Casa abierta al tiempo
UNIVERSIDAD AUTÓNOMA METROPOLITANA
Unidad Iztapalapa

DIVISIÓN DE CIENCIAS BÁSICAS E INGENIERÍA

MODELADO DE DISTRIBUCIONES CONJUNTAS PARA MODELOS LINEALES GENERALIZADOS CON DATOS FALTANTES

Tesis que presenta

LUIS CARLOS PÉREZ RUIZ

Para obtener el grado de

DOCTOR EN CIENCIAS (MATEMÁTICAS)

Asesor

Dr. GABRIEL ESCARELA PÉREZ

Sinodales

Dra. SILVIA RUIZ-VELASCO ACOSTA

Dr. ALBERTO CASTILLO MORALES

Dr. GABRIEL ESCARELA PÉREZ

Dr. ERNESTO JUVENAL BARRIOS ZAMUDIO

Dr. CARLOS ERWIN RODRÍGUEZ HERNÁNDEZ-VELA

Ciudad de México, a 28 de junio de 2018

Índice

Resumen

La falta de datos en las variables explicativas de los modelos lineales generalizados es un problema común que se ha estudiado por muchos años y se han propuesto diversos métodos para enfrentarlo. Entre estos métodos, un procedimiento basado en modelos como lo es máxima verosimilitud, representa una metodología de estimación de parámetros sólida y flexible, ya que la función de verosimilitud está disponible en forma computable. Sin embargo, para lograr esto último, es necesario modelar adecuadamente las distribuciones conjuntas tanto de las variables explicativas parcialmente observadas, como de las correspondientes variables indicadoras de pérdida de datos. En este trabajo, se propone una nueva metodología basada en modelos para el análisis de regresión de modelos lineales generalizados cuando las variables explicativas parcialmente observadas son categóricas. La propuesta consiste en usar construcciones con cópulas pareadas bivariadas como una herramienta versátil para facilitar el modelado de distribuciones conjuntas multivariadas de alta dimensión. De esta manera, los parámetros del modelo pueden ser estimados maximizando la función log-verosimilitud mediante el uso del algoritmo EM vía ponderaciones. Para la estimación de los errores estándares se usa el método de matriz de información observada.

Con el fin de comparar el desempeño de la metodología propuesta con otros enfoques ya bien establecidos, incluyendo análisis con casos completos e imputación múltiple, se llevaron a cabo varios experimentos de simulación bajo diferentes escenarios de pérdida de datos, tanto aleatoria como no aleatoria. Adicionalmente, se realizaron simulaciones con variables respuesta tipo Binomial, Poisson y Normal, utilizando para ello diversas estructuras de dependencia entre las variables explicativas con datos faltantes y entre las variables indicadoras. Además, para ilustrar la viabilidad práctica de los métodos planteados, se realizó el modelado de datos del ensayo clínico E1684 sobre un melanoma en fase III, y esto se comparó con lo obtenido mediante imputación múltiple y con el software *LogXact*. También se efectuaron los correspondientes análisis de sensibilidad y diagnósticos para evaluar las suposiciones hechas acerca del modelo. Los resultados de las simulaciones y de la aplicación muestran que la metodología aquí propuesta es robusta y flexible, representando una alternativa competitiva a las técnicas tradicionales. Finalmente, se plantean como temas de investigación a futuro, tanto la mejora computacional del método como la inclusión de variables explicativas continuas con datos faltantes.

Capítulo 1

Introducción

Tarde o temprano (usualmente temprano) todo analista estadístico se encontrará con datos faltantes en los conjuntos de datos a analizar. Tradicionalmente, un conjunto de datos es una matriz rectangular, donde los renglones representan, dependiendo del contexto, unidades experimentales, casos, observaciones o sujetos, mientras que las columnas representan variables explicativas o variables respuesta para cada unidad experimental. Las entradas de la matriz suelen ser números reales que representan las categorías o los valores cuantitativos de las variables para los sujetos correspondientes. En un conjunto de datos típico algunas de dichas entradas para las variables explicativas carecen de valores medidos (datos faltantes) por muy diversas razones. Pero, ¿por qué los datos faltantes son un problema? Porque los métodos estadísticos estándar y la mayoría del software estadístico disponible han sido desarrollados para analizar conjuntos de datos completos; esto es, los modelos presuponen que todos los casos tienen información en todas las variables incluidas en el análisis, que si no toma en cuenta a los datos faltantes, puede generar estimadores sesgados y poco eficientes (Horton and Kleinman, 2007). Por ejemplo, Van Buuren (2012) afirma que el procedimiento estándar para los casos con valores faltantes es eliminarlos. Para ilustrar esto, cita que Hand et al. (1994) publicaron una colección de pequeños conjuntos de datos encontrados en la literatura estadística. Pero sólo 13 de los 510 conjuntos de datos en la colección tienen un código para identificar a los datos faltantes. Para el resto, el problema de los valores faltantes probablemente ha sido "resuelto" de alguna manera. Sin embargo, Van Buuren descubrió que cierto conjunto de datos completo de 34 sujetos en el libro de Hand, originalmente se componía de 39 sujetos. Entonces concluye que es razonable asumir que la eliminación de casos con valores faltantes ocurrió silenciosamente en muchos de los otros conjuntos de datos.

1.1. Clasificación de los datos faltantes

Rubin (1976) introdujo un sistema de clasificación para datos faltantes que aún hoy en día es ampliamente usado en la literatura estadística. De su trabajo resultaron los llamados *mecanismos de pérdida de datos* que abordan de manera conceptual la siguiente cuestión: el hecho de que una variable tenga datos faltantes ¿está relacionado con los valores subyacentes de alguna o algunas de las variables en el conjunto de datos? Entender estos mecanismos es crucial ya que las propiedades de los diferentes métodos para manipular datos faltantes están estrechamente relacionadas con el tipo de mecanismo de pérdida presente en el conjunto de datos.

Considérese el conjunto de datos de observaciones independientes $\{\mathbf{x}_i, \mathbf{z}_i, y_i\}$, con $i = 1, \ldots, n$ mostrado en la Figura 1.1, donde $\mathbf{x}_i$ es un vector de variables explicativas completamente observadas, $\mathbf{z}_i$ es un vector m-dimensional de variables explicativas parcialmente observadas, y y_i es la variable respuesta observada. Sea $\mathbf{r}_i$ un vector m-dimensional de variables indicadoras de datos faltantes, cuya k-ésima componente r_{ik} es igual a 1 si la k-ésima componente de $\mathbf{z}_i$ es valor observado y 0 si es valor faltante, con $k = 1, \ldots, m$. Debido a que la falta en una observación ocurre al azar, $\mathbf{r}$ es un vector aleatorio y se le asociara una distribución de probabilidad conjunta dada por $f_{\mathbf{r}}(\mathbf{r}|\mathbf{w}; \boldsymbol{\nu})$, donde $\mathbf{w} = (\mathbf{x}, \mathbf{z}, y)$ y $\boldsymbol{\nu}$ es el vector de parámetros que caracteriza a la distribución conjunta.

	$\mathbf{x}$	z_1	z_2	$\cdots$	z_m	y	r_1	r_2	$\cdots$	r_m
1	•	•	–		–	•	1	0		0
2	•	•	•		•	•	1	1		1
3	•	–	–		•	•	0	0		1
4	•	–	•		–	•	0	1		0
5	•	–	–		–	•	0	0		0
6	•	–	•		•	•	0	1		1
7	•	•	•		–	•	1	1		0
⋮										
n	•	•	–		•	•	1	0		1

Figura 1.1. Conjunto de datos con valores observados (•) y valores faltantes (–).

El primer mecanismo de datos faltantes es el completamente aleatorio (MCAR, por sus siglas en inglés), y se define como aquel en el que la probabilidad de que falte un valor en cierta variable es totalmente independiente de los valores de todas las variables en el conjunto de datos; es decir, de manera formal $f_{\mathbf{r}}(\mathbf{r}|\mathbf{w}; \boldsymbol{\nu}) = f(\mathbf{r}|\boldsymbol{\nu})$. Aunque MCAR es un supuesto bastante restrictivo y pareciera

poco práctico, hay ocasiones en que es razonable; por ejemplo, cuando los datos son eliminados porque una variable en particular es muy costosa de medir. Aquí la estrategia consiste en medir dicha variable sólo para un subconjunto aleatorio de la muestra completa, implicando que los datos son MCAR para el resto de la muestra. A modo de ejemplo de simulación, supóngase que la variable indicadora r_k tiene una distribución dada por $f(r_k|\mathbf{w};\mathrm{p}) = \mathrm{p}^{r_k}(1-\mathrm{p})^{1-r_k}$, i.e. Bernoulli(p) con p constante, entonces la pérdida de datos en su variable explicativa correspondiente z_k es MCAR.

El segundo mecanismo de datos faltantes es el ignorable o aleatorio (MAR, por sus siglas en inglés), y se define como aquel en el que la probabilidad de que falte un valor en cierta variable es independiente de los valores de la variable misma, aunque podría o no depender de los valores de las demás variables; esto es, $f_{\mathbf{r}}(\mathbf{r}|\mathbf{w};\boldsymbol{\nu}) = f(\mathbf{r}|\mathbf{x}, y;\boldsymbol{\nu})$. Ahora bien, en el caso de que tampoco haya dependencia de $\mathbf{r}$ con ninguna de las demás variables nos remite al mecanismo MCAR; es decir, MCAR sólo es un caso particular de MAR, donde éste último es menos restrictivo. Nótese que a pesar de su nombre, MAR no significa que los datos faltantes sean una muestra aleatoria simple de todos los datos, como sucede con el mecanismo MCAR. Un problema práctico importante con el mecanismo MAR es que no existe manera para validarlo; esto es, no se puede verificar si la probabilidad de valores faltantes en cierta variable es independiente de sus propios valores porque, precisamente, no están disponibles. El término ignorable se refiere al hecho de que para realizar un análisis de datos con valores faltantes MAR no es necesario modelar el mecanismo de pérdida subyacente, o en otra palabras, se puede ignorar dicho mecanismo de pérdida. Como ilustración, supóngase que la variable indicadora r_k tiene una distribución dada por $f(r_k|\mathbf{w};\mathrm{p}) = \mathrm{p}^{r_k}(1-\mathrm{p})^{1-r_k}$, con $\mathrm{p} = \exp(\mathbf{x}+y)/(1+\exp(\mathbf{x}+y))$, entonces la pérdida de datos en su variable explicativa correspondiente z_k es MAR.

El tercer mecanismo de datos faltantes es el no-ignorable o no-aleatorio (MNAR o NMAR, por sus siglas en inglés), y se define como aquel en el que la probabilidad de que falte un valor en cierta variable depende de los valores de la variable misma, y podría o no depender de los valores de las demás variables; esto es, $f_{\mathbf{r}}(\mathbf{r}|\mathbf{w};\boldsymbol{\nu}) = f(\mathbf{r}|\mathbf{w};\boldsymbol{\nu})$. Inferencias válidas bajo MNAR generalmente requieren especificar un modelo plausible para el mecanismo de pérdida de datos, de ahí el término no-ignorable. Al igual que con el mecanismo MAR, no hay manera de verificar si la pérdida de datos es MNAR sin conocer los valores de dichos datos. Por esta razón, un componente clave en un análisis con datos faltantes MNAR es hacer un análisis de sensibilidad, lo cual implica ajustar diferentes modelos del mecanismo de pérdida para examinar que tan sensibles son los resultados

a los diferentes modelos propuestos. Como ejemplo de este mecanismo de pérdida, supóngase que la variable indicadora r_k tiene una distribución dada por $f(r_k|\mathbf{w};\mathrm{p}) = \mathrm{p}^{r_k}(1-\mathrm{p})^{1-r_k}$, con $\mathrm{p} = \exp(\mathbf{x} + z_k + y)/(1 + \exp(\mathbf{x} + z_k + y))$, entonces la pérdida en z_k es MNAR.

A modo de ilustración adicional, considérese el conjunto de datos mostrado en la Tabla 1.1.

Tabla 1.1. Eliminación de valores de z bajo los tres mecanismos de pérdida.

		z		
x	z	**MCAR**	**MAR**	**MNAR**
78	9	–	–	9
84	13	13	–	13
84	10	–	–	10
85	8	8	–	–
87	7	7	–	–
91	7	7	7	–
92	9	9	9	9
94	9	9	9	9
94	11	11	11	11
96	7	–	7	–
99	7	7	7	–
105	10	10	10	10
105	11	11	11	11
106	15	15	15	15
108	10	10	10	10
112	10	–	10	10
113	12	12	12	12
115	14	14	14	14
118	16	16	16	16
134	12	–	12	12

En las tres últimas columnas de la tabla anterior se reproduce la misma variable z pero con ciertos valores faltantes, los cuales fueron eliminados de acuerdo a los siguientes modelos de censura. Sea r la variable indicadora de z, entonces $f(r|x,z;\mathrm{p}) = \mathrm{p}^r(1-\mathrm{p})^{1-r}$, donde $\mathrm{p} = 0.75$ para MCAR,

$$\mathrm{p} = \begin{cases} 0 & \text{si} \quad x \leq 90 \\ 1 & \text{si} \quad x > 90 \end{cases} \quad \text{para MAR,} \quad \text{y} \quad \mathrm{p} = \begin{cases} 0 & \text{si} \quad z \leq 8 \\ 1 & \text{si} \quad z > 8 \end{cases} \quad \text{para MNAR.}$$

Los datos faltantes se han estudiado por décadas y se han propuesto docenas de métodos para enfrentar el problema. Muchos de estos métodos son actualmente de uso generalizado, mientras que otros son ahora poco más que pies de página históricos. A continuación se describen de manera breve algunos de los métodos tradicionales desde los más simples hasta los más elaborados.

1.2. Métodos basados en eliminación

1.2.1. Análisis con casos completos

Conocido en inglés como *complete-case analysis* o *listwise deletion* es el método por defecto para tratar datos incompletos en muchos paquetes estadísticos, incluyendo `SPSS`, `SAS` y `Stata`. La función `na.omit()` hace lo mismo en `S-PLUS` y `R`. El procedimiento elimina todos los casos con uno o más valores faltantes en las variables bajo análisis. Su principal ventaja es la conveniencia ya que no requiere técnicas complejas. Si la pérdida de datos es MCAR, este método produce estimadores insesgados de medias, varianzas y coeficientes de regresión. Además, bajo MCAR, genera errores estándares que son adecuados para el subconjunto de datos reducido, pero a menudo son mayores con respecto al conjunto de datos completo, ya que el tamaño de la muestra es un componente clave en el cálculo de los mismos. Una desventaja de este método es que es potencialmente dispendioso. No es raro en aplicaciones reales que más de la mitad de la muestra original se pierda, especialmente si el número de variables es grande. Es claro que una submuestra muy pequeña podría degradar seriamente la habilidad para detectar los efectos de interés (Van Buuren, 2012).

Si el mecanismo de pérdida no es MCAR, un análisis con casos completos puede generar estimadores bastante sesgados de medias, coeficientes de regresión y correlaciones. Little and Rubin (2002) muestran que el sesgo en la media estimada se incrementa con la proporción de datos faltantes. Schafer and Graham (2002) hacen simulaciones que ilustran el sesgo del método bajo MAR y MNAR. Sin embargo, ellos también afirman que este método no siempre es malo, ya que si un problema de datos faltantes puede ser resuelto eliminando sólo una parte pequeña de la muestra, entonces el método puede ser bastante efectivo. Sin embargo, muchos metodólogos son cautelosos de dar consejos acerca del porcentaje de casos incompletos por debajo del cual es razonable utilizar esta estrategia. Little and Rubin (2002) argumentan que es difícil formular reglas generales dado que las consecuencias de usar casos completos dependen en más que sólo la proporción de datos faltantes. En el contexto de análisis de regresión, este procedimiento posee algunas propiedades únicas que lo hacen atractivo en situaciones particulares. Hay casos en los cuales un análisis con casos completos puede proveer mejores estimadores que aún los procedimientos más sofisticados. Por ejemplo, el método puede producir estimadores insesgados de coeficientes de regresión bajo cualquier mecanismo de datos faltantes, siempre y cuando la pérdida de datos esté en función de las variables explicativas y no de la variable respuesta (Little, 1992).

1.2.2. Análisis con casos disponibles

Este método también llamado en inglés *available-case analysis* o *pairwise deletion* intenta mitigar la pérdida de información que se produce con el método de casos completos. La idea detrás de este método es calcular el vector de medias y la matriz de (co)varianzas usando todos los casos con valores disponibles. Esto es, la media de cierta variable Z_1 se calcula con todos los casos con valores observados en Z_1, la media de una variable Z_2 se calcula con todos los casos con valores observados en Z_2, y así sucesivamente. Para calcular la covarianza y la correlación entre dos variables Z_1 y Z_2, todos los casos que tengan valores observados tanto en Z_1 como en Z_2 serán utilizados. Después, las matrices con los estadísticos resultantes se introducen a una rutina de análisis de regresión, de análisis factorial o de cualquier otro procedimiento de modelado.

Los paquetes `SPSS`, `SAS`, `Stata`, `S-PLUS` y `R`, entre otros, contienen procedimientos con alguna opción para realizar análisis con casos disponibles. El método es simple, usa toda la información disponible y produce estimadores consistentes de medias, covarianzas y correlaciones bajo el mecanismo de pérdida MCAR. Sin embargo, los estimadores pueden ser sesgados si la pérdida de datos no es MCAR. Además, existen problemas computacionales. La matriz de correlaciones puede no ser definida positiva, que es una condición para la mayoría de los procedimientos multivariados. Correlaciones fuera del rango $[-1, +1]$ pueden ocurrir, lo cual es un problema originado por el uso de diferentes subconjuntos de casos para calcular las covarianzas y las varianzas. Tales problemas son más severos para las variables altamente correlacionadas (Little and Rubin, 2002).

Otro problema es que no es claro cuál tamaño de muestra debería ser usado para calcular los errores estándares. Por ejemplo, considérese un análisis de regresión que usa como entrada una matriz de covarianzas obtenida con este método. En este caso no hay un tamaño de muestra único aplicable a la matriz de covarianzas completa. Consecuentemente, no existe una manera directa para calcular los errores estándares, por lo que los paquetes de software estadístico utilizan diversas técnicas para aproximarlos. Algunos paquetes toman un tamaño de muestra promedio, pero es probable que este enfoque produzca errores estándares subestimados para unas variables y sobrestimados para otras (Little, 1992). Aunque usar tanta información como sea posible es ciertamente una buena idea, el propio análisis de una matriz por pares requiere técnicas sofisticadas de optimización y fórmulas complejas para calcular los errores estándares. De este modo, la simplicidad atractiva del análisis con casos disponibles como un método general para tratar datos faltantes se pierde, lo cual limita su utilidad como una buena alternativa (Van Buuren, 2012).

1.3. Métodos basados en imputación simple paramétrica

1.3.1. Imputación de la media

Conocida en inglés como *unconditional mean imputation*, toma el camino aparentemente atractivo de imputar (sustituir) los valores faltantes de cierta variable con la media aritmética de los valores observados de la misma. Los metodólogos a menudo atribuyen a Wilks (1932) esta vieja idea (citado en Enders, 2010, p. 42). Esta estrategia es atractiva porque genera conjuntos de datos completos que pueden ser analizados con metodología convencional, por lo que la comodidad es una de sus principales ventajas. Otro beneficio de esta técnica es que hace uso de todos los datos que los métodos basados en casos completos podrían descartar. Sin embargo, es claro que imputar valores en el centro de la distribución reduce la variabilidad de los datos. Como consecuencia se reducen los errores estándares, las varianzas, covarianzas y las correlaciones. A modo de ilustración, considérese la siguiente fórmula para calcular la covarianza muestral: $\hat{\sigma}_{Z_1Z_2} = \sum_i[(z_{1i} - \hat{\mu}_{Z_1})(z_{2i} - \hat{\mu}_{Z_2})]/(n-1)$. Casos con valores faltantes ya sea en Z_1, en Z_2 o en ambas, atenúan la magnitud de la covarianza muestral así calculada, ya que Z_1 y Z_2 al ser imputadas con la media contribuyen con ceros en el numerador de la fórmula anterior. Algo similar ocurre con las correlaciones que tienen a las covarianzas como numerador: $\hat{\rho} = \hat{\sigma}_{Z_1Z_2}/\hat{\sigma}_{Z_1}\hat{\sigma}_{Z_2}$. Little and Rubin (2002) proporcionan fórmulas de ajuste que producen estimadores insesgados de varianzas y covarianzas con pérdida de datos MCAR, pero estas correcciones terminan generando estimadores que son idénticos a los obtenidos con el método de casos disponibles. De hecho, estudios de simulación sugieren que la imputación de la media es posiblemente el peor método disponible para el manejo de datos faltantes. Consecuentemente, bajo ninguna circunstancia este método es defendible y deberá evitarse (Enders, 2010).

1.3.2. Imputación por regresión

Llamada en inglés *conditional mean imputation*, este método reemplaza los datos faltantes con valores estimados a partir de ecuaciones de regresión. También conocida como *método de Buck* (Buck, 1960), al igual que la imputación de la media, tiene una larga historia de cerca de 55 años (citado en Little and Rubin, 2002, p. 63). La idea básica detrás de este enfoque es intuitivamente simple: usar la información de unas variables para imputar a otras. Como las variables tienden a estar correlacionadas, tiene sentido generar imputaciones que toman información de las mismas.

El primer paso del proceso de imputación es obtener un conjunto de ecuaciones de regresión por medio de un análisis con casos completos. El segundo paso es generar valores estimados para los datos faltantes a partir de dichas ecuaciones, y así obtener un conjunto de datos completo. Por ejemplo, considérese un conjunto de datos hipotético de tres variables con valores faltantes Z_1, Z_2 y Z_3. Sin incluir los casos completos ni los casos incompletos en las tres variables, habrá seis patrones posibles de datos faltantes a tomar en cuenta: casos con valores faltantes en (1) sólo Z_1, (2) sólo Z_2, (3) sólo Z_3, (4) Z_1 y Z_2, (5) Z_1 y Z_3 y (6) Z_2 y Z_3. En la Tabla 1.2 se muestra el conjunto de las ecuaciones de regresión para los seis patrones de datos faltantes.

Tabla 1.2. Conjunto de ecuaciones de regresión.

Patrón de datos faltantes	**Ecuaciones de regresión**
Z_1	$\hat{z}_1 = \beta_0 + \beta_1 z_2 + \beta_2 z_3$
Z_2	$\hat{z}_2 = \beta_0 + \beta_1 z_1 + \beta_2 z_3$
Z_3	$\hat{z}_3 = \beta_0 + \beta_1 z_1 + \beta_2 z_2$
Z_1 y Z_2	$\hat{z}_1 = \beta_0 + \beta_1 z_3, \quad \hat{z}_2 = \beta_0 + \beta_1 z_3$
Z_1 y Z_3	$\hat{z}_1 = \beta_0 + \beta_1 z_2, \quad \hat{z}_3 = \beta_0 + \beta_1 z_2$
Z_2 y Z_3	$\hat{z}_2 = \beta_0 + \beta_1 z_1, \quad \hat{z}_3 = \beta_0 + \beta_1 z_1$

Una manera fácil de calcular los coeficientes de regresión es usando las entradas del vector de medias y de la matriz de covarianzas muestrales previamente estimadas mediante el análisis de casos completos. Aunque suena tedioso construir ecuaciones de regresión para cada patrón de datos faltantes, un algoritmo computacional llamado *sweep operator* puede automatizar este proceso.

De acuerdo a Enders (2010), aunque la imputación por regresión es superior a la imputación de la media, también genera sesgos. El hecho de que los valores imputados queden ubicados directamente sobre el hiperplano de regresión, implica una sobreestimación de las correlaciones entre las variables, además de la pérdida de variabilidad, con la consecuente subestimación de las varianzas, covarianzas y errores estándares, aunque esto en menor grado que con la imputación de la media. La magnitud de los sesgos en las varianzas y covarianzas es predecible y se han propuesto ajustes correctivos para estos parámetros. Bajo el mecanismo de pérdida MCAR estas correcciones proporcionan estimadores consistentes de la matriz de covarianzas, significando que los estimadores están cerca de sus verdaderos valores poblacionales conforme el tamaño de la muestra se incrementa. Sin embargo, no hay razón para hacer un esfuerzo adicional aplicando estas correcciones, ya que mejores métodos están disponibles (Enders, 2010).

1.3.3. Imputación por regresión estocástica

Este método también usa ecuaciones de regresión para sustituir los datos faltantes con valores estimados, pero con el paso extra de añadir a cada valor estimado un término residual normalmente distribuido. Sumando residuales a los valores imputados se restablece la pérdida de variabilidad de los datos y elimina los sesgos asociados con el esquema de imputación por regresión estándar. De hecho, ambos métodos de imputación por regresión son los únicos que proporcionan estimadores insesgados bajo el mecanismo MAR. Consecuentemente, éste es el único de los métodos ya vistos que posiblemente todavía tenga algún mérito. Aunque, al igual que cualquier técnica de imputación simple, la imputación por regresión estocástica también subestima los errores estándares.

El proceso de construcción del conjunto de ecuaciones de regresión es idéntico al descrito en el método de imputación por regresión estándar, con la salvedad de que cada ecuación de regresión requiere su propia distribución residual. Cada distribución residual es una curva normal con media cero, pero la varianza difiere a través de los patrones de datos faltantes, ya que es igual a la varianza residual de la regresión. En patrones que tienen dos o más variables con valores faltantes, la distribución residual es normal multivariada con un vector de medias cero y una matriz de covarianza igual a la matriz de covarianza residual de la regresión multivariada. Por ejemplo, reconsidérese el patrón de datos faltantes de la Tabla 1.2 donde tanto Z_1 como Z_2 tienen valores faltantes. Este patrón requiere residuales de una distribución normal multivariada con una matriz de covarianza igual a la matriz de covarianza residual de la regresión de Z_1 y Z_2 en Z_3.

Para clarificar y comparar los tres métodos basados en imputación simple paramétrica, considérese de nuevo el mismo conjunto de datos hipotético de dos variables x y z visto anteriormente, y reproducido en las dos primeras columnas de la Tabla 1.3. En la tercera columna aparece de nuevo z pero sin sus diez primeros valores, los cuales fueron eliminados en función de los valores de x ubicados por debajo de la mediana de su distribución. Con esto se simula una pérdida de datos bajo el mecanismo MAR (Enders, 2010).

Primero se utilizó imputación de la media para sustituir los valores faltantes de z con la media aritmética de los diez últimos valores de z (columna 4). Después, mediante un análisis con casos completos se hizo una regresión lineal de la z en x, obteniéndose la siguiente ecuación de regresión: $\hat{z} = -2.0646 + 0.1234\ x$. Con esta ecuación se obtuvieron los valores estimados de z que sirvieron para imputar sus valores faltantes (columna 5). De la regresión anterior se obtuvo una varianza residual de 6.65 que sirvió para generar 10 residuales aleatorios de una distribución normal con

Tabla 1.3. Ejemplo ilustrativo de los métodos de imputación simple paramétrica.

x	z	z incompleta	Imputación de la media	Imputación por regresión	Residual aleatorio	Imputación estocástica
78	9	--	11.7	7.56	1.69	9.25
84	13	--	11.7	8.31	-2.30	6.00
84	10	--	11.7	8.31	1.23	9.54
85	8	--	11.7	8.43	3.60	12.02
87	7	--	11.7	8.68	-4.21	4.47
91	7	--	11.7	9.17	-0.32	8.85
92	9	--	11.7	9.29	2.09	11.38
94	9	--	11.7	9.54	-0.63	8.91
94	11	--	11.7	9.54	1.92	11.45
96	7	--	11.7	9.79	-1.05	8.74
99	7	7	7.0	7.00	--	7.00
105	10	10	10.0	10.00	--	10.00
105	11	11	11.0	11.00	--	11.00
106	15	15	15.0	15.00	--	15.00
108	10	10	10.0	10.00	--	10.00
112	10	10	10.0	10.00	--	10.00
113	12	12	12.0	12.00	--	12.00
115	14	14	14.0	14.00	--	14.00
118	16	16	16.0	16.00	--	16.00
134	12	12	12.0	12.00	--	12.00

media cero y varianza igual a 6.65 (columna 6). Finalmente, se usó la ecuación de regresión estocástica $\hat{z} = -2.0646 + 0.1234\ x + e$, donde el término adicional e representa a los residuales aleatorios obtenidos anteriormente (columna 7).

Con el fin de visualizar el impacto que causan los tres métodos de imputación simple paramétrica en la recuperación de la variabilidad de los datos, en la Figura 1.2 se muestran los diagramas de dispersión del ejemplo previo. Nótese que con la imputación de la media los valores imputados quedan ubicados sobre una línea horizontal, lo cual implica que la correlación entre x y z es nula para estos casos. En consecuencia, este método de imputación atenúa las medidas de asociación generando un sesgo en los estimadores bajo cualquier mecanismo de pérdida de datos. No sorprende que dicho sesgo se incremente conforme la proporción de datos faltantes también lo hace. En cuanto a la imputación por regresión se observa que los valores imputados se ubican en una línea con pendiente diferente de cero. Esto implica que la correlación entre x y z es máxima en estos casos. Por lo tanto, este método de imputación conlleva al problema opuesto que la imputación de la media; es decir, sobrestima las medidas de asociación generando también sesgos bajo cualquier mecanismo de pérdida de datos. En referencia a la imputación estocástica se ve

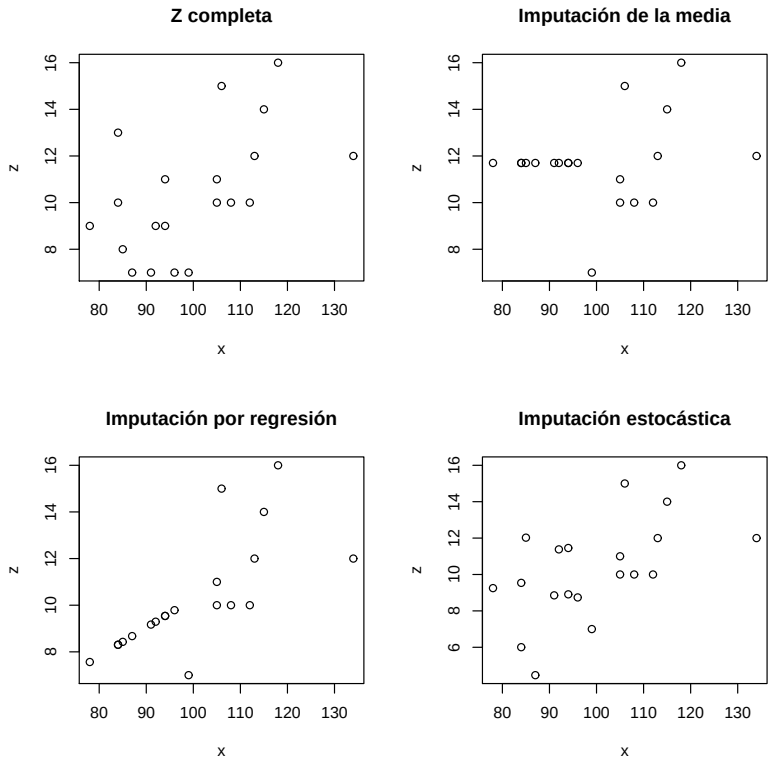

Figura 1.2. Diagramas de dispersión del ejemplo ilustrativo.

de inmediato como este método preserva hasta cierto punto la variabilidad original de los datos. Esto sugiere que la regresión estocástica genera estimadores insesgados bajo el mecanismo MAR. Aunque este pequeño ejemplo ilustrativo no provea evidencia convincente en este sentido, varios autores han usado técnicas analíticas para demostrar que este es el caso (Enders, 2010).

1.4. Métodos basados en imputación simple no paramétrica

1.4.1. Imputación Hot Deck

Esta es una colección de técnicas que imputan los datos faltantes con valores de otros casos "similares" en la misma muestra. Estadísticos del Buró de Censos en Estados Unidos desarrollaron originalmente este método para tratar con datos faltantes en bases de datos de dominio público, y este procedimiento también tiene una larga historia en aplicación de encuestas (Scheuren, 2005). Históricamente, el término *hot deck* se refiere a la vieja práctica de usar tarjetas perforadas para almacenar datos en una computadora. Con respecto a datos faltantes, se refería a seleccionar

valores de remplazo de una pila de tarjetas (deck) que están actualmente en uso (hot). En contraste, el calificativo *cold deck* se refería a usar una pila de tarjetas con datos de otra muestra.

Se han propuesto diversas variaciones dentro de esta metodología. Entre ellas están hot deck por muestreo aleatorio simple con reemplazo y hot deck dentro de grupos, que en una de sus formas más simples, sustituye cada dato faltante con un valor aleatorio de un grupo de casos con valores similares en variables correspondientes. Por ejemplo, considérese una encuesta poblacional en la cual algunos encuestados se rehúsan a declarar su ingreso. Este procedimiento clasifica a los encuestados en grupos de características demográficas similares tales como género, edad, raza o estado civil. Entonces se reemplazan los ingresos faltantes con un valor aleatorio de la distribución de ingreso de los encuestados que comparten las mismas características demográficas. Otras variaciones disponibles son hot deck del vecino más cercano y hot deck secuencial ordenado por una variable explicativa, entre otras.

La imputación hot deck generalmente preserva las distribuciones univariadas de los datos, y no atenúa la variabilidad de los datos imputados con el mismo grado que otros métodos de imputación. Sin embargo, hot deck no está hecho para estimar medidas de asociación y puede producir sesgos sustanciales en estimadores de correlaciones y coeficientes de regresión (Schafer and Graham, 2002). Al igual que otros procedimientos de imputación, hot deck subestima los errores estándares al sustituir los datos faltantes con valores que ya están en la misma muestra, lo cual reduce la variabilidad. En consecuencia, se aumenta la posibilidad de cometer un error del Tipo I; es decir, es posible que con la manipulación de datos faltantes se incremente la probabilidad de detectar una diferencia cuando ésta en realidad no existe.

1.4.2. Imputación Cold Deck

Con este método se intenta mitigar la reducción de variabilidad que se produce por el uso de valores de la misma muestra. Hay situaciones donde puede ser posible el uso de otros conjuntos de datos como proveedores de valores que sirvan para imputar datos faltantes en el conjunto de datos a analizar. La situación más probable donde la imputación cold deck podría ser posible es en encuestas. También hay estudios donde los investigadores separan conjuntos de datos para realizar pruebas de modelos tanto exploratorias como confirmatorias. Esto es, el conjunto de datos completo se separa en dos subconjuntos: uno para el análisis exploratorio y el otro para confirmar los resultados de dicho análisis. En esta circunstancia, el conjunto de datos parcial que

sirvió para el análisis confirmatorio puede servir como el "cold deck" para imputar datos faltantes en el subconjunto exploratorio y viceversa. Con este método también se subestiman los errores estándares, aunque en menor medida que con hot deck imputation.

1.4.3. Last observation carried forward (LOCF)

Este método de imputación generalmente se aplica en estudios longitudinales. Como su nombre lo indica, este procedimiento imputa datos faltantes repetidos con el valor que inmediatamente les precede. La suposición subyacente es que la observación más reciente es el mejor estimador para valores faltantes subsecuentes. A modo de ilustración, en la Tabla 1.4 se muestran cuatro ciclos de datos longitudinales de una muestra pequeña de casos. Nótese que la última observación registrada de cada caso se usa para imputar los valores faltantes en los ciclos subsecuentes. Esta estrategia se aplica a los casos con abandonos permanentes o intermitentes.

Tabla 1.4. Datos longitudinales imputados mediante LOCF.

	Datos originales				**Datos imputados**			
Caso	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4	Ciclo 1	Ciclo 2	Ciclo 3	Ciclo 4
1	50	53	–	–	50	53	53	53
2	47	46	49	51	47	46	49	51
3	43	–	–	–	43	43	43	43
4	55	–	56	59	55	55	56	59
5	45	45	47	46	45	45	47	46

La opinión convencional es que LOCF produce estimadores conservadores de las diferencias entre grupos, ya que incorpora valores que no cambian en el tiempo. Sin embargo, estudios empíricos han mostrado que esto no es necesariamente cierto. De hecho, este esquema de imputación realmente puede exagerar diferencias grupales. Es probable que LOCF produzca estimadores sesgados bajo cualquier mecanismo de pérdida de datos. La magnitud del sesgo es difícil de predecir y depende de características específicas de los datos. LOCF necesita ser complementado con un método de análisis estadístico propio que distinga entre datos reales y datos imputados, pero típicamente esto no se hace (Molenberghs and Kenward, 2007).

A pesar de su uso frecuente en ciencias médicas y de la salud, como el hecho de que la Administración de Alimentos y Medicamentos de Estados Unidos considere a LOCF como el método preferido de análisis, un creciente número de estudios empíricos sugieren que este enfoque es una estrategia pobre para tratar con datos longitudinales faltantes. De hecho, el Panel de

Manejo de Datos Faltantes en Ensayos Clínicos de Estados Unidos recomienda que LOCF no debería ser usado como el primer método para manejar datos faltantes, a menos que los supuestos subyacentes estén científicamente justificados (Van Buuren, 2012). Existe un método similar a LOCF conocido como **next observation carried backward (NOCB)**, el cual imputa valores faltantes repetidos con la observación que inmediatamente les sucede. Resultados de simulaciones tienden a apoyar el uso de NOCB sobre LOCF, aunque ambos comparten defectos similares.

La Tabla 1.5 muestra un resumen de algunos de los métodos antes descritos. En ella se identifican los mecanismos de pérdida de datos que cada método debe cumplir para generar estimadores insesgados de la media, de los coeficientes de regresión y de correlación. También se identifican las propiedades de los errores estándares por método. Como se puede ver, ambos métodos de eliminación siempre requieren pérdida MCAR, mientras que las imputaciones por regresión y por regresión estocástica son las únicas que pueden proporcionar estimadores insesgados bajo MAR, siempre y cuando el modelo está correctamente especificado. Por otro lado, LOCF es incapaz de proveer estimadores consistentes, aún bajo MCAR.

Tabla 1.5. Resumen de los mecanismos de pérdida supuestos para obtener estimadores insesgados según el método empleado para los datos faltantes.

	Estimadores insesgados			
Método	Media	Regresión	Correlación	**Error estándar**
Casos completos	MCAR	MCAR	MCAR	sobrestimado
Casos disponibles	MCAR	MCAR	MCAR	ambiguo
Imp. de la media	MCAR	–	–	subestimado
Imp. por regresión	MAR	MAR	–	subestimado
Imp. estocástica	MAR	MAR	MAR	subestimado
LOCF	–	–	–	subestimado

Todos los métodos anteriores comparten el defecto de proporcionar errores estándares inconsistentes. Por ejemplo, el problema central con los métodos de imputación simple es que las inferencias basadas en datos imputados no toman en cuenta la incertidumbre que añade la imputación. Esto se debe a que las técnicas de análisis estándar no pueden distinguir entre los valores imputados y los valores reales. En consecuencia, los errores estándares son sistemáticamente subestimados. Existen varias estrategias para corregir este problema en ciertas situaciones, tales como las técnicas de remuestreo (e.g., bootstrap, jackknife, etc.). Pero una solución más conveniente es utilizar imputación múltiple (Van Buuren, 2012).

1.5. Métodos de vanguardia

1.5.1. Imputación Múltiple

Propuesta inicialmente por Rubin (1987) dentro del ámbito de las encuestas, imputación múltiple (MI, por sus siglas en inglés) consta de tres etapas: (1) la etapa de imputación crea múltiples copias del conjunto de datos con valores faltantes, e imputa cada una con diferentes valores plausibles, (2) la etapa de análisis estima de manera individual los parámetros de interés de cada conjunto de datos imputado, y (3) la etapa de combinación obtiene un estimador global de cada parámetro agrupando adecuadamente los estimadores individuales. De esta manera, y a diferencia de la imputación simple, la imputación múltiple añade otra fuente de incertidumbre que es capturada por los errores estándares, lo cual evita la subestimación sistemática de los mismos. Las tres etapas de imputación múltiple se muestra de manera gráfica en la Figura 1.3.

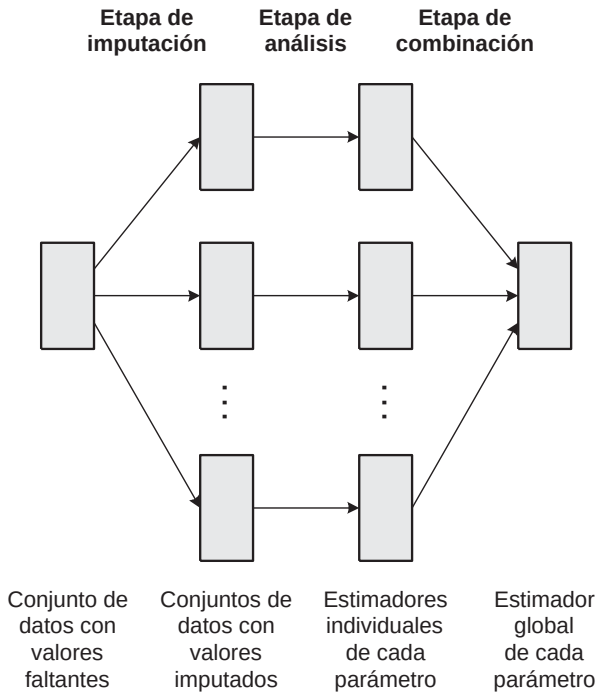

Figura 1.3. Esquema gráfico de las tres etapas de imputación múltiple.

Conceptualmente, la etapa de imputación es una versión iterativa de la imputación por regresión estocástica, aunque su base matemática descansa fuertemente en principios de estimación Bayesiana, como el método Monte Carlo vía cadenas de Markov (MCMC, por sus siglas en inglés). Para esta etapa los metodólogos han propuesto varias estrategias, entre las que se destacan dos.

La primera es *Joint modeling* (JM) o también llamada *data augmentation*. Se basa en el supuesto de que los datos pueden ser descritos mediante alguna distribución multivariada explícita. Así, las imputaciones se obtienen a partir de la distribución ajustada. En principio, puede usarse cualquier distribución multivariada, pero el modelo normal multivariado es por mucho el más ampliamente utilizado (Schafer, 1997). Actualmente, JM es más aplicado al manejo de variables continuas con datos faltantes. Este método está disponible en paquetes de software estadístico como `PROC MI` en `SAS`, `S+MissingData` en `S-PLUS` y `norm` en `R`, por mencionar sólo algunos.

La segunda estrategia es *fully conditional specification* (FCS), también conocida como *sequential regressions* o *chained equations*. A diferencia de JM, el modelo multivariado está implícitamente especificado por medio de un conjunto de modelos condicionales univariados. Las imputaciones son creadas a partir de la iteración de los modelos condicionales. El método requiere la especificación de un modelo de imputación para cada variable incompleta, y crea imputaciones variable por variable de una manera iterativa, adecuando el modelo de imputación con el tipo de distribución de la variable en cuestión. Por ejemplo, el algoritmo puede usar una regresión lineal para imputar variables continuas, una regresión logística para imputar variables binarias, una regresión Poisson para imputar variables contables, y así sucesivamente. De esta manera, FCS se ha convertido en una buena alternativa a JM para tratar variables tanto continuas como categóricas con valores faltantes. Esta técnica ha sido implementada en paquetes tales como `Missing Values` en `SPSS`, `IVEware` en `SAS`, `ice` en `Stata` y `mice` en `R`, entre otros.

FCS es similar a JM en algunos casos especiales. Por ejemplo, la imputación mediante FCS usando regresiones lineales para puras variables continuas, es idéntica a la imputación bajo el modelo normal multivariado de JM. Por otro lado, existen diferencias entre ambas metodologías. FCS no puede usar atajos computacionales como el *sweep operator*, por lo que los cálculos son más intensivos que bajo JM, lo cual lo hace mucho más lento computacionalmente. Además, JM tiene mejores bases teóricas que garantizan su convergencia. Sin embargo, FCS permite una tremenda flexibilidad de crear modelos multivariados, especificando fácilmente modelos más allá de los típicos conocidos. FCS puede usar métodos de imputación especializados que son difíciles de formular en JM como parte de una distribución multivariada. Los métodos de imputación de FCS preservan las características únicas en los datos (Van Buuren, 2012).

Como su nombre lo dice, el objetivo de la etapa de análisis es analizar los conjuntos de datos imputados obtenidos de la etapa de imputación previa. En esta etapa se aplican los mismos

procedimientos estadísticos que usualmente se usan con datos completos. Desde un punto de vista procedimental, la única diferencia es que cada análisis se realiza varias veces, una por cada conjunto de datos imputado. De esta manera, la etapa proporciona varios estimadores y errores estándares por cada parámetro de interés. La etapa de análisis es tal vez la parte menos compleja del proceso de imputación múltiple, por lo que no requiere mayor explicación.

Para la etapa final de combinación de resultados se cuenta con las llamadas *reglas de Rubin* (Rubin, 1987). Supóngase que el conjunto de datos original con valores faltantes fue imputado M veces, generándose M conjuntos de datos imputados con sus respectivos M estimadores diferentes. Sea $\hat{\theta}_k$ el estimador del parámetro de interés θ obtenido del análisis del conjunto de datos imputado k (con $k = 1, \ldots, M$), y sea V_k la varianza estimada de $\hat{\theta}_k$. Entonces, el estimador global de θ es la media de los estimadores individuales: $\bar{\theta} = M^{-1} \sum_{k=1}^{M} \hat{\theta}_k$, mientras que la varianza global de $\bar{\theta}$ está dada por $\text{Var}(\bar{\theta}) = \bar{W} + (1 + M^{-1})B$, donde $\bar{W}$ es la media de las varianzas individuales: $\bar{W} = M^{-1} \sum_{k=1}^{M} V_k$, y B es la varianza entre imputaciones: $B = (M-1)^{-1} \sum_{k=1}^{M} (\hat{\theta}_k - \bar{\theta})(\hat{\theta}_k - \bar{\theta})^T$. Dividir la varianza en dos componentes —dentro y entre imputaciones— es análogo a lo que se hace en análisis de varianza (ANOVA). ANOVA divide la variación en dos fuentes: la atribuida a una variable explicativa (i.e., variabilidad entre grupos) y la variación residual que resta después de tomar en cuenta a la variable explicativa (i.e., variabilidad dentro de grupos) (Enders, 2010).

Imputación múltiple se ha convertido rápidamente en una técnica popular para manipular datos faltantes, especialmente por su flexibilidad y relativa facilidad de implementación y uso en paquetes como `SPSS`, `SAS`, `Stata`, `S-PLUS` y `R`, entre otros. Sin embargo, tiene varias desventajas: (i) debido a la aleatoriedad involucrada en el proceso de imputación, cada vez que se aplica al mismo conjunto de datos produce diferentes resultados que pueden llevar a diferentes conclusiones, (ii) requiere muchas decisiones no triviales por parte del usuario como ¿cuántos conjuntos de datos imputados generar?, ¿cuántas iteraciones realizar dentro y entre cada conjunto de datos?, ¿cuáles distribuciones a priori utilizar?, ¿cómo preservar las interacciones durante la imputación?, etc., y aunque la mayoría de éstas son tomadas por defecto en los paquetes de software estadístico, el usuario debe revisar con cuidado si las elecciones por defecto son las apropiadas para su propósito, (iii) siempre hay una incompatibilidad potencial entre el modelo de imputación y el modelo de análisis, lo cual puede causar sesgo en los resultados; por ejemplo, mientras que el modelo de imputación puede ser estrictamente lineal, el modelo de análisis puede contener interacciones o términos no lineales, (iv) no existen criterios estadísticos para diagnósticos, pruebas gráficas,

selección de variables, etc., y a pesar de que se han propuesto algunos criterios empíricos en este sentido, desde el punto de vista frecuentista aún no hay un consenso sobre ellos; por ejemplo, está el criterio de que las variables que aparezcan en al menos el 50 % de los múltiples modelos parsimoniosos, serán las que se queden en el modelo final (Wood et al., 2008), y (v) es difícil introducir modelos para el mecanismo de pérdida MNAR.

1.5.2. Máxima Verosimilitud

Establecida de manera formal por Fisher (1922) (citado en McCullagh and Nelder, 1989, p. 11), la idea básica detrás del método de estimación por máxima verosimilitud (ML, por sus siglas en inglés) es la siguiente. Sea w un conjunto de datos con distribución paramétrica $f(\mathrm{w}|\theta)$, donde θ es el parámetro que la caracteriza. La función de verosimilitud $L(\theta|\mathrm{w})$ es cualquier función de θ proporcional a $f(\mathrm{w}|\theta)$, mientras que la función log-verosimilitud $l(\theta|\mathrm{w})$ es el logaritmo natural de $L(\theta|\mathrm{w})$. Entonces, el estimador de máxima verosimilitud $\hat{\theta}$ es el valor de θ que maximiza a $L(\theta|\mathrm{w})$, o equivalentemente, a $l(\theta|\mathrm{w})$; es decir, $l(\hat{\theta}|\mathrm{w}) \geq l(\theta|\mathrm{w})\ \forall\theta$. Esta idea está esquematizada en la Figura 1.4, donde se muestra el perfil de una función de verosimilitud para el caso especial de un sólo parámetro θ.

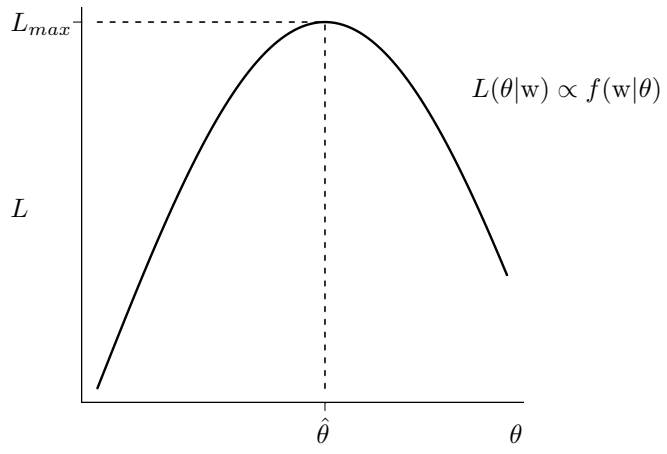

Figura 1.4. Esquema de máxima verosimilitud.

Siguiendo con la misma notación usada en la sección 1.1, y sabiendo que $L(\boldsymbol{\theta}|\mathbf{w},\mathbf{r})$ es proporcional a $f(\mathbf{w},\mathbf{r}|\boldsymbol{\theta})$, lo que resta es especificar de manera adecuada a la distribución conjunta f. Bajo el mecanismo de pérdida de datos MNAR, dicha especificación se puede establecer como

el siguiente producto:

$$f(\mathbf{z}, y, \mathbf{r}|\mathbf{x}; \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\nu}) = f_{\mathbf{z}}(\mathbf{z}|\mathbf{x}; \boldsymbol{\alpha})\ f_y(y|\mathbf{x}, \mathbf{z}; \boldsymbol{\beta})\ f_{\mathbf{r}}(\mathbf{r}|\mathbf{x}, \mathbf{z}, y; \boldsymbol{\nu}), \tag{1.1}$$

donde $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ y $\boldsymbol{\nu}$ son vectores de parámetros que caracterizan a $f_{\mathbf{z}}$, f_y y $f_{\mathbf{r}}$, respectivamente. Dado que $\mathbf{x}$ está completamente observada se puede considerar como no aleatoria sujeta a modelado. Por otro lado, bajo el mecanismo de pérdida de datos MAR, $f_{\mathbf{r}}$ puede ser ignorada y la Ecuación (1.1) se reduce a

$$f(\mathbf{z}, y|\mathbf{x}; \boldsymbol{\alpha}, \boldsymbol{\beta}) = f_{\mathbf{z}}(\mathbf{z}|\mathbf{x}; \boldsymbol{\alpha})\ f_y(y|\mathbf{x}, \mathbf{z}; \boldsymbol{\beta}). \tag{1.2}$$

La cuestión clave que motivó este trabajo de investigación es que de las Ecuaciones (1.1) y (1.2) surge la necesidad de modelar apropiadamente la distribución conjunta $f_{\mathbf{z}}$, y en su caso $f_{\mathbf{r}}$. Una propuesta ya dada en este sentido es la siguiente. Para reducir el número de parámetros indeseables $\boldsymbol{\alpha}$ a estimar, Lipsitz and Ibrahim (1996) propusieron expresar la conjunta $f_{\mathbf{z}}$ como un producto de distribuciones condicionales univariadas:

$$\begin{aligned} f_{\mathbf{z}}(\mathbf{z}|\mathbf{x}; \boldsymbol{\alpha}) &= f(z_m|z_1, \ldots, z_{m-1}, \mathbf{x}; \boldsymbol{\alpha}_m)\ f(z_{m-1}|z_1, \ldots, z_{m-2}, \mathbf{x}; \boldsymbol{\alpha}_{m-1}) \cdots \\ &\quad f(z_2|z_1, \mathbf{x}; \boldsymbol{\alpha}_2)\ f(z_1|\mathbf{x}; \boldsymbol{\alpha}_1), \end{aligned} \tag{1.3}$$

donde $\boldsymbol{\alpha}_k,\ k = 1, \ldots, m$, es un vector de parámetros que caracteriza a la k-ésima distribución condicional. El modelo dado por la Ecuación (1.3) está completamente especificado cuando cada distribución condicional correspondiente a z_k está especificada de acuerdo a las características de z_k. Sin embargo, a falta de distribuciones condicionales propias, Lipsitz and Ibrahim (1996) proponen aproximar cada condicional univariada en la Ecuación (1.3) mediante regresiones. Por ejemplo, si z_k es dicotómica, entonces su distribución condicional es aproximada con una regresión logística dejando a las variables explicativas restantes como los regresores. De manera similar, si z_k es continua, entonces su distribución condicional es aproximada con una regresión lineal dejando también a las variables explicativas restantes como los regresores.

De manera análoga al modelo expresado por la Ecuación (1.3), Ibrahim et al. (1999b) propusieron especificar la distribución conjunta $f_{\mathbf{r}}$ de la siguiente manera:

$$\begin{aligned} f_{\mathbf{r}}(\mathbf{r}|\mathbf{w}; \boldsymbol{\nu}) &= f(r_m|r_1, \ldots, r_{m-1}, \mathbf{w}; \boldsymbol{\nu}_m)\ f(r_{m-1}|r_1, \ldots, r_{m-2}, \mathbf{w}; \boldsymbol{\nu}_{m-1}) \cdots \\ &\quad f(r_2|r_1, \mathbf{w}; \boldsymbol{\nu}_2)\ f(r_1|\mathbf{w}; \boldsymbol{\nu}_1), \end{aligned} \tag{1.4}$$

donde $\boldsymbol{\nu}_k,\ k = 1, \ldots, m$, es un vector de parámetros que caracteriza a la k-ésima distribución condicional. De nuevo, a falta de condicionales univariadas propias, Ibrahim et al. (1999b)

proponen aproximar cada condicional univariada en la Ecuación (1.4) mediante regresiones logísticas para las variables binarias r_k, dejando a las restantes variables explicativas en el componente lineal de cada modelo logístico.

En cuanto al modelado de la distribución $f_y(y|\mathbf{x}, \mathbf{z}; \boldsymbol{\beta})$ también presente en las Ecuaciones (1.1) y (1.2) no hay mayor problema, ya que se asume que la variable respuesta univariada y sigue una distribución probabilística perteneciente a la familia de dispersión exponencial; y por lo tanto, f_y se puede especificar completamente mediante modelos lineales generalizados. Este tema se retomará con mayor detalle en el siguiente capítulo.

Máxima verosimilitud ha probado ser el método por excelencia para analizar datos faltantes en una amplia variedad de situaciones. Si los modelos para $f_{\mathbf{z}}$, f_y y $f_{\mathbf{r}}$ están correctamente especificados, este método es capaz de generar estimadores insesgados y eficientes bajo cualquier mecanismo de pérdida de datos. Con respecto a MI tiene varias ventajas: (i) cada vez que se aplica al mismo conjunto de datos produce los mismos resultados, (ii) su aplicación es más directa ya que requiere menos decisiones por parte del usuario, y en este sentido, se puede ver como una metodología más "transparente", (iii) no existe conflicto potencial entre imputación y análisis ya que todo está hecho bajo un sólo modelo, (iv) existen criterios estadísticos bien establecidos para diagnósticos, pruebas gráficas, selección de variables, etc., y (v) es posible modelar el mecanismo de pérdida de datos MNAR.

Por otro lado, la implementación y uso de máxima verosimilitud para datos faltantes generalmente requiere software más especializado. Muchas de las recientes innovaciones en software han ocurrido dentro del marco del Modelado de Ecuaciones Estructurales (SEM, por sus siglas en inglés), cuya metodología es conocida como *direct maximum likelihood* (DML) o *full information maximum likelihood* (FIML). Como su nombre lo dice, FIML obtiene los estimadores de los parámetros maximizando la función de verosimilitud con la incorporación apropiada de los casos con valores faltantes. La noción general de FIML fue bosquejada por Hartley and Hocking (1971). Este marco provee de un vasto número de métodos de análisis tales como correlación, regresión, ANOVA, análisis factorial, etc. Algunos paquetes de software estadístico que se han implementado bajo esta metodología son `sem` en `Stata`, `AMOS` en `SPSS`, `EQS`, `LISREL`, `PROC CALIS` en `SAS`, `lavaan` en `R`, y tal vez el más completo actualmente `Mplus`. Los paquetes anteriores tienen diferentes capacidades entre ellos, pero las características que comparten es que están diseñados sólo para modelos lineales bajo el supuesto de normalidad multivariada y mecanismo de pérdida de datos

MAR, a excepción de `Mplus` cuyas últimas versiones incluyen modelos lineales generalizados con datos MNAR sin el supuesto de normalidad multivariada.

Una segunda estrategia para manejar datos faltantes con máxima verosimilitud fuera de SEM, es utilizar el *algoritmo EM*. Este algoritmo genera estimadores del vector de medias y de la matriz de (co)varianzas, las cuales pueden ser usadas para obtener estimadores de los parámetro de interés. El algoritmo EM se ha implementado en varios paquetes de software estadístico como `Amelia II`, `SPSS`, `SAS`, `S-PLUS`, `LISREL`, y `Mplus`. Los fundamentos básicos de este algoritmo se tratarán con mayor detalle en el siguiente capítulo.

Este trabajo de investigación tiene como objetivo general proponer una metodología basada en máxima verosimilitud para el modelado adecuado de las distribuciones conjuntas $f_{\mathbf{z}}$ y $f_{\mathbf{r}}$, cuando la variable respuesta y pertenece a la familia de dispersión exponencial y está completamente observada. Para lograrlo se propone el uso de funciones *cópula* como una estrategia alternativa conveniente para modelar las distribuciones conjuntas. Además, se sugiere especificar la función de verosimilitud mediante ponderaciones y aplicar el algoritmo de estimación EM. Esto último para variables explicativas categóricas con datos faltantes tanto MAR como MNAR.

Capítulo 2

Modelos lineales generalizados y el algoritmo EM

2.1. Introducción a los modelos lineales generalizados

Los modelos lineales generalizados (GLM, por sus siglas en inglés) fueron formulados por Nelder and Wedderburn (1972) para disponer de una teoría unificadora de varios métodos estadísticos. En principio, un *modelo lineal general* es aquel que establece que el valor esperado de una variable respuesta normalmente distribuida, está en función lineal de una combinación de variables explicativas con un vector de parámetros desconocidos. Por otro lado, los GLM son en esencia una extensión natural de los modelos lineales generales, que permiten que la variable respuesta tenga otras distribuciones probabilísticas, además de la normal, y que su valor esperado sea una función no lineal de las variables explicativas y los parámetros. A través los años los GLM fueron ganado popularidad como herramienta de modelado estadístico, debido a su flexibilidad en el manejo de una gran variedad de distribuciones y a su amplia disponibilidad en paquetes de software estadístico comercial.

En un GLM se asume que la respuesta univariada $\mathbf{y}$ sigue una distribución perteneciente a la *familia de dispersión exponencial* cuya función de probabilidad puede ser escrita como

$$f_{\mathbf{y}}(\mathbf{y}; \boldsymbol{\vartheta}, \varphi) = \exp\{[\mathbf{y}\boldsymbol{\vartheta} - b(\boldsymbol{\vartheta})]/a(\varphi) + c(\mathbf{y}; \varphi)\}, \tag{2.1}$$

para algunas funciones específicas $a(\cdot)$, $b(\cdot)$ y $c(\cdot)$. Si el parámetro de dispersión o de escala $\varphi > 0$ es conocido, entonces la Ecuación (2.1) es un caso especial llamado *familia exponencial lineal* con parámetro canónico o natural $\boldsymbol{\vartheta}$ (Lindsey, 1996). Por ejemplo, si $\mathbf{y} \sim N(\boldsymbol{\mu}, \sigma^2)$, entonces $\mathbf{y}$

sigue una distribución que pertenece a la familia de dispersión exponencial ya que la función de probabilidad normal puede ser escrita de la forma

$$\begin{aligned} f_{\mathbf{y}}(\mathbf{y};\boldsymbol{\mu},\sigma^2) &= \frac{1}{\sqrt{2\pi\sigma^2}}\exp\{-(\mathbf{y}-\boldsymbol{\mu})^2/2\sigma^2\} \\ &= \exp\{[\mathbf{y}\boldsymbol{\mu}-\boldsymbol{\mu}^2/2]/\sigma^2 - \tfrac{1}{2}[\mathbf{y}^2/\sigma^2+\log(2\pi\sigma^2)]\}, \end{aligned}$$

donde $\boldsymbol{\vartheta}=\boldsymbol{\mu}$, $a(\varphi)=\sigma^2$, $b(\boldsymbol{\vartheta})=\boldsymbol{\mu}^2/2=\boldsymbol{\vartheta}^2/2$ y $c(\mathbf{y};\varphi)=-\frac{1}{2}[\mathbf{y}^2/\sigma^2+\log(2\pi\sigma^2)]$.

Un GLM también postula que $\boldsymbol{\mu}=\mathrm{E}(\mathbf{y})$ está relacionado con un vector de variables explicativas $\mathbf{x}=(1,x_1,\ldots,x_p)^T$ por medio de una *función de enlace* monótona y diferenciable $g(\cdot)$ de la siguiente manera

$$g(\boldsymbol{\mu})=\boldsymbol{\eta}=\mathbf{x}^T\boldsymbol{\beta},$$

donde $\boldsymbol{\beta}=(\beta_0,\beta_1,\ldots,\beta_p)$ es el vector de coeficientes de regresión, y $\boldsymbol{\eta}$ es el llamado *componente lineal* del modelo. Notar que $\boldsymbol{\mu}=g^{-1}(\boldsymbol{\eta})$. Por ejemplo, para la distribución Poisson se tiene que

$$f_{\mathbf{y}}(\mathbf{y};\boldsymbol{\mu})=e^{-\boldsymbol{\mu}}\boldsymbol{\mu}^{\mathbf{y}}/\mathbf{y}!=\exp(\mathbf{y}\log\boldsymbol{\mu}-\boldsymbol{\mu}-\log\mathbf{y}!),$$

donde $\boldsymbol{\vartheta}=\log\boldsymbol{\mu}$, $a(\varphi)=1$, $b(\boldsymbol{\vartheta})=\boldsymbol{\mu}=e^{\boldsymbol{\vartheta}}$ y $c(\mathbf{y};\varphi)=-\log\mathbf{y}!$. Dado que para esta distribución $\mathbf{y}$ es el número de ocurrencias de un evento, es necesario que $\boldsymbol{\mu}=\mathrm{E}(\mathbf{y})>0$. Dos relaciones que cumplen con esta restricción son $\boldsymbol{\mu}=\exp(\boldsymbol{\eta})=\exp[g(\boldsymbol{\mu})]$ y $\boldsymbol{\mu}=\boldsymbol{\eta}^2=[g(\boldsymbol{\mu})]^2$, de las cuales se obtienen dos funciones de enlace posibles para la distribución Poisson: $g(\boldsymbol{\mu})=\log\boldsymbol{\mu}$ y $g(\boldsymbol{\mu})=\sqrt{\boldsymbol{\mu}}$. Ahora bien, cuando una función de enlace para una distribución dada es igual al parámetro canónico de dicha distribución, se dice que tal función de enlace es la *canónica*. Entonces, la liga canónica para la distribución de Poisson es la logarítmica ya que $g(\boldsymbol{\mu})=\log\boldsymbol{\mu}=\boldsymbol{\vartheta}$.

Hay algunas otras distribuciones que pertenecen a la familia de dispersión exponencial como la binomial:

$$f_{\mathbf{y}}(\mathbf{y};\mathbf{p},\mathrm{n})=\binom{\mathrm{n}}{\mathbf{y}}\mathbf{p}^{\mathbf{y}}(1-\mathbf{p})^{\mathrm{n}-\mathbf{y}}=\exp\left[\mathbf{y}\log\left(\frac{\mathbf{p}}{1-\mathbf{p}}\right)+\mathrm{n}\log(1-\mathbf{p})+\log\binom{\mathrm{n}}{\mathbf{y}}\right],$$

donde $\boldsymbol{\vartheta}=\log[\mathbf{p}/(1-\mathbf{p})]$, $a(\varphi)=1$, $b(\boldsymbol{\vartheta})=-\mathrm{n}\log(1-\mathbf{p})=\mathrm{n}\log(1+e^{\boldsymbol{\vartheta}})$ y $c(\mathbf{y};\varphi)=\log\binom{\mathrm{n}}{\mathbf{y}}$. En caso de que $\mathrm{n}=1$ esta es la distribución Bernoulli. También se tiene la distribución gamma:

$$\begin{aligned} f_{\mathbf{y}}(\mathbf{y};\boldsymbol{\gamma},\lambda) &= \frac{\boldsymbol{\gamma}^{\lambda}}{\Gamma(\lambda)}\mathbf{y}^{\lambda-1}e^{-\boldsymbol{\gamma}\mathbf{y}} \\ &= \exp\left[\frac{\mathbf{y}(-\boldsymbol{\gamma}/\lambda)+\log(\boldsymbol{\gamma}/\lambda)}{1/\lambda}+\lambda\log\lambda+\lambda\log\mathbf{y}-\log\mathbf{y}-\log\Gamma(\lambda)\right], \end{aligned}$$

donde $\boldsymbol{\vartheta}=-\boldsymbol{\gamma}/\lambda$, $a(\varphi)=1/\lambda$, $b(\boldsymbol{\vartheta})=-\log(\boldsymbol{\gamma}/\lambda)=-\log(-\boldsymbol{\vartheta})$ y $c(\mathbf{y};\varphi)=\lambda\log(\lambda\mathbf{y})-\log\mathbf{y}-\log\Gamma(\lambda)$. Nótese que si $\lambda=1$ esta es la distribución exponencial.

La función log-verosimilitud escrita en forma canónica queda expresada como

$$l(\boldsymbol{\vartheta}, \varphi; \mathbf{y}) = \log[f_{\mathbf{y}}(\mathbf{y}; \boldsymbol{\vartheta}, \varphi)] = [\mathbf{y}\boldsymbol{\vartheta} - b(\boldsymbol{\vartheta})]/a(\varphi) + c(\mathbf{y}; \varphi),$$

de donde se obtiene

$$\frac{\partial l}{\partial \boldsymbol{\vartheta}} = \frac{\mathbf{y} - b'(\boldsymbol{\vartheta})}{a(\varphi)} \quad \text{y} \quad \frac{\partial^2 l}{\partial \boldsymbol{\vartheta}^2} = -\frac{b''(\boldsymbol{\vartheta})}{a(\varphi)}.$$

A partir de los siguientes resultados conocidos

$$\mathrm{E}\left(\frac{\partial l}{\partial \boldsymbol{\vartheta}}\right) = 0 \quad \text{y} \quad \mathrm{E}\left(\frac{\partial l}{\partial \boldsymbol{\vartheta}}\right)^2 = -\mathrm{E}\left(\frac{\partial^2 l}{\partial \boldsymbol{\vartheta}^2}\right),$$

se tiene que

$$\frac{\mathrm{E}(\mathbf{y}) - b'(\boldsymbol{\vartheta})}{a(\varphi)} = 0 \quad \text{y} \quad \frac{\mathrm{E}[\mathbf{y} - b'(\boldsymbol{\vartheta})]^2}{a^2(\varphi)} = \frac{b''(\boldsymbol{\vartheta})}{a(\varphi)},$$

o bien

$$\boldsymbol{\mu} = \mathrm{E}(\mathbf{y}) = b'(\boldsymbol{\vartheta}) \quad \text{y} \quad \mathrm{Var}(\mathbf{y}) = \mathrm{E}(\mathbf{y} - \boldsymbol{\mu})^2 = a(\varphi)b''(\boldsymbol{\vartheta}).$$

Dado que $b''(\boldsymbol{\vartheta})$ depende de $\boldsymbol{\mu}$ vía $b'(\boldsymbol{\vartheta})$ y $a(\varphi)$ es independiente de $\boldsymbol{\mu}$, es común expresar $\mathrm{Var}(\mathbf{y})$ como $\mathrm{Var}(\mathbf{y}) = a(\varphi)V(\boldsymbol{\mu})$, donde $V(\boldsymbol{\mu})$ es la llamada *función de varianza*. Si $g(\boldsymbol{\mu})$ es la función de enlace canónica, entonces se cumple que $V(\boldsymbol{\mu}) = [g'(\boldsymbol{\mu})]^{-1}$.

En la Tabla 2.1 se muestran algunos de los resultados obtenidos para cuatro distribuciones que pertenecen a la familia de dispersión exponencial.

Tabla 2.1. Características de cuatro distribuciones de la familia de dispersión exponencial.

	Normal $N(\mu, \sigma^2)$	Poisson $P(\mu)$	Binomial $B(\mathrm{p}, \mathrm{n})$	Gamma $G(\gamma, \lambda)$
y	$(-\infty, \infty)$	$\{0, 1, 2, \ldots\}$	$\{0, 1, \ldots, \mathrm{n}\}$	$(0, \infty)$
ϑ	μ	$\log \mu$	$\log[\mathrm{p}/(1-\mathrm{p})]$	$-\gamma/\lambda$
$a(\varphi)$	σ^2	1	1	$1/\lambda$
$b(\vartheta)$	$\vartheta^2/2$	e^{ϑ}	$\mathrm{n}\log(1+e^{\vartheta})$	$-\log(-\vartheta)$
$c(y; \varphi)$	$-\frac{1}{2}[y^2/\sigma^2 + \log(2\pi\sigma^2)]$	$-\log y!$	$\log\binom{\mathrm{n}}{y}$	$\lambda\log(\lambda y) - \log y - \log\Gamma(\lambda)$
$\mu(\vartheta) = b'(\vartheta)$	$\vartheta = \mu$	$e^{\vartheta} = \mu$	$\mathrm{n}e^{\vartheta}/(1+e^{\vartheta}) = \mathrm{np}$	$-1/\vartheta = \lambda/\gamma$
$g(\mu) = \vartheta(\mu)$	μ	$\log \mu$	$\log[\mu/(\mathrm{n}-\mu)]$	$-1/\mu$
$V(\mu) = [g'(\mu)]^{-1}$	1	μ	$\mu(1-\mu/\mathrm{n})$	μ^2
$\mathrm{Var}(y) = aV$	σ^2	μ	$\mathrm{np}(1-\mathrm{p})$	λ/γ^2

Sin pérdida de generalidad se puede asumir que la función $a(\varphi)$ es de la forma $a(\varphi) = \varphi/\mathrm{m}$, donde m es un peso conocido a priori que usualmente es igual a 1. Las inferencias estadísticas

acerca del vector de parámetros de regresión $\boldsymbol{\beta}$ es uno de los principales objetivos en la teoría de los GLM. Al aplicar el método de máxima verosimilitud para la estimación de $\boldsymbol{\beta}$ se tiene que

$$\frac{\partial l}{\partial \boldsymbol{\beta}} = \frac{\partial l}{\partial \boldsymbol{\vartheta}} \frac{\partial \boldsymbol{\vartheta}}{\partial \boldsymbol{\eta}} \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{\beta}},$$

donde

$$\frac{\partial l}{\partial \boldsymbol{\vartheta}} = \frac{1}{a(\varphi)}(\mathbf{y} - \boldsymbol{\mu}) \quad \text{y} \quad \frac{\partial \boldsymbol{\eta}}{\partial \boldsymbol{\beta}} = \mathbf{X},$$

siendo $\mathbf{X}$ la matriz de diseño cuyos renglones son los vectores $\mathbf{x}^T$. Entonces, la *función score* es de la forma

$$\frac{\partial l}{\partial \boldsymbol{\beta}} = \frac{1}{a(\varphi)}(\mathbf{y} - \boldsymbol{\mu}) \frac{\partial \boldsymbol{\vartheta}}{\partial \boldsymbol{\eta}} \mathbf{X},$$

y dado que $a(\varphi)$ es típicamente una constante, las ecuaciones de verosimilitud para $\boldsymbol{\beta}$ pueden ser expresadas en forma matricial como

$$\mathbf{X}^T \boldsymbol{\Delta}(\mathbf{y} - \boldsymbol{\mu}) = \mathbf{0}, \tag{2.2}$$

donde $\boldsymbol{\Delta} = \text{diag}(\partial \boldsymbol{\vartheta} / \partial \boldsymbol{\eta})$. Generalmente estas ecuaciones son no lineales en $\boldsymbol{\beta}$ y se requiere de un procedimiento iterativo para obtener $\hat{\boldsymbol{\beta}}$. Una expansión en series de Taylor de primer orden de las ecuaciones de verosimilitud en $\hat{\boldsymbol{\beta}}$ conduce al procedimiento Newton-Raphson (Ibrahim, 1990):

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} + [\mathbf{X}^T(\boldsymbol{\Delta}\mathbf{V}\boldsymbol{\Delta} - \dot{\boldsymbol{\Delta}}\mathbf{H})\mathbf{X}]^{-1}\mathbf{X}^T\boldsymbol{\Delta}(\mathbf{y} - \boldsymbol{\mu})\Big|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(t)}}, \tag{2.3}$$

donde $\mathbf{V} = \text{diag}(b''(\boldsymbol{\vartheta}))$, $\mathbf{H} = \text{diag}(\mathbf{y} - \boldsymbol{\mu})$ y $\dot{\boldsymbol{\Delta}} = \text{diag}(\partial^2 \boldsymbol{\vartheta} / \partial \boldsymbol{\eta}^2)$. Por lo tanto, la matriz de información observada para $\boldsymbol{\beta}$ queda expresada como

$$I(\hat{\boldsymbol{\beta}}) = \frac{1}{\varphi}\mathbf{X}^T(\boldsymbol{\Delta}\mathbf{V}\boldsymbol{\Delta} - \dot{\boldsymbol{\Delta}}\mathbf{H})\mathbf{X}. \tag{2.4}$$

Cuando se usa una función de enlace canónica resulta que $\boldsymbol{\eta} = \boldsymbol{\vartheta}$, $\boldsymbol{\Delta} = \text{diag}(1) = \mathbf{I}$ y $\dot{\boldsymbol{\Delta}} = \text{diag}(0) = \mathbf{0}$, por lo que las Ecuaciones (2.2), (2.3) y (2.4) se simplifican a

$$\mathbf{X}^T(\mathbf{y} - \boldsymbol{\mu}) = \mathbf{0},$$

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} + (\mathbf{X}^T\mathbf{V}\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{y} - \boldsymbol{\mu})\big|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(t)}},$$

$$I(\hat{\boldsymbol{\beta}}) = \frac{1}{\varphi}\mathbf{X}^T\mathbf{V}\mathbf{X}, \tag{2.5}$$

respectivamente. Así, los estimadores de los errores estándares se obtienen de la raíz cuadrada de los elementos en la diagonal principal de $I^{-1}(\hat{\boldsymbol{\beta}})$.

2.2. Fundamentos del algoritmo EM

El *algoritmo EM* (acrónimo de Expectation–Maximization) es un método iterativo de optimización que permite obtener estimadores de máxima verosimilitud en un amplio rango de aplicaciones, especialmente en aquellas con ausencia de datos donde otros métodos iterativos tal como el Newton-Raphson tienden a ser muy complicados. La formulación general del algoritmo fue establecida por Dempster et al. (1977). En cada iteración del algoritmo EM hay dos etapas llamadas *Etapa E* (Expectation) y *Etapa M* (Maximization). Los conjuntos de datos con valores faltantes o censurados, así como los modelos con distribuciones truncadas, ocurren con frecuencia en situaciones prácticas, dando como resultado modelos con funciones de verosimilitud complicadas. Sin embargo, el desarrollo del algoritmo EM y toda su metodología relacionada, junto con la disponibilidad actual de mayor poder de cómputo, han hecho que el análisis de tales modelos sea mucho más factible que en el pasado. Por ello, el algoritmo EM ya se ha convertido en una herramienta estándar dentro del repertorio estadístico.

A continuación se describe de manera breve la formulación general del algoritmo EM. Sea $\mathbf{x}$ un vector de variables completamente observadas y sea $\mathbf{z}$ un vector de variables parcialmente observadas, cuya distribución conjunta está dada por $f(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})$, donde $\boldsymbol{\theta}$ es el vector de parámetros que la caracteriza. Así, la distribución marginal de $\mathbf{x}$ está dada por (Robert and Casella, 2010):

$$f(\mathbf{x}|\boldsymbol{\theta}) = \int_Z f(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) \, \mathrm{d}\mathbf{z}.$$

Entonces, la distribución condicional de $\mathbf{Z}$ dado $\mathbf{x}$ es $f(\mathbf{z}|\mathbf{x}; \boldsymbol{\theta}) = f(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})/f(\mathbf{x}|\boldsymbol{\theta})$, de la cual resulta $f(\mathbf{x}|\boldsymbol{\theta}) = f(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})/f(\mathbf{z}|\mathbf{x}; \boldsymbol{\theta})$ o, equivalentemente, $L(\boldsymbol{\theta}|\mathbf{x}) = L(\boldsymbol{\theta}|\mathbf{x}, \mathbf{z})/f(\mathbf{z}|\mathbf{x}; \boldsymbol{\theta})$. Tomando logaritmos en esta última expresión nos conduce a la siguiente relación entre las funciones log-verosimilitud:

$$l(\boldsymbol{\theta}|\mathbf{x}) = \mathrm{E}[l(\boldsymbol{\theta})|\mathbf{x}, \mathbf{z}] - \mathrm{E}[\log f(\mathbf{z}|\mathbf{x}; \boldsymbol{\theta})],$$

donde el valor esperado es con respecto a $f(\mathbf{z}|\mathbf{x}; \boldsymbol{\theta})$. Aunque en el algoritmo EM el objetivo es maximizar $l(\boldsymbol{\theta}|\mathbf{x})$, sólo el primer término del lado derecho será considerado.

Para comenzar la primera iteración del algoritmo EM se da un valor inicial $\boldsymbol{\theta}^{(0)}$, con el cual en la etapa E del algoritmo se calcula $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(0)}) := \mathrm{E}[l(\boldsymbol{\theta})|\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}^{(0)}]$. Después, en la etapa M del algoritmo se lleva a cabo la maximización de $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(0)})$ con respecto a $\boldsymbol{\theta}$; esto es, se elige $\boldsymbol{\theta}^{(1)}$ tal que $Q(\boldsymbol{\theta}^{(1)}|\boldsymbol{\theta}^{(0)}) \geq Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(0)})$ para todo $\boldsymbol{\theta}$ sobre el espacio de parámetros. Entonces ambas etapas E y M se ejecutan de nuevo, pero esta vez reemplazando $\boldsymbol{\theta}^{(0)}$ con el valor actual $\boldsymbol{\theta}^{(1)}$. De esta

manera las etapas E y M en la $(t+1)$-ésima iteración del algoritmo quedan definidas como:

Etapa E: Se calcula $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) := \mathrm{E}[l(\boldsymbol{\theta})|\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}^{(t)}]$.

Etapa M: Se maximiza $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$; i.e., se elige $\boldsymbol{\theta}^{(t+1)}$ tal que $Q(\boldsymbol{\theta}^{(t+1)}|\boldsymbol{\theta}^{(t)}) \geq Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})\ \forall \boldsymbol{\theta}$.

Así, las etapas E y M del algoritmo se ejecutan alternadamente de manera iterativa hasta que la diferencia $\boldsymbol{\theta}^{(t+1)} - \boldsymbol{\theta}^{(t)}$, o bien $L(\boldsymbol{\theta}^{(t+1)}) - L(\boldsymbol{\theta}^{(t)})$, sea menor o igual a una cantidad arbitrariamente pequeña elegida de antemano como tolerancia para logar la convergencia. Dempster et al. (1977) demostraron que la función de verosimilitud para datos incompletos $L(\boldsymbol{\theta})$ es no decreciente después de cada iteración del algoritmo EM; es decir, $L(\boldsymbol{\theta}^{(t+1)}) \geq L(\boldsymbol{\theta}^{(t)})$ para $t = 0, 1, 2, \ldots$. Por lo tanto, la convergencia debe ser obtenida mediante una sucesión de valores de verosimilitud acotados inferiormente.

El algoritmo EM tiene varias propiedades atractivas con respecto a otros algoritmos iterativos tipo Newton para encontrar estimadores de máxima verosimilitud. Algunas de estas ventajas son las siguientes. Es numéricamente estable ya que en cada iteración se incrementa la verosimilitud. Bajo ciertas condiciones generales tiene convergencia global segura; es decir, iniciando desde un punto arbitrario $\boldsymbol{\theta}^{(0)}$ la convergencia siempre está cerca a un máximo local, salvo una elección desafortunada de $\boldsymbol{\theta}^{(0)}$ o por alguna patología local en la función de verosimilitud. Es relativamente fácil de programar ya que no requiere la evaluación de derivadas; además, como no se tienen que guardar grandes matrices, tampoco se requiere demasiado espacio de memoria, por lo que generalmente puede ser implementado en computadoras estándar. Dado que a un problema de datos faltantes lo convierte en uno de datos completos, la etapa M puede ser implementada usando herramientas de optimización convencionales. El trabajo analítico requerido es mucho más simple que con otros métodos, ya que sólo la esperanza condicional de la función log-verosimilitud necesita ser maximizada; aunque cierta cantidad de trabajo analítico puede ser requerida durante la implementación de la etapa E, en muchas de las aplicaciones esto no es demasiado complicado. El tiempo demandado por cada iteración es generalmente bajo, lo cual suele compensarse con el alto número de iteraciones requeridas. Observando el incremento monótono en verosimilitud durante las iteraciones, es fácil monitorear la convergencia y detectar errores.

Por otro lado, algunas de las desventajas del algoritmo EM con respecto a otros métodos tipo Newton son las siguientes. No cuenta con ningún procedimiento propio para obtener estimadores de los errores estándares, aunque esta desventaja puede ser superada usando la metodología apropiada

y especialmente asociada con el algoritmo para este propósito. Puede converger lentamente aún en aplicaciones aparentemente simples o con una fracción importante de datos faltantes. Al igual que los métodos tipo Newton, el algoritmo EM no garantiza convergencia a un máximo global cuando hay múltiples máximos; además, en este caso el estimador obtenido depende en gran parte del valor inicial. En algunas aplicaciones la etapa E del algoritmo puede ser analíticamente intratable, aunque en tales situaciones existe la posibilidad de recurrir a versiones modificadas del algoritmo con metodología Monte Carlo.

2.3. El algoritmo EM vía ponderaciones

El algoritmo EM basado en ponderaciones fue propuesto inicialmente por Ibrahim (1990) para estimar parámetros por máxima verosimilitud en modelos lineales generalizados con variables explicativas categóricas tipo factores parcialmente observadas y una variable respuesta completamente observada. Aunque en dicha propuesta se asume que el mecanismo de pérdida de datos en las variables explicativas es MAR, el método se puede generalizar fácilmente para resolver problemas con datos faltantes bajo el mecanismo MNAR.

Para especificar la función log-verosimilitud por medio de ponderaciones, lo primero es obtener a partir del conjunto de datos original con valores faltantes y n casos, un conjunto de datos *aumentado* a N casos y que incluya una columna de ponderaciones w. Para ello, se sigue un procedimiento análogo al mostrado en la Figura 2.1 con variables dicotómicas como caso especial. Una vez obtenido el conjunto de datos aumentado, la función log-verosimilitud esperada para el i-ésimo caso del conjunto de datos original, puede expresarse como una suma de funciones log-verosimilitud ponderadas de los (i, q)-ésimos casos del conjunto de datos aumentado; es decir,

$$\mathrm{E}[l_i(\boldsymbol{\theta})|\mathbf{x}_i, \mathbf{z}_{i,obs}, y_i, \mathbf{r}_i] = \sum_q w_{i,q} \; [l_{i,q}(\boldsymbol{\theta})|\mathbf{x}_i, \mathbf{z}_{i,q}, y_i, \mathbf{r}_i], \tag{2.6}$$

donde q es el número de todos los patrones posibles de $\mathbf{z}_i$, $0 < w_{i,q} < 1$ y $\sum_q w_{i,q} = 1$. Por ejemplo, de la Figura 2.1 se tiene que $\mathrm{E}[l_2(\boldsymbol{\theta})|\mathbf{x}_2, \mathbf{z}_{2,obs}, y_2, \mathbf{r}_2] = w_{2,1} \; [l_{2,1}(\boldsymbol{\theta})|\mathbf{x}_2, \mathbf{z}_{2,1}, y_2, \mathbf{r}_2] + w_{2,2} \; [l_{2,2}(\boldsymbol{\theta})|\mathbf{x}_2, \mathbf{z}_{2,2}, y_2, \mathbf{r}_2]$, donde $w_{2,1} + w_{2,2} = 1$. De manera análoga para $l_4(\boldsymbol{\theta})$, $l_6(\boldsymbol{\theta})$, etc.

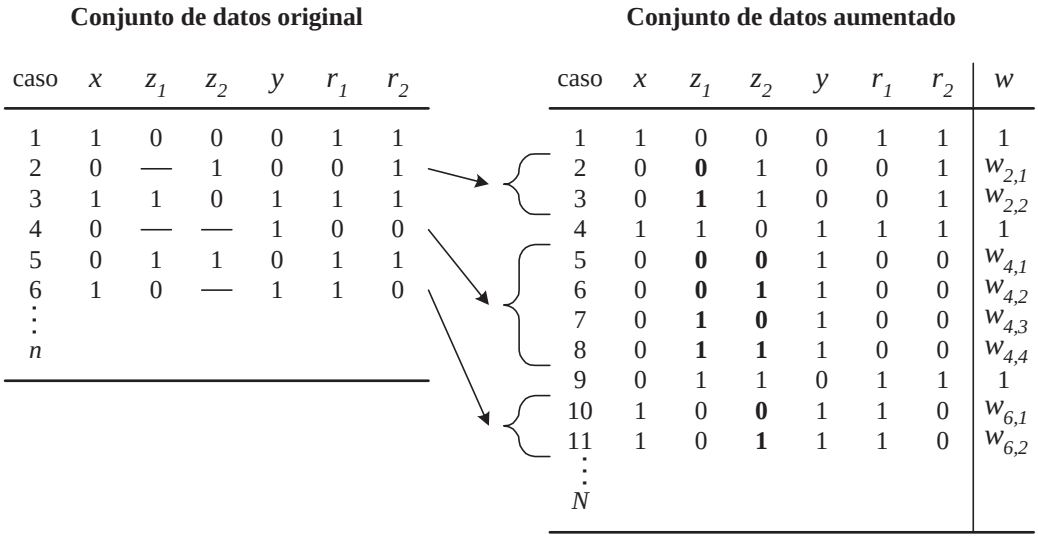

Figura 2.1. Procedimiento para aumentar un conjunto de datos vía ponderaciones.

Así, la etapa E del algoritmo EM en la $(t+1)$-ésima iteración queda como (Ibrahim, 1990):

$$\begin{aligned} Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) &= \sum_{i=1}^{n} Q_i(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \\ &:= \sum_{i=1}^{n} \mathrm{E}[l_i(\boldsymbol{\theta})|\mathbf{x}_i, \mathbf{z}_{i,obs}, y_i, \mathbf{r}_i, \boldsymbol{\theta}^{(t)}] \\ &\overset{(2.6)}{=} \sum_{i=1}^{n}\sum_{q} w_{i,q}^{(t)}\ [l_{i,q}(\boldsymbol{\theta})|\mathbf{x}_i, \mathbf{z}_{i,q}, y_i, \mathbf{r}_i, \boldsymbol{\theta}^{(t)}], \end{aligned} \tag{2.7}$$

donde las ponderaciones $w_{i,q}^{(t)}$ pueden ser calculadas mediante el teorema de Bayes utilizando el valor actual del vector de parámetros $\boldsymbol{\theta}^{(t)} = (\boldsymbol{\alpha}^{(t)}, \boldsymbol{\beta}^{(t)}, \boldsymbol{\nu}^{(t)})$:

$$\begin{aligned} w_{i,q}^{(t)} &:= f(\mathbf{z}_{i,q}|y_i, \mathbf{r}_i; \mathbf{x}_i, \boldsymbol{\theta}^{(t)}) \\ &\overset{\text{Bayes}}{=} \frac{f(y_i, \mathbf{r}_i|\mathbf{z}_{i,q}; \mathbf{x}_i, \boldsymbol{\beta}^{(t)}, \boldsymbol{\nu}^{(t)})\ f(\mathbf{z}_{i,q}|\mathbf{x}_i; \boldsymbol{\alpha}^{(t)})}{\sum_q f(y_i, \mathbf{r}_i|\mathbf{z}_{i,q}; \mathbf{x}_i, \boldsymbol{\beta}^{(t)}, \boldsymbol{\nu}^{(t)})\ f(\mathbf{z}_{i,q}|\mathbf{x}_i; \boldsymbol{\alpha}^{(t)})} \\ &\overset{\text{def}}{=} \frac{f(\mathbf{z}_{i,q}, y_i, \mathbf{r}_i|\mathbf{x}_i; \boldsymbol{\alpha}^{(t)}, \boldsymbol{\beta}^{(t)}, \boldsymbol{\nu}^{(t)})}{\sum_q f(\mathbf{z}_{i,q}, y_i, \mathbf{r}_i|\mathbf{x}_i; \boldsymbol{\alpha}^{(t)}, \boldsymbol{\beta}^{(t)}, \boldsymbol{\nu}^{(t)})} \\ &\overset{(1.1)}{=} \frac{f_{\mathbf{z}}(\mathbf{z}_{i,q}|\mathbf{x}_i; \boldsymbol{\alpha}^{(t)})\ f_y(y_i|\mathbf{x}_i, \mathbf{z}_{i,q}; \boldsymbol{\beta}^{(t)})\ f_{\mathbf{r}}(\mathbf{r}_i|\mathbf{x}_i, \mathbf{z}_{i,q}, y_i; \boldsymbol{\nu}^{(t)})}{\sum_q f_{\mathbf{z}}(\mathbf{z}_{i,q}|\mathbf{x}_i; \boldsymbol{\alpha}^{(t)})\ f_y(y_i|\mathbf{x}_i, \mathbf{z}_{i,q}; \boldsymbol{\beta}^{(t)})\ f_{\mathbf{r}}(\mathbf{r}_i|\mathbf{x}_i, \mathbf{z}_{i,q}, y_i; \boldsymbol{\nu}^{(t)})}. \end{aligned} \tag{2.8}$$

Nótese que la etapa E del algoritmo EM dada por la Ecuación (2.7) toma la forma de una función log-verosimilitud para datos completos ponderados, por lo que realmente en esta etapa sólo se requiere el cálculo de las ponderaciones w por medio de la Ecuación (2.8). Por otro lado, de la

Ecuación (1.1) se tiene que $l(\boldsymbol{\theta}) = l(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\nu}) = l(\boldsymbol{\alpha}) + l(\boldsymbol{\beta}) + l(\boldsymbol{\nu})$; así, el modelo dado en la Ecuación (2.7) también puede ser expresado como

$$\begin{aligned}
Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) &= \sum_{i=1}^{n}\sum_{q} w_{i,q}^{(t)}\ [l_{i,q}(\boldsymbol{\theta})|\boldsymbol{\theta}^{(t)}] \\
&= \sum_{i=1}^{n}\sum_{q} w_{i,q}^{(t)}\ [l_{i,q}(\boldsymbol{\alpha}) + l_{i,q}(\boldsymbol{\beta}) + l_{i,q}(\boldsymbol{\nu})|\boldsymbol{\theta}^{(t)}] \\
&= \sum_{i=1}^{n}\sum_{q} w_{i,q}^{(t)}\ [l_{i,q}(\boldsymbol{\alpha})|\boldsymbol{\theta}^{(t)}] + \sum_{i=1}^{n}\sum_{q} w_{i,q}^{(t)}\ [l_{i,q}(\boldsymbol{\beta})|\boldsymbol{\theta}^{(t)}] + \sum_{i=1}^{n}\sum_{q} w_{i,q}^{(t)}\ [l_{i,q}(\boldsymbol{\nu})|\boldsymbol{\theta}^{(t)}] \\
&= \sum_{i=1}^{n} \mathrm{E}[l_i(\boldsymbol{\alpha})|\boldsymbol{\theta}^{(t)}] + \sum_{i=1}^{n} \mathrm{E}[l_i(\boldsymbol{\beta})|\boldsymbol{\theta}^{(t)}] + \sum_{i=1}^{n} \mathrm{E}[l_i(\boldsymbol{\nu})|\boldsymbol{\theta}^{(t)}] \\
&= \sum_{i=1}^{n} Q_i(\boldsymbol{\alpha}|\boldsymbol{\theta}^{(t)}) + \sum_{i=1}^{n} Q_i(\boldsymbol{\beta}|\boldsymbol{\theta}^{(t)}) + \sum_{i=1}^{n} Q_i(\boldsymbol{\nu}|\boldsymbol{\theta}^{(t)}) \\
&= Q_{\boldsymbol{\alpha}}(\boldsymbol{\alpha}|\boldsymbol{\theta}^{(t)}) + Q_{\boldsymbol{\beta}}(\boldsymbol{\beta}|\boldsymbol{\theta}^{(t)}) + Q_{\boldsymbol{\nu}}(\boldsymbol{\nu}|\boldsymbol{\theta}^{(t)}),
\end{aligned} \tag{2.9}$$

donde las variables involucradas $(\mathbf{x}, \mathbf{z}, y, \mathbf{r})$ se han omitido para simplificar la notación.

Una vez calculadas las ponderaciones w en la etapa E del algoritmo, lo que resta hacer para completar la $(t+1)$-ésima iteración, es que en la etapa M se lleve a cabo la maximización de la función dada por la Ecuación (2.9), esto con el objetivo de actualizar el valor de los parámetros involucrados $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ y $\boldsymbol{\nu}$, usando para ello el valor de las ponderaciones recién calculadas. Para lograr esto, se pueden maximizar por separado las funciones $Q_{\boldsymbol{\alpha}}$, $Q_{\boldsymbol{\beta}}$ y $Q_{\boldsymbol{\nu}}$ aplicando métodos de optimización que permitan la inclusión de ponderaciones (Ibrahim et al., 1999b). De esta manera, mientras que en la etapa E del algoritmo se van actualizando las ponderaciones con los parámetros obtenidos, en la etapa M se van actualizando los parámetros con las ponderaciones calculadas, y así sucesivamente de manera iterativa hasta lograr la convergencia deseada.

Para la optimización de $Q_{\boldsymbol{\beta}}$ se sigue un procedimiento análogo al desarrollado en la sección 2.1, por lo que ahora las ecuaciones de verosimilitud para $\boldsymbol{\beta}$ quedan expresadas como (Ibrahim, 1990):

$$\mathbf{X}^T\mathbf{W}\boldsymbol{\Delta}(\mathbf{y} - \boldsymbol{\mu}) = \mathbf{0}, \tag{2.10}$$

donde $\mathbf{X}$ es la matriz de diseño aumentada $N \times (p+1)$ que consiste de las observaciones completas y del conjunto extra de observaciones imputadas y ponderadas, $\mathbf{W} = \mathrm{diag}(w)$ es la matriz diagonal de ponderaciones $N \times N$, siendo N el tamaño del conjunto de datos aumentado según se muestra en la Figura 2.1. Una expansión en series de Taylor conduce de nuevo al procedimiento Newton-

Raphson, por lo que las ecuaciones iterativas para $\boldsymbol{\beta}$ quedan establecidas como

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} + [\mathbf{X}^T\mathbf{W}(\boldsymbol{\Delta}\mathbf{V}\boldsymbol{\Delta} - \dot{\boldsymbol{\Delta}}\mathbf{H})\mathbf{X}]^{-1}\mathbf{X}^T\mathbf{W}\boldsymbol{\Delta}(\mathbf{y} - \boldsymbol{\mu})\Big|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(t)}}, \tag{2.11}$$

donde todas las matrices y vectores involucrados corresponden al conjunto de datos aumentado de tamaño N. Cuando se usa una función de enlace canónica, $\boldsymbol{\Delta} = \mathbf{I}$ y $\dot{\boldsymbol{\Delta}} = \mathbf{0}$, por lo que las Ecuaciones (2.10) y (2.11) se reducen a

$$\mathbf{X}^T\mathbf{W}(\mathbf{y} - \boldsymbol{\mu}) = \mathbf{0},$$

y

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} + (\mathbf{X}^T\mathbf{W}\mathbf{V}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}(\mathbf{y} - \boldsymbol{\mu})\big|_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(t)}}, \tag{2.12}$$

respectivamente. Por otro lado, para la maximización de las funciones $Q_{\boldsymbol{\alpha}}$ y $Q_{\boldsymbol{\nu}}$ es necesario modelar apropiadamente las distribuciones conjuntas $f_{\mathbf{z}}$ y $f_{\mathbf{r}}$, lo cual es para este trabajo de investigación el tema central, que se abordará con mayor detalle en el capítulo siguiente mediante una novedosa propuesta al respecto.

El algoritmo EM no cuenta con algún procedimiento propio que le permita generar estimadores de los errores estándares para los parámetros obtenidos. Para solventar esta carencia se tiene que recurrir a métodos externos especialmente diseñados para tal propósito, tales como las aproximaciones bootstrap de Efron (1979), o los métodos de Louis (1982), de Baker (1992), de Oakes (1999), entre otros (citados en McLachlan and Krishnan, 2008, pp. 130–131). Ibrahim (1990) recomienda utilizar el método de Louis (1982) ya que sólo involucra cantidades que son obtenidas directamente del propio algoritmo EM vía ponderaciones. En particular, y dado que el interés primordial es obtener los errores estándares sólo para los parámetros de regresión $\boldsymbol{\beta}$, ya que $\boldsymbol{\alpha}$ y $\boldsymbol{\nu}$ son considerados parámetros indeseables pero necesarios para obtener $\boldsymbol{\beta}$, la expresión para la matriz de información observada implica cantidades que son calculadas durante la estimación de $\boldsymbol{\beta}$ según se muestra a continuación (Ibrahim, 1990):

$$I(\hat{\boldsymbol{\beta}}) = \frac{1}{\varphi}\mathbf{X}^T\mathbf{W}(\boldsymbol{\Delta}\mathbf{V}\boldsymbol{\Delta} - \dot{\boldsymbol{\Delta}}\mathbf{H})\mathbf{X} - \frac{1}{\varphi^2}\mathbf{X}^T\mathbf{W}\boldsymbol{\Delta}^2\mathbf{H}^2(\mathbf{I} - \mathbf{W})\mathbf{X}, \tag{2.13}$$

donde $\mathbf{I}$ es la matriz identidad de tamaño N. Así, la matriz asintótica de covarianzas es $I^{-1}(\hat{\boldsymbol{\beta}})$. Cuando se usa una función de enlace canónica la matriz de información observada dada en la Ecuación (2.13) se simplifica a

$$I(\hat{\boldsymbol{\beta}}) = \frac{1}{\varphi}\mathbf{X}^T\mathbf{W}\mathbf{V}\mathbf{X} - \frac{1}{\varphi^2}\mathbf{X}^T\mathbf{W}\mathbf{H}^2(\mathbf{I} - \mathbf{W})\mathbf{X}. \tag{2.14}$$

Si el parámetro de dispersión φ es desconocido, entonces éste puede ser estimado por máxima verosimilitud y su estimador se sustituye en $I(\hat{\boldsymbol{\beta}})$.

2.4. Diagnósticos

Similar a un escenario sin datos faltantes, las suposiciones hechas acerca de un GLM en particular pueden ser evaluadas mediante el cálculo y análisis de los *residuales de cuantiles aleatorizados* propuestos por Dunn and Smyth (1996). Bajo el GLM asumido tales residuales son independientes y normalmente distribuidos, lo cual permite la revisión de gráficas estándar para diagnosticar la calidad del ajuste. Si la respuesta Y es discreta, los residuales de cuantiles aleatorizados r_i de y_i están definidos por

$$r_i = \Phi^{-1}(u_i),$$

donde $\Phi(\cdot)$ es la función de distribución acumulada normal estándar, y u_i es una variable aleatoria uniforme en el intervalo $(a_i, b_i]$ con $a_i = F(y_i - 1)$ y $b_i = F(y_i)$. Por ejemplo, si Y es dicotómica con valores 0 y 1, entonces su función de probabilidad está dada por

$$f(y_i) = y_i \sum_q w_{i,q} \mathrm{p}_{i,q} + (1 - y_i) \sum_q w_{i,q}(1 - \mathrm{p}_{i,q}),$$

o bien

$$f(y_i) = \begin{cases} \sum_q w_{i,q}(1 - \mathrm{p}_{i,q}) & \text{si} \quad y_i = 0 \\ \sum_q w_{i,q} \mathrm{p}_{i,q} & \text{si} \quad y_i = 1 \end{cases}$$

donde $\mathrm{p}_{i,q} = \frac{\exp(X_{i,q}\hat{\boldsymbol{\beta}})}{1+\exp(X_{i,q}\hat{\boldsymbol{\beta}})}$. Mientras que su función de distribución es de la forma

$$F(y_i) = \begin{cases} 0 & \text{si} \quad y_i < 0 \\ \sum_q w_{i,q}(1 - \mathrm{p}_{i,q}) & \text{si} \quad 0 \leq y_i < 1 \\ \sum_q w_{i,q} = 1 & \text{si} \quad y_i \geq 1 \end{cases}$$

Por lo tanto, si $y_i = 0$ entonces $a_i = 0$ y $b_i = \sum_q w_{i,q}(1 - \mathrm{p}_{i,q})$, y si $y_i = 1$ entonces $a_i = \sum_q w_{i,q}(1 - \mathrm{p}_{i,q})$ y $b_i = 1$.

Si Y es continua, entonces $F(y_i)$ está uniformemente distribuida en un intervalo unitario, y los residuales de cuantiles aleatorizados están definidos por

$$r_i = \Phi^{-1}(\hat{F}(y_i)).$$

Por ejemplo, si Y es exponencial, entonces

$$\hat{F}(y_i) = \begin{cases} 0 & \text{si} \quad y_i \leq 0 \\ \sum_q w_{i,q}[1 - \exp(-y_i/\mu_{i,q})] & \text{si} \quad y_i > 0 \end{cases}$$

Capítulo 3

Modelado con funciones cópula

Como se mencionó anteriormente, este trabajo de investigación tiene como objetivo primordial proponer una metodología basada en verosimilitud para el modelado adecuado de las distribuciones conjuntas $f_{\mathbf{z}}$ y $f_{\mathbf{r}}$, cuando la variable respuesta y pertenece a la familia de dispersión exponencial. Ante esta necesidad y dado que en la literatura estadística hay muy pocas propuestas de modelos para distribuciones conjuntas, y las pocas que hay son para un limitado tipo de marginales, es oportuno sugerir el uso de funciones cópula como una alternativa, ya que por varios años las cópulas han proporcionado una estructura general unificadora y han sido una poderosa herramienta en el modelado multivariado. Una de las características más atractivas de las funciones cópula en el modelado conjunto es permitir que las marginales correspondientes tengan diferentes distribuciones probabilísticas, tanto discretas (e.g., binomial, multinomial, poisson, etc.) como continuas (e.g., normal, gamma, beta, etc.).

3.1. Conceptos preliminares

Una *cópula* es una función que relaciona o "copula" una distribución conjunta multivariada con sus respectivas distribuciones marginales univariadas. El teorema de Sklar aclara este hecho estableciendo que dado un vector aleatorio $\mathbf{Z} = (Z_1, \ldots, Z_m)$ con distribución conjunta F y distribuciones marginales $F_1, \ldots, F_m$, respectivamente, entonces existe una cópula C

$$C : (0,1)^m \to (0,1) \mid F(z_1, \ldots, z_m) = C[F_1(z_1), \ldots, F_m(z_m)] \quad \forall\ z_1, \ldots, z_m \in \mathbb{R}. \tag{3.1}$$

De manera inversa, dada una cópula C y funciones de distribución univariadas $F_1, \ldots, F_m$, entonces $F(z_1, \ldots, z_m) = C[F_1(z_1), \ldots, F_m(z_m)]$ define una función de distribución conjunta con marginales

$F_1, \ldots, F_m$, las cuales siendo continuas determinan unicidad para F, mientras que si son discretas, determinan unicidad para F en el rango $F_1 \times \cdots \times F_m$.

Una cópula *bivariada* es una función $C : (0,1)^2 \to (0,1) \mid F(z_1, z_2) = C[F_1(z_1), F_2(z_2)]$ que satisface las siguientes propiedades: (i) para todo $v_1 = F_1(z_1)$ y $v_2 = F_2(z_2)$, ambos en $(0,1)$, se tiene que $\lim_{v_j \to 1} C(v_1, v_2) = v_{3-j}$ y $\lim_{v_j \to 0} C(v_1, v_2) = 0$, con $j = 1, 2$, (ii) para cada v_1, v_2, u_1 y u_2, todos en $(0,1)$ tales que $v_1 \leq v_2$ y $u_1 \leq u_2$, se tiene que $C(v_2, u_2) - C(v_2, u_1) - C(v_1, u_2) + C(v_1, u_1) \geq 0$. Debido a esta propiedad algunos autores se refieren a la cópula C como *bi-creciente* o *cuasi-monótona*; además, una cópula también es creciente en cada uno de sus argumentos, y (iii) para todo (v_1, v_2) en $(0,1)^2$ se cumple que $W(v_1, v_2) := \text{máx}\{v_1 + v_2 - 1, 0\} \leq C(v_1, v_2) \leq \text{mín}\{v_1, v_2\} := M(v_1, v_2)$.

Como una consecuencia de la tercera propiedad y del teorema de Sklar se satisface la desigualdad $W[F_1(z_1), F_2(z_2)] \leq F(z_1, z_2) \leq M[F_1(z_1), F_2(z_2)]$, conocida como *desigualdad de las cotas de Fréchet-Hoeffding* y las funciones W y M como las cotas inferior y superior, respectivamente. También se puede establecer la continuidad uniforme de las cópulas en su dominio vía una condición de Lipschitz en $(0,1)^2$; es decir, para todo (v_1, v_2) y (u_1, u_2) ambos en $(0,1)^2$, se establece que $|C(v_2, u_2) - C(v_1, u_1)| \leq |v_2 - v_1| + |u_2 - u_1|$. Esto implica que la representación gráfica de una cópula es una superficie continua $s = C(u, v)$ dentro del cubo unitario $(0,1)^3$ y situada entre las gráficas de las cotas de Frèchet $s = W(u, v)$ y $s = M(u, v)$ como se muestra en la Figura 3.1, en la cual se incluyó como caso especial la gráfica de una cópula importante llamada cópula *producto* o *independiente* $s = \Pi(u, v) = uv$ (Nelsen, 2006).

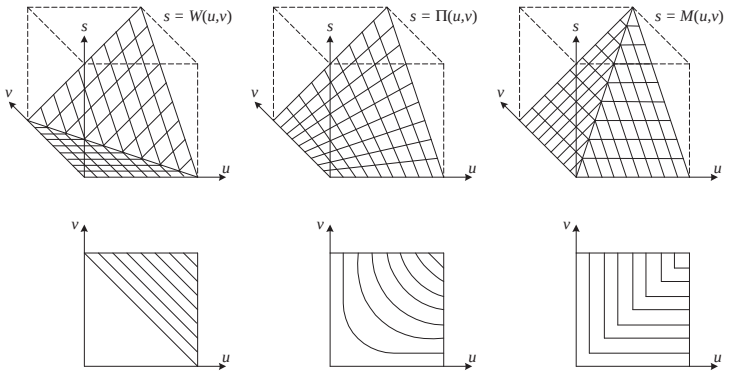

Figura 3.1. Gráficas y diagramas de contorno de las cópulas W, Π y M.

Con una extensión apropiada de su dominio a $\mathbb{R}^2$, toda cópula bivariada es una función de distribución conjunta con marginales uniformes en $(0,1)$; es decir, si C es una cópula en $(0,1)^2$,

entonces se puede extender a $\mathbb{R}^2$ por medio de F_C definida como

$$F_C(x,y) = \begin{cases} 0 & : \quad x \leq 0 \quad \text{o} \quad y \leq 0 \\ C(x,y) & : \quad (x,y) \in (0,1)^2 \\ x & : \quad y \geq 1, \ x \in (0,1) \\ y & : \quad x \geq 1, \ y \in (0,1) \\ 1 & : \quad x \geq 1 \quad \text{y} \quad y \geq 1 \end{cases}$$

Así, F_C es una conjunta con marginales uniformes $\mathrm{U}(0,1)$. De hecho, con bastante frecuencia es útil pensar en las cópulas como la restricción a $(0,1)^2$ de funciones conjuntas cuyas marginales son $\mathrm{U}(0,1)$. En la Figura 3.2 se muestra la gráfica de $z = F_C(x,y)$, donde $C(x,y) = \Pi(x,y) = xy$ como caso especial.

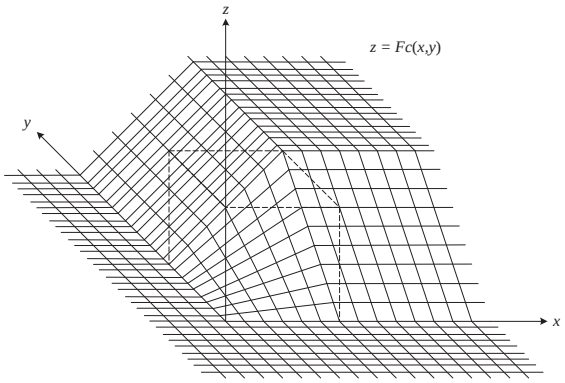

Figura 3.2. Gráfica de $z = F_C(x,y)$.

A continuación se establece un resultado importante con respecto a las derivadas parciales de las cópulas. Si las funciones de distribución marginales $F_1(z_1), \ldots, F_m(z_m)$ y la cópula $C_{1\cdots m}[F_1(z_1), \ldots, F_m(z_m)]$ son diferenciables, entonces usando la regla de la cadena se tiene que la función de densidad conjunta $f_{\mathbf{z}}$ puede ser expresada como

$$\begin{aligned} f_{\mathbf{z}}(z_1, \ldots, z_m) &:= \frac{\partial^m F_{\mathbf{z}}(z_1, \ldots, z_m)}{\partial z_1 \cdots \partial z_m} \\ &\overset{(3.1)}{=} \frac{\partial}{\partial z_1} \left\{ \cdots \left\{ \frac{\partial}{\partial z_m} C_{1\cdots m} \left[F_1(z_1), \ldots, F_m(z_m) \right] \right\} \cdots \right\} \\ &= \frac{\partial}{\partial z_1} \left\{ \cdots \left\{ \frac{\partial}{\partial F_m(z_m)} C_{1\cdots m} \left[F_1(z_1), \ldots, F_m(z_m) \right] \frac{dF_m(z_m)}{dz_m} \right\} \cdots \right\} \\ &= \frac{\partial}{\partial z_1} \left\{ \cdots \left\{ \frac{\partial}{\partial F_m(z_m)} C_{1\cdots m} \left[F_1(z_1), \ldots, F_m(z_m) \right] \right\} \cdots \right\} f_m(z_m) \end{aligned}$$

$$
\begin{aligned}
&\vdots \\
&= \frac{\partial}{\partial z_1}\left\{\frac{\partial^{m-1}C_{1\cdots m}\left[F_1(z_1),\ldots,F_m(z_m)\right]}{\partial F_2(z_2)\cdots\partial F_m(z_m)}\right\} f_2(z_2)\cdots f_m(z_m) \\
&= \frac{\partial}{\partial F_1(z_1)}\left\{\frac{\partial^{m-1}C_{1\cdots m}\left[F_1(z_1),\ldots,F_m(z_m)\right]}{\partial F_2(z_2)\cdots\partial F_m(z_m)}\right\}\frac{dF_1(z_1)}{dz_1} f_2(z_2)\cdots f_m(z_m) \\
&= \frac{\partial^m C_{1\cdots m}\left[F_1(z_1),\ldots,F_m(z_m)\right]}{\partial F_1(z_1)\cdots\partial F_m(z_m)} f_1(z_1)\cdots f_m(z_m) \\
&:= c_{1\cdots m}\left[F_1(z_1),\ldots,F_m(z_m)\right] f_1(z_1)\cdots f_m(z_m), \qquad (3.2)
\end{aligned}
$$

donde $f_1(z_1),\ldots,f_m(z_m)$ son funciones de densidad marginales y $c_{1\cdots m}$ es la llamada *función de densidad de la cópula* $C_{1\cdots m}$. Para el caso bivariado la Ecuación (3.2) se reduce a

$$f_{\mathbf{z}}(z_1,z_2) = c_{12}[F_1(z_1),F_2(z_2)]f_1(z_1)f_2(z_2). \qquad (3.3)$$

Como consecuencia, se tiene que la función de densidad condicional $f_{1|2}$ puede convenientemente expresarse de la forma

$$f_{1|2}(z_1|z_2) := \frac{f_{\mathbf{z}}(z_1,z_2)}{f_2(z_2)} = c_{12}[F_1(z_1),F_2(z_2)]f_1(z_1). \qquad (3.4)$$

Por otro lado, en la literatura estadística no existen modelos para distribuciones conjuntas multivariadas discretas, por lo que una de las principales metodologías propuestas en este sentido es la presentada por Song (2000). En esta metodología se establece que dado el vector aleatorio discreto $\mathbf{Z} = (Z_1,\ldots,Z_m)$, su función de probabilidad conjunta $f_{\mathbf{z}}$ se obtiene aplicando la derivada de Radon-Nikodym a la Ecuación (3.1) con respecto a la medida de conteo, resultando la expresión

$$f_{\mathbf{z}}(z_1,\ldots,z_m) = \sum_{k_1=1}^{2}\cdots\sum_{k_m=1}^{2}(-1)^{k_1+\cdots+k_m}\, C_{1\cdots m}(u_{1k_1},\ldots,u_{mk_m}), \qquad (3.5)$$

donde $u_{j1} = F_j(z_j)$ y $u_{j2} = F_j(z_j-1),\ j=1,\ldots,m$. En el caso bivariado (3.5) se simplifica a

$$
\begin{aligned}
f_{\mathbf{z}}(z_1,z_2) &= C_{12}[F_1(z_1),F_2(z_2)] - C_{12}[F_1(z_1),F_2(z_2-1)] - \\
&\quad C_{12}[F_1(z_1-1),F_2(z_2)] + C_{12}[F_1(z_1-1),F_2(z_2-1)],
\end{aligned}
$$

por lo cual, la función de probabilidad condicional $f_{1|2}$ queda expresada como

$$
\begin{aligned}
f_{1|2}(z_1|z_2) &:= \frac{f_{\mathbf{z}}(z_1,z_2)}{f_2(z_2)} \\
&= \{C_{12}[F_1(z_1),F_2(z_2)] - C_{12}[F_1(z_1),F_2(z_2-1)] - \\
&\quad C_{12}[F_1(z_1-1),F_2(z_2)] + C_{12}[F_1(z_1-1),F_2(z_2-1)]\}/f_2(z_2).
\end{aligned}
$$

El modelo dado por la Ecuación (3.5) contiene 2^m términos, lo que lo hace computacionalmente inmanejable para $m \geq 4$. Para evitar este inconveniente de tener que sumar tantos términos con las no tan accesibles cópulas multivariadas, en la siguiente sección se propone el uso de la metodología llamada *construcción con cópulas pareadas* (PCC, por sus siglas en inglés), en la cual se destaca la ventaja de que sólo requiere el uso de familias de cópulas bivariadas para el modelado de distribuciones conjuntas multivariadas.

3.2. Construcciones con cópulas pareadas (PCC)

3.2.1. PCC para variables continuas

Las PCC para variables continuas fueron establecidas formalmente por Aas et al. (2009). La función de densidad conjunta de un vector aleatorio continuo $\mathbf{Z} = (Z_1, \ldots, Z_m)$ puede ser factorizada como un producto de funciones de densidad condicionales de la siguiente manera

$$f_{\mathbf{z}}(z_1, \ldots, z_m) = f_{1|2\ldots m}(z_1|z_2, \ldots, z_m)\ f_{2|3\ldots m}(z_2|z_3, \ldots, z_m)\ \cdots\ f_{m-1|m}(z_{m-1}|z_m)\ f_m(z_m), \quad (3.6)$$

donde cada factor del lado derecho tiene la forma general $f_{j|\mathbf{j}}(z_j|\mathbf{v})$ siendo $\mathbf{v}$ un subconjunto de $\mathbf{z}$, mientras que $\mathbf{j}$ son los subíndices de los elementos de $\mathbf{v}$. Sea v_h cualquier elemento escalar de $\mathbf{v}$ y $\mathbf{v}_{\setminus h}$ el subconjunto de $\mathbf{v}$ que no incluye a v_h (i.e., $\mathbf{v}_{\setminus h} := \mathbf{v} \setminus \{v_h\}$), con z_j no perteneciente a $\mathbf{v}$. Panagiotelis et al. (2012) establecen que cada factor en la Ecuación (3.6) está dado por

$$f_{j|\mathbf{j}}(z_j|\mathbf{v}) = \frac{f_{jh|\setminus h}(z_j|\mathbf{v}_{\setminus h}, v_h|\mathbf{v}_{\setminus h})}{f_{h|\setminus h}(v_h|\mathbf{v}_{\setminus h})},$$

donde al numerador se le puede aplicar el resultado de la Ecuación (3.3) para obtener

$$\begin{aligned} f_{j|\mathbf{j}}(z_j|\mathbf{v}) &= \frac{c_{jh|\setminus h}[F_{j|\setminus h}(z_j|\mathbf{v}_{\setminus h}), F_{h|\setminus h}(v_h|\mathbf{v}_{\setminus h})]\ f_{j|\setminus h}(z_j|\mathbf{v}_{\setminus h})\ f_{h|\setminus h}(v_h|\mathbf{v}_{\setminus h})}{f_{h|\setminus h}(v_h|\mathbf{v}_{\setminus h})} \\ &= c_{jh|\setminus h}[F_{j|\setminus h}(z_j|\mathbf{v}_{\setminus h}), F_{h|\setminus h}(v_h|\mathbf{v}_{\setminus h})]\ f_{j|\setminus h}(z_j|\mathbf{v}_{\setminus h}), \end{aligned} \quad (3.7)$$

cuya función de densidad de la cópula pareada (bivariada) $C_{jh|\setminus h}$ se define como

$$c_{jh|\setminus h}[F_{j|\setminus h}(z_j|\mathbf{v}_{\setminus h}), F_{h|\setminus h}(v_h|\mathbf{v}_{\setminus h})] := \frac{\partial^2 C_{jh|\setminus h}[F_{j|\setminus h}(z_j|\mathbf{v}_{\setminus h}), F_{h|\setminus h}(v_h|\mathbf{v}_{\setminus h})]}{\partial F_{j|\setminus h}(z_j|\mathbf{v}_{\setminus h})\ \partial F_{h|\setminus h}(v_h|\mathbf{v}_{\setminus h})}.$$

Joe (1996) demostró que los argumentos de la cópula pareada en la Ecuación (3.7) son funciones de distribución condicionales que pueden ser evaluadas como

$$F_{k|\setminus h}(u_k|\mathbf{v}_{\setminus h}) = \frac{\partial C_{jh|\setminus h}[F_{j|\setminus h}(z_j|\mathbf{v}_{\setminus h}), F_{h|\setminus h}(v_h|\mathbf{v}_{\setminus h})]}{\partial F_{h|\setminus h}(v_h|\mathbf{v}_{\setminus h})}, \quad (3.8)$$

donde $u_k = z_j$ o bien $u_k = v_h$.

La Ecuación (3.7) puede ser aplicada recursivamente a cada producto en la Ecuación (3.6) hasta que la función de densidad conjunta $f_{\mathbf{z}}$ quede factorizada como el producto de $m(m-1)/2$ cópulas pareadas y m funciones de densidad marginales. Como ejemplo ilustrativo, considérese el vector continuo $\mathbf{z} = (z_1, z_2, z_3)$ cuya función de densidad conjunta puede ser factorizada como

$$f_{\mathbf{z}}(z_1, z_2, z_3) = f_{1|23}(z_1|z_2, z_3)\ f_{2|3}(z_2|z_3)\ f_3(z_3),$$

donde $f_3(z_3)$ es la función de densidad marginal de z_3, y de acuerdo a la Ecuación (3.4)

$$f_{2|3}(z_2|z_3) = c_{23}[F_2(z_2), F_3(z_3)]\ f_2(z_2),$$

con una apropiada función de densidad de la cópula pareada c_{23}. Por otro lado, si se elige $v_h = z_2$ entonces $\mathbf{v}_{\setminus h} = z_3$, y aplicando la Ecuación (3.7) y de nuevo la Ecuación (3.4) se obtiene

$$\begin{aligned} f_{1|23}(z_1|z_2, z_3) &= c_{12|3}[F_{1|3}(z_1|z_3), F_{2|3}(z_2|z_3)]\ f_{1|3}(z_1|z_3) \\ &= c_{12|3}[F_{1|3}(z_1|z_3), F_{2|3}(z_2|z_3)]\ c_{13}[F_1(z_1), F_3(z_3)]\ f_1(z_1), \end{aligned}$$

para cópulas pareadas $c_{12|3}$ y c_{13} apropiadas. De manera análoga, también se pudo haber elegido $v_h = z_3$ por lo que $\mathbf{v}_{\setminus h} = z_2$, y en consecuencia se tendría que

$$\begin{aligned} f_{1|23}(z_1|z_2, z_3) &= c_{13|2}[F_{1|2}(z_1|z_2), F_{3|2}(z_3|z_2)]\ f_{1|2}(z_1|z_2) \\ &= c_{13|2}[F_{1|2}(z_1|z_2), F_{3|2}(z_3|z_2)]\ c_{12}[F_1(z_1), F_2(z_2)]\ f_1(z_1), \end{aligned}$$

donde $c_{13|2}$ es diferente a la $c_{12|3}$ de la primera descomposición. Finalmente, utilizando todos estos resultados se encuentra que la función de densidad conjunta $f_{\mathbf{z}}$ puede expresarse como el producto de tres cópulas pareadas y las tres funciones de densidad marginales correspondientes:

$$\begin{aligned} f_{\mathbf{z}}(z_1, z_2, z_3) &= c_{12|3}[F_{1|3}(z_1|z_3), F_{2|3}(z_2|z_3)]\ c_{13}[F_1(z_1), F_3(z_3)]\ c_{23}[F_2(z_2), F_3(z_3)] \times \\ &\quad f_1(z_1)\ f_2(z_2)\ f_3(z_3) \\ &= c_{13|2}[F_{1|2}(z_1|z_2), F_{3|2}(z_3|z_2)]\ c_{12}[F_1(z_1), F_2(z_2)]\ c_{23}[F_2(z_2), F_3(z_3)] \times \\ &\quad f_1(z_1)\ f_2(z_2)\ f_3(z_3). \end{aligned}$$

En resumen, bajo ciertas condiciones de regularidad apropiadas, una función de densidad conjunta m-variada puede ser expresada como un producto de $m(m-1)/2$ cópulas pareadas y de m funciones de densidad marginales. Además, queda claro que por su propia naturaleza

PCC es un procedimiento iterativo, y que dada una factorización específica, aún hay muchas re-parametrizaciones diferentes; es decir, como v_h puede ser cualquier elemento de $\mathbf{v}$ en la Ecuación (3.7), hay muchas maneras en las cuales una función de densidad conjunta puede ser descompuesta de este modo. Sin embargo, lo más práctico es encontrar un conjunto de descomposiciones donde los argumentos de las cópulas pareadas puedan ser calculados usando la Ecuación (3.8). Las diferentes maneras en las cuales una función de densidad conjunta puede ser descompuesta para satisfacer esta última condición, se presentan sistematizadas en términos de modelos gráficos llamados *vines* (Bedford and Cooke, 2001a,b, 2002).

3.2.2. Teoría de gráficas: vines

Una *gráfica* o *grafo* es un par ordenado $G = (N, A)$ que consiste de un conjunto de *nodos* N y de un conjunto de *aristas* A, donde cada elemento $a \in A$ es un subconjunto de dos elementos de N. Un *camino* de longitud n en G es una sucesión de aristas $(a_1, a_2, \ldots, a_n)$ en la cual aristas consecutivas comparten un nodo común. Un camino *cerrado* es aquel en el que coinciden el primero y el último nodo. Un *ciclo* es un camino cerrado que no admite repetición de nodos ni aristas. Un grafo G es *conexo* si para cada par de nodos diferentes (n_i, n_j) existe al menos un camino posible entre n_i y n_j. Un *árbol* T es un grafo conexo que no tiene ciclos. Un *bosque* B es un conjunto de árboles.

Bedford and Cooke (2001a,b, 2002) introdujeron un importante grafo denominado *regular vine* (R-vine), el cual es una sucesión de árboles con m nodos iniciales $B = (T_1, T_2, \ldots, T_{m-1})$, $T_i = (N_i, A_i)$, $i = 1, 2, \ldots, m-1$, conectados bajo las siguientes condiciones:

(i) $N_1 = \{1, 2, \ldots, m\}$ (El árbol 1 tiene nodos $1, 2, \ldots, m$).

(ii) Para $i \in 2, \ldots, m-1, N_i = A_{i-1}$ (El nodo del árbol i es la arista del árbol $i-1$).

(iii) Para $i \in 2, \ldots, m-1, \forall a = \{n_j, n_k\} \in A_i$, se cumple que $n_j \cap n_k$ tiene sólo un elemento, i.e. $\#(n_j \cap n_k) = 1$ (Si dos nodos en el árbol i están conectados por una arista, las aristas correspondientes en el árbol $i-1$ deben tener un nodo común (*condición de proximidad*)).

Como el número de posibles R-vines en m dimensiones es enorme $\left[m!/2 \cdot 2^{\binom{m-2}{2}}\right]$, muchos autores prefieren utilizar dos casos especiales de R-vines llamados *drawable vines* (D-vines) y *canonical vines* (C-vines), los cuales se definen de la siguiente manera (Stöber, 2013). Un R-vine $B = (T_1, T_2, \ldots, T_{m-1})$ es llamado

- D-vine si $\forall\ i = 1, \ldots, m-1$, con $n \in N_i$, se cumple que $\#\{a \in A_i | n \in a\} \leq 2$ (El número de aristas que conectan a cada nodo es a lo más de dos).
- C-vine si $\forall\ i = 1, \ldots, m-1$, se cumple que $\exists\ n \in N_i$ tal que $\#\{a \in A_i | n \in a\} = m - i$ (En el árbol i existe un único nodo conectado a $m - i$ aristas).

En la Figura 3.3 se muestran las gráficas de un D-vine y de un C-vine en 4 dimensiones (4 nodos iniciales) que consisten de 3 árboles y 6 aristas.

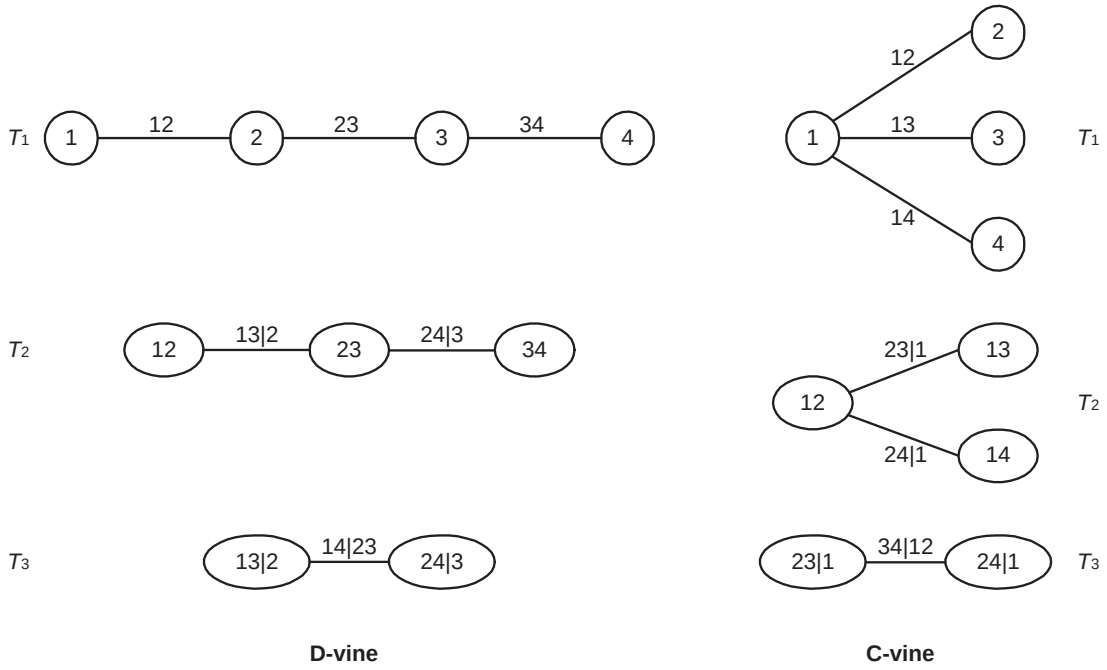

Figura 3.3. Gráficas de un D-vine y de un C-vine en 4 dimensiones.

Retomando el ejemplo ilustrativo de la subsección 3.2.1, se estableció que una de las opciones para la descomposición de la función de densidad conjunta en términos de cópulas pareadas queda expresada como

$$f_{\mathbf{z}}(z_1, z_2, z_3) = c_{13|2}[F_{1|2}(z_1|z_2), F_{3|2}(z_3|z_2)]\ c_{12}[F_1(z_1), F_2(z_2)]\ c_{23}[F_2(z_2), F_3(z_3)]\ f_1(z_1)\ f_2(z_2)\ f_3(z_3),$$

o bien, reordenando los factores y obviando sus argumentos se tiene que

$$f_{\mathbf{z}} = f_1\ f_2\ f_3\ c_{12}\ c_{23}\ c_{13|2}. \tag{3.9}$$

Si se comparan los subíndices en el lado derecho de la Ecuación 3.9 con la parte correspondiente de los dos primeros árboles T_1 y T_2 del D-vine de la Figura 3.3, se pueden establecer en general

las siguientes relaciones. Los nodos del primer árbol corresponden a las funciones de densidad marginales de las variables involucradas, mientras que las aristas de los árboles corresponden a las cópulas pareadas necesarias para la factorización de una función de densidad conjunta dentro de la metodología PCC. Además, a partir del segundo árbol los nodos de éstos corresponden a las aristas del árbol que les precede. Por otro lado, los nodos de los árboles sirven para etiquetar a las aristas que los unen; por ejemplo, en el árbol T_3 del D-vine los números no repetidos en los nodos 13|2 y 24|3 son el 1 y el 4 que forman la parte condicionada de la arista que los une, mientras que los números repetidos 2 y 3 forman la parte condicionante de dicha arista, i.e., 14|23. Nótese que el método gráfico basado en vines no es estrictamente necesario para aplicar la metodología PCC, pero ayuda en mucho a identificar eficientemente las diferentes descomposiciones por medio de cópulas pareadas, sobre todo cuando el número de variables a modelar es relativamente alto.

Bedford and Cooke (2001a,b) generalizaron una función de densidad conjunta m-variada en términos de un R-vine, mientras que Aas et al. (2009) la particularizaron en términos de un D-vine como

$$f_{\mathbf{z}}(z_1,\ldots,z_m)=\prod_{k=1}^{m} f_k \prod_{j=1}^{m-1}\prod_{i=1}^{m-j} c_{i,i+j|i+1,\ldots,i+j-1}[F_{i|i+1,\ldots,i+j-1}, F_{i+j|i+1,\ldots,i+j-1}],$$

donde el índice j identifica a los árboles, mientras que i identifica a las aristas en cada árbol. Aas et al. (2009) también la particularizaron en términos de un C-vine como

$$f_{\mathbf{z}}(z_1,\ldots,z_m)=\prod_{k=1}^{m} f_k \prod_{j=1}^{m-1}\prod_{i=1}^{m-j} c_{j,j+i|1,\ldots,j-1}[F_{j|1,\ldots,j-1}, F_{j+i|1,\ldots,j-1}].$$

Aplicar la estructura C-vine podría ser muy adecuado cuando se sabe de antemano que una variable en particular es clave para las interacciones en un conjunto de datos. En tal situación uno podría decidir colocar dicha variable en la raíz del C-vine, tal como aparece la variable 1 en el árbol T_1 del C-vine de la Figura 3.3.

Continuando con el ejemplo ilustrativo anterior para el vector aleatorio $\mathbf{z}=(z_1,z_2,z_3)$, existen en general seis maneras de permutar z_1, z_2 y z_3, pero sólo tres de las seis permutaciones ofrecen diferentes descomposiciones para $f_{\mathbf{z}}$, que son a la vez D-vine y C-vine. Estas tres permutaciones con estructura D-vine se muestran a continuación con subíndice D, mientras que sus permutaciones equivalentes con estructura C-vine se muestran con subíndice C. El símbolo $\sim$ denota equivalencia entre permutaciones en el sentido de que ofrecen la misma descomposición para $f_{\mathbf{z}}$.

(1) $(z_1, z_2, z_3)_D \sim (z_3, z_2, z_1)_D \sim (z_2, z_1, z_3)_C \sim (z_2, z_3, z_1)_C$ generan la factorización

$$f_{\mathbf{z}} = f_1 \; f_2 \; f_3 \; c_{12} \; c_{23} \; c_{13|2}.$$

(2) $(z_1, z_3, z_2)_D \sim (z_2, z_3, z_1)_D \sim (z_3, z_1, z_2)_C \sim (z_3, z_2, z_1)_C$ generan la factorización

$$f_{\mathbf{z}} = f_1 \; f_2 \; f_3 \; c_{13} \; c_{23} \; c_{12|3}.$$

(3) $(z_2, z_1, z_3)_D \sim (z_3, z_1, z_2)_D \sim (z_1, z_2, z_3)_C \sim (z_1, z_3, z_2)_C$ generan la factorización

$$f_{\mathbf{z}} = f_1 \; f_2 \; f_3 \; c_{12} \; c_{13} \; c_{23|1}.$$

A modo de ilustración adicional considérese el vector aleatorio continuo $\mathbf{z} = (z_1, z_2, z_3, z_4)$. Su función de densidad conjunta $f_{\mathbf{z}}$ puede ser expresada mediante una estructura D-vine de la siguiente manera (ver Figura 3.3):

$$f_{\mathbf{z}} = f_1 \; f_2 \; f_3 \; f_4 \; c_{12} \; c_{23} \; c_{34} \; c_{13|2} \; c_{24|3} \; c_{14|23},$$

o bien, con una estructura C-vine como

$$f_{\mathbf{z}} = f_1 \; f_2 \; f_3 \; f_4 \; c_{12} \; c_{13} \; c_{14} \; c_{23|1} \; c_{24|1} \; c_{34|12}.$$

En este caso con cuatro variables, hay en total 24 permutaciones que proporcionan igual número de descomposiciones diferentes, de las cuales 12 son D-vine y 12 son C-vine, y ninguna de las descomposiciones D-vine es igual a alguna descomposición C-vine, o viceversa. De hecho, no hay otras descomposiciones posibles R-vine. Como dato adicional, en el caso de cinco variables hay 240 posibles descomposiciones diferentes mediante PCC, de las cuales 60 son D-vine, 60 son C-vine y 120 tienen estructura R-vine que no son D-vine ni C-vine. En general, para m variables hay en total $m!/2$ D-vines diferentes y el mismo número de C-vines también diferentes.

Mientras que los autores usualmente han restringido la clase de R-vines a las subclases anteriores de C-vines y D-vines, los algoritmos desarrollados por Dißmann (2010) hacen a la clase completa de cópulas R-vine más accesible (citado en Stöber, 2013, p. 22). Las técnicas computacionales presentadas por Dißmann han sido desarrolladas aún más y están disponibles para profesionales e investigadores en el paquete `VineCopula` (Schepsmeier et al., 2012) del lenguaje estadístico R (R Core Team, 2014). Mientras que este paquete y mucha de la literatura sobre el tema consideran la metodología PCC sólo para distribuciones donde todas las marginales univariadas son continuas, el principio es más general. Panagiotelis et al. (2012) presentan una discusión pionera del método PCC basado en una estructura D-vine para datos discretos, la cual se verá a continuación.

3.2.3. PCC para variables discretas

Utilizando una notación similar y un procedimiento análogo al de PCC para variables continuas, la función de probabilidad conjunta de un vector aleatorio discreto $\mathbf{Z} = (Z_1, \ldots, Z_m)$ puede ser factorizada como un producto de funciones de probabilidad condicionales de la siguiente manera

$$f_{\mathbf{z}}(z_1, \ldots, z_m) = f_{1|2\ldots m}(z_1|z_2, \ldots, z_m)\ f_{2|3\ldots m}(z_2|z_3, \ldots, z_m)\ \cdots\ f_{m-1|m}(z_{m-1}|z_m)\ f_m(z_m), \quad (3.10)$$

donde cada factor del lado derecho tiene la forma general $f_{j|\mathbf{j}}(z_j|\mathbf{v})$ siendo $\mathbf{v}$ un subconjunto de $\mathbf{z}$, mientras que $\mathbf{j}$ son los subíndices de los elementos de $\mathbf{v}$. Sea v_h cualquier elemento escalar de $\mathbf{v}$ y $\mathbf{v}_{\setminus h}$ el subconjunto de $\mathbf{v}$ que no incluye a v_h (i.e., $\mathbf{v}_{\setminus h} := \mathbf{v} \setminus \{v_h\}$), con z_j no perteneciente a $\mathbf{v}$. Panagiotelis et al. (2012) establecen que cada factor en la Ecuación (3.10) está dado por

$$\begin{aligned} f_{j|\mathbf{j}}(z_j|\mathbf{v}) &= \frac{f_{jh|\setminus h}(z_j|\mathbf{v}_{\setminus h}, v_h|\mathbf{v}_{\setminus h})}{f_{h|\setminus h}(v_h|\mathbf{v}_{\setminus h})} \\ &= \frac{\sum_{i_j=0}^{1}\sum_{i_h=0}^{1}(-1)^{i_j+i_h}F_{jh|\setminus h}(z_j - i_j|\mathbf{v}_{\setminus h}, v_h - i_h|\mathbf{v}_{\setminus h})}{f_{h|\setminus h}(v_h|\mathbf{v}_{\setminus h})}, \end{aligned}$$

donde al numerador se le puede aplicar la Ecuación (3.1) para obtener

$$f_{j|\mathbf{j}}(z_j|\mathbf{v}) = \frac{\sum_{i_j=0}^{1}\sum_{i_h=0}^{1}(-1)^{i_j+i_h}C_{jh|\setminus h}[F_{j|\setminus h}(z_j - i_j|\mathbf{v}_{\setminus h}), F_{h|\setminus h}(v_h - i_h|\mathbf{v}_{\setminus h})]}{f_{h|\setminus h}(v_h|\mathbf{v}_{\setminus h})}, \quad (3.11)$$

que es el análogo discreto de la Ecuación (3.7), y puede ser aplicado recursivamente a la Ecuación (3.10) para descomponer la función de probabilidad conjunta en términos de cópulas pareadas. Los argumentos de la cópula pareada en la Ecuación (3.11) son funciones de distribución condicionales que pueden ser evaluadas mediante la expresión

$$\begin{aligned} F_{k|\setminus h}(u_k - i_k|\mathbf{v}_{\setminus h}) &= \{C_{jh|\setminus h}[F_{j|\setminus h}(z_j|\mathbf{v}_{\setminus h}), F_{h|\setminus h}(v_h|\mathbf{v}_{\setminus h})] - \\ &\quad C_{jh|\setminus h}[F_{j|\setminus h}(z_j|\mathbf{v}_{\setminus h}), F_{h|\setminus h}(v_h - 1|\mathbf{v}_{\setminus h})]\}/f_{h|\setminus h}(v_h|\mathbf{v}_{\setminus h}), \end{aligned}$$

donde $u_k = z_j$ o bien $u_k = v_h$, y es el análogo discreto de la Ecuación (3.8).

Si se considera que $\mathbf{Z} = (Z_1, \ldots, Z_m)$ es un vector aleatorio de m variables categóricas, entonces la función de distribución marginal correspondiente a Z_j $(j = 1, \ldots, m)$ está dada por

$$F_j(z) = \begin{cases} 0 & \text{si} \quad z < 1 \\ \sum_{l=1}^{\lfloor z \rfloor} q_{jl} & \text{si} \quad 1 \leq z < b_j \\ 1 & \text{si} \quad z \geq b_j, \end{cases} \quad (3.12)$$

donde $\lfloor z \rfloor = \text{máx}\{l \in \mathbb{Z} | l \leq z\}$ es la función piso, $\mathbb{Z}$ es el conjunto de los enteros, b_j es el número de niveles posibles que la j-ésima variable puede tomar, y $q_{jl} := \Pr\{\text{la } j\text{-ésima variable toma el } l\text{-ésimo nivel posible}\}$. La presencia del vector de variables explicativas completamente observadas $\mathbf{x}$

en cada probabilidad marginal de $f_{\mathbf{z}}$ puede ser modelada usando un modelo logístico multinomial de la siguiente manera:

$$q_{jl} = \frac{\exp(\alpha_{jl} + \boldsymbol{\alpha}_{jl}^T \mathbf{x})}{\sum_{r=1}^{a_j} \exp(\alpha_{jr} + \boldsymbol{\alpha}_{jr}^T \mathbf{x})}, \tag{3.13}$$

donde α_{jl} y $\boldsymbol{\alpha}_{jl}$ son los parámetros del factor j y nivel l ($l = 1, \ldots, a_j$). Para evitar redundancia, α_{ja_j} y $\boldsymbol{\alpha}_{ja_j}$ se han igualado a cero. La razón para modelar con cópulas pareadas la función de probabilidad conjunta sólo del vector $\mathbf{z}$ dado el vector $\mathbf{x}$, y no modelar la conjunta del vector completo $(\mathbf{z}, \mathbf{x})$, es que para mantener esta metodología propuesta lo más general posible, el vector $\mathbf{x}$ puede ser continuo, pero en este caso el método PCC es sólo para variables discretas.

De manera análoga, dado que $\mathbf{r} = (r_1, \ldots, r_m)$ es un vector aleatorio de m variables dicotómicas, entonces la función de distribución marginal correspondiente a r_j ($j = 1, \ldots, m$) está dada por

$$\mathcal{F}_j(r) = \begin{cases} 0 & \text{si} \quad r < 0 \\ 1 - p_j & \text{si} \quad 0 \leq r < 1 \\ 1 & \text{si} \quad r \geq 1, \end{cases} \tag{3.14}$$

donde $p_j := \Pr\{r_j = 1\}$ es la probabilidad de que la j-ésima entrada de $\mathbf{z}$ sea un valor observado. La presencia del vector $\mathbf{w}$ en cada probabilidad marginal de $f_{\mathbf{r}}$ puede ser modelada usando un modelo logístico binomial de la siguiente manera:

$$p_j = \frac{\exp(\boldsymbol{\nu}_j^T \mathbf{w})}{1 + \exp(\boldsymbol{\nu}_j^T \mathbf{w})}, \tag{3.15}$$

donde $\boldsymbol{\nu}_j$ es el vector de parámetros de la j-ésima variable indicadora.

Como cópula pareada básica para la aplicación de la metodología PCC se eligió a la familia Frank, la cual se define como

$$C(v_1, v_2; \rho) = -\frac{1}{\rho} \log \left\{ 1 + \frac{[\exp(-\rho v_1) - 1][\exp(-\rho v_2) - 1]}{\exp(-\rho) - 1} \right\},$$

donde $\rho \neq 0$ es el parámetro que controla la dependencia entre las marginales. La elección de la cópula de Frank es apropiada ya que es de las pocas familias capaces de capturar el rango completo de dependencia. Este incluye la cópula cota inferior de Fréchet cuando $\rho \to -\infty$, la cópula cota superior de Fréchet cuando $\rho \to \infty$, así como la cópula producto que ocurre cuando $\rho \to 0$ definiendo al modelo independiente $v_1 v_2$. Además, a diferencia de la cópula Gaussiana que también captura el rango completo de dependencia, la cópula de Frank es más fácil de programar y permite que el tiempo de ejecución del código que la contiene sea mucho más breve. De hecho, para este proyecto se codificó en C++ la cópula de Frank mediante el paquete `Rcpp` (Eddelbuettel and François, 2011). El código guardado en el archivo `frankC.cpp` quedo de la siguiente manera:

```
#include <Rcpp.h>
// [[Rcpp::export()]]
Rcpp::NumericVector frankC(Rcpp::NumericVector u, Rcpp::NumericVector v,
double rho)
{
     Rcpp::NumericVector fC;
     if(rho==0) fC = u*v;
     else fC = (-1/rho)*log(1+(exp(-rho*u)-1)*(exp(-rho*v)-1)/(exp(-rho)-1));
     return fC;
}
```

Panagiotelis et al. (2012) proporcionan un algoritmo para obtener la función de probabilidad conjunta de un vector de variables aleatorias discretas m-variado usando la metodología PCC con estructura D-vine. El algoritmo se puede generalizar a estructuras C-vine o R-vine. Dado el vector aleatorio discreto $\mathbf{Z} = (Z_1, \ldots, Z_m)$, por simplicidad y sin pérdida de generalidad se asume que Z_j $(j = 1, \ldots, m) \in \mathbb{N}$. En el algoritmo se utilizarán las siguientes definiciones:

$F_j^+ := F_j(z_j)$,
$F_j^- := F_j(z_j - 1)$,
$f_j := F_j^+ - F_j^-$,
$C_{j\ j+1}^{++} := C_{j\ j+1}(F_j^+, F_{j+1}^+; \rho_{j\ j+1})$,
$C_{j\ j+1}^{+-} := C_{j\ j+1}(F_j^+, F_{j+1}^-; \rho_{j\ j+1})$,
$C_{j\ j+1}^{-+} := C_{j\ j+1}(F_j^-, F_{j+1}^+; \rho_{j\ j+1})$,
$C_{j\ j+1}^{--} := C_{j\ j+1}(F_j^-, F_{j+1}^-; \rho_{j\ j+1})$,

donde $F_j(\cdot)$ deberá ser evaluada según la Ecuación (3.12), y $\rho_{j\ j+1}$ es el parámetro que caracteriza a la cópula pareada de Frank.

El algoritmo que se describe a continuación se representa de manera gráfica en la Figura 3.4 con cinco variables aleatorias como caso especial.

1. Para $j = 1, \ldots, m$, evaluar
   F_j^+,
   F_j^- y
   f_j.

2. Para $j = 1, \ldots, m-1$, evaluar

   $C^{++}_{j\ j+1}$,
   $C^{+-}_{j\ j+1}$,
   $C^{-+}_{j\ j+1}$ y
   $C^{--}_{j\ j+1}$.

3. Para $j = 1, \ldots, m-2$, evaluar

   a) $F^{+}_{j|j+1} = (C^{++}_{j\ j+1} - C^{+-}_{j\ j+1})/f_{j+1}$,
   $F^{-}_{j|j+1} = (C^{-+}_{j\ j+1} - C^{--}_{j\ j+1})/f_{j+1}$ y
   $f_{j|j+1} = F^{+}_{j|j+1} - F^{-}_{j|j+1}$.

   b) $F^{+}_{j+2|j+1} = (C^{++}_{j+1\ j+2} - C^{-+}_{j+1\ j+2})/f_{j+1}$,
   $F^{-}_{j+2|j+1} = (C^{+-}_{j+1\ j+2} - C^{--}_{j+1\ j+2})/f_{j+1}$ y
   $f_{j+2|j+1} = F^{+}_{j+2|j+1} - F^{-}_{j+2|j+1}$.

   c) $C^{ab}_{j\ j+2|j+1}(F^{a}_{j|j+1}, F^{b}_{j+2|j+1})$, con $a, b \in \{+, -\}$.

4. Para $t = 3, \ldots, m-1$ y $j = 1, \ldots, m-t$, evaluar

   a) $F^{+}_{j|j+1\ldots j+t-1} = (C^{++}_{j\ j+t-1|j+1\ldots j+t-2} - C^{+-}_{j\ j+t-1|j+1\ldots j+t-2})/f_{j+t-1|j+1\ldots j+t-2}$,
   $F^{-}_{j|j+1\ldots j+t-1} = (C^{-+}_{j\ j+t-1|j+1\ldots j+t-2} - C^{--}_{j\ j+t-1|j+1\ldots j+t-2})/f_{j+t-1|j+1\ldots j+t-2}$ y
   $f_{j|j+1\ldots j+t-1} = F^{+}_{j|j+1\ldots j+t-1} - F^{-}_{j|j+1\ldots j+t-1}$.

   b) $F^{+}_{j+t|j+1\ldots j+t-1} = (C^{++}_{j+1\ j+t|j+2\ldots j+t-1} - C^{-+}_{j+1\ j+t|j+2\ldots j+t-1})/f_{j+1|j+2\ldots j+t-1}$,
   $F^{-}_{j+t|j+1\ldots j+t-1} = (C^{+-}_{j+1\ j+t|j+2\ldots j+t-1} - C^{--}_{j+1\ j+t|j+2\ldots j+t-1})/f_{j+1|j+2\ldots j+t-1}$ y
   $f_{j+t|j+1\ldots j+t-1} = F^{+}_{j+t|j+1\ldots j+t-1} - F^{-}_{j+t|j+1\ldots j+t-1}$.

   c) $C^{ab}_{j\ j+t|j+1\ldots j+t-1}(F^{a}_{j|j+1\ldots j+t-1}, F^{b}_{j+t|j+1\ldots j+t-1})$, con $a, b \in \{+, -\}$.

5. Evaluar

   $F^{+}_{1|2\ldots m} = (C^{++}_{1\ m|2\ldots m-1} - C^{+-}_{1\ m|2\ldots m-1})/f_{m|2\ldots m-1}$,
   $F^{-}_{1|2\ldots m} = (C^{-+}_{1\ m|2\ldots m-1} - C^{--}_{1\ m|2\ldots m-1})/f_{m|2\ldots m-1}$ y
   $f_{1|2\ldots m} = F^{+}_{1|2\ldots m} - F^{-}_{1|2\ldots m}$.

6. Finalmente, evaluar la función de probabilidad conjunta m-variada

   $f_{\mathbf{z}}(z_1, \ldots, z_m) = f_2 \cdot \prod_{k=3}^{m} f_{k|2\ldots k-1} \cdot f_{1|2\ldots m}$.

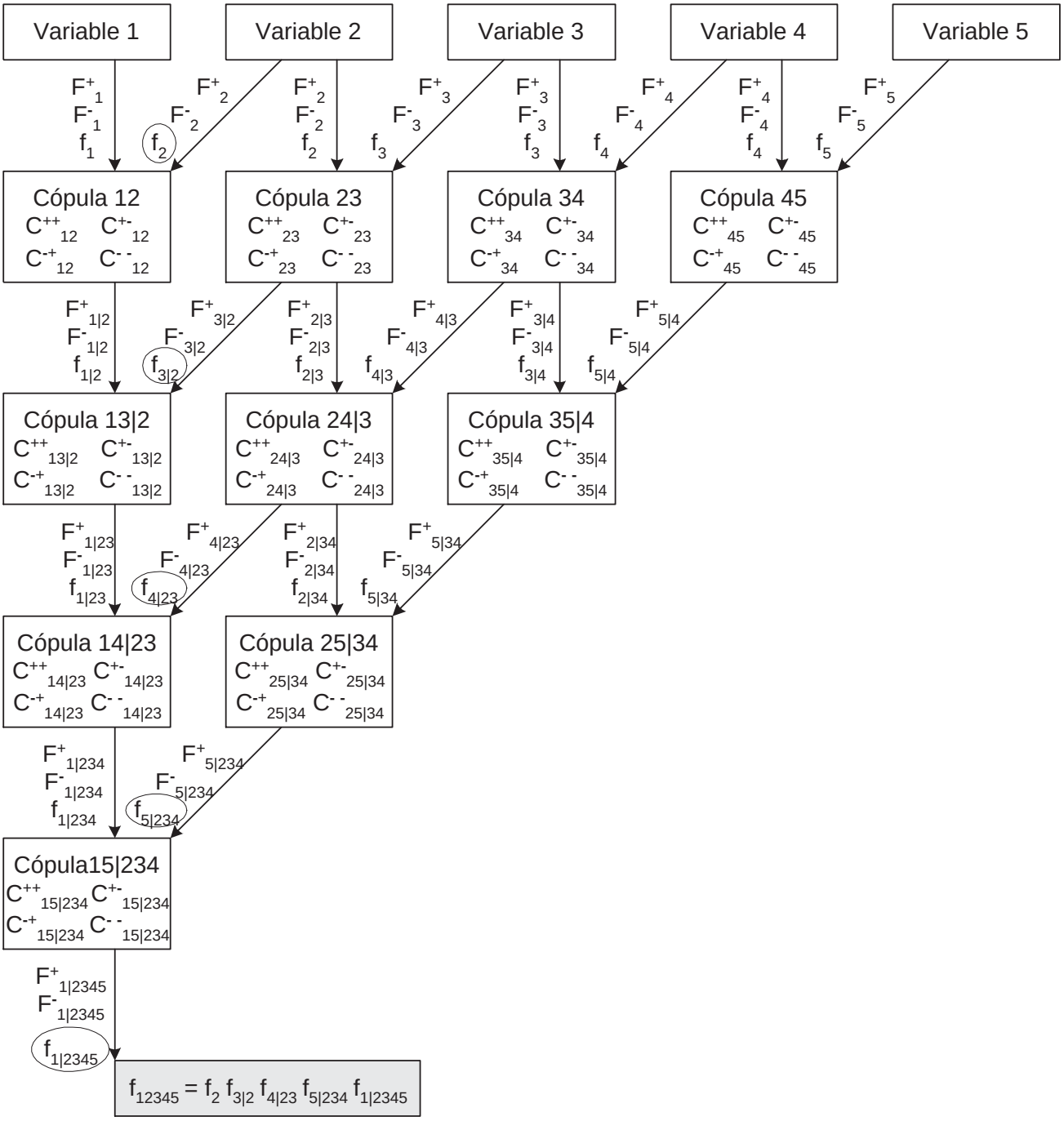

Figura 3.4. Representación gráfica de la estructura D-vine del algoritmo para cinco variables.

Como ejemplo ilustrativo de la aplicación del algoritmo, considérese el vector de variables aleatorias categóricas $\mathbf{z} = (z_1, z_2, z_3)$, donde $z_1 \in \{1, 2\}$, $z_2 \in \{1, 2, 3\}$ y $z_3 \in \{1, 2, 3, 4\}$. Sea $\mathbf{X}$ la matriz de diseño del vector de variables explicativas completamente observadas $\mathbf{x}$, entonces de acuerdo a la Ecuación (3.13) se tiene que

$$q_{11} = \frac{\exp(\mathbf{X}\boldsymbol{\alpha}_{11})}{1 + \exp(\mathbf{X}\boldsymbol{\alpha}_{11})};\ q_{2l} = \frac{\exp(\mathbf{X}\boldsymbol{\alpha}_{2l})}{1 + \sum_{r=1}^{2} \exp(\mathbf{X}\boldsymbol{\alpha}_{2r})}, l = 1, 2;\ q_{3l} = \frac{\exp(\mathbf{X}\boldsymbol{\alpha}_{3l})}{1 + \sum_{r=1}^{3} \exp(\mathbf{X}\boldsymbol{\alpha}_{3r})}, l = 1, 2, 3.$$

Mientras que de la Ecuación (3.12) se obtiene

$$F_1(z) = \begin{cases} 0 & \text{si} \quad z < 1 \\ q_{11} & \text{si} \quad 1 \leq z < 2 \\ 1 & \text{si} \quad z \geq 2 \end{cases} \qquad F_2(z) = \begin{cases} 0 & \text{si} \quad z < 1 \\ q_{21} & \text{si} \quad 1 \leq z < 2 \\ q_{21} + q_{22} & \text{si} \quad 2 \leq z < 3 \\ 1 & \text{si} \quad z \geq 3 \end{cases}$$

$$F_3(z) = \begin{cases} 0 & \text{si} \quad z < 1 \\ q_{31} & \text{si} \quad 1 \leq z < 2 \\ q_{31} + q_{32} & \text{si} \quad 2 \leq z < 3 \\ q_{31} + q_{32} + q_{33} & \text{si} \quad 3 \leq z < 4 \\ 1 & \text{si} \quad z \geq 4 \end{cases}$$

Después, siguiendo los pasos del algoritmo anterior se evalúa lo siguiente.

$$F_j^+ = F_j(z_j), \quad F_j^- = F_j(z_j - 1), \quad f_j = F_j^+ - F_j^-, \quad j = 1, 2, 3.$$

$$C_{12}^{ab} = C_{12}(F_1^a, F_2^b; \rho_{12}), \quad a, b \in \{+, -\}.$$
$$C_{23}^{ab} = C_{23}(F_2^a, F_3^b; \rho_{23}), \quad a, b \in \{+, -\}.$$

$$F_{1|2}^+ = (C_{12}^{++} - C_{12}^{+-})/f_2, \quad F_{1|2}^- = (C_{12}^{-+} - C_{12}^{--})/f_2, \quad f_{1|2} = F_{1|2}^+ - F_{1|2}^-.$$
$$F_{3|2}^+ = (C_{23}^{++} - C_{23}^{-+})/f_2, \quad F_{3|2}^- = (C_{23}^{+-} - C_{23}^{--})/f_2, \quad f_{3|2} = F_{3|2}^+ - F_{3|2}^-.$$

$$C_{13|2}^{ab} = C_{13|2}(F_{1|2}^a, F_{3|2}^b; \rho_{13|2}), \quad a, b \in \{+, -\}.$$

$$F_{1|23}^+ = (C_{13|2}^{++} - C_{13|2}^{+-})/f_{3|2}, \quad F_{1|23}^- = (C_{13|2}^{-+} - C_{13|2}^{--})/f_{3|2}, \quad f_{1|23} = F_{1|23}^+ - F_{1|23}^-.$$

$$f_{\mathbf{z}}(z_1, z_2, z_3) = f_2 \cdot f_{3|2} \cdot f_{1|23}.$$

Donde los parámetros a estimar son: $\boldsymbol{\alpha}_{11}$, $\boldsymbol{\alpha}_{21}$, $\boldsymbol{\alpha}_{22}$, $\boldsymbol{\alpha}_{31}$, $\boldsymbol{\alpha}_{32}$, $\boldsymbol{\alpha}_{33}$, ρ_{12}, ρ_{23} y $\rho_{13|2}$. Mediante un procedimiento análogo y usando las Ecuaciones (3.14) y (3.15) se puede estimar $f_{\mathbf{r}}(r_1, r_2, r_3)$.

Capítulo 4

Simulaciones, aplicación y resultados

4.1. Simulaciones

Para la validación del desempeño de la metodología aquí propuesta, a la que llamaremos *máxima verosimilitud con cópulas pareadas* (PCML, por sus siglas en inglés), se establecen las siguientes pautas generales para aplicarlas a diferentes escenarios con conjuntos de datos simulados mediante elementos del diseño y análisis de experimentos. (Box et al., 2005).

Elección de factores y niveles

Los siguientes factores y niveles que conformarán los diferentes escenarios de validación, se eligieron entre los de mayor influencia para el desempeño de PCML.

A: Tamaño del conjunto de datos (número de observaciones) con niveles $500^{(-)}$ y $1000^{(+)}$.

B: Cantidad de variables explicativas con datos faltantes con niveles $3^{(-)}$ y $5^{(+)}$.

C: Tipo de variables explicativas con datos faltantes con niveles *dicotómica*$^{(-)}$ y *tricotómica*$^{(+)}$.

D: Proporción de datos faltantes con niveles $30\,\%^{(-)}$ y $50\,\%^{(+)}$.

E: Mecanismo de datos faltantes con niveles $MAR^{(-)}$ y $MNAR^{(+)}$.

Elección del tipo de diseño experimental

Para explorar los efectos que tienen los cinco factores de dos niveles cada uno sobre el desempeño de PCML, una réplica completa de un diseño factorial 2^5 requeriría validar 32 escenarios diferentes, lo cual sería computacionalmente muy demandante. Es por ello que se eligió utilizar el diseño

factorial fraccionado $2^5/4 = 2^5/2^2 = 2^{5-2}$, que sólo requiere probar los ocho escenarios mostrados en la Tabla 4.1 sin limitar la base de inferencia y con viabilidad computacional. De las cuatro fracciones posibles se optó por usar la principal, ya que es la que tiene a todos los factores en sus niveles altos.

Tabla 4.1. Diseño factorial fraccionado 2^{5-2}.

	Factor				
Escenario	A	B	C	D = AB	E = AC
1	−	−	−	+	+
2	+	−	−	−	−
3	−	+	−	−	+
4	+	+	−	+	−
5	−	−	+	+	−
6	+	−	+	−	+
7	−	+	+	−	−
8	+	+	+	+	+

Para verificar la robustez de la cópula de Frank, se generaron vectores de variables explicativas con diversas estructuras de dependencia a partir de una cópula Gaussiana multivariada mediante la función `rCopula` del paquete `copula` (Hofert et al., 2015). La función `rCopula` genera un vector $\mathbf{v} = (\mathrm{v}_1, \ldots, \mathrm{v}_{m+1})$ de variables aleatorias continuas con distribución uniforme en el intervalo $(0, 1)$, de las cuales la primera se tomó como la variable explicativa completamente observada x, mientras que las restantes se discretizaron para formar el vector de variables explicativas parcialmente observadas $\mathbf{z} = (z_1, \ldots, z_m)$. Un ejemplo en particular de discretización es el siguiente:

$$z_{j(j=1,\ldots,m)} = \begin{cases} 0 & \text{si} \quad \mathrm{v}_{j+1} \leq 1/3 \\ 1 & \text{si} \quad 1/3 < \mathrm{v}_{j+1} \leq 2/3 \\ 2 & \text{si} \quad \mathrm{v}_{j+1} > 2/3, \end{cases}$$

obteniéndose m variables tipo factor de tres categorías cada una como caso especial. Después se realizaron 100 réplicas de lo siguiente. Se generó la variable respuesta y tipo Binomial, Poisson o Normal con funciones de enlace logística, logarítmica e identidad, respectivamente, y donde a los coeficientes de regresión $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_{m+1})$ se les asignó un valor fijo de antemano. De esta manera, se obtiene un conjunto de datos completo (DC) $\{x, \mathbf{z}, y\}$ al que aún no se le han eliminado valores, y cuya regresión inicial hará el papel de regresión de referencia con la cual se compararán los resultados de regresiones posteriores con datos faltantes. Usando el mismo método descrito anteriormente para obtener el vector $\mathbf{z}$, se generó el vector de variables dicotómicas indicadoras de datos faltantes $\mathbf{r} = (r_1, \ldots, r_m)$ simulando los mecanismos de pérdida

de datos MAR y MNAR. El paso siguiente fue eliminar los valores de las variables explicativas $z_1, \ldots, z_m$ correspondientes a los ceros de sus variables indicadoras respectivas $r_1, \ldots, r_m$. Una vez que el conjunto de datos con valores faltantes en $\mathbf{z}$ estuvo disponible, se le aplicó un análisis de regresión utilizando tres metodologías diferentes: (i) casos completos (CC) como método estándar de comparación, esto mediante la función `glm` en R, (ii) imputación múltiple (MI) como método de validación, esto mediante 30 imputaciones con el uso las funciones `mice`, `with` y `pool` del paquete `mice` (Van Buuren and Groothuis-Oudshoorn, 2011), y (iii) PCML bajo los mecanismos MAR y MNAR como la metodología aquí propuesta sujeta a validación.

Para aplicar máxima verosimilitud se recurrió al algoritmo EM vía ponderaciones, y en consecuencia lo primero que se tuvo que obtener fue el conjunto de datos aumentado con su respectiva columna de ponderaciones w, según el procedimiento mostrado en la Figura 2.1. El criterio de convergencia usado para el algoritmo EM fue que la diferencia entre estimadores sucesivos de los coeficientes de regresión $\boldsymbol{\beta}$ fuera menor de 10^{-6}. En la etapa E del algoritmo se calculan las ponderaciones w por medio de la Ecuación (2.8). En la etapa M del algoritmo se actualizan los valores de los coeficientes de regresión $\boldsymbol{\beta}$ mediante la Ecuación (2.12). En esta misma etapa se modelaron las distribuciones conjuntas $f_{\mathbf{z}}(\mathbf{z}|\mathbf{x};\boldsymbol{\alpha})$, y en su caso $f_{\mathbf{r}}(\mathbf{r}|\mathbf{x},\mathbf{z},y;\boldsymbol{\nu})$, utilizando PCML a través al algoritmo de la Figura 3.4. De esta manera, ya fue posible actualizar los valores los parámetros $\boldsymbol{\alpha}$ y $\boldsymbol{\nu}$ maximizando sus funciones log-verosimilitud respectivas mediante la función `nlminb` en R.

Con respecto a la estimación de los errores estándares de los coeficientes de regresión $\boldsymbol{\beta}$, en el caso de los métodos DC y CC estos se obtuvieron directamente de la función `glm`, siendo iguales a los obtenidos mediante la Ecuación (2.5). En cuanto al método MI se usaron los que proporciona la función `pool`, que a su vez utiliza las reglas de Rubin para calcularlos. Para PCML se utilizó el método de Louis (1982) dado por la Ecuación (2.14).

Los últimos cálculos hechos dentro de las 100 réplicas, fueron los relativos a los sesgos y a los errores cuadráticos medios de los estimadores de los coeficientes de regresión $\boldsymbol{\beta}$. El sesgo en valor absoluto se define como $\mathrm{B} = |\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}|$, donde $\boldsymbol{\beta}$ son los valores fijos asignados de antemano a los coeficientes, mientras que $\hat{\boldsymbol{\beta}}$ son los estimadores puntuales obtenidos. Sea SE los errores estándares de los estimadores, el error cuadrático medio de la estimación se define como $\mathrm{MSE} = \mathrm{B}^2 + \mathrm{SE}^2$ (Ibrahim et al., 1999b). Finalmente, como última etapa del proceso, se obtuvieron los promedios dentro de las 100 réplicas de todos los resultados anteriormente descritos. Para la ejecución de los

códigos en R, se utilizó una PC de escritorio HP Pavilion Slimline de 32 bits, con procesador Intel Core 2 Duo E4300 con Tecnología Intel Viiv de 1.8 GHz, sistema operativo Windows Vista Home Premium 2007, memoria RAM DDR2 de 2 núcleos con 1 GB cada uno.

En la Tabla 4.2 se muestran los resultados obtenidos para una respuesta con distribución Binomial, una estructura independiente entre las variables explicativas y las variables indicadoras, y establecido el escenario número ocho de la Tabla 4.1. Se resaltan en negritas los valores mínimos de los errores estándares, los sesgos y los errores cuadráticos medios entre los métodos CC, MI, PCML$_{\text{MAR}}$ y PCML$_{\text{MNAR}}$. El comportamiento fue similar para los siete escenarios restantes. En las Tablas 4.3 y 4.4 se muestran los resultados para respuestas Poisson y Normal, respectivamente, con estructuras independientes entre las variables explicativas y las variables indicadoras, y un escenario adicional que quedó establecido de la siguiente manera. El tamaño del conjunto de datos fue de 1000 observaciones, con una variable explicativa continua completamente observada, y con tres variables explicativas dicotómicas parcialmente observadas, mientras que la proporción de datos faltantes fue del 30 % bajo el mecanismo MNAR.

En simulaciones adicionales se utilizaron las siguientes estructuras de dependencia para las variables explicativas y las variables indicadoras, respectivamente:

	x	z_1	z_2	z_3
x	1.00	0.50	0.55	0.60
z_1	0.50	1.00	0.65	0.70
z_2	0.55	0.65	1.00	0.75
z_3	0.60	0.70	0.75	1.00

y

	r_1	r_2	r_3
r_1	1.00	0.50	0.70
r_2	0.50	1.00	0.90
r_3	0.70	0.90	1.00

En las Tablas 4.5, 4.6 y 4.7 se muestran los resultados obtenidos para respuestas con distribución Binomial, Poisson y Normal, respectivamente, usando las estructuras de dependencia anteriores, y un escenario que quedó establecido de la siguiente manera. El tamaño del conjunto de datos fue de 1000 observaciones, con una variable explicativa continua completamente observada, y con tres variables explicativas dicotómicas parcialmente observadas, mientras que la proporción de datos faltantes fue del 30 % bajo el mecanismo MNAR. También se resaltan en negritas los valores mínimos de los errores estándares, los sesgos y los errores cuadráticos medios entre los métodos CC, MI, PCML$_{\text{MAR}}$ y PCML$_{\text{MNAR}}$.

Tabla 4.2. Simulación con respuesta Binomial y estructura independiente.

	Método	$\beta_0 = 3$	$\beta_1 = -2.5$	$\beta_{2_1} = 1.5$	$\beta_{2_2} = -1.5$	$\beta_{3_1} = 1$	$\beta_{3_2} = -0.5$
Estimador	DC	3.0246	−2.5694	1.5119	−1.5884	1.1084	−0.5074
	CC	2.9255	−2.5412	1.4967	−1.6282	1.0900	−0.4748
	MI	3.3241	−2.3190	1.3131	−1.5053	1.1357	−0.3541
	PCML$_{\text{MAR}}$	3.3233	−2.3691	1.3226	−1.4995	1.1664	−0.3373
	PCML$_{\text{MNAR}}$	2.6947	−2.3310	1.3646	−1.4840	1.1089	−0.3323
Error estándar	DC	0.3899	0.3860	0.3076	0.2483	0.2776	0.2477
	CC	0.5541	0.5084	0.3989	0.3489	0.3726	0.3410
	MI	0.4391	**0.3843**	0.3228	0.2707	0.2983	0.2692
	PCML$_{\text{MAR}}$	0.4927	0.4443	0.3542	0.3026	0.3252	0.2959
	PCML$_{\text{MNAR}}$	**0.4293**	0.3983	**0.3159**	**0.2673**	**0.2922**	**0.2652**
Sesgo	DC	0.0246	0.0694	0.0119	0.0884	0.1084	0.0074
	CC	**0.0745**	**0.0412**	**0.0033**	0.1282	**0.0900**	**0.0252**
	MI	0.3241	0.1810	0.1869	0.0053	0.1357	0.1459
	PCML$_{\text{MAR}}$	0.3233	0.1309	0.1774	**0.0005**	0.1664	0.1627
	PCML$_{\text{MNAR}}$	0.3053	0.1690	0.1354	0.0160	0.1089	0.1677
Error cuadrático medio	DC	0.1526	0.1538	0.0948	0.0695	0.0888	0.0614
	CC	0.3125	0.2601	0.1591	0.1381	0.1469	0.1169
	MI	0.2978	**0.1805**	0.1391	0.0733	0.1074	**0.0938**
	PCML$_{\text{MAR}}$	0.3473	0.2145	0.1569	0.0916	0.1334	0.1140
	PCML$_{\text{MNAR}}$	**0.2775**	0.1872	**0.1182**	**0.0717**	**0.0972**	0.0984
	Método	$\beta_{4_1} = 0$	$\beta_{4_2} = -1$	$\beta_{5_1} = 0.5$	$\beta_{5_2} = 2$	$\beta_{6_1} = -2$	$\beta_{6_2} = 2.5$
Estimador	DC	−0.0006	−1.0304	0.5207	1.9954	−1.9379	2.6459
	CC	0.0780	−0.9647	0.5712	1.9382	−1.9156	2.5655
	MI	−0.0581	−1.0054	0.5209	1.8274	−1.8875	2.6414
	PCML$_{\text{MAR}}$	−0.0464	−1.0135	0.5541	1.8326	−1.8696	2.6636
	PCML$_{\text{MNAR}}$	0.0088	−0.9830	0.5145	1.9338	−1.8347	2.6506
Error estándar	DC	0.2604	0.2637	0.2426	0.2886	0.2446	0.4233
	CC	0.3635	0.3699	0.3051	0.4250	0.3180	0.6705
	MI	0.2890	0.2933	**0.2444**	0.3500	0.2570	0.5813
	PCML$_{\text{MAR}}$	0.3181	0.3222	0.2631	0.3773	0.2772	0.6065
	PCML$_{\text{MNAR}}$	**0.2786**	**0.2872**	0.2465	**0.3048**	**0.2513**	**0.5154**
Sesgo	DC	0.0006	0.0304	0.0207	0.0046	0.0621	0.1459
	CC	0.0780	0.0353	0.0712	**0.0618**	**0.0844**	**0.0655**
	MI	0.0581	**0.0054**	0.0209	0.1726	0.1125	0.1414
	PCML$_{\text{MAR}}$	0.0464	0.0135	0.0541	0.1674	0.1304	0.1636
	PCML$_{\text{MNAR}}$	**0.0088**	0.0170	**0.0145**	0.0662	0.1653	0.1506
Error cuadrático medio	DC	0.0678	0.0705	0.0593	0.0833	0.0637	0.2004
	CC	0.1382	0.1381	0.0982	0.1844	0.1083	0.4538
	MI	0.0869	0.0860	**0.0602**	0.1523	**0.0787**	0.3579
	PCML$_{\text{MAR}}$	0.1034	0.1040	0.0721	0.1704	0.0938	0.3946
	PCML$_{\text{MNAR}}$	**0.0777**	**0.0828**	0.0609	**0.0973**	0.0905	**0.2883**

Tiempo medio de ejecución (min): 2.7 (MI), 3.3 (PCML$_{\text{MAR}}$), 25.4 (PCML$_{\text{MNAR}}$).

Promedio de iteraciones del algoritmo EM: 10 (PCML$_{\text{MAR}}$), 78 (PCML$_{\text{MNAR}}$).

Tabla 4.3. Simulación con respuesta Poisson y estructura independiente.

	Método	$\beta_0 = 1$	$\beta_1 = -0.5$	$\beta_{2_1} = 0$	$\beta_{3_1} = -1$	$\beta_{4_1} = 0.5$
Estimador	DC	1.0199	−0.5143	0.0056	−1.0034	0.4766
	CC	1.0209	−0.5127	0.0051	−1.0144	0.4775
	MI	1.0670	−0.5116	0.0078	−0.9873	0.4700
	PCML$_{\text{MAR}}$	1.0687	−0.5130	0.0072	−0.9868	0.4690
	PCML$_{\text{MNAR}}$	1.0221	−0.5154	0.0089	−1.0056	0.4817
Error estándar	DC	0.0580	0.0790	0.0455	0.0513	0.0468
	CC	0.0705	0.0947	0.0550	0.0587	0.0552
	MI	**0.0612**	**0.0806**	0.0496	**0.0516**	0.0485
	PCML$_{\text{MAR}}$	0.0627	0.0842	0.0490	0.0535	0.0492
	PCML$_{\text{MNAR}}$	0.0613	0.0817	**0.0476**	0.0530	**0.0482**
Sesgo	DC	0.0199	0.0143	0.0056	0.0034	0.0234
	CC	**0.0209**	0.0127	**0.0051**	0.0144	0.0225
	MI	0.0670	**0.0116**	0.0078	0.0127	0.0300
	PCML$_{\text{MAR}}$	0.0687	0.0130	0.0072	0.0132	0.0310
	PCML$_{\text{MNAR}}$	0.0221	0.0154	0.0089	**0.0056**	**0.0183**
Error cuadrático medio	DC	0.0038	0.0064	0.0021	0.0026	0.0027
	CC	0.0054	0.0091	0.0031	0.0036	0.0036
	MI	0.0082	**0.0066**	0.0025	**0.0028**	0.0033
	PCML$_{\text{MAR}}$	0.0087	0.0073	0.0025	0.0030	0.0034
	PCML$_{\text{MNAR}}$	**0.0042**	0.0069	**0.0023**	0.0028	**0.0027**

Tabla 4.4. Simulación con respuesta Normal y estructura independiente.

	Método	$\beta_0 = 1$	$\beta_1 = -0.5$	$\beta_{2_1} = 0$	$\beta_{3_1} = -1$	$\beta_{4_1} = 0.5$
Estimador	DC	0.9906	−0.4831	0.0093	−0.9939	0.4922
	CC	0.9842	−0.5098	0.0180	−0.9808	0.4993
	MI	1.0449	−0.4893	0.0184	−0.9888	0.4961
	PCML$_{\text{MAR}}$	1.0451	−0.4890	0.0190	−0.9889	0.4954
	PCML$_{\text{MNAR}}$	0.9917	−0.4840	0.0209	−0.9992	0.5012
Error estándar	DC	0.0793	0.1076	0.0631	0.0631	0.0630
	CC	0.0952	0.1254	0.0735	0.0737	0.0738
	MI	**0.0835**	**0.1093**	0.0678	0.0661	0.0664
	PCML$_{\text{MAR}}$	0.0871	0.1157	0.0680	0.0678	0.0681
	PCML$_{\text{MNAR}}$	0.0838	0.1121	**0.0658**	**0.0656**	**0.0657**
Sesgo	DC	0.0094	0.0169	0.0093	0.0061	0.0078
	CC	0.0158	**0.0098**	**0.0180**	0.0192	**0.0007**
	MI	0.0449	0.0107	0.0184	0.0112	0.0039
	PCML$_{\text{MAR}}$	0.0451	0.0110	0.0190	0.0111	0.0046
	PCML$_{\text{MNAR}}$	**0.0083**	0.0160	0.0209	**0.0008**	0.0012
Error cuadrático medio	DC	0.0064	0.0119	0.0041	0.0040	0.0040
	CC	0.0093	0.0158	0.0057	0.0058	0.0054
	MI	0.0090	**0.0121**	0.0049	0.0045	0.0044
	PCML$_{\text{MAR}}$	0.0096	0.0135	0.0050	0.0047	0.0047
	PCML$_{\text{MNAR}}$	**0.0071**	0.0128	**0.0048**	**0.0043**	**0.0043**

Tabla 4.5. Simulación con respuesta Binomial y estructura dependiente.

	Método	$\beta_0 = 1$	$\beta_1 = -0.5$	$\beta_{2_1} = 0$	$\beta_{3_1} = -1$	$\beta_{4_1} = 0.5$
Estimador	DC	0.9964	−0.4999	0.0375	−1.0227	0.4776
	CC	1.0112	−0.5155	0.0339	−1.0240	0.4840
	MI	1.0728	−0.4848	0.0109	−0.9492	0.3883
	PCML$_{\text{MAR}}$	1.0554	−0.4707	0.0135	−1.0047	0.4390
	PCML$_{\text{MNAR}}$	1.0012	−0.4951	0.0274	−1.0242	0.4807
Error estándar	DC	0.1440	0.2805	0.1621	0.1694	0.1757
	CC	0.1705	0.3348	0.1936	0.1982	0.2084
	MI	**0.1476**	0.2843	0.1789	**0.1805**	**0.1885**
	PCML$_{\text{MAR}}$	0.1538	0.3054	0.1802	0.1837	0.1938
	PCML$_{\text{MNAR}}$	0.1496	**0.2830**	**0.1731**	0.1814	0.1886
Sesgo	DC	0.0036	0.0001	0.0375	0.0227	0.0224
	CC	0.0112	0.0155	0.0339	0.0240	**0.0160**
	MI	0.0728	0.0152	**0.0109**	0.0508	0.1117
	PCML$_{\text{MAR}}$	0.0554	0.0293	0.0135	**0.0047**	0.0610
	PCML$_{\text{MNAR}}$	**0.0012**	**0.0049**	0.0274	0.0242	0.0193
Error cuadrático medio	DC	0.0207	0.0787	0.0277	0.0292	0.0314
	CC	0.0292	0.1123	0.0386	0.0398	0.0437
	MI	0.0271	0.0810	0.0321	0.0352	0.0480
	PCML$_{\text{MAR}}$	0.0267	0.0941	0.0327	0.0338	0.0413
	PCML$_{\text{MNAR}}$	**0.0224**	**0.0801**	**0.0307**	**0.0335**	**0.0359**

Tabla 4.6. Simulación con respuesta Poisson y estructura dependiente.

	Método	$\beta_0 = 1$	$\beta_1 = -0.5$	$\beta_{2_1} = 0$	$\beta_{3_1} = -1$	$\beta_{4_1} = 0.5$
Estimador	DC	1.0108	−0.5227	0.0083	−0.9986	0.4909
	CC	1.0129	−0.5318	−0.0051	−0.9891	0.4923
	MI	1.0625	−0.4873	−0.0189	−0.9351	0.4422
	PCML$_{\text{MAR}}$	1.0463	−0.4981	−0.0078	−0.9832	0.4787
	PCML$_{\text{MNAR}}$	1.0137	−0.5224	0.0037	−0.9984	0.4935
Error estándar	DC	0.0426	0.0952	0.0562	0.0603	0.0570
	CC	0.0505	0.1152	0.0688	0.0724	0.0691
	MI	**0.0430**	**0.0988**	0.0646	0.0666	0.0653
	PCML$_{\text{MAR}}$	0.0442	0.1032	0.0633	0.0659	0.0635
	PCML$_{\text{MNAR}}$	0.0441	0.0996	**0.0598**	**0.0633**	**0.0606**
Sesgo	DC	0.0108	0.0227	0.0083	0.0014	0.0091
	CC	**0.0129**	0.0318	0.0051	0.0109	0.0077
	MI	0.0625	0.0127	0.0189	0.0649	0.0578
	PCML$_{\text{MAR}}$	0.0463	**0.0019**	0.0078	0.0168	0.0213
	PCML$_{\text{MNAR}}$	0.0137	0.0224	**0.0037**	**0.0016**	**0.0065**
Error cuadrático medio	DC	0.0019	0.0096	0.0032	0.0036	0.0033
	CC	0.0027	0.0143	0.0048	0.0054	0.0048
	MI	0.0057	**0.0099**	0.0045	0.0086	0.0076
	PCML$_{\text{MAR}}$	0.0041	0.0107	0.0041	0.0046	0.0045
	PCML$_{\text{MNAR}}$	**0.0021**	0.0104	**0.0036**	**0.0040**	**0.0037**

Tabla 4.7. Simulación con respuesta Normal y estructura dependiente.

	Método	$\beta_0 = 1$	$\beta_1 = -0.5$	$\beta_{2_1} = 0$	$\beta_{3_1} = -1$	$\beta_{4_1} = 0.5$
Estimador	DC	1.0025	−0.4969	−0.0233	−0.9966	0.4945
	CC	0.9937	−0.4765	−0.0241	−1.0037	0.4873
	MI	1.0633	−0.4799	−0.0561	−0.9170	0.4146
	PCML_{MAR}	1.0462	−0.4738	−0.0427	−0.9754	0.4608
	$\text{PCML}_{\text{MNAR}}$	1.0059	−0.4978	−0.0272	−1.0006	0.4980
Error estándar	DC	0.0653	0.1301	0.0746	0.0806	0.0830
	CC	0.0770	0.1552	0.0893	0.0945	0.0984
	MI	**0.0669**	**0.1348**	0.0842	0.0883	0.0930
	PCML_{MAR}	0.0689	0.1420	0.0836	0.0884	0.0927
	$\text{PCML}_{\text{MNAR}}$	0.0680	0.1374	**0.0798**	**0.0860**	**0.0889**
Sesgo	DC	0.0025	0.0031	0.0233	0.0034	0.0055
	CC	0.0063	0.0235	**0.0241**	0.0037	0.0127
	MI	0.0633	0.0201	0.0561	0.0830	0.0854
	PCML_{MAR}	0.0462	0.0262	0.0427	0.0246	0.0392
	$\text{PCML}_{\text{MNAR}}$	**0.0059**	**0.0022**	0.0272	**0.0006**	**0.0020**
Error cuadrático medio	DC	0.0043	0.0169	0.0061	0.0065	0.0069
	CC	0.0060	0.0246	0.0086	0.0089	0.0098
	MI	0.0085	**0.0186**	0.0102	0.0147	0.0159
	PCML_{MAR}	0.0069	0.0209	0.0088	0.0084	0.0101
	$\text{PCML}_{\text{MNAR}}$	**0.0047**	0.0189	**0.0071**	**0.0074**	**0.0079**

De acuerdo con los resultados obtenidos y mostrados en las tablas anteriores, se puede observar que, por obvias razones, con DC se obtienen los menores errores estándares y errores cuadráticos medios, ya que DC es aplicado al conjunto de datos completo. Por el contrario, con el método CC se generan los mayores errores estándares y errores cuadráticos medios, como consecuencia de la eliminación completa de una gran cantidad de información útil para el análisis. Sin embargo, con CC también se obtienen muchos de los estimadores menos sesgados, y aunque esto a primera vista pudiera sorprender, no debería ser así, ya que como bien se mencionó en la subsección 1.2.1, Little (1992) establece que CC puede producir estimadores insesgados de coeficientes de regresión bajo cualquier mecanismo de datos faltantes, siempre y cuando la pérdida de datos esté en función de las variables explicativas y no de la variable respuesta, como fue el caso de las simulaciones hechas.

Los estimadores de los coeficientes de regresión que fueron generados con los métodos MI y PCML_{MAR} con estructura independiente son similares, con lo cual queda validado el correcto desempeño de PCML_{MAR} propuesto en este trabajo, tomando en cuenta que el método MI ya está bien establecido en cuanto a la estimación de parámetros para el caso de pérdida de datos tipo MAR. Por otro lado, con la inclusión del mecanismo de pérdida de datos mediante el método

PCML$_{\text{MNAR}}$ con estructura dependiente, se obtienen estimadores ligeramente diferentes. También se observa que los métodos MI y PCML$_{\text{MNAR}}$ tienden a producir los menores errores estándares. Nótese el hecho destacable que con el método PCML$_{\text{MNAR}}$ con estructura dependiente se obtienen los menores sesgos y errores cuadráticos medios en su gran mayoría.

4.2. Aplicación

El conjunto de datos reales utilizado como ejemplo para ilustrar la viabilidad práctica de PCML, proviene del estudio clínico E1684 sobre el melanoma en su fase III realizado por el Grupo Oncológico Cooperativo del Este (ECOG, por sus siglas en inglés). Los resultados de este estudio usando el método de casos completos fueron publicados por Kirkwood et al. (1996). En el estudio se involucraron a 285 pacientes que después de su cirugía fueron asignados aleatoriamente a uno de dos tratamientos: observación o alta dosis de interferón. El interferón alfa-2b es un tratamiento postoperatorio de quimioterapia. Los resultados de este estudio sugirieron que el interferón tiene un efecto significativo en la supervivencia libre de recaída, que es el período de tiempo entre el inicio del tratamiento y la recaída por el cáncer. Esto lleva a la Administración de Alimentos y Medicamentos de los Estados Unidos (FDA, por sus siglas en inglés) a aprobar este tratamiento como una terapia coadyuvante estándar para pacientes con melanoma de alto riesgo.

La variable respuesta dicotómica de interés es la recaída que consta de dos niveles: 1 si el paciente ha recaído dentro de los primeros 0.55 años después del inicio del tratamiento (69 %) y 2 si no lo ha hecho (31 %). Todos los valores censurados en el conjunto de datos original excedieron de 0.55 años. Las variables explicativas consideradas en el análisis se describen a continuación, denotando con mayúsculas a las variables discretas tipo factor. edad: variable continua sin datos faltantes de la edad del paciente en años, con un valor mínimo de 17.04, un valor máximo de 78.73 y una media de 47.12 años. A esta variable se le aplicó una transformación de escala para estandarizarla con una media de cero antes del análisis. GENERO: variable dicotómica sin datos faltantes del género del paciente que consta de dos niveles: 1 si es masculino (60 %) y 2 si es femenino (40 %). TRATAMIENTO: variable dicotómica sin datos faltantes del tratamiento asignado al paciente que se compone de dos niveles: 1 si es observación (49 %) y 2 si es interferón (51 %). TAMAÑO: variable continua del tamaño del tumor primario en cm^2 con 19 % de datos faltantes, la cual fue discretizada en su mediana con dos niveles: 1 si es menor que la mediana (40 %) y 2 en otro caso (41 %). TIPO: variable dicotómica del tipo de tumor primario con 12 % de datos faltantes y

dos niveles: 1 si es de propagación superficial (54 %) y 2 de otros tipos (34 %). BRESLOW: variable continua del *grosor de Breslow* o profundidad del tumor en mm con 10 % de datos faltantes, la cual fue categorizada en dos niveles: 1 si es mayor o igual a 2.5 mm (48 %) y 2 en otro caso (42 %). El conjunto de datos tiene en total una proporción del 29 % de información faltante en las tres últimas variables explicativas antes descritas.

La Tabla 4.8 muestra los estimadores de los coeficientes de regresión junto con sus respectivos errores estándares, valores–p y niveles de significancia (Sig) dados por los métodos CC, LX, MI y PCML$_{\text{MAR}}$. Aquí LX se refiere al paquete de software estadístico *LogXact* version 10 propiedad de la compañía de biotecnología Cytel Inc. (`www.cytel.com`). Esta compañía ofrece servicios de investigación clínica, consultoría estratégica y soluciones integrales de software para el diseño y análisis de estudios clínicos. En el software LX las variables explicativas parcialmente observadas deben ser categóricas y se asume que son de tipo MAR, por lo que no incluye la capacidad de modelar el mecanismo de pérdida de datos para aquellos casos de variables con valores faltantes MNAR. Por otro lado, se basa en la Ecuación (1.3) propuesta por Lipsitz and Ibrahim (1996) para el modelado de la distribución conjunta de las variables explicativas con datos faltantes. Con respecto a la distribución de la variable respuesta completamente observada, esta puede ser Binomial, Poisson o Normal. Para las variables explicativas completamente observadas no hay restricción en cuanto a su tipo de distribución. La versión 10 del software soporta un máximo de 50 variables explicativas, de las cuales hasta 10 pueden ser dicotómicas con valores faltantes. Para la estimación de los parámetros utiliza el algoritmo EM vía ponderaciones propuesto por Ibrahim (1990), mientras que para la obtención de los errores estándares usa el método de Louis (1982).

De acuerdo con los resultados mostrados en la Tabla 4.8, se puede observar que el software LX presenta cierta deficiencia en la estimación del Intercepto ya que lo reporta como no significativo, mientras que con los otros tres métodos sí lo es. Por otra parte, con los métodos LX, MI y PCML$_{\text{MAR}}$ se confirma que el tratamiento con alta dosis de interferón si es efectivo en el decremento de la proporción de pacientes que recaen. Sin embargo, con el método CC dicho tratamiento aparece como no significativo, por lo que se podría concluir de manera errónea que la terapia con interferón no es efectiva en incrementar la supervivencia de los pacientes con melanoma. Esto muestra que la pérdida de información relevante ocasionada con el método CC puede generar sesgo e ineficiencia en la estimación.

Tabla 4.8. Resultados del ejemplo ilustrativo de aplicación.

Variable	Método	Estimador	Error estándar	valor–p	Sig
Intercepto	CC	1.2541	0.4145	0.0025	**
	LX	1.2858	0.7904	0.1038	
	MI	1.0030	0.3439	0.0039	**
	PCML_{MAR}	0.9998	0.3728	0.0073	**
edad	CC	0.0080	0.0120	0.5081	
	LX	0.0094	0.0101	0.3509	
	MI	0.0091	0.0102	0.3714	
	PCML_{MAR}	0.0091	0.0111	0.4133	
GENERO_2	CC	−0.3015	0.3233	0.3510	
	LX	−0.1161	0.2672	0.6639	
	MI	−0.1230	0.2685	0.6472	
	PCML_{MAR}	−0.1195	0.2936	0.6841	
TRATAMIENTO_2	CC	−0.5545	0.3225	0.0856	°
	LX	−0.6009	0.2651	0.0234	*
	MI	−0.6037	0.2666	0.0244	*
	PCML_{MAR}	−0.6045	0.2908	0.0376	*
TAMAÑO_2	CC	0.2222	0.3321	0.5035	
	LX	0.2800	0.3049	0.3583	
	MI	0.2886	0.2942	0.3277	
	PCML_{MAR}	0.2885	0.3027	0.3405	
TIPO_2	CC	−0.0278	0.3379	0.9344	
	LX	−0.0246	0.3024	0.9351	
	MI	−0.0279	0.3044	0.9272	
	PCML_{MAR}	−0.0200	0.3111	0.9489	
BRESLOW_2	CC	−0.1355	0.3522	0.7004	
	LX	0.0393	0.3078	0.8983	
	MI	0.0583	0.3006	0.8464	
	PCML_{MAR}	0.0603	0.3224	0.8516	

valor–$p \leq 0.01$ (**), valor–$p \leq 0.05$ (*), valor–$p \leq 0.1$ (°).

Un componente clave en el estudio de estos datos es llevar a cabo un análisis de sensibilidad para evaluar la robustez en contra de las desviaciones del modelo MAR mediante la comparación del sesgo introducido cuando se emplean mecanismos MNAR. En esta aplicación, dichos mecanismos MNAR se especificaron de la siguiente manera:

$$f_{\mathbf{r}}(\mathbf{r}|\mathbf{w};\boldsymbol{\nu}) = f(\mathbf{r}|\mathbf{w};\boldsymbol{\nu}),$$

donde $\mathbf{w} = (\mathbf{x}, \mathbf{z}, y)$, $\mathbf{x} = (\text{edad}, \text{GENERO}, \text{TRATAMIENTO})$, y $\mathbf{z}$ define el mecanismo de pérdida

de acuerdo con los siguientes modelos: 1) $\mathbf{z} = (\mathsf{TAMAÑO})$, 2) $\mathbf{z} = (\mathsf{TIPO})$, 3) $\mathbf{z} = (\mathsf{BRESLOW})$, 4) $\mathbf{z} = (\mathsf{TAMAÑO} + \mathsf{TIPO})$, 5) $\mathbf{z} = (\mathsf{TAMAÑO} + \mathsf{BRESLOW})$, 6) $\mathbf{z} = (\mathsf{TIPO} + \mathsf{BRESLOW})$, 7) $\mathbf{z} = (\mathsf{TAMAÑO} + \mathsf{TIPO} + \mathsf{BRESLOW})$. La Tabla 4.9 muestra los estimadores de los coeficientes de regresión junto con sus correspondientes errores estándares y valores–p para los siete modelos MNAR anteriores. Tanto los estimadores como los errores estándares parecen ser suficientemente robustos con respecto al cambio del mecanismo de pérdida, lo cual podría sugerir que los datos son tipo MAR. Sin embargo, existen ciertas discrepancias en la variable TIPO para los modelos 2 y 4, indicando la presencia de un mecanismo de pérdida MNAR relativamente débil.

Tabla 4.9. Resultados del análisis de sensibilidad para los siete modelos MNAR.

Variable	**Modelo**	**Estimador**	**Error estándar**	**valor–p**
Intercepto	1	1.0183	0.3497	0.0036
	2	0.9487	0.3674	0.0098
	3	1.0158	0.3727	0.0064
	4	0.9453	0.3583	0.0083
	5	1.0284	0.3504	0.0033
	6	1.0487	0.3443	0.0023
	7	1.0479	0.3231	0.0012
edad	1	0.0092	0.0107	0.3908
	2	0.0092	0.0108	0.3953
	3	0.0092	0.0111	0.4084
	4	0.0090	0.0107	0.4004
	5	0.0092	0.0107	0.3876
	6	0.0092	0.0108	0.3916
	7	0.0092	0.0103	0.3702
GENERO_2	1	−0.1150	0.2793	0.6806
	2	−0.1225	0.2870	0.6694
	3	−0.1222	0.2930	0.6767
	4	−0.1264	0.2819	0.6539
	5	−0.1169	0.2787	0.6750
	6	−0.1255	0.2853	0.6601
	7	−0.1175	0.2708	0.6644
$\mathsf{TRATAMIENTO}_2$	1	−0.6001	0.2761	0.0298
	2	−0.6059	0.2855	0.0338
	3	−0.6073	0.2903	0.0365
	4	−0.6112	0.2807	0.0295
	5	−0.6024	0.2753	0.0287
	6	−0.6103	0.2844	0.0319
	7	−0.6018	0.2692	0.0254

Continúa en la página siguiente.

Tabla 4.9. Continuación de la página anterior.

Variable	Modelo	Estimador	Error estándar	valor–p
$\mathsf{TAMAÑO}_2$	1	0.3170	0.2897	0.2738
	2	0.2855	0.2951	0.3333
	3	0.2902	0.3019	0.3365
	4	0.3510	0.2918	0.2291
	5	0.3209	0.2887	0.2664
	6	0.2938	0.2939	0.3176
	7	0.3209	0.2813	0.2540
TIPO_2	1	−0.0248	0.2990	0.9339
	2	**0.0789**	0.3015	0.7935
	3	−0.0198	0.3109	0.9491
	4	**0.0825**	0.2969	0.7812
	5	−0.0262	0.2989	0.9301
	6	−0.0649	0.2988	0.8282
	7	−0.0727	0.2881	0.8009
$\mathsf{BRESLOW}_2$	1	0.0497	0.3058	0.8708
	2	0.0811	0.3166	0.7979
	3	0.0347	0.3219	0.9141
	4	0.0611	0.3118	0.8446
	5	0.0321	0.3055	0.9164
	6	0.0013	0.3023	0.9966
	7	0.0231	0.2883	0.9362

La Figura 4.1 muestra las gráficas de los residuales de cuantiles aleatorizados correspondientes al modelo MAR anteriormente propuesto para ajustar los datos del melanoma. Las gráficas (a) y (b) sugieren que el supuesto de normalidad estándar de los residuales se cumple, y adicionalmente muestran la presencia de un valor atípico. Por otro lado, las gráficas (c) y (d) plantean que el supuesto de independencia entre los residuales es razonable, ya que las correlaciones no siguen un patrón definido ni son significativas (se mantienen dentro del intervalo de significancia al 95 % representado con las líneas punteadas). Entonces se puede concluir que el modelo MAR asumido ajusta razonablemente bien a los datos.

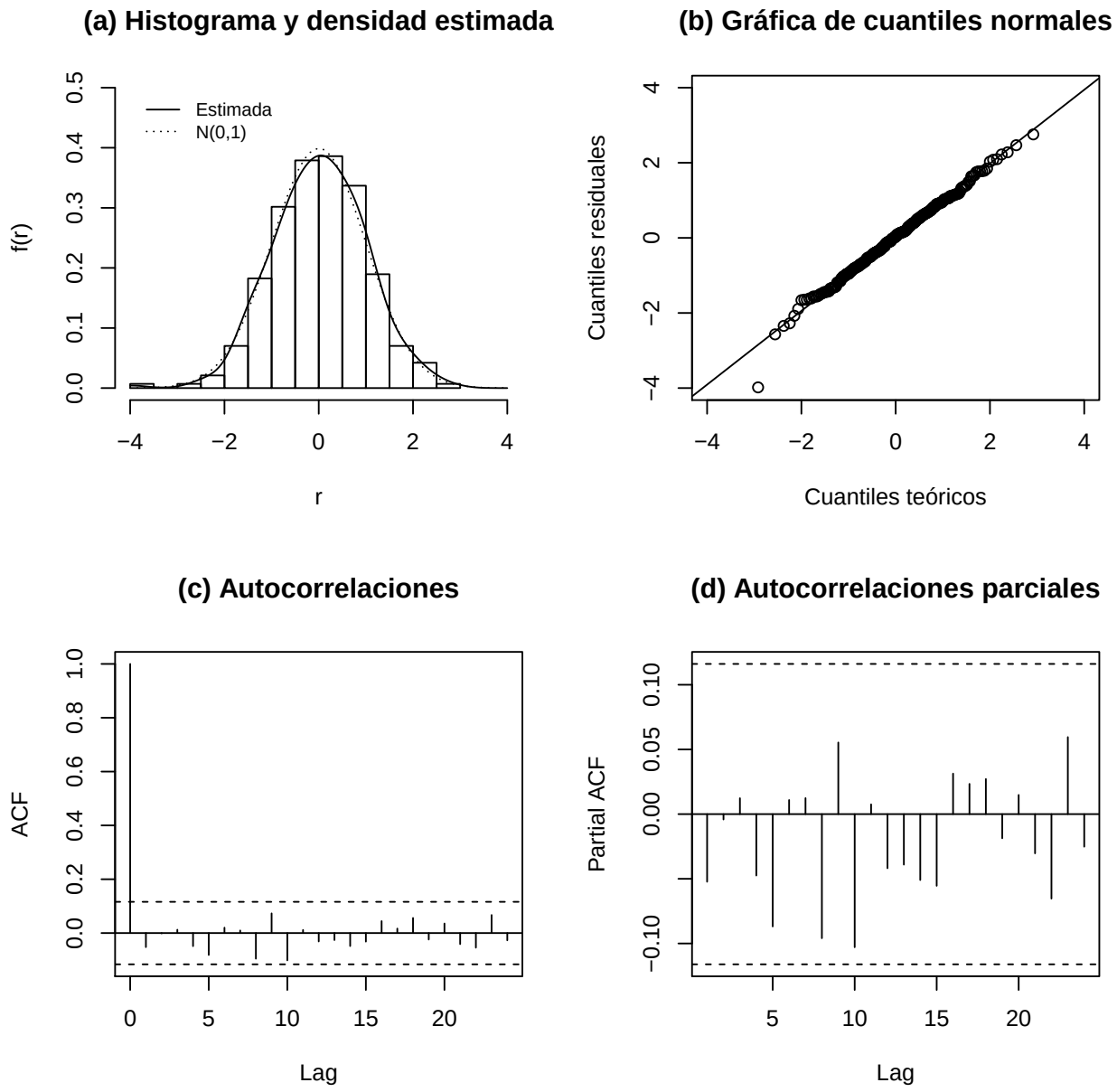

Figura 4.1. Gráficas de los residuales de cuantiles aleatorizados correspondientes al modelo MAR.

Capítulo 5

Conclusiones y perspectivas

En este trabajo se ha presentado la metodología PCML de modelado de distribuciones conjuntas basada en cópulas para el análisis de modelos lineales generalizados con la presencia de variables categóricas parcialmente observadas. PCML consistió en especificar distribuciones conjuntas tanto para las variables explicativas con datos faltantes como para sus respectivas variables indicadoras usando la técnica reciente de construcción con cópulas pareadas, y enlazando las distribuciones marginales a un componente lineal en términos de variables discretas o continuas completamente observadas. Se implementó el algoritmo EM vía ponderaciones para estimar simultáneamente los coeficientes de regresión en el modelo lineal generalizado y los parámetros característicos de las distribuciones conjuntas. Un resultado derivado del mismo procedimiento fueron las ponderaciones estimadas, lo cual permitió la formulación de un modelo de evaluación usando residuales de cuantiles aleatorizados. La inclusión de mecanismos de pérdida de datos en el modelo también permitió el desarrollo de una estrategia para llevar acabo análisis de sensibilidad que ayudaran a distinguir entre modelos MAR y MNAR.

Con los resultados obtenidos mediante simulaciones y por medio de la aplicación a datos reales, se puede constatar que PCML es una alternativa competitiva, robusta, flexible y con buen desempeño bajo diversos escenarios de conjuntos de datos. En comparación con otros métodos establecidos como imputación múltiple, se destaca su capacidad de modelado de mecanismos de datos faltantes cuando la pérdida de éstos es no aleatoria o MNAR. La técnica de construcción con cópulas pareadas empleada en este trabajo tiene la ventaja de una gran flexibilidad en la especificación de distribuciones conjuntas. Además, es capaz de detectar una amplia variedad de estructuras de dependencia más allá de asociaciones lineales y monótonas,

y es computacionalmente factible dado que sólo requiere $m(m-1)/2$ cópulas bivariadas para una distribución m-dimensional a ser construida. Las simulaciones mostraron que los tiempos promedios de ejecución del algoritmo EM hasta lograr la convergencia bajo el modelo MAR fueron similares a los de imputación múltiple. Sin embargo, bajo el mecanismo MNAR, los tiempos medios de ejecución fueron alrededor de ocho veces mayores que bajo MAR. La mejora computacional en este sentido es una oportunidad de investigación a futuro.

Se deja como tema de investigación posterior, la posibilidad de especificar la función de verosimilitud en base a integrales multivariadas, ya sea para variables explicativas sólo continuas o discretas y continuas combinadas con pérdida de datos MAR o MNAR. Además, proponer algún método numérico para evaluar dichas integrales. Para este caso, la especificación correcta de la función log-verosimilitud esperada involucra integrales multivariadas de la forma

$$E[l_i(\boldsymbol{\theta})|\mathbf{x}_i, \mathbf{z}_i, y_i, \mathbf{r}_i] = \int f(\mathbf{z}_{i,q}|\mathbf{x}_i, y_i, \mathbf{r}_i; \boldsymbol{\theta}) \; [l_{i,q}(\boldsymbol{\theta})|\mathbf{x}_i, \mathbf{z}_{i,q}, y_i, \mathbf{r}_i] \; \mathrm{d}\mathbf{z}_{i,q}.$$

Ibrahim and Weisberg (1992) propusieron aproximar tales integrales mediante métodos de cuadratura Gaussiana, donde las integrales son discretizadas y en consecuencia el modelo (2.7) es restaurado, permitiendo así la aplicación del algoritmo EM mediante ponderaciones. Una segunda opción es la presentada en Ibrahim et al. (1999a) donde utilizan una versión Monte Carlo del algoritmo EM. Asumiendo cierta distribución paramétrica de las variables explicativas, un muestreador de Gibbs puede ser usado para generar una serie de N muestras de dicha distribución, que posteriormente serán utilizadas en el modelo (2.7) con ponderaciones iguales a $1/N$. Esta técnica es computacionalmente muy demandante, dado que dentro de cada iteración del algoritmo EM una gran cantidad de muestras deben ser tomadas de la distribución (e.g. 2000), y el número de observaciones en el conjunto de datos aumentado puede llegar a ser bastante elevado si N es grande.

Apéndice A

Código en lenguaje R

```
#-------------------------------------------------------------------------------------------
#########################################################################
#                                                                       #
# MODELADO DE DISTRIBUCIONES CONJUNTAS PARA MODELOS                     #
# LINEALES GENERALIZADOS CON DATOS FALTANTES                            #
#                                                                       #
# CODIGO EN LENGUAJE R PARA REALIZAR ANALISIS DE REGRESION LOGISTICA    #
# AL CONJUNTO DE DATOS REALES OBTENIDO DEL ESTUDIO CLINICO E1684        #
# MEDIANTE LOS CUATRO METODOS SIGUIENTES: CC, MI, PCML_MAR Y PCML_MNAR  #
#                                                                       #
# LUIS CARLOS PEREZ RUIZ (2018)                                         #
#                                                                       #
#########################################################################
#-------------------------------------------------------------------------------------------
################################################
#                                              #
# MACRO PARA COMPILAR LA COPULA DE FRANK EN C++ #
#                                              #
################################################

library(mice)
library(Rcpp); sourceCpp("frankC.cpp")
#-------------------------------------------------------------------------------------------
#########################################################################
#                                                                       #
# MACRO PARA AÑADIR AL CONJUNTO DE DATOS LAS VARIABLES INDICADORAS r    #
#                                                                       #
#########################################################################

E1684 <- read.table("E1684.txt",header=T)
G <- data.frame(x1 = E1684$age,
             x2 = factor(E1684$sex),
             x3 = factor(E1684$trt),
             z1 = factor(E1684$size,labels=c(1,2)),
             z2 = factor(E1684$type,labels=c(1,2)),
             z3 = factor(E1684$bres,labels=c(1,2)),
             y  = E1684$failcens)
G <- transform(G,x1=scale(G$x1))
n <- nrow(G)
```

```
r1 <- rep(1,n); r1[which(is.na(G$z1)==T)] <- 0
r2 <- rep(1,n); r2[which(is.na(G$z2)==T)] <- 0
r3 <- rep(1,n); r3[which(is.na(G$z3)==T)] <- 0
G <- data.frame(G,r1,r2,r3)
#-------------------------------------------------------------------------------------------
#################################################################
#                                                               #
# MACRO DE REGRESION CON EL METODO DE IMPUTACION MULTIPLE (MI) #
#                                                               #
#################################################################

imp <- mice(G,m=50,meth=c("","","","logreg","logreg","logreg","","","","",""),printFlag=F)
fit <- with(data=imp,exp=glm(y~x1+x2+x3+z1+z2+z3,family=binomial(link=logit)))
pol <- pool(fit)
MI.sum <- summary(pol)
#-------------------------------------------------------------------------------------------
#########################################################
#                                                       #
# MACRO PARA COMPLETAR Y PONDERAR AL CONJUNTO DE DATOS #
#                                                       #
#########################################################

A <- data.frame(x1 = G$x1,
                x2 = as.numeric(levels(G$x2))[G$x2],
                x3 = as.numeric(levels(G$x3))[G$x3],
                z1 = as.numeric(levels(G$z1))[G$z1],
                z2 = as.numeric(levels(G$z2))[G$z2],
                z3 = as.numeric(levels(G$z3))[G$z3],
                y  = G$y,
                r1 = G$r1, r2 = G$r2, r3 = G$r3)

nz1 <- nlevels(G$z1); Lz1 <- as.numeric(levels(G$z1))
nz2 <- nlevels(G$z2); Lz2 <- as.numeric(levels(G$z2))
nz3 <- nlevels(G$z3); Lz3 <- as.numeric(levels(G$z3))

num <- numeric()
X1 <- num; X2 <- num; X3 <- num; Z1 <- num; Z2 <- num; Z3 <- num
R1 <- num; R2 <- num; R3 <- num; YY <- num; WW <- num
TFF <- num; FTF <- num; FFT <- num; TTF <- num; TFT <- num; FTT <- num; TTT <- num
tff <- 0;   ftf <- 0;   fft <- 0;   ttf <- 0;   tft <- 0;   ftt <- 0;   ttt <- 0
c <- 0

for(i in 1:n){
      if(is.na(A$z1[i])==F & is.na(A$z2[i])==F & is.na(A$z3[i])==F){
            c <- c+1
            X1[c] <- A$x1[i]; X2[c] <- A$x2[i]; X3[c] <- A$x3[i]
            Z1[c] <- A$z1[i]; Z2[c] <- A$z2[i]; Z3[c] <- A$z3[i]
            R1[c] <- A$r1[i]; R2[c] <- A$r2[i]; R3[c] <- A$r3[i]
            YY[c] <- A$y[i];  WW[c] <- 1
      }
      else if(is.na(A$z1[i])==T & is.na(A$z2[i])==F & is.na(A$z3[i])==F){
            for(j in 1:nz1){
                  c <- c+1; tff <- tff+1; TFF[tff] <- c
                  X1[c] <- A$x1[i]; X2[c] <- A$x2[i]; X3[c] <- A$x3[i]
                  Z1[c] <- Lz1[j];  Z2[c] <- A$z2[i]; Z3[c] <- A$z3[i]
                  R1[c] <- A$r1[i]; R2[c] <- A$r2[i]; R3[c] <- A$r3[i]
                  YY[c] <- A$y[i];  WW[c] <- 1/nz1
            }
      }
      else if(is.na(A$z1[i])==F & is.na(A$z2[i])==T & is.na(A$z3[i])==F){
            for(j in 1:nz2){
                  c <- c+1; ftf <- ftf+1; FTF[ftf] <- c
                  X1[c] <- A$x1[i]; X2[c] <- A$x2[i]; X3[c] <- A$x3[i]
                  Z1[c] <- A$z1[i]; Z2[c] <- Lz2[j];  Z3[c] <- A$z3[i]
                  R1[c] <- A$r1[i]; R2[c] <- A$r2[i]; R3[c] <- A$r3[i]
                  YY[c] <- A$y[i];  WW[c] <- 1/nz2
            }
      }
```

```
      else if(is.na(A$z1[i])==F & is.na(A$z2[i])==F & is.na(A$z3[i])==T){
            for(j in 1:nz3){
                  c <- c+1; fft <- fft+1; FFT[fft] <- c
                  X1[c] <- A$x1[i]; X2[c] <- A$x2[i]; X3[c] <- A$x3[i]
                  Z1[c] <- A$z1[i]; Z2[c] <- A$z2[i]; Z3[c] <- Lz3[j]
                  R1[c] <- A$r1[i]; R2[c] <- A$r2[i]; R3[c] <- A$r3[i]
                  YY[c] <- A$y[i];  WW[c] <- 1/nz3
            }
      }
      else if(is.na(A$z1[i])==T & is.na(A$z2[i])==T & is.na(A$z3[i])==F){
            for(j in 1:nz1){
                  for(k in 1:nz2){
                        c <- c+1; ttf <- ttf+1; TTF[ttf] <- c
                        X1[c] <- A$x1[i]; X2[c] <- A$x2[i]; X3[c] <- A$x3[i]
                        Z1[c] <- Lz1[j];  Z2[c] <- Lz2[k];  Z3[c] <- A$z3[i]
                        R1[c] <- A$r1[i]; R2[c] <- A$r2[i]; R3[c] <- A$r3[i]
                        YY[c] <- A$y[i];  WW[c] <- 1/(nz1*nz2)
                  }
            }
      }
      else if(is.na(A$z1[i])==T & is.na(A$z2[i])==F & is.na(A$z3[i])==T){
            for(j in 1:nz1){
                  for(k in 1:nz3){
                        c <- c+1; tft <- tft+1; TFT[tft] <- c
                        X1[c] <- A$x1[i]; X2[c] <- A$x2[i]; X3[c] <- A$x3[i]
                        Z1[c] <- Lz1[j];  Z2[c] <- A$z2[i]; Z3[c] <- Lz3[k]
                        R1[c] <- A$r1[i]; R2[c] <- A$r2[i]; R3[c] <- A$r3[i]
                        YY[c] <- A$y[i];  WW[c] <- 1/(nz1*nz3)
                  }
            }
      }
      else if(is.na(A$z1[i])==F & is.na(A$z2[i])==T & is.na(A$z3[i])==T){
            for(j in 1:nz2){
                  for(k in 1:nz3){
                        c <- c+1; ftt <- ftt+1; FTT[ftt] <- c
                        X1[c] <- A$x1[i]; X2[c] <- A$x2[i]; X3[c] <- A$x3[i]
                        Z1[c] <- A$z1[i]; Z2[c] <- Lz2[j];  Z3[c] <- Lz3[k]
                        R1[c] <- A$r1[i]; R2[c] <- A$r2[i]; R3[c] <- A$r3[i]
                        YY[c] <- A$y[i];  WW[c] <- 1/(nz2*nz3)
                  }
            }
      }
      else{
            for(j in 1:nz1){
                  for(k in 1:nz2){
                        for(l in 1:nz3){
                              c <- c+1; ttt <- ttt+1; TTT[ttt] <- c
                              X1[c] <- A$x1[i]; X2[c] <- A$x2[i]; X3[c] <- A$x3[i]
                              Z1[c] <- Lz1[j];  Z2[c] <- Lz2[k];  Z3[c] <- Lz3[l]
                              R1[c] <- A$r1[i]; R2[c] <- A$r2[i]; R3[c] <- A$r3[i]
                              YY[c] <- A$y[i];  WW[c] <- 1/(nz1*nz2*nz3)
                        }
                  }
            }
      }
}

M <- matrix(c(X1,X2,X3,Z1,Z2,Z3,YY,R1,R2,R3,WW),c)
D <- data.frame(x1 = M[,1], x2 = factor(M[,2]), x3 = factor(M[,3]),
                z1 = factor(M[,4]), z2 = factor(M[,5]), z3 = factor(M[,6]), y  = M[,7],
                r1 = factor(M[,8]), r2 = factor(M[,9]), r3 = factor(M[,10]), w  = M[,11])
N <- nrow(D)

X <- model.matrix(~x1+x2+x3,data=D); dX <- dim(X)[2]
Z <- model.matrix(~x1+x2+x3+z1+z2+z3,data=D)
#.....................................................
# PARA EL CASO MAR SE DEBERA USAR LA SIGUIENTE MATRIZ DE DISEÑO:

Y <- model.matrix(~x1+x2+x3,data=D); dY <- dim(Y)[2]
```

```
# PARA EL CASO MNAR SE DEBERA USAR LA SIGUIENTE MATRIZ DE DISEÑO:

Y <- model.matrix(~x1+x2+x3+z1+z2+z3+y,data=D); dY <- dim(Y)[2]
#.....................................................................
In <- rep(0,N)
Iw <- which(D$w!=1)
Iw2 <- matrix(which(D$w==1/2),2)
Iw4 <- matrix(which(D$w==1/4),4)
Iw8 <- matrix(which(D$w==1/8),8)

Dz1 <- as.numeric(levels(D$z1))[D$z1]; Dr1 <- as.numeric(levels(D$r1))[D$r1]
Dz2 <- as.numeric(levels(D$z2))[D$z2]; Dr2 <- as.numeric(levels(D$r2))[D$r2]
Dz3 <- as.numeric(levels(D$z3))[D$z3]; Dr3 <- as.numeric(levels(D$r3))[D$r3]
#-------------------------------------------------------------------------------------------
############################################################
#                                                          #
# MACRO DE REGRESION CON EL METODO DE CASOS COMPLETOS (CC) #
# Y DE INICIALIZACION DE PARAMETROS                        #
#                                                          #
############################################################

CC <- glm(y~x1+x2+x3+z1+z2+z3,family=binomial(link=logit),data=G)
B0 <- CC$coef[1]; B1 <- CC$coef[2]; B2 <- CC$coef[3]; B3 <- CC$coef[4]
B4 <- CC$coef[5]; B5 <- CC$coef[6]; B6 <- CC$coef[7]
D0 <- 1; D1 <- 1; D2 <- 1; D3 <- 1; D4 <- 1; D5 <- 1; D6 <- 1
ro12 <- 0.5; ro23 <- 0.5; ro13.2 <- 0.5
Ro12 <- 0.5; Ro23 <- 0.5; Ro13.2 <- 0.5
a11 <- rep(0.5,dX); a21 <- rep(0.5,dX); a31 <- rep(0.5,dX)
n1  <- rep(0.5,dY); n2  <- rep(0.5,dY); n3  <- rep(0.5,dY)

fz <- rep(1/8,N); fy <- rep(1/8,N); fr <- rep(1/8,N)
tol <- 1e-9
#-------------------------------------------------------------------------------------------
#########################################################
#                                                       #
# MACRO DE LAS FUNCIONES LOG-VEROSIMILITUD A MAXIMIZAR  #
#                                                       #
#########################################################

logV.z <- function(p){
ro12 <- p[1]; ro23 <- p[2]; ro13.2 <- p[3]
a11 <- p[(4+0*dX):(3+1*dX)]
a21 <- p[(4+1*dX):(3+2*dX)]
a31 <- p[(4+2*dX):(3+3*dX)]

eXa11 <- exp(X%*%a11); q11 <- eXa11/(1+eXa11)
eXa21 <- exp(X%*%a21); q21 <- eXa21/(1+eXa21)
eXa31 <- exp(X%*%a31); q31 <- eXa31/(1+eXa31)

F1z <- function(z){
      I10 <- which(z <  Lz1[1])
      I11 <- which(z >= Lz1[1] & z < Lz1[2])
      I12 <- which(z >= Lz1[2])

      F[I10] <- 0
      F[I11] <- q11[I11]
      F[I12] <- 1
      F
}
F2z <- function(z){
      I20 <- which(z <  Lz2[1])
      I21 <- which(z >= Lz2[1] & z < Lz2[2])
      I22 <- which(z >= Lz2[2])

      F[I20] <- 0
      F[I21] <- q21[I21]
      F[I22] <- 1
      F
}
```

```
F3z <- function(z){
      I30 <- which(z <  Lz3[1])
      I31 <- which(z >= Lz3[1] & z < Lz3[2])
      I32 <- which(z >= Lz3[2])

      F[I30] <- 0
      F[I31] <- q31[I31]
      F[I32] <- 1
      F
}
F1p <- F1z(Dz1); F1m <- F1z(Dz1-1)
F2p <- F2z(Dz2); F2m <- F2z(Dz2-1); f2 <- F2p-F2m
F3p <- F3z(Dz3); F3m <- F3z(Dz3-1)

C12pp <- frankC(F1p,F2p,ro12); C23pp <- frankC(F2p,F3p,ro23)
C12pm <- frankC(F1p,F2m,ro12); C23pm <- frankC(F2p,F3m,ro23)
C12mp <- frankC(F1m,F2p,ro12); C23mp <- frankC(F2m,F3p,ro23)
C12mm <- frankC(F1m,F2m,ro12); C23mm <- frankC(F2m,F3m,ro23)

F1.2p <- (C12pp-C12pm)/f2; F1.2m <- (C12mp-C12mm)/f2
F3.2p <- (C23pp-C23mp)/f2; F3.2m <- (C23pm-C23mm)/f2; f3.2 <- F3.2p-F3.2m

C13.2pp <- frankC(F1.2p,F3.2p,ro13.2)
C13.2pm <- frankC(F1.2p,F3.2m,ro13.2)
C13.2mp <- frankC(F1.2m,F3.2p,ro13.2)
C13.2mm <- frankC(F1.2m,F3.2m,ro13.2)

F1.23p <- (C13.2pp-C13.2pm)/f3.2; F1.23m <- (C13.2mp-C13.2mm)/f3.2; f1.23 <- F1.23p-F1.23m

logL.z <- D$w*log(f2*f3.2*f1.23)
-sum(logL.z)
}
#.....................................................................................

logV.r <- function(p){
Ro12 <- p[1]; Ro23 <- p[2]; Ro13.2 <- p[3]
n1 <- p[(4+0*dY):(3+1*dY)]
n2 <- p[(4+1*dY):(3+2*dY)]
n3 <- p[(4+2*dY):(3+3*dY)]

eYn1 <- exp(Y%*%n1); p1 <- eYn1/(1+eYn1)
eYn2 <- exp(Y%*%n2); p2 <- eYn2/(1+eYn2)
eYn3 <- exp(Y%*%n3); p3 <- eYn3/(1+eYn3)

F1r <- function(r){
      I10 <- which(r <  0)
      I11 <- which(r >= 0 & r < 1)
      I12 <- which(r >= 1)

      F[I10] <- 0
      F[I11] <- p1[I11]
      F[I12] <- 1
      F
}
F2r <- function(r){
      I20 <- which(r <  0)
      I21 <- which(r >= 0 & r < 1)
      I22 <- which(r >= 1)

      F[I20] <- 0
      F[I21] <- p2[I21]
      F[I22] <- 1
      F
}
F3r <- function(r){
      I30 <- which(r <  0)
      I31 <- which(r >= 0 & r < 1)
      I32 <- which(r >= 1)
```

```
      F[I30] <- 0
      F[I31] <- p3[I31]
      F[I32] <- 1
      F
}
F1.p <- F1r(Dr1); F1.m <- F1r(Dr1-1)
F2.p <- F2r(Dr2); F2.m <- F2r(Dr2-1); f.2 <- F2.p-F2.m
F3.p <- F3r(Dr3); F3.m <- F3r(Dr3-1)

c12pp <- frankC(F1.p,F2.p,Ro12); c23pp <- frankC(F2.p,F3.p,Ro23)
c12pm <- frankC(F1.p,F2.m,Ro12); c23pm <- frankC(F2.p,F3.m,Ro23)
c12mp <- frankC(F1.m,F2.p,Ro12); c23mp <- frankC(F2.m,F3.p,Ro23)
c12mm <- frankC(F1.m,F2.m,Ro12); c23mm <- frankC(F2.m,F3.m,Ro23)

F1.2.p <- (c12pp-c12pm)/f.2; F1.2.m <- (c12mp-c12mm)/f.2
F3.2.p <- (c23pp-c23mp)/f.2; F3.2.m <- (c23pm-c23mm)/f.2; f.3.2 <- F3.2.p-F3.2.m

c13.2pp <- frankC(F1.2.p,F3.2.p,Ro13.2)
c13.2pm <- frankC(F1.2.p,F3.2.m,Ro13.2)
c13.2mp <- frankC(F1.2.m,F3.2.p,Ro13.2)
c13.2mm <- frankC(F1.2.m,F3.2.m,Ro13.2)

F1.23.p <- (c13.2pp-c13.2pm)/f.3.2; F1.23.m <- (c13.2mp-c13.2mm)/f.3.2
f.1.23 <- F1.23.p-F1.23.m

logL.r <- D$w*log(f.2*f.3.2*f.1.23)
-sum(logL.r)
}
#-------------------------------------------------------------------------------------------
########################################
#                                      #
# MACRO DE LA ETAPA E DEL ALGORITMO EM #
#                                      #
########################################

while(D0>tol & D1>tol & D2>tol & D3>tol & D4>tol & D5>tol & D6>tol){
#-------------------------#
# ESTIMACION DE LOS PESOS #
#-------------------------#
B <- as.matrix(c(B0,B1,B2,B3,B4,B5,B6))
fy[Iw] <- dbinom(D$y[Iw],1,exp(Z[Iw,]%*%B)/(1+exp(Z[Iw,]%*%B)))
In[Iw] <- fz[Iw]*fy[Iw]*fr[Iw]

for(Ii in 1:2){
      D$w[Iw2[Ii,]] <- In[Iw2[Ii,]]/(In[Iw2[1,]]+In[Iw2[2,]])
}
for(Ii in 1:4){
      D$w[Iw4[Ii,]] <- In[Iw4[Ii,]]/(In[Iw4[1,]]+In[Iw4[2,]]+In[Iw4[3,]]+In[Iw4[4,]])
}
for(Ii in 1:8){
      D$w[Iw8[Ii,]] <- In[Iw8[Ii,]]/(In[Iw8[1,]]+In[Iw8[2,]]+In[Iw8[3,]]+In[Iw8[4,]]+
                                     In[Iw8[5,]]+In[Iw8[6,]]+In[Iw8[7,]]+In[Iw8[8,]])
}
#-------------------------------------------------------------------------------------------
########################################
#                                      #
# MACRO DE LA ETAPA M DEL ALGORITMO EM #
#                                      #
########################################

#-----------------------------------------------------------#
# ESTIMACION DE LOS PARAMETROS DE REGRESION DE LA RESPUESTA y #
#-----------------------------------------------------------#
M0 <- B0; M1 <- B1; M2 <- B2; M3 <- B3; M4 <- B4; M5 <- B5; M6 <- B6
BE <- glm(y~x1+x2+x3+z1+z2+z3,family=binomial(link=logit),weights=w,data=D)
B0 <- BE$coef[1]; B1 <- BE$coef[2]; B2 <- BE$coef[3]; B3 <- BE$coef[4]
B4 <- BE$coef[5]; B5 <- BE$coef[6]; B6 <- BE$coef[7]
D0 <- abs(M0-B0); D1 <- abs(M1-B1); D2 <- abs(M2-B2); D3 <- abs(M3-B3)
D4 <- abs(M4-B4); D5 <- abs(M5-B5); D6 <- abs(M6-B6)
```

```
#-------------------------------------------------------------------------#
# ESTIMACION DE LOS PARAMETROS DE LA DISTRIBUCION CONJUNTA DEL VECTOR z #
#-------------------------------------------------------------------------#
epz <- nlminb(c(ro12,ro23,ro13.2,a11,a21,a31),logV.z)

ro12 <- epz$par[1]; ro23 <- epz$par[2]; ro13.2 <- epz$par[3]
a11 <- epz$par[(4+0*dX):(3+1*dX)]
a21 <- epz$par[(4+1*dX):(3+2*dX)]
a31 <- epz$par[(4+2*dX):(3+3*dX)]

eXa11 <- exp(X%*%a11); q11 <- eXa11/(1+eXa11)
eXa21 <- exp(X%*%a21); q21 <- eXa21/(1+eXa21)
eXa31 <- exp(X%*%a31); q31 <- eXa31/(1+eXa31)

F1z <- function(z){
      I10 <- which(z <  Lz1[1])
      I11 <- which(z >= Lz1[1] & z < Lz1[2])
      I12 <- which(z >= Lz1[2])

      F[I10] <- 0
      F[I11] <- q11[I11]
      F[I12] <- 1
      F
}
F2z <- function(z){
      I20 <- which(z <  Lz2[1])
      I21 <- which(z >= Lz2[1] & z < Lz2[2])
      I22 <- which(z >= Lz2[2])

      F[I20] <- 0
      F[I21] <- q21[I21]
      F[I22] <- 1
      F
}
F3z <- function(z){
      I30 <- which(z <  Lz3[1])
      I31 <- which(z >= Lz3[1] & z < Lz3[2])
      I32 <- which(z >= Lz3[2])

      F[I30] <- 0
      F[I31] <- q31[I31]
      F[I32] <- 1
      F
}
F1p <- F1z(Dz1); F1m <- F1z(Dz1-1)
F2p <- F2z(Dz2); F2m <- F2z(Dz2-1); f2 <- F2p-F2m
F3p <- F3z(Dz3); F3m <- F3z(Dz3-1)

C12pp <- frankC(F1p,F2p,ro12); C23pp <- frankC(F2p,F3p,ro23)
C12pm <- frankC(F1p,F2m,ro12); C23pm <- frankC(F2p,F3m,ro23)
C12mp <- frankC(F1m,F2p,ro12); C23mp <- frankC(F2m,F3p,ro23)
C12mm <- frankC(F1m,F2m,ro12); C23mm <- frankC(F2m,F3m,ro23)

F1.2p <- (C12pp-C12pm)/f2; F1.2m <- (C12mp-C12mm)/f2
F3.2p <- (C23pp-C23mp)/f2; F3.2m <- (C23pm-C23mm)/f2; f3.2 <- F3.2p-F3.2m

C13.2pp <- frankC(F1.2p,F3.2p,ro13.2)
C13.2pm <- frankC(F1.2p,F3.2m,ro13.2)
C13.2mp <- frankC(F1.2m,F3.2p,ro13.2)
C13.2mm <- frankC(F1.2m,F3.2m,ro13.2)

F1.23p <- (C13.2pp-C13.2pm)/f3.2; F1.23m <- (C13.2mp-C13.2mm)/f3.2; f1.23 <- F1.23p-F1.23m

fz <- f2*f3.2*f1.23
```

```
#-------------------------------------------------------------------------#
# ESTIMACION DE LOS PARAMETROS DE LA DISTRIBUCION CONJUNTA DEL VECTOR r #
#-------------------------------------------------------------------------#
epr <- nlminb(c(Ro12,Ro23,Ro13.2,n1,n2,n3),logV.r)

Ro12 <- epr$par[1]; Ro23 <- epr$par[2]; Ro13.2 <- epr$par[3]
n1 <- epr$par[(4+0*dY):(3+1*dY)]
n2 <- epr$par[(4+1*dY):(3+2*dY)]
n3 <- epr$par[(4+2*dY):(3+3*dY)]

eYn1 <- exp(Y%*%n1); p1 <- eYn1/(1+eYn1)
eYn2 <- exp(Y%*%n2); p2 <- eYn2/(1+eYn2)
eYn3 <- exp(Y%*%n3); p3 <- eYn3/(1+eYn3)

F1r <- function(r){
      I10 <- which(r <  0)
      I11 <- which(r >= 0 & r < 1)
      I12 <- which(r >= 1)

      F[I10] <- 0
      F[I11] <- p1[I11]
      F[I12] <- 1
      F
}
F2r <- function(r){
      I20 <- which(r <  0)
      I21 <- which(r >= 0 & r < 1)
      I22 <- which(r >= 1)

      F[I20] <- 0
      F[I21] <- p2[I21]
      F[I22] <- 1
      F
}
F3r <- function(r){
      I30 <- which(r <  0)
      I31 <- which(r >= 0 & r < 1)
      I32 <- which(r >= 1)

      F[I30] <- 0
      F[I31] <- p3[I31]
      F[I32] <- 1
      F
}
F1.p <- F1r(Dr1); F1.m <- F1r(Dr1-1)
F2.p <- F2r(Dr2); F2.m <- F2r(Dr2-1); f.2 <- F2.p-F2.m
F3.p <- F3r(Dr3); F3.m <- F3r(Dr3-1)

c12pp <- frankC(F1.p,F2.p,Ro12); c23pp <- frankC(F2.p,F3.p,Ro23)
c12pm <- frankC(F1.p,F2.m,Ro12); c23pm <- frankC(F2.p,F3.m,Ro23)
c12mp <- frankC(F1.m,F2.p,Ro12); c23mp <- frankC(F2.m,F3.p,Ro23)
c12mm <- frankC(F1.m,F2.m,Ro12); c23mm <- frankC(F2.m,F3.m,Ro23)

F1.2.p <- (c12pp-c12pm)/f.2; F1.2.m <- (c12mp-c12mm)/f.2
F3.2.p <- (c23pp-c23mp)/f.2; F3.2.m <- (c23pm-c23mm)/f.2; f.3.2 <- F3.2.p-F3.2.m

c13.2pp <- frankC(F1.2.p,F3.2.p,Ro13.2)
c13.2pm <- frankC(F1.2.p,F3.2.m,Ro13.2)
c13.2mp <- frankC(F1.2.m,F3.2.p,Ro13.2)
c13.2mm <- frankC(F1.2.m,F3.2.m,Ro13.2)

F1.23.p <- (c13.2pp-c13.2pm)/f.3.2; F1.23.m <- (c13.2mp-c13.2mm)/f.3.2
f.1.23 <- F1.23.p-F1.23.m

fr <- f.2*f.3.2*f.1.23
}
```

```
#############################################################################
#                                                                           #
# MACRO PARA ESTIMAR LOS ERRORES ESTANDAR DE LOS PARAMETROS DE REGRESION #
#                                                                           #
#############################################################################

y <- D$y
m <- exp(Z%*%B)/(1+exp(Z%*%B))
d <- exp(Z%*%B)/(1+exp(Z%*%B))^2

W <- diag(D$w)
V <- diag(as.vector(d))
H <- diag(as.vector(y-m))

I <- t(Z)%*%W%*%V%*%Z - t(Z)%*%W%*%(H%*%H)%*%(diag(dim(W)[1])-W)%*%Z
C <- solve(I)

EMV <- B; E.E <- sqrt(diag(C)); V.Z <- EMV/E.E; V.P <- 2*(1-pnorm(abs(V.Z)))
#-----------------------------------------------------------------------------------------------
########################################
#                                      #
# MACRO PARA DESPLEGAR LOS RESULTADOS #
#                                      #
########################################

cat("METODO DE CASOS COMPLETOS (CC):\n")
CC.sum <- summary(CC); print(CC.sum)
cat("x1(desestandarizada): Estimate =",round(CC.sum$coef[2,1]/sd(E1684$age),5),
    ", Std.Error =",round(CC.sum$coef[2,2]/sd(E1684$age),5),"\n")

cat("METODO DE IMPUTACION MULTIPLE (MI):\n")
print(data.frame(Estimate=round(MI.sum[,1],5),Std.Error=round(MI.sum[,2],7),
                 z.value=round(MI.sum[,3],4),P.value=round(MI.sum[,5],5)))
cat("x1(desestandarizada): Estimate =",round(MI.sum[2,1]/sd(E1684$age),5),
    ", Std.Error =",round(MI.sum[2,2]/sd(E1684$age),5),"\n")

cat("METODO DE MAXIMA VEROSIMILITUD CON COPULAS PAREADAS (PCML_MAR):\n")
print(data.frame(Estimate=round(EMV,5),Std.Error=round(E.E,7),
                 z.value=round(V.Z,4),P.value=round(V.P,5)))
cat("x1(desestandarizada): Estimate =",round(EMV[2]/sd(E1684$age),5),
    ", Std.Error =",round(E.E[2]/sd(E1684$age),7),"\n")

cat("METODO DE MAXIMA VEROSIMILITUD CON COPULAS PAREADAS (PCML_MNAR):\n")
print(data.frame(Estimate=round(EMV,5),Std.Error=round(E.E,7),
                 z.value=round(V.Z,4),P.value=round(V.P,5)))
cat("x1(desestandarizada): Estimate =",round(EMV[2]/sd(E1684$age),5),
    ", Std.Error =",round(E.E[2]/sd(E1684$age),7),"\n")
#-----------------------------------------------------------------------------------------------
```

Bibliografía

Aas, K., Czado, C., Frigessi, A., and Bakken, H. (2009). Pair-copula constructions of multiple dependence. *Insurance: Mathematics and Economics*, 44(2):182–198.

Bedford, T. and Cooke, R. M. (2001a). Monte Carlo simulation of vine dependent random variables for applications in uncertainty analysis. In *2001 Proceedings of ESREL2001*, Turin, Italy.

Bedford, T. and Cooke, R. M. (2001b). Probability density decomposition for conditionally dependent random variables modeled by vines. *Annals of Mathematic and Artificial Intelligense*, 32(1):245–268.

Bedford, T. and Cooke, R. M. (2002). Vines – a new graphical model for dependent random variables. *Annals of Statistics*, 30(4):1031–1068.

Box, G. E. P., Hunter, J. S., and Hunter, W. G. (2005). *Statistics for Experimenters: Design, Innovation, and Discovery*. John Wiley & Sons, Hoboken, NJ, 2nd edition.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 39(1):1–38.

Dunn, P. K. and Smyth, G. K. (1996). Randomized quantile residuals. *Journal of Computational and Graphical Statistics*, 5(3):236–244.

Eddelbuettel, D. and François, R. (2011). Rcpp: Seamless R and C++ integration. *Journal of Statistical Software*, 40(8):1–18.

Enders, C. K. (2010). *Applied Missing Data Analysis*. Guilford Press, New York, NY.

Hartley, H. O. and Hocking, R. R. (1971). The analysis of incomplete data. *Biometrics*, 27(4):783–823.

Hofert, M., Kojadinovic, I., Maechler, M., and Yan, J. (2015). *Copula: Multivariate dependence with copulas*. R package version 0.999-13.

Horton, N. J. and Kleinman, K. P. (2007). Much ado about nothing. *The American Statistician*, 61(1):79–90.

Ibrahim, J. G. (1990). Incomplete data in generalized linear models. *Journal of the American Statistical Association*, 85(411):765–769.

Ibrahim, J. G., Chen, M. H., and Lipsitz, S. R. (1999a). Monte Carlo EM for missing covariates in parametric regression models. *Biometrics*, 55(2):591–596.

Ibrahim, J. G., Lipsitz, S. R., and Chen, M. H. (1999b). Missing covariates in generalized linear models when the missing data mechanism is non–ignorable. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(1):173–190.

Ibrahim, J. G. and Weisberg, S. (1992). Incomplete data in generalized linear models with continuous covariates. *Australian Journal of Statistics*, 34(3):461–470.

Joe, H. (1996). *Families of m-variate distributions with given margins and $m(m-1)/2$ bivariate dependence parameters*, volume 28 of *Lecture Notes–Monograph Series*, pages 120–141. Institute of Mathematical Statistics, Hayward, CA.

Kirkwood, J. M., Strawderman, M. H., Ernstoff, M. S., Smith, T. J., Borden, E. C., and Blum, R. H. (1996). Interferon alfa-2b adjuvant therapy of high-risk resected cutaneous melanoma: The Eastern Cooperative Oncology Group Trial EST 1684. *Journal of Clinical Oncology*, 14(1):7–17.

Lindsey, J. K. (1996). *Parametric Statistical Inference*. Clarendon Press, Oxford, UK.

Lipsitz, S. R. and Ibrahim, J. G. (1996). A conditional model for incomplete covariates in parametric regression models. *Biometrika*, 83(4):916–922.

Little, R. J. A. (1992). Regression with missing X's: A review. *Journal of the American Statistical Association*, 87(420):1227–1237.

Little, R. J. A. and Rubin, D. B. (2002). *Statistical Analysis with Missing Data*. John Wiley & Sons, Hoboken, NJ, 2nd edition.

Louis, T. A. (1982). Finding the observed information matrix when using the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 44(2):226–233.

McCullagh, P. and Nelder, J. A. (1989). *Generalized Linear Models*. Chapman & Hall/CRC, New York, NY, 2nd edition.

McLachlan, G. J. and Krishnan, T. (2008). *The EM Algorithm and Extensions*. John Wiley & Sons, Hoboken, NJ, 2nd edition.

Molenberghs, G. and Kenward, M. G. (2007). *Missing Data in Clinical Studies*. John Wiley & Sons, Hoboken, NJ.

Nelder, J. A. and Wedderburn, R. W. M. (1972). Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, 135(3):370–384.

Nelsen, R. B. (2006). *An Introduction to Copulas*. Springer-Verlag, New York, NY, 2nd edition.

Panagiotelis, A., Czado, C., and Joe, H. (2012). Pair copula constructions for multivariate discrete data. *Journal of the American Statistical Association*, 107(499):1063–1072.

R Core Team (2014). R: A language and environment for statistical computing. R Foundation for Statistical Computing. Vienna, Autria.

Robert, C. P. and Casella, G. (2010). *Introducing Monte Carlo Methods with R*. Springer, New York, NY.

Rubin, D. B. (1976). Inference and missing data. *Biometrika*, 63(3):581–592.

Rubin, D. B. (1987). *Multiple Imputation for Nonresponse in Surveys*. John Wiley & Sons, Hoboken, NJ.

Schafer, J. L. (1997). *Analysis of Incomplete Multivariate Data*. Chapman & Hall/CRC, New York, NY.

Schafer, J. L. and Graham, J. W. (2002). Missing data: Our view of the state of the art. *Psychological Methods*, 7(2):147–177.

Schepsmeier, U., Stöber, J. M., and Brechmann, E. C. (2012). *Vinecopula: Statistical inference of vine copulas*. Technische Universität München, Germany.

Scheuren, F. (2005). Multiple imputation: How it began and continues. *The American Statistician*, 59(4):315–319.

Song, P. X. K. (2000). Multivariate dispersion models generated from Gaussian copula. *Scandinavian Journal of Statistics*, 27(2):305–320.

Stöber, J. M. (2013). *Regular vine copulas with the simplifying assumption, time-variation, and mixed discrete and continuous margins*. PhD thesis, Technische Universität München, Germany.

Van Buuren, S. (2012). *Flexible Imputation of Missing Data*. Chapman & Hall, Boca Raton, FL.

Van Buuren, S. and Groothuis-Oudshoorn, K. (2011). mice: Multivariate imputation by chained equations in R. *Journal of Statistical Software*, 45(3):1–67.

Wood, A. M., White, I. R., and Royston, P. (2008). How should variable selection be performed with multiply imputed data? *Statistics in Medicine*, 27(17):3227–3246.

Casa abierta al tiempo
UNIVERSIDAD AUTÓNOMA METROPOLITANA

ACTA DE DISERTACIÓN PÚBLICA

No. 00060
Matrícula: 2131802118

MODELACIÓN DE DISTRIBUCIONES CONJUNTAS PARA MODELOS LINEALES GERALIZADOS CON DATOS FALTANTES

En la Ciudad de México, se presentaron a las 11:00 horas del día 28 del mes de junio del año 2018 en la Unidad Iztapalapa de la Universidad Autónoma Metropolitana, los suscritos miembros del jurado:

DRA. SILVIA RUIZ VELASCO ACOSTA
DR. GABRIEL ESCARELA PEREZ
DR. ERNESTO JUVENAL BARRIOS ZAMUDIO
DR. CARLOS ERWIN RODRIGUEZ HERNANDEZ VELA
DR. ALBERTO CASTILLO MORALES

Bajo la Presidencia de la primera y con carácter de Secretario el último, se reunieron a la presentación de la Disertación Pública cuya denominación aparece al margen, para la obtención del grado de:

DOCTOR EN CIENCIAS (MATEMATICAS)

DE: LUIS CARLOS PEREZ RUIZ

y de acuerdo con el artículo 78 fracción IV del Reglamento de Estudios Superiores de la Universidad Autónoma Metropolitana, los miembros del jurado resolvieron:

Aprobar

Acto continuo, la presidenta del jurado comunicó al interesado el resultado de la evaluación y, en caso aprobatorio, le fue tomada la protesta.

LUIS CARLOS PEREZ RUIZ
ALUMNO

REVISÓ

LIC. JULIO CESAR DE LARA ISASSI
DIRECTOR DE SISTEMAS ESCOLARES

DIRECTOR DE LA DIVISIÓN DE CBI	PRESIDENTA	VOCAL
DR. JESUS ALBERTO OCHOA TAPIA	DRA. SILVIA RUIZ VELASCO ACOSTA	DR. GABRIEL ESCARELA PEREZ
VOCAL	**VOCAL**	**SECRETARIO**
DR. ERNESTO JUVENAL BARRIOS ZAMUDIO	DR. CARLOS ERWIN RODRIGUEZ HERNANDEZ VELA	DR. ALBERTO CASTILLO MORALES