



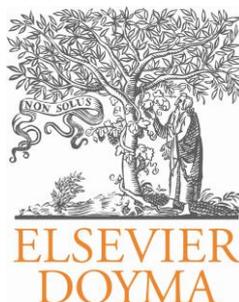
UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

MÓDULO 6:

Apartado 3

Simulación del cálculo de n

Preparado por Laura Riba



Simulación realizada a partir del “Protocolo de ensayo clínico de paliperidona frente a risperidona inyectable de larga duración en psicosis de nuevo diagnóstico” expuesto en el informe del Módulo 6, “Resumen de un protocolo con SPIRIT”.

SIMULACIÓN de un ECA

Nota: La simulación propuesta utiliza distribuciones normales, ya que permiten determinar la correlación desde el principio. Sin embargo, somos conscientes de que las distribuciones normales multivariantes no son un componente del curso de BNE. Para poder generar este tipo de distribuciones, hay que definir la matriz Σ de varianzas y covarianzas, y usar un paquete especial de R (mvtnorm).

Para el caso en que la correlación basal – final es 0.5, el tamaño muestral es de 252 pacientes por rama si se quiere obtener una potencia del 80%. Simularemos para cada paciente el valor basal y final de la escala PANSS, teniendo en cuenta que siguen una distribución normal multivariante. Sin considerar aún el efecto del tratamiento, si ‘0’ representa el valor inicial, ‘f’ el final, y $\sigma(0,f)$ su covarianza:

$$X \sim \text{Normal multivariante} \quad \begin{aligned} \mu &= \mu_0, \mu_f \\ \Sigma &= \begin{pmatrix} \sigma_0^2 & \sigma_{0,f} \\ \sigma_{0,f} & \sigma_f^2 \end{pmatrix} \end{aligned}$$

Los valores de los parámetros fijados son:

- $\mu = 60$
- $\Delta = 5$
- $\sigma = 20$
- $\rho = 0.5$

Así pues, la varianza es 400 y la covarianza es:

$$\rho = \frac{\text{cov } Y_0, Y_f}{\sigma_0 \cdot \sigma_f} \rightarrow \text{cov } Y_0, Y_f = \rho \cdot \sigma_0 \cdot \sigma_f = 0.5 \cdot 20 \cdot 20 = 200$$

Por lo tanto, la matriz de varianzas y covarianzas es:

$$\Sigma = \begin{pmatrix} \sigma_0^2 & \sigma_{0,f} \\ \sigma_{0,f} & \sigma_f^2 \end{pmatrix} = \begin{pmatrix} 400 & 200 \\ 200 & 400 \end{pmatrix}$$

Para cada simulación, los datos se guardarán en una matriz Y de **N = 504** filas (cada fila será un paciente) y dos columnas, donde la columna 1 será la PANSS basal y la columna 2 será la PANSS final.

Un vector adicional g contendrá el grupo de asignación, determinado al azar. Creamos los dos grupos de pacientes, según el tratamiento recibido:

- En el grupo 1, suponemos que la media de la escala PANSS al inicio y al final es la misma.

$$X_1 \sim \text{Normal multivariante} \quad \Sigma = \begin{matrix} \mu = \mu, \mu & & \mu = 60, 60 \\ \sigma_0^2 & \sigma_{0,f} & \\ \sigma_{0,f} & \sigma_f^2 & \end{matrix} \quad \Sigma = \begin{matrix} 400 & 200 \\ 200 & 400 \end{matrix}$$

- En el grupo 2, suponemos que la media de la escala PANSS al final es delta unidades mayor que al principio.

$$X_2 \sim \text{Normal multivariante} \quad \Sigma = \begin{matrix} \mu = \mu, \mu + \Delta & & \mu = 60, 65 \\ \sigma_0^2 & \sigma_{0,f} & \\ \sigma_{0,f} & \sigma_f^2 & \end{matrix} \quad \Sigma = \begin{matrix} 400 & 200 \\ 200 & 400 \end{matrix}$$

Por lo tanto, para uno de los tratamientos no hay cambio en la escala PANSS y para el otro tratamiento sí. Queremos ver si podemos detectar esa diferencia de 5 unidades.

Dado que la escala PANSS se mueve en un rango de valores entre 30 y 210, al escoger una $\mu = 60$, es posible que simulemos algún valor por debajo de 30, por lo que truncamos a 30 y ponemos los valores inferiores iguales a 30. También redondeamos las variables, para que den valores enteros (ello da a los valores simulados un aspecto más realista, pero no es un paso necesario, en verdad).

Este proceso lo repetiremos M veces (5000, en nuestra simulación). En cada ejecución, una vez generados los datos, calcularemos el cambio $C = Y[,2] - Y[,1]$ entre la respuesta final y la inicial. Y luego miraremos si hay diferencias entre la media del cambio del grupo 1 y la media del cambio del grupo 2 mediante el test de la t de Student (sin suponer igualdad de varianzas entre los dos grupos).

Así pues, haremos la comparación M , y obtendremos M p-valores. Contaremos cuántos de estos no son significativos y qué proporción del total son, por lo que estamos estimando el valor de β (en cuantos casos no rechazamos la hipótesis nula de que hay diferencias cuando en realidad sí que las hay). O, si lo que contamos es la proporción de p-valores significativos (menores del 5%, en este caso), estaremos estimando la potencia, que debería de ser del 80%.

Aquí partimos de que conocemos el tamaño muestral que da una potencia del 80%. Pero en caso de que la distribución de la variable sea más compleja y no sepamos calcular el tamaño muestral con una fórmula, habría que probar con varios tamaños muestrales (cambiar la N del código) hasta dar con aquella que proporcione una potencia similar a la deseada. Esta es la gran ventaja de la simulación.

Nuestro problema tiene la dificultad añadida de que la variable simulada no es normal, ni sabemos cómo se distribuye. Realmente podríamos llegar a generar valores aleatorios de alguna forma posiblemente más realista que la empleada (tal vez, dando un carácter más asimétrico a la variable PANSS), pero la mayor dificultad estaría en lograr generar pares de valores con cierta correlación. Este punto no es trivial, y se ha obviado.

A continuación se muestra el código de R comentado:

```

### SIMULACIÓN

N = 504      # N° de pacientes
X = matrix(NA, nrow=N, ncol=2) # Matriz donde se guardaran los datos
M = 5000    # N° de simulaciones
F = 0      # Contador de p-valores, inicialmente a 0

# Parámetros
delta = 5
mu = 60
sigma = 20
var = sigma^2
rho = 0.5
# Covariancia
u = rho*sigma*sigma
# Matriz de varianzas y covarianzas
var_covar = matrix(c(var, u, u, var), nrow=2)

# Simulación
for (i in 1:M)
{
  # Asignamos cada paciente (cada fila) a un grupo de tratamiento
  g = sample(c(rep(1,N/2), rep(2, N/2)))
  # Generamos los valores
  X[g==1,] = rmvnorm(n = N/2, mean = c(mu, mu), sigma = var_covar)
  X[g==2,] = rmvnorm(n = N/2, mean = c(mu, mu + delta), sigma =
var_covar)

  # Truncamos y redondeamos
  X[X[,1]<30,1]=30
  X[X[,2]<30,2]=30
  X = round(X)

  # Calculamos el cambio y realizamos el contraste
  C = X[,2]-X[,1]
  T=t.test(C~g) # unequal vars
  if (T$p.value > 0.05) F=F+1
}

# β
F/M

# Potencia
(M-F)/M

```

Si, por ejemplo, la correlación fuera 0.3, esta es una posible salida (la potencia sería insuficiente):

```

> F/M
[1] 0.5594
> (M-F)/M
[1] 0.4406

```