Análisis estadístico con datos faltantes

Una breve introducción

Mario José Pacheco López
Universidad Santo Tomás
mariopacheco@usta.edu.co

Octubre 18, 2024

El problema de los datos faltantes

El problema de los datos faltantes

Statistics is a missing-data problem

(Roderick J.A. Little)

Little, R. J. (2013). In Praise of Simplicity not Mathematistry! Ten Simple Powerful Ideas for the Statistical Scientist. *Journal of the American Statistical Association, 108*(502), 359–369. https://doi.org/10.1080/01621459.2013.787932

Un problema de datos faltantes

Comparación de dos protocolos de recuperación postoperatoria en pacientes con cardiopatías congénitas y coronarias sometidos a intervenciones quirúrgicas cardíacas

Variable	Descripción
Edad	Edad en años
Sexo	Sexo masculino o femenino
IMC	Índice de masa corporal (kg/m2)
Hospitalización	Tiempo total de internación (en días)
Intervención	Tiempo de duración de la intervención quirúrgica (en horas)
Anestesia	Tiempo de duración de la anestesia (en horas)
Perfusión	Tiempo durante el cual el paciente está conectado a la circulación extracorpórea (en horas)
Recuperación	Tiempo de permanencia en la sala de recuperación (en horas)
Tipo	Paciente con cardiopatía congénita o coronária
Protocolo	Protocolo de recuperación acelerada o convencional

Un problema de datos faltantes

Conjunto de datos

Edad	Sexo	IMC	Hospitalización	Anestesia	Intervención	Recuperación	Perfusión	Tipo	I
1.00	Femenino	16.6	5	3.42	1.75	26.83	0.00	Congénito	C
5.00	Masculino	18.0	7	3.08	1.83	20.83	0.50	Congénito	C
13.00	Masculino	17.5	6	3.33	2.67	38.83	0.83	Congénito	C
11.00	Femenino	18.2	8	2.50	2.00	32.00	0.75	Congénito	C
16.00	Femenino	22.1	6	5.00	3.67	21.75	1.08	Congénito	C
18.00	Femenino	24.7	5	2.42	1.42	31.08	0.00	Congénito	C
11.00	Masculino	14.5	5	3.50	2.33	21.00	1.00	Congénito	C
17.00	Masculino	24.0	5	2.75	1.58	24.75	0.45	Congénito	C
2.00	Masculino	17.8	6	1.75	1.17	25.67	0.00	Congénito	C
1.00	Masculino	14.9	6	2.58	2.00	21.83	0.47	Congénito	C

Un problema de datos faltantes

Posibles análisis

- Explorar el comportamiento de las variables
- Ajustar distribuciones de probabilidad para las variables
- Realizar pruebas de hipótesis entre los tipos de pacientes
- Aplicar un método multivariado como reducción de dimensionalidad o agrupamiento
- Ajustar un modelo de regresión para el número de días de hospitalización

Limitación

- 62 de 172 pacientes presentan valores faltantes en alguna de las variables

Un problema de datos faltantes

Importando el conjunto de datos

```
library(readxl)

# Conjunto de datos
url = "http://www.ime.usp.br/~jmsinger/Dados/Fernandes2002.xls"
temp = tempfile(fileext = ".xls")
download.file(url, destfile = temp, mode = "wb")

datos = read_excel(temp) %>%
  select(-c(1, 4:6, 8:13, 15, 20)) %>%
  rename(Edad = IDADE, Sexo = SEXO, IMC = IMC,
         Hospitalización = DIAS, Intervención = T_IC,
         Anestesia = T_A, Recuperación = T_REC1,
         Perfusión = T_P, Tipo = `TIPO DE PACIENTE`,
         Protocolo = `MÉTODO DE RECUPERAÇÃO`) %>%
  mutate(Sexo = factor(ifelse(Sexo == 0, "Femenino", "Masculino")),
         Tipo = factor(ifelse(Tipo == 0, "Congénito","Coronario")),
         Protocolo = factor(ifelse(Protocolo == 0, "Convencional", "Acelerada"),
                            labels = c("Convencional", "Acelerada")),
         IMC = ifelse(IMC==0, NA, IMC))
```

Un problema de datos faltantes

Datos faltantes

Edad	Sexo	IMC	Hospitalización	Anestesia	Intervención	Recuperación	Perfusión	Tipo	I
4.0	Femenino	16.7	NA	3.67	2.50	21.0	0.50	Congénito	C
2.0	Masculino	NA	5	2.50	1.17	25.3	0.00	Congénito	C
1.0	Femenino	16.9	NA	2.00	1.08	25.0	0.00	Congénito	C
19.0	Femenino	18.8	6	NA	2.67	22.0	0.83	Congénito	C
15.0	Femenino	NA	10	3.00	1.67	23.8	0.58	Congénito	C
7.0	Masculino	15.4	6	3.08	NA	44.1	0.83	Congénito	C
1.0	Masculino	NA	6	3.25	1.75	24.7	2.00	Congénito	C
5.0	Femenino	NA	NA	2.33	1.75	21.7	1.75	Congénito	C
12.0	Masculino	17.3	3	3.25	2.50	NA	1.08	Congénito	C
1.0	Femenino	NA	8	5.92	3.25	43.0	1.03	Congénito	A

Pasos para desarrollar el análisis

Descripción de los datos faltantes

% de faltantes, descripción de patrones, adopción de un mecanismo

Tratamiento de los datos faltantes

Eliminación, imputación, máxima verosimilitud, ...

Análisis estadístico

AED, análisis multivariado, modelos de regresión, pronóstico, ...

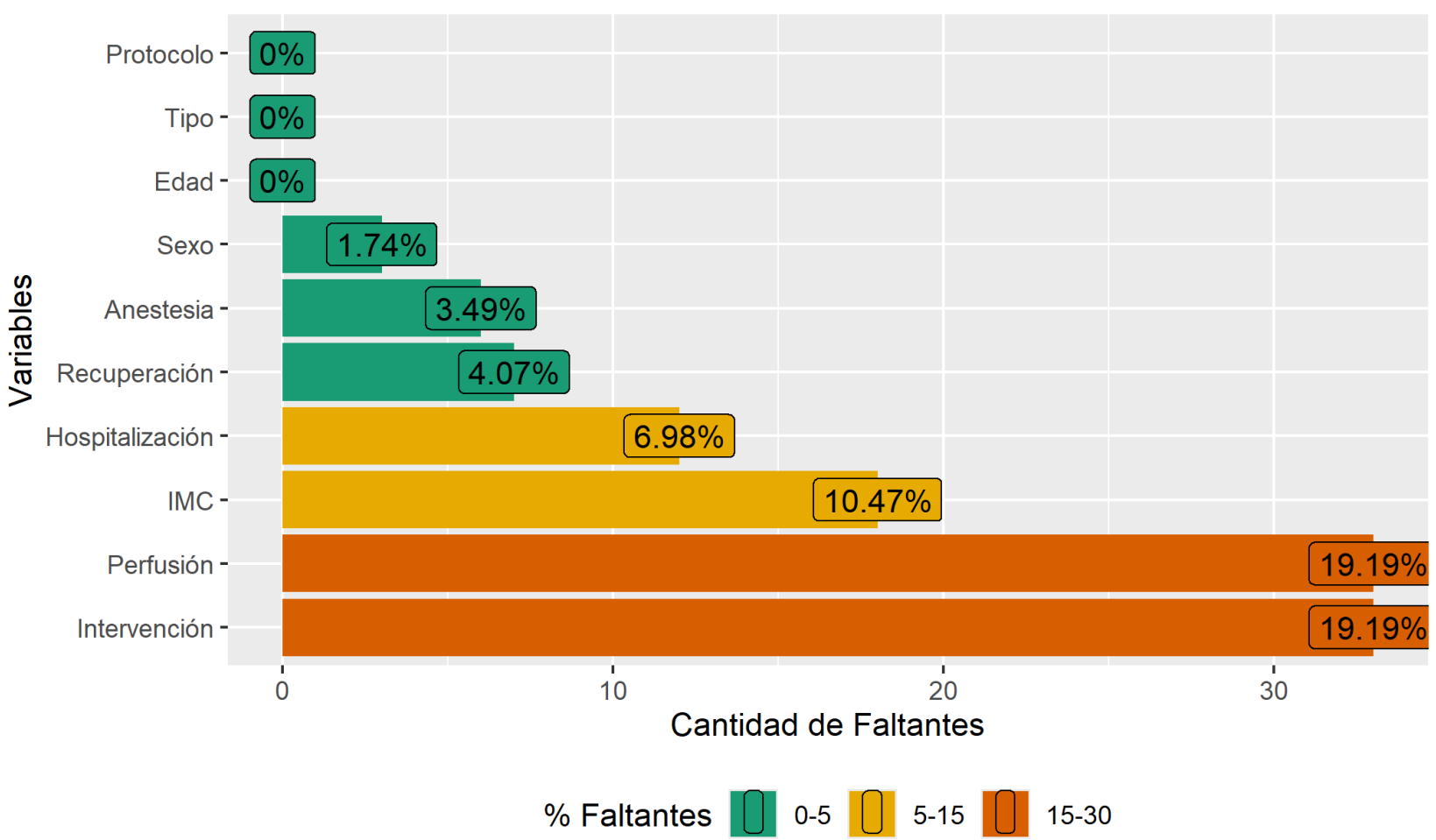
Evaluación de los resultados

Medidas de severidad

Descripción de los datos faltantes

Descripción de los datos faltantes

Porcentaje de faltantes



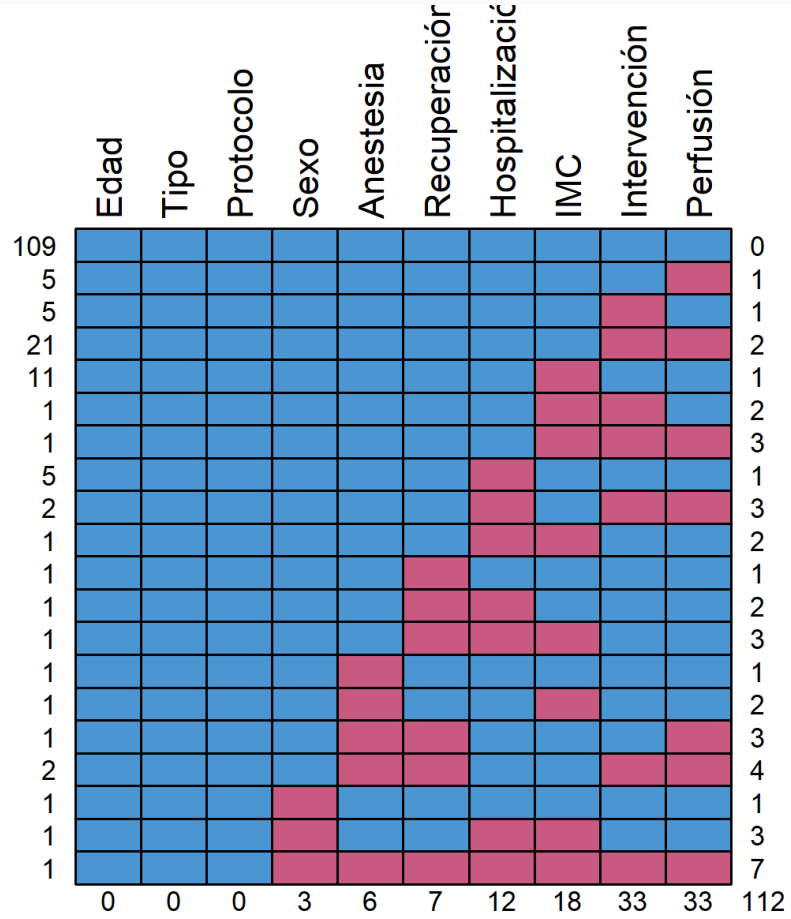
Descripción de los datos faltantes

Porcentaje de faltantes

```
library(DataExplorer)
library(ggplot2)

# Frecuencia de faltantes por variable
datos %>%
  plot_missing(
    group = list(`0-5` = 0.05, `5-15` = 0.15,
                 `15-30` = 0.3, `> 30` = 1),
    group_color = list(`0-5` = "#1B9E77", `5-15` = "#E6AB02",
                       `15-30` = "#D95F02", `> 30` = "#E41A1C")) +
  labs(y = "Cantidad de Faltantes", x = "Variable", fill = "% de faltantes")
```

Descripción de los datos faltantes

Patrones de datos faltantes



Descripción de los datos faltantes

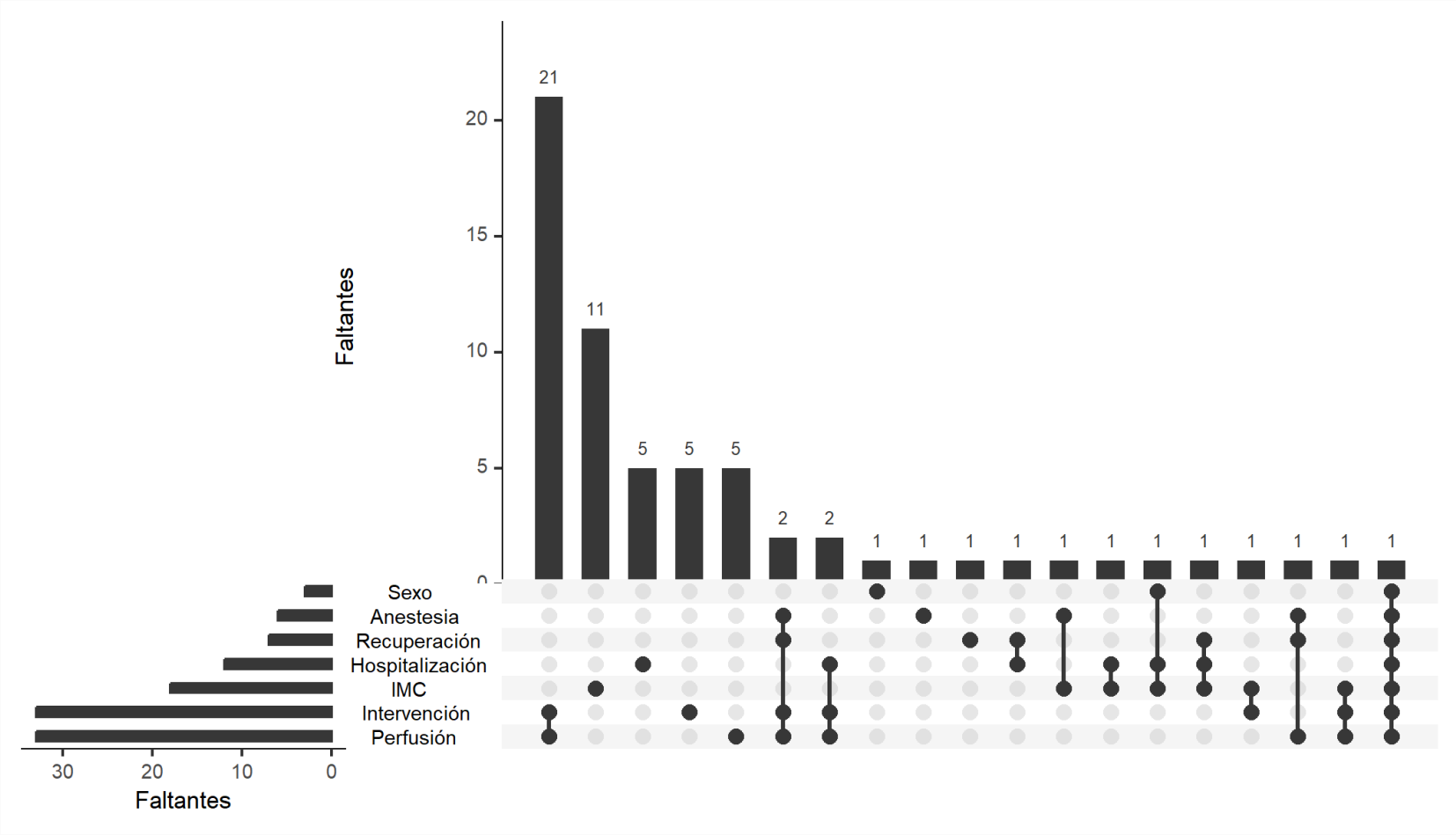
Patrones de datos faltantes

```
library(mice)

# Patrones de faltantes
datos %>%
  md.pattern(rotate.names = TRUE, plot = TRUE)
```

```
library(naniar)

# Otra forma
datos %>%
  gg_miss_upset(nsets = 7)
```

Descripción de los datos faltantes

Patrones de datos faltantes



Descripción de los datos faltantes

Mecanismos de datos faltantes

Faltantes completamente alatorios (MCAR)

Los faltantes en una variable no tienen relación con los valores no observados de la variable ni con los valores observados de otras variables

Faltantes alatorios (MAR)

Los faltantes en una variable están relacionados con los valores observados de otras variables de análisis o covariables, pero no están relacionados con los valores no observados de la variable

Faltantes no alatorios (MNAR)

Los faltantes en una variable están relacionados con los valores no observados de la propia variable

Descripción de los datos faltantes

Test de Little

- H_0 : el mecanismo es MCAR
- H_0 : el mecanismo no es MCAR

Little's test			
statistic	**df**	**p.value**	**missing.patterns**
201	81	0	18

```
library(naniar)

# Prueba MCAR de Little
datos %>%
  select(where(is.numeric)) %>%
  mcar_test()
```

Tratamiento de los datos faltantes

Métodos de eliminación

- Análisis de casos completos (*Listwise deletion*): elimina todos los individuos con al menos una variable faltante
- Análisis de casos disponibles (*Pairwise deletion*): intenta minimizar la pérdida que se produce en el análisis de casos completos, haciendo una eliminación por pares de variables
- No recomendados debido a la pérdida de información
- Un valor faltante puede tener tanta o más información que un valor observado

Tratamiento de datos faltantes

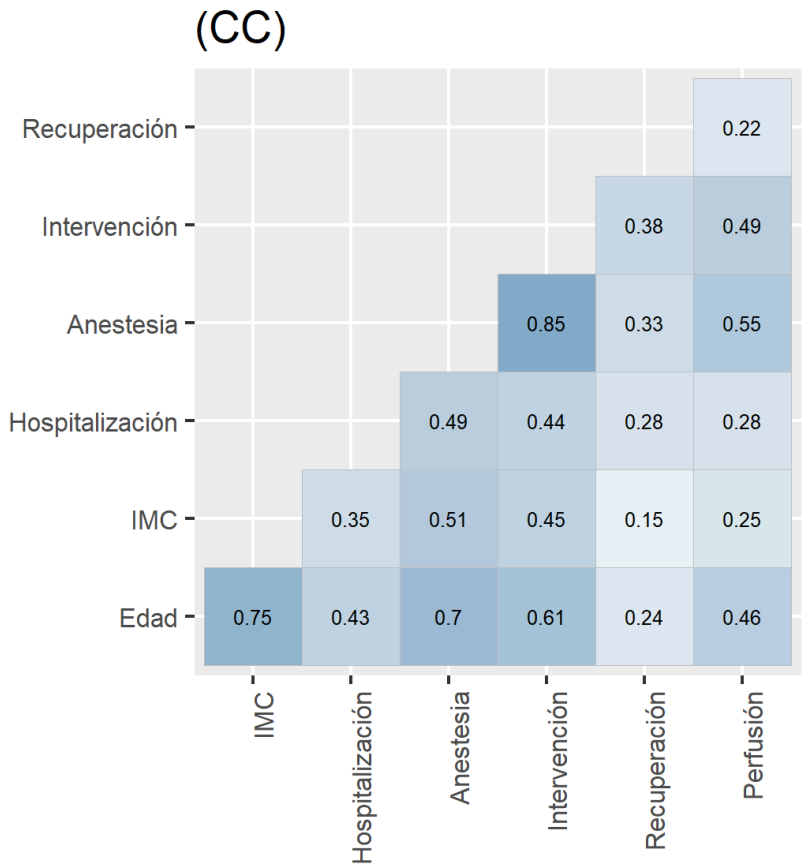
Vectores de promedios

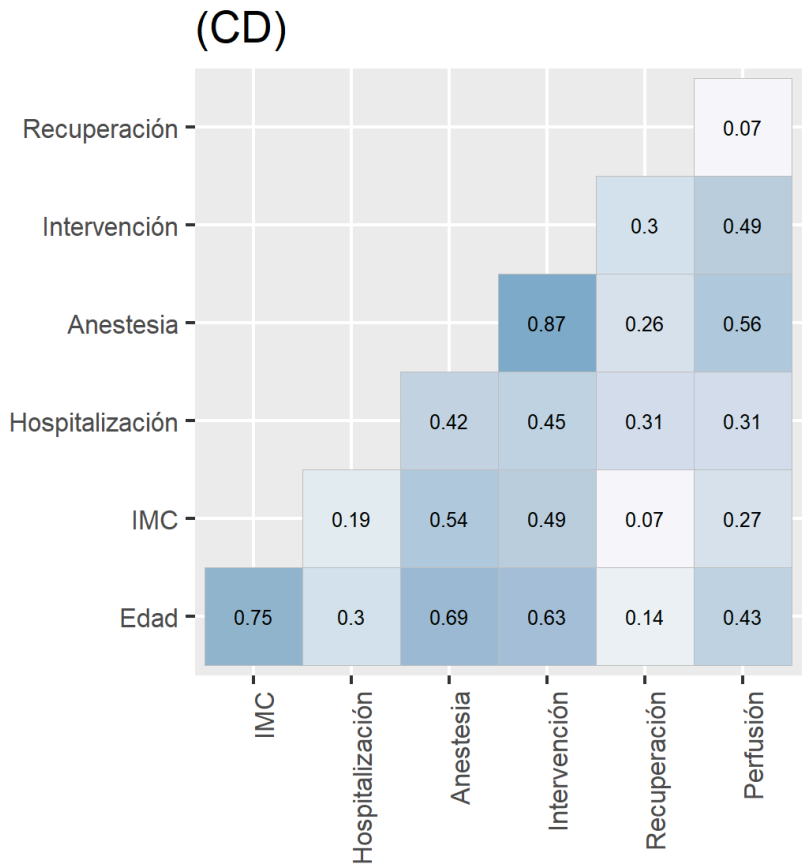
Promedios		
Variable	**CC**	**CD**
Edad	32.325	36.166
IMC	21.730	22.509
Hospitalización	11.764	12.019
Anestesia	4.749	5.025
Intervención	3.435	3.441
Recuperación	32.089	31.847
Perfusión	0.925	0.978

```
datos %>%
  select(where(is.numeric)) %>%
  na.omit() %>%
  colMeans()

datos %>%
  select(where(is.numeric)) %>%
  colMeans(na.rm = TRUE)
```

Tratamiento de datos faltantes

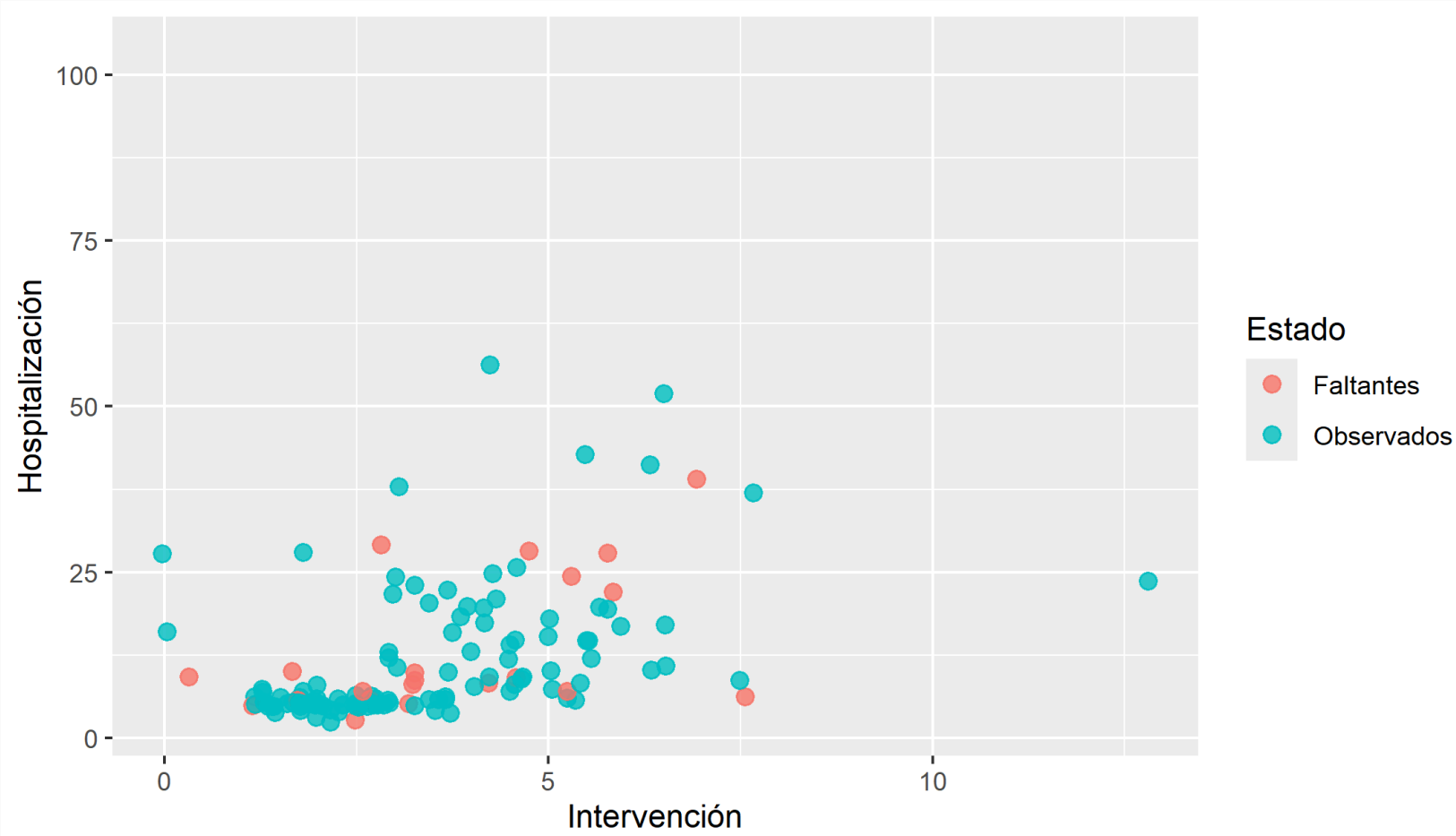
Matrices de correlación





Tratamiento de datos faltantes

Relación entre variables



Imputación

Si no eliminamos los valores faltantes, podemos reemplazarlos. La imputación implica utilizar un procedimiento estadístico para reemplazar los valores faltantes con valores **plausibles** tomando en cuenta, generalmente, el resto de los datos.

> **Imputing one value for a missing datum cannot be correct in general, because we don't know what value to impute with certainty (if we did, it wouldn't be missing)**
>
> **(Donald B. Rubin)**

Rubin, D. J. (1987). Multiple Imputation for Nonresponse in Surveys. Wiley.

Métodos de imputación

Modelos de imputación

Determinísticos o estocásticos

Estrategias de imputación

Imputación simple o múltiple

Métodos de imputación

Basados en regresión, basados en donantes, ...

Algoritmos de imputación

Imputación multivariada mediante ecuaciones encadenadas (MICE)

Imputación

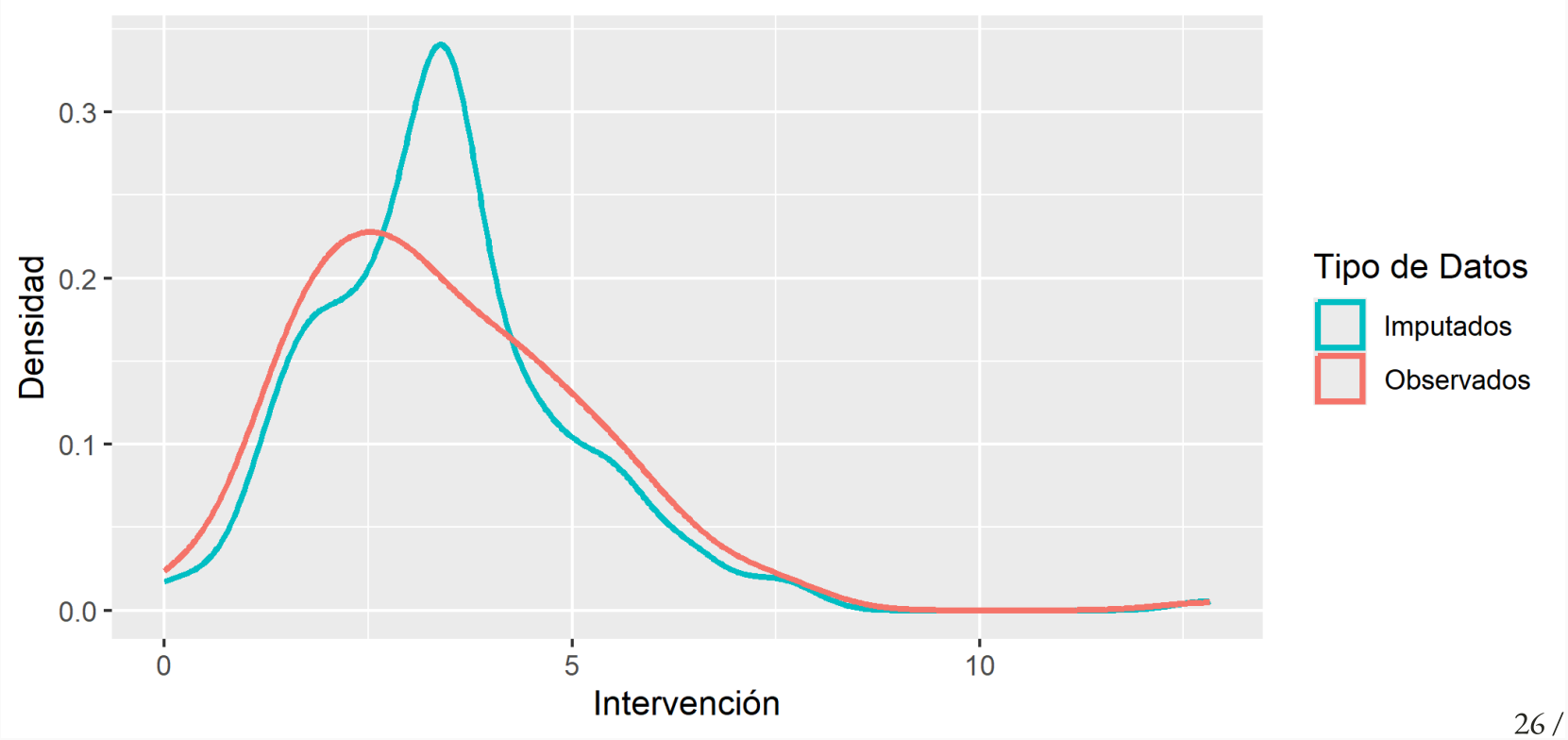
Los métodos de imputación deben garantizar:

- La distribución de las variables no se altere
- Las asociaciones entre variables se mantenga
- Los valores imputados sean consistentes

Métodos de imputación

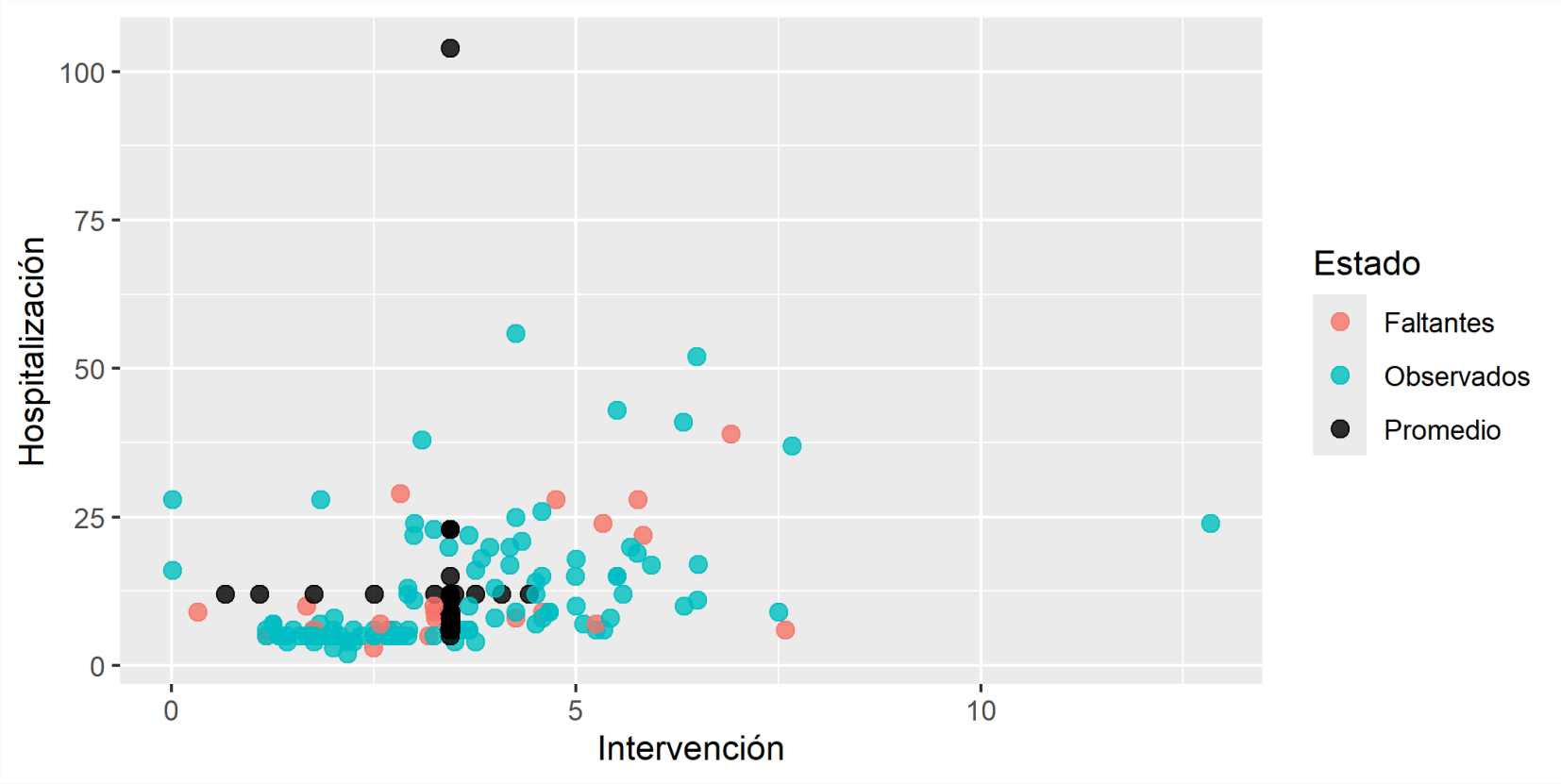
Imputación por media incondicional

Consiste en reemplazar cada valor faltante en una variable por el promedio de los valores observados de la variable.



Métodos de imputación

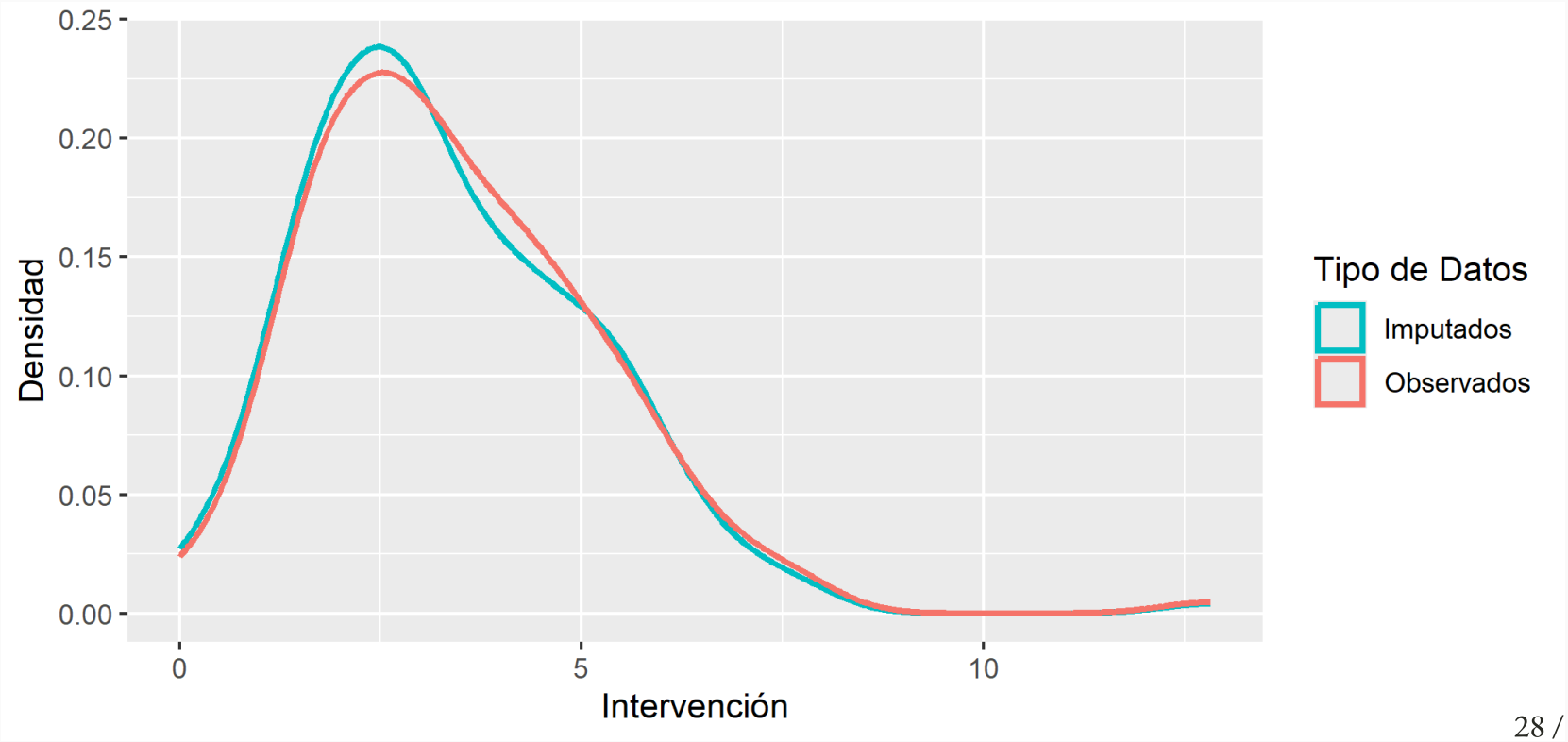
Imputación por media incondicional



Métodos de imputación

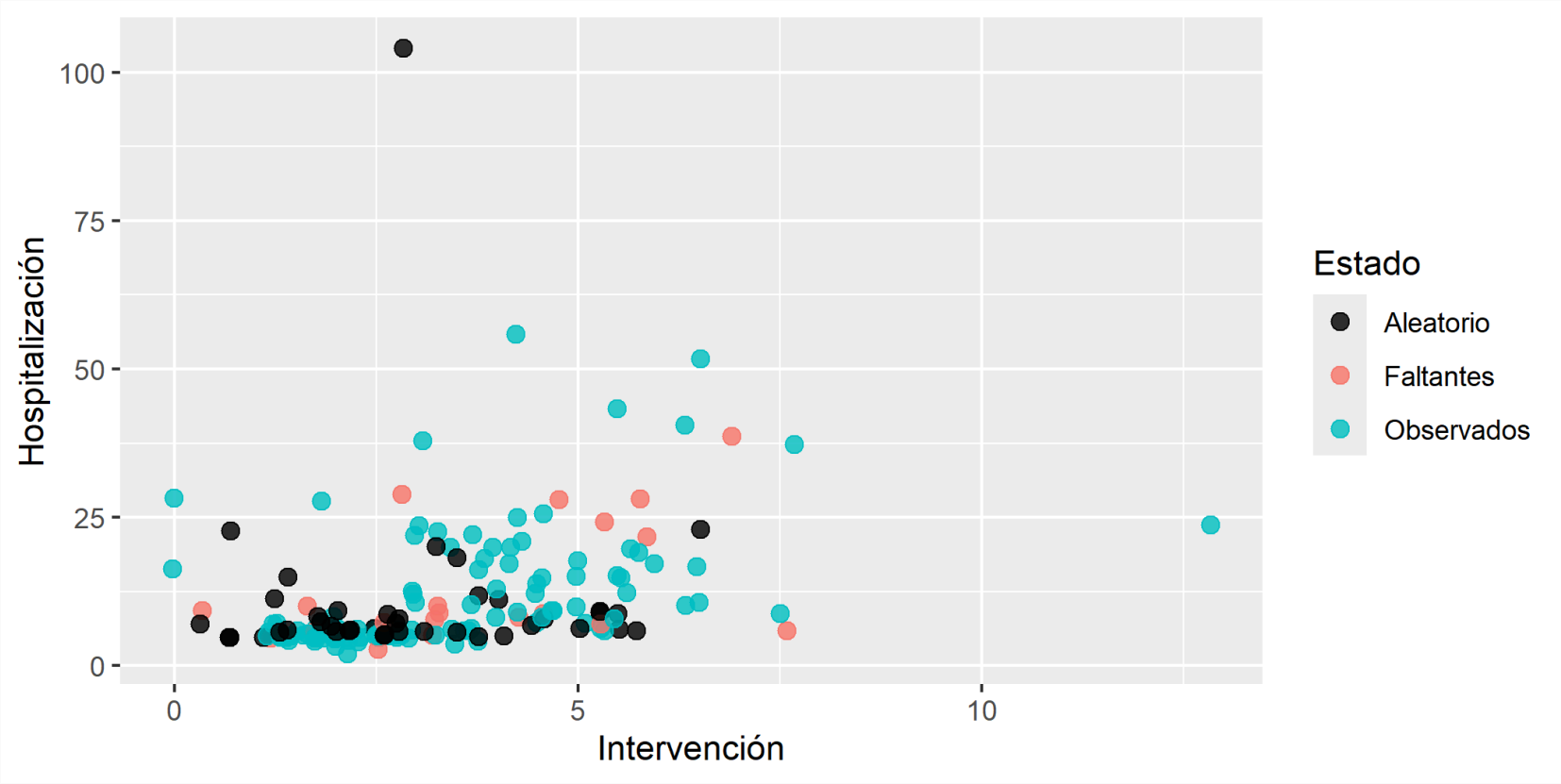
Imputación aleatoria

Consiste en reemplazar cada valor faltante en una variable por valores aleatorios seleccionados de los valores observados de la variable.



Métodos de imputación

Imputación aleatoria



Métodos de imputación

Imputación por media incondicional o aleatoria

```
library(Hmisc)

datos.imp = datos %>%
  select(where(is.numeric)) %>%
  mutate(across(everything(), \(x) impute(x, fun = mean))) # media incondicional
# mutate(across(everything(), \(x) impute(x, "random")) # aleatoria

Estado = ifelse(ici(datos[c("Intervención", "Hospitalización")]), "Promedio", Estado)

ggplot(datos.imp, aes(x = Intervención, y = Hospitalización, colour = Estado)) +
  geom_jitter(size = 2.5, alpha = 0.8) +
  scale_color_manual(values = c("Faltantes" = "#F8766D",
                                "Observados" = "#00BFC4",
                                "Promedio" = "black"))
```

Métodos basados en regresión

- Se construyen modelos de regresión para cada variable con valores faltantes tomando como variables predictoras el resto de variables disponibles (**variables de análisis**) o **covariables** de acuerdo a cada patrón de valores faltantes, por ejemplo

$$\hat{y}_k^{miss} = \hat{\beta}_0 + \sum_{j \neq k} \hat{\beta}_j y_j^{obs}$$

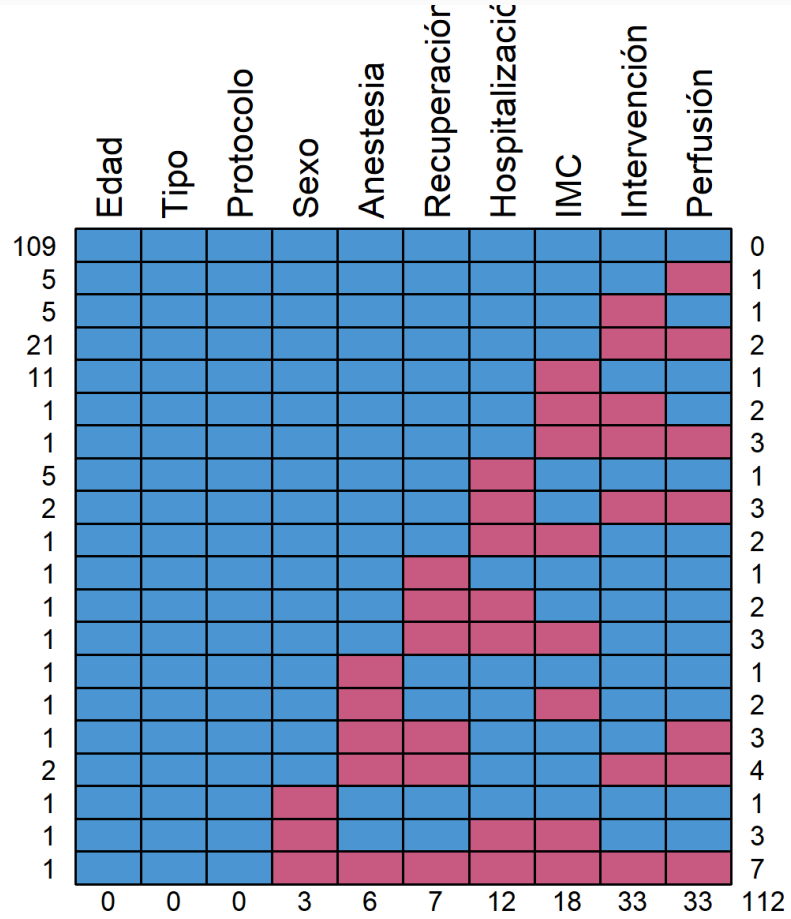
- Imputación por regresión estocástica: le agrega un factor aleatorio a cada modelo

$$\hat{y}_k^{miss} = \hat{\beta}_0 + \sum_{j \neq k} \hat{\beta}_j y_j^{obs} + e_k, \;\; e_k \sim N(0, \hat{\sigma}_k^2)$$

- Puede generar imputaciones por fuera del soporte de la variable
- Requiere ajustar un modelo de regresión para cada variable dentro de cada patrón de faltantes

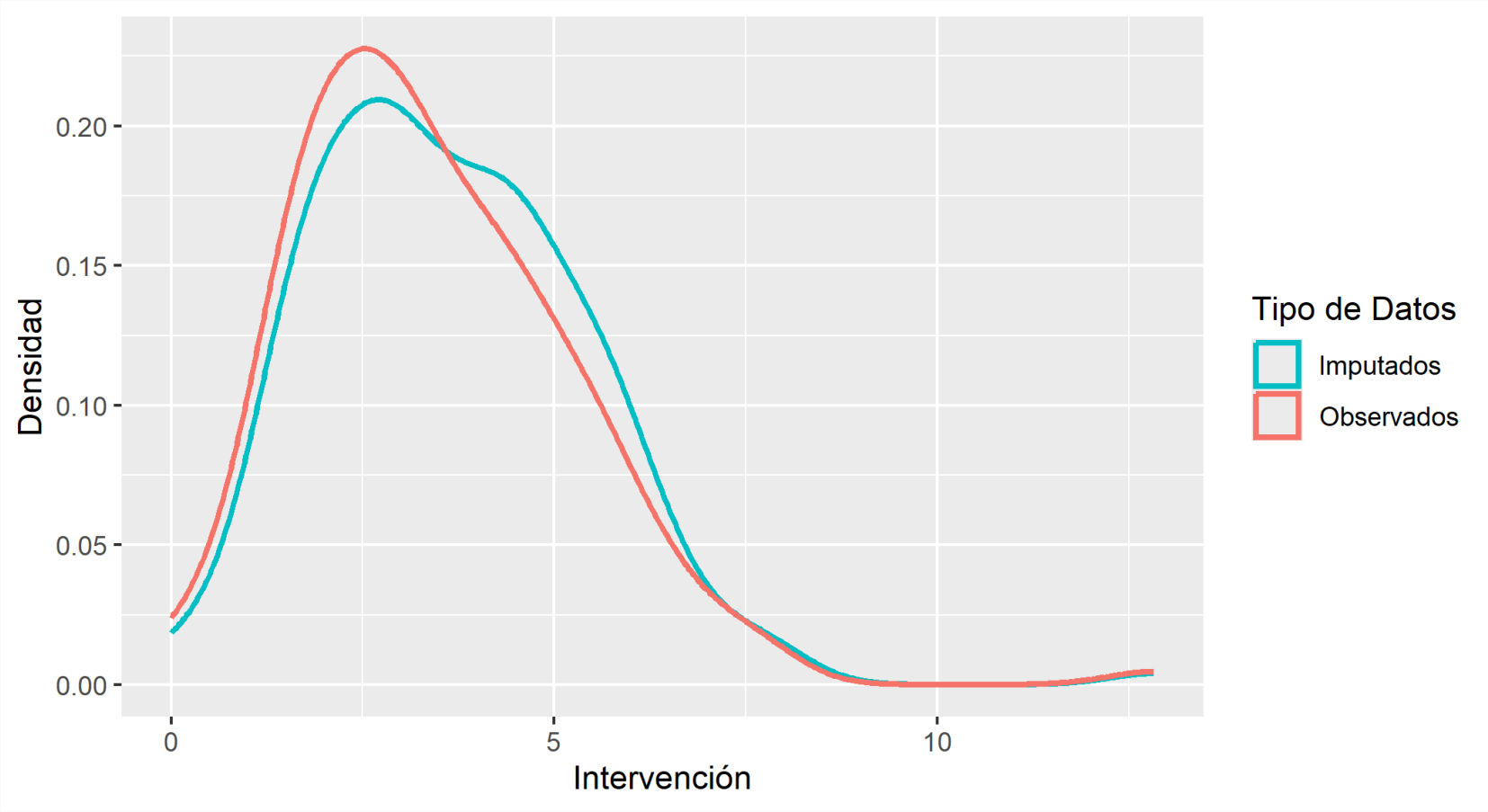
Métodos de imputación

Métodos basados en regresión



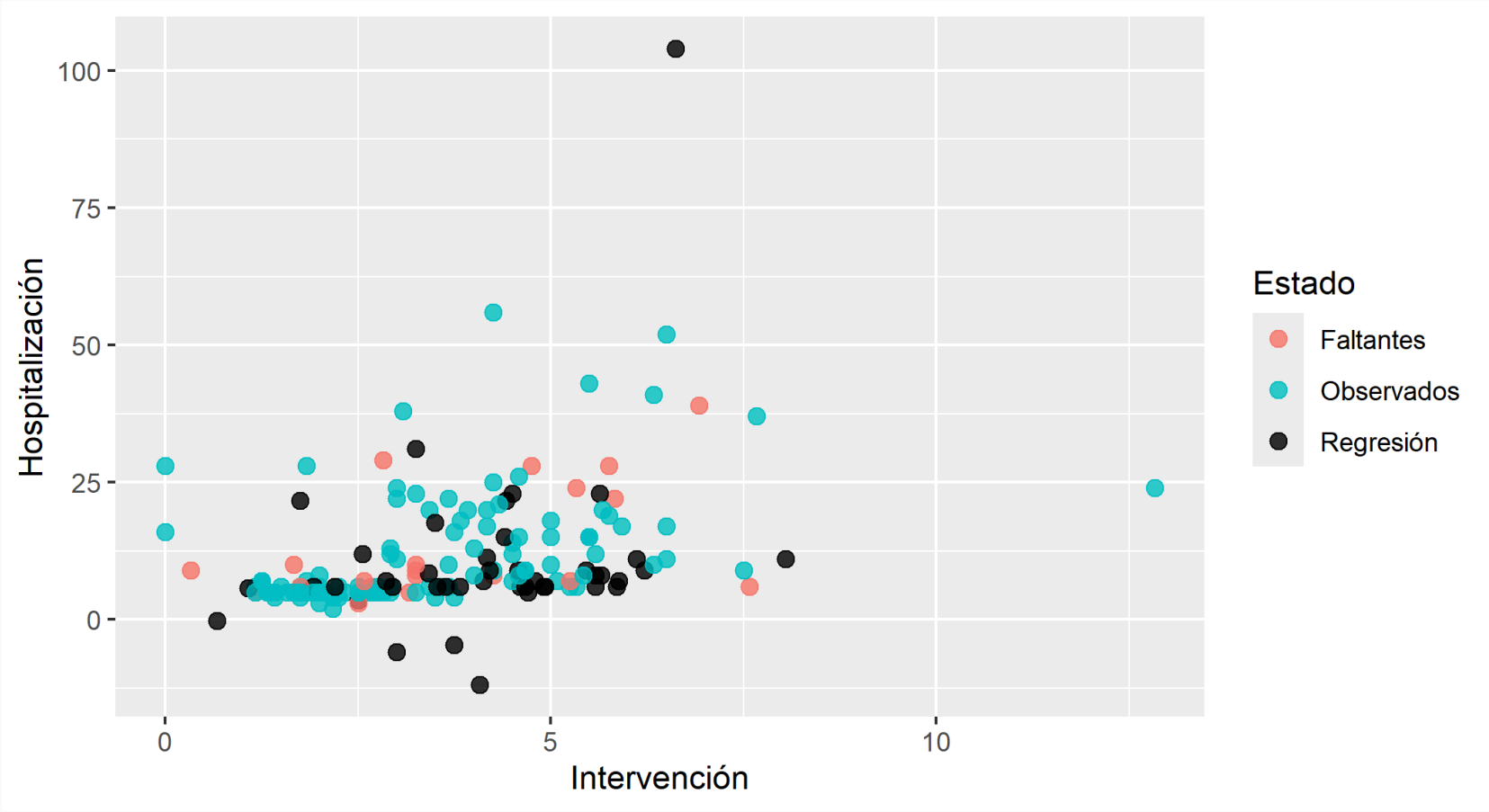
Métodos de imputación

Imputación por regresión estocástica



Métodos de imputación

Imputación por regresión estocástica



Métodos de imputación

Imputación por regresión determinística y estocástica

```
datos.imp = datos
patrones = data.frame(md.pattern(datos, plot = FALSE))[,1:ncol(datos)]
variables = names(patrones)
k = nrow(patrones) - 1 ; k
set.seed(123)
for(j in 2:k){
  resp = variables[patrones[j,]==0]
  expl = variables[patrones[j,]==1]
  for(v in resp){
    if(class(datos[[v]])=="numeric"){
      mod = lm(paste(v, "~", paste(expl, collapse = " + ")), data = datos)
      sig = sigma(mod)
    # pred = predict(mod, datos) # determinística
      pred = predict(mod, datos) + rnorm(nrow(datos), 0, sig) # estocástica
      datos.imp[is.na(datos.imp[,v]),v] = pred[is.na(datos.imp[,v])]
    }
  }
}
```

Métodos de imputación

Métodos basados en donantes

- Imputar los valores faltantes de un individuo (receptor) empleando valores observados de otros individuos (donantes)
- Cada donante se elige de forma que sea lo más parecido al individuo imputado en una o más características observadas
- Algunos métodos:
  - Imputación *Hot-Deck*
  - Imputación *kNN*
  - Imputación por coincidencia media predictiva

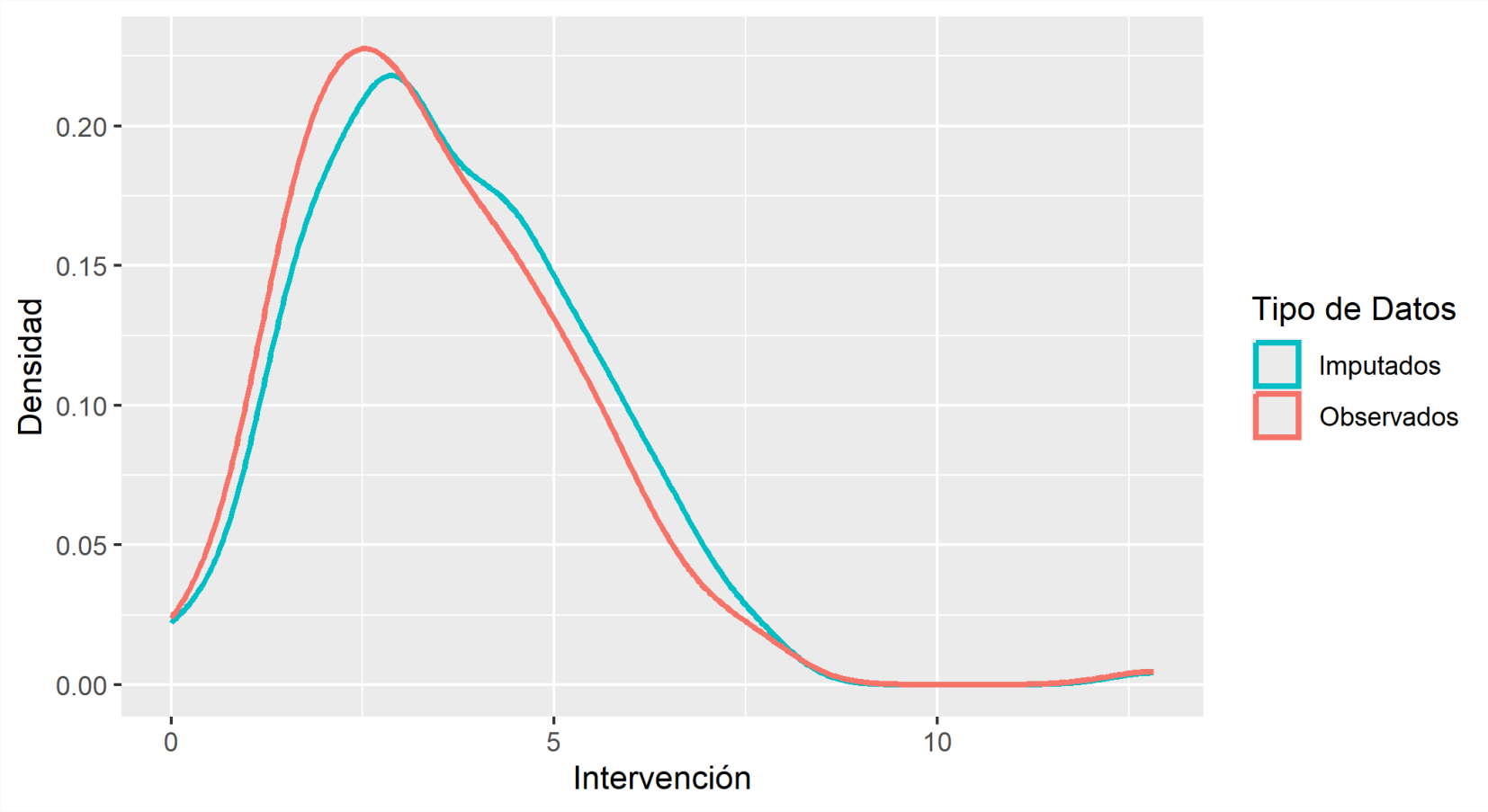
Métodos de imputación

El método *Hot-Deck*

- Crear clases o grupos de imputación en función de variables auxiliares categóricas
- El grupo de donantes potenciales de la unidad receptora consiste en las unidades dentro de la misma clase con la variable observada
- De estos donantes potenciales, se selecciona uno al azar, a través de un muestreo aleatorio simple, y se utiliza para imputar al receptor
- Este procedimiento implica que el donante y el receptor tienen exactamente los mismos valores en todas las variables auxiliares que se utilizan para definir las clases de imputación

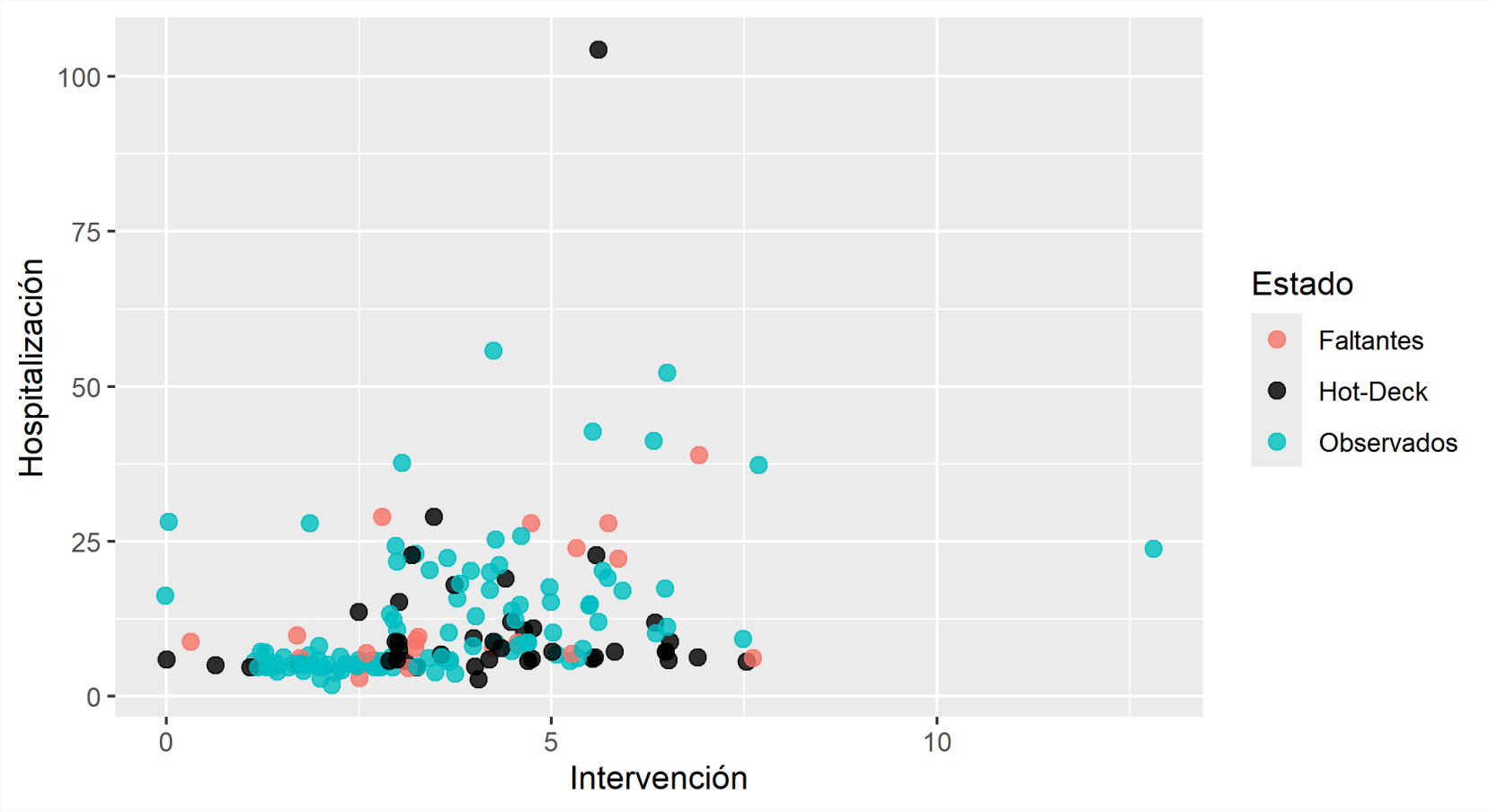
Métodos de imputación

El método *Hot-Deck*



Métodos de imputación

El método *Hot-Deck*



Métodos de imputación

El método *Hot-Deck*

```
library(VIM)

datos.imp = hotdeck(datos,
  domain_var = "Tipo", # variable de grupo de imputación
  imp_var = FALSE
)
```

Métodos de imputación

El método *kNN*

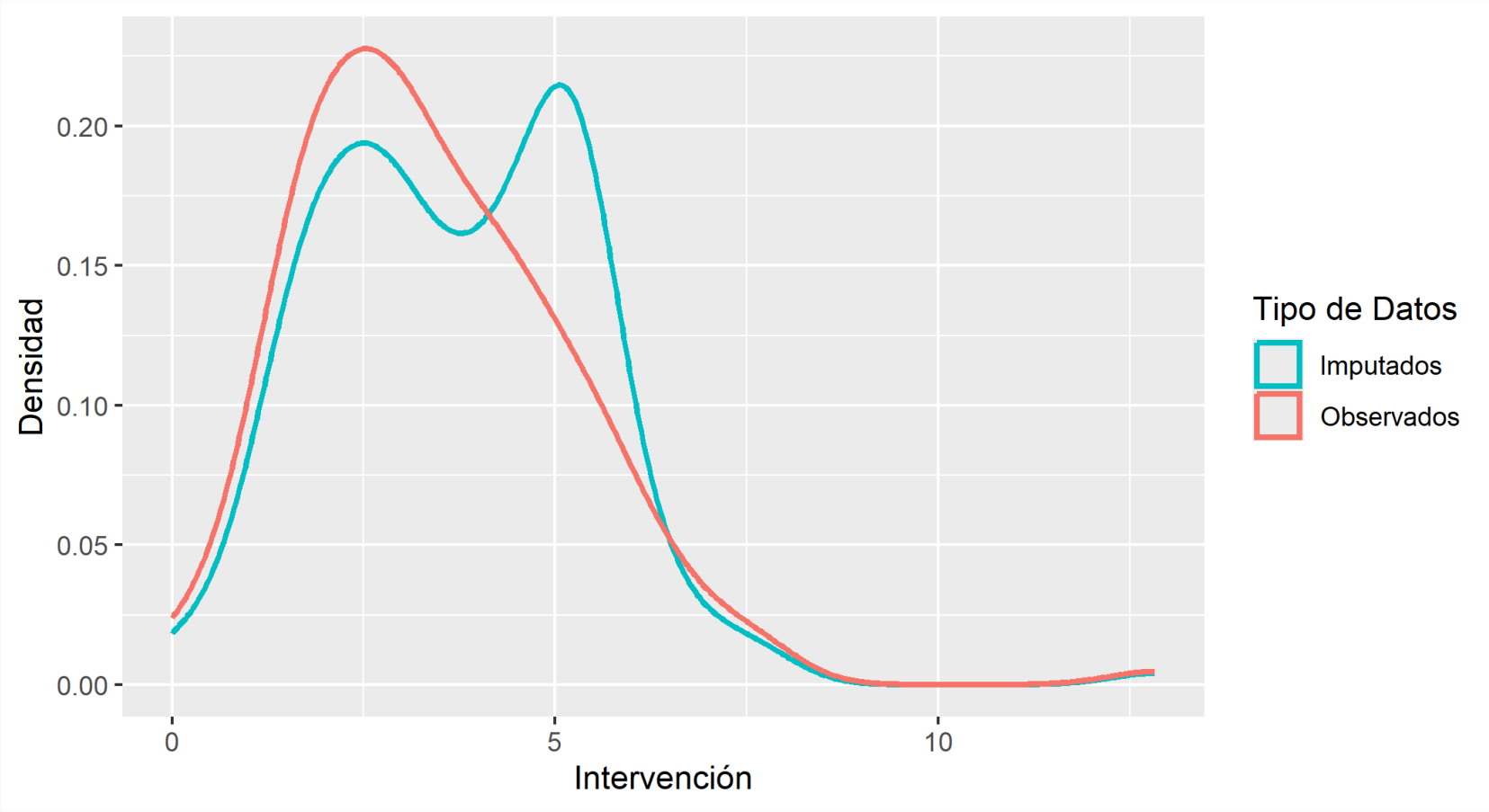
- Se desea imputar el valor faltante y_{rs} del individuo receptor r en la variable s
- Se define una medida de distancia entre el individuo receptor y los posibles donantes, (i's), en términos de las variables observadas, por ejemplo,

$$D(i,r) = \left(\sum_{j \neq s} (y_{ij} - y_{rj})^2 \right)^{1/2}$$

- Se eligen los k vecinos más cercanos del individuo receptor como posibles donantes
- Imputar el individuo receptor tomando al azar el valor de la variable de uno de los k vecinos más cercanos

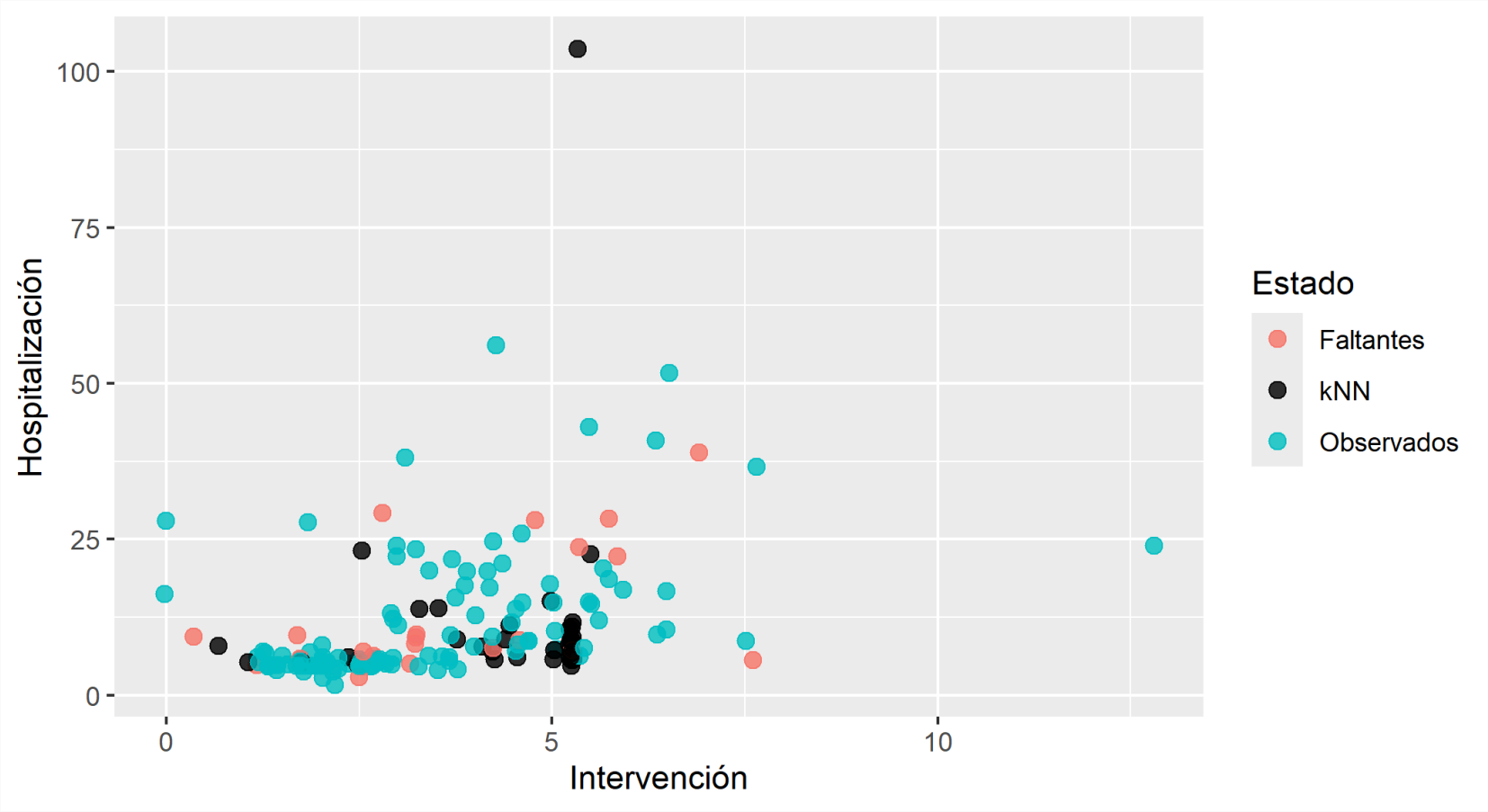
Métodos de imputación

El método *kNN*



Métodos de imputación

El método *kNN*



Métodos de imputación

El método *kNN*

```
# Emplea la distancia de Gower

datos.imp = kNN(datos,
                k = 5, # número de vecinos más cercanos
                imp_var = FALSE)
```

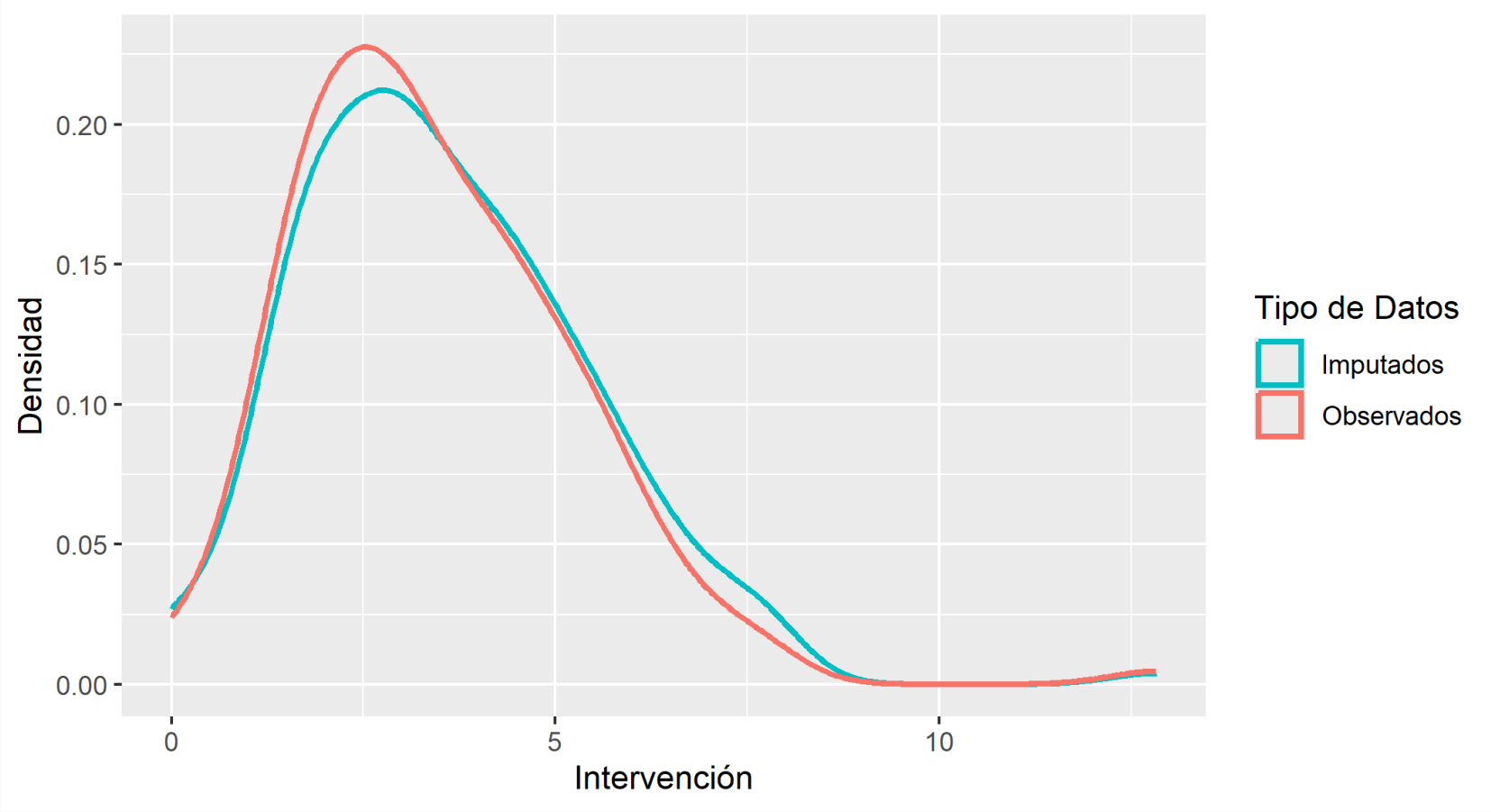
Métodos de imputación

Coincidencia media predictiva (*pmm*)

- Se desea imputar el valor faltante y_{rj} del individuo r en la variable j
- Calcular los valores predichos $\hat{y}_{ij}$ de acuerdo a un modelo específico de imputación
- Seleccionar como donante de y_{rj} , el candidato más próximo, esto es, el valor y_{ij} tal que $|\hat{y}_{ij} - \hat{y}_{rj}|$ es mínimo
- Se puede también seleccionar aleatoriamente uno de los k candidatos más cercanos

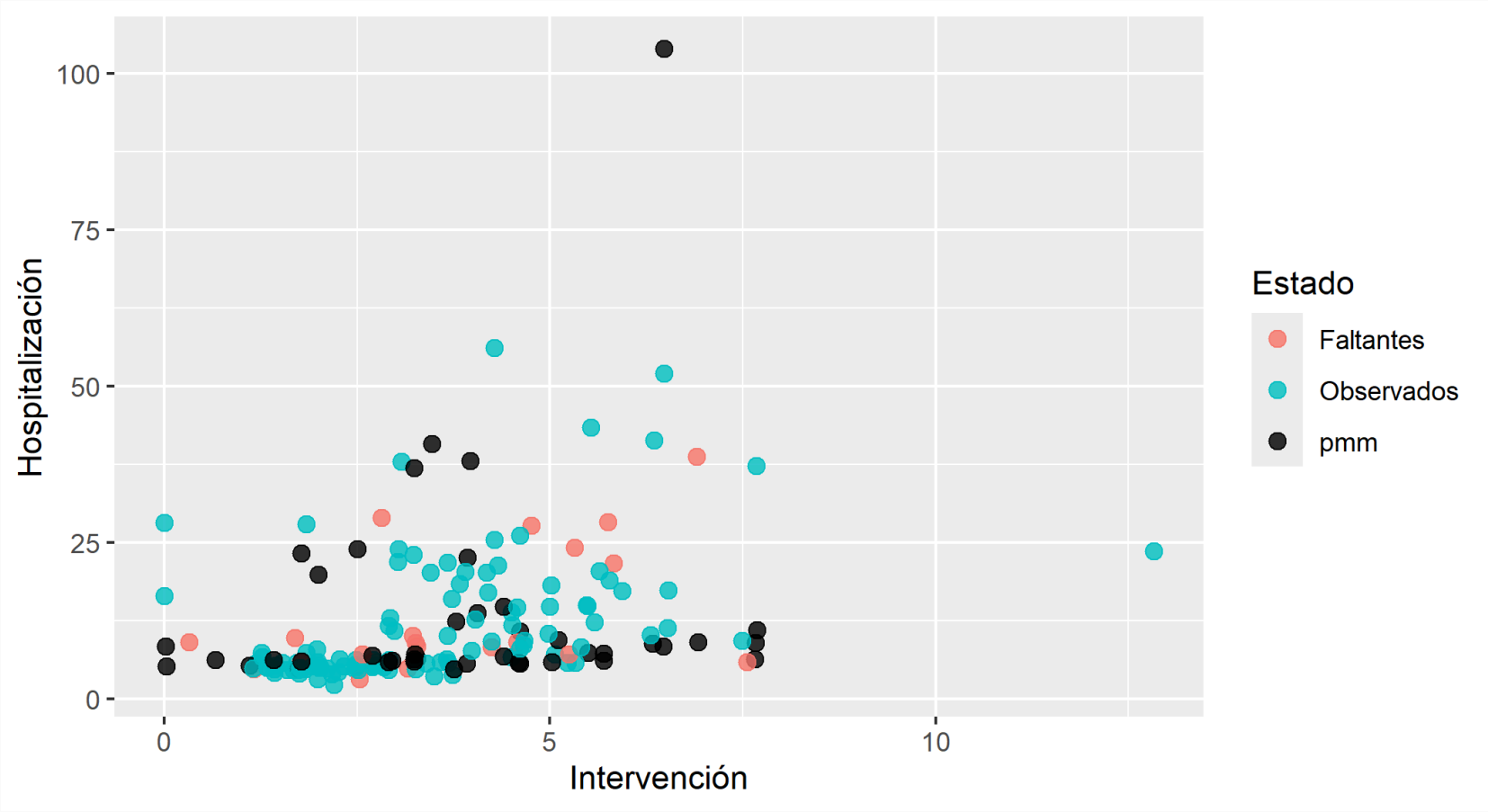
Métodos de imputación

Coincidencia media predictiva (*pmm*)



Métodos de imputación

Coincidencia media predictiva (*pmm*)



Métodos de imputación

Coincidencia media predictiva (*pmm*)

```
# Emplea regresión estocástica
datos.imp = datos
patrones = data.frame(md.pattern(datos, plot = FALSE))[,1:ncol(datos)]
variables = names(patrones) ; k = nrow(patrones) - 1
K = 5 # vecinos más cercanos
set.seed(123)
for (j in 2:k) {
  resp = variables[patrones[j, ] == 0] ; expl = variables[patrones[j, ] == 1]
  for(v in resp){
    if(class(datos[[v]]) == "numeric"){
      mod = lm(paste(v, "~", paste(expl, collapse = " + ")), data = datos)
      sig = sigma(mod) ; pred = predict(mod, datos)
      for(i in which(is.na(datos[[v]]))){
        if(!is.na(pred[i])){
         distancias = abs(pred - (pred[i]+rnorm(1, 0, sig)))
         orden = order(distancias)
         cercanos = orden[!is.na(datos[[v]][orden])][1:K]
         donante = sample(datos[[v]][cercanos], 1)
         datos.imp[i, v] = donante
}}}}}
```

Métodos de imputación

Algoritmo MICE

- *Iter. 0*: Realizar una imputación aleatoria de todos los valores faltantes del conjunto de datos
- *Iter. 1*: Seleccionar un método de imputación para cada variable y realizar el proceso de imputación empleando los valores observados e imputados del resto de variables
- *Iter. t*: actualizar las imputaciones de los valores faltantes empleando los valores observados e imputados de la iteración anterior (Repetir)

```
library(mice)

# Proceso de imputación
proc.imp = mice(datos, # conjunto de datos con faltantes
  m = 5, # número de imputaciones
  maxit = 15, # número de iteraciones
  defaultMethod = c("pmm", "logreg", "polyreg", "polr"), # métodos
  seed = 123 # semilla
)
```

van Buuren, S., Groothuis-Oudshoorn, K. (2011). mice: Multivariate Imputation by Chained Equations in R. *Journal of Statistical Software, 45*(3), 1-67. https://doi.org/10.18637/jss.v045.i03

Métodos de imputación

Imputation is not prediction

(Stef van Buuren)

van Buuren, S. (2018). Flexible Imputation of Missing Data, 2nd Ed. Chapman & Hall

Métodos de imputación

Algoritmo MICE

```
proc.imp$imp$Recuperación
```

```
##        1    2    3     4     5
## 55  16.2 24.8 16.7  32.0  37.5
## 70  14.0 30.0 15.8  20.5  33.8
## 73  21.0 44.1 19.0  36.5  16.5
## 74  25.1  0.0 20.0  45.0  44.0
## 92  29.5 36.5 21.8 108.0  36.2
## 117 33.8 43.3 43.8  21.0  24.5
## 171 35.3 35.3 29.7  26.0 103.0
```

```
proc.imp$imp$Hospitalización[1:10,]
```

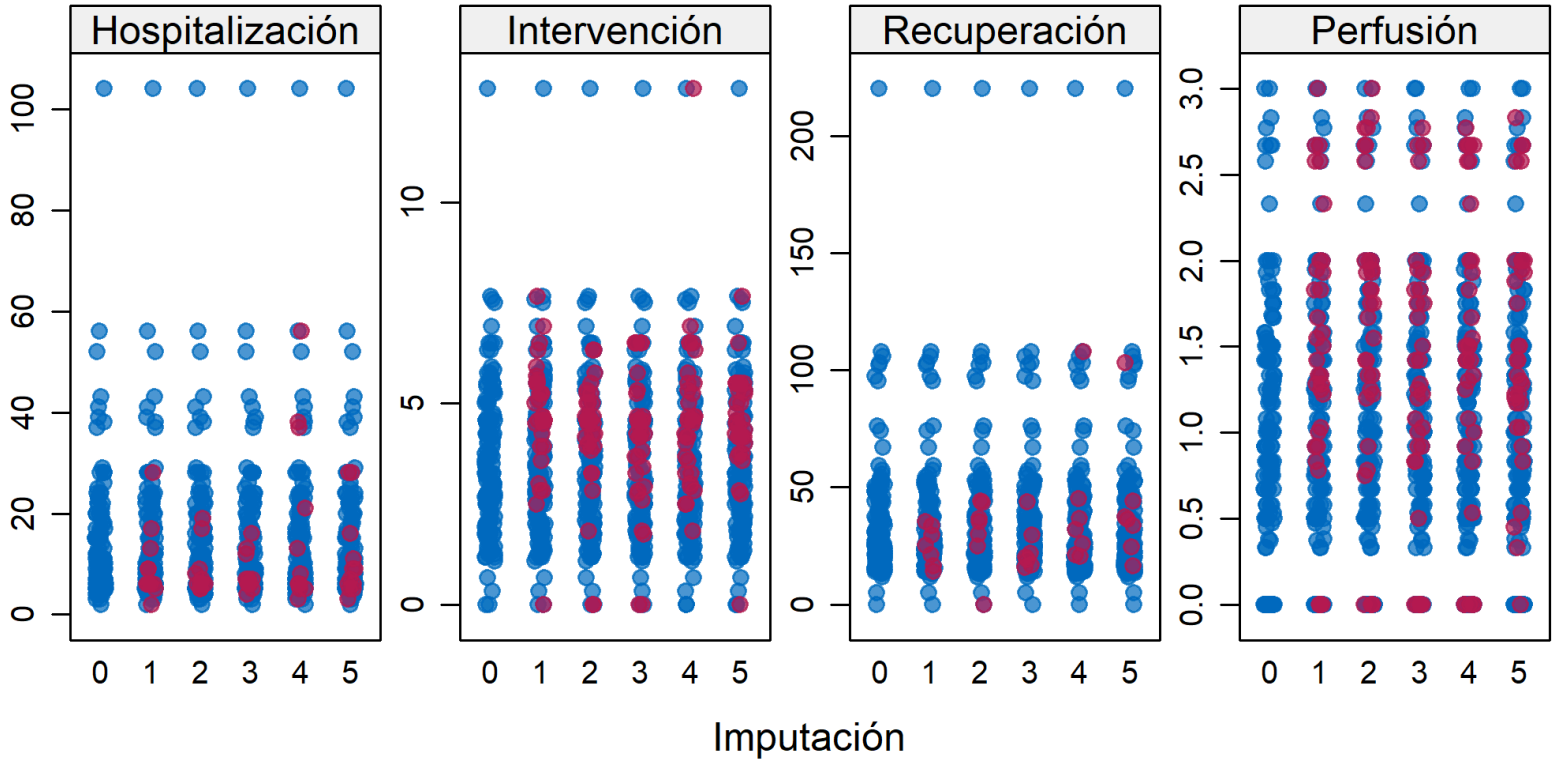
```
##      1  2  3  4  5
## 15   2  6  7  6  7
## 20   5  6  7  3  6
## 38   6  5  6  6  5
## 69  28 17 12 56  9
## 70   9  7  7 21  5
## 79   6 19  7 37 16
## 80   6  9 16 38  6
## 92  17  8  6  6 11
## 107  6  6  6  8 28
## 114 13  6  5 13 28
```

```
proc.imp$imp$Sexo
```

```
##             1         2         3         4         5
## 79   Femenino  Femenino  Femenino  Femenino  Femenino
## 81  Masculino Masculino Masculino  Femenino Masculino
## 117 Masculino  Femenino Masculino Masculino Masculino
```

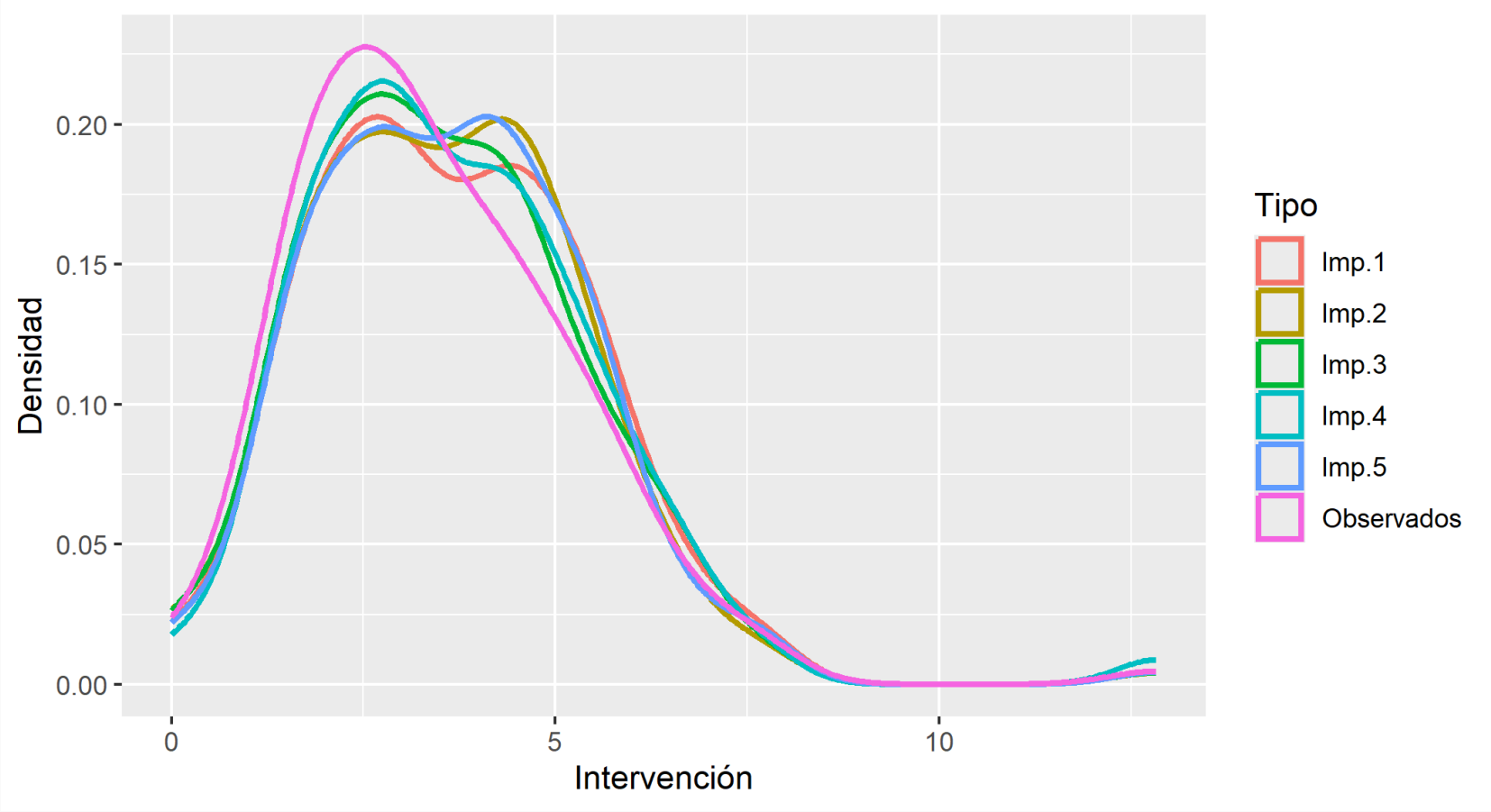
Métodos de imputación

Algoritmo MICE



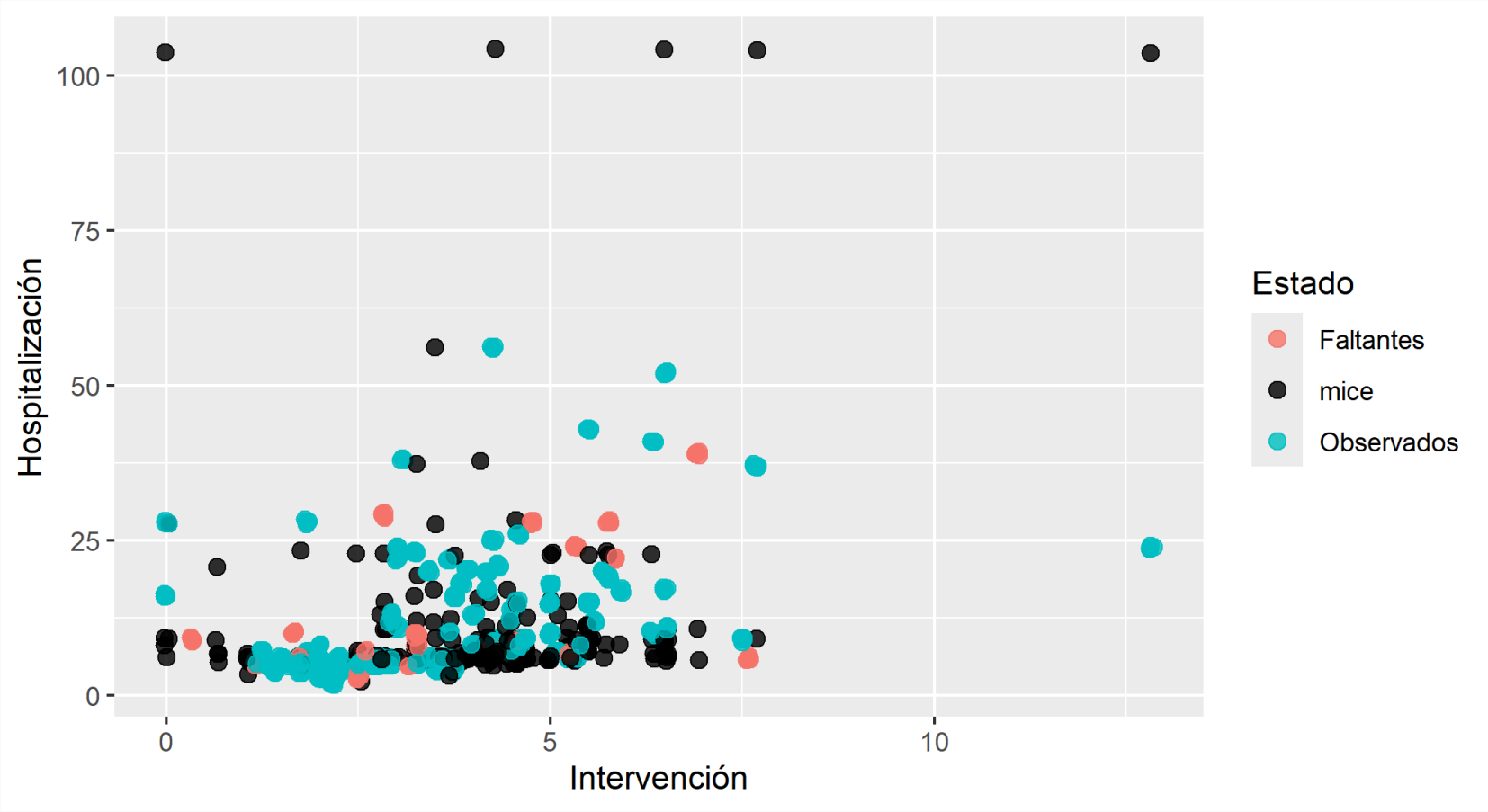
Métodos de imputación

Algoritmo MICE



Métodos de imputación

Algoritmo MICE



Métodos de imputación

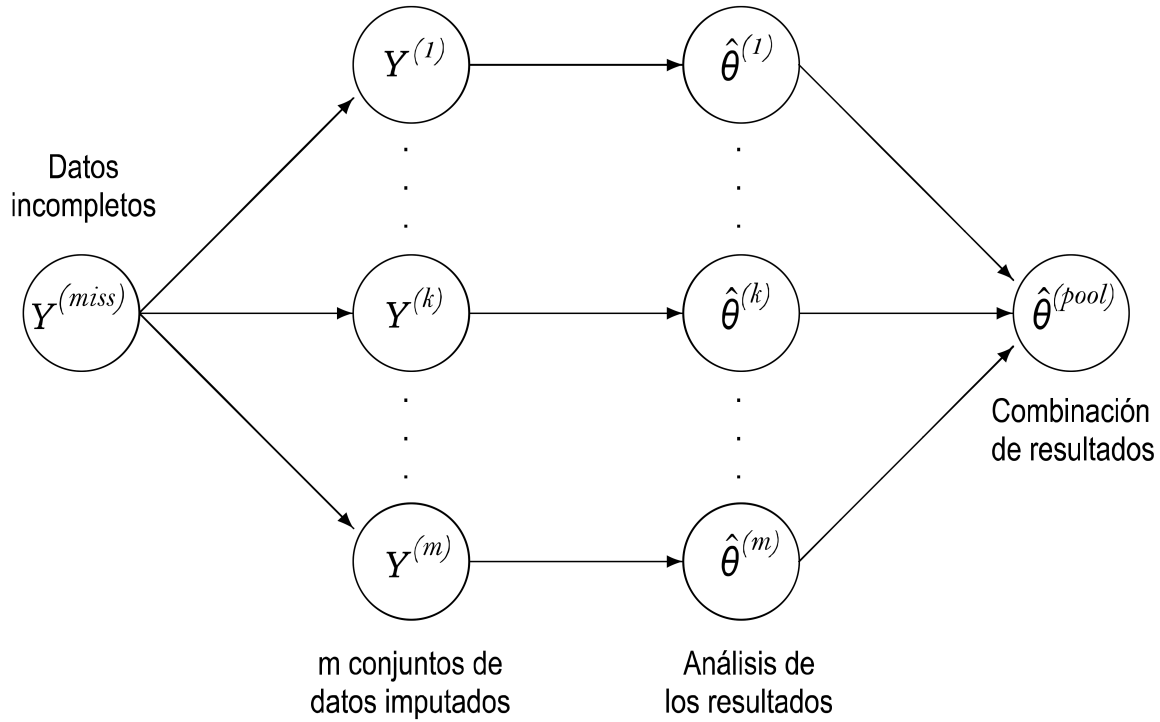
Algoritmo MICE

```r
library(mice)

# Proceso de imputación
proc.imp = mice(datos, # conjunto de datos con faltantes
  m = 5, # número de imputaciones
  maxit = 15, # número de iteraciones
  defaultMethod = c("pmm", "logreg", "polyreg", "polr"), # métodos
  seed = 12345, # semilla
  printFlag = FALSE
)

# Conjuntos de datos imputados
complete(proc.imp, 1)
complete(proc.imp, 2)
complete(proc.imp, 3)
complete(proc.imp, 4)
complete(proc.imp, 5)
```

Análisis estadístico

Estimación

Flujo de trabajo



Combinación de resultados (Reglas de Rubin)

- Dado el estimador $\hat{\boldsymbol{\theta}}$ de la cantidad poblacional $\boldsymbol{\theta}$ $(q \times 1)$, con

$$E\left(\hat{\boldsymbol{\theta}}\right) = \boldsymbol{\theta}, \quad V\left(\hat{\boldsymbol{\theta}}\right) = \Sigma$$

  y sea $\hat{\Sigma}$ el estimador de Σ, tal que $E\left(\hat{\Sigma}\right) \geq V\left(\hat{\boldsymbol{\theta}}\right)$

- Para cada conjunto de datos imputado calculamos

$$\begin{aligned} \hat{\boldsymbol{\theta}}^{(k)}: &\quad \hat{\boldsymbol{\theta}}^{(1)}, \hat{\boldsymbol{\theta}}^{(2)}, \ldots, \hat{\boldsymbol{\theta}}^{(m)} \\ \hat{\Sigma}^{(k)}: &\quad \hat{\Sigma}^{(1)}, \hat{\Sigma}^{(2)}, \ldots, \hat{\Sigma}^{(m)} \end{aligned}$$

Combinación de resultados (Reglas de Rubin)

- Estimación combinada de $\boldsymbol{\theta}$

$$\bar{\boldsymbol{\theta}} = \frac{1}{m} \sum_{k=1}^{m} \hat{\boldsymbol{\theta}}^{(k)}$$

- Varianza estimada de $\bar{\boldsymbol{\theta}}$:

$$\hat{V}\left(\bar{\boldsymbol{\theta}}\right) = \bar{\Sigma} + \left(1 + \frac{1}{m}\right) \boldsymbol{B}$$

donde $\bar{\Sigma} = \frac{1}{m} \sum_{k=1}^{m} \hat{\Sigma}^{(k)}$ y

$$\boldsymbol{B} = \frac{1}{m-1} \sum_{k=1}^{m} \left(\hat{\boldsymbol{\theta}}^{(k)} - \bar{\boldsymbol{\theta}}\right)\left(\hat{\boldsymbol{\theta}}^{(k)} - \bar{\boldsymbol{\theta}}\right)^{\top}$$

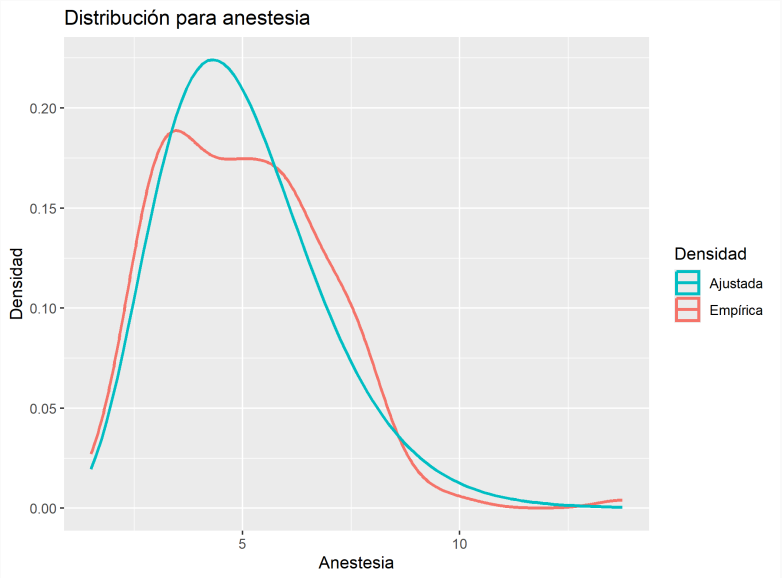
Ajuste de una distribución

Estimaciones

	Imp-1	Imp-2	Imp-3	Imp-4	Imp-5
shape	7.00	7.11	7.09	7.06	6.95
rate	1.39	1.41	1.42	1.40	1.38

Combinación de resultados

	Estimación	Var	B	V. Comb
shape	7.04	0.551	0.004	0.556
rate	1.40	0.023	0.000	0.024




```
library(MASS, exclude = "select")
# Estimaciones por imputación
the.imp = with(proc.imp, fitdistr(Anestesia, "gamma"))
# Combinación de resultados
the.comb = pool(the.imp)
```

Evaluación de los resultados

Evaluación de resultados

Medidas de severidad

- Proporción de la varianza debida a los datos faltantes:

$$\lambda = (1 + m^{-1}) q^{-1} \text{tr} \left(\boldsymbol{B} \left[\hat{V} \left(\bar{\boldsymbol{\theta}} \right) \right]^{-1} \right)$$

- Incremento relativo de la varianza

$$r = (1 + m^{-1}) q^{-1} \text{tr} \left(\boldsymbol{B} \bar{\Sigma}^{-1} \right)$$

- Fracción de información faltante sobre θ

$$\gamma = (1 + r)^{-1} \left[r + 2(\nu + 3)^{-1} \right]$$

con $v = (m - 1)\left(1 + r^{-1}\right)^2$ grados de libertad

Evaluación de resultados

Medidas de severidad

Combinación de resultados

	m	Estimación	Var	B	V. Comb	df	v	r	lambda	gamma
shape	5	7.04	0.551	0.004	0.556	170	166	0.010	0.009	0.021
rate	5	1.40	0.023	0.000	0.024	170	163	0.017	0.017	0.028

```
library(MASS, exclude = "select")
# Estimaciones por imputación
the.imp = with(proc.imp, fitdistr(Anestesia, "gamma"))
# Combinación de resultados
the.comb = pool(the.imp)
```

Corrección de los grados de libertad

- Sea θ un parámetro escalar y $\hat{\theta}$ su estimador, tal que

$$(\theta - \hat{\theta}) \sim N(0, \Sigma)$$

- Un intervalo del $100(1-\alpha)\%$ de confianza para θ es

$$\left(\bar{\theta} - t_{\alpha/2,v}\sqrt{\hat{V}\left(\bar{\theta}\right)}, \bar{\theta} + t_{\alpha/2,v}\sqrt{\hat{V}\left(\bar{\theta}\right)}\right)$$

  con $v = (m-1)\left(1 + r^{-1}\right)^2$ grados de libertad.

- El nivel de significancia asociado con el valor θ_0 está dado por

$$Pr\left(F_{1,v} > \left(\theta_0 - \bar{\theta}\right)^2 / \hat{V}(\bar{\theta})\right)$$

Comparación de los tiempos medios de perfusión

- Se desea comparar los tiempos medios de perfusión entre los paciente con cardiopatía congénita y coronária
- Test de Levene para la comparación de varianzas

```
library(miceafter)

proc.imp.ml <- mids2milist(proc.imp)
lev.imp = with(proc.imp.ml, expr=levene_test(Perfusión ~ Tipo))
pool_levenetest(lev.imp, method="D2")
```

```
##  D2.numdf   p.numdf df1.numdf df2.numdf
##  9.29e+00  2.35e-03  1.00e+00  1.65e+03
## attr(,"class")
## [1] "mipool"
```

Comparación de los tiempos medios de perfusión

- t.test para la comparación de medias

```
t.imp = with(proc.imp.ml,
  expr = t_test(Perfusión ~ Tipo, var_equal=FALSE, paired=FALSE))
 pool_t_test(t.imp, statistic=TRUE)
```

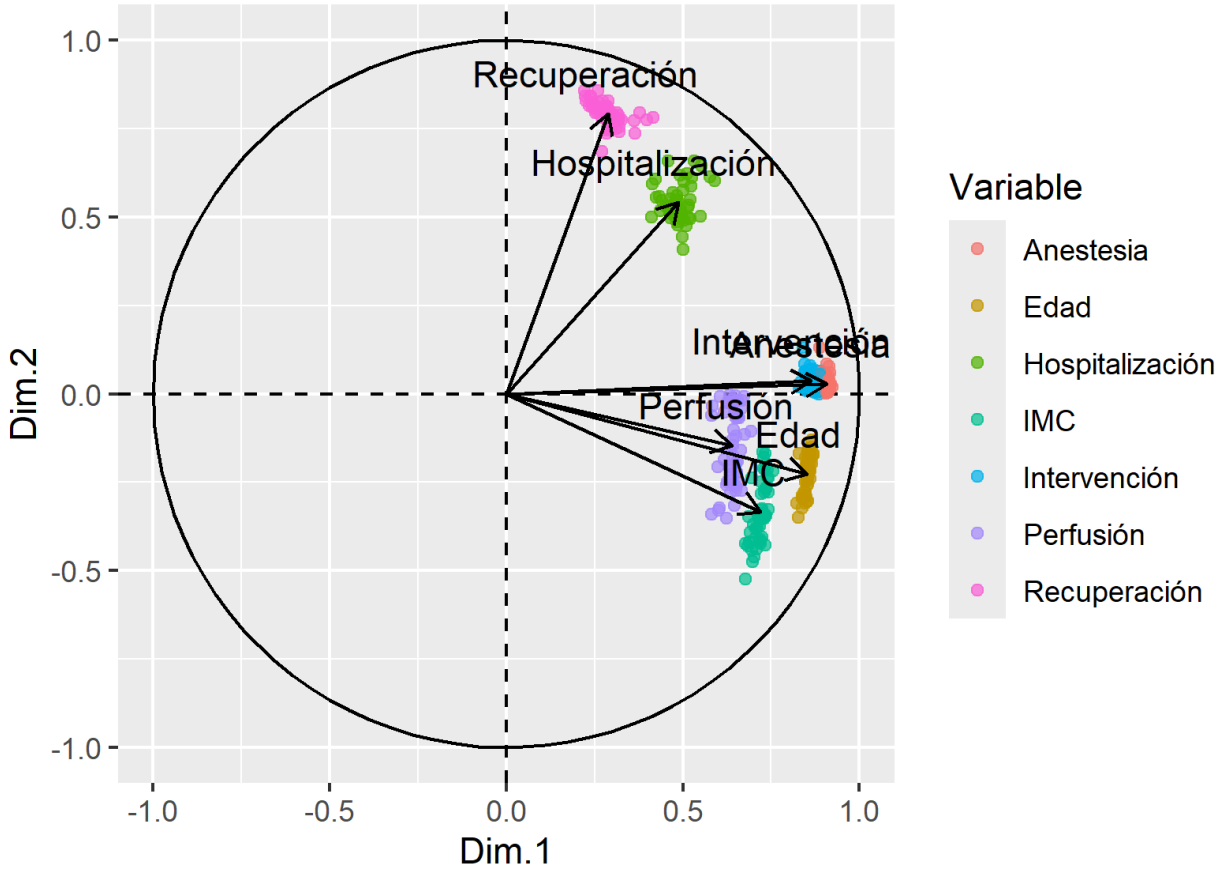
```
##      Mean diff    SE t 95 CI low 95 CI high statistic     pval
## [1,]    -0.712 0.118 2     -0.95     -0.475     -6.01 1.45e-07
## attr(,"class")
## [1] "mipool"
```

Análisis de componentes principales

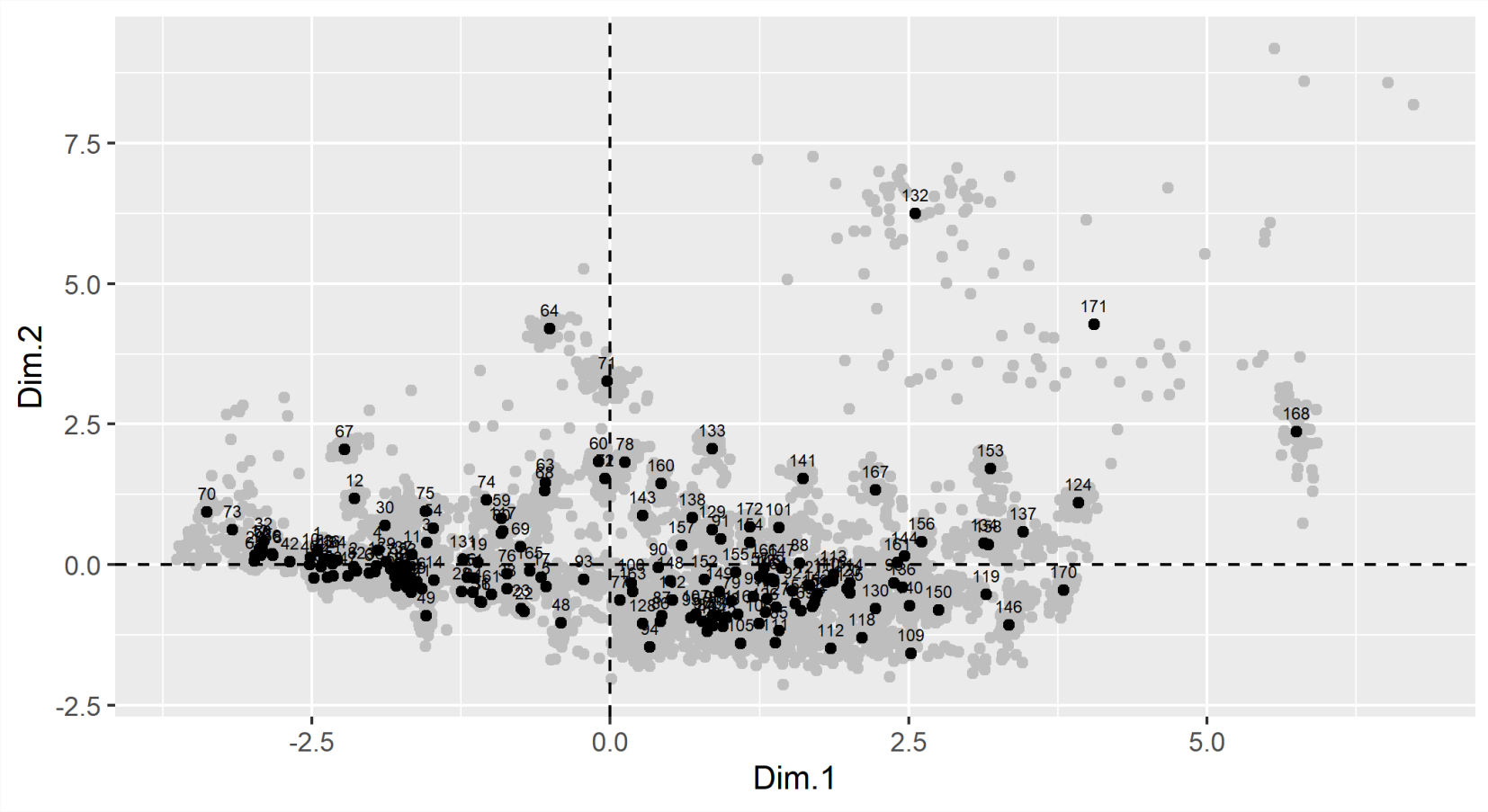
- Aplicamos el algoritmo MICE
- Calculamos un ACP para cada base imputada
- Combinamos los resultados de los m ACP

Valores propios (m = 50)			
Comp.	**eigen**	**% var.**	**% acum.**
1	3.555	50.79	50.8
2	1.132	16.18	67.0
3	0.761	10.88	77.8
4	0.725	10.36	88.2
5	0.449	6.41	94.6
6	0.235	3.36	98.0
7	0.143	2.04	100.0

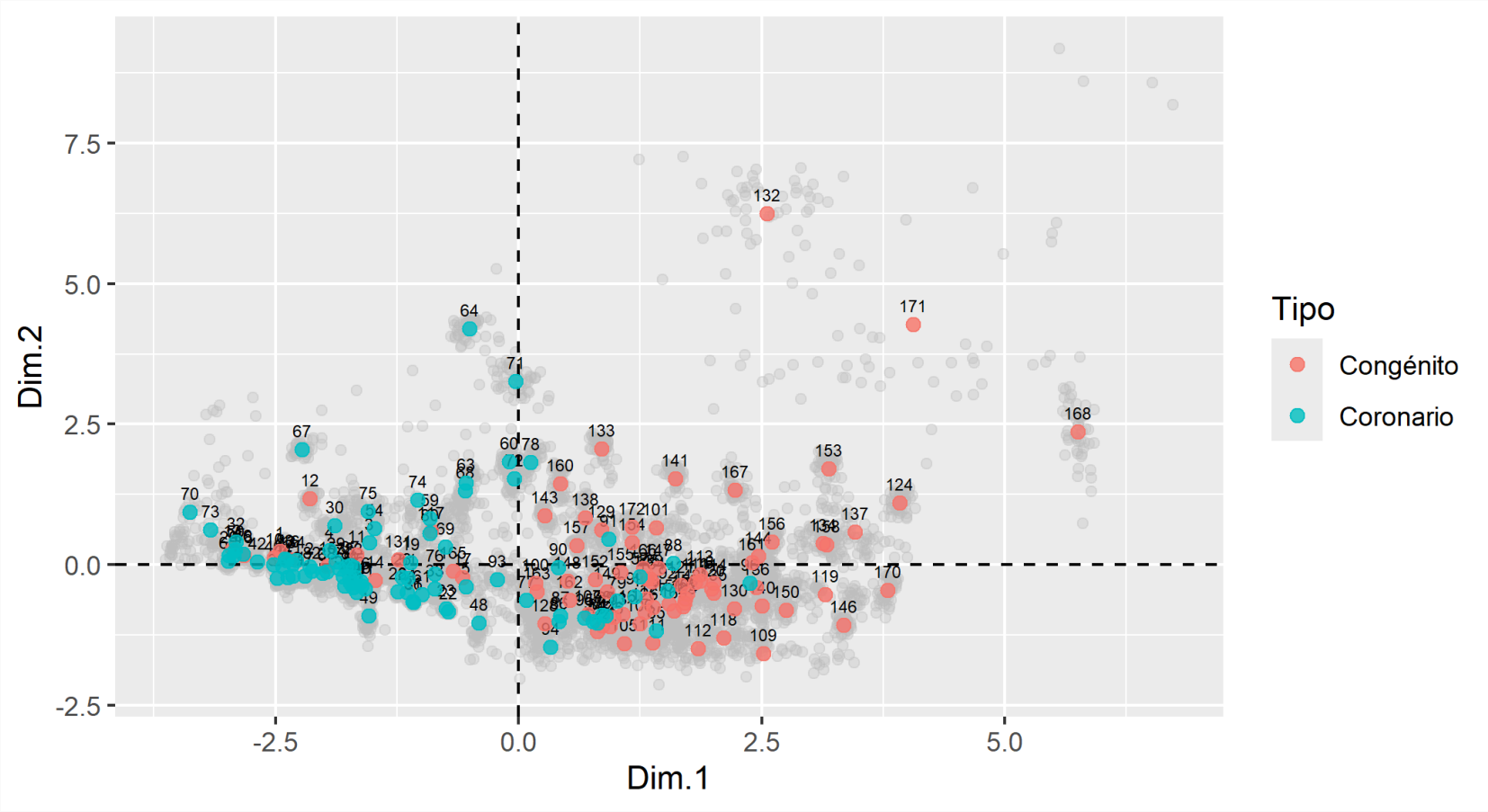
Análisis de componentes principales



Análisis de componentes principales



Análisis de componentes principales



Análisis de componentes principales

```
# Imputación
proc.imp = mice(
  datos, m = 50, maxit = 15, seed = 123, printFlag = FALSE,
  defaultMethod = c("pmm", "logreg", "polyreg", "polr")
)

library(FactoMineR)
library(factoextra)

# ACP con cada base imputada
acp.m = with(proc.imp,
  PCA(data.frame(
        Sexo, Tipo, Protocolo, # ilustrativas
        Edad, IMC, Hospitalización, Intervención,
        Anestesia, Perfusión, Recuperación),
      quali.sup = 1:3,
      graph = FALSE))

# Resultados a combinar
acp.m$analyses
```

Agrupamiento

- Aplicamos el algoritmo MICE
- Realizamos un agrupamiento para cada base imputada
- Combinamos los resultados de los m agrupamientos
- El número ótimo de grupos es seleccionado maximizando la estadística

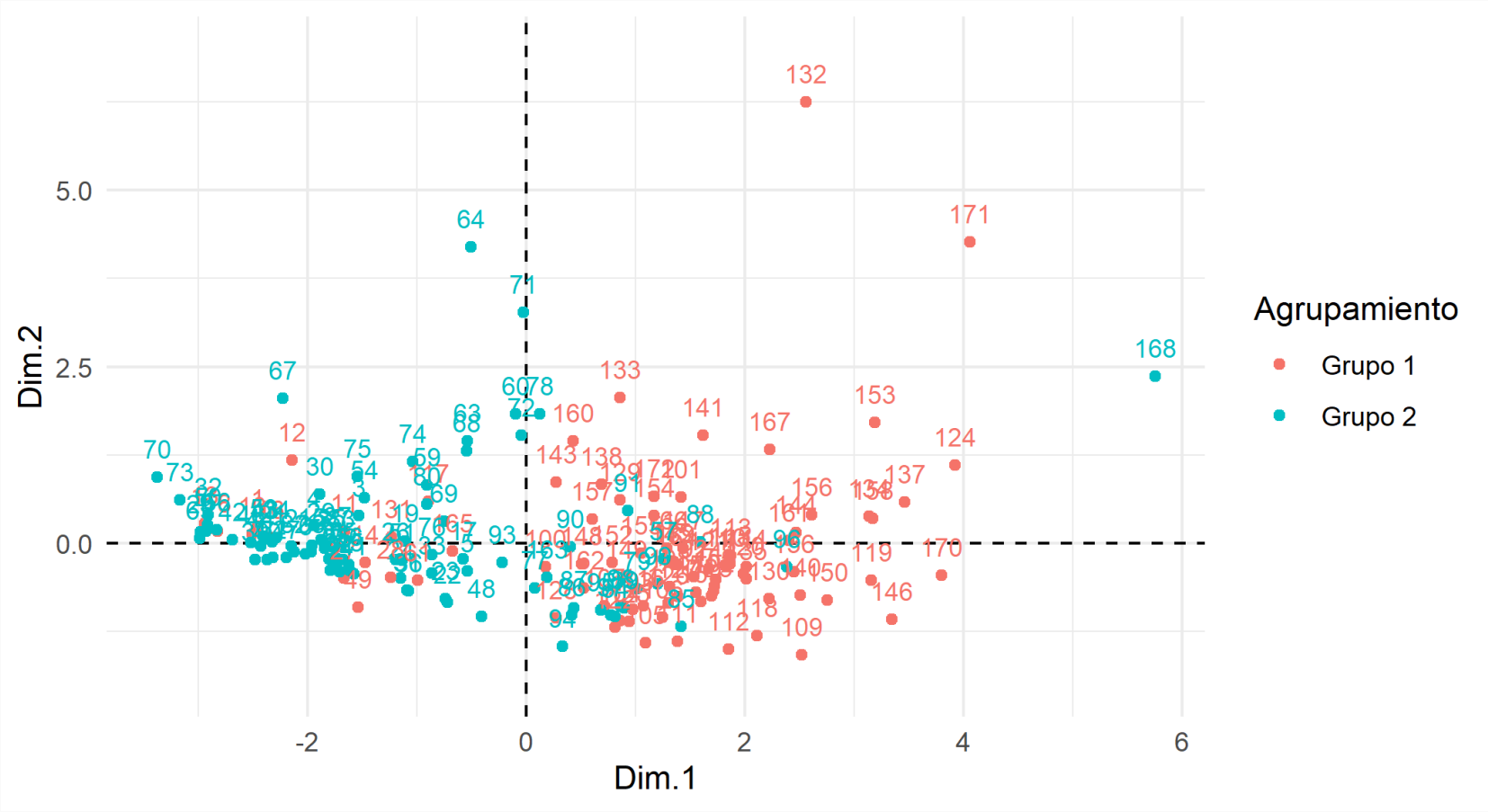
$$CritCF = \left(\frac{2p}{2p+1} \frac{1}{1+D/E} \right)^{\frac{1+log_2(k+1)}{1+log_2(p+1)}}$$

con D y E medidas de inercia dentro y entre grupos

Agrupamiento

Agrupamientos (k = 2, m = 50)										
	imp2	**imp11**	**imp14**	**imp16**	**imp24**	**imp26**	**imp28**	**imp29**	**imp32**	**imp46**
10	1	1	1	1	1	1	1	1	1	1
30	1	1	1	1	1	1	1	1	1	1
38	1	1	1	1	1	1	1	1	1	1
40	1	1	1	1	1	1	1	1	1	1
75	1	1	1	1	1	1	1	1	1	1
86	2	1	2	2	2	2	2	1	2	2
94	2	1	1	2	1	2	1	1	2	1
103	2	1	2	2	2	2	2	1	2	2
160	2	1	2	2	2	2	2	1	2	2
170	2	1	2	2	2	2	2	1	2	2

Agrupamiento



Agrupamiento

```
library(miclust)

mc = miclust(datos.imp,
             ks = 2:5, # número de grupos
             method = "kmeans", # método
             initcl = "hc" # jerárquico + kmeans
)

summary(mc)
```

Modelos de regresión

Variables y modelo

- Variable de respuesta: Hospitalización (en días)
- Variables predictoras: Edad, Sexo, IMC, Intervención, Anestesia, Perfusión, Recuperación, Tipo, Protocolo
- Modelo: lineal generalizado Poisson con enlace logarítmico

Ajuste del modelo

- Aplicamos el algoritmo MICE
- Estimamos los parámetros del modelo para cada base imputada
- Combinamos los resultados de los m modelos

Ajuste del modelo

Coeficientes (m = 5)					
	Imp. 1	**Imp. 2**	**Imp. 3**	**Imp. 4**	**Imp. 5**
(Intercept)	1.564	1.611	1.652	1.207	1.084
Edad	0.002	0.001	0.000	0.003	0.001
SexoMasculino	-0.131	-0.138	-0.131	-0.011	-0.015
IMC	0.006	-0.002	0.003	0.016	0.008
Intervención	0.076	0.098	0.129	-0.027	-0.005
Anestesia	-0.042	-0.014	-0.076	0.032	0.074
Perfusión	0.022	0.033	0.098	0.070	0.174
Recuperación	0.004	0.000	0.001	0.005	0.005
TipoCoronario	0.149	0.269	0.335	0.049	-0.005
ProtocoloAcelerada	0.833	0.816	0.768	0.801	0.729

Combinación de resultados

Coeficientes y medidas de severidad

	m	Estimado	Var	B	V. Comb	df	v	r	lambda	gamma
(Intercept)	5	1.424	0.014	0.067	0.095	162	4.50	5.669	0.850	0.890
Edad	5	0.001	0.000	0.000	0.000	162	33.69	0.407	0.289	0.328
SexoMasculino	5	-0.085	0.003	0.004	0.008	162	7.53	2.070	0.674	0.736
IMC	5	0.006	0.000	0.000	0.000	162	8.97	1.612	0.617	0.681
Intervención	5	0.054	0.000	0.004	0.006	162	3.47	11.353	0.919	0.944
Anestesia	5	-0.005	0.001	0.004	0.005	162	4.00	7.686	0.885	0.918
Perfusión	5	0.079	0.001	0.004	0.005	162	4.80	4.839	0.829	0.873
Recuperación	5	0.003	0.000	0.000	0.000	162	3.47	11.298	0.919	0.944
TipoCoronario	5	0.159	0.011	0.021	0.036	162	7.33	2.157	0.683	0.745
ProtocoloAcelerada	5	0.789	0.003	0.002	0.005	162	16.14	0.818	0.450	0.507

Modelos de regresión

Inferencias

Coeficientes y medidas de severidad					
	Estimado	**ee**	**estadística**	**df**	**valor p**
(Intercept)	1.424	0.308	4.620	4.50	0.007
Edad	0.001	0.002	0.564	33.69	0.577
SexoMasculino	-0.085	0.088	-0.965	7.53	0.364
IMC	0.006	0.009	0.687	8.97	0.509
Intervención	0.054	0.077	0.709	3.47	0.523
Anestesia	-0.005	0.069	-0.076	4.00	0.943
Perfusión	0.079	0.073	1.081	4.80	0.331
Recuperación	0.003	0.003	1.132	3.47	0.330
TipoCoronario	0.159	0.190	0.838	7.33	0.428
ProtocoloAcelerada	0.789	0.068	11.610	16.14	0.000

Ajuste del modelo

```
# Imputación
proc.imp = mice(
  datos, m = 5, maxit = 15, seed = 123, printFlag = FALSE,
  defaultMethod = c("pmm", "logreg", "polyreg", "polr")
)

# Modelos ajustados con cada imputación
mod.imp = with(proc.imp,
  glm(Hospitalización ~ Edad + Sexo + IMC + Intervención +
        Anestesia + Perfusión + Recuperación + Tipo + Protocolo,
      family = poisson(link = "log")))

# Combinación de resultados
comb = pool(mod.imp) ; comb

# Inferencias
summary(comb)
```

Regresión paso a paso

Coeficientes y medidas de severidad										
	m	Estimado	Var	B	V. Comb	df	v	r	lambda	gamma
(Intercept)	5	1.445	0.007	0.091	0.117	164	3.11	15.379	0.939	0.959
Edad	2	0.003	0.000	0.000	0.000	164	150.82	0.019	0.019	0.031
SexoMasculino	3	-0.139	0.002	0.000	0.002	164	145.23	0.033	0.032	0.045
Intervención	3	0.100	0.000	0.001	0.001	164	3.68	2.429	0.708	0.796
Anestesia	3	-0.013	0.000	0.006	0.008	164	1.80	16.838	0.944	0.967
Recuperación	3	0.005	0.000	0.000	0.000	164	4.04	2.083	0.676	0.768
TipoCoronario	3	0.266	0.006	0.010	0.020	164	3.82	2.275	0.695	0.784
ProtocoloAcelerada	5	0.793	0.002	0.002	0.004	164	17.91	0.738	0.425	0.480
Perfusión	3	0.117	0.001	0.003	0.005	164	2.52	5.457	0.845	0.901
IMC	2	0.014	0.000	0.000	0.000	164	4.78	0.802	0.445	0.588

Inferencias

Coeficientes y medidas de severidad					
	Estimado	**ee**	**estadística**	**df**	**valor p**
(Intercept)	1.445	0.342	4.229	3.11	0.023
Edad	0.003	0.002	2.108	150.82	0.037
SexoMasculino	-0.139	0.050	-2.783	145.23	0.006
Intervención	0.100	0.037	2.712	3.68	0.059
Anestesia	-0.013	0.090	-0.148	1.80	0.898
Recuperación	0.005	0.001	3.903	4.04	0.017
TipoCoronario	0.266	0.141	1.885	3.82	0.136
ProtocoloAcelerada	0.793	0.065	12.143	17.91	0.000
Perfusión	0.117	0.073	1.604	2.52	0.224
IMC	0.014	0.006	2.250	4.78	0.077

Regresión paso a paso

```
# Imputación
proc.imp = mice(
  datos, m = 5, maxit = 15, seed = 123, printFlag = FALSE,
  defaultMethod = c("pmm", "logreg", "polyreg", "polr")
)

# Modelos ajustados con cada imputación
mod.imp = with(proc.imp,
  step(glm(Hospitalización ~ Edad + Sexo + IMC + Intervención +
        Anestesia + Perfusión + Recuperación + Tipo + Protocolo,
      family = poisson(link = "log"))))

# Combinación de resultados
comb = pool(mod.imp) ; comb

# Inferencias
summary(comb)
```

Algunas referencias

Libros

- Little, R. J. A., Rubin, D. B. (2020) Statistical Analysis with Missing Data, 3rd Ed. Wiley
- McKnight, P.E., McKnight, K. M., Sidani, S., Figueredo, A. J. (2007) Missing Data. The Guilford Press
- Raghunathan, T. (2016) Missing Data Analysis in Practice. CRC Press
- van Buuren, S. (2018). Flexible Imputation of Missing Data, 2nd Ed. Chapman & Hall

Software

- CRAN Task View: Missing Data (https://cran.r-project.org/web/views/MissingData.html)
- Librerías de R y modulos de Python: https://rmisstastic.netlify.app/rpkg/
- Algoritmo MICE en Python: https://pypi.org/project/miceforest/