,PM,S,VP,XX,YY,KP,KN,IP,IN)

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
PARAMETER(NP=151)
INTEGER P
DIMENSION BC(IP),B0(IP),BI(IP),PM(IP,IP),VP(IN),XX(IN,IP)
DIMENSION A1(NP),A11(NP),YY(IN)

DO 101 J=1,KP
  A1(J)=BC(J)-BI(J)
101 CONTINUE

c  call dwrmm('a1',1,kp,a1,1,0)

DO 102 J=1,KP
  A11(J)=B0(J)-BI(J)
102 CONTINUE

c  call dwrmm('a11',1,kp,a11,1,0)

A2=0.0D0
DO 103 J=1,KP
  DO 104 JJ=1,KP
    A2=PM(JJ,J)*A1(JJ)*A1(J)+A2
104 CONTINUE
103 CONTINUE

write(*,*)a2='a2'

A22=0.0D0
DO 105 J=1,KP
  DO 106 JJ=1,KP
    A22=PM(JJ,J)*A11(JJ)*A11(J)+A22
106 CONTINUE
105 CONTINUE

write(*,*)a22='a22'

A3=0.0D0
A4=0.0D0
DO 107 I=1,KN
  A3=0.0D0
  DO 108 J=1,KP
    A3=XX(L,J)*BC(J)+A3
108 CONTINUE
  A4=VP(I)*(YY(I)*A3 + DLOG(A3)) + A4
107 CONTINUE

write(*,*)a3='a3'

```

```

write(*,*)a4=',a4
A =-(A2/(2.0D0)) + (A4/S)

write(*,*)a=',a

A33=0.0D0
A44=0.0D0
DO 109 I=1,KN
  A33=0.0D0
  DO 110 J=1,KP
    A33=XX(I,J)*B0(J)+A33
110 CONTINUE
  A44=VP(I)*(YY(I)*A33 + DLOG(A33)) + A44
109 CONTINUE

write(*,*)a33=',a33
write(*,*)a44=',a44

B =-(A22/(2.0D0)) + (A44/S)

write(*,*)b=',b

RETURN
END
C
SUBROUTINE UH_P(A,B,BC,B0,BI,PM,S,VP,XX,YY,KP,KN,IP,IN)

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
PARAMETER(NP=151)
INTEGER P
DIMENSION BC(IP),B0(IP),BI(IP),PM(IP,IP),VP(IN),XX(IN,IP)
DIMENSION A1(NP),A11(NP),YY(IN)

DO 101 J=1,KP
  A1(J)=BC(J)-BI(J)
101 CONTINUE

DO 102 J=1,KP
  A11(J)=B0(J)-BI(J)
102 CONTINUE

A2=0.0D0
DO 103 J=1,KP
  DO 104 JJ=1,KP
    A2=PM(JJ,J)*A1(JJ)*A1(J)+A2
104 CONTINUE
103 CONTINUE

A22=0.0D0
DO 105 J=1,KP
  DO 106 JJ=1,KP
    A22=PM(JJ,J)*A11(JJ)*A11(J)+A22
106 CONTINUE
105 CONTINUE

A3=0.0D0
A4=0.0D0
DO 107 I=1,KN
  A3=0.0D0
  DO 108 J=1,KP
    A3=XX(I,J)*BC(J)+A3
108 CONTINUE
  A4=VP(I)*(YY(I)*A3 - DEXP(A3)) + A4
107 CONTINUE

```

```

A = -(A2/(2.0D0)) + (A4/S)
A33=0.0D0
A44=0.0D0
DO 109 I=1,KN
  A33=0.0D0
  DO 110 J=1,KP
    A33=XX(I,J)*B0(J)+A33
110 CONTINUE
  A44=VP(I)*(YY(I)*A33 - DEXP(A33)) + A44
109 CONTINUE

```

```

B = -(A22/(2.0D0)) + (A44/S)
RETURN
END

```

C
SUBROUTINE UH_B(A,B,BC,B0,BI,PM,S,VP,XX,YY,KP,KN,IP,IN)

```

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
PARAMETER(NP=151)
INTEGER P
DIMENSION BC(IP),B0(IP),BI(IP),PM(IP,IP),VP(IN),XX(IN,IP)
DIMENSION A1(NP),A11(NP),YY(IN)

```

```

DO 101 J=1,KP
  A1(J)=BC(J)-BI(J)
101 CONTINUE

```

```

DO 102 J=1,KP
  A11(J)=B0(J)-BI(J)
102 CONTINUE

```

```

A2=0.0D0
DO 103 J=1,KP
  DO 104 JJ=1,KP
    A2=PM(JJ,J)*A1(JJ)*A1(J)+A2
104 CONTINUE
103 CONTINUE

```

```

A22=0.0D0
DO 105 J=1,KP
  DO 106 JJ=1,KP
    A22=PM(JJ,J)*A11(JJ)*A11(J)+A22
106 CONTINUE
105 CONTINUE

```

```

A3=0.0D0
A4=0.0D0
DO 107 I=1,KN
  A3=0.0D0
  DO 108 J=1,KP
    A3=XX(I,J)*BC(J)+A3
108 CONTINUE
  A4=VP(I)*(YY(I)*A3 - DLOG(1.0D0 + DEXP(A3))) + A4
107 CONTINUE

```

```

A = -(A2/(2.0D0)) + (A4/S)
A33=0.0D0
A44=0.0D0
DO 109 I=1,KN
  A33=0.0D0
  DO 110 J=1,KP
    A33=XX(I,J)*B0(J)+A33
110 CONTINUE
  A44=VP(I)*(YY(I)*A33 - DLOG(1.0D0 + DEXP(A33))) + A44
109 CONTINUE

```

```

B = -(A22/(2.0D0)) + (A44/S)

RETURN
END
C*****
C      SUBRITNAS DEL PROGRAMA M2.F
C*****

SUBROUTINE UH_N(EU,EH,BC,B0,BI,PM,BII,S,VP,XX,YY,KP,KN,IP,IN)

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
PARAMETER(NP=151)
INTEGER P
DIMENSION BC(IP),B0(IP),BI(IP),PM(IP,IP),VP(IN),XX(IN,IP)
DIMENSION A1(NP),A11(NP),YY(IN)
DIMENSION AA1(NP),AA11(NP),BII(IP)

DO 101 J=1,KP
  A1(J) = BC(J)-BI(J)
  AA1(J) = BC(J)-BII(J)
101 CONTINUE

DO 102 J=1,KP
  A11(J) = B0(J)-BI(J)
  AA11(J) = B0(J)-BII(J)
102 CONTINUE

AA2=0.0D0
A2 =0.0D0
DO 103 J=1,KP
  DO 104 JJ=1,KP
    A2 = PM(JJ,J)*A1(JJ)*A1(J) + A2
    AA2 = PM(JJ,J)*AA1(JJ)*AA1(J) + AA2
104 CONTINUE
103 CONTINUE

AA22=0.0D0
A22 =0.0D0
DO 105 J=1,KP
  DO 106 JJ=1,KP
    A22 = PM(JJ,J)*A11(JJ)*A11(J) + A22
    AA22 = PM(JJ,J)*AA11(JJ)*AA11(J) + AA22
106 CONTINUE
105 CONTINUE

A3=0.0D0
A4=0.0D0
DO 107 I=1,KN
  A3=0.0D0
  DO 108 J=1,KP
    A3=XX(I,J)*BC(J)+A3
108 CONTINUE
  A4=VP(I)*YY(I)*A3 - (A3**2.0D0)/2.0D0 + A4
107 CONTINUE

A = -A2/2.0D0 + (A4/S)
AA = -AA2/2.0D0
EU = A-AA

```

```

A33=0.0D0
A44=0.0D0
DO 109 I=1,KN
  A33=0.0D0
  DO 110 J=1,KP
    A33=XX(I,J)*B0(J)+A33
110 CONTINUE
  A44=VP(I)*(YY(I)*A33 - (A33**2.0D0)/2.0D0) + A44
109 CONTINUE

  B = -A22/2.0D0 + (A44/S)
  BB = -AA22/2.0D0
  EH = B-BB
  RETURN
  END
C
SUBROUTINE UH_G(A,B,BC,B0,BI,PM,BII,S,VP,XX,YY,KP,KN,IP,IN)

  IMPLICIT DOUBLE PRECISION (A-H,O-Z)
  PARAMETER(NP=151)
  INTEGER P
  DIMENSION BC(IP),B0(IP),BI(IP),PM(IP,IP),VP(IN),XX(IN,IP)
  DIMENSION A1(NP),A11(NP),YY(IN)
  DIMENSION AA1(NP),AA11(NP),BII(IP)

  DO 101 J=1,KP
    A1(J) = BC(J)-BI(J)
    AA1(J) = BC(J)-BII(J)
101 CONTINUE

c   call dvrmm('a1',1,kp,a1,1,0)

  DO 102 J=1,KP
    A11(J) = B0(J)-BI(J)
    AA11(J) = B0(J)-BII(J)
102 CONTINUE

c   call dvrmm('a11',1,kp,a11,1,0)

  AA2=0.0D0
  A2 =0.0D0
  DO 103 J=1,KP
    DO 104 JJ=1,KP
      A2 = PM(JJ,J)*A1(JJ)*A1(J) + A2
      AA2 = PM(JJ,J)*AA1(JJ)*AA1(J) + AA2
104 CONTINUE
103 CONTINUE

  write(*,*)a2=',a2'

  AA22=0.0D0
  A22 =0.0D0
  DO 105 J=1,KP
    DO 106 JJ=1,KP
      A22 = PM(JJ,J)*A11(JJ)*A11(J) + A22
      AA22 = PM(JJ,J)*AA11(JJ)*AA11(J) + AA22
106 CONTINUE
105 CONTINUE

  write(*,*)a22=',a22'

  A3=0.0D0
  A4=0.0D0

```

```

DO 107 I=1,KN
  A3=0.0D0
  DO 108 J=1,KP
    A3=XX(I,J)*BC(J)+A3
108 CONTINUE
  A4=VP(I)*(YY(I)*A3 + DLOG(A3)) + A4
107 CONTINUE

  write(*,*)a3=',a3
  write(*,*)a4=',a4

  A = -A2/2.0D0 + (A4/S)
  AA = -AA2/2.0D0
  EU = A-AA

  write(*,*)a=',a

  A33=0.0D0
  A44=0.0D0
  DO 109 I=1,KN
    A33=0.0D0
    DO 110 J=1,KP
      A33=XX(I,J)*B0(J)+A33
110 CONTINUE
    A44=VP(I)*(YY(I)*A33 + DLOG(A33)) + A44
109 CONTINUE

  write(*,*)a33=',a33
  write(*,*)a44=',a44

  B = -A22/2.0D0 + (A44/S)
  BB = -AA22/2.0D0
  EH = B-BB

  write(*,*)b=',b

  RETURN
  END
C
C
C
SUBROUTINE UH_P(EU,EH,BC,B0,BI,PM,BILS,VP,XX,YY,KP,KN,IP,IN)

  IMPLICIT DOUBLE PRECISION (A-H,O-Z)
  PARAMETER(NP=151)
  INTEGER P
  DIMENSION BC(IP),B0(IP),BI(IP),PM(IP,IP),VP(IN),XX(IN,IP)
  DIMENSION A1(NP),A11(NP),YY(IN)
  DIMENSION AA1(NP),AA11(NP),BII(IP)

  DO 101 J=1,KP
    A1(J) = BC(J)-BI(J)
    AA1(J) = BC(J)-BII(J)
101 CONTINUE

  DO 102 J=1,KP
    A11(J) = B0(J)-BI(J)
    AA11(J) = B0(J)-BII(J)
102 CONTINUE

  AA2 =0.0D0
  A2 =0.0D0

```

```

DO 103 J=1,KP
DO 104 JJ=1,KP
  A2 = PM(JJ,J)*A1(JJ)*A1(J) + A2
  AA2 = PM(JJ,J)*AA1(JJ)*AA1(J) + AA2
104 CONTINUE
103 CONTINUE

AA22=0.0D0
A22 =0.0D0
DO 105 J=1,KP
DO 106 JJ=1,KP
  A22 = PM(JJ,J)*A11(JJ)*A11(J) + A22
  AA22 = PM(JJ,J)*AA11(JJ)*AA11(J) + AA22
106 CONTINUE
105 CONTINUE

A3=0.0D0
A4=0.0D0
DO 107 I=1,KN
  A3=0.0D0
DO 108 J=1,KP
  A3=XX(I,J)*BC(J)+A3
108 CONTINUE
  A4=VP(I)*(YY(I)*A3 - DEXP(A3)) + A4
107 CONTINUE

A = -A2/2.0D0 + (A4/S)
AA = -AA2/2.0D0
EU = A-AA

A33=0.0D0
A44=0.0D0
DO 109 I=1,KN
  A33=0.0D0
DO 110 J=1,KP
  A33=XX(I,J)*B0(J)+A33
110 CONTINUE
  A44=VP(I)*(YY(I)*A33 - DEXP(A33)) + A44
109 CONTINUE

B = -A22/2.0D0 + (A44/S)
BB = -AA22/2.0D0
EH = B-BB

RETURN
END
C
C
C
SUBROUTINE UH_B(EU,EH,BC,B0,BI,PM,BII,S,VP,XX,YY,KP,KN,IP,IN)

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
PARAMETER(NP=151)
INTEGER P
DIMENSION BC(IP),B0(IP),BI(IP),PM(IP,IP),VP(IN),XX(IN,IP)
DIMENSION A1(NP),A11(NP),YY(IN)
DIMENSION AA1(NP),AA11(NP),BII(IP)

DO 101 J=1,KP
  A1(J) = BC(J)-BI(J)
  AA1(J) = BC(J)-BII(J)
101 CONTINUE

```

```
DO 102 J=1,KP
  A11(J) = B0(J)-BI(J)
  AA11(J) = B0(J)-BII(J)
102 CONTINUE
```

```
AA2=0.0D0
A2 =0.0D0
DO 103 J=1,KP
  DO 104 JJ=1,KP
    A2 = PM(JJ,J)*A1(JJ)*A1(J) + A2
    AA2 = PM(JJ,J)*AA1(JJ)*AA1(J) + AA2
104 CONTINUE
103 CONTINUE
```

```
AA22=0.0D0
A22=0.0D0
DO 105 J=1,KP
  DO 106 JJ=1,KP
    A22=PM(JJ,J)*A11(JJ)*A11(J) + A22
    AA22 = PM(JJ,J)*AA11(JJ)*AA11(J) + AA22
106 CONTINUE
105 CONTINUE
```

```
A3=0.0D0
A4=0.0D0
DO 107 I=1,KN
  A3=0.0D0
  DO 108 J=1,KP
    A3=XX(L,J)*BC(J)+A3
108 CONTINUE
  A4=VP(I)*(YY(I)*A3 - DLOG(1.0D0 + DEXP(A3))) + A4
107 CONTINUE
```

```
A = -A2/2.0D0 + (A4/S)
AA = -AA2/2.0D0
EU = A-AA
```

```
A33=0.0D0
A44=0.0D0
DO 109 I=1,KN
  A33=0.0D0
  DO 110 J=1,KP
    A33=XX(L,J)*B0(J)+A33
110 CONTINUE
  A44=VP(I)*(YY(I)*A33 - DLOG(10.0D0 + DEXP(A33))) + A44
109 CONTINUE
```

```
B = -A22/2.0D0 + (A44/S)
BB = -AA22/2.0D0
EH = B-BB
```

```
RETURN
END
```

C

```

C*****
C      SUBROUTINAS DEL PROGRAMA Predic.F
C*****
C
C      SUBROUTINE VECTOR_C(A,TETA,XX,PM,VP,KP,KN,LP,LN,IB)
C
C      ESTA SUBROUTINA GENERA EL VECTOR A(KN). QUE CONTIENE LAS
C      CORRELACIONES ENTRE EL VECTOR VP(KP) Y LOS VECTORES DE
C      LA MATRIZ XX(KN,KP)
C      KP,KN -----> DIMENSIONES REALES.
C      LP,LN -----> DIMENSIONES DECLARADAS COMO
C                      EN EL PRINCIPAL.
C      IB -----> BANDERA PARA INDICAR SI HAY INTERCEPTO.
C
C
C      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
C      DIMENSION A(LN),XX(LN,LP),PM(LP),VP(LP)
C      PARAMETER(NNP=150)
C      DIMENSION A1(NNP)
C
C      DO 1 I=1,KN
C        DO 2 K=(1+IB),KP
C          A1(K)=VP(K)-XX(I,K)
C        2 CONTINUE
C        A2=0.0D0
C        DO 3 KK=(1+IB),KP
C          A2=PM(KK)*A1(KK)*A1(KK) + A2
C        3 CONTINUE
C      A(I)=DEXP(-TETA*A2)
C
C      1 CONTINUE
C
C      RETURN
C      END
C
C*****
C      SUBROUTINAS COMPARTIDAS
C*****
C
C      SUBROUTINE MULTNORM(Y,N,ND,T,S,IX,IY,IZ)
C
C      GENERATES A RANDOM VECTOR OF SIZE N
C      FROM A MULTIVARIATE NORMAL DISTRIBUTION WITH
C      MEAN VECTOR T AND COVARIANCE MATRIX S
C
C
C      INPUT: N -----> SAMPLE SIZE
C             ND -----> DIMENSION OF Z, T AND S AS
C                      DECLARED IN MAIN PROGRAM
C             T -----> MEAN VECTOR (SIZE N)
C             S -----> COVARIANCE MATRIX (ORDER NxN)
C             IX,IY,IZ --> SEEDS (0 < IX,IY,IZ < 30000)
C
C      OUTPUT: Y -----> VECTOR OF SIZE N CONTAINING THE
C                      MULTIVARIATE NORMAL RANDOM VECTOR
C
C
C      NOTES: 1. SEED ONCE ONLY
C            2. AUXILIARY SUBROUTINE --> UNIF
C
C      *** LAST UPDATING: 13 FEB 1992 ***
C
C

```

```

C  NOTA: EN ESTA SUBROUTINA LA DIMENSION DE LA NORMAL MULTIVARIADA
C  (TERCER ENTRADA EN LA SUBROUTINA,ND) DEBE SER MENOR O IGUAL AL
C  VALOR DEL PARAMETRO NNP
C
C
C  IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C  PARAMETER(NNP=151)
C
C  DIMENSION Y(ND), T(ND), S(ND,ND)
C  DIMENSION Z(NNP), U(NNP,NNP), PP(NNP)
C
C  *** PERFORM CHOLESKY FACTORIZATION S=U*U ***
C  *** WHERE U IS UPPER TRIANGULAR, USING ***
C  *** IMSL SUBROUTINE DLCHRG. ***
C
C  *** PERFORM CHOLESKY FACTORIZATION S=U*U' ***
C  *** WHERE UX IS LOWER TRIANGULAR ***
C
C  CALL CHOLDC(U,S,N,ND)
C *****
C  *** GENERATE A VECTOR OF N INDEPENDENT ***
C  *** STANDARD NORMAL VARIATES AND TRANSFORM ***
C
C  CALL NORMAL(Z,N,NNP,IX,IY,IZ)
C
C  CALL MULTMV(PP,U,Z,N,N,NNP,NNP)
C
C  DO 300 I=1,N
C  Y(I)=T(I)+PP(I)
C  300 CONTINUE
C *****
C
C  RETURN
C  END
C
C  SUBROUTINE NORMAL(Z,N,ND,IX,IY,IZ)
C
C  GENERATES N INDEPENDENT STANDARD NORMAL VARIATES, USING
C  BOX & MULLER METHOD < BOX & MULLER, 1950>
C
C  INPUT: N -----> SAMPLE SIZE
C  ND-----> DIMENSION OF Z AS DECLARED IN
C  MAIN PROGRAM
C  IX,IY,IZ --> SEEDS (0 < IX,IY,IZ < 30000)
C
C  OUTPUT: Z -----> VECTOR OF SIZE N
C  CONTAINING THE SAMPLE
C
C  NOTES: 1. SEED ONCE ONLY
C  2. AUXILIARY SUBROUTINE --> UNIF
C
C  *** LAST UPDATING: 13 FEB 1992 ***
C
C  IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C  DIMENSION Z(ND), UN(2)
C
C  PI=4D0*DATAN(1D0)
C  L=MOD(N,2)
C  IF(L.EQ.0) K=N
C  IF(L.EQ.1) K=N-1
C

```

```

DO 1 J=1,K,2
DO 2 J1=1,2
CALL UNIF(U,IX,IY,IZ)
UN(J1)=U
2 CONTINUE
A1=-2D0*DLOG(UN(1))
A2=2D0*PI*UN(2)
Z(J)=DSQRT(A1)*DCOS(A2)
Z(J+1)=DSQRT(A1)*DSIN(A2)
1 CONTINUE
C
IF(L.EQ.0) GOTO 4
DO 3 J1=1,2
CALL UNIF(U,IX,IY,IZ)
UN(J1)=U
3 CONTINUE
A1=-2D0*DLOG(UN(1))
A2=2D0*PI*UN(2)
IF(UN(1).LT.0.5D0) THEN
Z(N)=DSQRT(A1)*DCOS(A2)
ELSE
Z(N)=DSQRT(A1)*DSIN(A2)
ENDIF
C
4 RETURN
END
C
SUBROUTINE UNIF(U,IX,IY,IZ)
C
C GENERATES UNIFORM(0,1) RANDOM VARIATES, USING
C ALGORITHM AS183 <APPLIED STATISTICS (1982); VOL 31>
C
C INPUT: IX,IY,IZ -> SEEDS (0 < IX,IY,IZ < 30000)
C
C OUTPUT: U -----> UNIFORM RANDOM NUMBER
C
C NOTE: SEED ONCE ONLY
C
C *** LAST UPDATING: 13 FEB 1992 ***
C
C
C IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
IX=171*MOD(IX,177)-2*(IX/177)
IY=172*MOD(IY,176)-35*(IY/176)
IZ=170*MOD(IZ,178)-63*(IZ/178)
C
IF(IX.LE.0) IX=IX+30269
IF(IY.LE.0) IY=IY+30307
IF(IZ.LE.0) IZ=IZ+30323
C
UX=DBLE(IX)/30269D0
UY=DBLE(IY)/30307D0
UZ=DBLE(IZ)/30323D0
C
U=DMOD(UX+UY+UZ,DBLE(1))
C
IF(U.LE.0D0) U=1D-10
IF(U.GE.1D0) U=0.9999999999D0
C
RETURN
END
C

```

```

SUBROUTINE INVERSA(AINV,A,INDX,N,NP)
C
C ESTA SUBROUTINA CALCULA LA INVERSA DE UNA MATRIX A(NXN) Y LA
C DEPOSITA EN AINV.
C
C
C INDX <- ARREGLO AUXILIAR, SOLO SE DEBE DEFINIR EN EL PRINCIPAL.
C
C PROGRAMA TOMADO DE: NUMERICAL RECIPES IN FORTRAN.
C
C ESTA SUBROUTINA USA LAS SUBROUTINAS LUDCMP Y LUBKSB.
C
C
C IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C PARAMETER(IP=100)
C DIMENSION AINV(NP,NP),A(NP,NP),INDX(NP)
C DIMENSION AUX1(IP,IP)
C***** SE EVITA QUE LA MATRIX A SE PIERDA *****
DO 9 I=1,N
DO 10 J=1,N
AUX1(I,J)=A(L,J)
10 CONTINUE
9 CONTINUE
C*****
DO 12 I=1,N
INDX(I)=0
DO 11 J=1,N
AINV(L,J)=0.0D0
11 CONTINUE
AINV(L,I)=1.0D0
12 CONTINUE
C
CALL LUDCMP(A,N,NP,INDX,D)
C
DO 13 J=1,N
CALL LUBKSB(A,N,NP,INDX,AINV(1,J))
13 CONTINUE
C***** SE RECUPERA LA MATRIX A *****
DO 14 I=1,N
DO 15 J=1,N
A(L,J)=AUX1(I,J)
15 CONTINUE
14 CONTINUE
C*****
RETURN
END
C
SUBROUTINE LUDCMP(A,N,NP,INDX,D)
C
C IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C PARAMETER (NMAX=100,TINY=1.0D-20)
C DIMENSION A(NP,NP),INDX(NP),VV(NMAX)
C D=1.0D0
C DO 12 I=1,N
C A:AMAX=0.D0
C DO 11 J=1,N
C IF (ABS(A(I,J)).GT.A:AMAX) A:AMAX=ABS(A(I,J))
11 CONTINUE
C IF (A:AMAX.EQ.0.0D0) PAUSE 'Singular matrix.'
C VV(I)=1.0D0/A:AMAX
12 CONTINUE
C DO 19 J=1,N
C IF (J.GT.1) THEN
C DO 14 I=1,J-1

```

```

SUM=A(I,J)
IF (I.GT.1)THEN
  DO 13 K=1,I-1
    SUM=SUM-A(I,K)*A(K,J)
13  CONTINUE
  A(I,J)=SUM
ENDIF
14 CONTINUE
ENDIF
AAMAX=0.0D0
DO 16 I=J,N
  SUM=A(I,J)
  IF (J.GT.1)THEN
    DO 15 K=1,J-1
      SUM=SUM-A(I,K)*A(K,J)
15  CONTINUE
  A(I,J)=SUM
ENDIF
DUM=VV(I)*ABS(SUM)
IF (DUM.GE.AAMAX) THEN
  IMAX=I
  AAMAX=DUM
ENDIF
16 CONTINUE
IF (J.NE.IMAX)THEN
  DO 17 K=1,N
    DUM=A(IMAX,K)
    A(IMAX,K)=A(J,K)
    A(J,K)=DUM
17  CONTINUE
  D=-D
  VV(IMAX)=VV(J)
ENDIF
INDX(J)=IMAX
IF (J.NE.N)THEN
  IF (A(J,J).EQ.0.0D0)A(J,J)=TINY
  DUM=1.0D0/A(J,J)
  DO 18 I=J+1,N
    A(I,J)=A(I,J)*DUM
18  CONTINUE
ENDIF
19 CONTINUE
IF (A(N,N).EQ.0.0D0)A(N,N)=TINY
RETURN
END
C
SUBROUTINE LUBKSB(A,N,NP,INDX,B)

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
DIMENSION A(NP,NP),INDX(NP),B(NP)
II=0
DO 12 I=1,N
  LL=INDX(I)
  SUM=B(LL)
  B(LL)=B(I)
  IF (II.NE.0)THEN
    DO 11 J=II,I-1
      SUM=SUM-A(I,J)*B(J)
11  CONTINUE
  ELSE IF (SUM.NE.0.0D0) THEN
    II=I
  ENDIF
  B(I)=SUM
12 CONTINUE

```

```

DO 14 I=N,1,-1
SUM=B(I)
IF(I.LT.N)THEN
DO 13 J=I+1,N
SUM=SUM-A(I,J)*B(J)
13 CONTINUE
ENDIF
B(I)=SUM/A(I,I)
14 CONTINUE
RETURN
END

```

C
SUBROUTINE TRANSP(AT,AA,KN,KP,IN,IP)

C ESTA SUBRUTINA GENERA LA MATRIZ TRANSPUESTA AT(KP X KN),
C DE LA MATRIZ AA(KN X KP)

C
C INPUT: KN-----> DIMENSION REAL
C KP-----> DIMENSION REAL
C IN-----> DIMENSION DECLARADA COMO EN EL PRINCIPAL
C IP-----> DIMENSION DECLARADA COMO EN EL PRINCIPAL

IMPLICIT DOUBLE PRECISION (A-H,O-Z)
DIMENSION AT(IP,IN), AA(IN,IP)

```

DO 200 J=1,KP
DO 210 I=1,KN
AT(J,I) = AA(I,J)
210 CONTINUE
200 CONTINUE
RETURN
END

```

C
SUBROUTINE MULMMDIA(AA,BB,VD,KN,KP,IN,IP)

C ESTA SUBRUTINA MULTIPLICA UNA MATRIZ BB(KN X KP),
C POR UNA MATRIZ DIAGONAL(KP X KP) CUYO VECTOR DIAGONAL
C ES VD(KP X 1), Y EL RESULTADO ES LA MATRIZ AA(KN X KP)

C
IMPLICIT DOUBLE PRECISION (A-H,O-Z)
DIMENSION AA(IN,IP),BB(IN,IP),VD(IP)

```

DO 200 I=1,KN
DO 210 J=1,KP
AA(I,J) = BB(I,J)*VD(J)
210 CONTINUE
200 CONTINUE
RETURN
END

```

SUBROUTINE MULTMV(AA,BB,VV,KN,KP,IN,IP)

C THIS SUBROUTINE MULTIPLY MATRIX BB (KN x KP) AND
C VECTOR VV (KPx1). PRODUCT IN AA (KNx1).
C KN,KP: ACTUAL DIMENSIONS.
C IN,IP: DIMENSIONS AS DECLARED IN THE MAIN PROGRAM.
C

IMPLICIT DOUBLE PRECISION (A-H,O-Z)

DIMENSION AA(IN), BB(IN,IP), VV(IP)

DO 550 I=1,KN

SUM=0.0D0

DO 560 J=1,KP

SUM=SUM+(BB(I,J)*VV(J))

560 CONTINUE

AA(I)=SUM

550 CONTINUE

RETURN

END

C

SUBROUTINE MULTMM(AA,BB,CC,KN,KP,KM,IN,IP,IM)

C

C THIS SUBROUTINE MULTIPLY MATRICES BB (KNxKP) AND CC (KPxKM).

C PRODUCT IN AA (KNxKM).

C KN,KP,KM: ACTUAL DIMENSIONS.

C IN,IP,IM: DIMENSIONS AS DECLARED IN THE MAIN PROGRAM.

IMPLICIT DOUBLE PRECISION (A-H,O-Z)

DIMENSION AA(IN,IM), BB(IN,IP), CC(IP,IM)

DO 550 I=1,KN

DO 555 J=1,KM

SUM=0.0D0

DO 560 L=1,KP

SUM = SUM+(BB(I,L)*CC(L,J))

560 CONTINUE

AA(I,J) = SUM

555 CONTINUE

550 CONTINUE

RETURN

END

C

SUBROUTINE PPRINT(A,N,NN,ND,NP)

C

C ESTA SUBROUTINA ESCRIBE LA MATRIZ DE A DE NXNN

IMPLICIT DOUBLE PRECISION (A-H,O-Z)

DIMENSION A(ND,NP)

DO 2 I=1,N

WRITE(*,5000) (A(I,J), J=1,NN)

2 CONTINUE

5000 FORMAT(5D15.5,2X)

RETURN

END

Bibliografía

1. Achcar, J. A. y Smith, A. F. M. (1989). Aspects of reparametrisation in approximate Bayesian inference. *Bayesian and Likelihood Methods in Statistics and Econometrics: Essay in Honor of George A. Barnard* (eds. J. S. Geisser *et al.*), pp 439-452. Amsterdam: North-Holland.
2. Berger, J. O. y Pericchi, L. R. (1996). The intrinsic Bayes Factor for Model Selection and Prediction. *J. Am. Statist. Assoc.*, **91**, 109-122.
3. Bernardo, J. M. y Smith, A. F. M. (1994). *Bayesian Theory*. Chichester: Wiley.
4. Blight, B. J. N. y Ott, L. (1975). A Bayesian Approach to Model Inadequacy for Polynomial Regression. *Biometrika*, **62**, 79-88.
5. Burden, L. R. y Faires, J. D. (1989). *Numerical Analysis*. PWS-KENT Publishing Company.
6. Dellaportas, P. y Smith, A. F. M. (1993). Bayesian Inference for Generalised Linear and Proportional Hazard Model via Gibbs Sampling. *Appl. Statist.*, **42**, 443-459.
7. Donald, M. M. (1989). *A Crash Course in Fortran 77*. London: Edward Arnold.
8. Everitt, B. S. (1996). *A Handbook of Statistical Analyses using S-plus*. London: Chapman & Hall.
9. Geman, S. y Geman, D. (1984). Stochastic relaxation, Gibbs distributions

- and the Bayesian restoration of images. *IEEE Trans. Patt Anal. Mach. Intell.*, **6**, 721-741
10. Gutiérrez-Peña, E. (1997a). Métodos Computacionales en la Inferencia Bayesiana. Monografía, Vol. 6, No. 5. IIMAS, UNAM.
 11. Gutiérrez-Peña, E. (1997b). A Bayesian Predictive Semiparametric Approach to Variable Selection and Model Comparison in Regression. *Proceedings of the 51st Session of the ISI*. Invited Papers, Book 1. Istanbul, Turkey, pp. 17-29.
 12. Gutiérrez-Peña, E. y Smith, A. F. M. (1997). Exponential and Bayesian Conjugate Families: Review and Extensions (with discussion). *Test*, **6**, 1-90.
 13. Gutiérrez-Peña, E. y Smith, A. F. M. (1998). Aspects of Smoothing and Model Adequacy in Generalised Regression. *Journal of Statistical Planning and Inference*, **67**, 273-286.
 14. Gutiérrez-Peña, E. y Walker, S. G. (1996). A Bayesian Predictive Approach to Model Selection. *Technical Report, TR9614*. Department of Mathematics, Imperial College London.
 15. Hasting, W.K. (1970). Monte Carlo sampling methods using Markov Chain and their applications. *Biometrika*, **57**, 97-109.
 16. Howard, M. T. y Karlin, S. (1984). *An Introduction to Stochastic Modeling*. San Diego: Academic Press, Inc.
 17. Hurvich, M. C. y Tsai, C. (1995). Model Selection for Extended Quasi-Likelihood Models in Small Samples. *Biometrics*, **51**, 1077-1084.
 18. Kass, R. E., Tierney, L. y Kadane, J. B. (1988). Asymptotic in Bayesian computations (with discussion). *Bayesian Statistics 3* (eds. J. M. Bernardo *et al.*), pp 261-278. Oxford: University Press.

19. Kass, R. E., Tierney, L. y Kadane, J. B. (1989a). The validity of posterior expansion based on Laplace's method. *Bayesian and Likelihood Methods in Statistics and Econometrics: Essay in Honor of George A. Barnard* (eds. J. S. Geisser *et al.*), pp 473-488. Amsterdam: North-Holland.
20. Kass, R. E., Tierney, L. y Kadane, J. B. (1989b). Aproximate methods for aasessing influence and sensitivity in Bayesian Analysis. *Biometrika*, **76**, 663-674.
21. Kass, R. E., Tierney, L. y Kadane, J. B. (1991). Laplace's method in Bayesian Analysis. *Statistical Multiple Integration* (eds N. Flournoy *et. al.*), pp 89-99. Providence: ASA.
22. Kuo, L. y Mallick, B. (1994). Variable Selection for Regression Models. *Technical Report, TR9426*. Department of Mathematics, Imperial College London.
23. Knuiman, M. W. y Speed, T. P. (1988). Incorporating prior information into the analysis of contingency tablas. *Biometrics*, **44**, 1061-1071.
24. Lee, E. T. (1980). *Statistical Methods for Survirval Data Analysis*. Lifetime Learning Publications. Belmont, California.
25. McCullagh, P. y Nelder, J. A. (1989). *Generalized Linear Models*. London: Chapman & Hall.
26. Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. y Teller, E. (1953). Equations of state calculations by fast computing machine. *J. Chem. Phys.*, **21**, 1087-1091.
27. Morris, C. N. (1982). Natural exponential families with quadratic variance functions. *The Annals of Statist.*, **10**, 65-80.
28. Morris, C. N. (1988). Approximating posterior distributions and posterior moments (with discussion). *Bayesian Statistics 3* (eds. J. M. Bernardo *et. al.*), pp 327- 344. Oxford: University Press.

29. Muller, P. (1991). A generic approach to posterior integration and Gibbs sampling.
30. O'Hagan, A. (1978). Curve fitting and optimal design for prediction (with discussion). *J. of the Roy. Statistical Society B*, **40**, 1-42.
31. O'Hagan, A. (1992). Some Bayesian numerical analysis (with discussion). *Bayesian Statistics 4* (eds. J. M. Bernardo *et al.*), pp 345-363. Oxford: University Press.
32. O'Hagan, A. (1994). *Kendall's Advanced Theory of Statistics*. Vol. 2B. Bayesian Inference. London: Edward Arnold.
33. O'Hagan, A. (1995). Fractional Bayes factors for model comparison (with discussion). *J. of the Roy. Statistical Society B*, **57**, 99-138.
34. Press, H. W., Flannery, P. B., Teukolsky, A. S. y Vetterling, T. W. (1988). *Numerical Recipes: the art of scientific computing*. Cambridge: University Press.
35. Qian, G., Gabor, G. y Gupta, R. P. (1996). Generalised linear model selection by the predictive least quasi-deviance criterion. *Biometrika*, **83**, 41-54.
36. Raftery, E. A. (1994). Approximate Bayes Factors and Accounting for Model Uncertainty in Generalized Linear Models. *Technical Report, TR-255*. Department of Statistics, University of Washington.
37. Ripley, B. D. (1987). *Stochastic Simulation*. New York: J. Wiley.
38. Roos, S. M. (1990). *A course in simulation*. New York: Macmillan.
39. Rubin, D. B. (1988). Using the SIR algorithm to simulate posterior distributions (with discussion). *Bayesian Statistics 3* (eds. J. M. Bernardo *et al.*), pp 395-402. Oxford: University Press.
40. Sacks, J., Welch, W. J., Mitchell, T. J. y Wynn, H. P. (1989). Design and Analysis of Computer Experiments. *Statist. Sci.*, **4**, 409-435.

41. San Martini, A. y Spezzaferri, F. (1984). A predictive Model Selection Criterion. *J. of the Roy. Statistical Society B*, **46**, 296-303.
42. Smith, A. F. M. (1991). Bayesian computational methods. *Phil. Trans. Roy. Soc. London A*, **337**, 369-386.
43. Smith, A. F. M. y Gelfand, A. E. (1992). Bayesian statistics without tears: a sampling-resampling perspective. *Amer. Statist.*, **46**, 84-88.
44. Smith, A. F. M. y Roberts G. O. (1993). Bayesian Computation via the Gibbs Sampler and Related Markov Chain Monte Carlo Methods. *J. of the Roy. Statistical Society B*, **55**, 3-23.
45. Spiegelhalter, J. D. y Smith, A. F. M. (1982). Bayes Factors for Linear and Log-linear Models with Vague Prior Information. *J. of the Roy. Statistical Society B*, **44**, 377-387.
46. Stewart, L. (1979). Multiparameter univariate Bayesian analysis. *J. Am. Statist. Assoc.*, **74**, 684-693.
47. Stewart, L. (1983). Bayesian Analysis using Monte Carlo Integration, a powerfull methodology for handling some difficult problems. *The Statistician*, **32**, 195-200.
48. Stewart, L. (1985). Multiparameter Bayesian Inference using Monte Carlo Integration, some techniques for bivariate analysis. *Bayesian Statistics 2* (eds. J. M. Bernardo *et al.*), pp 495-510. Amsterdam: North-Holland.
49. Stewart, L. (1987). Hierarchical Bayesian analysis using Monte Carlo Integration: computing posterior distributions when there are many models. *The Statistician*, **36**, 211-219.
50. Stewart, L. y Davis, W. W. (1986). Bayesian posterior distributions over sets of possible models with inferences computed by Monte Carlo Integration. *The Statistician*, **35**, 175-182.
51. Tierney, L. (1994). Markov Chains for exploring posterior distributions. *The Annals of Statist.*, **22**, 4, 1701-1762.