ACTIVIDAD

DATA: COVID19_DIABETES

Integrantes:

- Llancari Nivin Meyli
- Vera Fonseca July
- Alejo Huamán Melissa
- Ccuro Minaya Lucia
- Flores Diaz, Christian

Instalar y cargar los paquetes

Instalamos y cargamos los paquetes necesarios para realizar la imputación y análisis de datos. Datos perdidos en investigación en salud

```{r}
# Instalar paquetes si no están instalados
if (!requireNamespace("mice", quietly = TRUE)) install.packages("mice")
if (!requireNamespace("ggmice", quietly = TRUE)) install.packages("ggmice")
if (!requireNamespace("tidyverse", quietly = TRUE)) install.packages("tidyverse")
if (!requireNamespace("here", quietly = TRUE)) install.packages("here")
if (!requireNamespace("rio", quietly = TRUE)) install.packages("rio")
if (!requireNamespace("gtsummary", quietly = TRUE)) install.packages("gtsummary")

# Cargar paquetes
library(mice)
library(ggmice)
library(tidyverse)
library(here)
library(rio)
library(gtsummary)
# cargar el dataset
data <- read_csv("C:/Users/Franco Rodrigo/Desktop/DATA/covid_19_diabetes.csv")
```

```
Rows: 686 Columns: 85
— Column specification
Delimiter: ","
chr (60): pac_fue_hospital, desenla_fallecido, edad, raza_negra, raza_blanca,
asiatico, latino, infacto_m...
dbl (25): Derivation.cohort, duraci_hospita_diaz, severidad, Edad, Puntuación_edad,
Saturación_O2, Temper...

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

En investigaciones en salud, como estudios sobre pacientes con COVID-19 y diabetes, los datos faltantes son frecuentes debido a la falta de registro en historias clínicas o a la omisión de ciertas pruebas por parte del personal médico. Por ejemplo, los niveles de glucosa o ferritina pueden no estar disponibles para todos los pacientes. La práctica común de eliminar observaciones con datos faltantes (análisis de casos completos) puede introducir sesgos y reducir la potencia estadística, especialmente en datasets pequeños.

2 Imputación de datos

La imputación de datos permite aprovechar todas las observaciones disponibles, mejorando la precisión de los análisis. En este ejercicio, utilizaremos la imputación múltiple mediante el paquete `mice` en R, una técnica avanzada que genera múltiples conjuntos de datos imputados para reflejar la incertidumbre asociada a los valores faltantes, superando métodos más simples como el reemplazo por la media.

3 El dataset para este ejercicio

Utilizaremos un conjunto de datos ficticio sobre 686 pacientes con COVID-19 y diabetes. Este dataset incluye variables como edad (en años), severidad de la enfermedad (leve, moderada, grave), raza (negra, blanca, asiática, latina), glucosa (mg/dL), ferritina (ng/mL) y desenlace (fallecido o no). Algunas de estas variables presentan valores faltantes.

```{r}
# cargar el dataset
data <- read_csv("C:/Users/Franco Rodrigo/Desktop/DATA/covid_19_diabetes.csv")
# observamos los datos:
head(data)
```

A tibble: 6 × 85

Derivation.cohort <dbl>	pac_fue_hospital <chr>	duraci_hospita_diaz <dbl>
1	Sí	15
1	Sí	14
1	Sí	11
1	Sí	1
1	Sí	3
1	Sí	26

6 rows | 1-3 of 85 columns

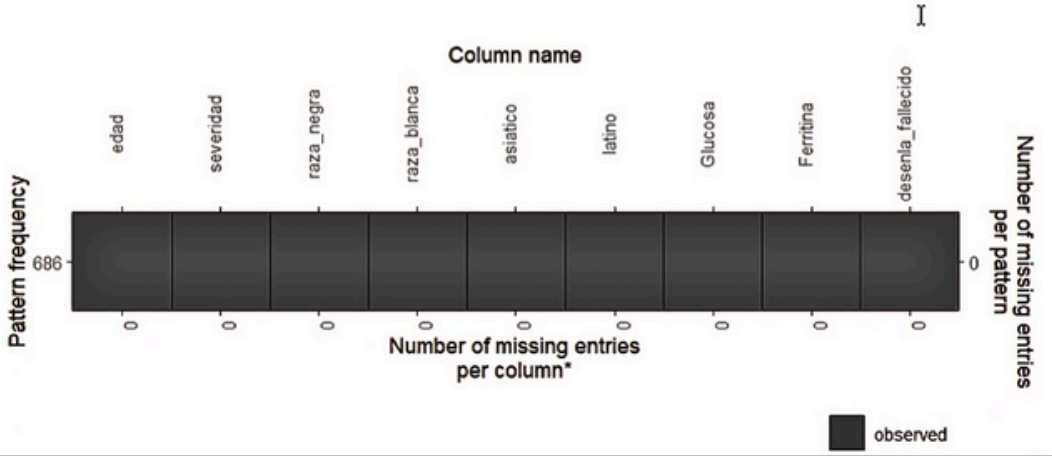
4 Realizando la imputación de datos

4.1 ¿Dónde están los valores perdidos?

Primero, identificamos las variables con datos faltantes usando `colSums()` y visualizamos los patrones de pérdida con `plot_pattern()` del paquete `ggmice`.4.2 Comparación de participantes con y sin valores perdidos

```
# Contar valores NA por columna
colSums(is.na(data))

# Visualizar patrones de datos faltantes
data %>%
  select(edad, severidad, raza_negra, raza_blanca, asiatico, latino, Glucosa, Ferritina,
desenla_fallecido) %>%  # Corregidos "glucosa" a "Glucosa" y "ferritina" a "Ferritina"
  ggmice::plot_pattern(square = TRUE, rotate = TRUE)
```



Comparación de participantes con y sin valores perdidos

Comparamos las características de los pacientes con y sin valores perdidos en la variable `glucosa` para evaluar si la imputación es necesaria.

```
{r}
tabla_glucosa <- data %>%
  select(edad, severidad, raza_negra, raza_blanca, asiatico, latino, Glucosa, Ferritina,
desenla_fallecido) %>%
  mutate(missing = factor(is.na(Glucosa), levels = c(FALSE, TRUE), labels = c("Sin NA",
"Con NA"))) %>%
  tbl_summary(by = missing, statistic = list(all_continuous() ~ "{mean} ({sd})",
all_categorical() ~ "{n} ({p}%)")) %>%
  modify_header(label = "**Variable**", all_stat_cols() ~ "**{level}**<br>N = {n}
({style_percent(p, digits=1)}%)") %>%
  modify_caption("Características según valores perdidos en Glucosa") %>%
  bold_labels()

# Mostrar la tabla
tabla_glucosa
```

Características según valores perdidos en Glucosa		
Variable	**Sin NA** N = 686 (100.0%)[1]	**Con NA** N = 0 (0%)[1]
edad		
>60	178 (26%)	0 (NA%)
>70	164 (24%)	0 (NA%)
>80	102 (15%)	0 (NA%)
0-60	242 (35%)	0 (NA%)
severidad	3.61 (2.21)	NA (NA)
raza_negra		
No	424 (62%)	0 (NA%)

Si las diferencias son significativas, la imputación será preferible al análisis de casos completos.

4.3 ¿Qué variables debo incluir en el proceso de imputación?

Incluimos todas las variables relevantes para los análisis posteriores, incluso aquellas sin valores perdidos, para que el modelo de imputación sea robusto. Las variables categóricas deben convertirse a factores.

```r
{r}
input_data <- data %>%
  select(edad, severidad, raza_negra, raza_blanca, asiatico, latino, Glucosa, Ferritina,
desenla_fallecido) %>%  # Corregidos "glucosa" a "Glucosa" y "ferritina" a "Ferritina"
  mutate(severidad = as.factor(severidad),
         raza_negra = as.factor(raza_negra),
         raza_blanca = as.factor(raza_blanca),
         asiatico = as.factor(asiatico),
         latino = as.factor(latino),
         desenla_fallecido = as.factor(desenla_fallecido))
```

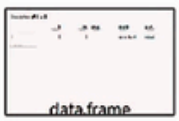
4.4 La función `mice()` para imputar datos

Usamos `mice()` para imputar los datos, especificando el número de imputaciones (`m`), una semilla para reproducibilidad, y métodos de imputación según el tipo de variable.

```r
{r}
# Definir métodos de imputación
method_vector <- rep("", ncol(input_data))
method_vector[names(input_data) %in% c("Glucosa", "Ferritina")] <- "pmm"  # Continuas
method_vector[names(input_data) %in% c("severidad", "raza_negra", "raza_blanca",
"asiatico", "latino", "desenla_fallecido")] <- "logreg"  # Categóricas binarias o
politómicas

# Realizar imputación
data_imputada <- mice(input_data, m = 20, method = method_vector, maxit = 20, seed = 123
, print = FALSE)

# Resumen del objeto imputado
data_imputada
```



Description: **df [1 × 5]**

	it	im	dep	meth	out

Description: df [1 × 5]

	it <dbl>	im <dbl>	dep <chr>	meth <chr>	out <chr>
1	0	0		constant	edad

1 row

5 Analizando los datos imputados

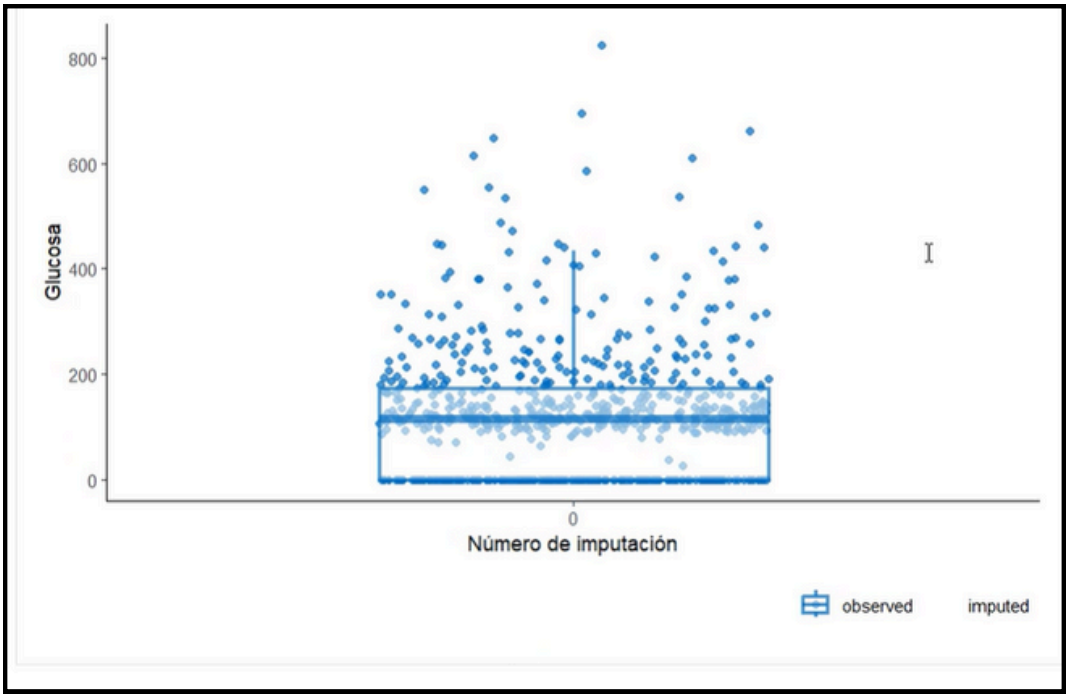
Verificamos la plausibilidad de los valores imputados comparándolos con los observados.

```{r}
# Visualizar imputación de la variable Glucosa
ggmice(data_imputada, aes(x = .imp, y = Glucosa)) +
  geom_jitter(height = 0, width = 0.25) +
  geom_boxplot(width = 0.5, size = 1, alpha = 0.55, outlier.shape = NA) +
  labs(x = "Número de imputación")

# Para la variable desenla_fallecido:

data_imputada_l <- complete(data_imputada, "long", include = TRUE)
data_imputada_l <- data_imputada_l %>%
  mutate(imputed = .imp > 0,
         imputed = factor(imputed, levels = c(FALSE, TRUE), labels = c("Observado",
"Imputado")))

# Tabla de proporciones para desenla_fallecido e imputación
prop.table(table(data_imputada_l$desenla_fallecido, data_imputada_l$imputed), margin = 2
)
```



Los valores imputados deben ser coherentes con los observados en términos de rango y distribución.

5.1 Procedimientos adicionales luego de la imputación

Realizamos una regresión logística para evaluar el efecto de las variables sobre el desenlace desenla_fallecido, usando los datos imputados.

```{r}
# Modelo de regresión logística
modelo <- with(data_imputada, glm(desenla_fallecido ~ edad + severidad + Glucosa +
Ferritina, family = binomial(link = "logit")))  # Corregido "glucosa" y "ferritina"

# Presentar resultados con gtsummary
tabla_resultados <- tbl_regression(modelo, exponentiate = TRUE) %>%
  bold_p(t = 0.05) %>%
  modify_header(estimate = "**OR ajustado**", p.value = "**p valor**")

# Mostrar la tabla
tabla_resultados
```

Characteristic	OR ajustado	95% CI	p valor
edad			
>60	—	—	
>70	1.07	0.63, 1.82	0.8

Characteristic	OR ajustado	95% CI	p valor
edad			
>60	—	—	
>70	1.07	0.63, 1.82	0.8
>80	0.97	0.52, 1.80	>0.9
0-60	0.68	0.38, 1.22	0.2
severidad			
0	—	—	
1	4.27	0.49, 37.3	0.2
2	3.45	0.42, 28.4	0.2
3	7.62	0.95, 61.1	0.056

Este análisis combina automáticamente los resultados de las 20 imputaciones, proporcionando estimaciones ajustadas y sus intervalos de confianza.

Este documento es reproducible y adaptable a otros datasets ajustando las variables y rutas de archivo según sea necesario.