Documentación BLIO: Preparación del dataset

Atrys Health

2026-01-05

Contents

```
library(dplyr)
library(stringr)
```

```r
library(readxl)
library(writexl)
library(caret)
library(tidyr)
library(skimr)
library(readr)
library(lubridate)
library(forcats)
```

Datos

Eliminamos los pacientes de los cuales no tenemos datos de TCR.

```r
# Lista con codigo de pacientes obtenido con script en bash
pacientes_retenidos <- readLines("/home/agombau/modelo_pipeline/procesed_data/new_names.txt")
# Pacientes en el txt que no están en clinic_data
no_en_data <- setdiff(pacientes_retenidos, clinic_full_data$Paciente)
cat("Pacientes del archivo que no se encontraron en clinic_data:\n")
```

```
## Pacientes del archivo que no se encontraron en clinic_data:
```

```r
print(no_en_data)
```

```
##  [1] "PH-001-PU" "PH-182-PU" "PH-160-PU" "PH-031-PU" "IL-009-PU" "PH-105-PU"
##  [7] "PH-145-PU" "PH-193-PU" "PH-114-PU" "PH-203-PU" "LA-004-PU" "FA-007-PU"
## [13] "PH-009-PU" "FA-008-PU" "LE-032-PU" "PH-195-PU" "LE-006-PU" "PH-014-PU"
## [19] "SL-007-PU" "PH-185-PU" "HB-005-PU" "PH-128-PU" "UA-003-PU" "PH-042-PU"
## [25] "PH-072-PU" "PH-083-PU" "PH-152-PU" "PH-163-PU" "IL-010-PU" "PH-184-PU"
## [31] "PH-094-PU" "PH-043-PU" "X-001-PU"
```

```r
# Pacientes en clinic_data que no están en el txt
no_en_txt <- setdiff(clinic_full_data$Paciente, pacientes_retenidos)
cat("Pacientes en clinic_data que no están en el archivo:\n")
```

```
## Pacientes en clinic_data que no están en el archivo:
```

```r
print(no_en_txt)
```

```
##   [1] "CG-034-PU" "FA-022-PU" "PH-204-PU" "PH-208-PU" "LA-030-PU" "FA-020-PU"
##   [7] "SL-010-PU" "PH-211-PU" "PC-011-PU" "CG-035-PU" "FA-014-PU" "FA-023-PU"
##  [13] "FA-024-PU" "FA-027-PU" "LA-031-PU" "LE-037-PU" "LE-038-PU" "LE-039-PU"
##  [19] "SO-009-PU" "PH-216-PU" "PH-217-PU" "PH-219-PU" "PH-221-PU" "PH-222-PU"
##  [25] "PH-224-PU" "PH-225-PU" "PH-226-PU" "PH-227-PU" "PH-229-PU" "PH-230-PU"
##  [31] "PH-231-PU" "PH-232-PU" "PH-233-PU" "PH-234-PU" "PH-236-PU" "PH-237-PU"
##  [37] "PH-238-PU" "PH-239-PU" "PH-241-PU" "PH-244-PU" "PH-248-PU" "PH-250-PU"
##  [43] "PH-251-PU" "PH-253-PU" "PH-254-PU" "PH-255-PU" "PH-257-PU" "PH-258-PU"
##  [49] "PH-259-PU" "PH-260-PU" "PH-261-PU" "PH-262-PU" "PH-263-PU" "PH-264-PU"
##  [55] "PH-265-PU" "PH-266-PU" "PH-267-PU" "PH-268-PU" "PH-270-PU" "PH-271-PU"
##  [61] "PH-272-PU" "PH-273-PU" "PH-274-PU" "PH-275-PU" "PH-276-PU" "PH-277-PU"
##  [67] "PH-278-PU" "PH-279-PU" "SO-007-PU" "LA-018-PU" "UA-031-PU" "PH-209-PU"
##  [73] "LA-016-PU" "UA-015-PU" "PH-179-PU" "CG-031-PU" "PH-032-PU" "SO-006-PU"
##  [79] "CG-029-PU" "PH-178-PU" "UA-026-PU" "HB-010-PU" "UA-039-PU" "HB-007-PU"
##  [85] "SO-014-PU" "PH-183-PU" "PH-131-PU" "PH-070-PU" "LA-021-PU" "UA-035-PU"
##  [91] "PH-002-PU" "LA-006-PU" "SO-004-PU" "PH-177-PU" "PH-187-PU" "LA-012-PU"
##  [97] "PH-007-PU" "CG-032-PU" "PH-240-PU" "LA-024-PU" "PH-067-PU" "PH-053-PU"
## [103] "SO-016-PU" "SO-005-PU" "CG-016-PU" "FA-018-PU" "SO-008-PU" "UA-020-PU"
```

```
## [109] "PH-214-PU" "FA-016-PU" "LA-015-PU" "SO-011-PU" "FA-017-PU" "FA-021-PU"
## [115] "PH-218-PU" "UA-034-PU" "PH-176-PU" "PH-169-PU" "SO-013-PU" "UC-001-PU"
## [121] "PH-154-PU" "PH-200-PU" "UA-036-PU" "PH-206-PU" "UA-009-PU" "PH-205-PU"
## [127] "SO-012-PU" "SL-001-PU" "PH-210-PU" "PH-080-PU" "PH-079-PU" "PH-130-PU"
## [133] "FA-019-PU" "LA-022-PU" "LA-028-PU" "PH-243-PU" "PH-170-PU" "HB-011-PU"
## [139] "PH-181-PU" "SO-015-PU" "PH-173-PU" "PH-088-PU" "LA-020-PU" "UA-018-PU"
## [145] "SO-010-PU" "UA-001-PU" "LA-023-PU" "PH-245-PU" "SL-008-PU" "PH-196-PU"
## [151] "FA-006-PU" "PH-235-PU" "LE-027-PU" "MO-001-PU" "UA-008-PU" "UC-011-PU"
## [157] "UA-030-PU" "PH-194-PU" "UA-016-PU" "UA-023-PU" "CG-028-PU" "PH-020-PU"
## [163] "PH-213-PU" "HB-015-PU" "PH-228-PU" "UA-024-PU" "HB-013-PU" "PH-030-PU"
## [169] "UA-004-PU" "CG-027-PU" "UA-021-PU" "LA-026-PU" "UA-019-PU" "PH-063-PU"
## [175] "SL-002-PU" "CG-033-PU" "IL-004-PU" "UA-014-PU" "PH-220-PU" "HB-012-PU"
## [181] "PH-078-PU" "CG-030-PU" "UA-022-PU" "LA-019-PU" "UA-027-PU" "PH-212-PU"
## [187] "PH-246-PU" "PH-186-PU" "FA-004-PU"
```

```
# Filtramos
clinic_full_data <- clinic_full_data[ clinic_full_data$Paciente %in% pacientes_retenidos, ]
```

```
# Filtramos
clinic_full_data <- clinic_full_data[ clinic_full_data$Paciente %in% pacientes_retenidos, ]
```

Eliminamos los pacientes con registros problemáticos: PH-157-PU, PH-126-PU, LE-011-PU, LE-008-PU, PH-103-PU, LE-013-PU.

```
# Definir vector con los identificadores de pacientes a eliminar
pacientes_eliminar <- c("PH-157-PU", "PH-126-PU", "LE-011-PU", "LE-008-PU", "PH-103-PU", "LE-013-PU")

# Filtrar el dataframe eliminando las filas donde la columna "Pacientes" contenga esos valores
clinic_full_data <- clinic_full_data[ !(clinic_full_data$Paciente %in% pacientes_eliminar), ]
```

```
## El conjunto de datos clinic_full_data contiene:

## Filas (correspondiente a los pacientes): 244

## Columnas (correspondientes a los parametros): 168
```

EDA

```
colnames(clinic_full_data)
```

```
##   [1] "Paciente"
##   [2] "Paciente.fuera.del.estudio"
##   [3] "Motivo.fuera.estudio"
##   [4] "Comentarios"
##   [5] "Fecha.de.inicio.IO"
##   [6] "Progresión...6"
##   [7] "Fecha.de.progresión"
##   [8] "Tipo.de.Progresión"
##   [9] "Evidencia.de.Progresión"
##  [10] "EXITUS"
##  [11] "Fecha.Exitus"
##  [12] "EXITUS...CAUSA.EXITUS"
##  [13] "EXITUS...CAUSA.EXITUS...Otras.causas.Exitus"
##  [14] "Fecha.última.visita"
##  [15] "Progresión...15"
##  [16] "PFS.(d)"
##  [17] "PFS.(m)"
```

```
##  [18] "Exitus"
##  [19] "OS.(d)"
##  [20] "OS.(m)"
##  [21] "Follow.Up.(d)"
##  [22] "FU.(months)"
##  [23] "Fecha.de.nacimiento"
##  [24] "Sexo.(0=Mujer;.1=Varón)"
##  [25] "Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)"
##  [26] "Tumor.previo"
##  [27] "Tipo.de.tumor.previo"
##  [28] "Escala.ECOG"
##  [29] "Fecha.de.diagnóstico"
##  [30] "Edad.al.diagnóstico"
##  [31] "Estadio.al.diagnóstico"
##  [32] "Histología"
##  [33] "EGFR"
##  [34] "EGFR...Qué.mutación.presenta?"
##  [35] "ALK"
##  [36] "ROS1"
##  [37] "RET"
##  [38] "BRAF.(V600)"
##  [39] "KRAS"
##  [40] "KRAS...Qué.mutación.presenta?"
##  [41] "PD-L1"
##  [42] "Fecha.de.diagnóstico.de.enfermedad.metastática"
##  [43] "Quimioterapia.adyuvante"
##  [44] "Quimioterapia.adyuvante...Fecha.inicio"
##  [45] "Quimioterapia.adyuvante...Fecha.fin"
##  [46] "Radioterapia.adyuvante"
##  [47] "Radioterapia.adyuvante...Dosis.de.radioterapia"
##  [48] "Radioterapia.adyuvante...Fecha.inicio"
##  [49] "Radioterapia.adyuvante...Fecha.fin"
##  [50] "RT.QT.Radical"
##  [51] "RT.QT.Radical...Tipo.de.RT.QT"
##  [52] "RT.QT.Radical...Dosis.de.radioterapia"
##  [53] "RT.QT.Radical...Fecha.inicio.RT"
##  [54] "RT.QT.Radical...Fecha.fin.RT"
##  [55] "RT.QT.Radical...Fecha.inicio.QT"
##  [56] "RT.QT.Radical...Fecha.fin.QT"
##  [57] "IO.adyuvante"
##  [58] "IO.adyuvante...Tipo.de.IO"
##  [59] "IO.adyuvante...Fecha.inicio"
##  [60] "IO.adyuvante...Fecha.fin"
##  [61] "Fecha.de.inicio.IO.(metastáticos)"
##  [62] "TIPO.IO.(metastáticos)"
##  [63] "Tipo.de.IO.Cat.(0=IO;.1=ChIO)"
##  [64] "Fármaco.de.IO.(metastáticos)"
##  [65] "Fármaco.de.IO.(metastáticos)...Especificar"
##  [66] "Diana.IO"
##  [67] "Diana.IO...Especificar"
##  [68] "Nº.de.líneas.previas"
##  [69] "Nº.de.líneas.previas...1ª.Línea"
##  [70] "Nº.de.líneas.previas...2ª.Linea"
##  [71] "Nº.de.líneas.previas...3ª.Línea"
```

```
##  [72] "Tejido"
##  [73] "Tejido...Tipo.de.tejido"
##  [74] "Tejido...Microtomía"
##  [75] "Tejido...Microtomía...Fecha.de.envío"
##  [76] "Tejido...RNA"
##  [77] "Tejido...RNA...Fecha.extracción.RNA"
##  [78] "Tejido...RNA...Material.válido"
##  [79] "Tejido...RNA...Material.válido...Envío.a.Atrys"
##  [80] "Tejido...RNA...Material.válido...Envío.a.Atrys...Fecha.envío.RNA"
##  [81] "Tejido...RNA...Material.válido...Envío.a.Atrys...TCR.Secuenciado"
##  [82] "Tejido...IHQ"
##  [83] "BS1.Fecha.1ª.extracción...83"
##  [84] "BS1.Fecha.1ª.extracción...84"
##  [85] "BS1.Fecha.recepción.al.laboratorio"
##  [86] "BS1.Procesamiento"
##  [87] "BS1.Plasma"
##  [88] "BS1.Plasma...Citoquinas"
##  [89] "BS1.Plasma...Extracción.cfDNA"
##  [90] "BS1.Plasma...Extracción.cfDNA...NGS"
##  [91] "BS1.PBMCs"
##  [92] "BS1.PBMCs...Número.de.viales.de.PBMCs"
##  [93] "BS1.PBMCs...Citometría.de.flujo"
##  [94] "BS1.PBMCs...Citometría.de.flujo...Inclusión.en.estadística"
##  [95] "BS1.PBMCs...RNA"
##  [96] "BS1.PBMCs...RNA...Fecha.de.extracción"
##  [97] "BS1.PBMCs...RNA...Material.válido"
##  [98] "BS1.PBMCs...RNA...Material.válido...Envío.Atrys"
##  [99] "BS1.PBMCs...RNA...Material.válido...Envío.Atrys...Fecha.de.envío"
## [100] "BS1.PBMCs...RNA...Material.válido...Envío.Atrys...TCR"
## [101] "BS1.PBMCs...RNA...Material.válido...Envío.Leitat"
## [102] "BS1.PBMCs...RNA...Material.válido...Envío.Leitat...Librería.B"
## [103] "BS1.PBMCs...DNA"
## [104] "BS1.PBMCs...DNA...Fecha.de.extracción"
## [105] "BS1.PBMCs...DNA...Material.válido"
## [106] "BS1.PBMCs...DNA...Material.válido...Envío.Atrys"
## [107] "BS1.PBMCs...DNA...Material.válido...Envío.Atrys...Fecha.de.envío"
## [108] "BS1.PBMCs...DNA...Material.válido...Envío.Atrys...SNPs"
## [109] "BS2"
## [110] "BS2.Fecha.2ª.extracción"
## [111] "BS2.Fecha.recepción.al.laboratorio"
## [112] "BS2.Procesamiento"
## [113] "BS2.Plasma"
## [114] "BS2.Plasma...Citoquinas"
## [115] "BS2.Plasma...Extracción.cfDNA"
## [116] "BS2.Plasma...Extracción.cfDNA...NGS"
## [117] "BS2.PBMCs"
## [118] "BS2.PBMCs...Número.de.viales.de.PBMCs"
## [119] "BS2.PBMCs...Citometría.de.flujo"
## [120] "BS2.PBMCs...Citometría.de.flujo...Inclusión.en.estadística"
## [121] "BS2.PBMCs...RNA"
## [122] "BS2.PBMCs...RNA...Fecha.de.extracción"
## [123] "BS2.PBMCs...RNA...Material.válido"
## [124] "BS2.PBMCs...RNA...Material.válido...Envío.Atrys"
## [125] "BS2.PBMCs...RNA...Material.válido...Envío.Atrys...Fecha.de.envío"
```

```
## [126] "BS2.PBMCs...RNA...Material.válido...Envío.Atrys...TCR"
## [127] "BS2.PBMCs...RNA...Material.válido...Envío.Leitat"
## [128] "BS2.PBMCs...RNA...Material.válido...Envío.Leitat...Librería.B"
## [129] "BS2.PBMCs...DNA"
## [130] "BS2.PBMCs...DNA...Fecha.de.extracción"
## [131] "BS2.PBMCs...DNA...Material.válido"
## [132] "BS2.PBMCs...DNA...Material.válido...Envío.Atrys"
## [133] "BS2.PBMCs...DNA...Material.válido...Envío.Atrys...Fecha.de.envío"
## [134] "BS2.PBMCs...DNA...Material.válido...Envío.Atrys...SNPs"
## [135] "BS3"
## [136] "BS3.Fecha.3ª.extracción"
## [137] "BS3.Evento"
## [138] "BS3.Fecha.recepción.al.laboratorio"
## [139] "BS3.Procesamiento"
## [140] "BS3.Plasma"
## [141] "BS3.Plasma...Citoquinas"
## [142] "BS3.Plasma...Extracción.cfDNA"
## [143] "BS3.Plasma...Extracción.cfDNA...NGS"
## [144] "BS3.PBMCs"
## [145] "BS3.PBMCs...Número.de.viales.de.PBMCs"
## [146] "BS3.PBMCs...Citometría.de.flujo"
## [147] "BS3.PBMCs...Citometría.de.flujo...Inclusión.en.estadística"
## [148] "BS3.PBMCs...RNA"
## [149] "BS3.PBMCs...RNA...Fecha.de.extracción"
## [150] "BS3.PBMCs...RNA...Material.válido"
## [151] "BS3.PBMCs...RNA...Material.válido...Envío.Atrys"
## [152] "BS3.PBMCs...RNA...Material.válido...Envío.Atrys...Fecha.de.envío"
## [153] "BS3.PBMCs...RNA...Material.válido...Envío.Atrys...TCR"
## [154] "BS3.PBMCs...RNA...Material.válido...Envío.Leitat"
## [155] "BS3.PBMCs...RNA...Material.válido...Envío.Leitat...Librería.B"
## [156] "BS3.PBMCs...DNA"
## [157] "BS3.PBMCs...DNA...Fecha.de.extracción"
## [158] "BS3.PBMCs...DNA...Material.válido"
## [159] "BS3.PBMCs...DNA...Material.válido...Envío.Atrys"
## [160] "BS3.PBMCs...DNA...Material.válido...Envío.Atrys...Fecha.de.envío"
## [161] "BS3.PBMCs...DNA...Material.válido...Envío.Atrys...SNPs"
## [162] "BS1.BS2.BS3"
## [163] "BS1+BS2+BS3"
## [164] "BS1+BS2"
## [165] "BS1+BS3"
## [166] "BS2+BS3"
## [167] "Código.Inmunosight"
## [168] "TERAPIA.BIOLÓGICA"
```

A lo largo del documento describiremos los distintos parámetros o variables que encontramos, asi como las transformaciones que debemos aplicar en los distintos casos.

Paciente

Identificación del paciente.

Fecha.de.inicio.IO

Fecha de inicio del tratamiento de Imnunoterapia. Nos será util para crear una nueva variable que será resultado de la fecha de inicio de inmunoterapia menos la fecha de diagnóstico. Alternativamente se puede

considerar la edad de inicio de la imnunoterapia menos la edad al diagnóstico.

```
head(clinic_full_data$Fecha.de.inicio.IO)
```

```
## [1] "19/06/2018" "23/08/2019" "28/04/2023" "17/04/2019" "08/07/2021"
## [6] "21/08/2019"
```

```
class(clinic_full_data$Fecha.de.inicio.IO)
```

```
## [1] "character"
```

```
cat("Número de NAs:", sum(is.na(clinic_full_data$Fecha.de.inicio.IO)), "\n")
```

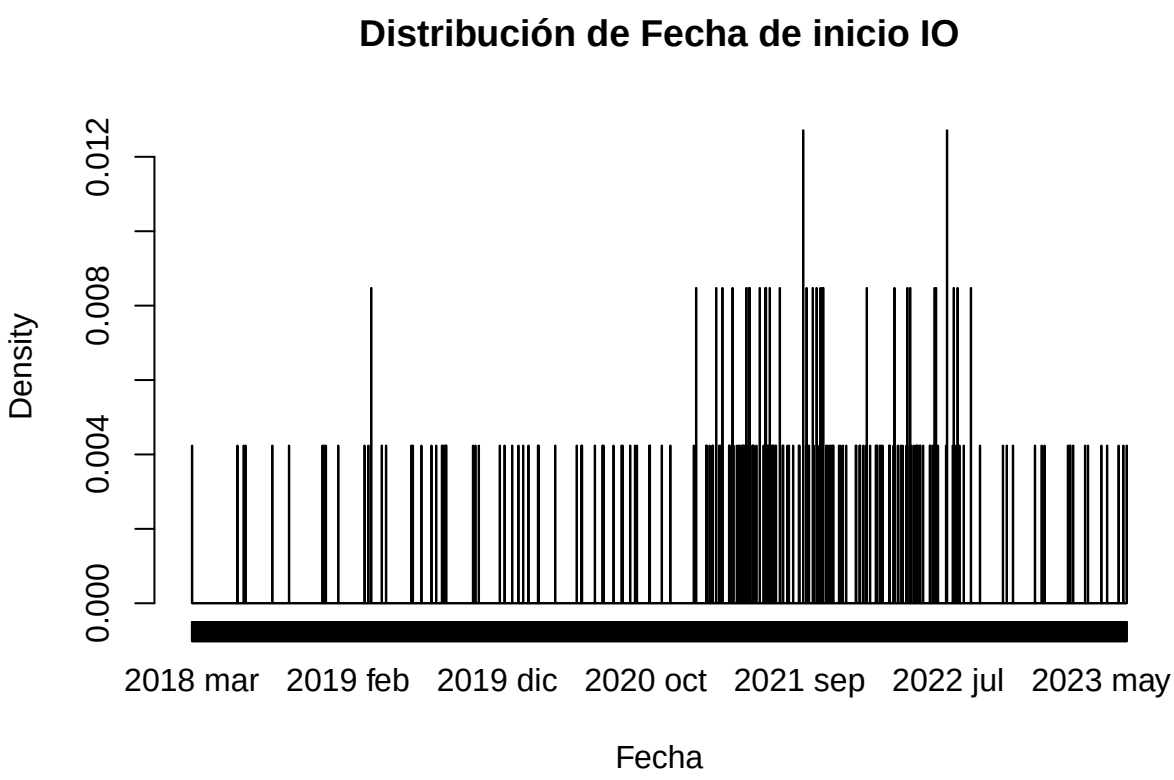
```
## Número de NAs: 8
```

```
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Fecha.de.inicio.IO)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 3.28 %
```

Se propone su correcta conversión a formato de fecha

```
# Convertir la columna a Date
clinic_full_data$Fecha.de.inicio.IO <- as.Date(clinic_full_data$Fecha.de.inicio.IO, format = "%d/%m/%Y")

# Definir cortes cada 30 días
breaks_seq <- seq(min(clinic_full_data$Fecha.de.inicio.IO, na.rm = TRUE),
                  max(clinic_full_data$Fecha.de.inicio.IO, na.rm = TRUE),
                  by = "1 days")

# Crear el histograma
hist(clinic_full_data$Fecha.de.inicio.IO,
     breaks = breaks_seq,
     main = "Distribución de Fecha de inicio IO",
     xlab = "Fecha",
     col = "skyblue",
     border = "black")
```



PFS..d.

Define el tiempo que el paciente permanece vivo, medido en días, y en que la enfermedad permanece sin empeorar (sin progresar) desde un punto inicial definido.

```
head(clinic_full_data$`PFS.(d)`)
```

```
## [1]   50  258  234  107   36 1150
```

```
class(clinic_full_data$`PFS.(d)`)
```

```
## [1] "numeric"
```

```
cat("Número de NAs:", sum(is.na(clinic_full_data$`PFS.(d)`)), "\n")
```

```
## Número de NAs: 9
```

```
summary(clinic_full_data$`PFS.(d)`)
```

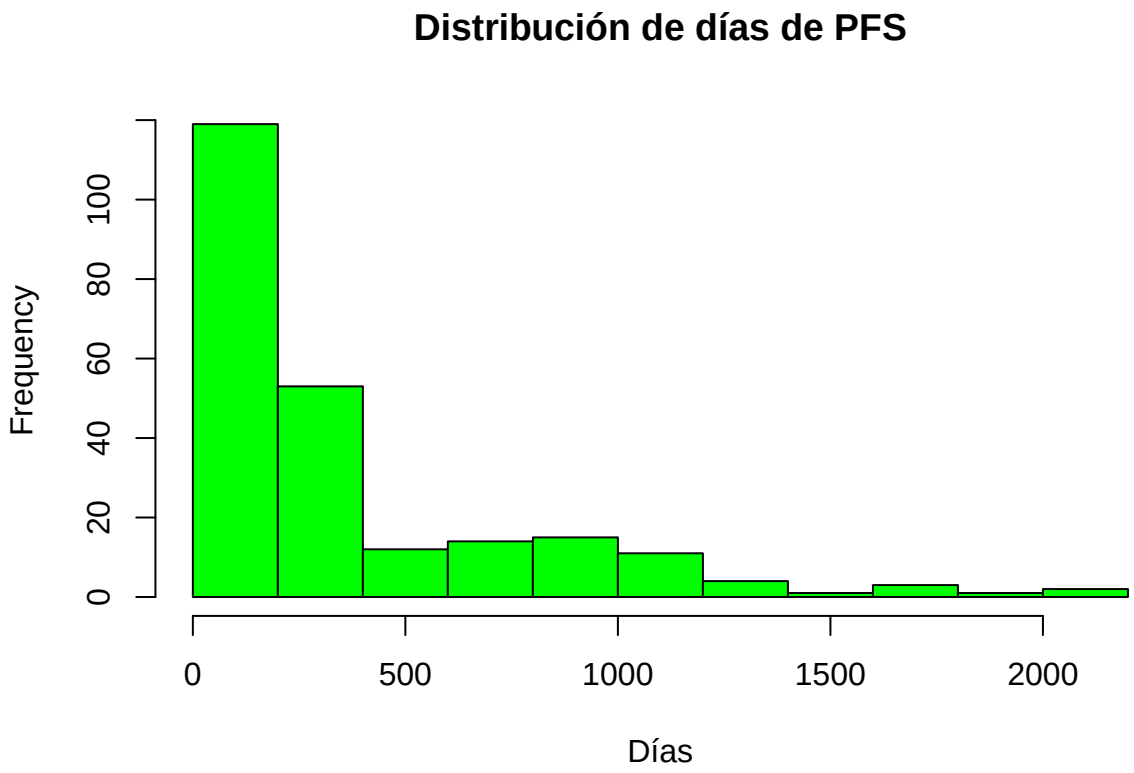
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##     1.0    84.0   193.0   366.9   516.0  2058.0       9
```

```
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$`PFS.(d)`)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 3.69 %
```

```
hist(clinic_full_data$`PFS.(d)`,
     breaks = 10,
     main = "Distribución de días de PFS",
     xlab = "Días",
```

```
     col = "green",
     border = "black")
```



Follow.Up.(d) Variavle que dice la duración de seguimiento, expresada en días, para el análisis de supervivencia global. Es decir, para cada paciente, mide el número de días transcurridos desde el punto de partida (por ejemplo, la fecha de diagnóstico o el inicio del tratamiento) hasta

```
head(clinic_full_data$`Follow.Up.(d)`)
```

```
## [1]   66  259  234  295   58 1150
```

```
class(clinic_full_data$`Follow.Up.(d)`)
```

```
## [1] "numeric"
```

```
cat("Número de NAs:", sum(is.na(clinic_full_data$`Follow.Up.(d)`)), "\n")
```

```
## Número de NAs: 9
```

```
summary(clinic_full_data$`Follow.Up.(d)`)
```

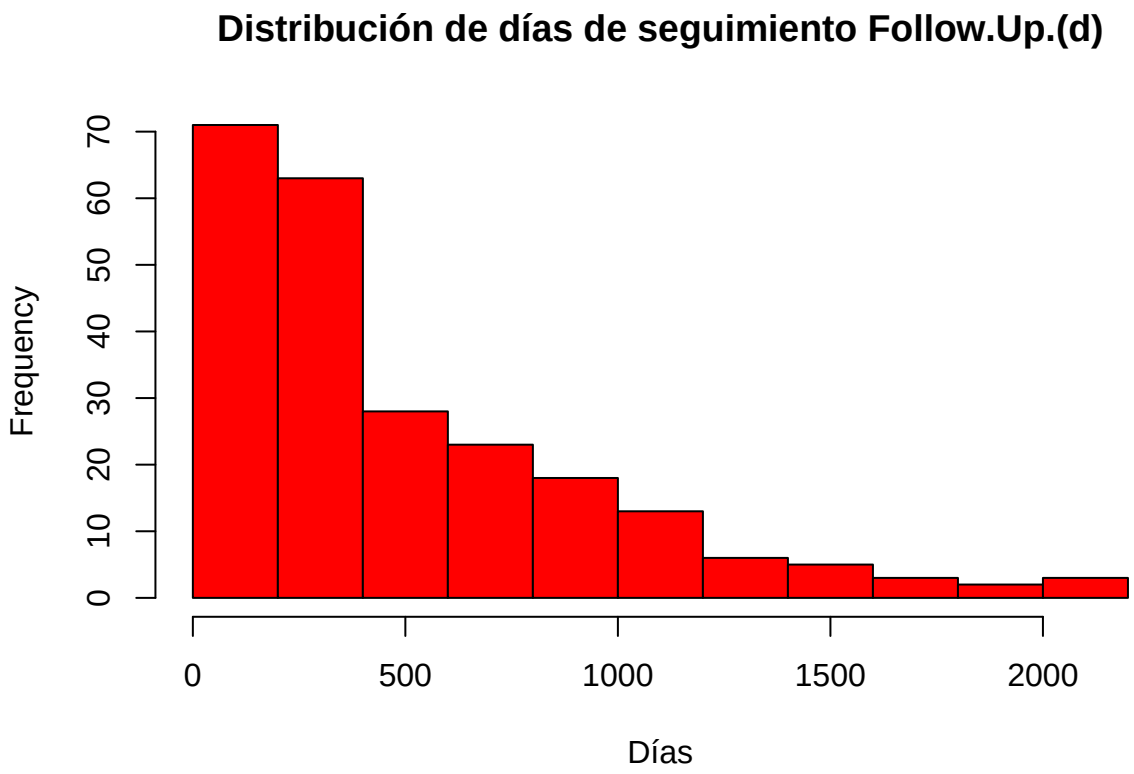
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##     2.0   167.0   357.0   497.3   713.0  2114.0       9
```

```
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$`Follow.Up.(d)`)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 3.69 %
```

```
hist(clinic_full_data$`Follow.Up.(d)`,
     breaks = 10,
     main = "Distribución de días de seguimiento Follow.Up.(d)",
```

```
     xlab = "Días",
     col = "red",
     border = "black")
```



PFS..m.

Define el tiempo que el paciente permanece vivo, medido en meses, y en que la enfermedad permanece sin empeorar (sin progresar) desde un punto inicial definido.

```
head(clinic_full_data$`PFS.(m)`)
```

```
## [1]  1.666667  8.600000  7.800000  3.566667  1.200000 38.333333
```

```
class(clinic_full_data$`PFS.(m)`)
```

```
## [1] "numeric"
```

```
cat("Número de NAs:", sum(is.na(clinic_full_data$`PFS.(m)`)), "\n")
```

```
## Número de NAs: 9
```

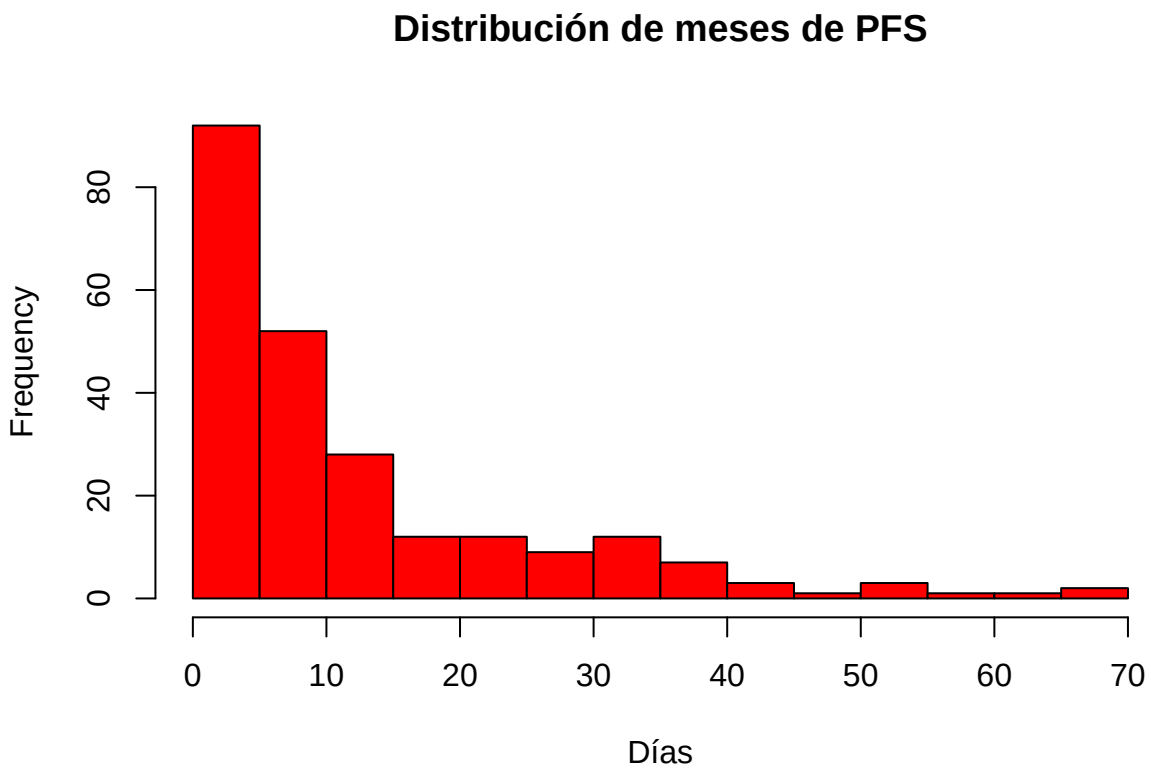
```
summary(clinic_full_data$`PFS.(m)`)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.     NA's
##  0.03333  2.80000  6.43333 12.23078 17.20000 68.60000        9
```

```
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$`PFS.(m)`)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 3.69 %
```

```r
hist(clinic_full_data$`PFS.(m)`,
     breaks = 10,
     main = "Distribución de meses de PFS",
     xlab = "Días",
     col = "red",
     border = "black")
```



Progresión...6

Variable booleana para indicar si hay o no progresión de la enfermedad.

```r
head(clinic_full_data$Progresión...6)
```

```
## [1] 1 1 1 1 1 1
```

```r
class(clinic_full_data$Progresión...6)
```

```
## [1] "numeric"
```

```r
cat("Número de NAs:", sum(is.na(clinic_full_data$Progresión...6)), "\n")
```

```
## Número de NAs: 0
```

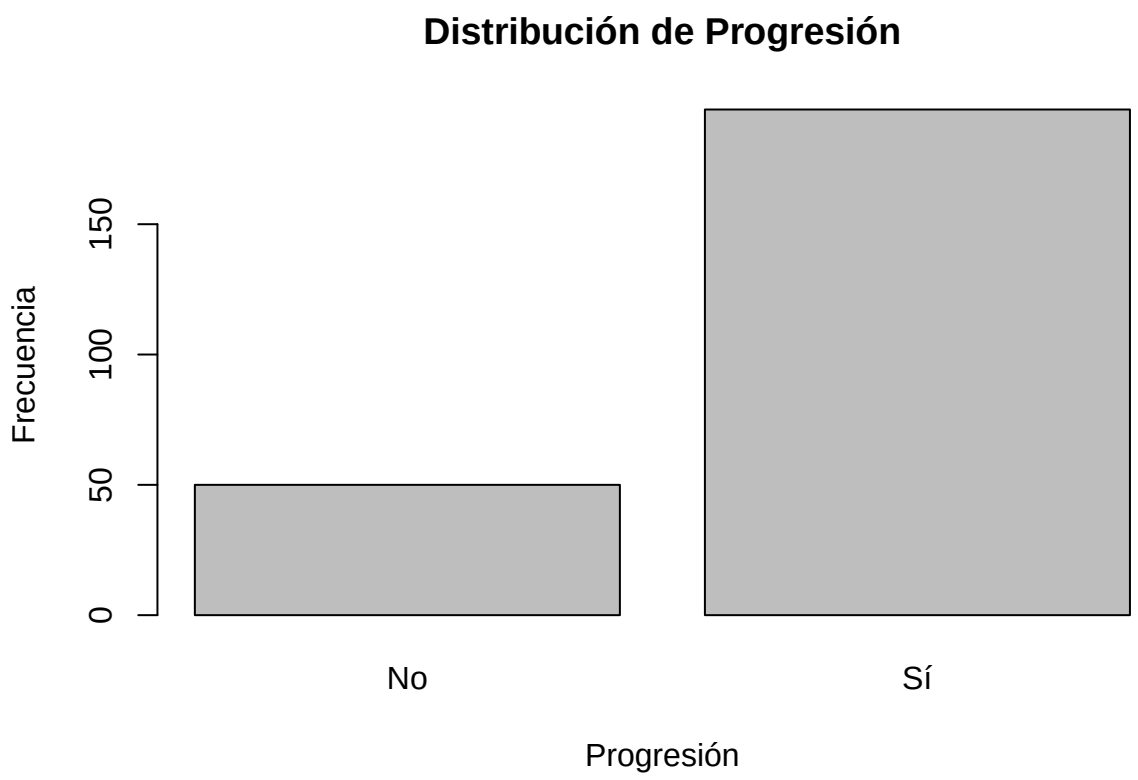
```r
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Progresión...6)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 0 %
```

Se propone realizar una factorización de las clases

```r
clinic_full_data$Progresión...6 <- factor(clinic_full_data$Progresión...6,
                                          levels = c(0, 1),
                                          labels = c("No", "Sí"))
```

```r
plot(clinic_full_data$Progresión...6,
     main = "Distribución de Progresión",
     xlab = "Progresión",
     ylab = "Frecuencia",
     border = "black")
```



EXITUS

Variable que determina el *exitus letalis* se emplea en el ámbito médico para referirse a los casos clínicos en los que la enfermedad ha llevado al paciente a la muerte

```r
head(clinic_full_data$Exitus)
```

```
## [1] 1 1 1 1 1 1
```

```r
class(clinic_full_data$Exitus)
```

```
## [1] "numeric"
```

```r
cat("Número de NAs:", sum(is.na(clinic_full_data$Exitus)), "\n")
```

```
## Número de NAs: 0
```

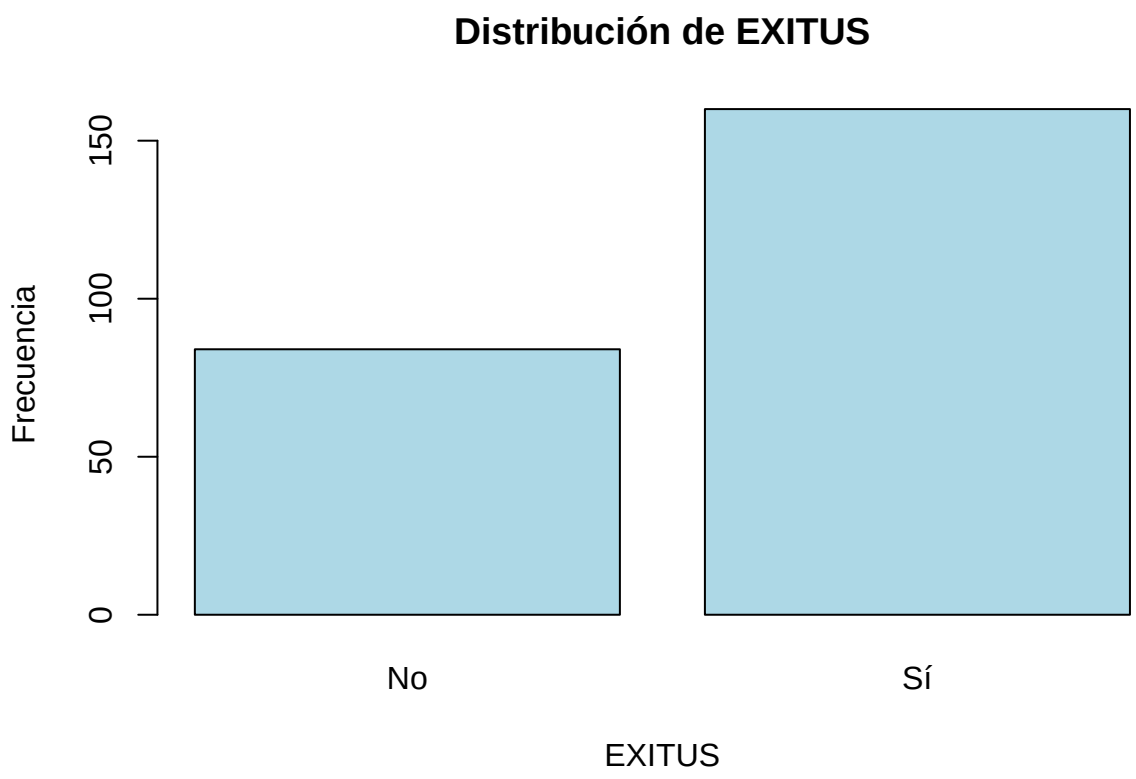
```r
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Exitus)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 0 %
```

Se propone realizar una factorización de las clases

```
clinic_full_data$Exitus <- factor(clinic_full_data$Exitus,
                                  levels = c(0, 1),
                                  labels = c("No", "Sí"))
```

```
plot(clinic_full_data$Exitus,
     main = "Distribución de EXITUS",
     xlab = "EXITUS",
     ylab = "Frecuencia",
     col = "lightblue",
     border = "black")
```



Nos aseguramos de que no hay casos de exitus sin recurrencia, es decir, sin progresión. Cabria esperar que los casos de exitus positivos sean menores o, como mucho, iguales que los de progresión positiva.

```
cat("Frecuencias de progresión:\n")
```

```
## Frecuencias de progresión:
```

```
table(clinic_full_data$Progresión...6)
```

```
##
##  No  Sí
##  50 194
```

```
cat("Frecuencias de Exitus:\n")
```

```
## Frecuencias de Exitus:
```

```
table(clinic_full_data$Exitus)
```

```
##
##  No  Sí
##  84 160
```

```
cat("Numero de casos de exitus en los que no ha habido progresión:", sum(clinic_full_data$Progresión...
```

```
## Numero de casos de exitus en los que no ha habido progresión: 0
```

```
cat("Numero de casos de progresión en los que no ha habido exitus:", sum(clinic_full_data$Progresión...
```

```
## Numero de casos de progresión en los que no ha habido exitus: 34
```

Fecha.de.nacimiento

Fecha de nacimiento del paciente

```
head(clinic_full_data$Fecha.de.nacimiento)
```

```
## [1] "31/10/1960" "13/01/1946" "28/12/1936" "02/06/1960" "22/04/1974"
## [6] "01/04/1951"
```

```
class(clinic_full_data$Fecha.de.nacimiento)
```

```
## [1] "character"
```

```
sum(is.na(clinic_full_data$Fecha.de.nacimiento))
```

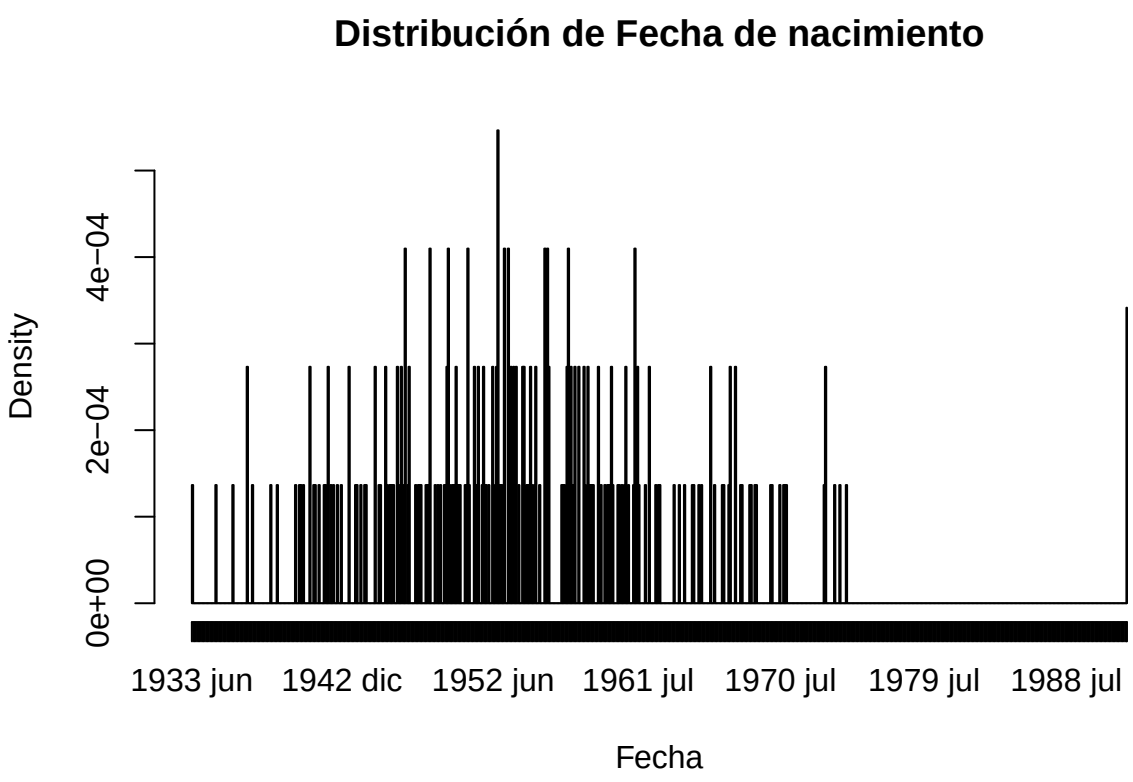
```
## [1] 0
```

```
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Fecha.de.nacimiento)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 0 %
```

Se propone su correcta conversion a formato de fecha

```
# Convertir la columna a Date
clinic_full_data$Fecha.de.nacimiento <- as.Date(clinic_full_data$Fecha.de.nacimiento, format = "%d/%m/%

# Obtener el mínimo y máximo de las fechas
min_date <- min(clinic_full_data$Fecha.de.nacimiento, na.rm = TRUE)
max_date <- max(clinic_full_data$Fecha.de.nacimiento, na.rm = TRUE)

# Crear secuencia de cortes cada 30 días
breaks_seq <- seq(min_date, max_date, by = "30 days")
# Si el último corte es menor que la fecha máxima, agregar la fecha máxima
if (max(breaks_seq) < max_date) {
  breaks_seq <- c(breaks_seq, max_date)
}

# Crear el histograma
hist(clinic_full_data$Fecha.de.nacimiento,
     breaks = breaks_seq,
     main = "Distribución de Fecha de nacimiento",
     xlab = "Fecha",
     col = "skyblue",
     border = "black")
```



De esta variable obtendremos una variable de Edad a día de la última actualización del estudio (14/06/2024).

```
# Define la fecha de referencia (14 de junio de 2024)
fecha_ref <- as.Date("14/06/2024", format = "%d/%m/%Y")
# Calcula la edad como diferencia en años (dividiendo entre 365.25) y conviértela a entero.
clinic_full_data$Edad.ultima.actualizacion <- as.integer((fecha_ref - clinic_full_data$Fecha.de.nacimien
```

Edad.ultima.actualizacion

Edad del paciente en la última actualización del estudio

```
head(clinic_full_data$Edad.ultima.actualizacion)
```

```
## [1] 63 78 87 64 50 73
```

```
class(clinic_full_data$Edad.ultima.actualizacion)
```

```
## [1] "integer"
```

```
cat("Número de NAs:", sum(is.na(clinic_full_data$Edad.ultima.actualizacion)), "\n")
```

```
## Número de NAs: 0
```

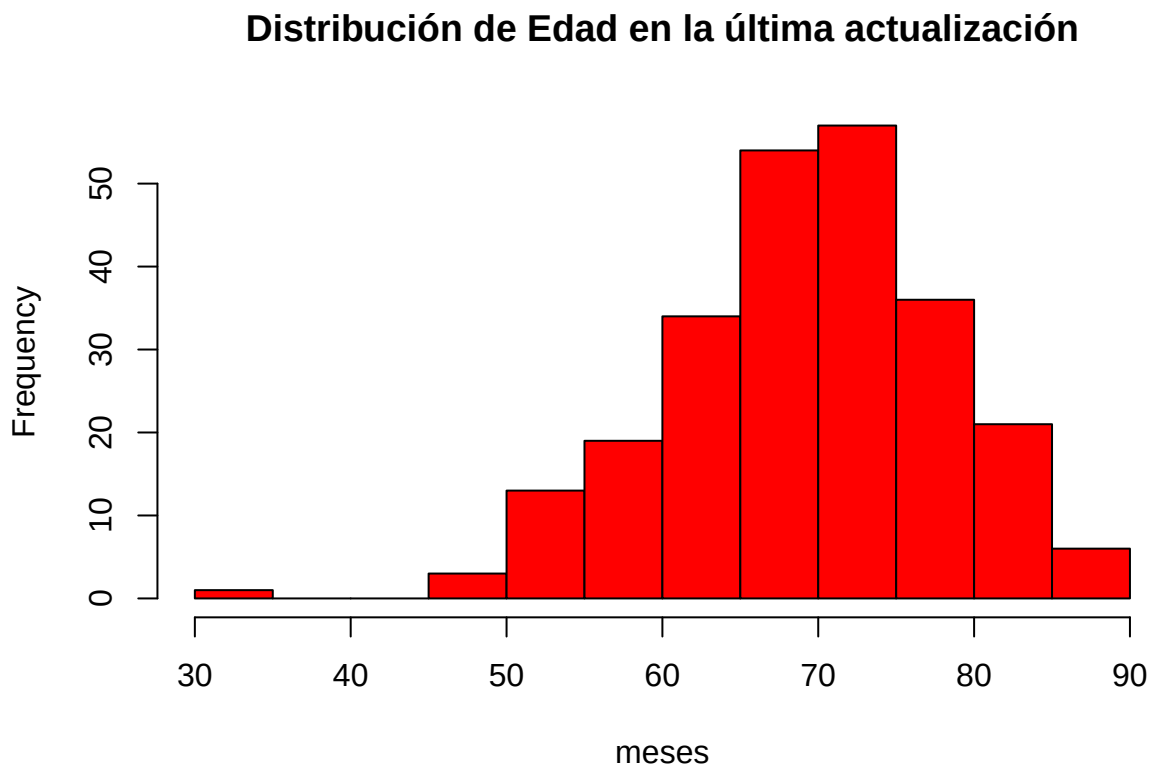
```
summary(clinic_full_data$Edad.ultima.actualizacion)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   32.00   64.00   70.00   69.69   76.00   90.00
```

```
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Edad.ultima.actualizacion)) * 100, 2), "%\n
```

```
## Porcentaje de NAs: 0 %
```

```r
hist(clinic_full_data$Edad.ultima.actualizacion,
     breaks = 10,
     main = "Distribución de Edad en la última actualización",
     xlab = "meses",
     col = "red",
     border = "black")
```



Edad.al.diagnóstico

Edad del paciente al tiempo de recibir el diagnóstico

```r
head(clinic_full_data$Edad.al.diagnóstico)
```

```
## [1] 55 68 84 58 47 60
```

```r
class(clinic_full_data$Edad.al.diagnóstico)
```

```
## [1] "numeric"
```

```r
sum(is.na(clinic_full_data$Edad.al.diagnóstico))
```

```
## [1] 1
```

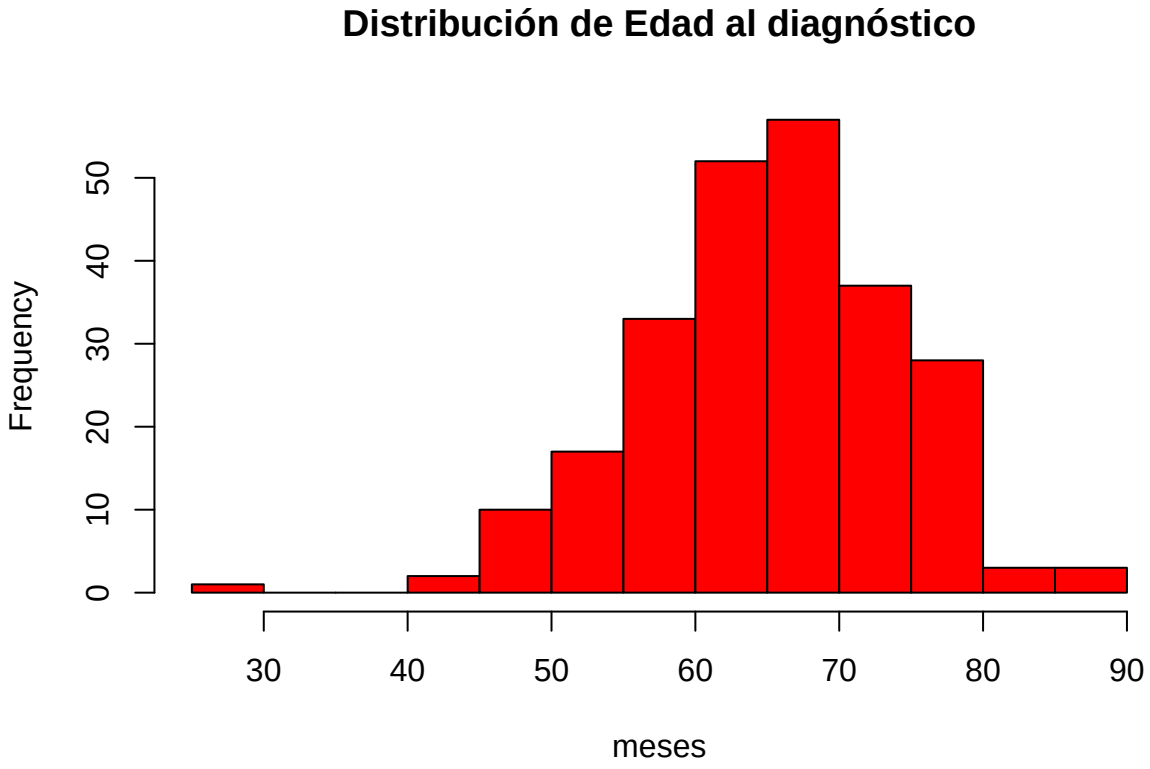
```r
summary(clinic_full_data$Edad.al.diagnóstico)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   29.00   60.00   66.00   65.76   72.00   87.00       1
```

```r
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Edad.al.diagnóstico)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 0.41 %
```

```r
hist(clinic_full_data$Edad.al.diagnóstico,
     breaks = 10,
     main = "Distribución de Edad al diagnóstico",
     xlab = "meses",
     col = "red",
     border = "black")
```



Sexo..0.Mujer..1.Varón.

Variable booleana para indicar sexo del paciente

```r
head(clinic_full_data$`Sexo.(0=Mujer;.1=Varón)`)
```

```
## [1] 1 1 1 0 1 1
```

```r
class(clinic_full_data$`Sexo.(0=Mujer;.1=Varón)`)
```

```
## [1] "numeric"
```

```r
cat("Número de NAs", sum(is.na(clinic_full_data$`Sexo.(0=Mujer;.1=Varón)`)), "\n")
```

```
## Número de NAs 0
```

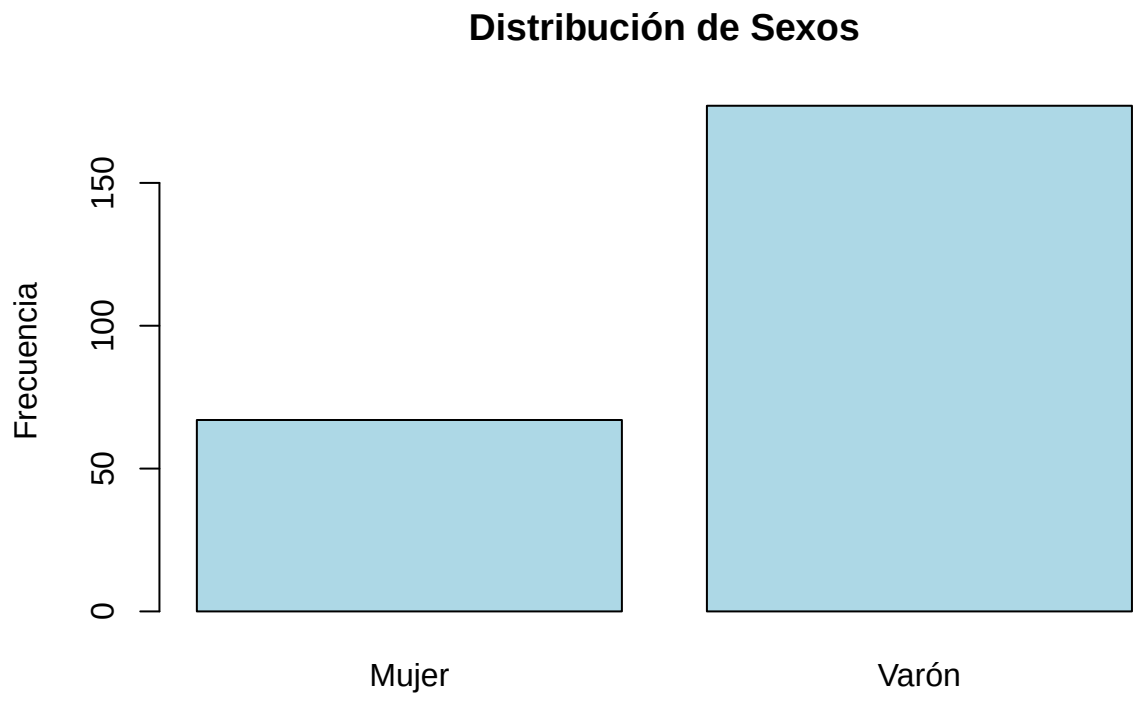
```r
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$`Sexo.(0=Mujer;.1=Varón)`)) * 100, 2), "%\n
```

```
## Porcentaje de NAs: 0 %
```

Se propone realizar una factorización de las clases

```
clinic_full_data$`Sexo.(0=Mujer;.1=Varón)` <- factor(clinic_full_data$`Sexo.(0=Mujer;.1=Varón)`,
                                                     levels = c("0", "1"),
                                                     labels = c("Mujer", "Varón"))
```

```
plot(clinic_full_data$`Sexo.(0=Mujer;.1=Varón)`,
     main = "Distribución de Sexos",
     ylab = "Frecuencia",
     col = "lightblue",
     border = "black")
```



Tabaquismo..0.Nunca.fumador..1.Exfumador..2..Fumador.Activo."

Variable que determina el habito del paciente con respecto al tabaco

```
head(clinic_full_data$`Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)`)
```

```
## [1] 1 1 2 1 2 1
```

```
class(clinic_full_data$`Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)`)
```

```
## [1] "numeric"
```

```
cat("Numero de NAs:", sum(is.na(clinic_full_data$`Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.A
```

```
## Numero de NAs: 1
```

```
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$`Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.1
```

```
## Porcentaje de NAs: 0.41 %
```

Comprobamos las clases dentro de la variable

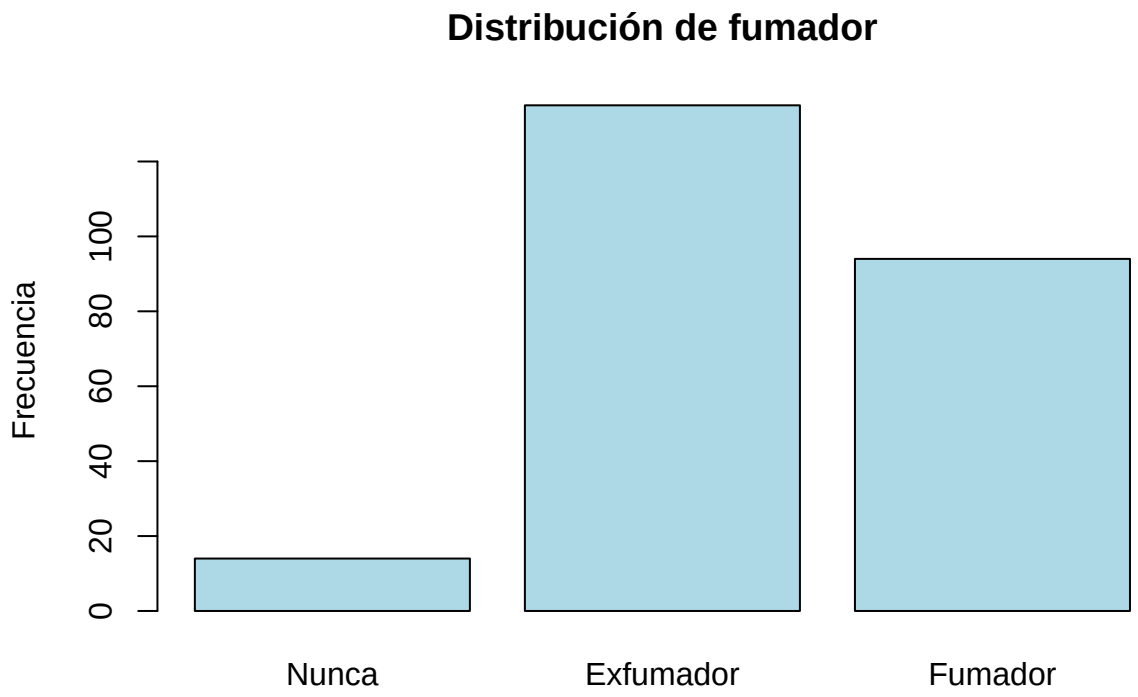
```
unique(clinic_full_data$`Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)`)
```

```
## [1]  1  2  0 NA
```

Se propone realizar una factorización de las clases

```
clinic_full_data$`Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)` <- factor(
  clinic_full_data$`Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)`,
  levels = c(0, 1, 2),
  labels = c("Nunca", "Exfumador", "Fumador")
)
```

```
plot(clinic_full_data$`Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)`,
     main = "Distribución de fumador",
     ylab = "Frecuencia",
     col = "lightblue",
     border = "black")
```



Tumor.previo

Variable booleana para indicar si el paciente padeció o no tumor previo a la condición del estudio

```
head(clinic_full_data$Tumor.previo)
```

```
## [1] 1 1 1 0 0 0
```

```r
class(clinic_full_data$Tumor.previo)
```

```
## [1] "numeric"
```

```r
cat("Número de NAs:", sum(is.na(clinic_full_data$Tumor.previo)), "\n")
```

```
## Número de NAs: 1
```

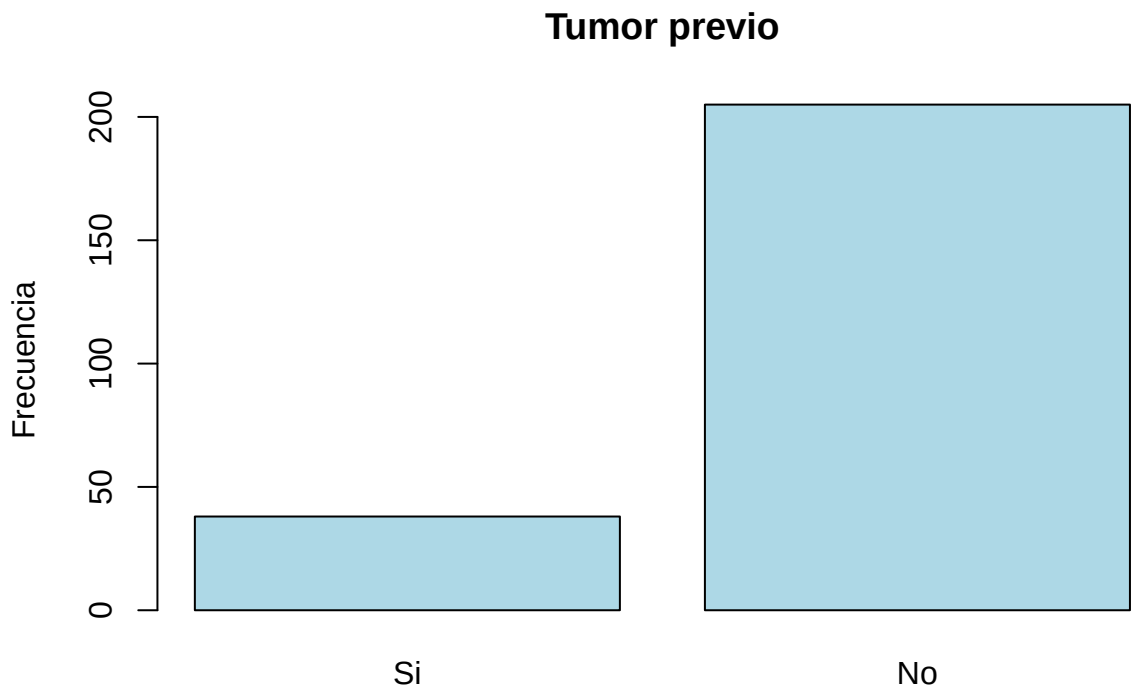
```r
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Tumor.previo)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 0.41 %
```

Se propone realizar una factorización de las clases

```r
clinic_full_data$Tumor.previo <- factor(clinic_full_data$Tumor.previo,
                                        levels = c(1, 0),
                                        labels = c("Si", "No"))
```

```r
plot(clinic_full_data$Tumor.previo,
     main = "Tumor previo",
     ylab = "Frecuencia",
     col = "lightblue",
     border = "black")
```



Escala.ECOG

Evaluación del estado funcional del paciente basada en la ECOG Performance Status. La escala varía de 0 a 5, donde 0 indica un estado óptimo (paciente totalmente activo) y 5 representa el fallecimiento

```r
head(clinic_full_data$Escala.ECOG)
```

```
## [1] 1 1 0 0 1 1
```

```r
class(clinic_full_data$Escala.ECOG)
```

```
## [1] "numeric"
```

```r
cat("Número de NAs", sum(is.na(clinic_full_data$Escala.ECOG)), "\n")
```

```
## Número de NAs 8
```

```r
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Escala.ECOG)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 3.28 %
```

Clases que tenemos dentro de la variable `Escala.ECOG`:

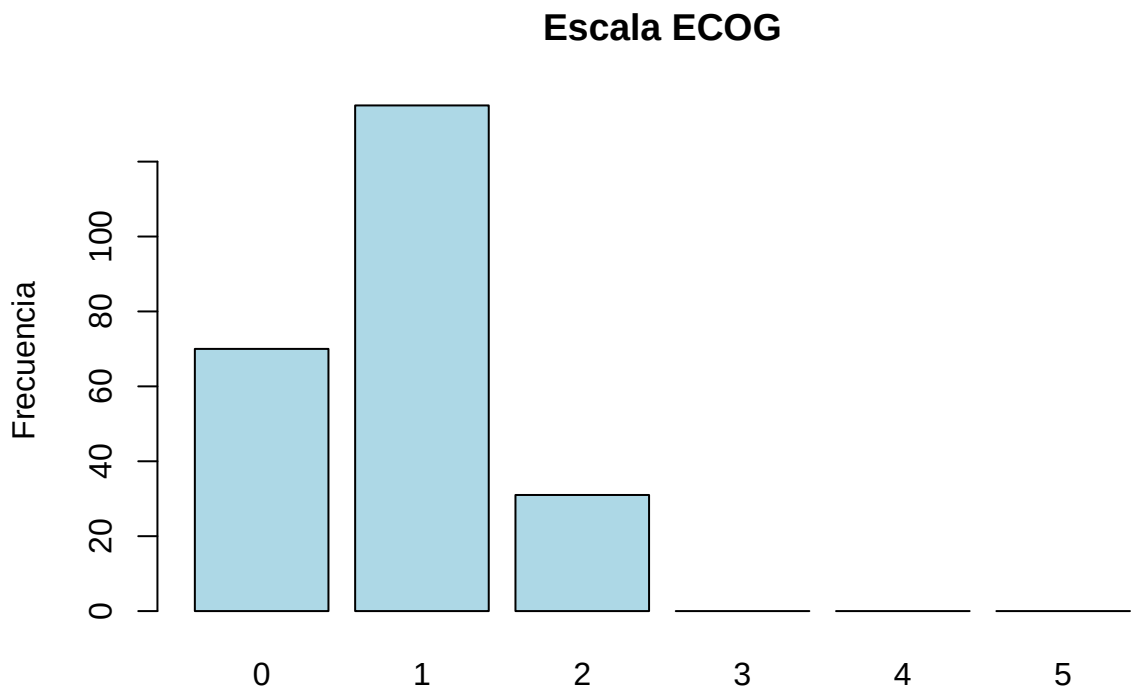
```r
unique(clinic_full_data$Escala.ECOG)
```

```
## [1]  1  0  2 NA
```

Se propone realizar una factorización de las clases

```r
clinic_full_data$Escala.ECOG <- factor(clinic_full_data$Escala.ECOG,
                                       levels = c(0,1,2,3,4,5))
```

```r
plot(clinic_full_data$Escala.ECOG,
     main = "Escala ECOG",
     ylab = "Frecuencia",
     col = "lightblue",
     border = "black")
```



Fecha.de.diagnóstico

Fecha en la que se le diagnosticó la enfermedad. Utilizaremos esta variable para sacar otra variable llamada "Tiempo.hasta.inicio.IO"

```
head(clinic_full_data$Fecha.de.diagnóstico)
```

```
## [1] "08/05/2016" "01/02/2014" "01/06/2021" "01/04/2019" "04/06/2021"
## [6] "15/11/2011"
```

```
class(clinic_full_data$Fecha.de.diagnóstico)
```

```
## [1] "character"
```

```
sum(is.na(clinic_full_data$Fecha.de.diagnóstico))
```

```
## [1] 2
```

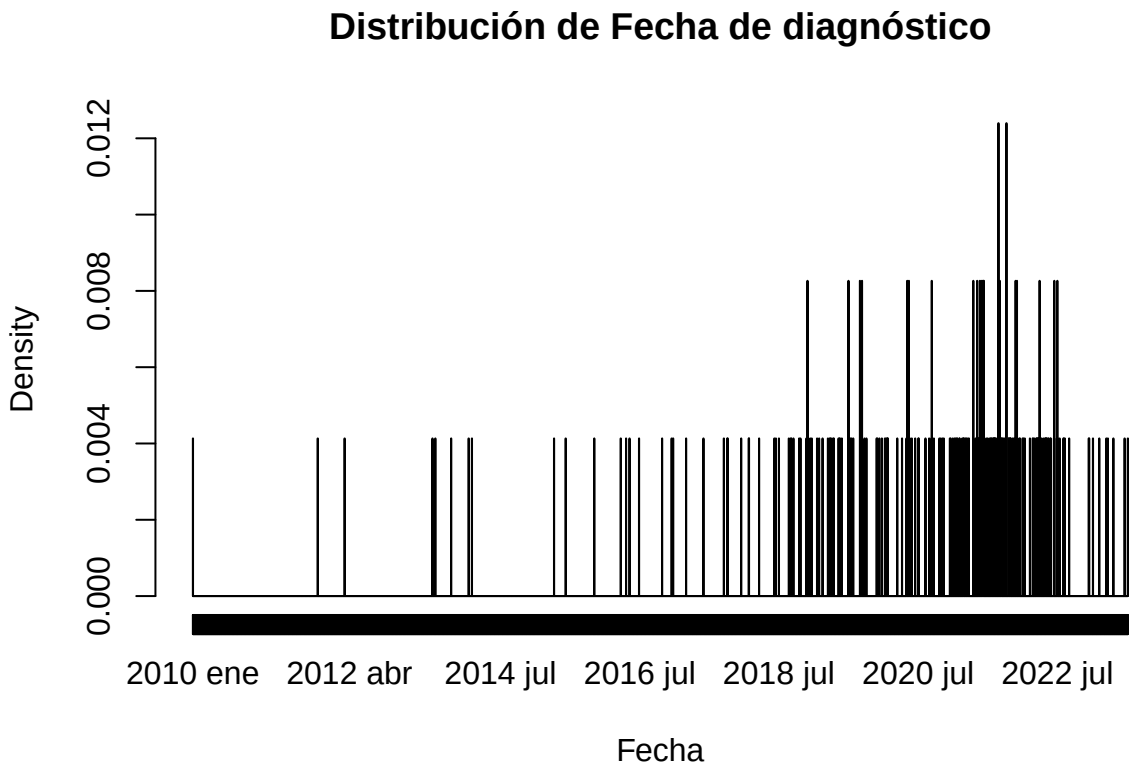
```
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Fecha.de.diagnóstico)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 0.82 %
```

Se propone su correcta conversion a formato de fecha

```
# Convertir la columma a Date
clinic_full_data$Fecha.de.diagnóstico <- as.Date(clinic_full_data$Fecha.de.diagnóstico, format = "%d/%m/

# Definir cortes
breaks_seq <- seq(min(clinic_full_data$Fecha.de.diagnóstico, na.rm = TRUE),
                  max(clinic_full_data$Fecha.de.diagnóstico, na.rm = TRUE),
```

```
                  by = "1 days")

# Crear el histograma
hist(clinic_full_data$Fecha.de.diagnóstico,
     breaks = breaks_seq,
     main = "Distribución de Fecha de diagnóstico",
     xlab = "Fecha",
     col = "skyblue",
     border = "black")
```



Tiempo.hasta.inicio.IO

```
clinic_full_data$Tiempo.hasta.inicio.IO <- as.numeric(clinic_full_data$Fecha.de.inicio.IO - clinic_full
```

Define el tiempo que pasa (medido en días) desde que el paciente recibe el diagnostico hasta que inicia el tratamiento con inmunoterapia.

Define el tiempo que el paciente permanece vivo, medido en días, y en que la enfermedad permanece sin empeorar (sin progresar) desde un punto inicial definido.

```
head(clinic_full_data$Tiempo.hasta.inicio.IO)
```

```
## [1]  772 2029  696   16   34 2836
```

```
class(clinic_full_data$Tiempo.hasta.inicio.IO)
```

```
## [1] "numeric"
```

```r
summary(clinic_full_data$Tiempo.hasta.inicio.IO)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##     2.0    28.0    78.5   364.0   446.5  4488.0      10
```

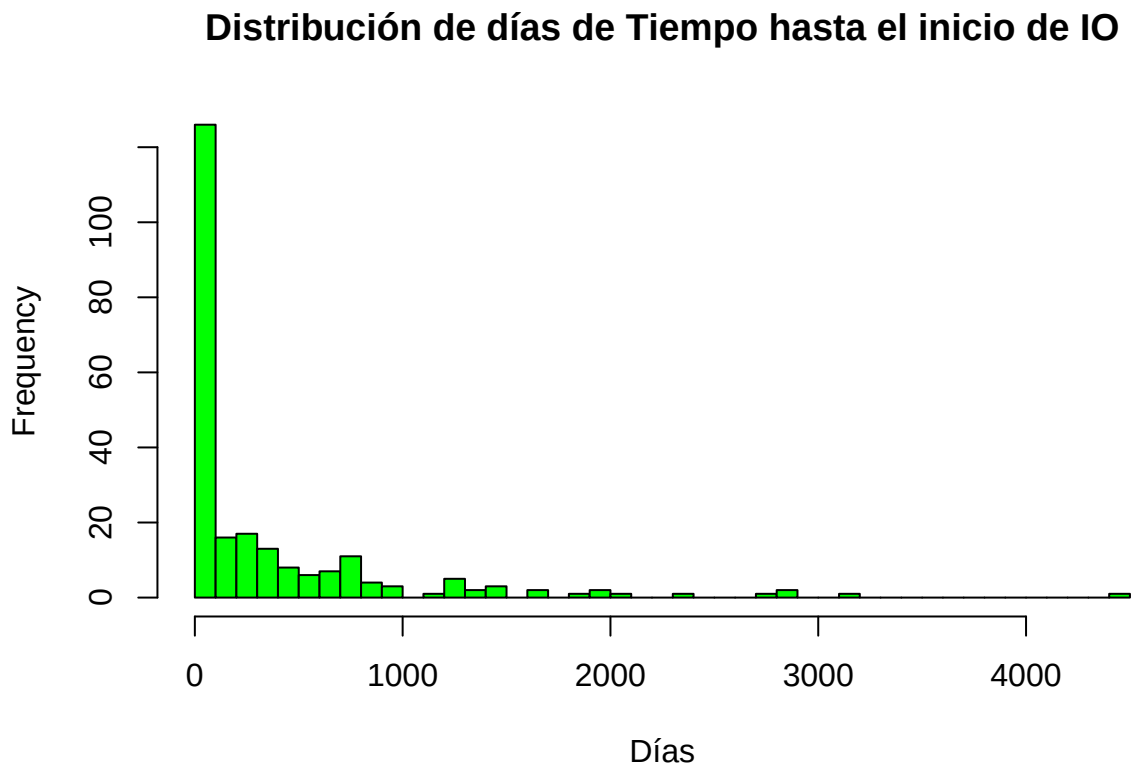
```r
cat("Número de NAs:", sum(is.na(clinic_full_data$Tiempo.hasta.inicio.IO)), "\n")
```

```
## Número de NAs: 10
```

```r
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Tiempo.hasta.inicio.IO)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 4.1 %
```

```r
hist(clinic_full_data$Tiempo.hasta.inicio.IO,
     breaks = 50,
     main = "Distribución de días de Tiempo hasta el inicio de IO",
     xlab = "Días",
     col = "green",
     border = "black")
```



Tenemos valores algo sospechosos de ser errores de entrada. Obtenemos un máximo de tiempo hasta el diagnóstico de 7057 días, lo que es lo mismo, 19 años. La fecha que genera esa cifra es la fecha de diagnostico: 01-01-2003. El hecho de que sea una cifra tan peculiar (el día de año nuevo) y tan lejana, hace sospechar que se trata de un error.

Estadio.al.diagnóstico

Indica el estadio del cáncer al momento del diagnóstico, expresado en categorías que reflejan la extensión y severidad de la enfermedad. Estadio I: Indica un tumor localizado y generalmente pequeño, sin evidencia

de diseminación a ganglios linfáticos ni a órganos distantes. Estadio II y III: Sugieren una mayor extensión local y regional del tumor, con compromiso de ganglios linfáticos en mayor o menor medida. Estadio IV: Representa enfermedad avanzada con metástasis a otros órganos.

```
head(clinic_full_data$Estadio.al.diagnóstico)
```

```
## [1] "IVB"  "IVA"  "IIIB" "IVA"  "IVB"  "IIIA"
```

```
class(clinic_full_data$Estadio.al.diagnóstico)
```

```
## [1] "character"
```

```
sum(is.na(clinic_full_data$Estadio.al.diagnóstico))
```

```
## [1] 7
```

```
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Estadio.al.diagnóstico)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 2.87 %
```

Clases que tenemos dentro de la variable `Estadio.al.diagnóstico`:

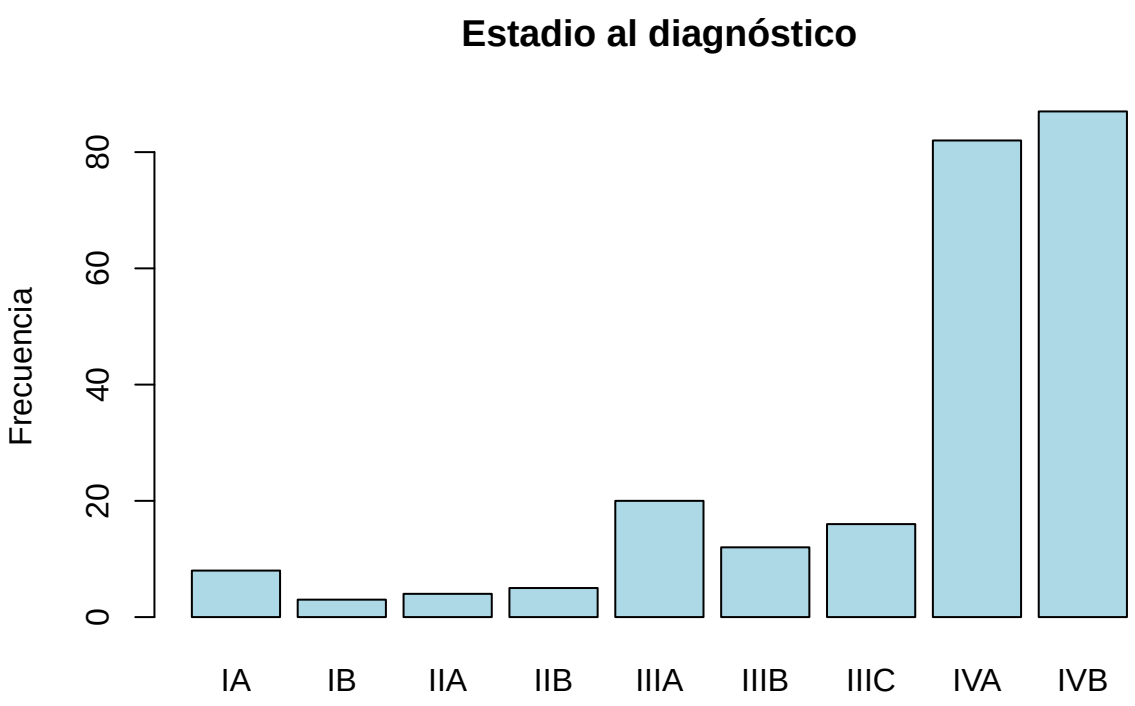
```
unique(clinic_full_data$Estadio.al.diagnóstico)
```

```
##  [1] "IVB"  "IVA"  "IIIB" "IIIA" "IB"   "IIB"  "IIIC" "IA"   "IIA"  NA
```

Se propone realizar una factorización de las clases

```
# Factorizamos por niveles para ordenarlos por severidad
clinic_full_data$Estadio.al.diagnóstico <- factor(clinic_full_data$Estadio.al.diagnóstico,
                                                  levels = c("IA", "IB", "IIA", "IIB", "IIIA", "IIIB",
                                                  ordered = TRUE)
```

```
plot(clinic_full_data$Estadio.al.diagnóstico,
     main = "Estadio al diagnóstico",
     ylab = "Frecuencia",
     col = "lightblue",
     border = "black")
```



Histología

Tipo histológico del tumor, que clasifica el cáncer según la apariencia de las células bajo el microscopio

```
clinic_full_data$Histología[clinic_full_data$Histología == "No especificado"] <- NA
```

```
head(clinic_full_data$Histología)
```

```
## [1] NA                   "Adenocarcinoma"     "Carcinoma escamoso"
## [4] "Adenocarcinoma"     "Adenocarcinoma"     "Adenocarcinoma"
```

```
class(clinic_full_data$Histología)
```

```
## [1] "character"
```

```
sum(is.na(clinic_full_data$Histología))
```

```
## [1] 18
```

```
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Histología)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 7.38 %
```

Clases que tenemos dentro de la variable `Histologia`:

```
unique(clinic_full_data$Histología)
```

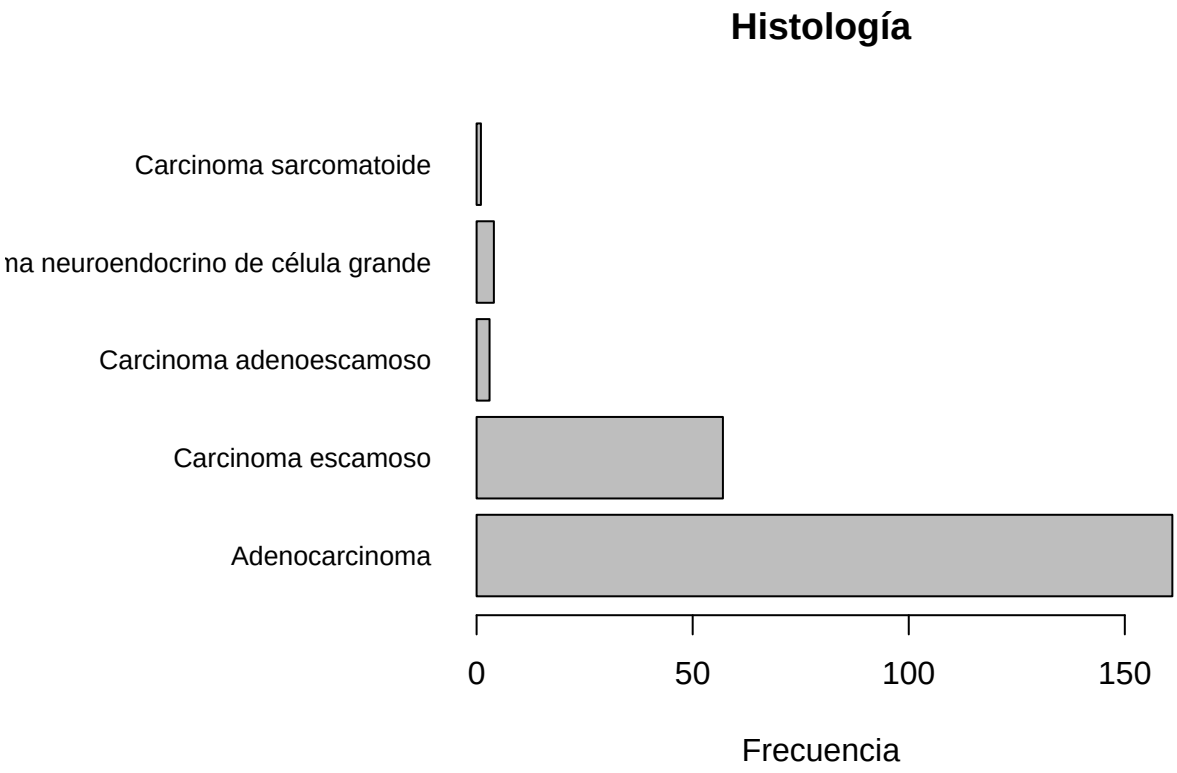
```
## [1] NA
## [2] "Adenocarcinoma"
## [3] "Carcinoma escamoso"
```

```
## [4] "Carcinoma adenoescamoso"
## [5] "Carcinoma neuroendocrino de célula grande"
## [6] "Carcinoma sarcomatoide"
```

Se propone realizar una factorización de las clases

```
clinic_full_data$Histología <- factor(clinic_full_data$Histología,
                                      levels = unique(clinic_full_data$Histología))
```

```
# Obtener la tabla de frecuencias de la variable Histología
tabla_histologia <- table(clinic_full_data$Histología)

# Ajustar márgenes para darle más espacio al eje con las etiquetas largas (margen izquierdo)
par(mar = c(5, 12, 4, 2) + 0.1)

# Graficar un barplot horizontal
barplot(tabla_histologia,
        main = "Histología",
        xlab = "Frecuencia",
        horiz = TRUE,
        las = 1,         # las=1 para que el texto de las etiquetas se muestre horizontal
        cex.names = 0.8) # Ajusta el tamaño del texto de los nombres si es necesario
```



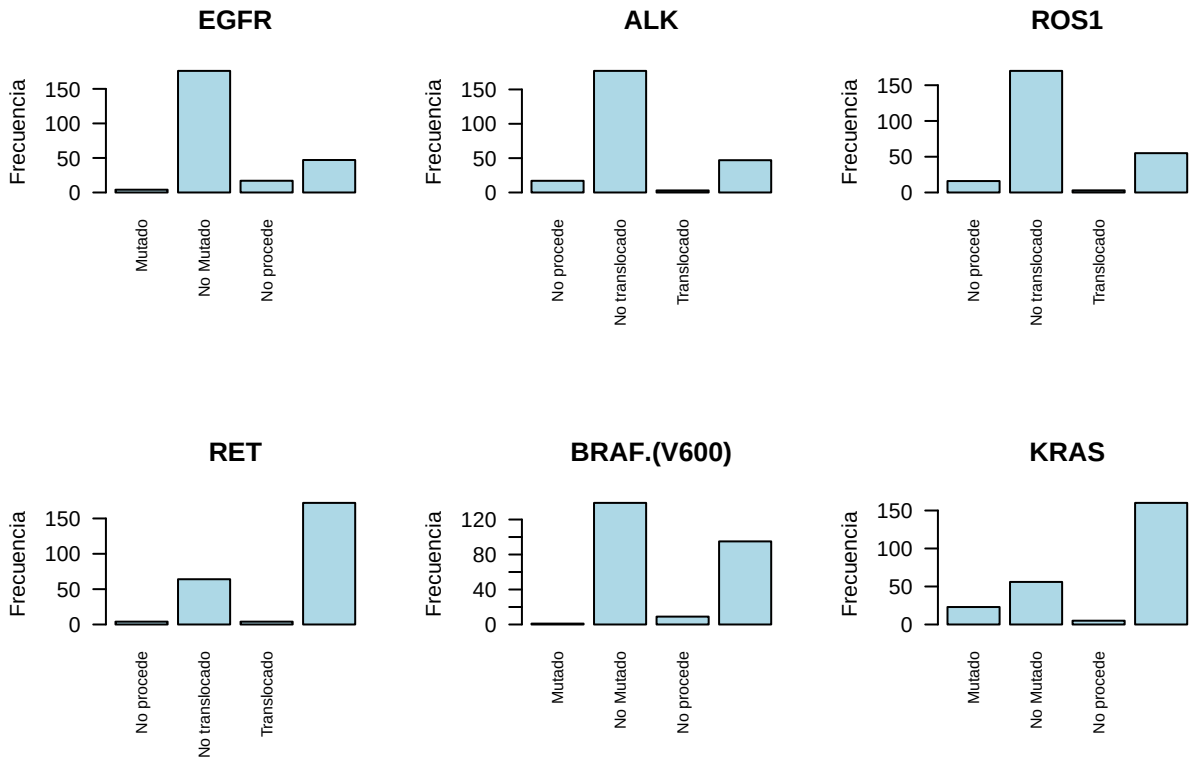
Mutaciones

Transformamos las variables de las mutaciones sustituyendo los registros "No dato" por `NA`.

```r
# Transformación de "No dato" a NA en cada variable
clinic_full_data$EGFR[clinic_full_data$EGFR == "No dato"] <- NA
clinic_full_data$ALK[clinic_full_data$ALK == "No dato"] <- NA
clinic_full_data$ROS1[clinic_full_data$ROS1 == "No dato"] <- NA
clinic_full_data$RET[clinic_full_data$RET == "No dato"] <- NA
clinic_full_data$`BRAF.(V600)`[clinic_full_data$`BRAF.(V600)` == "No dato"] <- NA
clinic_full_data$KRAS[clinic_full_data$KRAS == "No dato"] <- NA

# Vector con los nombres de las variables a convertir a factor
vars <- c("EGFR", "ALK", "ROS1", "RET", "BRAF.(V600)", "KRAS")

# Convertir cada variable de la lista a factor
clinic_full_data[vars] <- lapply(clinic_full_data[vars], factor)
```

Visualizamos las frecuencias.

```r
par(mfrow = c(2, 3), mar = c(8, 4, 4, 2) + 0.1)

for (v in vars) {
  # Crear tabla de frecuencias incluyendo NAs
  counts <- table(clinic_full_data[[v]], useNA = "ifany")

  # Graficar la tabla de frecuencias con barplot
  barplot(counts,
          main = v,
          ylab = "Frecuencia",
          col = "lightblue",
          border = "black",
          las = 2,         # Rota las etiquetas del eje x para mayor legibilidad
          cex.names = 0.8) # Ajusta el tamaño del texto en las etiquetas
}
```



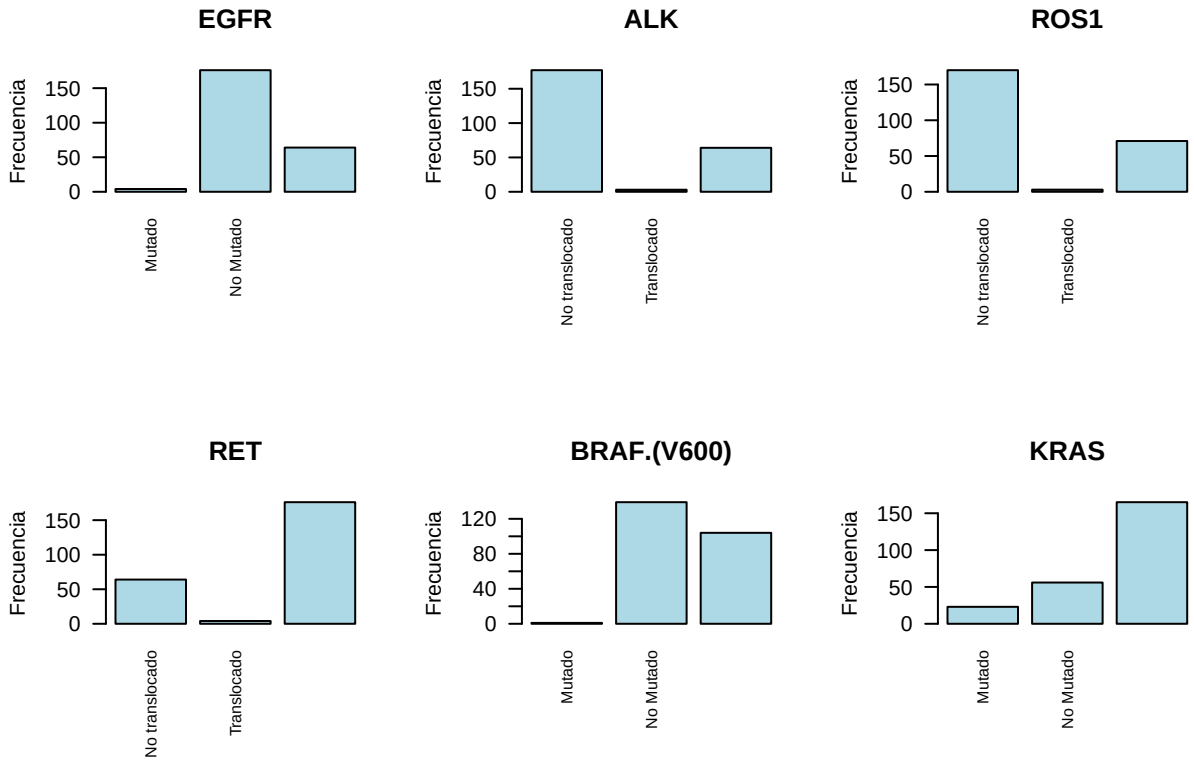
A continuación, realizaremos el mismo procedimiento sustituyendo "No procede" por `NA`.

```
clinic_full_data$EGFR[clinic_full_data$EGFR == "No procede"] <- NA
clinic_full_data$ALK[clinic_full_data$ALK == "No procede"] <- NA
clinic_full_data$ROS1[clinic_full_data$ROS1 == "No procede"] <- NA
clinic_full_data$RET[clinic_full_data$RET == "No procede"] <- NA
clinic_full_data$`BRAF.(V600)`[clinic_full_data$`BRAF.(V600)` == "No procede"] <- NA
clinic_full_data$KRAS[clinic_full_data$KRAS == "No procede"] <- NA

clinic_full_data$EGFR <- droplevels(clinic_full_data$EGFR)
clinic_full_data$ALK <- droplevels(clinic_full_data$ALK)
clinic_full_data$ROS1 <- droplevels(clinic_full_data$ROS1)
clinic_full_data$RET <- droplevels(clinic_full_data$RET)
clinic_full_data$`BRAF.(V600)` <- droplevels(clinic_full_data$`BRAF.(V600)`)
clinic_full_data$KRAS <- droplevels(clinic_full_data$KRAS)
```

Procedemos con la misma visualización y una tabla de frecuencias:

```
par(mfrow = c(2, 3), mar = c(8, 4, 4, 2) + 0.1)

for (v in vars) {
  # Crear tabla de frecuencias incluyendo NAs
  counts <- table(clinic_full_data[[v]], useNA = "ifany")

  # Graficar la tabla de frecuencias con barplot
  barplot(counts,
          main = v,
```

```
          ylab = "Frecuencia",
          col = "lightblue",
          border = "black",
          las = 2,          # Rota las etiquetas del eje x para mayor legibilidad
          cex.names = 0.8) # Ajusta el tamaño del texto en las etiquetas
}
```




```
# Crear una lista con tablas de frecuencia para cada variable, incluyendo los NA
freq_tables <- lapply(clinic_full_data[vars], function(x) table(x, useNA = "ifany"))

# Imprimir cada tabla de frecuencias
for (var_name in names(freq_tables)) {
  cat("Frecuencias para", var_name, ":\n")
  print(freq_tables[[var_name]])
  cat("\n")
}
```

```
## Frecuencias para EGFR :
## x
##    Mutado No Mutado      <NA>
##         4       176        64
##
## Frecuencias para ALK :
## x
## No translocado    Translocado           <NA>
##            177              3             64
##
```

```
## Frecuencias para ROS1 :
## x
## No translocado    Translocado           <NA>
##            170              3             71
##
## Frecuencias para RET :
## x
## No translocado    Translocado           <NA>
##             64              4            176
##
## Frecuencias para BRAF.(V600) :
## x
##    Mutado No Mutado      <NA>
##         1       139       104
##
## Frecuencias para KRAS :
## x
##    Mutado No Mutado      <NA>
##        23        56       165
```

Mutacion.general

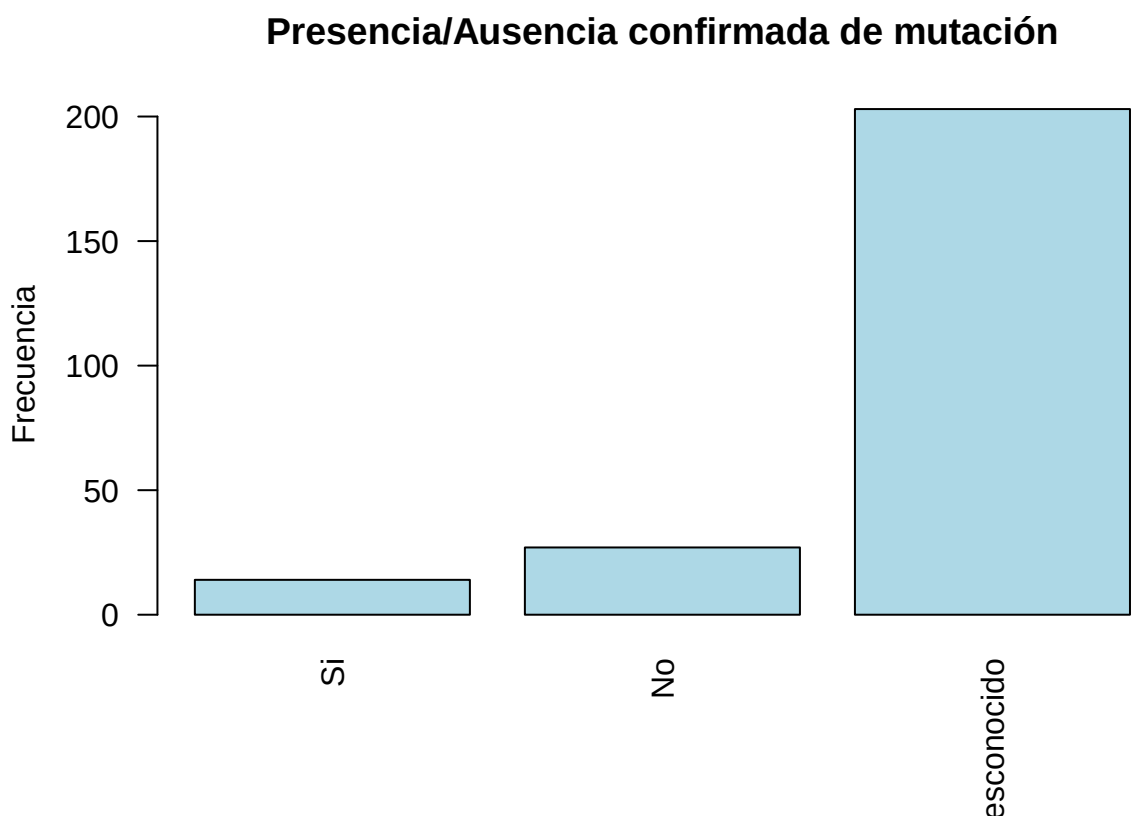
Creamos una nueva variable en la que recopilamos todas las mutaciones y factorizamos la presencia o ausencia de mutación.

```r
# Creamos una matriz con las condiciones, sin eliminar los NA
M <- cbind(
  clinic_full_data$EGFR == "Mutado",
  clinic_full_data$ALK == "Translocado",
  clinic_full_data$ROS1 == "Translocado",
  clinic_full_data$RET == "Translocado",
  clinic_full_data$`BRAF.(V600)` == "Mutado",
  clinic_full_data$KRAS == "Mutado"
)

# Al no especificar na.rm, si existe algún NA en la fila, rowSums devolverá NA.
clinic_full_data$Mutacion.General <- ifelse(rowSums(M) > 0, "Si", "No")
```

```r
# Reemplazamos los NA resultantes por "Desconocido"
clinic_full_data$Mutacion.General[is.na(clinic_full_data$Mutacion.General)] <- "Desconocido"

# Factorizamos incluyendo la categoría "Desconocido"
clinic_full_data$Mutacion.General <- factor(
  clinic_full_data$Mutacion.General,
  levels = c("Si", "No", "Desconocido")
)
```

```r
# Crear la tabla de frecuencias, incluyendo los NAs
tabla_mutacion <- table(clinic_full_data$Mutacion.General, useNA = "ifany")

# Graficar la tabla de frecuencias con barplot
barplot(tabla_mutacion,
        main = "Presencia/Ausencia confirmada de mutación",
        ylab = "Frecuencia",
        col = "lightblue",
```

```
        border = "black",
        las = 2)  # Rota las etiquetas del eje x si es necesario
```



Fecha.de.diagnóstico.de.enfermedad.metastática

Fecha en la que se dió el diagnóstico de metastasis

```
head(clinic_full_data$Fecha.de.diagnóstico.de.enfermedad.metastática)
```

```
## [1] "01/03/2018" "01/02/2014" "01/06/2021" "01/04/2019" "01/06/2021"
## [6] "01/07/2013"
```

```
class(clinic_full_data$Fecha.de.diagnóstico.de.enfermedad.metastática)
```

```
## [1] "character"
```

```
sum(is.na(clinic_full_data$Fecha.de.diagnóstico.de.enfermedad.metastática))
```

```
## [1] 11
```

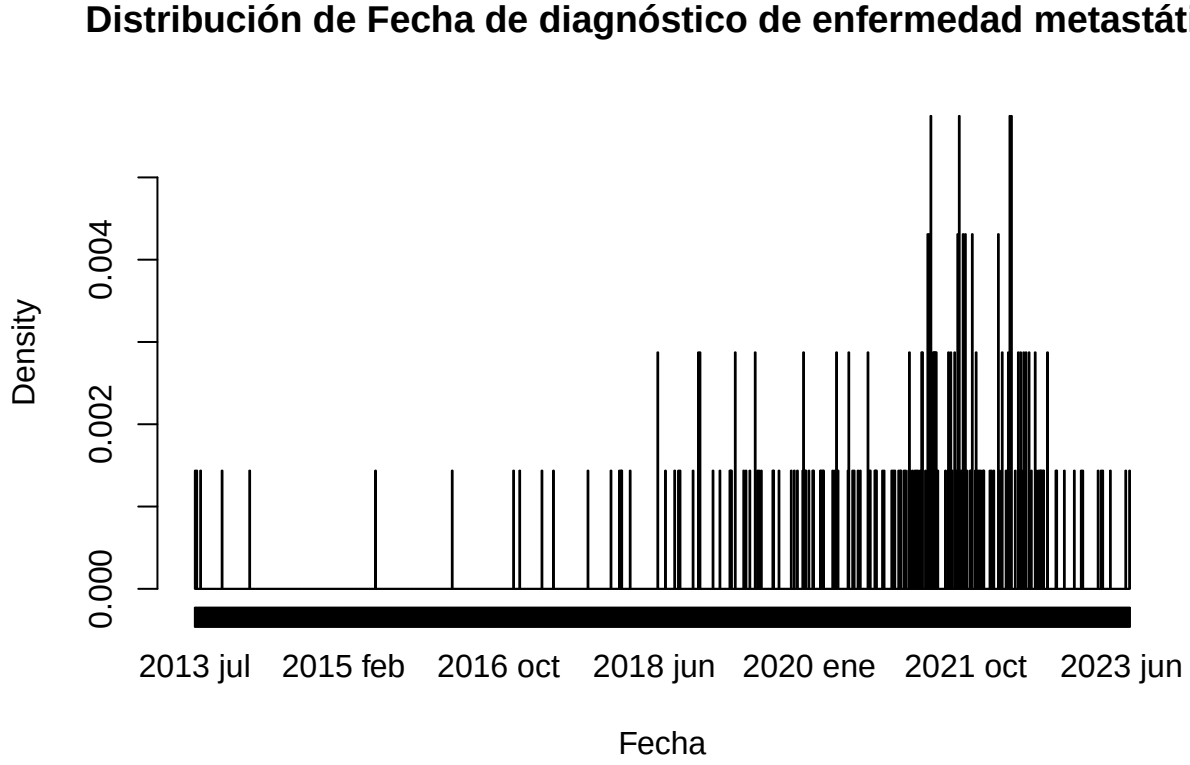
```
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Fecha.de.diagnóstico.de.enfermedad.metastáti
```

```
## Porcentaje de NAs: 4.51 %
```

Se propone su correcta conversión a formato de fecha

```
# Convertir la columna a Date
clinic_full_data$Fecha.de.diagnóstico.de.enfermedad.metastática <- as.Date(clinic_full_data$Fecha.de.dia

# Definir cortes
```

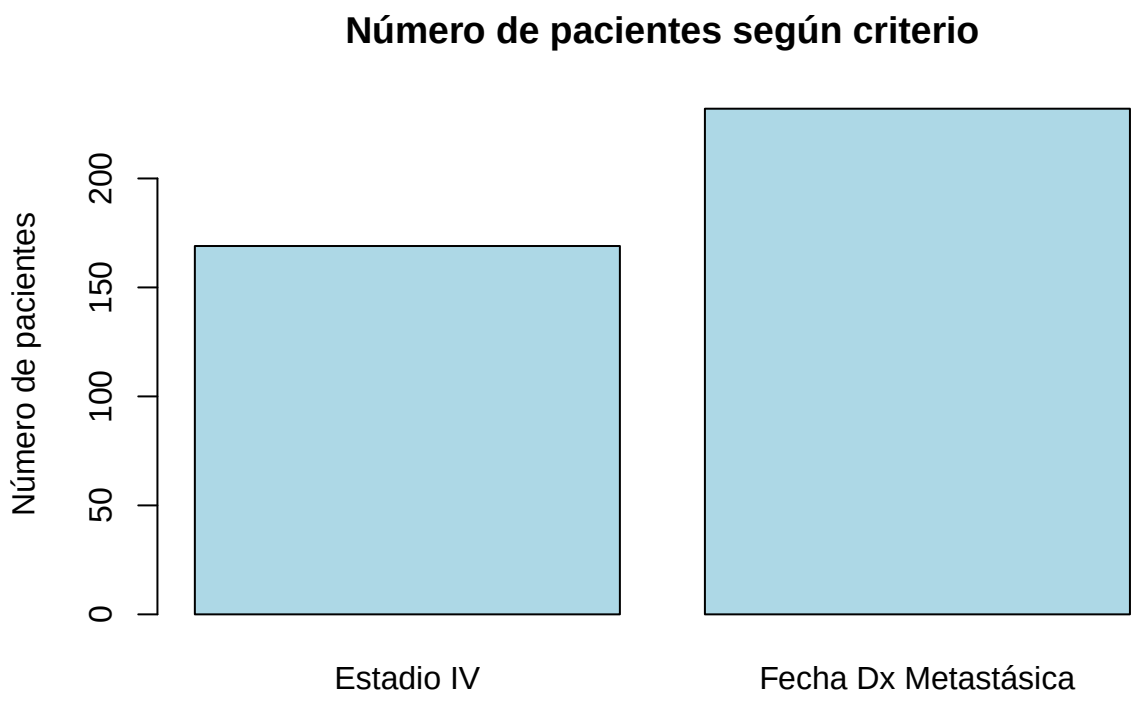
```r
breaks_seq <- seq(min(clinic_full_data$Fecha.de.diagnóstico.de.enfermedad.metastática, na.rm = TRUE),
                  max(clinic_full_data$Fecha.de.diagnóstico.de.enfermedad.metastática, na.rm = TRUE),
                  by = "3 days")

# Crear el histograma
hist(clinic_full_data$Fecha.de.diagnóstico.de.enfermedad.metastática,
     breaks = breaks_seq,
     main = "Distribución de Fecha de diagnóstico de enfermedad metastática",
     xlab = "Fecha",
     col = "skyblue",
     border = "black")
```



Para asegurarnos de que los registros tienen sentido vamos a visualizar la presencia o no fecha de diagnóstico, asumiendo que es que hay metastásis, junto con la variable de `Estadio.al.diagnóstico`. Concretamente con el estadío IV.

```r
# Contar pacientes con Estadio al diagnóstico de tipo IV (IVA o IVB)
num_estadioIV <- sum(clinic_full_data$`Estadio.al.diagnóstico` %in% c("IVA", "IVB"), na.rm = TRUE)

# Contar pacientes que tienen una fecha en Fecha de diagnóstico de enfermedad metastática (es decir, qu
num_fecha_metastasis <- sum(!is.na(clinic_full_data$`Fecha.de.diagnóstico.de.enfermedad.metastática`))

# Crear un vector con ambos conteos
counts <- c("Estadio IV" = num_estadioIV,
            "Fecha Dx Metastásica" = num_fecha_metastasis)
```

```r
# Graficar el resultado con un barplot sencillo
barplot(counts,
        main = "Número de pacientes según criterio",
        ylab = "Número de pacientes",
        col = "lightblue",
        border = "black")
```



En la gráfica vemos que casi todos los pacientes son que se les ha diagnosticado enfermedad metastásica, es decir que están en el estadío IV. Sin embargo, especificamente en el número de pacientes en estadío IV tenemos bastantes menos. Si queremos tener en cuenta el estadio del paciente, tal y como propuso Nadina, tenemos que esclarecer si fecha de diagnóstico de enfermedad con metástasis.

Quimioterapia.adyuvante

Indica el estado de la quimioterapia adyuvante en el paciente

```r
head(clinic_full_data$Quimioterapia.adyuvante)
```

```
## [1] "Si"        "No aplica" "No"        "No aplica" "No aplica" "No aplica"
```

```r
class(clinic_full_data$Quimioterapia.adyuvante)
```

```
## [1] "character"
```

```r
sum(is.na(clinic_full_data$Quimioterapia.adyuvante))
```

```
## [1] 1
```

```r
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Quimioterapia.adyuvante)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 0.41 %
```

Clases que tenemos dentro de la variable `Quimioterapia.adyuvante`:

```r
unique(clinic_full_data$Quimioterapia.adyuvante)
```

```
## [1] "Si"        "No aplica" "No"        NA
```

Se propone realizar una factorización de las clases. Considerando la categoría "No aplica" como `NA`.

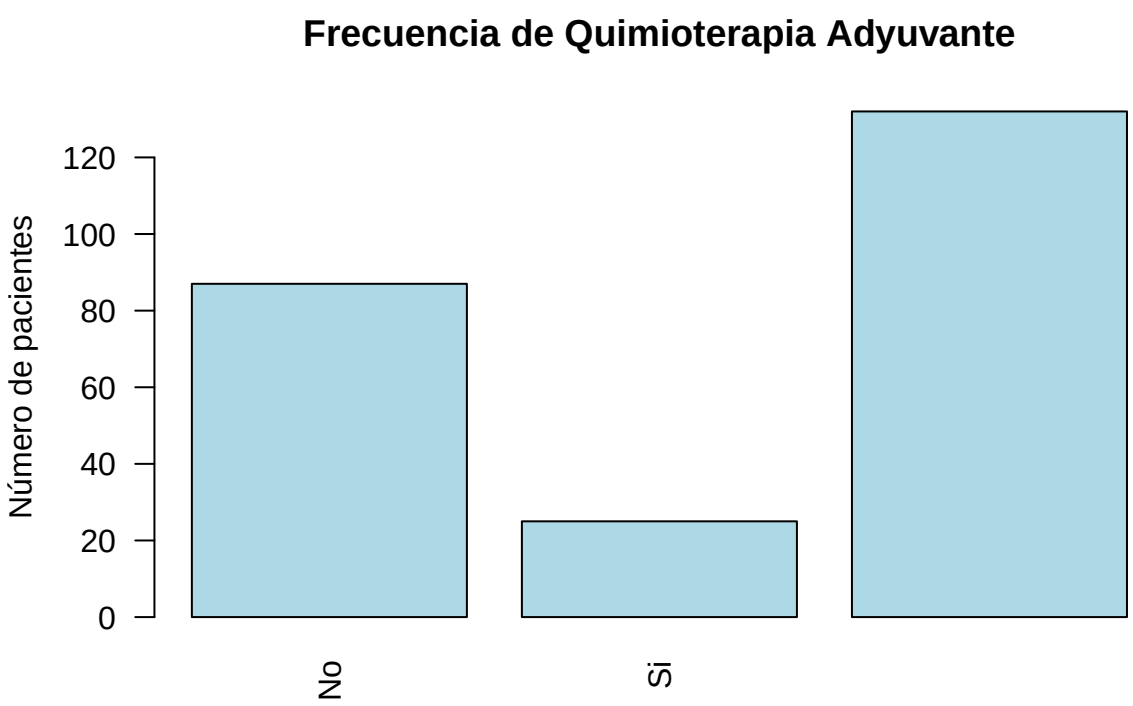
```r
# Reemplazar "No aplica" por NA en la variable Quimioterapia.adyuvante
clinic_full_data$Quimioterapia.adyuvante[clinic_full_data$Quimioterapia.adyuvante == "No aplica"] <- NA


# Contar el número y porcentaje de NAs
num_NA <- sum(is.na(clinic_full_data$Quimioterapia.adyuvante))
porcentaje_NA <- round(mean(is.na(clinic_full_data$Quimioterapia.adyuvante)) * 100, 2)
cat("Número de NAs:", num_NA, "\n")
```

```
## Número de NAs: 132
```

```r
cat("Porcentaje de NAs:", porcentaje_NA, "%\n")
```

```
## Porcentaje de NAs: 54.1 %
```

```r
# Crear una tabla de frecuencias que incluya los NAs
tabla_quimio <- table(clinic_full_data$Quimioterapia.adyuvante, useNA = "ifany")

# Graficar la tabla de frecuencias con barplot
barplot(tabla_quimio,
        main = "Frecuencia de Quimioterapia Adyuvante",
        ylab = "Número de pacientes",
        col = "lightblue",
        border = "black",
        las = 2)  # Rotar etiquetas si es necesario
```



Radioterapia.adyuvante

Indica el estado de la radioterapia adyuvante en el paciente.

Vemos las clases presentes en la variable

```
unique(clinic_full_data$Radioterapia.adyuvante)
```

```
## [1] "No aplica" "No"        "Si"
```

Reemplazar “No aplica” por NA en la variable Radioterapia.adyuvante

```
clinic_full_data$Radioterapia.adyuvante[clinic_full_data$Radioterapia.adyuvante == "No aplica"] <- NA
```

```
# Contar el número y porcentaje de NAs después de la transformación
num_NA <- sum(is.na(clinic_full_data$Radioterapia.adyuvante))
porcentaje_NA <- round(mean(is.na(clinic_full_data$Radioterapia.adyuvante)) * 100, 2)
cat("Número de NAs:", num_NA, "\n")
```

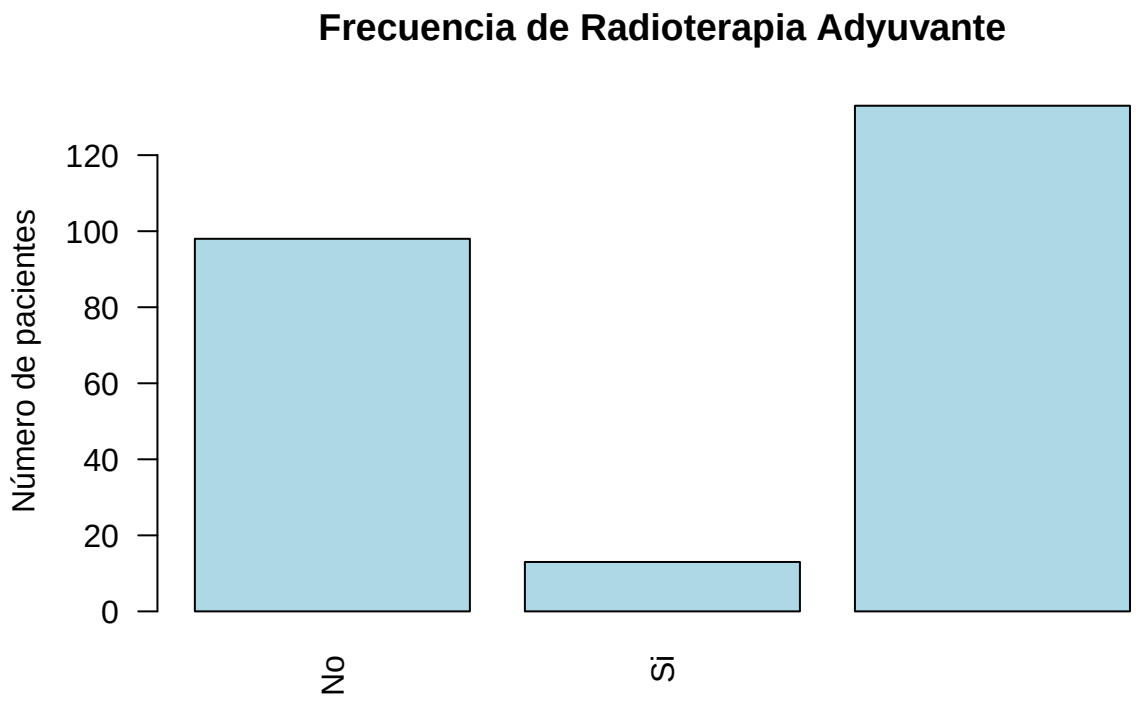
```
## Número de NAs: 133
```

```
cat("Porcentaje de NAs:", porcentaje_NA, "%\n")
```

```
## Porcentaje de NAs: 54.51 %
```

```
# Crear una tabla de frecuencias que incluya los NAs
tabla_radio <- table(clinic_full_data$Radioterapia.adyuvante, useNA = "ifany")

# Graficar la tabla de frecuencias con barplot
barplot(tabla_radio,
```

```
        main = "Frecuencia de Radioterapia Adyuvante",
        ylab = "Número de pacientes",
        col = "lightblue",
        border = "black",
        las = 2)  # Rotar etiquetas del eje x si es necesario
```



Quimio.Radio.Adj

Este bloque crea la variable `Quimio.Radio.Adj` combinando la información de `Quimioterapia.adyuvante` y `Radioterapia.adyuvante`:

- Asigna "Quimioterapia adyuvante" si solo la quimio es "Si",
- "Radioterapia adyuvante" si solo la radio es "Si",
- "Quimio y Radio adyuvante" si ambas son "Si",
- y "Desconocido/No aplica" para los casos en que ambas sean "No" o faltantes. Finalmente la factoriza con un orden lógico de niveles.

```
clinic_full_data <- clinic_full_data %>%
  mutate(
    Quimio.Radio.Adj = case_when(
      Quimioterapia.adyuvante == "Si" & Radioterapia.adyuvante == "Si" ~ "Quimio y Radio adyuvante",
      Quimioterapia.adyuvante == "Si"                                  ~ "Quimioterapia adyuvante",
      Radioterapia.adyuvante == "Si"                                   ~ "Radioterapia adyuvante",
      TRUE                                                             ~ "Desconocido/No aplica"
    ),
    Quimio.Radio.Adj = factor(
      Quimio.Radio.Adj,
      levels = c(
```

```
        "Quimioterapia adyuvante",
        "Radioterapia adyuvante",
        "Quimio y Radio adyuvante",
        "Desconocido/No aplica"
      )
    )
  ) %>% relocate(Quimio.Radio.Adj, .after = 2)
```

RT.QT.Radical

Indica el estado de RT.QT.Radical (la combinación de radioterapia y quimioterapia radical) en el paciente. Vemos las clases presentes en la variable:

```
unique(clinic_full_data$RT.QT.Radical)
```

```
## [1] "No aplica" "No"        "Si"        NA
```

Reemplazamos "No aplica" por NA en la variable RT.QT.Radical.

```
clinic_full_data$RT.QT.Radical[clinic_full_data$RT.QT.Radical == "No aplica"] <- NA
```

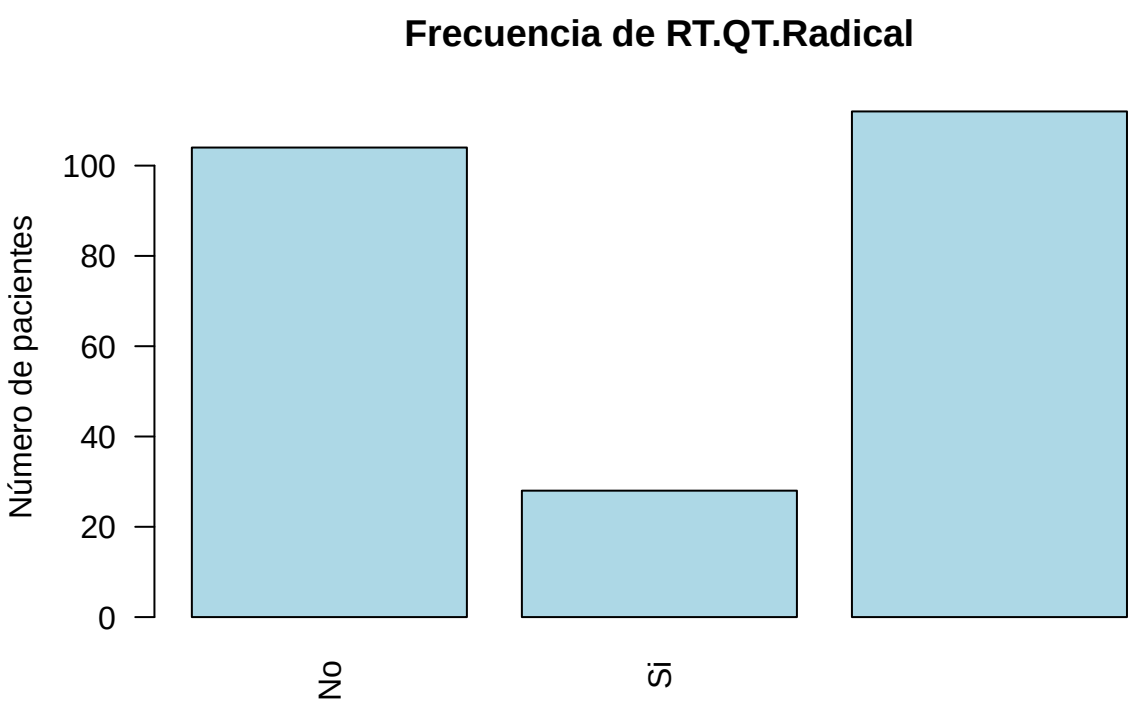
```
# Contar el número y porcentaje de NAs después de la transformación
num_NA <- sum(is.na(clinic_full_data$RT.QT.Radical))
porcentaje_NA <- round(mean(is.na(clinic_full_data$RT.QT.Radical)) * 100, 2)
cat("Número de NAs:", num_NA, "\n")
```

```
## Número de NAs: 112
```

```
cat("Porcentaje de NAs:", porcentaje_NA, "%\n")
```

```
## Porcentaje de NAs: 45.9 %
```

```
# Crear una tabla de frecuencias que incluya los NAs
tabla_radio <- table(clinic_full_data$RT.QT.Radical, useNA = "ifany")

# Graficar la tabla de frecuencias con barplot
barplot(tabla_radio,
        main = "Frecuencia de RT.QT.Radical",
        ylab = "Número de pacientes",
        col = "lightblue",
        border = "black",
        las = 2)  # Rotar etiquetas del eje x si es necesario
```



IO.adyuvante

Indica el estado de la inmunoterapia adyuvante en el paciente. Vemos las clases presentes en la variable:

```
unique(clinic_full_data$IO.adyuvante)
```

```
## [1] "No" "Sí" NA
```

Vemos que solo presenta dos clases, por lo que no hay que realizar transformación.

```
head(clinic_full_data$IO.adyuvante)
```

```
## [1] "No" "No" "No" "No" "No" "No"
```

```
class(clinic_full_data$IO.adyuvante)
```

```
## [1] "character"
```

```
cat("Número de NAs:", sum(is.na(clinic_full_data$IO.adyuvante)), "\n")
```

```
## Número de NAs: 2
```

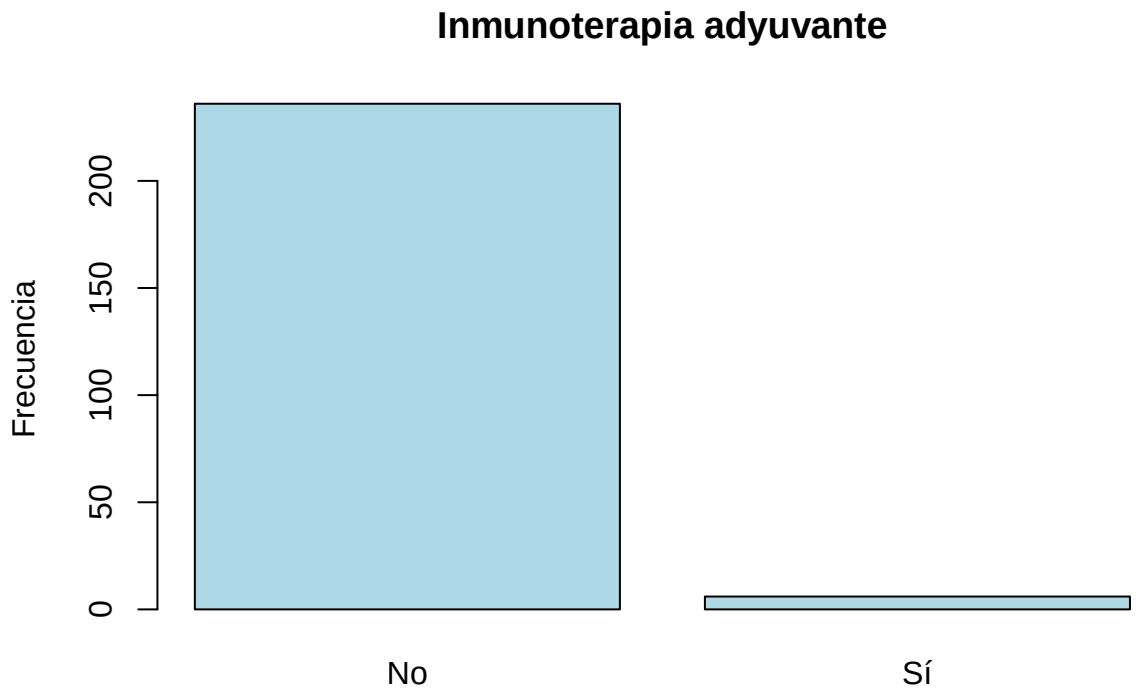
```
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$IO.adyuvante)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 0.82 %
```

Se propone realizar una factorización de las clases

```
clinic_full_data$IO.adyuvante <- factor(clinic_full_data$IO.adyuvante,
                                        levels = unique(clinic_full_data$IO.adyuvante))
```

```r
plot(clinic_full_data$IO.adyuvante,
     main = "Inmunoterapia adyuvante",
     ylab = "Frecuencia",
     col = "lightblue",
     border = "black")
```



Fecha.de.inicio.IO..metastáticos.

Fecha en la que se inició el tratamiento de IO (inmunoterapia o terapia oncológica) específicamente en pacientes con enfermedad metastásica. DE esta variable obtendremos una nueva variable obtenida a partir de la fecha de diagnostico de enfermedad metastásica.

```r
head(clinic_full_data$`Fecha.de.inicio.IO.(metastáticos)`)
```

```
## [1] "19/06/2018" "23/08/2019" "28/04/2023" "17/04/2019" "08/07/2021"
## [6] "21/08/2019"
```

```r
class(clinic_full_data$`Fecha.de.inicio.IO.(metastáticos)`)
```

```
## [1] "character"
```

```r
sum(is.na(clinic_full_data$`Fecha.de.inicio.IO.(metastáticos)`))
```

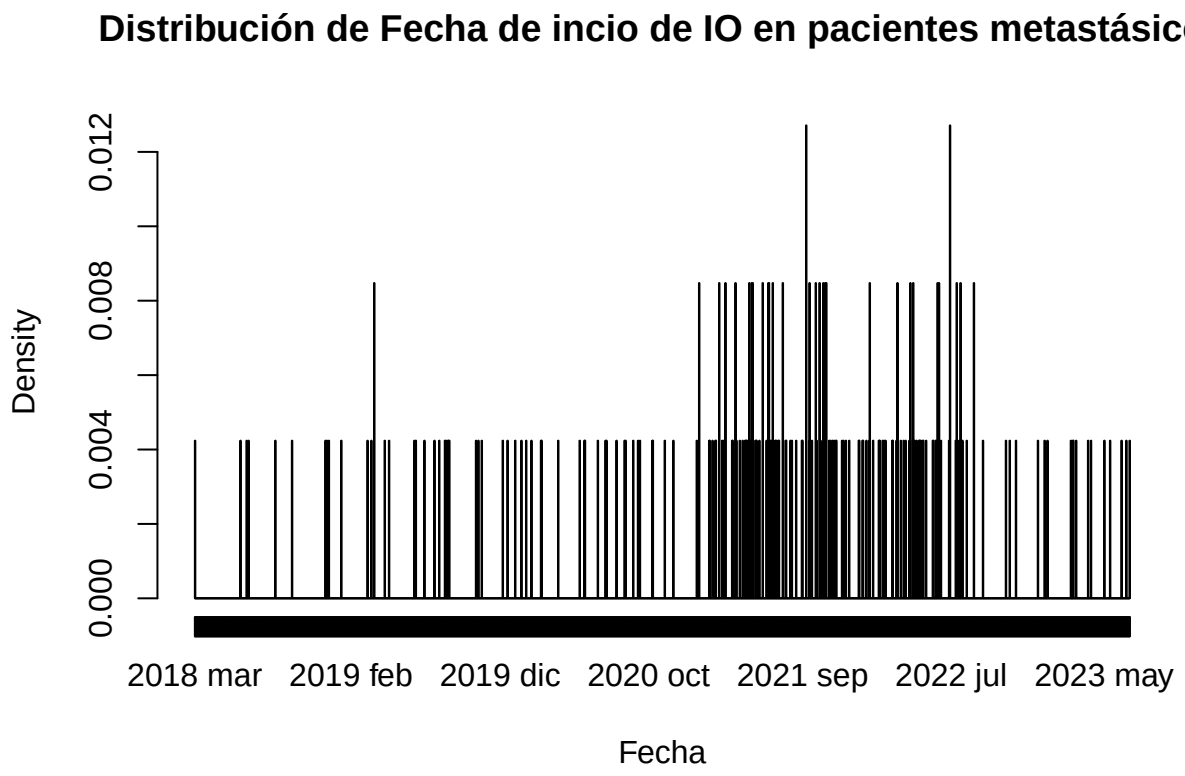
```
## [1] 8
```

```r
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$`Fecha.de.inicio.IO.(metastáticos)`)) * 100
```

```
## Porcentaje de NAs: 3.28 %
```

Se propone su correcta conversion a formato de fecha

```
# Convertir la columna a Date
clinic_full_data$`Fecha.de.inicio.IO.(metastáticos)` <- as.Date(clinic_full_data$`Fecha.de.inicio.IO.(m

# Definir cortes
breaks_seq <- seq(min(clinic_full_data$`Fecha.de.inicio.IO.(metastáticos)`, na.rm = TRUE),
                  max(clinic_full_data$`Fecha.de.inicio.IO.(metastáticos)`, na.rm = TRUE),
                  by = "1 days")

# Crear el histograma
hist(clinic_full_data$`Fecha.de.inicio.IO.(metastáticos)`,
     breaks = breaks_seq,
     main = "Distribución de Fecha de incio de IO en pacientes metastásicos",
     xlab = "Fecha",
     col = "skyblue",
     border = "black")
```



Tiempo.hasta.inicio.IO.metas

```
clinic_full_data$Tiempo.hasta.inicio.IO.metas <- as.numeric(
  clinic_full_data$`Fecha.de.inicio.IO.(metastáticos)` - clinic_full_data$`Fecha.de.diagnóstico.de.enfe
)
clinic_full_data %>% relocate(Tiempo.hasta.inicio.IO.metas, .after = 4)
```

```
## # A tibble: 244 x 173
##    Paciente  Paciente.fuera.del.estudio Quimio.Radio.Adj    Motivo.fuera.estudio
```

```
##    <chr>     <lgl>                    <fct>               <lgl>
##  1 PH-003-PU NA                       Quimioterapia adyu~ NA
##  2 PH-050-PU NA                       Desconocido/No apl~ NA
##  3 PH-191-PU NA                       Desconocido/No apl~ NA
##  4 PH-018-PU NA                       Desconocido/No apl~ NA
##  5 PC-003-PU NA                       Desconocido/No apl~ NA
##  6 PH-027-PU NA                       Desconocido/No apl~ NA
##  7 PH-102-PU NA                       Desconocido/No apl~ NA
##  8 PH-167-PU NA                       Radioterapia adyuv~ NA
##  9 UA-012-PU NA                       Desconocido/No apl~ NA
## 10 PH-138-PU NA                       Desconocido/No apl~ NA
## # i 234 more rows
## # i 169 more variables: Tiempo.hasta.inicio.IO.metas <dbl>, Comentarios <chr>,
## #   Fecha.de.inicio.IO <date>, Progresión...6 <fct>, Fecha.de.progresión <chr>,
## #   Tipo.de.Progresión <chr>, Evidencia.de.Progresión <chr>, EXITUS <dbl>,
## #   Fecha.Exitus <chr>, EXITUS...CAUSA.EXITUS <chr>,
## #   EXITUS...CAUSA.EXITUS...Otras.causas.Exitus <chr>,
## #   Fecha.última.visita <chr>, Progresión...15 <dbl>, `PFS.(d)` <dbl>, ...
```

```r
# Visualizar los primeros casos, la clase y el resumen de la nueva variable
head(clinic_full_data$Tiempo.hasta.inicio.IO.metas)
```

```
## [1]  110 2029  696   16   37 2242
```

```r
class(clinic_full_data$Tiempo.hasta.inicio.IO.metas)
```

```
## [1] "numeric"
```

```r
summary(clinic_full_data$Tiempo.hasta.inicio.IO.metas)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##     0.0    23.0    43.0   214.9   201.0  3116.0      15
```

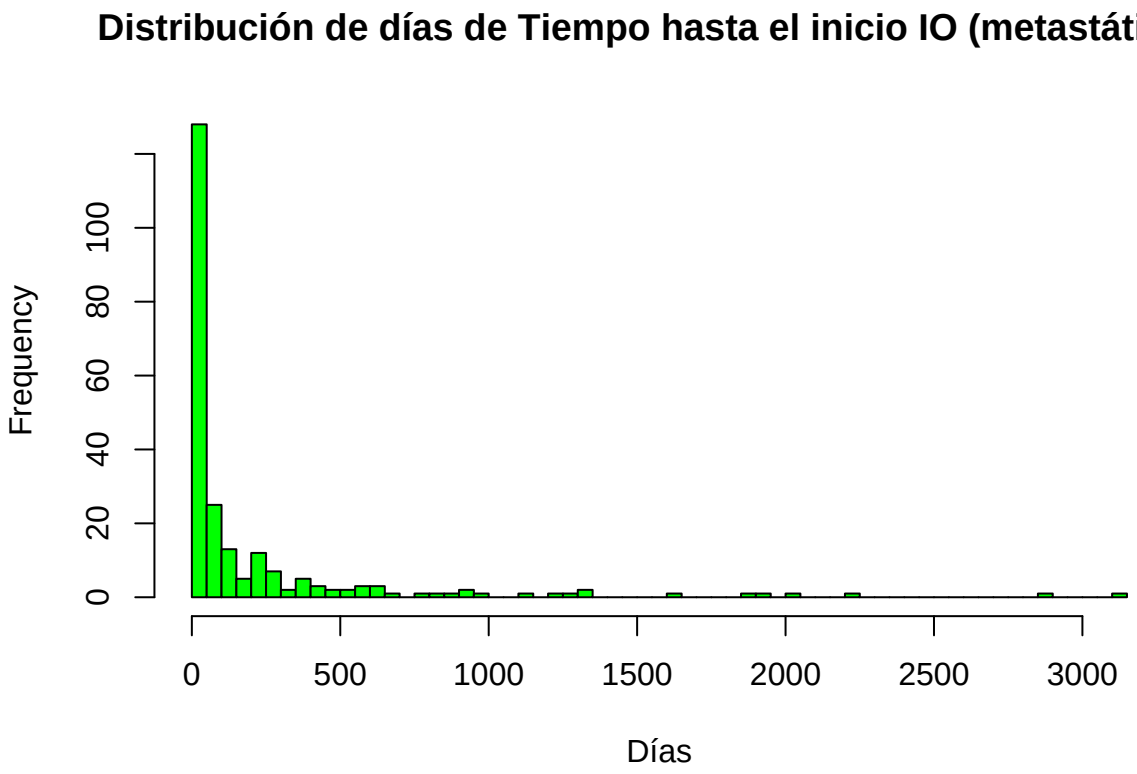
```r
cat("Número de NAs:", sum(is.na(clinic_full_data$Tiempo.hasta.inicio.IO.metas)), "\n")
```

```
## Número de NAs: 15
```

```r
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Tiempo.hasta.inicio.IO.metas)) * 100, 2), "%
```

```
## Porcentaje de NAs: 6.15 %
```

```r
# Graficar la distribución de la variable con un histograma
hist(clinic_full_data$Tiempo.hasta.inicio.IO.metas,
     breaks = 50,
     main = "Distribución de días de Tiempo hasta el inicio IO (metastáticos)",
     xlab = "Días",
     col = "green",
     border = "black")
```



TIPO.IO..metastáticos

Variable que indica el tipo de inmunoterapia realizada a los pacientes metastásicos

```
head(clinic_full_data$`TIPO.IO.(metastáticos)`)
```

```
## [1] "Inmunoterapia" "Inmunoterapia" "Inmunoterapia" "Inmunoterapia"
## [5] "Inmunoterapia" "Inmunoterapia"
```

```
class(clinic_full_data$`TIPO.IO.(metastáticos)`)
```

```
## [1] "character"
```

```
cat("Número de NAs:", sum(is.na(clinic_full_data$`TIPO.IO.(metastáticos)`)), "\n")
```

```
## Número de NAs: 7
```

```
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$`TIPO.IO.(metastáticos)`)) * 100, 2), "%\n"
```

```
## Porcentaje de NAs: 2.87 %
```

Clases que tenemos dentro de la variable `TIPO.IO..metastáticos.`:

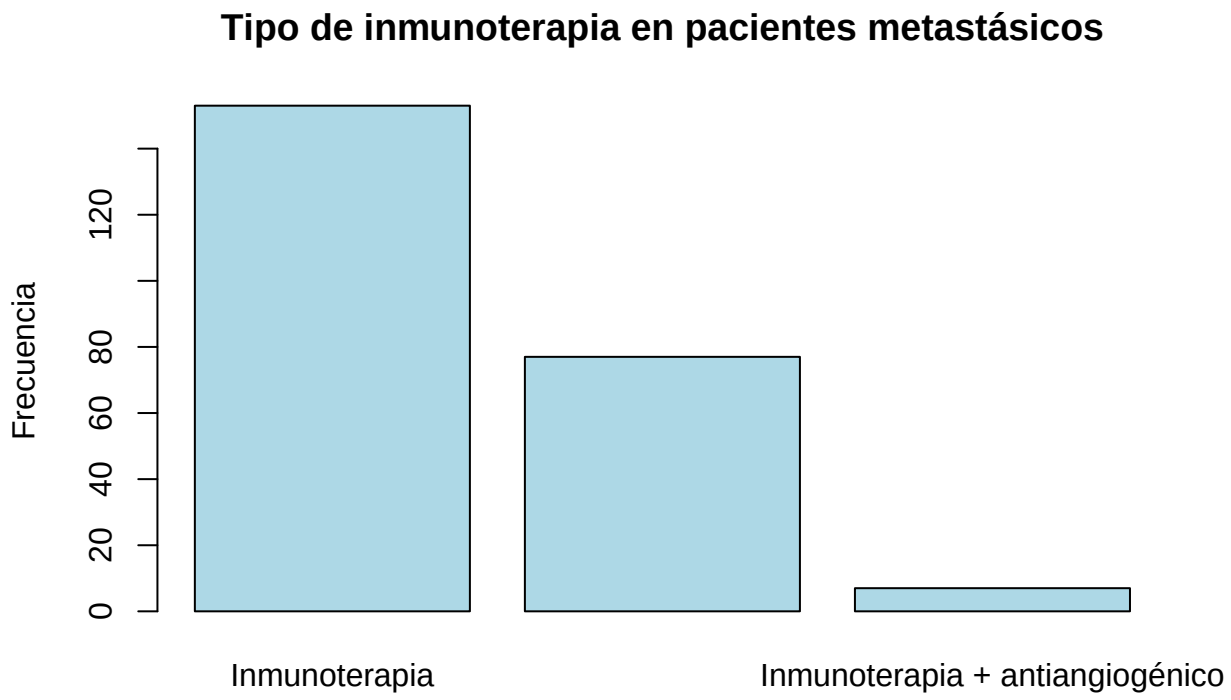
```
unique(clinic_full_data$`TIPO.IO.(metastáticos)`)
```

```
## [1] "Inmunoterapia"                   "Inmunoterapia + quimioterapia"
## [3] "Inmunoterapia + antiangiogénico" NA
```

Se propone realizar una factorización de las clases

```r
clinic_full_data$`TIPO.IO.(metastáticos)` <- factor(clinic_full_data$`TIPO.IO.(metastáticos)`,
                                                    levels = unique(clinic_full_data$`TIPO.IO.(metastátic
```

```r
plot(clinic_full_data$`TIPO.IO.(metastáticos)`,
     main = "Tipo de inmunoterapia en pacientes metastásicos",
     ylab = "Frecuencia",
     col = "lightblue",
     border = "black")
```



IO.Tipo.General

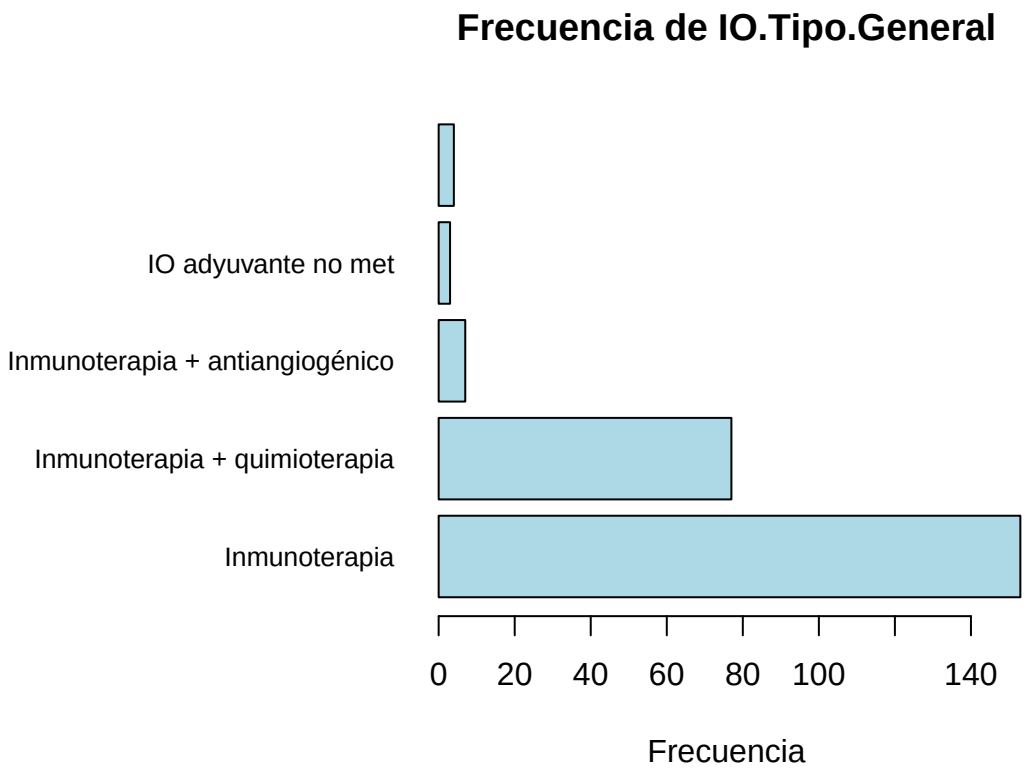
Variable que recopila a los pacientes no metastásicos que recibieron IO adyuvante junto con los pacientes metastásicos según tipo de IO.

```r
# Extraer las dos variables de interés:
io_met <- clinic_full_data$`TIPO.IO.(metastáticos)`
io_adyuv <- clinic_full_data$IO.adyuvante

# Crear la nueva variable.
# Si TIPO.IO.(metastáticos) es NA y IO.adyuvante es "Sí", asigna "IO adyuvante no met",
# de lo contrario, se usa el valor original de TIPO.IO.(metastáticos).
clinic_full_data$IO.Tipo.General <- ifelse(
  is.na(io_met) & io_adyuv == "Sí",
  "IO adyuvante no met",
  as.character(io_met)
)
```

```r
# Convertir la nueva variable a factor.
# Se toman los niveles existentes en TIPO.IO.(metastáticos) (sin incluir NA) y se añade el nuevo nivel.
current_levels <- levels(clinic_full_data$`TIPO.IO.(metastáticos)`)
new_levels <- c(current_levels, "IO adyuvante no met")
clinic_full_data$IO.Tipo.General <- factor(clinic_full_data$IO.Tipo.General, levels = new_levels)

# Verificar la tabla de frecuencias de la nueva variable, incluyendo NA
table(clinic_full_data$IO.Tipo.General, useNA = "ifany")
```

```
##
##                    Inmunoterapia   Inmunoterapia + quimioterapia
##                              153                              77
## Inmunoterapia + antiangiogénico             IO adyuvante no met
##                                7                               3
##                            <NA>
##                                4
```

```r
# Ajustar temporalmente los márgenes de la ventana gráfica
# Aumentamos especialmente el margen izquierdo (par(mar = c(bottom, left, top, right)))
old_par <- par(mar = c(5, 15, 4, 2) + 0.1)

# Crear la tabla de frecuencias
tabla_IO <- table(clinic_full_data$IO.Tipo.General, useNA = "ifany")

# Graficar el barplot horizontal
barplot(tabla_IO,
        main = "Frecuencia de IO.Tipo.General",
        xlab = "Frecuencia",
        horiz = TRUE,
        las = 1,         # las=1 para etiquetas horizontales en el eje y
        cex.names = 0.8, # Ajusta el tamaño del texto de las etiquetas (puedes disminuirlo aún más si e
        col = "lightblue",
        border = "black")
```




```r
# (Opcional) Restablecer los parámetros por defecto
par(old_par)
```

```r
cat("Número de NAs:", sum(is.na(clinic_full_data$IO.Tipo.General)), "\n",
    "Porcentaje de NAs:", round(mean(is.na(clinic_full_data$IO.Tipo.General)) * 100, 2), "%\n")
```

```
## Número de NAs: 4
##  Porcentaje de NAs: 1.64 %
```

Tipo.de.IO.Cat..0.IO..1.ChIO

Variable que indica si la inmunoterapia fue sola o acompañada de quimioterapia

```r
head(clinic_full_data$`Tipo.de.IO.Cat.(0=IO;.1=ChIO)`)
```

```
## [1] 0 0 0 0 0 0
```

```r
class(clinic_full_data$`Tipo.de.IO.Cat.(0=IO;.1=ChIO)`)
```

```
## [1] "numeric"
```

```r
cat("Número de NAs:", sum(is.na(clinic_full_data$`Tipo.de.IO.Cat.(0=IO;.1=ChIO)`)), "\n")
```

```
## Número de NAs: 14
```

```r
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$`Tipo.de.IO.Cat.(0=IO;.1=ChIO)`)) * 100, 2)
```

```
## Porcentaje de NAs: 5.74 %
```

Clases que tenemos dentro de la variable `Tipo.de.IO.Cat..0.IO..1.ChIO.`:

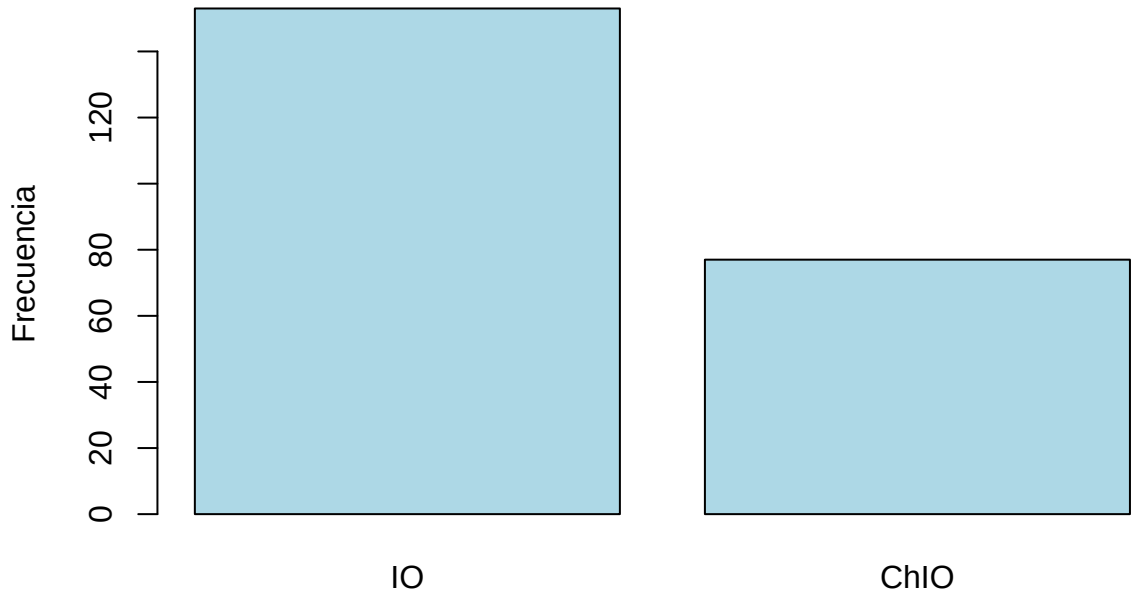
```r
unique(clinic_full_data$`Tipo.de.IO.Cat.(0=IO;.1=ChIO)`)
```

```
## [1]  0  1 NA
```

Se propone realizar una factorización de las clases

```r
clinic_full_data$`Tipo.de.IO.Cat.(0=IO;.1=ChIO)` <- factor(clinic_full_data$`Tipo.de.IO.Cat.(0=IO;.1=Ch
                                                    levels = c(0, 1),
                                                    labels = c("IO", "ChIO"))
```

```r
plot(clinic_full_data$`Tipo.de.IO.Cat.(0=IO;.1=ChIO)`,
     ylab = "Frecuencia",
     col = "lightblue",
     border = "black")
```



Este bloque recodifica la columna Tipo.de.IO.Cat.(0=IO;.1=ChIO) usando IO.Tipo.General, asignando tres niveles:

- "IO" para inmunoterapia sola,
- "ChIO" para inmunoterapia + quimioterapia,
- "IO combinado" para inmunoterapia + antiangiogénico o IO adyuvante en pacientes no metastásicos.

Luego factoriza esa misma variable con levels = c("IO","ChIO","IO combinado"), garantizando ese orden y dejando el resto en NA.

```r
clinic_full_data <- clinic_full_data %>%
  mutate(
    # Reconstruimos el factor original en 3 categorías
    `Tipo.de.IO.Cat.(0=IO;.1=ChIO)` = case_when(
```

```
      IO.Tipo.General == "Inmunoterapia"                      ~ "IO",
      IO.Tipo.General == "Inmunoterapia + quimioterapia"      ~ "ChIO",
      IO.Tipo.General %in% c("Inmunoterapia + antiangiogénico",
                             "IO adyuvante no met")           ~ "IO combinado",
      TRUE                                                     ~ NA_character_
    ),
    # Finalmente volvemos a factorizarlo con los 3 niveles
    `Tipo.de.IO.Cat.(0=IO;.1=ChIO)` = factor(
      `Tipo.de.IO.Cat.(0=IO;.1=ChIO)`,
      levels = c("IO", "ChIO", "IO combinado")
    )
  )
```

Diana.IO

Indica la diana de la terapia inmunológica aplicada

```
head(clinic_full_data$Diana.IO)
```

```
## [1] "PD-1"  "PD-L1" "PD-L1" "PD-1"  "PD-1"  "PD-1"
```

```
class(clinic_full_data$Diana.IO)
```

```
## [1] "character"
```

```
sum(is.na(clinic_full_data$Diana.IO))
```

```
## [1] 4
```

```
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Diana.IO)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 1.64 %
```

Clases que tenemos dentro de la variable `Diana.IO`:

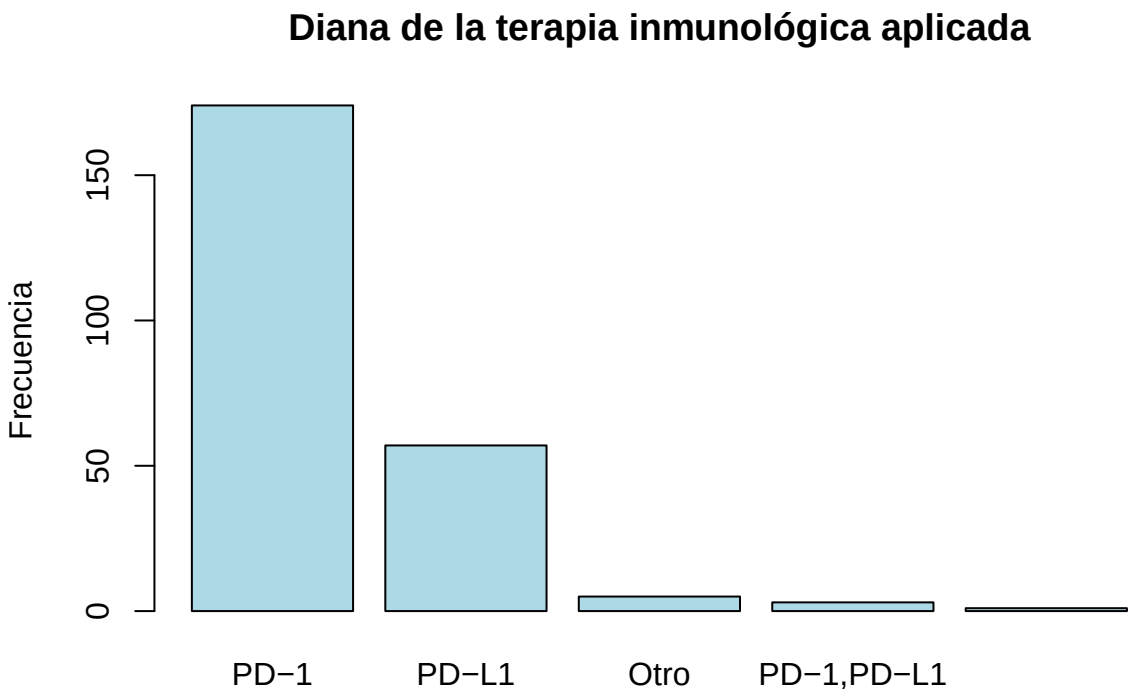
```
unique(clinic_full_data$Diana.IO)
```

```
## [1] "PD-1"       "PD-L1"      "Otro"       "PD-1,PD-L1" "PD-L1,Otro"
## [6] NA
```

Se propone realizar una factorización de las clases

```
clinic_full_data$Diana.IO <- factor(clinic_full_data$Diana.IO,
                                    levels = unique(clinic_full_data$Diana.IO))
```

```
plot(clinic_full_data$Diana.IO,
     main = "Diana de la terapia inmunológica aplicada",
     ylab = "Frecuencia",
     col = "lightblue",
     border = "black")
```



Nº.de.líneas.previas

Número de líneas de tratamiento que el paciente ha recibido antes del tratamiento actual. Usaremos esta variable para crear una nueva que indicará si el paciente ha recibido o no tratamiento previo.

```
head(clinic_full_data$Nº.de.líneas.previas)
```

```
## [1] 1 1 1 0 0 2
```

```
class(clinic_full_data$Nº.de.líneas.previas)
```

```
## [1] "numeric"
```

```
cat("Número de NAs:",sum(is.na(clinic_full_data$Nº.de.líneas.previas)), "\n")
```

```
## Número de NAs: 4
```

```
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Nº.de.líneas.previas)) * 100, 2), "%\n")
```

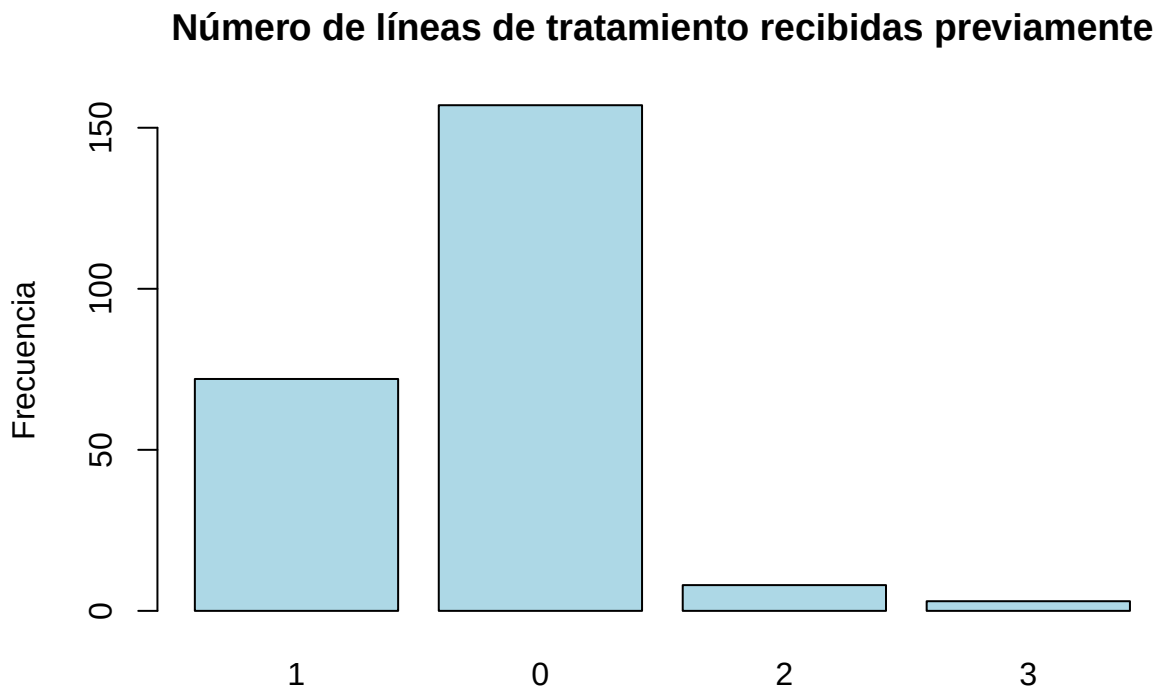
```
## Porcentaje de NAs: 1.64 %
```

Se propone realizar una factorización de las clases

```
clinic_full_data$Nº.de.líneas.previas <- factor(clinic_full_data$Nº.de.líneas.previas,
                                                levels = unique(clinic_full_data$Nº.de.líneas.previas)
```

```
plot(clinic_full_data$Nº.de.líneas.previas,
     main = "Número de líneas de tratamiento recibidas previamente",
     ylab = "Frecuencia",
     col = "lightblue",
```

```
     border = "black")
```



Presencia.linea.Previa

Variable obtenida a partir de la variable de numero de líneas previas. Nos indica si el paciente tiene o no tratamiento previo.

```
clinic_full_data$Presencia.linea.Previa <- ifelse(
  is.na(clinic_full_data$`Nº.de.líneas.previas`),
  NA,
  ifelse(as.numeric(as.character(clinic_full_data$`Nº.de.líneas.previas`)) > 0, "Si", "No")
)

# Convertir a factor y definir el orden de niveles, por ejemplo: No (antes) y Si (después)
clinic_full_data$Presencia.linea.Previa <- factor(clinic_full_data$Presencia.linea.Previa, levels = c("
```

```
head(clinic_full_data$Presencia.linea.Previa)
```

```
## [1] Si Si Si No No Si
## Levels: No Si
```

```
class(clinic_full_data$Presencia.linea.Previa)
```

```
## [1] "factor"
```

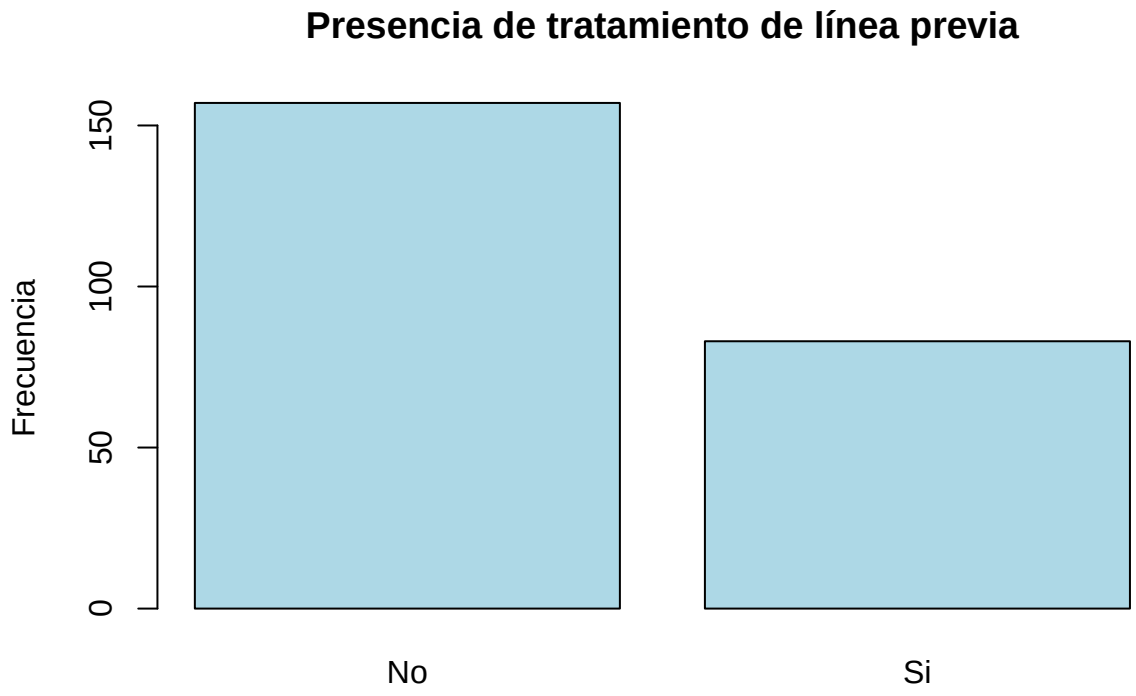
```
cat("Número de NAs:",sum(is.na(clinic_full_data$Presencia.linea.Previa)), "\n")
```

```
## Número de NAs: 4
```

```r
cat("Porcentaje de NAs:", round(mean(is.na(clinic_full_data$Presencia.linea.Previa)) * 100, 2), "%\n")
```

```
## Porcentaje de NAs: 1.64 %
```

```r
plot(clinic_full_data$Presencia.linea.Previa,
     main = "Presencia de tratamiento de línea previa",
     ylab = "Frecuencia",
     col = "lightblue",
     border = "black")
```



Eliminación de variables

Eliminamos las columnas discutudas en la reunión y consideradas no relevantes para el modelo

```r
clinic_full_data <- clinic_full_data %>%
  select(
    -Paciente.fuera.del.estudio,
    -Motivo.fuera.estudio,
    -Comentarios,
    -Fecha.de.progresión,
    -Tipo.de.Progresión,
    -Evidencia.de.Progresión,
    -Progresión...15,
    -Fecha.Exitus,
    -EXITUS,
    -EXITUS...CAUSA.EXITUS,
```

```
  -EXITUS...CAUSA.EXITUS...Otras.causas.Exitus,
  -Fecha.última.visita,
  -`OS.(d)`,           # aquí los backticks
  -`OS.(m)`,           # y aquí
  -`FU.(months)`,
  -Tipo.de.tumor.previo,
  -`EGFR...Qué.mutación.presenta?`,
  -`KRAS...Qué.mutación.presenta?`,
  -Quimioterapia.adyuvante...Fecha.inicio,
  -Quimioterapia.adyuvante...Fecha.fin,
  -Radioterapia.adyuvante...Dosis.de.radioterapia,
  -Radioterapia.adyuvante...Fecha.inicio,
  -Radioterapia.adyuvante...Fecha.fin,
  -RT.QT.Radical...Tipo.de.RT.QT,
  -RT.QT.Radical...Dosis.de.radioterapia,
  -RT.QT.Radical...Fecha.inicio.RT,
  -RT.QT.Radical...Fecha.fin.RT,
  -RT.QT.Radical...Fecha.inicio.QT,
  -RT.QT.Radical...Fecha.fin.QT,
  -IO.adyuvante...Tipo.de.IO,
  -IO.adyuvante...Fecha.inicio,
  -IO.adyuvante...Fecha.fin,
  -`Fármaco.de.IO.(metastáticos)`,
  -`Fármaco.de.IO.(metastáticos)...Especificar`,
  -IO.adyuvante,
  -Diana.IO...Especificar,
  -`Nº.de.líneas.previas...1ª.Línea`,
  -`Nº.de.líneas.previas...2ª.Linea`,
  -`Nº.de.líneas.previas...3ª.Línea`,
  -Tejido,
  -`Tejido...Tipo.de.tejido`,
  -Tejido...Microtomía,
  -`Tejido...Microtomía...Fecha.de.envío`,
  -Tejido...RNA,
  -`Tejido...RNA...Material.válido`,
  -`Tejido...RNA...Material.válido...Envío.a.Atrys`,
  -`Tejido...RNA...Fecha.extracción.RNA`,
  -`Tejido...RNA...Material.válido...Envío.a.Atrys...Fecha.envío.RNA`,
  -`Tejido...RNA...Material.válido...Envío.a.Atrys...TCR.Secuenciado`,
  -`Tejido...IHQ`,
  -`BS1.Fecha.1ª.extracción...83`,
  -`BS1.Fecha.1ª.extracción...84`,
  -`BS1.Fecha.recepción.al.laboratorio`,
  -BS1.Procesamiento,
  -BS1.Plasma,
  -`BS1.Plasma...Citoquinas`,
  -`BS1.Plasma...Extracción.cfDNA`,
  -`BS1.Plasma...Extracción.cfDNA...NGS`,
  -BS1.PBMCs,
  -`BS1.PBMCs...Número.de.viales.de.PBMCs`,
  -`BS1.PBMCs...Citometría.de.flujo`,
  -`BS1.PBMCs...Citometría.de.flujo...Inclusión.en.estadística`,
  -BS1.PBMCs...RNA,
```

```
  -`BS1.PBMCs...RNA...Fecha.de.extracción`,
  -`BS1.PBMCs...RNA...Material.válido`,
  -`BS1.PBMCs...RNA...Material.válido...Envío.Atrys`,
  -`BS1.PBMCs...RNA...Material.válido...Envío.Atrys...Fecha.de.envío`,
  -`BS1.PBMCs...RNA...Material.válido...Envío.Atrys...TCR`,
  -`BS1.PBMCs...RNA...Material.válido...Envío.Leitat`,
  -`BS1.PBMCs...RNA...Material.válido...Envío.Leitat...Librería.B`,
  -BS1.PBMCs...DNA,
  -`BS1.PBMCs...DNA...Fecha.de.extracción`,
  -`BS1.PBMCs...DNA...Material.válido`,
  -`BS1.PBMCs...DNA...Material.válido...Envío.Atrys`,
  -`BS1.PBMCs...DNA...Material.válido...Envío.Atrys...Fecha.de.envío`,
  -`BS1.PBMCs...DNA...Material.válido...Envío.Atrys...SNPs`,
  -BS2,
  -`BS2.Fecha.2ª.extracción`,
  -`BS2.Fecha.recepción.al.laboratorio`,
  -BS2.Procesamiento,
  -BS2.Plasma,
  -`BS2.Plasma...Citoquinas`,
  -`BS2.Plasma...Extracción.cfDNA`,
  -`BS2.Plasma...Extracción.cfDNA...NGS`,
  -BS2.PBMCs,
  -`BS2.PBMCs...Número.de.viales.de.PBMCs`,
  -`BS2.PBMCs...Citometría.de.flujo`,
  -`BS2.PBMCs...Citometría.de.flujo...Inclusión.en.estadística`,
  -BS2.PBMCs...RNA,
  -`BS2.PBMCs...RNA...Fecha.de.extracción`,
  -`BS2.PBMCs...RNA...Material.válido`,
  -`BS2.PBMCs...RNA...Material.válido...Envío.Atrys`,
  -`BS2.PBMCs...RNA...Material.válido...Envío.Atrys...Fecha.de.envío`,
  -`BS2.PBMCs...RNA...Material.válido...Envío.Atrys...TCR`,
  -`BS2.PBMCs...RNA...Material.válido...Envío.Leitat`,
  -`BS2.PBMCs...RNA...Material.válido...Envío.Leitat...Librería.B`,
  -BS2.PBMCs...DNA,
  -`BS2.PBMCs...DNA...Fecha.de.extracción`,
  -`BS2.PBMCs...DNA...Material.válido`,
  -`BS2.PBMCs...DNA...Material.válido...Envío.Atrys`,
  -`BS2.PBMCs...DNA...Material.válido...Envío.Atrys...Fecha.de.envío`,
  -`BS2.PBMCs...DNA...Material.válido...Envío.Atrys...SNPs`,
  -BS3,
  -`BS3.Fecha.3ª.extracción`,
  -BS3.Evento,
  -`BS3.Fecha.recepción.al.laboratorio`,
  -BS3.Procesamiento,
  -BS3.Plasma,
  -`BS3.Plasma...Citoquinas`,
  -`BS3.Plasma...Extracción.cfDNA`,
  -`BS3.Plasma...Extracción.cfDNA...NGS`,
  -BS3.PBMCs,
  -`BS3.PBMCs...Número.de.viales.de.PBMCs`,
  -`BS3.PBMCs...Citometría.de.flujo`,
  -`BS3.PBMCs...Citometría.de.flujo...Inclusión.en.estadística`,
  -BS3.PBMCs...RNA,
```

```
    -`BS3.PBMCs...RNA...Fecha.de.extracción`,
    -`BS3.PBMCs...RNA...Material.válido`,
    -`BS3.PBMCs...RNA...Material.válido...Envío.Atrys`,
    -`BS3.PBMCs...RNA...Material.válido...Envío.Atrys...Fecha.de.envío`,
    -`BS3.PBMCs...RNA...Material.válido...Envío.Atrys...TCR`,
    -`BS3.PBMCs...RNA...Material.válido...Envío.Leitat`,
    -`BS3.PBMCs...RNA...Material.válido...Envío.Leitat...Librería.B`,
    -`BS3.PBMCs...DNA`,
    -`BS3.PBMCs...DNA...Fecha.de.extracción`,
    -`BS3.PBMCs...DNA...Material.válido`,
    -`BS3.PBMCs...DNA...Material.válido...Envío.Atrys`,
    -`BS3.PBMCs...DNA...Material.válido...Envío.Atrys...Fecha.de.envío`,
    -`BS3.PBMCs...DNA...Material.válido...Envío.Atrys...SNPs`,
    -`BS1.BS2.BS3`,
    -`BS1+BS2+BS3`,
    -`BS1+BS3`,
    -`BS2+BS3`,
    -Código.Inmunosight,
    -Edad.ultima.actualizacion
  )
```

Introducción de la columna Paciente.code

```
# Definimos el vector:
map <- c("01"="PH", "02"="LE", "03"="SO", "04"="FA",
         "05"="IL", "06"="PC", "07"="CG", "08"="SL",
         "09"="LA", "10"="X",  "11"="HB", "12"="UC",
         "13"="UA")

# Invertimos el map para buscar por valor:
inv_map <- setNames(names(map), map)
#   inv_map["PH"] == "01", inv_map["LE"] == "02", etc.

# 2. Extraemos los componentes de la columna Paciente
#    a) Prefijo (antes del primer "-")
prefijo <- sub("-.*", "", clinic_full_data$Paciente)

#    b) Código numérico (entre el primer y segundo "-")
codigo  <- sub("^[^-]+-([0-9]+)-.*$", "\\1", clinic_full_data$Paciente)

# 3. Construimos Paciente.code
clinic_full_data$Paciente.code <- paste0(
  inv_map[prefijo],  # convierte PH → "01", LE → "02", …
  "-",
  codigo             # deja el "001", "011", …
)

# 4. Verificamos un par de ejemplos rápidos
head(clinic_full_data[, c("Paciente", "Paciente.code")])
```

```
## # A tibble: 6 x 2
##   Paciente  Paciente.code
##   <chr>     <chr>
```

```
## 1 PH-003-PU 01-003
## 2 PH-050-PU 01-050
## 3 PH-191-PU 01-191
## 4 PH-018-PU 01-018
## 5 PC-003-PU 06-003
## 6 PH-027-PU 01-027
```

Exportado de datos clinicos

```r
write.csv(clinic_full_data, file = "/home/agombau/modelo_pipeline/procesed_data/clinic_full_data.csv")
```

Tabla de resumen EDA

```r
# Calcular la tabla de métricas de nearZeroVar para el dataset
nzv_table <- nearZeroVar(clinic_full_data, saveMetrics = TRUE)

# Extraer la columna 'nzv' (TRUE/FALSE) para cada variable
near_zero <- sapply(names(clinic_full_data), function(var) {
  if (var %in% rownames(nzv_table)) {
    nzv_table[var, "nzv"]
  } else {
    NA
  }
})

# Crear la tabla resumen sin la columna de Percent_Variance y añadiendo nearZeroVar
summary_table <- data.frame(
  Variable = names(clinic_full_data),
  Type = sapply(clinic_full_data, function(x) paste(class(x), collapse = ", ")),
  n_obs = sapply(clinic_full_data, length),
  n_NAs = sapply(clinic_full_data, function(x) sum(is.na(x))),
  Percent_NAs = sapply(clinic_full_data, function(x) round(mean(is.na(x)) * 100, 2)),
  Levels = sapply(clinic_full_data, function(x) if (is.factor(x)) paste(levels(x), collapse = "; ") else
  Factor_Frequencies = sapply(clinic_full_data, function(x) {
    if (is.factor(x)) {
      freqs <- table(x, useNA = "ifany")
      paste(paste(names(freqs), as.vector(freqs), sep = ": "), collapse = "; ")
    } else {
      NA
    }
  }),
  nearZeroVar = near_zero,
  Sample = sapply(clinic_full_data, function(x) paste(head(as.character(x), 3), collapse = "; ")),
  Min = sapply(clinic_full_data, function(x) if (is.numeric(x)) round(min(x, na.rm = TRUE), 2) else NA)
  Median = sapply(clinic_full_data, function(x) if (is.numeric(x)) round(median(x, na.rm = TRUE), 2) el
  Max = sapply(clinic_full_data, function(x) if (is.numeric(x)) round(max(x, na.rm = TRUE), 2) else NA)
  stringsAsFactors = FALSE
)

summary_table
```

```
##
```

```
## Paciente
## Quimio.Radio.Adj
## Fecha.de.inicio.IO
## Progresión...6
## PFS.(d)
## PFS.(m)
## Exitus
## Follow.Up.(d)
## Fecha.de.nacimiento
## Sexo.(0=Mujer;.1=Varón)                                                                                                  Se
## Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo) Tabaquismo.(0=Nunca.fumador;.1=Exfumado
## Tumor.previo
## Escala.ECOG
## Fecha.de.diagnóstico
## Edad.al.diagnóstico
## Estadio.al.diagnóstico                                                                                                   Es
## Histología
## EGFR
## ALK
## ROS1
## RET
## BRAF.(V600)
## KRAS
## PD-L1
## Fecha.de.diagnóstico.de.enfermedad.metastática                                  Fecha.de.diagnóstico.de.en
## Quimioterapia.adyuvante                                                                                                  Qui
## Radioterapia.adyuvante                                                                                                   Ra
## RT.QT.Radical
## Fecha.de.inicio.IO.(metastáticos)                                                                      Fecha.de.inic
## TIPO.IO.(metastáticos)                                                                                                   TI
## Tipo.de.IO.Cat.(0=IO;.1=ChIO)                                                                             Tipo.de.I
## Diana.IO
## Nº.de.líneas.previas
## BS1+BS2
## TERAPIA.BIOLÓGICA
## Tiempo.hasta.inicio.IO                                                                                                   Ti
## Mutacion.General
## Tiempo.hasta.inicio.IO.metas                                                                                  Tiempo.h
## IO.Tipo.General
## Presencia.linea.Previa                                                                                                   Pr
## Paciente.code
##                                                                    Type
## Paciente                                                       character
## Quimio.Radio.Adj                                                  factor
## Fecha.de.inicio.IO                                                  Date
## Progresión...6                                                    factor
## PFS.(d)                                                          numeric
## PFS.(m)                                                          numeric
## Exitus                                                            factor
## Follow.Up.(d)                                                    numeric
## Fecha.de.nacimiento                                                 Date
## Sexo.(0=Mujer;.1=Varón)                                           factor
## Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)      factor
## Tumor.previo                                                      factor
```

```
## Escala.ECOG                                                          factor
## Fecha.de.diagnóstico                                                   Date
## Edad.al.diagnóstico                                                 numeric
## Estadio.al.diagnóstico                                    ordered, factor
## Histología                                                           factor
## EGFR                                                                 factor
## ALK                                                                  factor
## ROS1                                                                 factor
## RET                                                                  factor
## BRAF.(V600)                                                          factor
## KRAS                                                                 factor
## PD-L1                                                               numeric
## Fecha.de.diagnóstico.de.enfermedad.metastática                         Date
## Quimioterapia.adyuvante                                          character
## Radioterapia.adyuvante                                           character
## RT.QT.Radical                                                    character
## Fecha.de.inicio.IO.(metastáticos)                                      Date
## TIPO.IO.(metastáticos)                                             factor
## Tipo.de.IO.Cat.(0=IO;.1=ChIO)                                      factor
## Diana.IO                                                           factor
## Nº.de.líneas.previas                                               factor
## BS1+BS2                                                          character
## TERAPIA.BIOLÓGICA                                                character
## Tiempo.hasta.inicio.IO                                             numeric
## Mutacion.General                                                   factor
## Tiempo.hasta.inicio.IO.metas                                       numeric
## IO.Tipo.General                                                    factor
## Presencia.linea.Previa                                             factor
## Paciente.code                                                    character
##                                                                  n_obs n_NAs
## Paciente                                                           244     0
## Quimio.Radio.Adj                                                   244     0
## Fecha.de.inicio.IO                                                 244     8
## Progresión...6                                                     244     0
## PFS.(d)                                                            244     9
## PFS.(m)                                                            244     9
## Exitus                                                             244     0
## Follow.Up.(d)                                                      244     9
## Fecha.de.nacimiento                                                244     0
## Sexo.(0=Mujer;.1=Varón)                                            244     0
## Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)       244     1
## Tumor.previo                                                       244     1
## Escala.ECOG                                                        244     8
## Fecha.de.diagnóstico                                               244     2
## Edad.al.diagnóstico                                                244     1
## Estadio.al.diagnóstico                                             244     7
## Histología                                                         244    18
## EGFR                                                               244    64
## ALK                                                                244    64
## ROS1                                                               244    71
## RET                                                                244   176
## BRAF.(V600)                                                        244   104
## KRAS                                                               244   165
## PD-L1                                                              244    12
```

```
## Fecha.de.diagnóstico.de.enfermedad.metastática                 244    12
## Quimioterapia.adyuvante                                        244   132
## Radioterapia.adyuvante                                         244   133
## RT.QT.Radical                                                  244   112
## Fecha.de.inicio.IO.(metastáticos)                              244     8
## TIPO.IO.(metastáticos)                                         244     7
## Tipo.de.IO.Cat.(0=IO;.1=ChIO)                                  244     4
## Diana.IO                                                       244     4
## Nº.de.líneas.previas                                           244     4
## BS1+BS2                                                        244   159
## TERAPIA.BIOLÓGICA                                              244     0
## Tiempo.hasta.inicio.IO                                         244    10
## Mutacion.General                                               244     0
## Tiempo.hasta.inicio.IO.metas                                   244    15
## IO.Tipo.General                                                244     4
## Presencia.linea.Previa                                         244     4
## Paciente.code                                                  244     0
##                                                           Percent_NAs
## Paciente                                                         0.00
## Quimio.Radio.Adj                                                 0.00
## Fecha.de.inicio.IO                                               3.28
## Progresión...6                                                   0.00
## PFS.(d)                                                          3.69
## PFS.(m)                                                          3.69
## Exitus                                                           0.00
## Follow.Up.(d)                                                    3.69
## Fecha.de.nacimiento                                              0.00
## Sexo.(0=Mujer;.1=Varón)                                          0.00
## Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)     0.41
## Tumor.previo                                                     0.41
## Escala.ECOG                                                      3.28
## Fecha.de.diagnóstico                                             0.82
## Edad.al.diagnóstico                                              0.41
## Estadio.al.diagnóstico                                           2.87
## Histología                                                       7.38
## EGFR                                                            26.23
## ALK                                                             26.23
## ROS1                                                            29.10
## RET                                                             72.13
## BRAF.(V600)                                                     42.62
## KRAS                                                            67.62
## PD-L1                                                            4.92
## Fecha.de.diagnóstico.de.enfermedad.metastática                   4.92
## Quimioterapia.adyuvante                                         54.10
## Radioterapia.adyuvante                                          54.51
## RT.QT.Radical                                                   45.90
## Fecha.de.inicio.IO.(metastáticos)                                3.28
## TIPO.IO.(metastáticos)                                           2.87
## Tipo.de.IO.Cat.(0=IO;.1=ChIO)                                    1.64
## Diana.IO                                                         1.64
## Nº.de.líneas.previas                                             1.64
## BS1+BS2                                                         65.16
## TERAPIA.BIOLÓGICA                                                0.00
## Tiempo.hasta.inicio.IO                                           4.10
```

```
## Mutacion.General                                                    0.00
## Tiempo.hasta.inicio.IO.metas                                        6.15
## IO.Tipo.General                                                     1.64
## Presencia.linea.Previa                                              1.64
## Paciente.code                                                       0.00
##
## Paciente
## Quimio.Radio.Adj                                                                  Quimiotera
## Fecha.de.inicio.IO
## Progresión...6
## PFS.(d)
## PFS.(m)
## Exitus
## Follow.Up.(d)
## Fecha.de.nacimiento
## Sexo.(0=Mujer;.1=Varón)
## Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)
## Tumor.previo
## Escala.ECOG
## Fecha.de.diagnóstico
## Edad.al.diagnóstico
## Estadio.al.diagnóstico
## Histología                                                Adenocarcinoma; Carcinoma escamoso; Car
## EGFR
## ALK
## ROS1
## RET
## BRAF.(V600)
## KRAS
## PD-L1
## Fecha.de.diagnóstico.de.enfermedad.metastática
## Quimioterapia.adyuvante
## Radioterapia.adyuvante
## RT.QT.Radical
## Fecha.de.inicio.IO.(metastáticos)
## TIPO.IO.(metastáticos)
## Tipo.de.IO.Cat.(0=IO;.1=ChIO)
## Diana.IO
## Nº.de.líneas.previas
## BS1+BS2
## TERAPIA.BIOLÓGICA
## Tiempo.hasta.inicio.IO
## Mutacion.General
## Tiempo.hasta.inicio.IO.metas
## IO.Tipo.General                                                                  Inmunoterap
## Presencia.linea.Previa
## Paciente.code
##
## Paciente
## Quimio.Radio.Adj
## Fecha.de.inicio.IO
## Progresión...6
## PFS.(d)
## PFS.(m)
```

```
## Exitus
## Follow.Up.(d)
## Fecha.de.nacimiento
## Sexo.(0=Mujer;.1=Varón)
## Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)
## Tumor.previo
## Escala.ECOG
## Fecha.de.diagnóstico
## Edad.al.diagnóstico
## Estadio.al.diagnóstico
## Histología                                                    Adenocarcinoma: 161; Carcinoma escamoso
## EGFR
## ALK
## ROS1
## RET
## BRAF.(V600)
## KRAS
## PD-L1
## Fecha.de.diagnóstico.de.enfermedad.metastática
## Quimioterapia.adyuvante
## Radioterapia.adyuvante
## RT.QT.Radical
## Fecha.de.inicio.IO.(metastáticos)
## TIPO.IO.(metastáticos)
## Tipo.de.IO.Cat.(0=IO;.1=ChIO)
## Diana.IO
## Nº.de.líneas.previas
## BS1+BS2
## TERAPIA.BIOLÓGICA
## Tiempo.hasta.inicio.IO
## Mutacion.General
## Tiempo.hasta.inicio.IO.metas
## IO.Tipo.General                                                                                 Inmunot
## Presencia.linea.Previa
## Paciente.code
##                                                             nearZeroVar
## Paciente                                                          FALSE
## Quimio.Radio.Adj                                                  FALSE
## Fecha.de.inicio.IO                                                FALSE
## Progresión...6                                                    FALSE
## PFS.(d)                                                           FALSE
## PFS.(m)                                                           FALSE
## Exitus                                                            FALSE
## Follow.Up.(d)                                                     FALSE
## Fecha.de.nacimiento                                               FALSE
## Sexo.(0=Mujer;.1=Varón)                                           FALSE
## Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)      FALSE
## Tumor.previo                                                      FALSE
## Escala.ECOG                                                       FALSE
## Fecha.de.diagnóstico                                              FALSE
## Edad.al.diagnóstico                                               FALSE
## Estadio.al.diagnóstico                                            FALSE
## Histología                                                        FALSE
## EGFR                                                               TRUE
```

```
## ALK                                                                    TRUE
## ROS1                                                                   TRUE
## RET                                                                   FALSE
## BRAF.(V600)                                                            TRUE
## KRAS                                                                  FALSE
## PD-L1                                                                 FALSE
## Fecha.de.diagnóstico.de.enfermedad.metastática                        FALSE
## Quimioterapia.adyuvante                                               FALSE
## Radioterapia.adyuvante                                                FALSE
## RT.QT.Radical                                                         FALSE
## Fecha.de.inicio.IO.(metastáticos)                                     FALSE
## TIPO.IO.(metastáticos)                                                FALSE
## Tipo.de.IO.Cat.(0=IO;.1=ChIO)                                         FALSE
## Diana.IO                                                              FALSE
## Nº.de.líneas.previas                                                  FALSE
## BS1+BS2                                                                TRUE
## TERAPIA.BIOLÓGICA                                                     FALSE
## Tiempo.hasta.inicio.IO                                                FALSE
## Mutacion.General                                                      FALSE
## Tiempo.hasta.inicio.IO.metas                                          FALSE
## IO.Tipo.General                                                       FALSE
## Presencia.linea.Previa                                                FALSE
## Paciente.code                                                         FALSE
##
## Paciente                                                                                                  P
## Quimio.Radio.Adj                                                 Quimioterapia adyuvante; Desconocido/No
## Fecha.de.inicio.IO                                                                                     2018-
## Progresión...6
## PFS.(d)
## PFS.(m)
## Exitus
## Follow.Up.(d)
## Fecha.de.nacimiento                                                                                    1960-
## Sexo.(0=Mujer;.1=Varón)
## Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)
## Tumor.previo
## Escala.ECOG
## Fecha.de.diagnóstico                                                                                   2016-
## Edad.al.diagnóstico
## Estadio.al.diagnóstico
## Histología                                                                                    NA; Adeno
## EGFR
## ALK
## ROS1
## RET
## BRAF.(V600)
## KRAS
## PD-L1
## Fecha.de.diagnóstico.de.enfermedad.metastática                                                         2018-
## Quimioterapia.adyuvante
## Radioterapia.adyuvante
## RT.QT.Radical
## Fecha.de.inicio.IO.(metastáticos)                                                                      2018-
## TIPO.IO.(metastáticos)                                                                    Inmunoterapia
```

```
## Tipo.de.IO.Cat.(0=IO;.1=ChIO)
## Diana.IO
## Nº.de.líneas.previas
## BS1+BS2
## TERAPIA.BIOLÓGICA
## Tiempo.hasta.inicio.IO
## Mutacion.General                                                                                                    Desconoc
## Tiempo.hasta.inicio.IO.metas
## IO.Tipo.General                                                                                                  Inmunoterapia
## Presencia.linea.Previa
## Paciente.code
##                                                                  Min Median
## Paciente                                                          NA     NA
## Quimio.Radio.Adj                                                  NA     NA
## Fecha.de.inicio.IO                                                NA     NA
## Progresión...6                                                    NA     NA
## PFS.(d)                                                         1.00 193.00
## PFS.(m)                                                         0.03   6.43
## Exitus                                                            NA     NA
## Follow.Up.(d)                                                   2.00 357.00
## Fecha.de.nacimiento                                               NA     NA
## Sexo.(0=Mujer;.1=Varón)                                           NA     NA
## Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)      NA     NA
## Tumor.previo                                                      NA     NA
## Escala.ECOG                                                       NA     NA
## Fecha.de.diagnóstico                                              NA     NA
## Edad.al.diagnóstico                                             29.00  66.00
## Estadio.al.diagnóstico                                            NA     NA
## Histología                                                        NA     NA
## EGFR                                                              NA     NA
## ALK                                                               NA     NA
## ROS1                                                              NA     NA
## RET                                                               NA     NA
## BRAF.(V600)                                                       NA     NA
## KRAS                                                              NA     NA
## PD-L1                                                           0.00  10.00
## Fecha.de.diagnóstico.de.enfermedad.metastática                    NA     NA
## Quimioterapia.adyuvante                                           NA     NA
## Radioterapia.adyuvante                                            NA     NA
## RT.QT.Radical                                                     NA     NA
## Fecha.de.inicio.IO.(metastáticos)                                 NA     NA
## TIPO.IO.(metastáticos)                                            NA     NA
## Tipo.de.IO.Cat.(0=IO;.1=ChIO)                                     NA     NA
## Diana.IO                                                          NA     NA
## Nº.de.líneas.previas                                              NA     NA
## BS1+BS2                                                           NA     NA
## TERAPIA.BIOLÓGICA                                                 NA     NA
## Tiempo.hasta.inicio.IO                                          2.00  78.50
## Mutacion.General                                                  NA     NA
## Tiempo.hasta.inicio.IO.metas                                    0.00  43.00
## IO.Tipo.General                                                   NA     NA
## Presencia.linea.Previa                                            NA     NA
## Paciente.code                                                     NA     NA
##                                                                  Max
```

```
## Paciente                                                          NA
## Quimio.Radio.Adj                                                  NA
## Fecha.de.inicio.IO                                                NA
## Progresión...6                                                    NA
## PFS.(d)                                                       2058.0
## PFS.(m)                                                         68.6
## Exitus                                                            NA
## Follow.Up.(d)                                                 2114.0
## Fecha.de.nacimiento                                               NA
## Sexo.(0=Mujer;.1=Varón)                                           NA
## Tabaquismo.(0=Nunca.fumador;.1=Exfumador;.2=.Fumador.Activo)      NA
## Tumor.previo                                                      NA
## Escala.ECOG                                                       NA
## Fecha.de.diagnóstico                                              NA
## Edad.al.diagnóstico                                             87.0
## Estadio.al.diagnóstico                                            NA
## Histología                                                        NA
## EGFR                                                              NA
## ALK                                                               NA
## ROS1                                                              NA
## RET                                                               NA
## BRAF.(V600)                                                       NA
## KRAS                                                              NA
## PD-L1                                                          100.0
## Fecha.de.diagnóstico.de.enfermedad.metastática                    NA
## Quimioterapia.adyuvante                                           NA
## Radioterapia.adyuvante                                            NA
## RT.QT.Radical                                                     NA
## Fecha.de.inicio.IO.(metastáticos)                                 NA
## TIPO.IO.(metastáticos)                                            NA
## Tipo.de.IO.Cat.(0=IO;.1=ChIO)                                     NA
## Diana.IO                                                          NA
## Nº.de.líneas.previas                                              NA
## BS1+BS2                                                           NA
## TERAPIA.BIOLÓGICA                                                 NA
## Tiempo.hasta.inicio.IO                                        4488.0
## Mutacion.General                                                  NA
## Tiempo.hasta.inicio.IO.metas                                  3116.0
## IO.Tipo.General                                                   NA
## Presencia.linea.Previa                                            NA
## Paciente.code                                                     NA
```

Calculo de indices ecolologicos

Procesamos datos de repertorios de receptores TCR utilizando la librería immunarch, organizando las muestras en tres grupos (BS1, BS2 y BS3) según un identificador extraído del nombre de cada archivo. Primero se cargan `todos` los ficheros `.results.tsv` del directorio de trabajo y se filtran en tres listas de acuerdo con su etiqueta BS (1, 2 o 3). A continuación, cada conjunto de archivos se importa con repLoad(), y los nombres de muestra se acortan hasta el tercer bloque (por ejemplo, “01-002-1”) para mantener consistencia en los metadatos.

Una vez preparados los datos, el script calcula varios índices ecológicos para cada grupo: riqueza (Chao1), diversidad de Shannon, índice de Simpson (Gini–Simpson), convergencia de Gini, así como los percentiles D50 y D10. La equidad o “evenness” de Pielou se obtiene dividiendo el valor de Shannon entre el logaritmo

de la riqueza estimada. Todos estos valores se ensamblan en un único data.frame por grupo, se limpia la columna de muestras para eliminar sufijos redundantes.

Se recomienda correr el código en el server ya que es bastante pesado y en el equipo local puede dar problemas de memoria.

```
##############################################################################
###################### INDICES ECOLOGICOS ####################################
##############################################################################

#### ENTORNO R ####
# Solo necesitamos immunarch para este análisis
library(immunarch)
library(dplyr)


# Establecer directorio de trabajo
# Establecer ruta en local o server en función de si queremos solo testear el codigo con unos archivos
target_dir <- "/home/atrys/Documentos/TCR/pruebas"
setwd(target_dir)

## CARGA DE LOS ARCHIVOS ###
reads <- list.files(pattern = "*.results.tsv")

# Función auxiliar para extraer el BS a partir del segundo guión
get_bs <- function(filename) {
  parts <- strsplit(filename, "-")[[1]]
  parts[3]
}

# Filtrar muestras BS1, BS2 y BS3
bs1_reads <- reads[sapply(reads, get_bs) == "1"]
bs2_reads <- reads[sapply(reads, get_bs) == "2"]
bs3_reads <- reads[sapply(reads, get_bs) == "3"]

# Cargar los datos con immunarch
bs1_data <- repLoad(bs1_reads)
bs2_data <- repLoad(bs2_reads)
bs3_data <- repLoad(bs3_reads)

# Función para reducir nombres de muestra (hasta el tercer bloque, ej. 01-002-1)
reduce_names <- function(obj) {
  orig <- names(obj$data)
  reduced <- sub("^(([^-]+-){2}[^-]+).*", "\\1", orig)
  names(obj$data) <- reduced
  obj$meta$Sample <- reduced
  obj
}

# Aplicar reducción a cada conjunto de datos
bs1_data <- reduce_names(bs1_data)
bs2_data <- reduce_names(bs2_data)
bs3_data <- reduce_names(bs3_data)
```

```
### RIQUEZA (Chao1) ####

chao1_bs1 <- repDiversity(bs1_data$data, .method = "chao1")
chao1_bs2 <- repDiversity(bs2_data$data, .method = "chao1")
chao1_bs3 <- repDiversity(bs3_data$data, .method = "chao1")


#### Shanon diversity ###
shannon_bs1 <- repDiversity(bs1_data$data, .method="div")
shannon_bs2 <- repDiversity(bs2_data$data, .method="div")
shannon_bs3 <- repDiversity(bs1_data$data, .method="div")


### Diversidad de Simpsom o Gini Simpsom ###
simp_bs1 <- repDiversity(bs1_data$data, .method = "gini.simp")
simp_bs2 <- repDiversity(bs2_data$data, .method = "gini.simp")
simp_bs3 <- repDiversity(bs3_data$data, .method = "gini.simp")



### Indice de equidad o (eveness) ###

# Como immnuarch no tiene funcion para calcular la equidad o eveness utilizaremos la formula
# de Pielou

bs1_even <- shannon_bs1[,"Value"] / log(chao1_bs1[, "Estimator"])
bs2_even <- shannon_bs2[,"Value"] / log(chao1_bs2[, "Estimator"])
bs3_even <- shannon_bs3[,"Value"] / log(chao1_bs3[, "Estimator"])


### Índice de Convergencia (Gini) ###
convergencia_bs1 <- repDiversity(bs1_data$data, .method = "gini")
convergencia_bs2 <- repDiversity(bs2_data$data, .method = "gini")
convergencia_bs3 <- repDiversity(bs3_data$data, .method = "gini")


### Indice d50 ###
d50_bs1 <- repDiversity(bs1_data$data, .method = "d50")
d50_bs2 <- repDiversity(bs2_data$data, .method = "d50")
d50_bs3 <- repDiversity(bs3_data$data, .method = "d50")


### Indice d10 ###
d10_bs1 <- repDiversity(bs1_data$data, .method = "dxx", .perc = 9)
d10_bs2 <- repDiversity(bs2_data$data, .method = "dxx", .perc = 9)
d10_bs3 <- repDiversity(bs3_data$data, .method = "dxx", .perc = 9)



# ----------------------------
# BS1 eco index
# ----------------------------
```

```r
chao1_vals_bs1    <- chao1_bs1[, 1]
shannon_vals_bs1  <- shannon_bs1[, 2]
simp_vals_bs1     <- simp_bs1[, 2]
even_vals_bs1     <- bs1_even           # sigue siendo un vector
conv_vals_bs1     <- convergencia_bs1[, 1]
d50_vals_bs1      <- d50_bs1[, 1]
d10_vals_bs1      <- d10_bs1[, 1]

# Crear data.frame unificado
bs1_eco_index <- data.frame(
  Sample           = rownames(chao1_bs1),
  Chao1_bs1            = chao1_vals_bs1,
  Shannon_bs1          = shannon_vals_bs1,
  Simpson_GiniSimp_bs1 = simp_vals_bs1,
  Pielou_Evenness_bs1  = even_vals_bs1,
  Gini_Convergence_bs1 = conv_vals_bs1,
  D50_bs1              = d50_vals_bs1,
  D10_bs1              = d10_vals_bs1,
  stringsAsFactors = FALSE
)

# Quitar el sufijo tras el segundo guion en Sample
bs1_eco_index$Sample <- sub("-[^-]+$", "", bs1_eco_index$Sample)
# Quitar las filas de nombres duplicados
rownames(bs1_eco_index) <- NULL

write.csv(bs1_eco_index, file = "bs1_eco_index.csv", row.names = FALSE)


# ----------------------------
# BS2 eco index
# ----------------------------

# Extraer vectores de la primera/segunda columna según corresponda
chao1_vals_bs2    <- chao1_bs2[, 1]
shannon_vals_bs2  <- shannon_bs2[, 2]
simp_vals_bs2     <- simp_bs2[, 2]
even_vals_bs2     <- bs2_even           # vector de Pielou para BS2
conv_vals_bs2     <- convergencia_bs2[, 1]
d50_vals_bs2      <- d50_bs2[, 1]
d10_vals_bs2      <- d10_bs2[, 1]

# Crear data.frame unificado
bs2_eco_index <- data.frame(
  Sample           = rownames(chao1_bs2),
  Chao1_bs2            = chao1_vals_bs2,
  Shannon_bs2          = shannon_vals_bs2,
  Simpson_GiniSimp_bs2 = simp_vals_bs2,
  Pielou_Evenness_bs2  = even_vals_bs2,
  Gini_Convergence_bs2 = conv_vals_bs2,
  D50_bs2              = d50_vals_bs2,
  D10_bs2              = d10_vals_bs2,
  stringsAsFactors = FALSE
```

```
)

# Limpiar la columna Sample (quitar lo posterior al segundo guion)
bs2_eco_index$Sample <- sub("-[^-]+$", "", bs2_eco_index$Sample)

# Eliminar los rownames para que no se muestren duplicados
rownames(bs2_eco_index) <- NULL

write.csv(bs2_eco_index, file = "bs2_eco_index.csv", row.names = FALSE)



# ----------------------------
# BS3 eco index
# ----------------------------

# Extraer vectores de la primera/segunda columna según corresponda
chao1_vals_bs3    <- chao1_bs3[, 1]
shannon_vals_bs3  <- shannon_bs3[, 2]
simp_vals_bs3     <- simp_bs3[, 2]
even_vals_bs3     <- bs3_even           # vector de Pielou para BS3
conv_vals_bs3     <- convergencia_bs3[, 1]
d50_vals_bs3      <- d50_bs3[, 1]
d10_vals_bs3      <- d10_bs3[, 1]

# Crear data.frame unificado
bs3_eco_index <- data.frame(
  Sample           = rownames(chao1_bs3),
  chao1_bs3            = chao1_vals_bs3,
  Shannon_bs3          = shannon_vals_bs3,
  Simpson_GiniSimp_bs3 = simp_vals_bs3,
  Pielou_Evenness_bs3  = even_vals_bs3,
  Gini_Convergence_bs3 = conv_vals_bs3,
  D50_bs3              = d50_vals_bs3,
  D10_bs3              = d10_vals_bs3,
  stringsAsFactors = FALSE
)

# Limpiar la columna Sample (quitar lo posterior al segundo guion)
bs3_eco_index$Sample <- sub("-[^-]+$", "", bs3_eco_index$Sample)

# Eliminar los rownames para que no se muestren duplicados
rownames(bs3_eco_index) <- NULL

# Exportar a CSV (opcional)
write.csv(bs3_eco_index, file = "bs3_eco_index.csv", row.names = FALSE)
```

Datos de TCR

Este script en R está diseñado para generar una tabla única con los usos relativos de los genes TCR V y J en cada muestra, agrupadas en tres conjuntos (BS1, BS2 y BS3). Al igual que en el anterior, primero se cargan todos los ficheros `.results.tsv` del directorio de trabajo y se filtran según su etiqueta BS mediante la función get_bs(). Cada grupo de archivos se importa con repLoad() y sus nombres de muestra se reducen

al tercer bloque (XX-YYY-Z) usando reduce_names() para mantener consistencia.

A continuación, para cada grupo BS se calcula el uso relativo normalizado de los genes hs.trbv (V) y hs.trbj (J) con geneUsage(..., .norm = TRUE), rellenando valores NA con ceros. Cada resultado se transpone y se renombran las columnas para incluir el sufijo del grupo (-bs1, -bs2, -bs3) mediante transpose_and_suffix(). Por último, se fusionan todos los data.frames transpuestos en uno solo con Reduce(full_join), y se añade una columna Patient que extrae el identificador común a todas las réplicas de una misma muestra (eliminando el sufijo -1, -2 o -3). El resultado es una tabla definitiva donde cada fila corresponde a una réplica de un paciente y contiene, de forma organizada, los porcentajes de uso de cada gen V y J.

```
################################################################################
######################## TABLA DE TCRs #########################################
################################################################################

#### ENTORNO R ####
# Solo necesitamos immunarch para este análisis
library(immunarch)
library(dplyr)


# Establecer directorio de trabajo
# Establecer ruta en local o server en función de si queremos solo testear el codigo con unos archivos
target_dir <- "/home/atrys/Documentos/TCR/pruebas"
setwd(target_dir)

## CARGA DE LOS ARCHIVOS ###
reads <- list.files(pattern = "*.results.tsv")

# Función auxiliar para extraer el BS a partir del segundo guión
get_bs <- function(filename) {
  parts <- strsplit(filename, "-")[[1]]
  parts[3]
}

# Filtrar muestras BS1, BS2 y BS3
bs1_reads <- reads[sapply(reads, get_bs) == "1"]
bs2_reads <- reads[sapply(reads, get_bs) == "2"]
bs3_reads <- reads[sapply(reads, get_bs) == "3"]

# Cargar los datos con immunarch
bs1_data <- repLoad(bs1_reads)
bs2_data <- repLoad(bs2_reads)
bs3_data <- repLoad(bs3_reads)

# Función para reducir nombres de muestra (hasta el tercer bloque, ej. 01-002-1)
reduce_names <- function(obj) {
  orig <- names(obj$data)
  reduced <- sub("^(([^-]+-){2}[^-]+).*", "\\1", orig)
  names(obj$data) <- reduced
  obj$meta$Sample <- reduced
  obj
}

# Aplicar reducción a cada conjunto de datos
bs1_data <- reduce_names(bs1_data)
```

```r
bs2_data <- reduce_names(bs2_data)
bs3_data <- reduce_names(bs3_data)

### USO RELATIVO DE TCRs ###
get_rel <- function(data_obj, gene) {
  df <- geneUsage(data_obj$data, .gene = gene, .norm = TRUE)
  df[is.na(df)] <- 0
  df
}
bs1_TCR_v_rel <- get_rel(bs1_data, "hs.trbv")
bs1_TCR_j_rel <- get_rel(bs1_data, "hs.trbj")
bs2_TCR_v_rel <- get_rel(bs2_data, "hs.trbv")
bs2_TCR_j_rel <- get_rel(bs2_data, "hs.trbj")
bs3_TCR_v_rel <- get_rel(bs3_data, "hs.trbv")
bs3_TCR_j_rel <- get_rel(bs3_data, "hs.trbj")

# Función para transponer y añadir sufijo de BS a cada gen, manejando tibbles
transpose_and_suffix <- function(df, bs_label) {
  genes <- df[[1]]
  mat <- as.matrix(df[,-1])
  rownames(mat) <- genes
  mat_t <- t(mat)
  colnames(mat_t) <- paste0(colnames(mat_t), "-", bs_label)
  out <- as.data.frame(mat_t, stringsAsFactors = FALSE)
  out$Sample <- rownames(mat_t)
  rownames(out) <- NULL
  out[, c("Sample", setdiff(names(out), "Sample"))]
}

# Aplicar función a cada data frame de uso relativo con nombre correcto
bs1_TCR_v_rel_t <- transpose_and_suffix(bs1_TCR_v_rel, "bs1")
bs1_TCR_j_rel_t <- transpose_and_suffix(bs1_TCR_j_rel, "bs1")
bs2_TCR_v_rel_t <- transpose_and_suffix(bs2_TCR_v_rel, "bs2")
bs2_TCR_j_rel_t <- transpose_and_suffix(bs2_TCR_j_rel, "bs2")
bs3_TCR_v_rel_t <- transpose_and_suffix(bs3_TCR_v_rel, "bs3")
bs3_TCR_j_rel_t <- transpose_and_suffix(bs3_TCR_j_rel, "bs3")

### CREAR DATOS DEFINITIVOS POR PACIENTE ###
# Unir todos los data frames con Reduce (base R)
all_tcr <- Reduce(function(x, y) full_join(x, y, by = "Sample"),
                  list(
                    bs1_TCR_v_rel_t, bs1_TCR_j_rel_t,
                    bs2_TCR_v_rel_t, bs2_TCR_j_rel_t,
                    bs3_TCR_v_rel_t, bs3_TCR_j_rel_t
                  ))

# Extraer identificador de paciente (quita el sufijo -1/-2/-3)
all_tcr <- all_tcr %>%
  mutate(Patient = sub("-[123]$", "", Sample)) %>%
  select(Patient, Sample, everything())
```

Data merge

```
################################################################################
################################## DATA MERGE ##################################
################################################################################

# 0) Cargamos los archivos CSV
clinic_full_data <- read.csv(
  "/home/agombau/modelo_pipeline/procesed_data/clinic_full_data.csv",
  stringsAsFactors = FALSE
)

# 1) TCR relativos y absolutos (quitamos la primera columna de índices)
all_tcr_rel <- read.csv(
  "/home/agombau/modelo_pipeline/procesed_data/all_tcr_rel.csv",
  stringsAsFactors = FALSE
)[, -1]
all_tcr_abs <- read.csv(
  "/home/agombau/modelo_pipeline/procesed_data/all_tcr_abs.csv",
  stringsAsFactors = FALSE
)[, -1]

# 2) Estandarizamos la columna de muestra como "Sample"
names(all_tcr_abs)[1] <- "Sample"

# 3) Cargamos los índices ecológicos de bs1, bs2 y bs3
bs1_eco_index <- read.csv(
  "/home/agombau/modelo_pipeline/procesed_data/bs1_eco_index.csv",
  stringsAsFactors = FALSE
)
bs2_eco_index <- read.csv(
  "/home/agombau/modelo_pipeline/procesed_data/bs2_eco_index.csv",
  stringsAsFactors = FALSE
)
bs3_eco_index <- read.csv(
  "/home/agombau/modelo_pipeline/procesed_data/bs3_eco_index.csv",
  stringsAsFactors = FALSE
)

# 4) Convertimos a character los campos que usaremos para merge
clinic_full_data$Paciente.code <- as.character(clinic_full_data$Paciente.code)
all_tcr_rel$Sample             <- as.character(all_tcr_rel$Sample)
all_tcr_abs$Sample             <- as.character(all_tcr_abs$Sample)
bs1_eco_index$Sample           <- as.character(bs1_eco_index$Sample)
bs2_eco_index$Sample           <- as.character(bs2_eco_index$Sample)
bs3_eco_index$Sample           <- as.character(bs3_eco_index$Sample)

# 5) Función para extraer y renombrar TCR por batch (bs1, bs2, bs3) y tipo (rel o abs)
extract_tcr <- function(df, type, bs) {
  pat      <- paste0("-", bs, "$")
  sel_rows <- grepl(pat, df$Sample)
  cols     <- grep(paste0("\\.bs", bs, "$"), names(df), value = TRUE)
  sub_df   <- df[sel_rows, c("Sample", cols), drop = FALSE]
  sub_df$Paciente.code <- sub("^(.+)-\\d$", "\\1", sub_df$Sample)
```

```
  new_names <- gsub(
    paste0("\\.bs", bs, "$"),
    paste0("_bs", bs, "_", type),
    cols
  )
  names(sub_df)[match(cols, names(sub_df))] <- new_names
  sub_df$Sample <- NULL
  sub_df <- sub_df[, c("Paciente.code", new_names), drop = FALSE]
  return(sub_df)
}

# 6) Extraemos todos los subconjuntos de TCR (bs1, bs2, bs3; rel y abs)
tcr_list <- list()
for (bs in 1:3) {
  tcr_list[[paste0("bs", bs, "_rel")]] <- extract_tcr(all_tcr_rel, "rel", bs)
  tcr_list[[paste0("bs", bs, "_abs")]] <- extract_tcr(all_tcr_abs, "abs", bs)
}

# 7) Merge secuencial con datos clínicos usando LEFT JOIN
merged_full <- clinic_full_data
for (df in tcr_list) {
  merged_full <- merge(
    x     = merged_full,
    y     = df,
    by    = "Paciente.code",
    all.x = TRUE
  )
}

# 8) Función para preparar índices ecológicos (añadir sufijo _bs{n}_eco)
prep_eco <- function(eco_df, bs) {
  names(eco_df)[names(eco_df) == "Sample"] <- "Paciente.code"
  cols_to_rename <- setdiff(names(eco_df), "Paciente.code")
  names(eco_df)[names(eco_df) %in% cols_to_rename] <-
    paste0(cols_to_rename, "_bs", bs, "_eco")
  return(eco_df)
}

# 9) Preparamos y añadimos índices ecológicos con LEFT JOIN
bs1_eco <- prep_eco(bs1_eco_index, 1)
bs2_eco <- prep_eco(bs2_eco_index, 2)
bs3_eco <- prep_eco(bs3_eco_index, 3)

merged_full <- merge(merged_full, bs1_eco, by = "Paciente.code", all.x = TRUE)
merged_full <- merge(merged_full, bs2_eco, by = "Paciente.code", all.x = TRUE)
merged_full <- merge(merged_full, bs3_eco, by = "Paciente.code", all.x = TRUE)


# 10b) Filtrar pacientes sin ningún dato de bs1
library(dplyr)
merged_full <- merged_full %>%
  filter(
    if_any(
```

```
      matches("_bs1_(rel|abs)$"),
      ~ !is.na(.)
    )
  )

# 12) Guardamos el resultado
write.csv(
  merged_full,
  file = "/home/agombau/modelo_pipeline/procesed_data/merged_full.csv",
  row.names = FALSE
)
```

Imputación de valores faltantes

```
# Importamos los datos teniendo en cuenta las clases (factores, fechas, y numero)
# Leer todo como character
merged_full <- read_csv(
  "/home/agombau/modelo_pipeline/procesed_data/merged_full.csv",
  col_types = cols(.default = "c")
)
```

```
## New names:
## * `Progresión...6` -> `Progresión`
```

```
# Auto-detectar numéricas y fechas
merged_full <- type_convert(merged_full)
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double(),
##   Paciente.code = col_character(),
##   Paciente = col_character(),
##   Quimio.Radio.Adj = col_character(),
##   Fecha.de.inicio.IO = col_date(format = ""),
##   Progresión = col_character(),
##   Exitus = col_character(),
##   Fecha.de.nacimiento = col_date(format = ""),
##   Sexo..0.Mujer..1.Varón. = col_character(),
##   Tabaquismo..0.Nunca.fumador..1.Exfumador..2..Fumador.Activo. = col_character(),
##   Tumor.previo = col_character(),
##   Fecha.de.diagnóstico = col_date(format = ""),
##   Estadio.al.diagnóstico = col_character(),
##   Histología = col_character(),
##   EGFR = col_character(),
##   ALK = col_character(),
##   ROS1 = col_character(),
##   RET = col_character(),
##   BRAF..V600. = col_character(),
##   KRAS = col_character(),
##   Fecha.de.diagnóstico.de.enfermedad.metastática = col_date(format = "")
##   # ... with 12 more columns
## )
```

```
## i Use `spec()` for the full column specifications.
```

```
# Convertir todas las columnas character en factors
merged_full <- merged_full %>%
  mutate(across(where(is.character), as.factor))
```

Resumen de missingness y clases de variable

Creamos un dataframe con el nonbre de variable y su clase

```
var_info <- data.frame(
  variable = names(merged_full),
  clase    = sapply(merged_full, class),
  stringsAsFactors = FALSE
)
```

Calculamos el numero de missing y el % de missing por variable

```
missing_info <- merged_full %>%
  summarise(across(everything(), ~ sum(is.na(.)))) %>%
  pivot_longer(
    cols        = everything(),
    names_to    = "variable",
    values_to   = "n_missing"
  ) %>%
  mutate(pct_missing = n_missing / nrow(merged_full) * 100)
```

Unimos la información de clase y el missing y ordenamos de mayor a menor % de missing

```
tabla_exploratoria <- var_info %>%
  left_join(missing_info, by = "variable") %>%
  arrange(desc(pct_missing))
```

Resultado:

```
tabla_exploratoria
```

```
##                                                 variable   clase
## 1                                      TRBV10.1_bs3_rel numeric
## 2                                      TRBV10.2_bs3_rel numeric
## 3                                      TRBV10.3_bs3_rel numeric
## 4                                      TRBV11.1_bs3_rel numeric
## 5                                      TRBV11.2_bs3_rel numeric
## 6                                      TRBV11.3_bs3_rel numeric
## 7                                      TRBV12.3_bs3_rel numeric
## 8                                      TRBV12.4_bs3_rel numeric
## 9                                      TRBV12.5_bs3_rel numeric
## 10                                       TRBV13_bs3_rel numeric
## 11                                       TRBV14_bs3_rel numeric
## 12                                       TRBV15_bs3_rel numeric
## 13                                       TRBV16_bs3_rel numeric
## 14                                       TRBV18_bs3_rel numeric
## 15                                       TRBV19_bs3_rel numeric
## 16                                        TRBV2_bs3_rel numeric
## 17                                     TRBV20.1_bs3_rel numeric
## 18                                     TRBV24.1_bs3_rel numeric
## 19                                     TRBV25.1_bs3_rel numeric
## 20                                       TRBV27_bs3_rel numeric
```

```
## 21                                  TRBV28_bs3_rel numeric
## 22                                TRBV29.1_bs3_rel numeric
## 23                                 TRBV3.1_bs3_rel numeric
## 24                                  TRBV30_bs3_rel numeric
## 25                                 TRBV4.1_bs3_rel numeric
## 26                                 TRBV4.2_bs3_rel numeric
## 27                                 TRBV4.3_bs3_rel numeric
## 28                                 TRBV5.1_bs3_rel numeric
## 29                                 TRBV5.4_bs3_rel numeric
## 30                                 TRBV5.5_bs3_rel numeric
## 31                                 TRBV5.6_bs3_rel numeric
## 32                                 TRBV5.8_bs3_rel numeric
## 33                                 TRBV6.1_bs3_rel numeric
## 34                                 TRBV6.2_bs3_rel numeric
## 35                                 TRBV6.3_bs3_rel numeric
## 36                                 TRBV6.4_bs3_rel numeric
## 37                                 TRBV6.5_bs3_rel numeric
## 38                                 TRBV6.6_bs3_rel numeric
## 39                                 TRBV6.8_bs3_rel numeric
## 40                                 TRBV6.9_bs3_rel numeric
## 41                                 TRBV7.2_bs3_rel numeric
## 42                                 TRBV7.3_bs3_rel numeric
## 43                                 TRBV7.4_bs3_rel numeric
## 44                                 TRBV7.6_bs3_rel numeric
## 45                                 TRBV7.7_bs3_rel numeric
## 46                                 TRBV7.8_bs3_rel numeric
## 47                                 TRBV7.9_bs3_rel numeric
## 48                                   TRBV9_bs3_rel numeric
## 49                                 TRBJ1.1_bs3_rel numeric
## 50                                 TRBJ1.2_bs3_rel numeric
## 51                                 TRBJ1.3_bs3_rel numeric
## 52                                 TRBJ1.4_bs3_rel numeric
## 53                                 TRBJ1.5_bs3_rel numeric
## 54                                 TRBJ1.6_bs3_rel numeric
## 55                                 TRBJ2.1_bs3_rel numeric
## 56                                 TRBJ2.2_bs3_rel numeric
## 57                                 TRBJ2.3_bs3_rel numeric
## 58                                 TRBJ2.4_bs3_rel numeric
## 59                                 TRBJ2.5_bs3_rel numeric
## 60                                 TRBJ2.6_bs3_rel numeric
## 61                                 TRBJ2.7_bs3_rel numeric
## 62                               TRBV10.1_bs3_abs numeric
## 63                               TRBV10.2_bs3_abs numeric
## 64                               TRBV10.3_bs3_abs numeric
## 65                               TRBV11.1_bs3_abs numeric
## 66                               TRBV11.2_bs3_abs numeric
## 67                               TRBV11.3_bs3_abs numeric
## 68                               TRBV12.3_bs3_abs numeric
## 69                               TRBV12.4_bs3_abs numeric
## 70                               TRBV12.5_bs3_abs numeric
## 71                                 TRBV13_bs3_abs numeric
## 72                                 TRBV14_bs3_abs numeric
## 73                                 TRBV15_bs3_abs numeric
## 74                                 TRBV16_bs3_abs numeric
```

```
## 75                                      TRBV18_bs3_abs numeric
## 76                                      TRBV19_bs3_abs numeric
## 77                                       TRBV2_bs3_abs numeric
## 78                                    TRBV20.1_bs3_abs numeric
## 79                                    TRBV24.1_bs3_abs numeric
## 80                                    TRBV25.1_bs3_abs numeric
## 81                                      TRBV27_bs3_abs numeric
## 82                                      TRBV28_bs3_abs numeric
## 83                                    TRBV29.1_bs3_abs numeric
## 84                                     TRBV3.1_bs3_abs numeric
## 85                                      TRBV30_bs3_abs numeric
## 86                                     TRBV4.1_bs3_abs numeric
## 87                                     TRBV4.2_bs3_abs numeric
## 88                                     TRBV4.3_bs3_abs numeric
## 89                                     TRBV5.1_bs3_abs numeric
## 90                                     TRBV5.4_bs3_abs numeric
## 91                                     TRBV5.5_bs3_abs numeric
## 92                                     TRBV5.6_bs3_abs numeric
## 93                                     TRBV5.8_bs3_abs numeric
## 94                                     TRBV6.1_bs3_abs numeric
## 95                                     TRBV6.2_bs3_abs numeric
## 96                                     TRBV6.3_bs3_abs numeric
## 97                                     TRBV6.4_bs3_abs numeric
## 98                                     TRBV6.5_bs3_abs numeric
## 99                                     TRBV6.6_bs3_abs numeric
## 100                                    TRBV6.8_bs3_abs numeric
## 101                                    TRBV6.9_bs3_abs numeric
## 102                                    TRBV7.2_bs3_abs numeric
## 103                                    TRBV7.3_bs3_abs numeric
## 104                                    TRBV7.4_bs3_abs numeric
## 105                                    TRBV7.6_bs3_abs numeric
## 106                                    TRBV7.7_bs3_abs numeric
## 107                                    TRBV7.8_bs3_abs numeric
## 108                                    TRBV7.9_bs3_abs numeric
## 109                                      TRBV9_bs3_abs numeric
## 110                                    TRBJ1.1_bs3_abs numeric
## 111                                    TRBJ1.2_bs3_abs numeric
## 112                                    TRBJ1.3_bs3_abs numeric
## 113                                    TRBJ1.4_bs3_abs numeric
## 114                                    TRBJ1.5_bs3_abs numeric
## 115                                    TRBJ1.6_bs3_abs numeric
## 116                                    TRBJ2.1_bs3_abs numeric
## 117                                    TRBJ2.2_bs3_abs numeric
## 118                                    TRBJ2.3_bs3_abs numeric
## 119                                    TRBJ2.4_bs3_abs numeric
## 120                                    TRBJ2.5_bs3_abs numeric
## 121                                    TRBJ2.6_bs3_abs numeric
## 122                                    TRBJ2.7_bs3_abs numeric
## 123                                 Chao1_bs3_bs3_eco numeric
## 124                               Shannon_bs3_bs3_eco numeric
## 125                     Simpson_GiniSimp_bs3_bs3_eco numeric
## 126                      Pielou_Evenness_bs3_bs3_eco numeric
## 127                    Gini_Convergence_bs3_bs3_eco numeric
## 128                                   D50_bs3_bs3_eco numeric
```

```
## 129                                  D10_bs3_bs3_eco numeric
## 130                                              RET  factor
## 131                                             KRAS  factor
## 132                                         BS1.BS2  factor
## 133                                TRBV10.1_bs2_rel numeric
## 134                                TRBV10.2_bs2_rel numeric
## 135                                TRBV10.3_bs2_rel numeric
## 136                                TRBV11.1_bs2_rel numeric
## 137                                TRBV11.2_bs2_rel numeric
## 138                                TRBV11.3_bs2_rel numeric
## 139                                TRBV12.3_bs2_rel numeric
## 140                                TRBV12.4_bs2_rel numeric
## 141                                TRBV12.5_bs2_rel numeric
## 142                                  TRBV13_bs2_rel numeric
## 143                                  TRBV14_bs2_rel numeric
## 144                                  TRBV15_bs2_rel numeric
## 145                                  TRBV16_bs2_rel numeric
## 146                                  TRBV18_bs2_rel numeric
## 147                                  TRBV19_bs2_rel numeric
## 148                                   TRBV2_bs2_rel numeric
## 149                                TRBV20.1_bs2_rel numeric
## 150                                TRBV24.1_bs2_rel numeric
## 151                                TRBV25.1_bs2_rel numeric
## 152                                  TRBV27_bs2_rel numeric
## 153                                  TRBV28_bs2_rel numeric
## 154                                TRBV29.1_bs2_rel numeric
## 155                                 TRBV3.1_bs2_rel numeric
## 156                                  TRBV30_bs2_rel numeric
## 157                                 TRBV4.1_bs2_rel numeric
## 158                                 TRBV4.2_bs2_rel numeric
## 159                                 TRBV4.3_bs2_rel numeric
## 160                                 TRBV5.1_bs2_rel numeric
## 161                                 TRBV5.4_bs2_rel numeric
## 162                                 TRBV5.5_bs2_rel numeric
## 163                                 TRBV5.6_bs2_rel numeric
## 164                                 TRBV5.8_bs2_rel numeric
## 165                                 TRBV6.1_bs2_rel numeric
## 166                                 TRBV6.2_bs2_rel numeric
## 167                                 TRBV6.3_bs2_rel numeric
## 168                                 TRBV6.4_bs2_rel numeric
## 169                                 TRBV6.5_bs2_rel numeric
## 170                                 TRBV6.6_bs2_rel numeric
## 171                                 TRBV6.8_bs2_rel numeric
## 172                                 TRBV6.9_bs2_rel numeric
## 173                                 TRBV7.2_bs2_rel numeric
## 174                                 TRBV7.3_bs2_rel numeric
## 175                                 TRBV7.4_bs2_rel numeric
## 176                                 TRBV7.6_bs2_rel numeric
## 177                                 TRBV7.7_bs2_rel numeric
## 178                                 TRBV7.8_bs2_rel numeric
## 179                                 TRBV7.9_bs2_rel numeric
## 180                                   TRBV9_bs2_rel numeric
## 181                                TRBJ1.1_bs2_rel numeric
## 182                                TRBJ1.2_bs2_rel numeric
```

```
## 183                                  TRBJ1.3_bs2_rel numeric
## 184                                  TRBJ1.4_bs2_rel numeric
## 185                                  TRBJ1.5_bs2_rel numeric
## 186                                  TRBJ1.6_bs2_rel numeric
## 187                                  TRBJ2.1_bs2_rel numeric
## 188                                  TRBJ2.2_bs2_rel numeric
## 189                                  TRBJ2.3_bs2_rel numeric
## 190                                  TRBJ2.4_bs2_rel numeric
## 191                                  TRBJ2.5_bs2_rel numeric
## 192                                  TRBJ2.6_bs2_rel numeric
## 193                                  TRBJ2.7_bs2_rel numeric
## 194                                TRBV10.1_bs2_abs numeric
## 195                                TRBV10.2_bs2_abs numeric
## 196                                TRBV10.3_bs2_abs numeric
## 197                                TRBV11.1_bs2_abs numeric
## 198                                TRBV11.2_bs2_abs numeric
## 199                                TRBV11.3_bs2_abs numeric
## 200                                TRBV12.3_bs2_abs numeric
## 201                                TRBV12.4_bs2_abs numeric
## 202                                TRBV12.5_bs2_abs numeric
## 203                                  TRBV13_bs2_abs numeric
## 204                                  TRBV14_bs2_abs numeric
## 205                                  TRBV15_bs2_abs numeric
## 206                                  TRBV16_bs2_abs numeric
## 207                                  TRBV18_bs2_abs numeric
## 208                                  TRBV19_bs2_abs numeric
## 209                                   TRBV2_bs2_abs numeric
## 210                                TRBV20.1_bs2_abs numeric
## 211                                TRBV24.1_bs2_abs numeric
## 212                                TRBV25.1_bs2_abs numeric
## 213                                  TRBV27_bs2_abs numeric
## 214                                  TRBV28_bs2_abs numeric
## 215                                TRBV29.1_bs2_abs numeric
## 216                                 TRBV3.1_bs2_abs numeric
## 217                                  TRBV30_bs2_abs numeric
## 218                                 TRBV4.1_bs2_abs numeric
## 219                                 TRBV4.2_bs2_abs numeric
## 220                                 TRBV4.3_bs2_abs numeric
## 221                                 TRBV5.1_bs2_abs numeric
## 222                                 TRBV5.4_bs2_abs numeric
## 223                                 TRBV5.5_bs2_abs numeric
## 224                                 TRBV5.6_bs2_abs numeric
## 225                                 TRBV5.8_bs2_abs numeric
## 226                                 TRBV6.1_bs2_abs numeric
## 227                                 TRBV6.2_bs2_abs numeric
## 228                                 TRBV6.3_bs2_abs numeric
## 229                                 TRBV6.4_bs2_abs numeric
## 230                                 TRBV6.5_bs2_abs numeric
## 231                                 TRBV6.6_bs2_abs numeric
## 232                                 TRBV6.8_bs2_abs numeric
## 233                                 TRBV6.9_bs2_abs numeric
## 234                                 TRBV7.2_bs2_abs numeric
## 235                                 TRBV7.3_bs2_abs numeric
## 236                                 TRBV7.4_bs2_abs numeric
```

```
## 237                                          TRBV7.6_bs2_abs numeric
## 238                                          TRBV7.7_bs2_abs numeric
## 239                                          TRBV7.8_bs2_abs numeric
## 240                                          TRBV7.9_bs2_abs numeric
## 241                                            TRBV9_bs2_abs numeric
## 242                                          TRBJ1.1_bs2_abs numeric
## 243                                          TRBJ1.2_bs2_abs numeric
## 244                                          TRBJ1.3_bs2_abs numeric
## 245                                          TRBJ1.4_bs2_abs numeric
## 246                                          TRBJ1.5_bs2_abs numeric
## 247                                          TRBJ1.6_bs2_abs numeric
## 248                                          TRBJ2.1_bs2_abs numeric
## 249                                          TRBJ2.2_bs2_abs numeric
## 250                                          TRBJ2.3_bs2_abs numeric
## 251                                          TRBJ2.4_bs2_abs numeric
## 252                                          TRBJ2.5_bs2_abs numeric
## 253                                          TRBJ2.6_bs2_abs numeric
## 254                                          TRBJ2.7_bs2_abs numeric
## 255                                       Chao1_bs2_bs2_eco numeric
## 256                                    Shannon_bs2_bs2_eco numeric
## 257                                 Simpson_GiniSimp_bs2_bs2_eco numeric
## 258                                  Pielou_Evenness_bs2_bs2_eco numeric
## 259                                 Gini_Convergence_bs2_bs2_eco numeric
## 260                                       D50_bs2_bs2_eco numeric
## 261                                       D10_bs2_bs2_eco numeric
## 262                                  Radioterapia.adyuvante  factor
## 263                                 Quimioterapia.adyuvante  factor
## 264                                           RT.QT.Radical  factor
## 265                                             BRAF..V600.  factor
## 266                                                    ROS1  factor
## 267                                                    EGFR  factor
## 268                                                     ALK  factor
## 269                                              Histología  factor
## 270                                 Tiempo.hasta.inicio.IO.metas numeric
## 271                                                   PD.L1 numeric
## 272              Fecha.de.diagnóstico.de.enfermedad.metastática    Date
## 273                                    Tiempo.hasta.inicio.IO numeric
## 274                                                 PFS..d. numeric
## 275                                                 PFS..m. numeric
## 276                                           Follow.Up..d. numeric
## 277                                             Escala.ECOG numeric
## 278                                          Fecha.de.inicio.IO    Date
## 279                                       Estadio.al.diagnóstico  factor
## 280                           Fecha.de.inicio.IO..metastáticos.    Date
## 281                                       TIPO.IO..metastáticos.  factor
## 282                           Tipo.de.IO.Cat..0.IO..1.ChIO.  factor
## 283                                               Diana.IO  factor
## 284                                    Nº.de.líneas.previas numeric
## 285                                         IO.Tipo.General  factor
## 286                                    Presencia.linea.Previa  factor
## 287                                     Fecha.de.diagnóstico    Date
## 288 Tabaquismo..0.Nunca.fumador..1.Exfumador..2..Fumador.Activo.  factor
## 289                                              Tumor.previo  factor
## 290                                   Edad.al.diagnóstico numeric
```

```
## 291                                  Paciente.code  factor
## 292                                              X numeric
## 293                                      Paciente  factor
## 294                                Quimio.Radio.Adj  factor
## 295                                     Progresión  factor
## 296                                         Exitus  factor
## 297                             Fecha.de.nacimiento    Date
## 298                           Sexo..0.Mujer..1.Varón.  factor
## 299                                 TERAPIA.BIOLÓGICA  factor
## 300                                  Mutacion.General  factor
## 301                                 TRBV10.1_bs1_rel numeric
## 302                                 TRBV10.2_bs1_rel numeric
## 303                                 TRBV10.3_bs1_rel numeric
## 304                                 TRBV11.1_bs1_rel numeric
## 305                                 TRBV11.2_bs1_rel numeric
## 306                                 TRBV11.3_bs1_rel numeric
## 307                                 TRBV12.3_bs1_rel numeric
## 308                                 TRBV12.4_bs1_rel numeric
## 309                                 TRBV12.5_bs1_rel numeric
## 310                                   TRBV13_bs1_rel numeric
## 311                                   TRBV14_bs1_rel numeric
## 312                                   TRBV15_bs1_rel numeric
## 313                                   TRBV16_bs1_rel numeric
## 314                                   TRBV18_bs1_rel numeric
## 315                                   TRBV19_bs1_rel numeric
## 316                                    TRBV2_bs1_rel numeric
## 317                                  TRBV20.1_bs1_rel numeric
## 318                                  TRBV24.1_bs1_rel numeric
## 319                                  TRBV25.1_bs1_rel numeric
## 320                                   TRBV27_bs1_rel numeric
## 321                                   TRBV28_bs1_rel numeric
## 322                                  TRBV29.1_bs1_rel numeric
## 323                                   TRBV3.1_bs1_rel numeric
## 324                                   TRBV30_bs1_rel numeric
## 325                                   TRBV4.1_bs1_rel numeric
## 326                                   TRBV4.2_bs1_rel numeric
## 327                                   TRBV4.3_bs1_rel numeric
## 328                                   TRBV5.1_bs1_rel numeric
## 329                                   TRBV5.4_bs1_rel numeric
## 330                                   TRBV5.5_bs1_rel numeric
## 331                                   TRBV5.6_bs1_rel numeric
## 332                                   TRBV5.8_bs1_rel numeric
## 333                                   TRBV6.1_bs1_rel numeric
## 334                                   TRBV6.2_bs1_rel numeric
## 335                                   TRBV6.3_bs1_rel numeric
## 336                                   TRBV6.4_bs1_rel numeric
## 337                                   TRBV6.5_bs1_rel numeric
## 338                                   TRBV6.6_bs1_rel numeric
## 339                                   TRBV6.8_bs1_rel numeric
## 340                                   TRBV6.9_bs1_rel numeric
## 341                                   TRBV7.2_bs1_rel numeric
## 342                                   TRBV7.3_bs1_rel numeric
## 343                                   TRBV7.4_bs1_rel numeric
## 344                                   TRBV7.6_bs1_rel numeric
```

```
## 345                                  TRBV7.7_bs1_rel numeric
## 346                                  TRBV7.8_bs1_rel numeric
## 347                                  TRBV7.9_bs1_rel numeric
## 348                                    TRBV9_bs1_rel numeric
## 349                                  TRBJ1.1_bs1_rel numeric
## 350                                  TRBJ1.2_bs1_rel numeric
## 351                                  TRBJ1.3_bs1_rel numeric
## 352                                  TRBJ1.4_bs1_rel numeric
## 353                                  TRBJ1.5_bs1_rel numeric
## 354                                  TRBJ1.6_bs1_rel numeric
## 355                                  TRBJ2.1_bs1_rel numeric
## 356                                  TRBJ2.2_bs1_rel numeric
## 357                                  TRBJ2.3_bs1_rel numeric
## 358                                  TRBJ2.4_bs1_rel numeric
## 359                                  TRBJ2.5_bs1_rel numeric
## 360                                  TRBJ2.6_bs1_rel numeric
## 361                                  TRBJ2.7_bs1_rel numeric
## 362                                 TRBV10.1_bs1_abs numeric
## 363                                 TRBV10.2_bs1_abs numeric
## 364                                 TRBV10.3_bs1_abs numeric
## 365                                 TRBV11.1_bs1_abs numeric
## 366                                 TRBV11.2_bs1_abs numeric
## 367                                 TRBV11.3_bs1_abs numeric
## 368                                 TRBV12.3_bs1_abs numeric
## 369                                 TRBV12.4_bs1_abs numeric
## 370                                 TRBV12.5_bs1_abs numeric
## 371                                   TRBV13_bs1_abs numeric
## 372                                   TRBV14_bs1_abs numeric
## 373                                   TRBV15_bs1_abs numeric
## 374                                   TRBV16_bs1_abs numeric
## 375                                   TRBV18_bs1_abs numeric
## 376                                   TRBV19_bs1_abs numeric
## 377                                    TRBV2_bs1_abs numeric
## 378                                 TRBV20.1_bs1_abs numeric
## 379                                 TRBV24.1_bs1_abs numeric
## 380                                 TRBV25.1_bs1_abs numeric
## 381                                   TRBV27_bs1_abs numeric
## 382                                   TRBV28_bs1_abs numeric
## 383                                 TRBV29.1_bs1_abs numeric
## 384                                  TRBV3.1_bs1_abs numeric
## 385                                   TRBV30_bs1_abs numeric
## 386                                  TRBV4.1_bs1_abs numeric
## 387                                  TRBV4.2_bs1_abs numeric
## 388                                  TRBV4.3_bs1_abs numeric
## 389                                  TRBV5.1_bs1_abs numeric
## 390                                  TRBV5.4_bs1_abs numeric
## 391                                  TRBV5.5_bs1_abs numeric
## 392                                  TRBV5.6_bs1_abs numeric
## 393                                  TRBV5.8_bs1_abs numeric
## 394                                  TRBV6.1_bs1_abs numeric
## 395                                  TRBV6.2_bs1_abs numeric
## 396                                  TRBV6.3_bs1_abs numeric
## 397                                  TRBV6.4_bs1_abs numeric
## 398                                  TRBV6.5_bs1_abs numeric
```

```
## 399                                                TRBV6.6_bs1_abs numeric
## 400                                                TRBV6.8_bs1_abs numeric
## 401                                                TRBV6.9_bs1_abs numeric
## 402                                                TRBV7.2_bs1_abs numeric
## 403                                                TRBV7.3_bs1_abs numeric
## 404                                                TRBV7.4_bs1_abs numeric
## 405                                                TRBV7.6_bs1_abs numeric
## 406                                                TRBV7.7_bs1_abs numeric
## 407                                                TRBV7.8_bs1_abs numeric
## 408                                                TRBV7.9_bs1_abs numeric
## 409                                                  TRBV9_bs1_abs numeric
## 410                                                TRBJ1.1_bs1_abs numeric
## 411                                                TRBJ1.2_bs1_abs numeric
## 412                                                TRBJ1.3_bs1_abs numeric
## 413                                                TRBJ1.4_bs1_abs numeric
## 414                                                TRBJ1.5_bs1_abs numeric
## 415                                                TRBJ1.6_bs1_abs numeric
## 416                                                TRBJ2.1_bs1_abs numeric
## 417                                                TRBJ2.2_bs1_abs numeric
## 418                                                TRBJ2.3_bs1_abs numeric
## 419                                                TRBJ2.4_bs1_abs numeric
## 420                                                TRBJ2.5_bs1_abs numeric
## 421                                                TRBJ2.6_bs1_abs numeric
## 422                                                TRBJ2.7_bs1_abs numeric
## 423                                           Chao1_bs1_bs1_eco numeric
## 424                                         Shannon_bs1_bs1_eco numeric
## 425                                 Simpson_GiniSimp_bs1_bs1_eco numeric
## 426                                  Pielou_Evenness_bs1_bs1_eco numeric
## 427                                 Gini_Convergence_bs1_bs1_eco numeric
## 428                                              D50_bs1_bs1_eco numeric
## 429                                              D10_bs1_bs1_eco numeric
##     n_missing pct_missing
## 1         213  93.8325991
## 2         213  93.8325991
## 3         213  93.8325991
## 4         213  93.8325991
## 5         213  93.8325991
## 6         213  93.8325991
## 7         213  93.8325991
## 8         213  93.8325991
## 9         213  93.8325991
## 10        213  93.8325991
## 11        213  93.8325991
## 12        213  93.8325991
## 13        213  93.8325991
## 14        213  93.8325991
## 15        213  93.8325991
## 16        213  93.8325991
## 17        213  93.8325991
## 18        213  93.8325991
## 19        213  93.8325991
## 20        213  93.8325991
## 21        213  93.8325991
## 22        213  93.8325991
```

```
## 23        213  93.8325991
## 24        213  93.8325991
## 25        213  93.8325991
## 26        213  93.8325991
## 27        213  93.8325991
## 28        213  93.8325991
## 29        213  93.8325991
## 30        213  93.8325991
## 31        213  93.8325991
## 32        213  93.8325991
## 33        213  93.8325991
## 34        213  93.8325991
## 35        213  93.8325991
## 36        213  93.8325991
## 37        213  93.8325991
## 38        213  93.8325991
## 39        213  93.8325991
## 40        213  93.8325991
## 41        213  93.8325991
## 42        213  93.8325991
## 43        213  93.8325991
## 44        213  93.8325991
## 45        213  93.8325991
## 46        213  93.8325991
## 47        213  93.8325991
## 48        213  93.8325991
## 49        213  93.8325991
## 50        213  93.8325991
## 51        213  93.8325991
## 52        213  93.8325991
## 53        213  93.8325991
## 54        213  93.8325991
## 55        213  93.8325991
## 56        213  93.8325991
## 57        213  93.8325991
## 58        213  93.8325991
## 59        213  93.8325991
## 60        213  93.8325991
## 61        213  93.8325991
## 62        213  93.8325991
## 63        213  93.8325991
## 64        213  93.8325991
## 65        213  93.8325991
## 66        213  93.8325991
## 67        213  93.8325991
## 68        213  93.8325991
## 69        213  93.8325991
## 70        213  93.8325991
## 71        213  93.8325991
## 72        213  93.8325991
## 73        213  93.8325991
## 74        213  93.8325991
## 75        213  93.8325991
## 76        213  93.8325991
```

```
## 77        213  93.8325991
## 78        213  93.8325991
## 79        213  93.8325991
## 80        213  93.8325991
## 81        213  93.8325991
## 82        213  93.8325991
## 83        213  93.8325991
## 84        213  93.8325991
## 85        213  93.8325991
## 86        213  93.8325991
## 87        213  93.8325991
## 88        213  93.8325991
## 89        213  93.8325991
## 90        213  93.8325991
## 91        213  93.8325991
## 92        213  93.8325991
## 93        213  93.8325991
## 94        213  93.8325991
## 95        213  93.8325991
## 96        213  93.8325991
## 97        213  93.8325991
## 98        213  93.8325991
## 99        213  93.8325991
## 100       213  93.8325991
## 101       213  93.8325991
## 102       213  93.8325991
## 103       213  93.8325991
## 104       213  93.8325991
## 105       213  93.8325991
## 106       213  93.8325991
## 107       213  93.8325991
## 108       213  93.8325991
## 109       213  93.8325991
## 110       213  93.8325991
## 111       213  93.8325991
## 112       213  93.8325991
## 113       213  93.8325991
## 114       213  93.8325991
## 115       213  93.8325991
## 116       213  93.8325991
## 117       213  93.8325991
## 118       213  93.8325991
## 119       213  93.8325991
## 120       213  93.8325991
## 121       213  93.8325991
## 122       213  93.8325991
## 123       213  93.8325991
## 124       213  93.8325991
## 125       213  93.8325991
## 126       213  93.8325991
## 127       213  93.8325991
## 128       213  93.8325991
## 129       213  93.8325991
## 130       162  71.3656388
```

```
## 131       153  67.4008811
## 132       142  62.5550661
## 133       132  58.1497797
## 134       132  58.1497797
## 135       132  58.1497797
## 136       132  58.1497797
## 137       132  58.1497797
## 138       132  58.1497797
## 139       132  58.1497797
## 140       132  58.1497797
## 141       132  58.1497797
## 142       132  58.1497797
## 143       132  58.1497797
## 144       132  58.1497797
## 145       132  58.1497797
## 146       132  58.1497797
## 147       132  58.1497797
## 148       132  58.1497797
## 149       132  58.1497797
## 150       132  58.1497797
## 151       132  58.1497797
## 152       132  58.1497797
## 153       132  58.1497797
## 154       132  58.1497797
## 155       132  58.1497797
## 156       132  58.1497797
## 157       132  58.1497797
## 158       132  58.1497797
## 159       132  58.1497797
## 160       132  58.1497797
## 161       132  58.1497797
## 162       132  58.1497797
## 163       132  58.1497797
## 164       132  58.1497797
## 165       132  58.1497797
## 166       132  58.1497797
## 167       132  58.1497797
## 168       132  58.1497797
## 169       132  58.1497797
## 170       132  58.1497797
## 171       132  58.1497797
## 172       132  58.1497797
## 173       132  58.1497797
## 174       132  58.1497797
## 175       132  58.1497797
## 176       132  58.1497797
## 177       132  58.1497797
## 178       132  58.1497797
## 179       132  58.1497797
## 180       132  58.1497797
## 181       132  58.1497797
## 182       132  58.1497797
## 183       132  58.1497797
## 184       132  58.1497797
```

```
## 185       132  58.1497797
## 186       132  58.1497797
## 187       132  58.1497797
## 188       132  58.1497797
## 189       132  58.1497797
## 190       132  58.1497797
## 191       132  58.1497797
## 192       132  58.1497797
## 193       132  58.1497797
## 194       132  58.1497797
## 195       132  58.1497797
## 196       132  58.1497797
## 197       132  58.1497797
## 198       132  58.1497797
## 199       132  58.1497797
## 200       132  58.1497797
## 201       132  58.1497797
## 202       132  58.1497797
## 203       132  58.1497797
## 204       132  58.1497797
## 205       132  58.1497797
## 206       132  58.1497797
## 207       132  58.1497797
## 208       132  58.1497797
## 209       132  58.1497797
## 210       132  58.1497797
## 211       132  58.1497797
## 212       132  58.1497797
## 213       132  58.1497797
## 214       132  58.1497797
## 215       132  58.1497797
## 216       132  58.1497797
## 217       132  58.1497797
## 218       132  58.1497797
## 219       132  58.1497797
## 220       132  58.1497797
## 221       132  58.1497797
## 222       132  58.1497797
## 223       132  58.1497797
## 224       132  58.1497797
## 225       132  58.1497797
## 226       132  58.1497797
## 227       132  58.1497797
## 228       132  58.1497797
## 229       132  58.1497797
## 230       132  58.1497797
## 231       132  58.1497797
## 232       132  58.1497797
## 233       132  58.1497797
## 234       132  58.1497797
## 235       132  58.1497797
## 236       132  58.1497797
## 237       132  58.1497797
## 238       132  58.1497797
```

```
## 239       132  58.1497797
## 240       132  58.1497797
## 241       132  58.1497797
## 242       132  58.1497797
## 243       132  58.1497797
## 244       132  58.1497797
## 245       132  58.1497797
## 246       132  58.1497797
## 247       132  58.1497797
## 248       132  58.1497797
## 249       132  58.1497797
## 250       132  58.1497797
## 251       132  58.1497797
## 252       132  58.1497797
## 253       132  58.1497797
## 254       132  58.1497797
## 255       132  58.1497797
## 256       132  58.1497797
## 257       132  58.1497797
## 258       132  58.1497797
## 259       132  58.1497797
## 260       132  58.1497797
## 261       132  58.1497797
## 262       122  53.7444934
## 263       121  53.3039648
## 264       103  45.3744493
## 265        94  41.4096916
## 266        68  29.9559471
## 267        62  27.3127753
## 268        60  26.4317181
## 269        16   7.0484581
## 270        14   6.1674009
## 271        12   5.2863436
## 272        12   5.2863436
## 273         9   3.9647577
## 274         8   3.5242291
## 275         8   3.5242291
## 276         8   3.5242291
## 277         8   3.5242291
## 278         7   3.0837004
## 279         7   3.0837004
## 280         7   3.0837004
## 281         7   3.0837004
## 282         4   1.7621145
## 283         4   1.7621145
## 284         4   1.7621145
## 285         4   1.7621145
## 286         4   1.7621145
## 287         2   0.8810573
## 288         1   0.4405286
## 289         1   0.4405286
## 290         1   0.4405286
## 291         0   0.0000000
## 292         0   0.0000000
```

```
## 293         0   0.0000000
## 294         0   0.0000000
## 295         0   0.0000000
## 296         0   0.0000000
## 297         0   0.0000000
## 298         0   0.0000000
## 299         0   0.0000000
## 300         0   0.0000000
## 301         0   0.0000000
## 302         0   0.0000000
## 303         0   0.0000000
## 304         0   0.0000000
## 305         0   0.0000000
## 306         0   0.0000000
## 307         0   0.0000000
## 308         0   0.0000000
## 309         0   0.0000000
## 310         0   0.0000000
## 311         0   0.0000000
## 312         0   0.0000000
## 313         0   0.0000000
## 314         0   0.0000000
## 315         0   0.0000000
## 316         0   0.0000000
## 317         0   0.0000000
## 318         0   0.0000000
## 319         0   0.0000000
## 320         0   0.0000000
## 321         0   0.0000000
## 322         0   0.0000000
## 323         0   0.0000000
## 324         0   0.0000000
## 325         0   0.0000000
## 326         0   0.0000000
## 327         0   0.0000000
## 328         0   0.0000000
## 329         0   0.0000000
## 330         0   0.0000000
## 331         0   0.0000000
## 332         0   0.0000000
## 333         0   0.0000000
## 334         0   0.0000000
## 335         0   0.0000000
## 336         0   0.0000000
## 337         0   0.0000000
## 338         0   0.0000000
## 339         0   0.0000000
## 340         0   0.0000000
## 341         0   0.0000000
## 342         0   0.0000000
## 343         0   0.0000000
## 344         0   0.0000000
## 345         0   0.0000000
## 346         0   0.0000000
```

```
## 347         0   0.0000000
## 348         0   0.0000000
## 349         0   0.0000000
## 350         0   0.0000000
## 351         0   0.0000000
## 352         0   0.0000000
## 353         0   0.0000000
## 354         0   0.0000000
## 355         0   0.0000000
## 356         0   0.0000000
## 357         0   0.0000000
## 358         0   0.0000000
## 359         0   0.0000000
## 360         0   0.0000000
## 361         0   0.0000000
## 362         0   0.0000000
## 363         0   0.0000000
## 364         0   0.0000000
## 365         0   0.0000000
## 366         0   0.0000000
## 367         0   0.0000000
## 368         0   0.0000000
## 369         0   0.0000000
## 370         0   0.0000000
## 371         0   0.0000000
## 372         0   0.0000000
## 373         0   0.0000000
## 374         0   0.0000000
## 375         0   0.0000000
## 376         0   0.0000000
## 377         0   0.0000000
## 378         0   0.0000000
## 379         0   0.0000000
## 380         0   0.0000000
## 381         0   0.0000000
## 382         0   0.0000000
## 383         0   0.0000000
## 384         0   0.0000000
## 385         0   0.0000000
## 386         0   0.0000000
## 387         0   0.0000000
## 388         0   0.0000000
## 389         0   0.0000000
## 390         0   0.0000000
## 391         0   0.0000000
## 392         0   0.0000000
## 393         0   0.0000000
## 394         0   0.0000000
## 395         0   0.0000000
## 396         0   0.0000000
## 397         0   0.0000000
## 398         0   0.0000000
## 399         0   0.0000000
## 400         0   0.0000000
```

```
## 401         0   0.0000000
## 402         0   0.0000000
## 403         0   0.0000000
## 404         0   0.0000000
## 405         0   0.0000000
## 406         0   0.0000000
## 407         0   0.0000000
## 408         0   0.0000000
## 409         0   0.0000000
## 410         0   0.0000000
## 411         0   0.0000000
## 412         0   0.0000000
## 413         0   0.0000000
## 414         0   0.0000000
## 415         0   0.0000000
## 416         0   0.0000000
## 417         0   0.0000000
## 418         0   0.0000000
## 419         0   0.0000000
## 420         0   0.0000000
## 421         0   0.0000000
## 422         0   0.0000000
## 423         0   0.0000000
## 424         0   0.0000000
## 425         0   0.0000000
## 426         0   0.0000000
## 427         0   0.0000000
## 428         0   0.0000000
## 429         0   0.0000000
```

Estadísticos de variables numéricas

```
head(skim(merged_full))
```

Table 1: Data summary

Name	merged_full
Number of rows	227
Number of columns	429
_____	
Column type frequency:	
Date	5
factor	1
_____	
Group variables	None

Variable type: Date

skim_variable	n_missing	complete_rate	min	max	median	n_unique
Fecha.de.inicio.IO	7	0.97	2018-06-19	2023-07-24	2021-08-08	190

skim_variable	n_missing	complete_rate	min	max	median	n_unique
Fecha.de.nacimiento	0	1.00	1933-06-27	1992-04-30	1953-08-08	223
Fecha.de.diagnóstico	2	0.99	2010-01-28	2023-07-06	2021-05-06	204
Fecha.de.diagnóstico.de.enfermedad.metastática	12	0.95	2013-07-01	2023-07-06	2021-05-20	194
Fecha.de.inicio.IO..metastáticos.	7	0.97	2018-06-19	2023-07-24	2021-08-14	190

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
Paciente.code	0	1	FALSE	227	01-: 1, 01-: 1, 01-: 1, 01-: 1

Frecuencias de variables categóricas

```
# Seleccionar variables factor o character
cat_vars <- merged_full %>% select(where(~ is.factor(.) || is.character(.)))

# Para cada variable categórica, tabla de frecuencias (incluye NA)
freq_list <- lapply(cat_vars, function(x) {
  as.data.frame(table(x, useNA = "ifany"))
})
freq_list
```

```
## $Paciente.code
##          x Freq
## 1   01-003    1
## 2   01-004    1
## 3   01-005    1
## 4   01-010    1
## 5   01-011    1
## 6   01-012    1
## 7   01-013    1
## 8   01-015    1
## 9   01-016    1
## 10  01-017    1
## 11  01-018    1
## 12  01-019    1
## 13  01-023    1
## 14  01-024    1
## 15  01-025    1
## 16  01-026    1
## 17  01-027    1
## 18  01-028    1
## 19  01-029    1
## 20  01-034    1
## 21  01-035    1
## 22  01-036    1
## 23  01-038    1
```

```
## 24  01-039    1
## 25  01-040    1
## 26  01-041    1
## 27  01-044    1
## 28  01-045    1
## 29  01-046    1
## 30  01-047    1
## 31  01-048    1
## 32  01-049    1
## 33  01-050    1
## 34  01-051    1
## 35  01-054    1
## 36  01-059    1
## 37  01-061    1
## 38  01-062    1
## 39  01-064    1
## 40  01-066    1
## 41  01-071    1
## 42  01-073    1
## 43  01-074    1
## 44  01-076    1
## 45  01-077    1
## 46  01-081    1
## 47  01-082    1
## 48  01-084    1
## 49  01-085    1
## 50  01-086    1
## 51  01-089    1
## 52  01-090    1
## 53  01-091    1
## 54  01-092    1
## 55  01-093    1
## 56  01-095    1
## 57  01-096    1
## 58  01-097    1
## 59  01-098    1
## 60  01-099    1
## 61  01-100    1
## 62  01-102    1
## 63  01-104    1
## 64  01-107    1
## 65  01-108    1
## 66  01-109    1
## 67  01-110    1
## 68  01-111    1
## 69  01-112    1
## 70  01-113    1
## 71  01-115    1
## 72  01-116    1
## 73  01-117    1
## 74  01-118    1
## 75  01-119    1
## 76  01-120    1
## 77  01-121    1
```

```
## 78  01-122    1
## 79  01-123    1
## 80  01-124    1
## 81  01-125    1
## 82  01-127    1
## 83  01-129    1
## 84  01-132    1
## 85  01-133    1
## 86  01-134    1
## 87  01-135    1
## 88  01-136    1
## 89  01-137    1
## 90  01-138    1
## 91  01-139    1
## 92  01-140    1
## 93  01-141    1
## 94  01-143    1
## 95  01-144    1
## 96  01-146    1
## 97  01-147    1
## 98  01-148    1
## 99  01-149    1
## 100 01-150    1
## 101 01-151    1
## 102 01-153    1
## 103 01-156    1
## 104 01-158    1
## 105 01-159    1
## 106 01-164    1
## 107 01-165    1
## 108 01-166    1
## 109 01-167    1
## 110 01-168    1
## 111 01-171    1
## 112 01-172    1
## 113 01-174    1
## 114 01-180    1
## 115 01-189    1
## 116 01-190    1
## 117 01-191    1
## 118 01-192    1
## 119 01-197    1
## 120 01-198    1
## 121 01-199    1
## 122 01-202    1
## 123 02-001    1
## 124 02-003    1
## 125 02-004    1
## 126 02-005    1
## 127 02-007    1
## 128 02-009    1
## 129 02-010    1
## 130 02-012    1
## 131 02-014    1
```

```
## 132 02-015    1
## 133 02-016    1
## 134 02-017    1
## 135 02-018    1
## 136 02-019    1
## 137 02-020    1
## 138 02-021    1
## 139 02-022    1
## 140 02-025    1
## 141 02-026    1
## 142 03-001    1
## 143 03-002    1
## 144 03-003    1
## 145 04-001    1
## 146 04-002    1
## 147 04-003    1
## 148 04-005    1
## 149 04-009    1
## 150 04-010    1
## 151 04-011    1
## 152 04-012    1
## 153 04-015    1
## 154 05-001    1
## 155 05-002    1
## 156 05-003    1
## 157 05-005    1
## 158 05-006    1
## 159 05-007    1
## 160 05-008    1
## 161 05-011    1
## 162 06-003    1
## 163 06-004    1
## 164 06-005    1
## 165 06-006    1
## 166 06-007    1
## 167 06-008    1
## 168 07-001    1
## 169 07-002    1
## 170 07-003    1
## 171 07-004    1
## 172 07-005    1
## 173 07-006    1
## 174 07-007    1
## 175 07-008    1
## 176 07-009    1
## 177 07-010    1
## 178 07-011    1
## 179 07-012    1
## 180 07-013    1
## 181 07-014    1
## 182 07-015    1
## 183 07-017    1
## 184 07-018    1
## 185 07-019    1
```

```
## 186 07-020    1
## 187 07-021    1
## 188 07-022    1
## 189 07-023    1
## 190 07-024    1
## 191 07-025    1
## 192 07-026    1
## 193 08-004    1
## 194 08-005    1
## 195 08-006    1
## 196 08-009    1
## 197 09-001    1
## 198 09-003    1
## 199 09-005    1
## 200 09-007    1
## 201 09-008    1
## 202 09-009    1
## 203 09-010    1
## 204 09-011    1
## 205 09-013    1
## 206 09-014    1
## 207 09-025    1
## 208 11-001    1
## 209 11-002    1
## 210 11-003    1
## 211 11-004    1
## 212 11-006    1
## 213 11-008    1
## 214 12-002    1
## 215 12-003    1
## 216 12-004    1
## 217 12-005    1
## 218 12-006    1
## 219 12-007    1
## 220 12-008    1
## 221 12-009    1
## 222 12-010    1
## 223 12-012    1
## 224 13-002    1
## 225 13-007    1
## 226 13-010    1
## 227 13-013    1
##
## $Paciente
##             x Freq
## 1   CG-001-PU    1
## 2   CG-002-PU    1
## 3   CG-003-PU    1
## 4   CG-004-PU    1
## 5   CG-005-PU    1
## 6   CG-006-PU    1
## 7   CG-007-PU    1
## 8   CG-008-PU    1
## 9   CG-009-PU    1
```

```
## 10  CG-010-PU    1
## 11  CG-011-PU    1
## 12  CG-012-PU    1
## 13  CG-013-PU    1
## 14  CG-014-PU    1
## 15  CG-015-PU    1
## 16  CG-017-PU    1
## 17  CG-018-PU    1
## 18  CG-019-PU    1
## 19  CG-020-PU    1
## 20  CG-021-PU    1
## 21  CG-022-PU    1
## 22  CG-023-PU    1
## 23  CG-024-PU    1
## 24  CG-025-PU    1
## 25  CG-026-PU    1
## 26  FA-001-PU    1
## 27  FA-002-PU    1
## 28  FA-003-PU    1
## 29  FA-005-PU    1
## 30  FA-009-PU    1
## 31  FA-010-PU    1
## 32  FA-011-PU    1
## 33  FA-012-PU    1
## 34  FA-015-PU    1
## 35  HB-001-PU    1
## 36  HB-002-PU    1
## 37  HB-003-PU    1
## 38  HB-004-PU    1
## 39  HB-006-PU    1
## 40  HB-008-PU    1
## 41  IL-001-PU    1
## 42  IL-002-PU    1
## 43  IL-003-PU    1
## 44  IL-005-PU    1
## 45  IL-006-PU    1
## 46  IL-007-PU    1
## 47  IL-008-PU    1
## 48  IL-011-PU    1
## 49  LA-001-PU    1
## 50  LA-003-PU    1
## 51  LA-005-PU    1
## 52  LA-007-PU    1
## 53  LA-008-PU    1
## 54  LA-009-PU    1
## 55  LA-010-PU    1
## 56  LA-011-PU    1
## 57  LA-013-PU    1
## 58  LA-014-PU    1
## 59  LA-025-PU    1
## 60  LE-001-PU    1
## 61  LE-003-PU    1
## 62  LE-004-PU    1
## 63  LE-005-PU    1
```

```
## 64  LE-007-PU    1
## 65  LE-009-PU    1
## 66  LE-010-PU    1
## 67  LE-012-PU    1
## 68  LE-014-PU    1
## 69  LE-015-PU    1
## 70  LE-016-PU    1
## 71  LE-017-PU    1
## 72  LE-018-PU    1
## 73  LE-019-PU    1
## 74  LE-020-PU    1
## 75  LE-021-PU    1
## 76  LE-022-PU    1
## 77  LE-025-PU    1
## 78  LE-026-PU    1
## 79  PC-003-PU    1
## 80  PC-004-PU    1
## 81  PC-005-PU    1
## 82  PC-006-PU    1
## 83  PC-007-PU    1
## 84  PC-008-PU    1
## 85  PH-003-PU    1
## 86  PH-004-PU    1
## 87  PH-005-PU    1
## 88  PH-010-PU    1
## 89  PH-011-PU    1
## 90  PH-012-PU    1
## 91  PH-013-PU    1
## 92  PH-015-PU    1
## 93  PH-016-PU    1
## 94  PH-017-PU    1
## 95  PH-018-PU    1
## 96  PH-019-PU    1
## 97  PH-023-PU    1
## 98  PH-024-PU    1
## 99  PH-025-PU    1
## 100 PH-026-PU    1
## 101 PH-027-PU    1
## 102 PH-028-PU    1
## 103 PH-029-PU    1
## 104 PH-034-PU    1
## 105 PH-035-PU    1
## 106 PH-036-PU    1
## 107 PH-038-PU    1
## 108 PH-039-PU    1
## 109 PH-040-PU    1
## 110 PH-041-PU    1
## 111 PH-044-PU    1
## 112 PH-045-PU    1
## 113 PH-046-PU    1
## 114 PH-047-PU    1
## 115 PH-048-PU    1
## 116 PH-049-PU    1
## 117 PH-050-PU    1
```

```
## 118 PH-051-PU    1
## 119 PH-054-PU    1
## 120 PH-059-PU    1
## 121 PH-061-PU    1
## 122 PH-062-PU    1
## 123 PH-064-PU    1
## 124 PH-066-PU    1
## 125 PH-071-PU    1
## 126 PH-073-PU    1
## 127 PH-074-PU    1
## 128 PH-076-PU    1
## 129 PH-077-PU    1
## 130 PH-081-PU    1
## 131 PH-082-PU    1
## 132 PH-084-PU    1
## 133 PH-085-PU    1
## 134 PH-086-PU    1
## 135 PH-089-PU    1
## 136 PH-090-PU    1
## 137 PH-091-PU    1
## 138 PH-092-PU    1
## 139 PH-093-PU    1
## 140 PH-095-PU    1
## 141 PH-096-PU    1
## 142 PH-097-PU    1
## 143 PH-098-PU    1
## 144 PH-099-PU    1
## 145 PH-100-PU    1
## 146 PH-102-PU    1
## 147 PH-104-PU    1
## 148 PH-107-PU    1
## 149 PH-108-PU    1
## 150 PH-109-PU    1
## 151 PH-110-PU    1
## 152 PH-111-PU    1
## 153 PH-112-PU    1
## 154 PH-113-PU    1
## 155 PH-115-PU    1
## 156 PH-116-PU    1
## 157 PH-117-PU    1
## 158 PH-118-PU    1
## 159 PH-119-PU    1
## 160 PH-120-PU    1
## 161 PH-121-PU    1
## 162 PH-122-PU    1
## 163 PH-123-PU    1
## 164 PH-124-PU    1
## 165 PH-125-PU    1
## 166 PH-127-PU    1
## 167 PH-129-PU    1
## 168 PH-132-PU    1
## 169 PH-133-PU    1
## 170 PH-134-PU    1
## 171 PH-135-PU    1
```

```
## 172 PH-136-PU    1
## 173 PH-137-PU    1
## 174 PH-138-PU    1
## 175 PH-139-PU    1
## 176 PH-140-PU    1
## 177 PH-141-PU    1
## 178 PH-143-PU    1
## 179 PH-144-PU    1
## 180 PH-146-PU    1
## 181 PH-147-PU    1
## 182 PH-148-PU    1
## 183 PH-149-PU    1
## 184 PH-150-PU    1
## 185 PH-151-PU    1
## 186 PH-153-PU    1
## 187 PH-156-PU    1
## 188 PH-158-PU    1
## 189 PH-159-PU    1
## 190 PH-164-PU    1
## 191 PH-165-PU    1
## 192 PH-166-PU    1
## 193 PH-167-PU    1
## 194 PH-168-PU    1
## 195 PH-171-PU    1
## 196 PH-172-PU    1
## 197 PH-174-PU    1
## 198 PH-180-PU    1
## 199 PH-189-PU    1
## 200 PH-190-PU    1
## 201 PH-191-PU    1
## 202 PH-192-PU    1
## 203 PH-197-PU    1
## 204 PH-198-PU    1
## 205 PH-199-PU    1
## 206 PH-202-PU    1
## 207 SL-004-PU    1
## 208 SL-005-PU    1
## 209 SL-006-PU    1
## 210 SL-009-PU    1
## 211 SO-001-PU    1
## 212 SO-002-PU    1
## 213 SO-003-PU    1
## 214 UA-002-PU    1
## 215 UA-007-PU    1
## 216 UA-010-PU    1
## 217 UA-013-PU    1
## 218 UC-002-PU    1
## 219 UC-003-PU    1
## 220 UC-004-PU    1
## 221 UC-005-PU    1
## 222 UC-006-PU    1
## 223 UC-007-PU    1
## 224 UC-008-PU    1
## 225 UC-009-PU    1
```

```
## 226 UC-010-PU    1
## 227 UC-012-PU    1
##
## $Quimio.Radio.Adj
##                           x Freq
## 1    Desconocido/No aplica  197
## 2 Quimio y Radio adyuvante    8
## 3  Quimioterapia adyuvante   17
## 4   Radioterapia adyuvante    5
##
## $Progresión
##    x Freq
## 1 No   48
## 2 Sí  179
##
## $Exitus
##    x Freq
## 1 No   78
## 2 Sí  149
##
## $Sexo..0.Mujer..1.Varón.
##       x Freq
## 1 Mujer   62
## 2 Varón  165
##
## $Tabaquismo..0.Nunca.fumador..1.Exfumador..2..Fumador.Activo.
##           x Freq
## 1 Exfumador  128
## 2   Fumador   85
## 3     Nunca   13
## 4      <NA>    1
##
## $Tumor.previo
##      x Freq
## 1   No  191
## 2   Si   35
## 3 <NA>    1
##
## $Estadio.al.diagnóstico
##       x Freq
## 1    IA    7
## 2    IB    2
## 3   IIA    4
## 4   IIB    5
## 5  IIIA   18
## 6  IIIB   12
## 7  IIIC   16
## 8   IVA   79
## 9   IVB   77
## 10 <NA>    7
##
## $Histología
##                                       x Freq
## 1                           Adenocarcinoma  147
```

```
## 2                   Carcinoma adenoescamoso    3
## 3                        Carcinoma escamoso   56
## 4 Carcinoma neuroendocrino de célula grande    4
## 5                     Carcinoma sarcomatoide    1
## 6                                       <NA>   16
##
## $EGFR
##           x Freq
## 1    Mutado    4
## 2 No Mutado  161
## 3      <NA>   62
##
## $ALK
##                x Freq
## 1 No translocado  164
## 2    Translocado    3
## 3           <NA>   60
##
## $ROS1
##                x Freq
## 1 No translocado  156
## 2    Translocado    3
## 3           <NA>   68
##
## $RET
##                x Freq
## 1 No translocado   61
## 2    Translocado    4
## 3           <NA>  162
##
## $BRAF..V600.
##           x Freq
## 1    Mutado    1
## 2 No Mutado  132
## 3      <NA>   94
##
## $KRAS
##           x Freq
## 1    Mutado   21
## 2 No Mutado   53
## 3      <NA>  153
##
## $Quimioterapia.adyuvante
##      x Freq
## 1   No   81
## 2   Si   25
## 3 <NA>  121
##
## $Radioterapia.adyuvante
##      x Freq
## 1   No   92
## 2   Si   13
## 3 <NA>  122
##
```

```
## $RT.QT.Radical
##      x Freq
## 1   No   98
## 2   Si   26
## 3 <NA>  103
##
## $TIPO.IO..metastáticos.
##                                 x Freq
## 1                   Inmunoterapia  143
## 2 Inmunoterapia + antiangiogénico    7
## 3   Inmunoterapia + quimioterapia   70
## 4                           <NA>    7
##
## $Tipo.de.IO.Cat..0.IO..1.ChIO.
##              x Freq
## 1         ChIO   70
## 2           IO  143
## 3 IO combinado   10
## 4         <NA>    4
##
## $Diana.IO
##            x Freq
## 1       Otro    5
## 2       PD-1  162
## 3 PD-1,PD-L1    2
## 4      PD-L1   53
## 5 PD-L1,Otro    1
## 6       <NA>    4
##
## $BS1.BS2
##      x Freq
## 1   si   85
## 2 <NA>  142
##
## $TERAPIA.BIOLÓGICA
##    x Freq
## 1 NO  199
## 2 SI   28
##
## $Mutacion.General
##             x Freq
## 1 Desconocido  188
## 2          No   26
## 3          Si   13
##
## $IO.Tipo.General
##                                 x Freq
## 1                   Inmunoterapia  143
## 2 Inmunoterapia + antiangiogénico    7
## 3   Inmunoterapia + quimioterapia   70
## 4             IO adyuvante no met    3
## 5                           <NA>    4
##
## $Presencia.linea.Previa
```

```
##      x Freq
## 1   No  147
## 2   Si   76
## 3 <NA>    4
```

Eliminación previa de missing values

Antes de realizar la imputación de los missing values probamos a eliminar todos los registros con NAs y calcular el numero de pacientes final.

```r
# 1) Número de filas (pacientes) antes de limpiar
n_before <- nrow(merged_full)

# 2) Eliminamos filas con cualquier NA
merged_clean <- na.omit(merged_full)

# 3) Número de filas tras limpieza
n_after <- nrow(merged_clean)

# 4) Cálculo de cuántas filas se han eliminado
n_removed <- n_before - n_after

# 5) Cálculo del porcentaje eliminado
pct_removed <- n_removed / n_before * 100

# 6) Mostramos los resultados
cat("Pacientes antes de limpiar: ", n_before, "\n")
```

```
## Pacientes antes de limpiar:  227
```

```r
cat("Pacientes tras limpiar:    ", n_after, "\n")
```

```
## Pacientes tras limpiar:     0
```

```r
cat("Pacientes eliminados:      ", n_removed, "\n")
```

```
## Pacientes eliminados:       227
```

```r
cat(sprintf("Porcentaje eliminados:     %.2f%%\n", pct_removed))
```

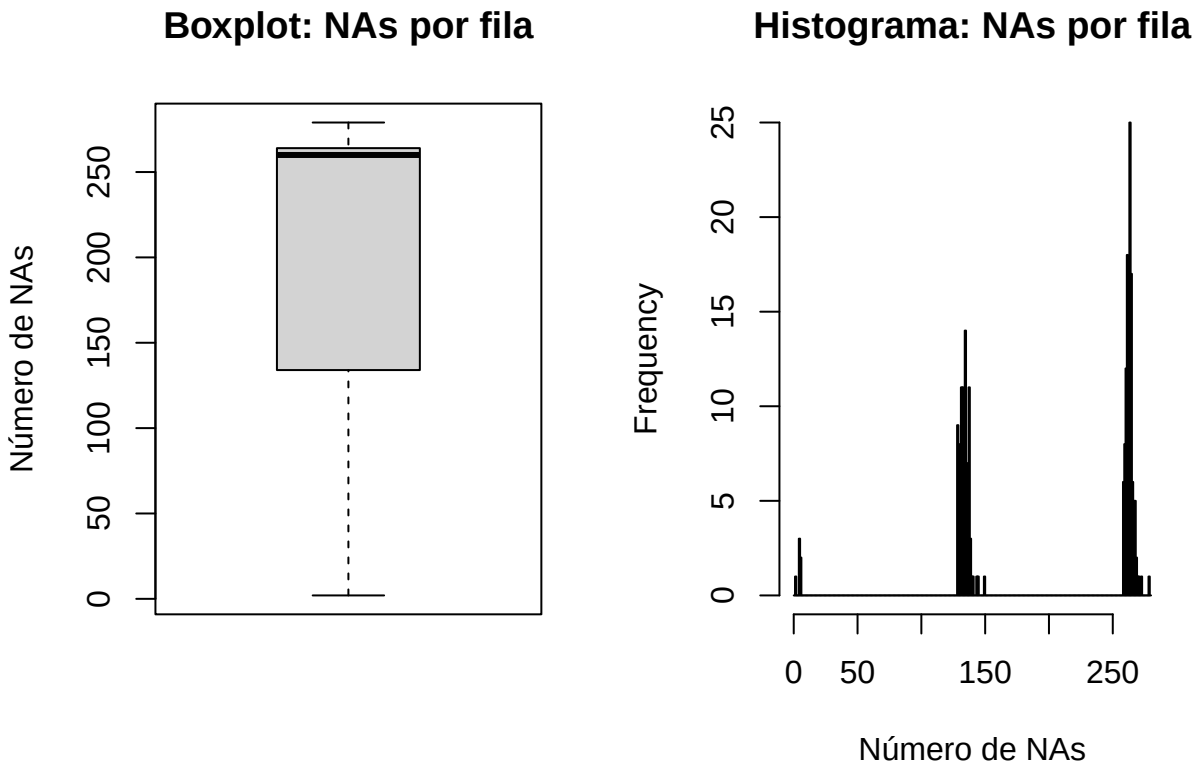
```
## Porcentaje eliminados:     100.00%
```

```r
# 1) Filas "eliminadas": las que NO son complete.cases()
rows_removed <- merged_full[!complete.cases(merged_full), ]

# 2) Contar NAs por fila
na_count <- apply(rows_removed, 1, function(x) sum(is.na(x)))

# 3) Media de NAs por fila eliminada
mean_na <- mean(na_count)
cat(sprintf("Media de NAs por fila eliminada: %.2f\n", mean_na))
```

```
## Media de NAs por fila eliminada: 201.50
```

```r
# 4) Boxplot + Histograma juntos
par(mfrow = c(1, 2))           # 1 fila, 2 columnas
boxplot(na_count,
```

```
        main = "Boxplot: NAs por fila",
        ylab = "Número de NAs")
hist(na_count,
     main = "Histograma: NAs por fila",
     xlab = "Número de NAs",
     breaks = seq(0, max(na_count)+1, by = 1))
```




```
# (Opcional) Restaurar parámetros gráficos por defecto
par(mfrow = c(1, 1))
```

Eliminación de Exitus

Descartamos la variable `Exitus` antes de la construcción de modelo ya que representa un **data leakage** al estar obviamente relacionada con la progresión.

```
# Correlación implícita: TRUE→1, FALSE→0
cor(
  merged_full$Progresión == "Sí",
  merged_full$Exitus      == "Sí",
  use = "complete.obs"
)
```

```
## [1] 0.7157147
```

Imputación de missing values

```r
# Conteos absolutos y relativos de Histología
tab <- table(merged_full$Histología, useNA = "ifany")
pct <- prop.table(tab) * 100
data.frame(
  nivel = names(tab),
  conteo = as.vector(tab),
  pct = round(as.vector(pct), 2)
)
```

Histología

```
##                                       nivel conteo   pct
## 1                             Adenocarcinoma    147 64.76
## 2                    Carcinoma adenoescamoso      3  1.32
## 3                         Carcinoma escamoso     56 24.67
## 4 Carcinoma neuroendocrino de célula grande      4  1.76
## 5                     Carcinoma sarcomatoide      1  0.44
## 6                                       <NA>     16  7.05
```

Los NAs serán imputados a una clase nueva llamada “Desconocido”.

```r
merged_full <- merged_full %>%
  mutate(
    Histología = as.factor(Histología),                              # asegurar factor
    Histología = fct_explicit_na(Histología, na_level = "Desconocido")  # NA → "Desconocido"
  )
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `Histología = fct_explicit_na(Histología, na_level =
##   "Desconocido")`.
## Caused by warning:
## ! `fct_explicit_na()` was deprecated in forcats 1.0.0.
## i Please use `fct_na_value_to_level()` instead.
```

```r
# Verifica
merged_full %>%
  count(Histología) %>%
  mutate(pct = round(n/sum(n)*100,2))
```

```
## # A tibble: 6 x 3
##   Histología                                n   pct
##   <fct>                                 <int> <dbl>
## 1 Adenocarcinoma                          147 64.8
## 2 Carcinoma adenoescamoso                   3  1.32
## 3 Carcinoma escamoso                       56 24.7
## 4 Carcinoma neuroendocrino de célula grande  4  1.76
## 5 Carcinoma sarcomatoide                    1  0.44
## 6 Desconocido                              16  7.05
```

Tipo.de.IO.Cat..0.IO..1.ChIO.

```r
# Frecuencias y % de missing de Tipo.de.IO.Cat..0.IO..1.ChIO.
tab_io <- table(merged_full$Tipo.de.IO.Cat..0.IO..1.ChIO., useNA = "ifany")
pct_io <- prop.table(tab_io) * 100
data.frame(
```

```
  nivel  = names(tab_io),
  conteo = as.vector(tab_io),
  pct    = round(as.vector(pct_io), 2)
)
```

```
##          nivel conteo   pct
## 1         ChIO     70 30.84
## 2           IO    143 63.00
## 3 IO combinado     10  4.41
## 4         <NA>      4  1.76
```

De nuevo, consideramos los NAs como una nueva clase llamada `Desconocido`:

```
merged_full <- merged_full %>%
  mutate(
    Tipo.de.IO.Cat..0.IO..1.ChIO. = as.factor(Tipo.de.IO.Cat..0.IO..1.ChIO.),        # renombra para có
    Tipo.de.IO.Cat..0.IO..1.ChIO. = fct_explicit_na(Tipo.de.IO.Cat..0.IO..1.ChIO., na_level = "Desconoci
  )

# Verifica frecuencias
merged_full %>%
  count(Tipo.de.IO.Cat..0.IO..1.ChIO.) %>%
  mutate(pct = round(n / sum(n) * 100, 2))
```

```
## # A tibble: 4 x 3
##   Tipo.de.IO.Cat..0.IO..1.ChIO.     n   pct
##   <fct>                         <int> <dbl>
## 1 ChIO                             70 30.8
## 2 IO                              143 63
## 3 IO combinado                     10  4.41
## 4 Desconocido                       4  1.76
```

Tiempo.hasta.inicio.IO.metas

```
# Conteo de NA y % missing
n_missing <- sum(is.na(merged_full$Tiempo.hasta.inicio.IO.metas))
pct_missing <- n_missing / nrow(merged_full) * 100

# Estadísticos básicos
stats <- summary(merged_full$Tiempo.hasta.inicio.IO.metas)

# Imprime todo junto
list(
  n_missing   = n_missing,
  pct_missing = round(pct_missing, 2),
  stats       = stats
)
```

```
## $n_missing
## [1] 14
##
## $pct_missing
## [1] 6.17
##
## $stats
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##     0.0    22.0    43.0   222.5   203.0  3116.0      14
```

Para Tiempo.hasta.inicio.IO.metas, con un 5.7 % de missing y una distribución muy sesgada (media 229 vs mediana 43, con un outlier en 3116), lo más sencillo y robusto es imputar por la mediana.

PD.L1

```
summary(merged_full$PD.L1)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   10.00   35.17   80.00  100.00      12
```

```
# 1) Número y % de NA en PD.L1
n_missing_pd <- sum(is.na(merged_full$PD.L1))
pct_missing_pd <- round(n_missing_pd / nrow(merged_full) * 100, 2)

# 2) Estadísticos básicos de PD.L1
stats_pd <- summary(merged_full$PD.L1)

# 3) Mostrar todo junto
list(
  n_missing   = n_missing_pd,
  pct_missing = pct_missing_pd,
  stats       = stats_pd
)
```

```
## $n_missing
## [1] 12
##
## $pct_missing
## [1] 5.29
##
## $stats
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    0.00   10.00   35.17   80.00  100.00      12
```

La imputación múltiple mediante MICE (Multiple Imputation by Chained Equations) es una técnica avanzada que genera varias estimaciones plausibles para cada dato faltante, basándose en modelos iterativos que aprovechan la información de todas las variables disponibles. En cada "cadena" de ecuaciones, se ajusta un modelo de predicción para una variable con valores perdidos (aquí, PD.L1) usando como covariables las restantes (estadio al diagnóstico, histología, edad, ECOG, tumor previo), se imputan sus valores faltantes, y a continuación se procede con la siguiente variable con NA. Este proceso se repite hasta converger, produciendo m conjuntos completos de datos que reflejan la incertidumbre inherente al relleno.

Elegimos en concreto Predictive Mean Matching (PMM) dentro de MICE porque garantiza que los valores imputados de PD.L1 sean siempre observaciones reales tomadas del conjunto original (evitando valores fuera de rango o suavizados artificialmente) y conserva la asimetría y cola larga de su distribución. Al generar varias imputaciones y luego combinar resultados, este enfoque reduce el sesgo que introduciría una imputación única (como la mediana global) y mejora la validez estadística de análisis posteriores, asegurando que los pocos NAs de PD.L1 se rellenen de manera coherente con las relaciones multivariables de nuestro cohort.

```
library(mice)
```

```
##
## Adjuntando el paquete: 'mice'

## The following object is masked from 'package:stats':
```

```
##
##     filter
```

```
## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```
# 1) Seleccionamos sólo las variables necesarias para la imputación
vars_imp <- c(
  "PD.L1",
  "Estadio.al.diagnóstico",
  "Histología",
  "Edad.al.diagnóstico",
  "Escala.ECOG",
  "Tumor.previo"
)

# 2) Creamos el mids object, imputando PD.L1 con PMM
imp <- mice(
  merged_full[ , vars_imp],
  method = "pmm",
  m      = 5,
  seed   = 123
)
```

```
##
##  iter imp variable
##    1   1  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    1   2  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    1   3  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    1   4  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    1   5  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    2   1  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    2   2  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    2   3  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    2   4  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    2   5  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    3   1  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    3   2  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    3   3  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    3   4  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    3   5  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    4   1  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    4   2  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    4   3  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    4   4  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    4   5  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    5   1  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    5   2  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    5   3  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    5   4  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
##    5   5  PD.L1  Estadio.al.diagnóstico  Edad.al.diagnóstico  Escala.ECOG  Tumor.previo
```

```
## Warning: Number of logged events: 25
```

```r
# 3) Extraemos uno de los datasets completos (aquí el primero)
imp_complete <- complete(imp, 1)

# 4) Sobrescribimos sólo los NA de PD.L1 en el data.frame original
na_idx <- is.na(merged_full$PD.L1)
merged_full$PD.L1[na_idx] <- imp_complete$PD.L1[na_idx]

# 5) Verificación: ya no hay NA en PD.L1
sum(is.na(merged_full$PD.L1))  # debe dar 0
```

```
## [1] 0
```

```r
# 1) Número y % de NA en PD.L1
n_missing_pd <- sum(is.na(merged_full$PD.L1))
pct_missing_pd <- round(n_missing_pd / nrow(merged_full) * 100, 2)

# 2) Estadísticos básicos de PD.L1
stats_pd <- summary(merged_full$PD.L1)

# 3) Mostrar todo junto
list(
  n_missing   = n_missing_pd,
  pct_missing = pct_missing_pd,
  stats       = stats_pd
)
```

```
## $n_missing
## [1] 0
##
## $pct_missing
## [1] 0
##
## $stats
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    0.00   10.00   36.12   80.00  100.00
```

Fecha.de.diagnóstico.de.enfermedad.metastática

```r
# Número y % de NA
n_miss <- sum(is.na(merged_full$Fecha.de.diagnóstico.de.enfermedad.metastática))
pct_miss <- round(n_miss / nrow(merged_full) * 100, 2)

# Rango y "fecha central"
fechas <- merged_full$Fecha.de.diagnóstico.de.enfermedad.metastática
fecha_min <- min(fechas, na.rm = TRUE)
fecha_max <- max(fechas, na.rm = TRUE)
fecha_med <- median(fechas, na.rm = TRUE)

list(
  n_missing   = n_miss,
  pct_missing = pct_miss,
  fecha_min   = fecha_min,
  fecha_med   = fecha_med,
  fecha_max   = fecha_max
```

```
)
```

```
## $n_missing
## [1] 12
##
## $pct_missing
## [1] 5.29
##
## $fecha_min
## [1] "2013-07-01"
##
## $fecha_med
## [1] "2021-05-20"
##
## $fecha_max
## [1] "2023-07-06"
```

Imputamos la fecha de diagnóstico de enfermedad metastásica —que presenta un 4,8 % de valores faltantes— de la siguiente manera: primero creamos un indicador binario (Metastasis_no_diag) que marca con TRUE los casos en que no hay fecha (es decir, nunca se diagnosticó metástasis o no se registró), preservando así esa información clínica. A continuación, rellenamos los NA con la mediana de las fechas observadas (20 de mayo de 2021), un valor real dentro del rango (desde el 1 de julio de 2013 hasta el 6 de julio de 2023) que evita sesgar la distribución hacia extremos tempranos o tardíos y permite al modelo tratar homogéneamente todas las observaciones.

```
# 1) Fecha mediana ya calculada:
fecha_mediana <- as_date("2021-05-20")

merged_full <- merged_full %>%
  # 2) Indicador de no metastásico
  mutate(
    Metastasis_no_diag = is.na(Fecha.de.diagnóstico.de.enfermedad.metastática),
    # 3) Imputar NA con la mediana
    Fecha.de.diagnóstico.de.enfermedad.metastática =
      if_else(
        Metastasis_no_diag,
        fecha_mediana,
        Fecha.de.diagnóstico.de.enfermedad.metastática
      )
  )

# Verificación
merged_full %>%
  summarize(
    n_na    = sum(is.na(Fecha.de.diagnóstico.de.enfermedad.metastática)),
    n_other = sum(Metastasis_no_diag),
    min     = min(Fecha.de.diagnóstico.de.enfermedad.metastática),
    median  = median(Fecha.de.diagnóstico.de.enfermedad.metastática),
    max     = max(Fecha.de.diagnóstico.de.enfermedad.metastática)
  )
```

```
## # A tibble: 1 x 5
##    n_na n_other min        median     max
##   <int>   <int> <date>     <date>     <date>
## 1     0      12 2013-07-01 2021-05-20 2023-07-06
```

Tiempo.hasta.inicio.IO

```
# 1) Número y % de NA en Tiempo.hasta.inicio.IO
n_missing_ti <- sum(is.na(merged_full$Tiempo.hasta.inicio.IO))
pct_missing_ti <- round(n_missing_ti / nrow(merged_full) * 100, 2)

# 2) Estadísticos básicos
stats_ti <- summary(merged_full$Tiempo.hasta.inicio.IO)

# 3) Mostrar todo junto
list(
  n_missing   = n_missing_ti,
  pct_missing = pct_missing_ti,
  stats       = stats_ti
)
```

```
## $n_missing
## [1] 9
##
## $pct_missing
## [1] 3.96
##
## $stats
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##     2.0    28.0    77.0   373.4   474.0  4488.0       9
```

Imputamos los 9 valores faltantes de la variable Tiempo.hasta.inicio.IO (3,95 % de la muestra) usando la mediana de la distribución (78 días), de modo que cada NA se reemplaza por este valor central resistente a outliers y se mantiene la coherencia numérica de la variable para el modelado.

```
# Calcular la mediana ignorando los NA
mediana_tiempoIO <- median(merged_full$Tiempo.hasta.inicio.IO, na.rm = TRUE)

# Imputar directamente los NA por la mediana
merged_full$Tiempo.hasta.inicio.IO[is.na(merged_full$Tiempo.hasta.inicio.IO)] <- mediana_tiempoIO

# Verificación
summary(merged_full$Tiempo.hasta.inicio.IO)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     2.0    28.5    77.0   361.7   445.0  4488.0
```

```
sum(is.na(merged_full$Tiempo.hasta.inicio.IO))
```

```
## [1] 0
```

Follow.Up.(d)

```
# 1) Número y % de NA en Follow up
n_missing_fu <- sum(is.na(merged_full$Follow.Up..d.))
pct_missing_fu <- round(n_missing_fu / nrow(merged_full) * 100, 2)

# 2) Estadísticos básicos
stats_fu <- summary(merged_full$Follow.Up..d.)

# 3) Mostrar resultados
```

```r
list(
  n_missing   = n_missing_fu,
  pct_missing = pct_missing_fu,
  stats       = stats_fu
)
```

```
## $n_missing
## [1] 8
##
## $pct_missing
## [1] 3.52
##
## $stats
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##     2.0   165.5   351.0   491.7   715.0  2058.0       8
```

Debido a que esta variable se va a utilizar en las curvas Kaplan - Meier, donde la imputación puede ser problemática debido a que el dato de Follow up es clave, vamos a dejar los `NA` y eliminar posteriormente esos registros. En caso de que no queramos eliminar esos registros, imputamos los 8 valores faltantes de Follow.Up..d. (3,51 % de la muestra) por la mediana de la variable (350 días), de manera que cada NA se reemplaza por este valor central resistente a los outliers y se preserva la coherencia del rango de días de seguimiento para el modelado.

PFS..d.

```r
# 1) Número y % de NA en PFS..d.
n_missing_pfs <- sum(is.na(merged_full$PFS..d.))
pct_missing_pfs <- round(n_missing_pfs / nrow(merged_full) * 100, 2)

# 2) Estadísticos básicos
stats_pfs <- summary(merged_full$PFS..d.)

# 3) Mostrar resultados
list(
  n_missing   = n_missing_pfs,
  pct_missing = pct_missing_pfs,
  stats       = stats_pfs
)
```

```
## $n_missing
## [1] 8
##
## $pct_missing
## [1] 3.52
##
## $stats
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##     1.0    81.5   184.0   365.0   524.5  2058.0       8
```

Antes de imputar, vamos a ver si esos valores de missing corresponden a valores `No` de la variable `Progresión`. En la variable de PFS respectiva a los días vemos que 3 de los valores faltantes efectivamente tienen progresión, y 5 no. No es plausible la teoría de que los NAs son correspondientes a la ausencia de progresión.

```r
merged_full %>%
  filter(is.na(PFS..d.)) %>%
```

```
  count(Progresión)
```

```
## # A tibble: 2 x 2
##   Progresión     n
##   <fct>      <int>
## 1 No             5
## 2 Sí             3
```

Además, tenemos 48 no progresados. Para que esta hipótesis tuviera sentido tendríamos que tener 8 valores `No` en la variable de progresión.

```
merged_full %>% count(Progresión)
```

```
## # A tibble: 2 x 2
##   Progresión     n
##   <fct>      <int>
## 1 No            48
## 2 Sí           179
```

Imputamos los 8 valores faltantes de PFS..d. (3,51 % de la muestra) por la mediana de la variable (184,5 días), de manera que cada NA se reemplaza por este valor central resistente a los outliers y se preserva la coherencia del rango de supervivencia libre de progresión para el modelado.

```
# Calcular mediana sin NA
mediana_pfs <- median(merged_full$PFS..d., na.rm = TRUE)

# Imputar NA por la mediana
merged_full$PFS..d.[is.na(merged_full$PFS..d.)] <- mediana_pfs

# Verificación
summary(merged_full$PFS..d.)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     1.0    84.0   184.0   358.6   497.5  2058.0
```

```
sum(is.na(merged_full$PFS..d.))
```

```
## [1] 0
```

PFS..m.

```
# 1) Número y % de NA en PFS..m.
n_missing_pfsm <- sum(is.na(merged_full$PFS..m.))
pct_missing_pfsm <- round(n_missing_pfsm / nrow(merged_full) * 100, 2)

# 2) Estadísticos básicos
stats_pfsm <- summary(merged_full$PFS..m.)

# 3) Mostrar todo junto
list(
  n_missing   = n_missing_pfsm,
  pct_missing = pct_missing_pfsm,
  stats       = stats_pfsm
)
```

```
## $n_missing
## [1] 8
```

```
##
## $pct_missing
## [1] 3.52
##
## $stats
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.     NA's
##  0.03333  2.71667  6.13333 12.16575 17.48333 68.60000        8
```

Para PFS..m. (3,5 % NA, distribución muy sesgada con outliers), lo más sencillo y robusto es imputar por la mediana (6,15 meses).

```
# 1) Calcular mediana ignorando NA
mediana_pfsm <- median(merged_full$PFS..m., na.rm = TRUE)

# 2) Imputar los NA con la mediana
merged_full$PFS..m.[is.na(merged_full$PFS..m.)] <- mediana_pfsm

# 3) Verificación
summary(merged_full$PFS..m.)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##  0.03333  2.80000  6.13333 11.95316 16.58333 68.60000
```

```
sum(is.na(merged_full$PFS..m.))
```

```
## [1] 0
```

Progresión a x meses

Antes de la eliminación de PFS, creamos dos variables derivadas de ellas.

```
# Crear variables de progresión a 3, 6 y 12 meses
merged_full <- merged_full %>%
  mutate(
    Prog.3.meses  = if_else(PFS..m. <=  3, "Sí", "No"),
    Prog.6.meses  = if_else(PFS..m. <=  6, "Sí", "No"),
    Prog.12.meses = if_else(PFS..m. <= 12, "Sí", "No"),
    Prog.3.meses  = factor(Prog.3.meses,  levels = c("No", "Sí")),
    Prog.6.meses  = factor(Prog.6.meses,  levels = c("No", "Sí")),
    Prog.12.meses = factor(Prog.12.meses, levels = c("No", "Sí"))
  )

# Resumen numérico con na.rm = TRUE, ahora incluyendo 3 meses
merged_full %>%
  summarise(
    n3    = sum(Prog.3.meses == "Sí", na.rm = TRUE),
    pct3  = round(mean(Prog.3.meses == "Sí", na.rm = TRUE) * 100, 2),
    n6    = sum(Prog.6.meses == "Sí", na.rm = TRUE),
    pct6  = round(mean(Prog.6.meses == "Sí", na.rm = TRUE) * 100, 2),
    n12   = sum(Prog.12.meses == "Sí", na.rm = TRUE),
    pct12 = round(mean(Prog.12.meses == "Sí", na.rm = TRUE) * 100, 2)
  )
```

```
## # A tibble: 1 x 6
##      n3  pct3    n6  pct6   n12 pct12
##   <int> <dbl> <int> <dbl> <int> <dbl>
## 1    60  26.4   106  46.7   162  71.4
```

```r
# Desglose de frecuencias para comprobar cada variable
freq_3m  <- merged_full %>% count(Prog.3.meses)  %>% mutate(pct = round(n/sum(n)*100,2))
freq_6m  <- merged_full %>% count(Prog.6.meses)  %>% mutate(pct = round(n/sum(n)*100,2))
freq_12m <- merged_full %>% count(Prog.12.meses) %>% mutate(pct = round(n/sum(n)*100,2))

print(freq_3m)
```

```
## # A tibble: 2 x 3
##   Prog.3.meses     n   pct
##   <fct>        <int> <dbl>
## 1 No             167  73.6
## 2 Sí              60  26.4
```

```r
print(freq_6m)
```

```
## # A tibble: 2 x 3
##   Prog.6.meses     n   pct
##   <fct>        <int> <dbl>
## 1 No             121  53.3
## 2 Sí             106  46.7
```

```r
print(freq_12m)
```

```
## # A tibble: 2 x 3
##   Prog.12.meses     n   pct
##   <fct>         <int> <dbl>
## 1 No               65  28.6
## 2 Sí              162  71.4
```

Eliminación de PFS (days and months)

Debido a que no es posible aclarar el funcionamiento de la variable se propone su eliminación.

Escala.ECOG

```r
# Frecuencias absolutas y relativas de Escala.ECOG
tab_ecog <- table(merged_full$Escala.ECOG, useNA = "ifany")
pct_ecog <- prop.table(tab_ecog) * 100
data.frame(
  nivel  = names(tab_ecog),
  conteo = as.vector(tab_ecog),
  pct    = round(as.vector(pct_ecog), 2)
)
```

```
##   nivel conteo   pct
## 1     0     65 28.63
## 2     1    125 55.07
## 3     2     29 12.78
## 4  <NA>      8  3.52
```

Para Escala.ECOG (variable ordinal con niveles 0, 1, 2 y un 3,5 % de NA) lo más sencillo y apropiado es imputar los NA por la moda (nivel más frecuente), que es 1 (54,8 % de los casos). Con tan pocos missing evitamos complicar con métodos multivariantes y preservamos el orden natural de la variable.

```r
# 1) Calcular la moda
tab_ecog <- table(merged_full$Escala.ECOG, useNA = "no")
moda_ecog <- as.numeric(names(tab_ecog)[which.max(tab_ecog)])
```

```r
# 2) Imputar los NA con la moda
merged_full$Escala.ECOG[is.na(merged_full$Escala.ECOG)] <- moda_ecog

# 3) Verificación
table(merged_full$Escala.ECOG, useNA = "ifany")
```

```
##
##   0   1   2
##  65 133  29
```

```r
prop.table(table(merged_full$Escala.ECOG, useNA = "ifany")) * 100
```

```
##
##        0        1        2
## 28.63436 58.59031 12.77533
```

Fecha.de.inicio.IO

```r
# 1) Número y % de NA en Fecha.de.inicio.IO
n_missing_io <- sum(is.na(merged_full$Fecha.de.inicio.IO))
pct_missing_io <- round(n_missing_io / nrow(merged_full) * 100, 2)

# 2) Fecha mínima, mediana y máxima
fechas_io <- merged_full$Fecha.de.inicio.IO
fecha_min_io <- min(fechas_io, na.rm = TRUE)
fecha_med_io <- median(fechas_io, na.rm = TRUE)
fecha_max_io <- max(fechas_io, na.rm = TRUE)

# 3) Mostrar resultados
list(
  n_missing   = n_missing_io,
  pct_missing = pct_missing_io,
  fecha_min   = fecha_min_io,
  fecha_med   = fecha_med_io,
  fecha_max   = fecha_max_io
)
```

```
## $n_missing
## [1] 7
##
## $pct_missing
## [1] 3.08
##
## $fecha_min
## [1] "2018-06-19"
##
## $fecha_med
## [1] "2021-08-08"
##
## $fecha_max
## [1] "2023-07-24"
```

Hemos imputado las 7 fechas faltantes de inicio de inmunoterapia sustituyéndolas por la mediana observada (13-08-2021) y, además, creado un indicador binario (IO_no_inicio) que marca qué pacientes carecían originalmente de fecha. De este modo conservamos la señal de "no registrado" y garantizamos que la variable

resultante sea numéricamente coherente para el modelo.

```
# Fecha mediana
fecha_mediana_io <- as_date("2021-08-13")

merged_full <- merged_full %>%
  mutate(
    IO_no_inicio = is.na(Fecha.de.inicio.IO),
    Fecha.de.inicio.IO = if_else(
      IO_no_inicio,
      fecha_mediana_io,
      Fecha.de.inicio.IO
    )
  )

# Verificación
merged_full %>%
  summarize(
    n_na    = sum(is.na(Fecha.de.inicio.IO)),
    n_flag  = sum(IO_no_inicio),
    min     = min(Fecha.de.inicio.IO),
    median  = median(Fecha.de.inicio.IO),
    max     = max(Fecha.de.inicio.IO)
  )
```

```
## # A tibble: 1 x 5
##    n_na n_flag min        median     max
##   <int>  <int> <date>     <date>     <date>
## 1     0      7 2018-06-19 2021-08-13 2023-07-24
```

Estadio.al.diagnóstico

```
# Frecuencias y % de missing de Estadio.al.diagnóstico
tab_estadio <- table(merged_full$Estadio.al.diagnóstico, useNA = "ifany")
pct_estadio <- prop.table(tab_estadio) * 100
data.frame(
  estadio = names(tab_estadio),
  conteo  = as.vector(tab_estadio),
  pct     = round(as.vector(pct_estadio), 2)
)
```

```
##    estadio conteo   pct
## 1       IA      7  3.08
## 2       IB      2  0.88
## 3      IIA      4  1.76
## 4      IIB      5  2.20
## 5     IIIA     18  7.93
## 6     IIIB     12  5.29
## 7     IIIC     16  7.05
## 8      IVA     79 34.80
## 9      IVB     77 33.92
## 10    <NA>      7  3.08
```

Para Estadio.al.diagnóstico (7 NA 3 %), al ser una variable ordinal con subniveles y con “IVA” como categoría más frecuente (34,7 %), lo más sencillo y consistente es imputar los NA por la moda (“IVA”). De este modo mantenemos la distribución original y respetamos el orden clínico de los estadios.

```r
# Calcular la moda de Estadio.al.diagnóstico (excluyendo NA)
tab <- table(merged_full$Estadio.al.diagnóstico, useNA = "no")
moda_estadio <- names(tab)[which.max(tab)]

# Imputar los NA con la moda
merged_full$Estadio.al.diagnóstico[is.na(merged_full$Estadio.al.diagnóstico)] <- moda_estadio

# Verificación
table(merged_full$Estadio.al.diagnóstico, useNA = "ifany")
```

```
##
##   IA   IB  IIA  IIB IIIA IIIB IIIC  IVA  IVB
##    7    2    4    5   18   12   16   86   77
```

```r
prop.table(table(merged_full$Estadio.al.diagnóstico, useNA = "ifany")) * 100
```

```
##
##         IA         IB        IIA        IIB       IIIA       IIIB       IIIC
##  3.0837004  0.8810573  1.7621145  2.2026432  7.9295154  5.2863436  7.0484581
##        IVA        IVB
## 37.8854626 33.9207048
```

Fecha.de.inicio.IO..metastáticos.

```r
# 1) Número y % de NA
n_missing_metaIO <- sum(is.na(merged_full$Fecha.de.inicio.IO..metastáticos.))
pct_missing_metaIO <- round(n_missing_metaIO / nrow(merged_full) * 100, 2)

# 2) Fecha mínima, mediana y máxima
fechas_metaIO <- merged_full$Fecha.de.inicio.IO..metastáticos.
fecha_min_metaIO <- min(fechas_metaIO, na.rm = TRUE)
fecha_med_metaIO <- median(fechas_metaIO, na.rm = TRUE)
fecha_max_metaIO <- max(fechas_metaIO, na.rm = TRUE)

# 3) Mostrar resultados
list(
  n_missing   = n_missing_metaIO,
  pct_missing = pct_missing_metaIO,
  fecha_min   = fecha_min_metaIO,
  fecha_med   = fecha_med_metaIO,
  fecha_max   = fecha_max_metaIO
)
```

```
## $n_missing
## [1] 7
##
## $pct_missing
## [1] 3.08
##
## $fecha_min
## [1] "2018-06-19"
##
## $fecha_med
## [1] "2021-08-14"
##
```

```
## $fecha_max
## [1] "2023-07-24"
```

Imputamos las 7 fechas faltantes de inicio de la segunda línea de inmunoterapia en situación metastásica sustituyéndolas por la mediana observada (16-08-2021) y, además, añadimos un indicador binario (MetaIO_no_inicio) que señala los pacientes sin fecha registrada. Así conservamos la información de "no iniciado" y aseguramos que la variable permanezca en un formato de fecha coherente para el modelo.

```
# Mediana de fecha
fecha_mediana_metaIO <- as_date("2021-08-16")

merged_full <- merged_full %>%
  mutate(
    MetaIO_no_inicio = is.na(Fecha.de.inicio.IO..metastáticos.),
    Fecha.de.inicio.IO..metastáticos. = if_else(
      MetaIO_no_inicio,
      fecha_mediana_metaIO,
      Fecha.de.inicio.IO..metastáticos.
    )
  )

# Verificación
merged_full %>%
  summarize(
    n_na    = sum(is.na(Fecha.de.inicio.IO..metastáticos.)),
    n_flag  = sum(MetaIO_no_inicio),
    min     = min(Fecha.de.inicio.IO..metastáticos.),
    median  = median(Fecha.de.inicio.IO..metastáticos.),
    max     = max(Fecha.de.inicio.IO..metastáticos.)
  )
```

```
## # A tibble: 1 x 5
##    n_na n_flag min        median     max
##   <int>  <int> <date>     <date>     <date>
## 1     0      7 2018-06-19 2021-08-16 2023-07-24
```

TIPO.IO..metastáticos.

```
# Frecuencias absolutas y relativas de TIPO.IO..metastáticos.
tab_tipo_meta <- table(merged_full$TIPO.IO..metastáticos., useNA = "ifany")
pct_tipo_meta <- prop.table(tab_tipo_meta) * 100
data.frame(
  nivel  = names(tab_tipo_meta),
  conteo = as.vector(tab_tipo_meta),
  pct    = round(as.vector(pct_tipo_meta), 2)
)
```

```
##                             nivel conteo   pct
## 1                     Inmunoterapia    143 63.00
## 2 Inmunoterapia + antiangiogénico      7  3.08
## 3   Inmunoterapia + quimioterapia     70 30.84
## 4                             <NA>      7  3.08
```

Para TIPO.IO..metastáticos., al ser categórica y tener sólo un 3 % de NA, lo más limpio es convertir esos NA en un nivel "Desconocido" sin alterar las tres categorías reales ("Inmunoterapia", "Inmunoterapia + antiangiogénico", "Inmunoterapia + quimioterapia"), de modo que el modelo capte que había un valor no

registrado.

```
merged_full <- merged_full %>%
  mutate(
    TIPO.IO..metastáticos. = as.factor(TIPO.IO..metastáticos.),                    # asegurar facto
    TIPO.IO..metastáticos. = fct_explicit_na(TIPO.IO..metastáticos., na_level = "Desconocido")    # NA
  )

# Verificación
merged_full %>%
  count(TIPO.IO..metastáticos.) %>%
  mutate(pct = round(n / sum(n) * 100, 2))
```

```
## # A tibble: 4 x 3
##   TIPO.IO..metastáticos.              n   pct
##   <fct>                           <int> <dbl>
## 1 Inmunoterapia                     143 63
## 2 Inmunoterapia + antiangiogénico     7  3.08
## 3 Inmunoterapia + quimioterapia      70 30.8
## 4 Desconocido                         7  3.08
```

Diana.IO

```
# Frecuencias absolutas y relativas de Diana.IO
tab_diana <- table(merged_full$Diana.IO, useNA = "ifany")
pct_diana <- prop.table(tab_diana) * 100
data.frame(
  nivel  = names(tab_diana),
  conteo = as.vector(tab_diana),
  pct    = round(as.vector(pct_diana), 2)
)
```

```
##        nivel conteo   pct
## 1       Otro      5  2.20
## 2       PD-1    162 71.37
## 3 PD-1,PD-L1      2  0.88
## 4      PD-L1     53 23.35
## 5 PD-L1,Otro      1  0.44
## 6       <NA>      4  1.76
```

Imputamos los 4 valores faltantes de la variable Diana.IO (1,75 % del total) añadiendo un nivel “Desconocido” y manteniendo intactas las categorías existentes (“PD-1”, “PD-L1”, “Otro” y las combinaciones raras), de modo que el modelo pueda distinguir los casos no registrados sin perder ninguna información original.

```
merged_full <- merged_full %>%
  mutate(
    Diana.IO = as.factor(Diana.IO),                                # asegurar factor
    Diana.IO = fct_explicit_na(Diana.IO, na_level = "Desconocido")  # NA → "Desconocido"
  )

# Verificación
merged_full %>%
  count(Diana.IO) %>%
  mutate(pct = round(n / sum(n) * 100, 2))
```

```
## # A tibble: 6 x 3
```

```
##   Diana.IO        n   pct
##   <fct>       <int> <dbl>
## 1 Otro            5  2.2
## 2 PD-1          162 71.4
## 3 PD-1,PD-L1      2  0.88
## 4 PD-L1          53 23.4
## 5 PD-L1,Otro      1  0.44
## 6 Desconocido     4  1.76
```

Nº.de.líneas.previas

```
# 1) Número y % de NA en Nº.de.líneas.previas
n_missing_lines <- sum(is.na(merged_full$Nº.de.líneas.previas))
pct_missing_lines <- round(n_missing_lines / nrow(merged_full) * 100, 2)

# 2) Estadísticos básicos
stats_lines <- summary(merged_full$Nº.de.líneas.previas)

# 3) Mostrar todo junto
list(
  n_missing   = n_missing_lines,
  pct_missing = pct_missing_lines,
  stats       = stats_lines
)
```

```
## $n_missing
## [1] 4
##
## $pct_missing
## [1] 1.76
##
## $stats
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  0.0000  0.0000  0.0000  0.3991  1.0000  3.0000       4
```

Imputamos los 4 valores faltantes de Nº.de.líneas.previas (1,75 % de la muestra) sustituyéndolos por la mediana (0 líneas), de modo que las observaciones sin dato se consideren como sin tratamientos previos y no se distorsione la concentración de valores bajos propia de esta variable.

```
# Calcular mediana ignorando NA
mediana_lines <- median(merged_full$Nº.de.líneas.previas, na.rm = TRUE)

# Imputar los NA con la mediana
merged_full$Nº.de.líneas.previas[is.na(merged_full$Nº.de.líneas.previas)] <- mediana_lines

# Verificación
summary(merged_full$Nº.de.líneas.previas)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  0.0000  0.3921  1.0000  3.0000
```

```
sum(is.na(merged_full$Nº.de.líneas.previas))
```

```
## [1] 0
```

LíneasPrevias_Cat

Este bloque genera la variable categórica ´LíneasPrevias_Cat´ a partir de `Nº.de.líneas.previas`, agrupando los valores en tres niveles: "0" para pacientes sin líneas previas, "1" para quienes tienen una línea, y "2 o más" para los que cuentan con dos o más. Finalmente, la factoriza con ese orden para asegurar consistencia en análisis y visualizaciones.

```
merged_full <- merged_full %>%
  mutate(
    LíneasPrevias_Cat = case_when(
      `Nº.de.líneas.previas` == 0       ~ "0",
      `Nº.de.líneas.previas` == 1       ~ "1",
      `Nº.de.líneas.previas` >= 2       ~ "2 o más",
      TRUE                              ~ NA_character_
    ),
    LíneasPrevias_Cat = factor(
      LíneasPrevias_Cat,
      levels = c("0", "1", "2 o más")
    )
  )
```

IO.Tipo.General

```
# Frecuencias absolutas y relativas de IO.Tipo.General
tab_io_gen <- table(merged_full$IO.Tipo.General, useNA = "ifany")
pct_io_gen <- prop.table(tab_io_gen) * 100
data.frame(
  nivel  = names(tab_io_gen),
  conteo = as.vector(tab_io_gen),
  pct    = round(as.vector(pct_io_gen), 2)
)
```

```
##                             nivel conteo   pct
## 1                     Inmunoterapia    143 63.00
## 2 Inmunoterapia + antiangiogénico      7  3.08
## 3   Inmunoterapia + quimioterapia     70 30.84
## 4             IO adyuvante no met      3  1.32
## 5                            <NA>      4  1.76
```

Imputamos los 4 valores faltantes de IO.Tipo.General (1,75 % del total) añadiendo un nivel "Desconocido" y conservando intactas las categorías existentes ("Inmunoterapia", "Inmunoterapia + antiangiogénico", "Inmunoterapia + quimioterapia" e "IO adyuvante no met"), de modo que el modelo pueda distinguir los casos no registrados sin perder información original.

```
merged_full <- merged_full %>%
  mutate(
    IO.Tipo.General = as.factor(IO.Tipo.General),                          # asegurar factor
    IO.Tipo.General = fct_explicit_na(IO.Tipo.General, na_level = "Desconocido")  # NA → "Desconocido"
  )

# Verificación
merged_full %>%
  count(IO.Tipo.General) %>%
  mutate(pct = round(n / sum(n) * 100, 2))
```

```
## # A tibble: 5 x 3
```

```
##   IO.Tipo.General                     n   pct
##   <fct>                           <int> <dbl>
## 1 Inmunoterapia                     143 63
## 2 Inmunoterapia + antiangiogénico     7  3.08
## 3 Inmunoterapia + quimioterapia      70 30.8
## 4 IO adyuvante no met                 3  1.32
## 5 Desconocido                         4  1.76
```

Presencia.linea.Previa

```
# Frecuencias absolutas y relativas de Presencia.linea.Previa
tab_previa <- table(merged_full$Presencia.linea.Previa, useNA = "ifany")
pct_previa <- prop.table(tab_previa) * 100
data.frame(
  nivel  = names(tab_previa),
  conteo = as.vector(tab_previa),
  pct    = round(as.vector(pct_previa), 2)
)
```

```
##   nivel conteo   pct
## 1    No    147 64.76
## 2    Si     76 33.48
## 3  <NA>      4  1.76
```

Para la variable Presencia.linea.Previa (1,75 % de valores faltantes y dos categorías “No” (64,5 %) y “Si” (33,8 %)), imputamos los NA por la moda (“No”), de forma que las pocas observaciones sin dato pasen al nivel mayoritario y no se altere la distribución ni el balance de la variable para el modelo.

```
# 1) Calcular la moda (nivel más frecuente) excluyendo NA
moda_previa <- names(which.max(table(merged_full$Presencia.linea.Previa, useNA = "no")))

# 2) Imputar los NA con la moda
merged_full$Presencia.linea.Previa[is.na(merged_full$Presencia.linea.Previa)] <- moda_previa

# 3) Verificación
table(merged_full$Presencia.linea.Previa, useNA = "ifany")
```

```
##
##  No  Si
## 151  76
```

```
prop.table(table(merged_full$Presencia.linea.Previa, useNA = "ifany")) * 100
```

```
##
##       No       Si
## 66.51982 33.48018
```

Fecha.de.diagnóstico

```
# 1) Número y % de NA en Fecha.de.diagnóstico
n_missing_diag <- sum(is.na(merged_full$Fecha.de.diagnóstico))
pct_missing_diag <- round(n_missing_diag / nrow(merged_full) * 100, 2)

# 2) Rango y fecha mediana
fechas_diag <- merged_full$Fecha.de.diagnóstico
fecha_min_diag <- min(fechas_diag, na.rm = TRUE)
```

```r
fecha_med_diag <- median(fechas_diag, na.rm = TRUE)
fecha_max_diag <- max(fechas_diag, na.rm = TRUE)

# 3) Mostrar resultados
list(
  n_missing   = n_missing_diag,
  pct_missing = pct_missing_diag,
  fecha_min   = fecha_min_diag,
  fecha_med   = fecha_med_diag,
  fecha_max   = fecha_max_diag
)
```

```
## $n_missing
## [1] 2
##
## $pct_missing
## [1] 0.88
##
## $fecha_min
## [1] "2010-01-28"
##
## $fecha_med
## [1] "2021-05-06"
##
## $fecha_max
## [1] "2023-07-06"
```

Imputamos las 2 fechas faltantes de diagnóstico (0,88 % del total) reemplazándolas por la mediana observada (01-05-2021) y añadimos un indicador (Diagnostico_no_reg) para marcar los casos originalmente sin fecha, de modo que conservemos la señal de "no registrado" y mantengamos la variable en un formato de fecha coherente para el modelo.

```r
# 1) Fecha mediana observada
fecha_mediana_diag <- as_date("2021-05-01")

merged_full <- merged_full %>%
  # 2) Indicador de missing
  mutate(
    Diagnostico_no_reg = is.na(Fecha.de.diagnóstico),
    # 3) Imputar NA con la mediana
    Fecha.de.diagnóstico = if_else(
      Diagnostico_no_reg,
      fecha_mediana_diag,
      Fecha.de.diagnóstico
    )
  )

# 4) Verificación
merged_full %>%
  summarize(
    n_na    = sum(is.na(Fecha.de.diagnóstico)),
    n_flag  = sum(Diagnostico_no_reg),
    min     = min(Fecha.de.diagnóstico),
    median  = median(Fecha.de.diagnóstico),
    max     = max(Fecha.de.diagnóstico)
```

```
  )
```

```
## # A tibble: 1 x 5
##    n_na n_flag min        median     max
##   <int>  <int> <date>     <date>     <date>
## 1     0      2 2010-01-28 2021-05-01 2023-07-06
```

Tabaquismo..0.Nunca.fumador..1.Exfumador..2..Fumador.Activo.

```
tab_tabq <- table(merged_full$Tabaquismo..0.Nunca.fumador..1.Exfumador..2..Fumador.Activo., useNA = "ifa
pct_tabq <- prop.table(tab_tabq) * 100
data.frame(
  nivel  = names(tab_tabq),
  conteo = as.vector(tab_tabq),
  pct    = round(as.vector(pct_tabq), 2)
)
```

```
##       nivel conteo   pct
## 1 Exfumador    128 56.39
## 2   Fumador     85 37.44
## 3     Nunca     13  5.73
## 4      <NA>      1  0.44
```

Para Tabaquismo (0,44 % NA; niveles “Exfumador” 56,6 %, “Fumador” 37,3 %, “Nunca” 5,7 %), lo más sencillo y coherente es imputar el único NA al nivel “Nunca”, de forma que la observación sin dato pase al nivel más bajo sin alterar la distribución de los demás.

```
# Imputar el NA al nivel "Nunca"
merged_full$Tabaquismo..0.Nunca.fumador..1.Exfumador..2..Fumador.Activo.[
  is.na(merged_full$Tabaquismo..0.Nunca.fumador..1.Exfumador..2..Fumador.Activo.)
] <- "Nunca"

# Verificación
table(merged_full$Tabaquismo..0.Nunca.fumador..1.Exfumador..2..Fumador.Activo., useNA = "ifany")
```

```
##
## Exfumador   Fumador     Nunca
##       128        85        14
```

```
prop.table(table(merged_full$Tabaquismo..0.Nunca.fumador..1.Exfumador..2..Fumador.Activo., useNA = "ifa
```

```
##
## Exfumador   Fumador     Nunca
## 56.387665 37.444934  6.167401
```

Tumor.previo

```
# Frecuencias absolutas y relativas de Tumor.previo
tab_tprevio <- table(merged_full$Tumor.previo, useNA = "ifany")
pct_tprevio <- prop.table(tab_tprevio) * 100
data.frame(
  nivel  = names(tab_tprevio),
  conteo = as.vector(tab_tprevio),
  pct    = round(as.vector(pct_tprevio), 2)
)
```

```
##   nivel conteo   pct
## 1    No    191 84.14
## 2    Si     35 15.42
## 3  <NA>      1  0.44
```

Para la variable Tumor.previo, que presenta un único valor faltante (0,44 %) y dos categorías (“No” 84,2 %; “Si” 15,4 %), hemos optado por una imputación estratificada según el Estadio.al.diagnóstico. Con este enfoque, calculamos la moda de Tumor.previo dentro de cada estadio y asignamos al paciente con dato ausente el valor modal de su propio estrato (por ejemplo, en el estadio IIIA la moda fue “No”). Así preservamos la distribución de la variable en cada subgrupo y minimizamos posibles sesgos derivados de una imputación global.

```
# 1) Función para moda
mode_fun <- function(x) {
  ux <- unique(x)
  ux[which.max(tabulate(match(x, ux)))]
}

# 2) Calcular la moda de Tumor.previo por estadio
modas_por_estadio <- merged_full %>%
  filter(!is.na(Tumor.previo)) %>%
  group_by(Estadio.al.diagnóstico) %>%
  summarise(
    moda = mode_fun(Tumor.previo),
    .groups = "drop"
  )

# 3) Mostrar la tabla
print(modas_por_estadio)
```

```
## # A tibble: 9 x 2
##   Estadio.al.diagnóstico moda
##   <fct>                  <fct>
## 1 IA                     No
## 2 IB                     Si
## 3 IIA                    No
## 4 IIB                    No
## 5 IIIA                   No
## 6 IIIB                   No
## 7 IIIC                   No
## 8 IVA                    No
## 9 IVB                    No
```

```
# Función para calcular moda
mode_fun <- function(x) {
  ux <- unique(x)
  ux[which.max(tabulate(match(x, ux)))]
}

# 1) Calcular la moda de Tumor.previo por cada estadio (sin contar NAs)
modas_por_estadio <- merged_full %>%
  filter(!is.na(Tumor.previo)) %>%
  group_by(Estadio.al.diagnóstico) %>%
  summarise(moda = mode_fun(Tumor.previo), .groups = "drop")

# 2) Imputar el NA usando la moda de su estadio
```

```r
merged_full <- merged_full %>%
  left_join(modas_por_estadio, by = "Estadio.al.diagnóstico") %>%
  mutate(
    Tumor.previo = if_else(
      is.na(Tumor.previo),
      moda,               # moda de su estrato
      Tumor.previo
    )
  ) %>%
  select(-moda)

# 3) Verificación: sigue habiendo el mismo total de niveles y sin NA
table(merged_full$Tumor.previo, useNA = "ifany")
```

```
##
##  No  Si
## 192  35
```

Edad.al.diagnóstico

```r
# 1) Nº y % de NA en Edad.al.diagnóstico
n_missing_age <- sum(is.na(merged_full$Edad.al.diagnóstico))
pct_missing_age <- round(n_missing_age / nrow(merged_full) * 100, 2)

# 2) Estadísticos básicos
stats_age <- summary(merged_full$Edad.al.diagnóstico)

# 3) Imprimir todo junto
list(
  n_missing   = n_missing_age,
  pct_missing = pct_missing_age,
  stats       = stats_age
)
```

```
## $n_missing
## [1] 1
##
## $pct_missing
## [1] 0.44
##
## $stats
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   29.00   60.25   66.00   65.76   72.00   87.00       1
```

Imputamos el único valor faltante de Edad.al.diagnóstico (0,44 % de la muestra) sustituyéndolo por la mediana observada (66 años), garantizando que la variable mantenga su centro real sin verse afectada por posibles outliers ni alterar la distribución global.

```r
# Calcular mediana ignorando el NA
mediana_age <- median(merged_full$Edad.al.diagnóstico, na.rm = TRUE)

# Imputar el NA con la mediana
merged_full$Edad.al.diagnóstico[is.na(merged_full$Edad.al.diagnóstico)] <- mediana_age

# Verificación
```

```r
summary(merged_full$Edad.al.diagnóstico)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   29.00   60.50   66.00   65.76   72.00   87.00
```

```r
sum(is.na(merged_full$Edad.al.diagnóstico))
```

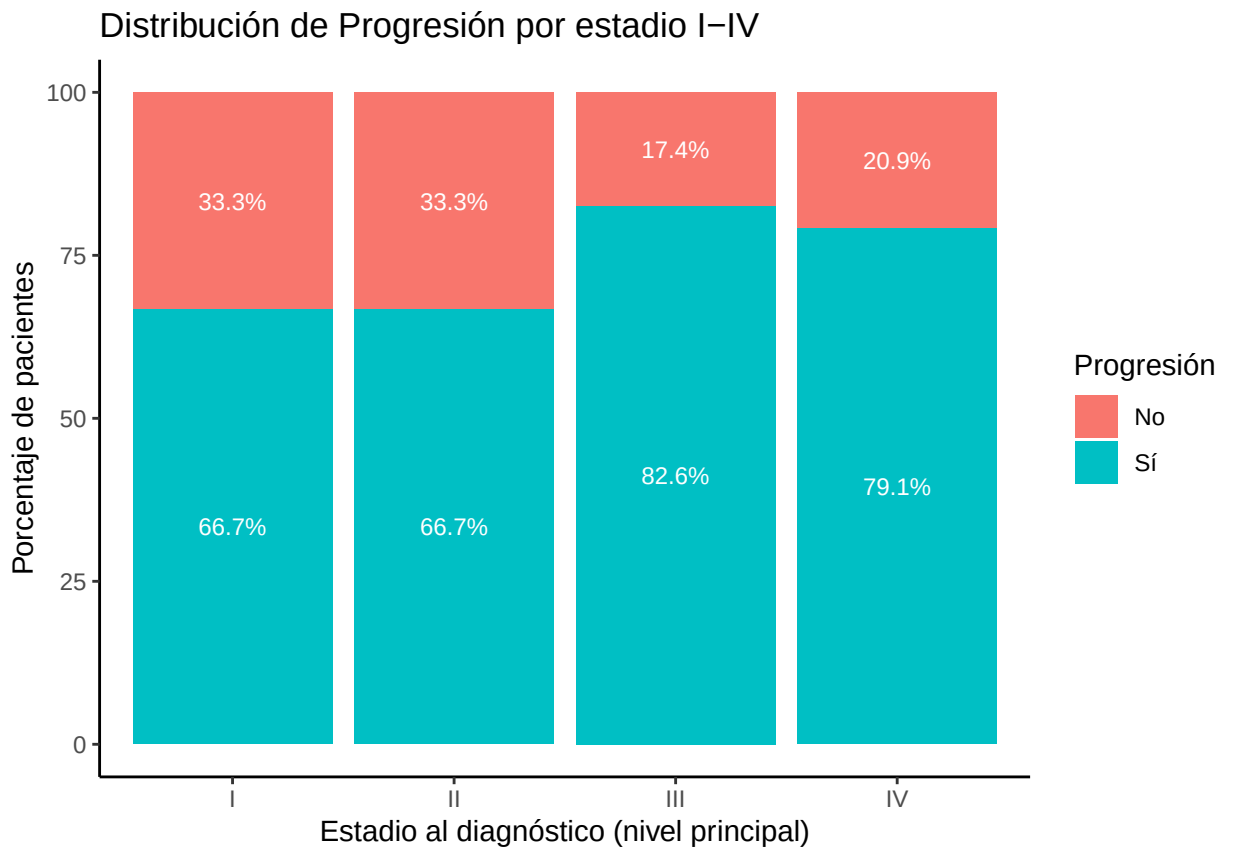
```
## [1] 0
```

Estratificación de progresión por estadíos

Progresión a punto final

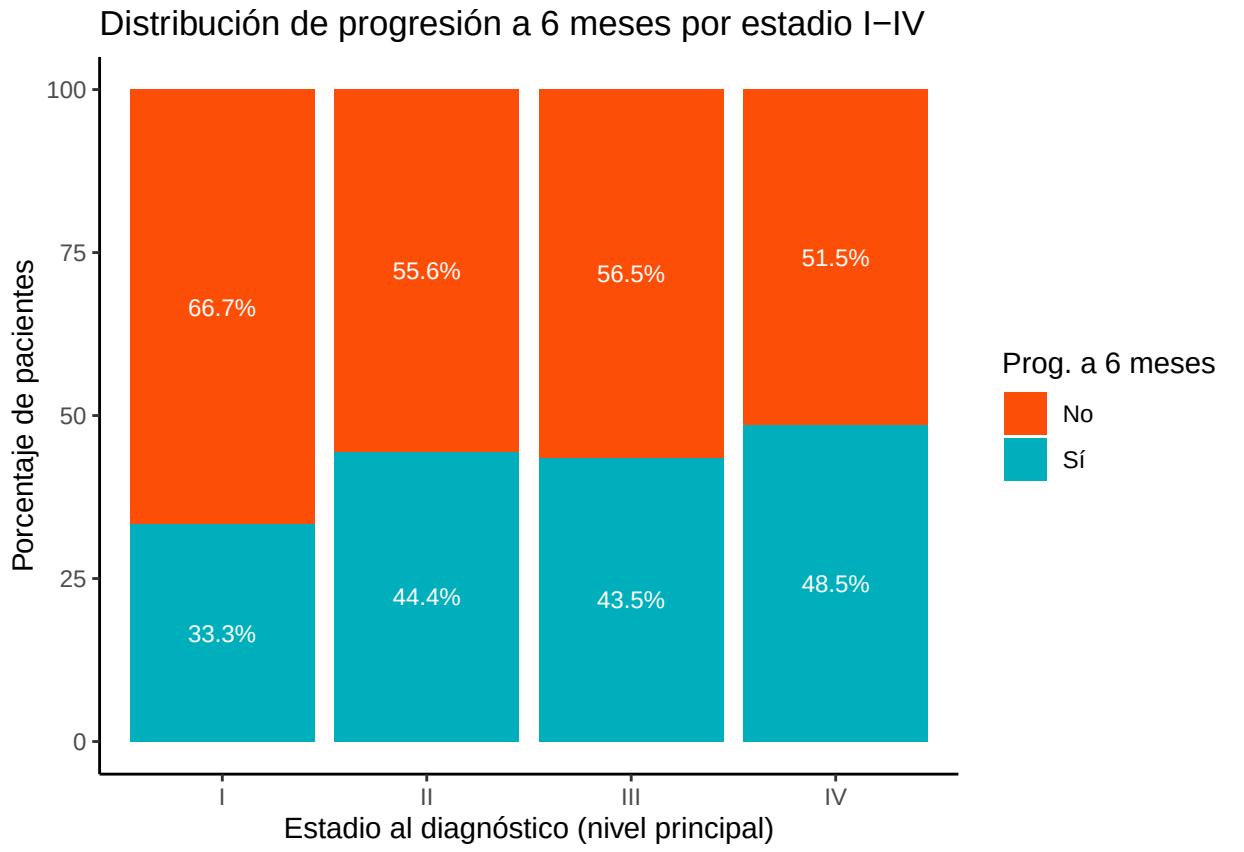
Estudiamos la distribución de los valores de progresión en base al estadio al diagnostico. Esta es una forma de balancear las clases previa a la aplicación del modelo.

```r
library(ggplot2)

# Extraer el nivel principal de estadio (I, II, III, IV)
merged_full2 <- merged_full %>%
  mutate(
    Estadio_princ = sub("[A-Z]$", "", Estadio.al.diagnóstico)
  ) %>%
  # Nos quedamos solo con I, II, III y IV
  filter(Estadio_princ %in% c("I", "II", "III", "IV"))

# Tabla de conteos y porcentajes por estadio principal y progresión
tabla_prog <- merged_full2 %>%
  count(Estadio_princ, Progresión) %>%
  group_by(Estadio_princ) %>%
  mutate(
    pct = round(n / sum(n) * 100, 1)
  ) %>%
  ungroup()

ggplot(tabla_prog, aes(x = Estadio_princ, y = pct, fill = Progresión)) +
  geom_col(position = "stack") +
  geom_text(aes(label = paste0(pct, "%")),
            position = position_stack(vjust = 0.5), size = 3, color = "white") +
  labs(
    x     = "Estadio al diagnóstico (nivel principal)",
    y     = "Porcentaje de pacientes",
    title = "Distribución de Progresión por estadio I-IV"
  ) +
  theme_classic()
```



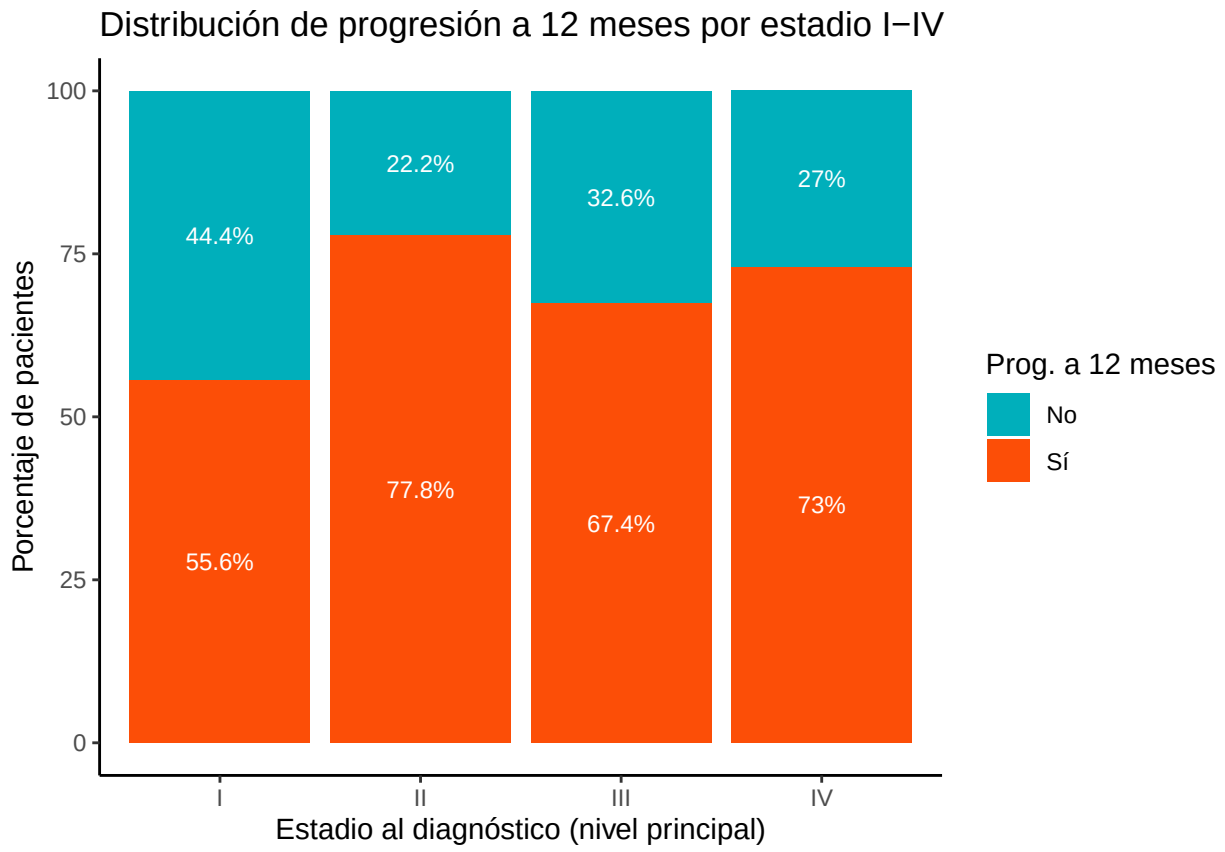

```r
# Extraer el nivel principal de estadio (I, II, III, IV)
merged_full2 <- merged_full %>%
  mutate(
    Estadio_princ = sub("[A-Z]$", "", Estadio.al.diagnóstico)
  ) %>%
  # Nos quedamos solo con I, II, III y IV
  filter(Estadio_princ %in% c("I", "II", "III", "IV"))

# Tabla de conteos y porcentajes por estadio principal y progresión a 6 meses
tabla_prog6 <- merged_full2 %>%
  count(Estadio_princ, Prog.6.meses) %>%
  group_by(Estadio_princ) %>%
  mutate(
    pct = round(n / sum(n) * 100, 1)
  ) %>%
  ungroup()

ggplot(tabla_prog6, aes(x = Estadio_princ, y = pct, fill = Prog.6.meses)) +
  geom_col(position = "stack") +
  geom_text(aes(label = paste0(pct, "%")),
            position = position_stack(vjust = 0.5),
            size = 3, color = "white") +
  scale_fill_manual(
    values = c("No" = "#FC4E07", "Sí" = "#00AFBB"),
    name   = "Prog. a 6 meses"
  ) +
```

```
  labs(
    x     = "Estadio al diagnóstico (nivel principal)",
    y     = "Porcentaje de pacientes",
    title = "Distribución de progresión a 6 meses por estadio I-IV"
  ) +
  theme_classic()
```




```
# Extraer el nivel principal de estadio (I, II, III, IV)
merged_full2 <- merged_full %>%
  mutate(
    Estadio_princ = sub("[A-Z]$", "", Estadio.al.diagnóstico)
  ) %>%
  filter(Estadio_princ %in% c("I", "II", "III", "IV"))

# Tabla de conteos y porcentajes por estadio principal y progresión a 12 meses
tabla_prog12 <- merged_full2 %>%
  count(Estadio_princ, Prog.12.meses) %>%
  group_by(Estadio_princ) %>%
  mutate(
    pct = round(n / sum(n) * 100, 1)
  ) %>%
  ungroup()

ggplot(tabla_prog12, aes(x = Estadio_princ, y = pct, fill = Prog.12.meses)) +
  geom_col(position = "stack") +
  geom_text(aes(label = paste0(pct, "%")),
```

```
            position = position_stack(vjust = 0.5),
            size = 3, color = "white") +
  scale_fill_manual(
    values = c("No" = "#00AFBB", "Sí" = "#FC4E07"),
    name   = "Prog. a 12 meses"
  ) +
  labs(
    x     = "Estadio al diagnóstico (nivel principal)",
    y     = "Porcentaje de pacientes",
    title = "Distribución de progresión a 12 meses por estadio I-IV"
  ) +
  theme_classic()
```



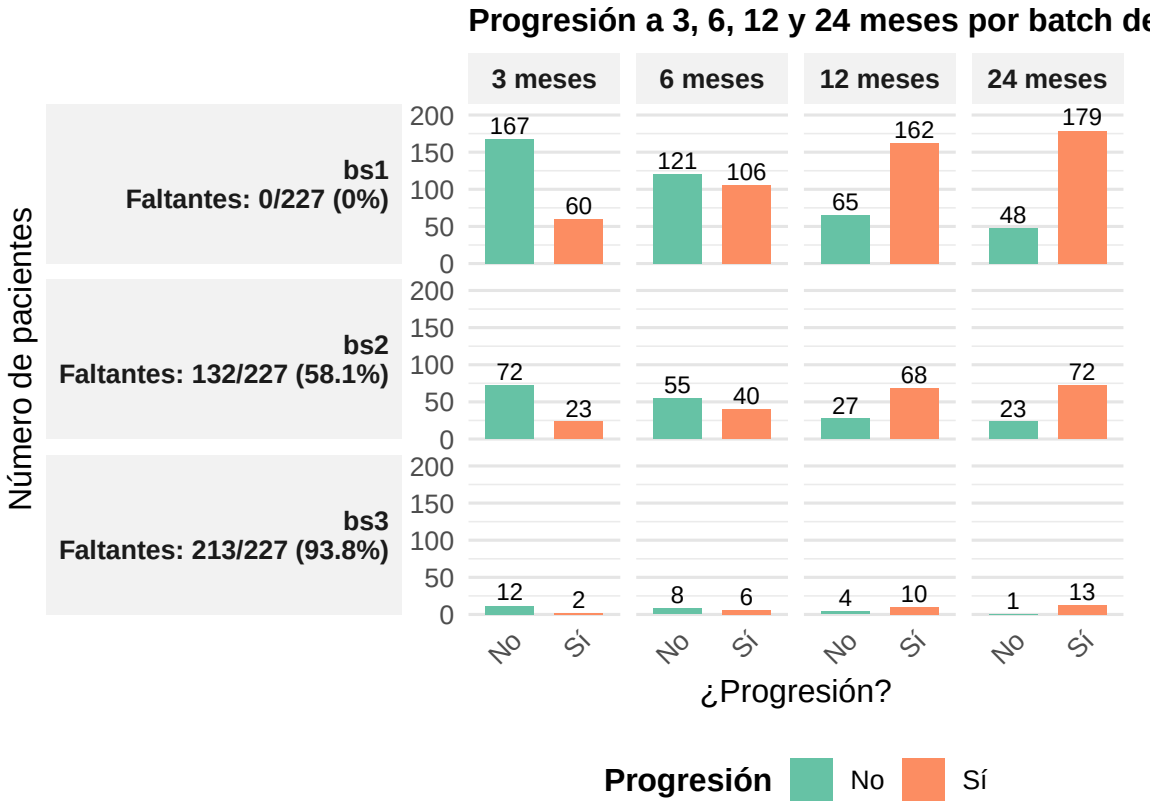
Calcular N de Progresión

Sobre el conjunto de datos plenamente integrado vamos a calcular la N de los progresados 3, 6 y 12 meses.

```
library(dplyr)
library(tidyr)
library(ggplot2)
library(RColorBrewer)

# 0) Creamos los flags has_bs1/has_bs2/has_bs3 sobre merged_full
merged_full <- merged_full %>%
  mutate(
```

```
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.))
  )


# 1) Preparamos los datos largos y filtramos sólo quienes tienen datos de cada batch
plot_data <- merged_full %>%
  pivot_longer(
    cols      = c(Prog.3.meses, Prog.6.meses, Prog.12.meses, Progresión),
    names_to  = "Periodo",
    values_to = "Progresion"
  ) %>%
  pivot_longer(
    cols      = c(has_bs1, has_bs2, has_bs3),
    names_to  = "Batch",
    values_to = "Disponible"
  ) %>%
  mutate(
    # renombramos cada nombre de columna a su etiqueta de meses
    Periodo = recode(Periodo,
      "Prog.3.meses"  = "3 meses",
      "Prog.6.meses"  = "6 meses",
      "Prog.12.meses" = "12 meses",
      "Progresión"    = "24 meses"
    ),
    # forzamos el orden deseado, incluyendo el nuevo nivel
    Periodo = factor(Periodo, levels = c("3 meses","6 meses","12 meses","24 meses")),
    Batch = recode(Batch,
      "has_bs1" = "bs1",
      "has_bs2" = "bs2",
      "has_bs3" = "bs3"
    )
  ) %>%
  filter(Disponible)   # sólo pacientes con datos para ese batch

# 2) Resumimos conteos por batch/periodo/progresión
plot_summary <- plot_data %>%
  count(Batch, Periodo, Progresion)

# 3) Calculamos cuántos faltan por batch y creamos labels para los strips
total_n <- nrow(merged_full)
batch_stats <- merged_full %>%
  summarise(
    bs1 = sum(has_bs1, na.rm = TRUE),
    bs2 = sum(has_bs2, na.rm = TRUE),
    bs3 = sum(has_bs3, na.rm = TRUE)
  ) %>%
  pivot_longer(everything(), names_to = "Batch", values_to = "presentes") %>%
  mutate(
    faltantes    = total_n - presentes,
    pct_faltante = round(faltantes/total_n * 100, 1),
    label_facet  = paste0(
```

```
      Batch, "\nFaltantes: ",
      faltantes, "/", total_n,
      " (", pct_faltante, "%)"
    )
  ) %>%
  select(Batch, label_facet)

batch_labels <- setNames(batch_stats$label_facet, batch_stats$Batch)

# 4) Dibujamos el gráfico
ggplot(plot_summary, aes(x = Progresion, y = n, fill = Progresion)) +
  geom_col(width = 0.7) +
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(
    Batch ~ Periodo,
    labeller  = labeller(Batch = batch_labels),
    switch    = "y"
  ) +
  scale_y_continuous(expand = expansion(mult = c(0, 0.2))) +
  scale_fill_brewer(
    palette = "Set2",
    name    = "Progresión",
    labels  = c("No", "Sí")
  ) +
  labs(
    x     = "¿Progresión?",
    y     = "Número de pacientes",
    title = "Progresión a 3, 6, 12 y 24 meses por batch de TCR"
  ) +
  coord_cartesian(clip = "off") +
  theme_minimal(base_size = 12) +
  theme(
    plot.title         = element_text(face = "bold", size = 12),
    strip.background   = element_rect(fill = "grey95", color = NA),
    strip.text         = element_text(face = "bold", size = 10),
    strip.placement    = "outside",
    strip.text.y.left  = element_text(angle = 0, hjust = 1),
    panel.grid.major.y = element_line(color = "grey90"),
    panel.grid.major.x = element_blank(),
    axis.text.x        = element_text(angle = 45, hjust = 1),
    legend.position    = "bottom",
    legend.title       = element_text(face = "bold"),
    legend.text        = element_text(size = 10),
    plot.margin        = margin(t = 10, r = 10, b = 10, l = 40)
  )
```

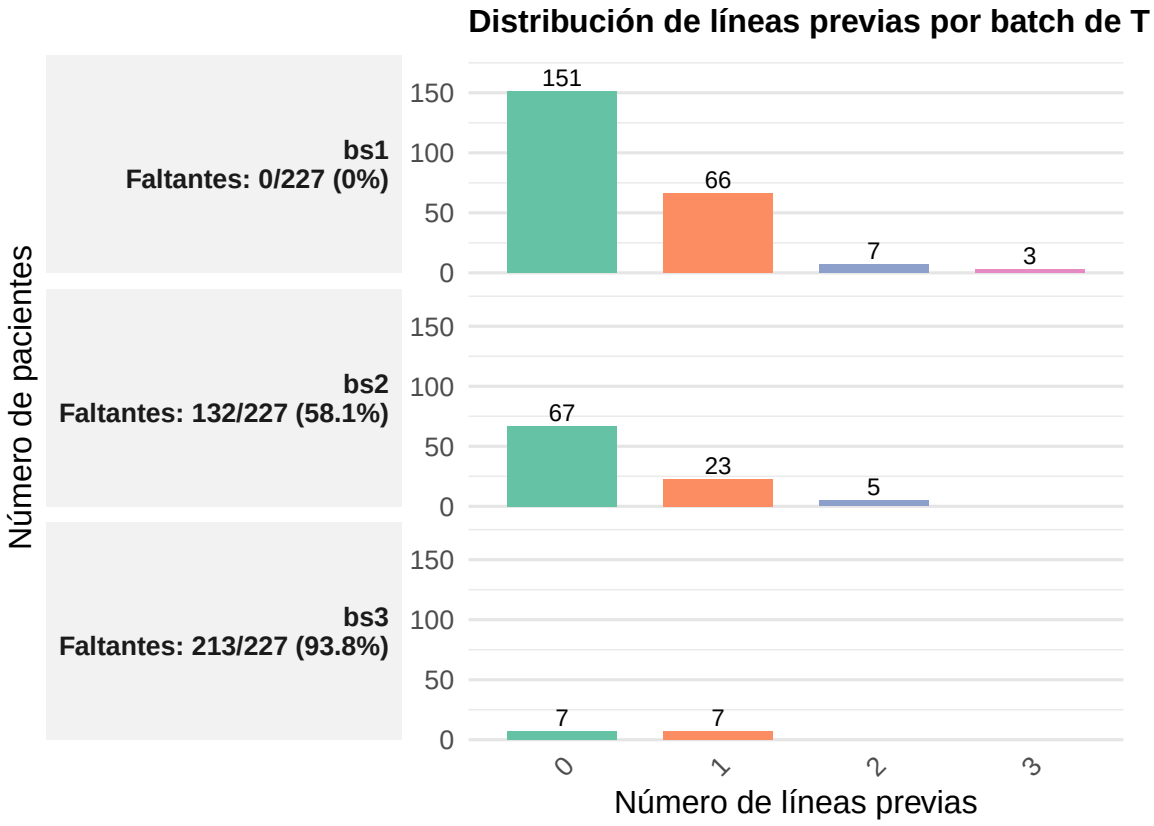


Calcular N por presencia de lineas previas

```r
# 0) Creamos los flags has_bs1/has_bs2/has_bs3 sobre merged_full
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.))
  )

# 1) Preparamos los datos "largos" para líneas previas
plot_data_lines <- merged_full %>%
  # Pivotamos las flags de disponibilidad para crear fila por batch
  pivot_longer(
    cols      = c(has_bs1, has_bs2, has_bs3),
    names_to = "Batch",
    values_to= "Disponible"
  ) %>%
  mutate(
    # Renombramos los batches
    Batch = recode(Batch,
      "has_bs1" = "bs1",
      "has_bs2" = "bs2",
      "has_bs3" = "bs3"
```

```
    ),
    # Convertimos Nº.de.líneas.previas a factor para que ggplot lo trate como categoría
    LinesPrevias = factor(`Nº.de.líneas.previas`)
  ) %>%
  # Sólo nos quedamos con los pacientes que realmente tienen datos de ese batch
  filter(Disponible)

# 2) Resumimos n por batch × valor de líneas previas
plot_summary_lines <- plot_data_lines %>%
  count(Batch, LinesPrevias)

# 3) Calculamos faltantes y montamos labels para los strips (igual que antes)
total_n <- nrow(merged_full)
batch_stats <- merged_full %>%
  summarise(
    bs1 = sum(has_bs1, na.rm = TRUE),
    bs2 = sum(has_bs2, na.rm = TRUE),
    bs3 = sum(has_bs3, na.rm = TRUE)
  ) %>%
  pivot_longer(everything(), names_to = "Batch", values_to = "presentes") %>%
  mutate(
    faltantes    = total_n - presentes,
    pct_faltante = round(faltantes/total_n * 100, 1),
    label_facet  = paste0(
      Batch, "\nFaltantes: ",
      faltantes, "/", total_n,
      " (", pct_faltante, "%)"
    )
  ) %>%
  select(Batch, label_facet)

batch_labels <- setNames(batch_stats$label_facet, batch_stats$Batch)

# 4) Dibujamos el bar-plot facetado por batch
ggplot(plot_summary_lines, aes(x = LinesPrevias, y = n, fill = LinesPrevias)) +
  geom_col(width = 0.7, show.legend = FALSE) +
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(
    Batch ~ .,
    labeller = labeller(Batch = batch_labels),
    switch   = "y"
  ) +
  scale_fill_brewer(palette = "Set2") +
  scale_y_continuous(expand = expansion(mult = c(0, 0.2))) +
  labs(
    x     = "Número de líneas previas",
    y     = "Número de pacientes",
    title = "Distribución de líneas previas por batch de TCR"
  ) +
  coord_cartesian(clip = "off") +
  theme_minimal(base_size = 12) +
  theme(
    plot.title         = element_text(face = "bold", size = 12),
```

```
    strip.background   = element_rect(fill = "grey95", color = NA),
    strip.text         = element_text(face = "bold", size = 10),
    strip.placement    = "outside",
    strip.text.y.left  = element_text(angle = 0, hjust = 1),
    panel.grid.major.y = element_line(color = "grey90"),
    panel.grid.major.x = element_blank(),
    axis.text.x        = element_text(angle = 45, hjust = 1),
    plot.margin        = margin(t = 10, r = 10, b = 10, l = 40)
  )
```

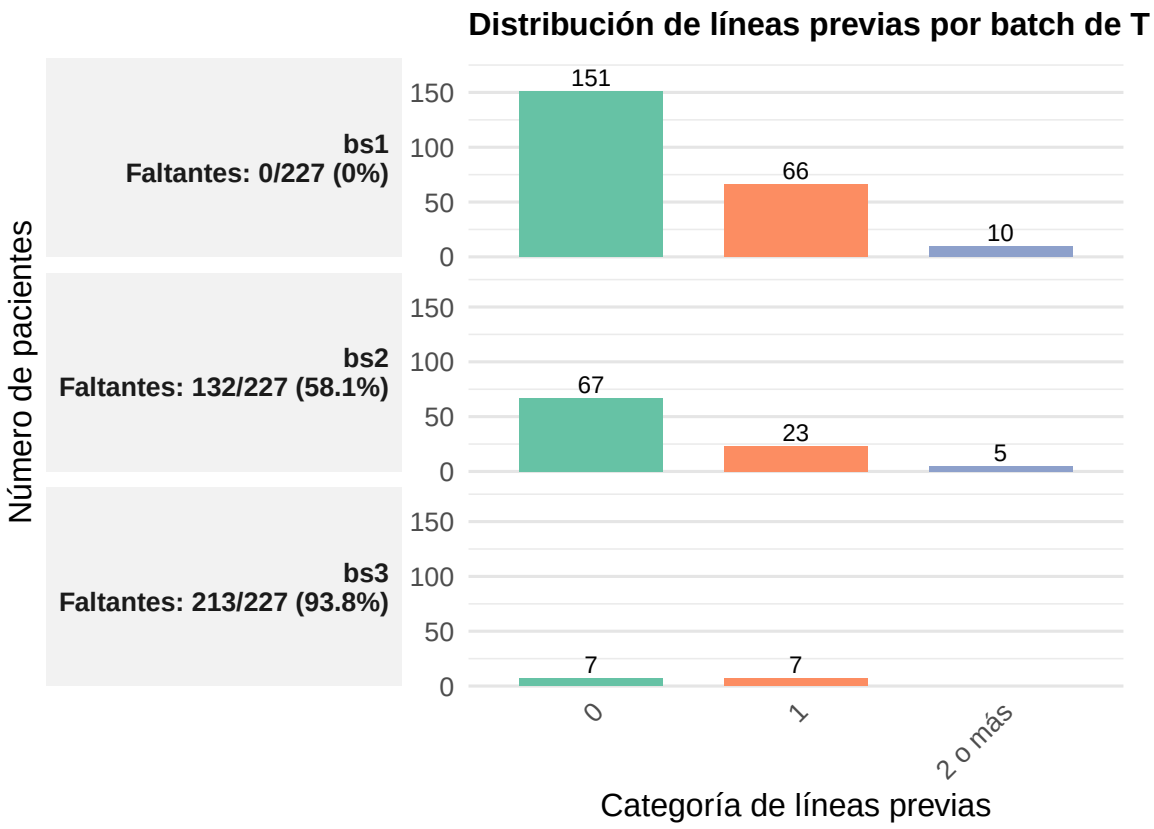


Calcular N por numero de lineas previas categorizadas

```
# 0) Creamos flags de disponibilidad bs1/bs2/bs3
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.))
  )

# 1) Preparamos los datos largos usando la nueva variable LíneasPrevias_Cat
plot_data_cat <- merged_full %>%
  pivot_longer(
    cols      = c(has_bs1, has_bs2, has_bs3),
    names_to = "Batch",
    values_to= "Disponible"
  ) %>%
```

```r
  mutate(
    Batch = recode(Batch,
                   "has_bs1" = "bs1",
                   "has_bs2" = "bs2",
                   "has_bs3" = "bs3")
  ) %>%
  filter(Disponible, !is.na(LíneasPrevias_Cat))

# 2) Resumimos n por batch × categoría de líneas previas
plot_summary_cat <- plot_data_cat %>%
  count(Batch, LíneasPrevias_Cat)

# 3) Calculamos etiquetas de faltantes para cada batch
total_n <- nrow(merged_full)
batch_stats <- merged_full %>%
  summarise(
    bs1 = sum(has_bs1, na.rm = TRUE),
    bs2 = sum(has_bs2, na.rm = TRUE),
    bs3 = sum(has_bs3, na.rm = TRUE)
  ) %>%
  pivot_longer(everything(), names_to = "Batch", values_to = "presentes") %>%
  mutate(
    faltantes    = total_n - presentes,
    pct_faltante = round(faltantes/total_n * 100, 1),
    label_facet  = paste0(
      Batch, "\nFaltantes: ",
      faltantes, "/", total_n,
      " (", pct_faltante, "%)"
    )
  ) %>%
  select(Batch, label_facet)

batch_labels <- setNames(batch_stats$label_facet, batch_stats$Batch)

# 4) Dibujamos el bar-plot facetado por batch, ahora con LíneasPrevias_Cat
ggplot(plot_summary_cat, aes(x = LíneasPrevias_Cat, y = n, fill = LíneasPrevias_Cat)) +
  geom_col(width = 0.7, show.legend = FALSE) +
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(
    Batch ~ .,
    labeller = labeller(Batch = batch_labels),
    switch   = "y"
  ) +
  scale_fill_brewer(palette = "Set2") +
  scale_y_continuous(expand = expansion(mult = c(0, 0.2))) +
  labs(
    x     = "Categoría de líneas previas",
    y     = "Número de pacientes",
    title = "Distribución de líneas previas por batch de TCR"
  ) +
  coord_cartesian(clip = "off") +
  theme_minimal(base_size = 12) +
  theme(
```

```
    plot.title         = element_text(face = "bold", size = 12),
    strip.background   = element_rect(fill = "grey95", color = NA),
    strip.text         = element_text(face = "bold", size = 10),
    strip.placement    = "outside",
    strip.text.y.left  = element_text(angle = 0, hjust = 1),
    panel.grid.major.y = element_line(color = "grey90"),
    panel.grid.major.x = element_blank(),
    axis.text.x        = element_text(angle = 45, hjust = 1),
    plot.margin        = margin(t = 10, r = 10, b = 10, l = 40)
  )
```

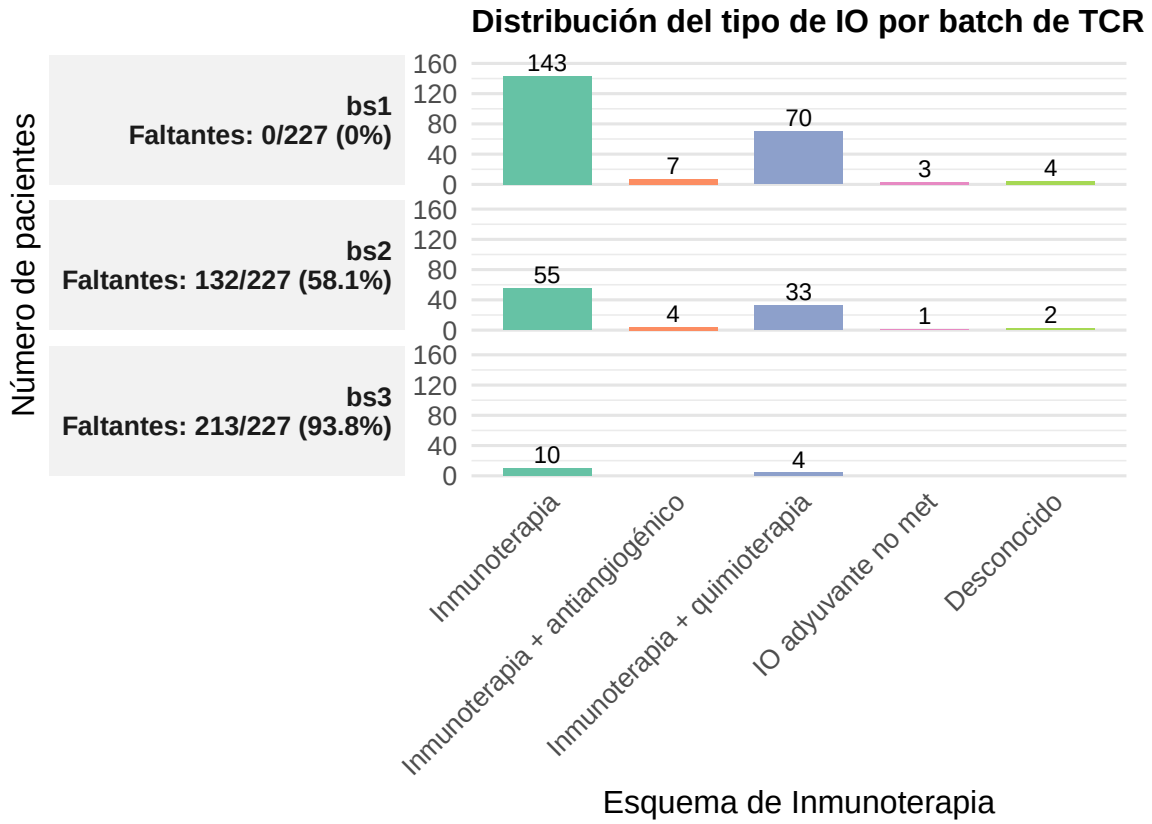


Calcular N por tipo de terapia

```
# 0) Aseguramos flags de disponibilidad bs1/bs2/bs3
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.))
  )

# 1) Preparamos datos para IO.Tipo.General
plot_data_io_gen <- merged_full %>%
  pivot_longer(
```

```
    cols      = c(has_bs1, has_bs2, has_bs3),
    names_to  = "Batch",
    values_to = "Disponible"
  ) %>%
  mutate(
    Batch = recode(Batch,
                   "has_bs1" = "bs1",
                   "has_bs2" = "bs2",
                   "has_bs3" = "bs3"),
    IO_gen = IO.Tipo.General
  ) %>%
  filter(Disponible, !is.na(IO_gen))

# 2) Resumimos conteos por batch × categoría IO.Tipo.General
plot_summary_io_gen <- plot_data_io_gen %>%
  count(Batch, IO_gen)

# 3) Creamos etiquetas de faltantes para cada batch
total_n <- nrow(merged_full)
batch_stats <- merged_full %>%
  summarise(
    bs1 = sum(has_bs1, na.rm = TRUE),
    bs2 = sum(has_bs2, na.rm = TRUE),
    bs3 = sum(has_bs3, na.rm = TRUE)
  ) %>%
  pivot_longer(everything(), names_to = "Batch", values_to = "presentes") %>%
  mutate(
    faltantes    = total_n - presentes,
    pct_faltante = round(faltantes / total_n * 100, 1),
    label_facet  = paste0(
      Batch, "\nFaltantes: ",
      faltantes, "/", total_n,
      " (", pct_faltante, "%)"
    )
  ) %>%
  select(Batch, label_facet)

batch_labels <- setNames(batch_stats$label_facet, batch_stats$Batch)

# 4) Gráfico de barras para IO.Tipo.General
ggplot(plot_summary_io_gen, aes(x = factor(IO_gen), y = n, fill = factor(IO_gen))) +
  geom_col(width = 0.7, show.legend = FALSE) +
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(
    Batch ~ .,
    labeller = labeller(Batch = batch_labels),
    switch   = "y"
  ) +
  scale_fill_brewer(
    palette = "Set2",
    name    = "Esquema IO",
    labels  = c(
      "Inmunoterapia",
```

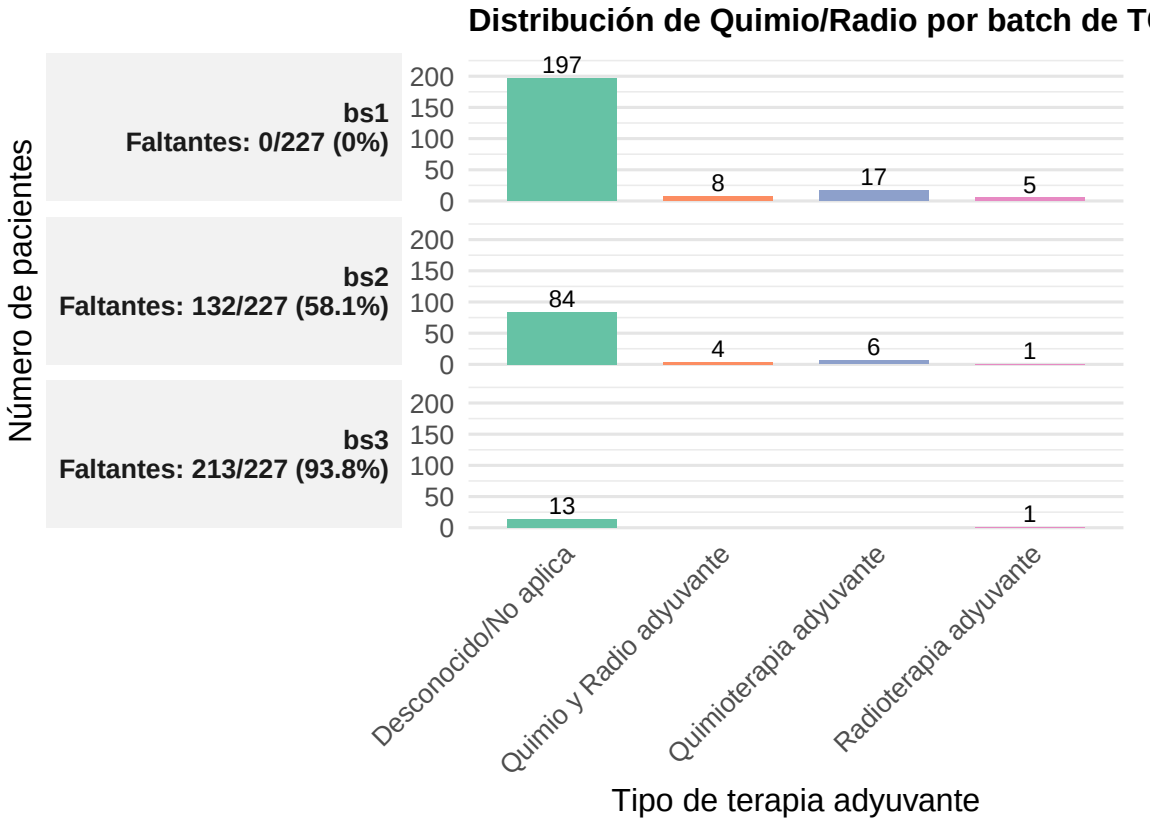
```
    "Inmunoterapia + quimioterapia",
    "Inmunoterapia + antiangiogénico",
    "IO adyuvante no met"
  )
) +
scale_y_continuous(expand = expansion(mult = c(0, 0.2))) +
labs(
  x     = "Esquema de Inmunoterapia",
  y     = "Número de pacientes",
  title = "Distribución del tipo de IO por batch de TCR"
) +
coord_cartesian(clip = "off") +
theme_minimal(base_size = 12) +
theme(
  plot.title         = element_text(face = "bold", size = 12),
  strip.background   = element_rect(fill = "grey95", color = NA),
  strip.text         = element_text(face = "bold", size = 10),
  strip.placement    = "outside",
  strip.text.y.left  = element_text(angle = 0, hjust = 1),
  panel.grid.major.y = element_line(color = "grey90"),
  panel.grid.major.x = element_blank(),
  axis.text.x        = element_text(angle = 45, hjust = 1),
  plot.margin        = margin(t = 10, r = 10, b = 10, l = 40)
)
```



Calcular N por tipo de terapia Adyuvante

```r
# 1) Creamos flags de disponibilidad bs1/bs2/bs3
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.))
  )

# 2) Preparamos los datos "largos" para Quimio.Radio.Adj
plot_data_qradj <- merged_full %>%
  pivot_longer(
    cols      = c(has_bs1, has_bs2, has_bs3),
    names_to  = "Batch",
    values_to = "Disponible"
  ) %>%
  mutate(
    Batch = recode(Batch,
                   "has_bs1" = "bs1",
                   "has_bs2" = "bs2",
                   "has_bs3" = "bs3"),
    QRA = Quimio.Radio.Adj
  ) %>%
  filter(Disponible, !is.na(QRA))

# 3) Resumimos conteos por batch × QRA
plot_summary_qradj <- plot_data_qradj %>%
  count(Batch, QRA)

# 4) Generamos las etiquetas de faltantes para los strips
total_n <- nrow(merged_full)
batch_stats <- merged_full %>%
  summarise(
    bs1 = sum(has_bs1, na.rm = TRUE),
    bs2 = sum(has_bs2, na.rm = TRUE),
    bs3 = sum(has_bs3, na.rm = TRUE)
  ) %>%
  pivot_longer(everything(), names_to = "Batch", values_to = "presentes") %>%
  mutate(
    faltantes    = total_n - presentes,
    pct_faltante = round(faltantes / total_n * 100, 1),
    label_facet  = paste0(
      Batch, "\nFaltantes: ",
      faltantes, "/", total_n,
      " (", pct_faltante, "%)"
    )
  ) %>%
  select(Batch, label_facet)

batch_labels <- setNames(batch_stats$label_facet, batch_stats$Batch)

# 5) Dibujamos el bar-plot facetado por batch
ggplot(plot_summary_qradj, aes(x = QRA, y = n, fill = QRA)) +
  geom_col(width = 0.7, show.legend = FALSE) +
```

```
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(
    Batch ~ .,
    labeller = labeller(Batch = batch_labels),
    switch   = "y"
  ) +
  scale_fill_brewer(
    palette = "Set2",
    name    = "Quimio/Radio adj",
    labels  = levels(merged_full$Quimio.Radio.Adj)
  ) +
  scale_y_continuous(expand = expansion(mult = c(0, 0.2))) +
  labs(
    x     = "Tipo de terapia adyuvante",
    y     = "Número de pacientes",
    title = "Distribución de Quimio/Radio por batch de TCR"
  ) +
  coord_cartesian(clip = "off") +
  theme_minimal(base_size = 12) +
  theme(
    plot.title         = element_text(face = "bold", size = 12),
    strip.background   = element_rect(fill = "grey95", color = NA),
    strip.text         = element_text(face = "bold", size = 10),
    strip.placement    = "outside",
    strip.text.y.left  = element_text(angle = 0, hjust = 1),
    panel.grid.major.y = element_line(color = "grey90"),
    panel.grid.major.x = element_blank(),
    axis.text.x        = element_text(angle = 45, hjust = 1),
    plot.margin        = margin(t = 10, r = 10, b = 10, l = 40)
  )
```

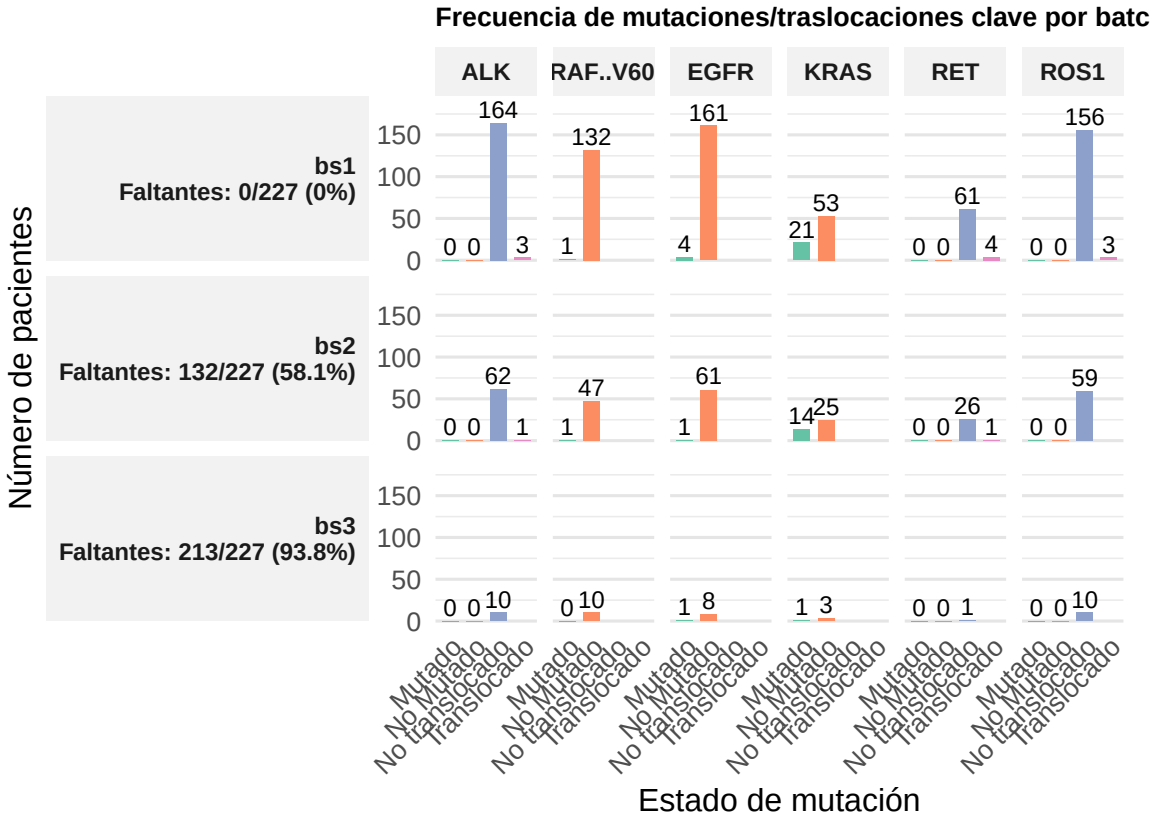


Calculo de N por mutación

```
library(dplyr)
library(tidyr)
library(ggplot2)
library(RColorBrewer)

# 0) (Re)crear flags de disponibilidad bs1/bs2/bs3
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.))
  )

# 1) Lista de genes y comprobación de existencia
genes_want    <- c("EGFR", "ALK", "ROS1", "RET", "BRAF..V600.", "KRAS")
genes_present <- intersect(genes_want, names(merged_full))
if (length(genes_present) == 0) {
  stop("Ninguno de los genes solicitados está presente en merged_full.")
}

# 2) Construcción del data.frame largo
gene_data <- merged_full %>%
  pivot_longer(
    cols      = all_of(genes_present),
    names_to = "Gen",
```

```
    values_to= "Estado"
  ) %>%
  pivot_longer(
    cols     = c(has_bs1, has_bs2, has_bs3),
    names_to = "Batch",
    values_to= "Disponible"
  ) %>%
  mutate(
    Batch = recode(Batch,
                   "has_bs1" = "bs1",
                   "has_bs2" = "bs2",
                   "has_bs3" = "bs3")
  ) %>%
  filter(Disponible, !is.na(Estado))

# 3) Contar y completar combinaciones faltantes
gene_summary <- gene_data %>%
  count(Batch, Gen, Estado) %>%
  complete(
    Batch  = c("bs1","bs2","bs3"),
    Gen    = genes_present,
    Estado = c("Mutado", "No Mutado"),
    fill   = list(n = 0)
  )

# 4) Calcular etiquetas de faltantes para cada batch
total_n <- nrow(merged_full)
batch_stats <- merged_full %>%
  summarise(
    bs1 = sum(has_bs1, na.rm = TRUE),
    bs2 = sum(has_bs2, na.rm = TRUE),
    bs3 = sum(has_bs3, na.rm = TRUE)
  ) %>%
  pivot_longer(everything(), names_to = "Batch", values_to = "presentes") %>%
  mutate(
    faltantes    = total_n - presentes,
    pct_faltante = round(faltantes/total_n * 100, 1),
    label_facet  = paste0(
      Batch, "\nFaltantes: ",
      faltantes, "/", total_n,
      " (", pct_faltante, "%)"
    )
  ) %>%
  select(Batch, label_facet)
batch_labels <- setNames(batch_stats$label_facet, batch_stats$Batch)

# 5) Gráfico facetado por Batch ~ Gen, mostrando Mutado vs No Mutado
ggplot(gene_summary, aes(x = Estado, y = n, fill = Estado)) +
  geom_col(width = 0.7, show.legend = FALSE) +
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(
    Batch ~ Gen,
    labeller  = labeller(Batch = batch_labels),
```

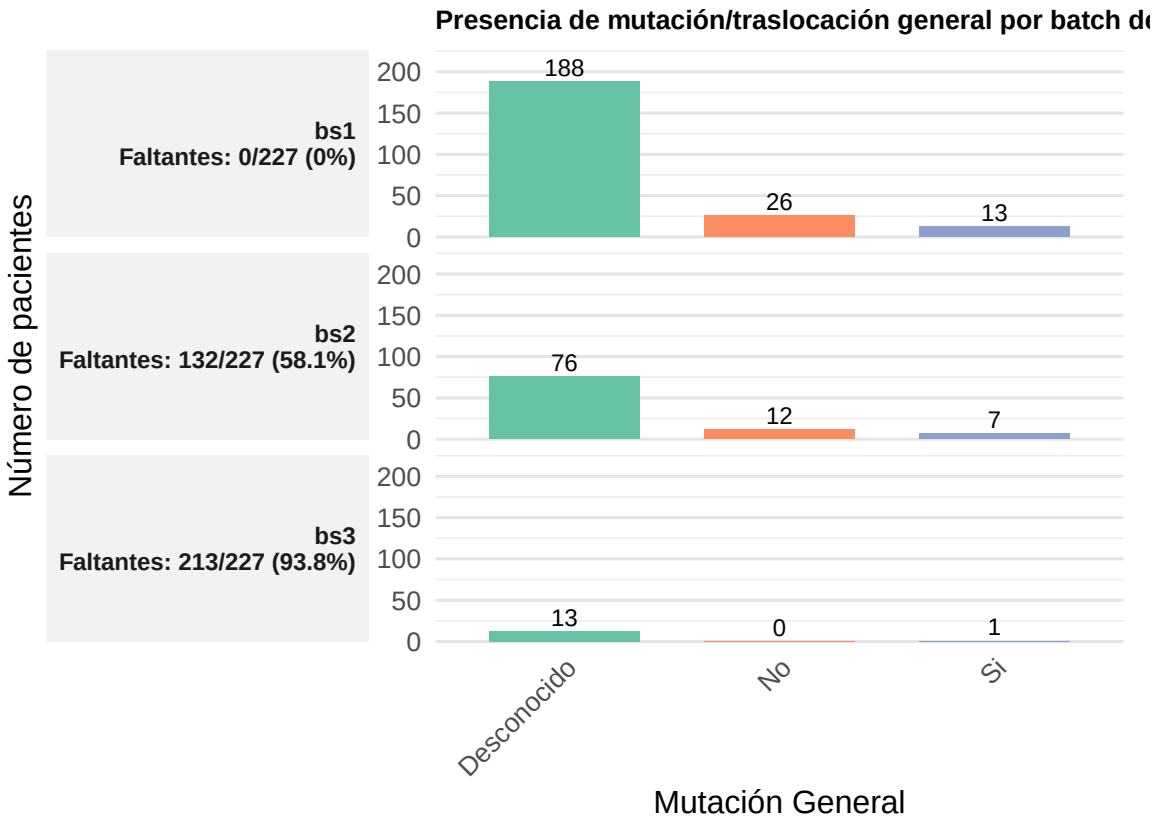
```
    switch    = "y",
    drop      = FALSE
  ) +
  scale_fill_brewer(palette = "Set2") +
  scale_y_continuous(expand = expansion(mult = c(0, 0.2))) +
  labs(
    x     = "Estado de mutación",
    y     = "Número de pacientes",
    title = "Frecuencia de mutaciones/traslocaciones clave por batch de TCR"
  ) +
  coord_cartesian(clip = "off") +
  theme_minimal(base_size = 12) +
  theme(
    plot.title        = element_text(face = "bold", size = 10),
    strip.background  = element_rect(fill = "grey95", color = NA),
    strip.text        = element_text(face = "bold", size = 9),
    strip.placement   = "outside",
    strip.text.y.left = element_text(angle = 0, hjust = 1),
    panel.grid.major.y = element_line(color = "grey90"),
    panel.grid.major.x = element_blank(),
    axis.text.x       = element_text(angle = 45, hjust = 1),
    plot.margin       = margin(t = 10, r = 10, b = 10, l = 40)
  )
```




```
library(dplyr)
library(tidyr)
```

```r
library(ggplot2)
library(RColorBrewer)

# 0) Aseguramos flags de disponibilidad bs1/bs2/bs3
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.))
  )

# 1) Preparamos los datos largos para Mutacion.General
mut_data <- merged_full %>%
  pivot_longer(
    cols      = c(has_bs1, has_bs2, has_bs3),
    names_to  = "Batch",
    values_to = "Disponible"
  ) %>%
  mutate(
    Batch = recode(Batch,
                   "has_bs1" = "bs1",
                   "has_bs2" = "bs2",
                   "has_bs3" = "bs3"),
    Mut = Mutacion.General
  ) %>%
  filter(Disponible, !is.na(Mut))

# 2) Contamos y completamos combinaciones faltantes
mut_summary <- mut_data %>%
  count(Batch, Mut) %>%
  complete(
    Batch = c("bs1","bs2","bs3"),
    Mut   = c("No","Si"),
    fill  = list(n = 0)
  )

# 3) Etiquetas de faltantes (reutilizamos batch_labels del gráfico anterior)
#    Si no lo tienes en el entorno, vuelve a crearlo como antes.

# 4) Gráfico facetado Batch ~
ggplot(mut_summary, aes(x = Mut, y = n, fill = Mut)) +
  geom_col(width = 0.7, show.legend = FALSE) +
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(
    Batch ~ .,
    labeller = labeller(Batch = batch_labels),
    switch   = "y",
    drop     = FALSE
  ) +
  scale_fill_brewer(palette = "Set2", name = "Mutación") +
  scale_y_continuous(expand = expansion(mult = c(0, 0.2))) +
  labs(
    x     = "Mutación General",
```

```
    y     = "Número de pacientes",
    title = "Presencia de mutación/traslocación general por batch de TCR"
  ) +
  coord_cartesian(clip = "off") +
  theme_minimal(base_size = 12) +
  theme(
    plot.title        = element_text(face = "bold", size = 10),
    strip.background  = element_rect(fill = "grey95", color = NA),
    strip.text        = element_text(face = "bold", size = 9),
    strip.placement   = "outside",
    strip.text.y.left = element_text(angle = 0, hjust = 1),
    panel.grid.major.y = element_line(color = "grey90"),
    panel.grid.major.x = element_blank(),
    axis.text.x       = element_text(angle = 45, hjust = 1),
    plot.margin       = margin(t = 10, r = 10, b = 10, l = 40)
  )
```

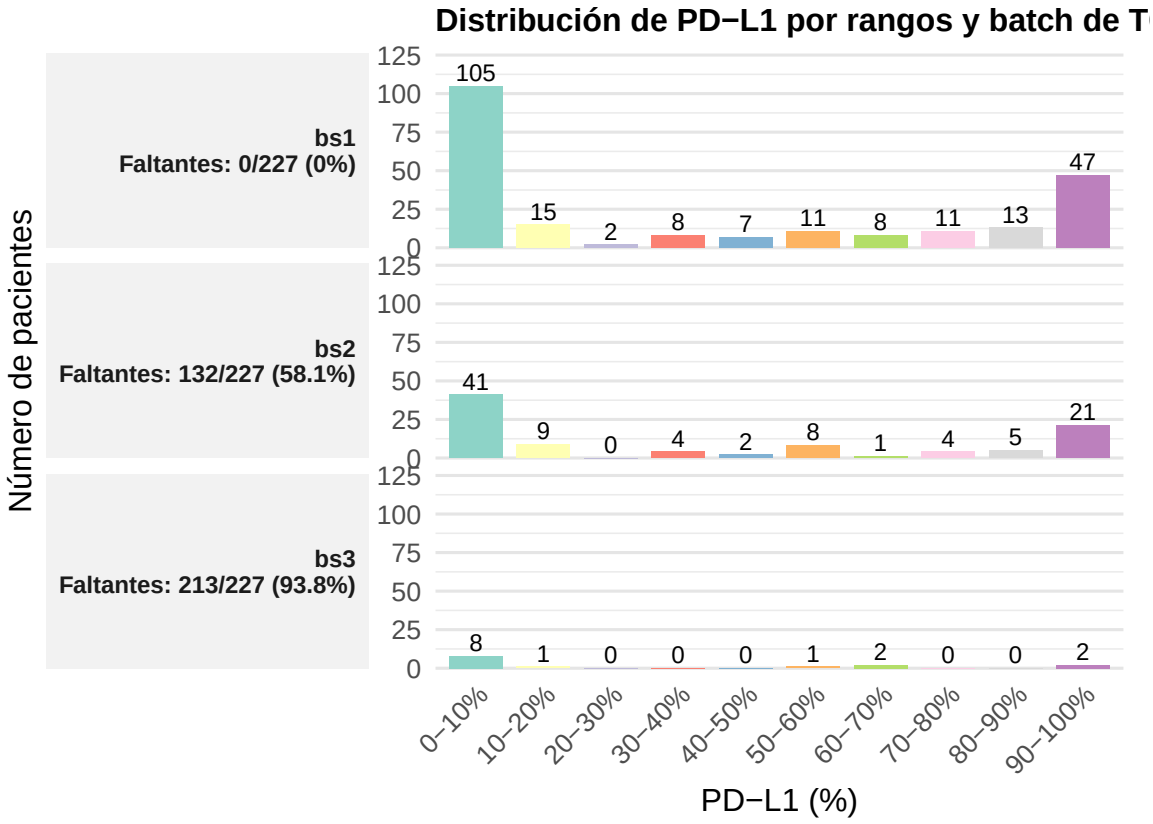


Calculo de N por rangos de PD-L1

```
library(dplyr)
library(tidyr)
library(ggplot2)
library(RColorBrewer)
```

```r
# 0) Aseguramos flags de disponibilidad bs1/bs2/bs3
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.))
  )

# 1) Creamos categorías de PD.L1 en intervalos de 10%
breaks <- seq(0, 100, by = 10)
labels <- paste0(head(breaks, -1), "-", tail(breaks, -1), "%")
merged_full <- merged_full %>%
  mutate(
    PD.L1_cat = cut(
      PD.L1,
      breaks = breaks,
      labels = labels,
      include.lowest = TRUE,
      right = FALSE
    )
  )

# 2) Preparamos datos largos para PD.L1_cat
pd_data <- merged_full %>%
  pivot_longer(
    cols      = c(has_bs1, has_bs2, has_bs3),
    names_to  = "Batch",
    values_to = "Disponible"
  ) %>%
  mutate(
    Batch = recode(Batch,
                   "has_bs1" = "bs1",
                   "has_bs2" = "bs2",
                   "has_bs3" = "bs3")
  ) %>%
  filter(Disponible, !is.na(PD.L1_cat))

# 3) Contamos y completamos combinaciones faltantes
pd_summary <- pd_data %>%
  count(Batch, PD.L1_cat) %>%
  complete(
    Batch     = c("bs1","bs2","bs3"),
    PD.L1_cat = labels,
    fill      = list(n = 0)
  )

# 4) Calculamos etiquetas de faltantes para cada batch
total_n <- nrow(merged_full)
batch_stats <- merged_full %>%
  summarise(
    bs1 = sum(has_bs1, na.rm = TRUE),
    bs2 = sum(has_bs2, na.rm = TRUE),
    bs3 = sum(has_bs3, na.rm = TRUE)
```

```
  ) %>%
  pivot_longer(everything(), names_to = "Batch", values_to = "presentes") %>%
  mutate(
    faltantes    = total_n - presentes,
    pct_faltante = round(faltantes/total_n * 100, 1),
    label_facet  = paste0(
      Batch, "\nFaltantes: ",
      faltantes, "/", total_n,
      " (", pct_faltante, "%)"
    )
  ) %>%
  select(Batch, label_facet)
batch_labels <- setNames(batch_stats$label_facet, batch_stats$Batch)

# 5) Gráfico facetado Batch ~ PD.L1_cat
ggplot(pd_summary, aes(x = PD.L1_cat, y = n, fill = PD.L1_cat)) +
  geom_col(width = 0.8, show.legend = FALSE) +
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(
    Batch ~ .,
    labeller = labeller(Batch = batch_labels),
    switch   = "y",
    drop     = FALSE
  ) +
  scale_fill_brewer(palette = "Set3") +
  scale_y_continuous(expand = expansion(mult = c(0, 0.2))) +
  labs(
    x     = "PD-L1 (%)",
    y     = "Número de pacientes",
    title = "Distribución de PD-L1 por rangos y batch de TCR"
  ) +
  coord_cartesian(clip = "off") +
  theme_minimal(base_size = 12) +
  theme(
    plot.title         = element_text(face = "bold", size = 12),
    strip.background   = element_rect(fill = "grey95", color = NA),
    strip.text         = element_text(face = "bold", size = 9),
    strip.placement    = "outside",
    strip.text.y.left  = element_text(angle = 0, hjust = 1),
    panel.grid.major.y = element_line(color = "grey90"),
    panel.grid.major.x = element_blank(),
    axis.text.x        = element_text(angle = 45, hjust = 1),
    plot.margin        = margin(t = 10, r = 10, b = 10, l = 40)
  )
```



Datos pareados

```
library(dplyr)
library(ggplot2)
library(scales)
```

```
##
## Adjuntando el paquete: 'scales'
```

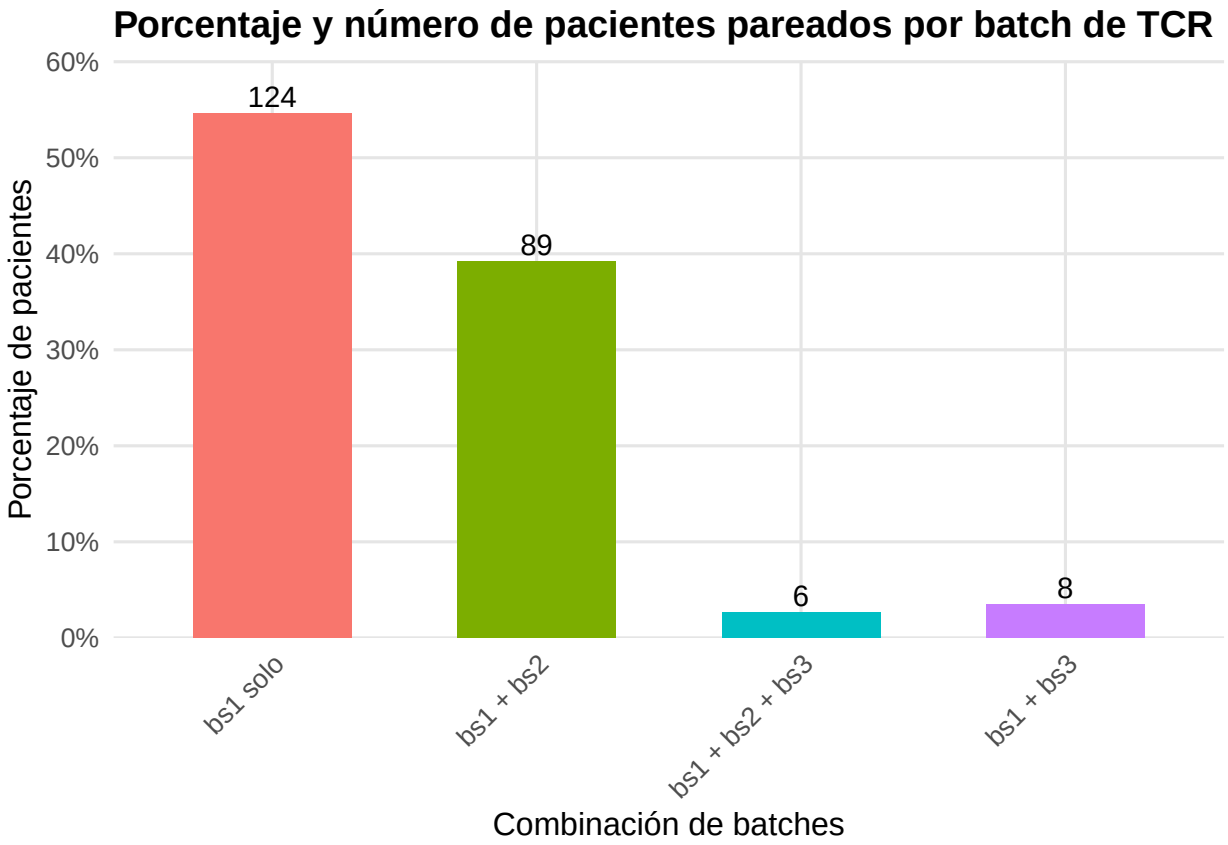
```
## The following object is masked from 'package:readr':
##
##     col_factor
```

```
# 0) Asegúrate de tener las flags de disponibilidad
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.))
  )

# 1) Calculamos los conteos y porcentajes de cada combinación
total_n <- nrow(merged_full)
combo_stats <- tibble(
  combo = c(
    "bs1 solo",
    "bs1 + bs2",
    "bs1 + bs2 + bs3",
```

```
    "bs1 + bs3"
  ),
  n = c(
    sum(merged_full$has_bs1 & !merged_full$has_bs2 & !merged_full$has_bs3, na.rm = TRUE),
    sum(merged_full$has_bs1 & merged_full$has_bs2 & !merged_full$has_bs3, na.rm = TRUE),
    sum(merged_full$has_bs1 & merged_full$has_bs2 & merged_full$has_bs3, na.rm = TRUE),
    sum(merged_full$has_bs1 & merged_full$has_bs3 & !merged_full$has_bs2, na.rm = TRUE)
  )
) %>%
  mutate(
    # forzamos el orden deseado
    combo = factor(combo, levels = c(
      "bs1 solo",
      "bs1 + bs2",
      "bs1 + bs2 + bs3",
      "bs1 + bs3"
    )),
    pct = n / total_n,
    pct_label = percent(pct, accuracy = 0.1)
  )

# 2) Mostrar los resultados en consola
print(combo_stats)
```

```
## # A tibble: 4 x 4
##   combo               n    pct pct_label
##   <fct>           <int>  <dbl> <chr>
## 1 bs1 solo          124 0.546  54.6%
## 2 bs1 + bs2          89 0.392  39.2%
## 3 bs1 + bs2 + bs3     6 0.0264 2.6%
## 4 bs1 + bs3           8 0.0352 3.5%
```

```
# 3) Gráfico de barras con el orden especificado y n encima
ggplot(combo_stats, aes(x = combo, y = pct, fill = combo)) +
  geom_col(width = 0.6, show.legend = FALSE) +
  geom_text(aes(label = n), vjust = -0.3, size = 4) +
  scale_y_continuous(
    labels = percent_format(accuracy = 1),
    expand = expansion(mult = c(0, 0.1))
  ) +
  labs(
    x     = "Combinación de batches",
    y     = "Porcentaje de pacientes",
    title = "Porcentaje y número de pacientes pareados por batch de TCR"
  ) +
  theme_minimal(base_size = 12) +
  theme(
    axis.text.x      = element_text(angle = 45, hjust = 1),
    plot.title       = element_text(face = "bold"),
    panel.grid.major = element_line(color = "grey90"),
    panel.grid.minor = element_blank()
  )
```

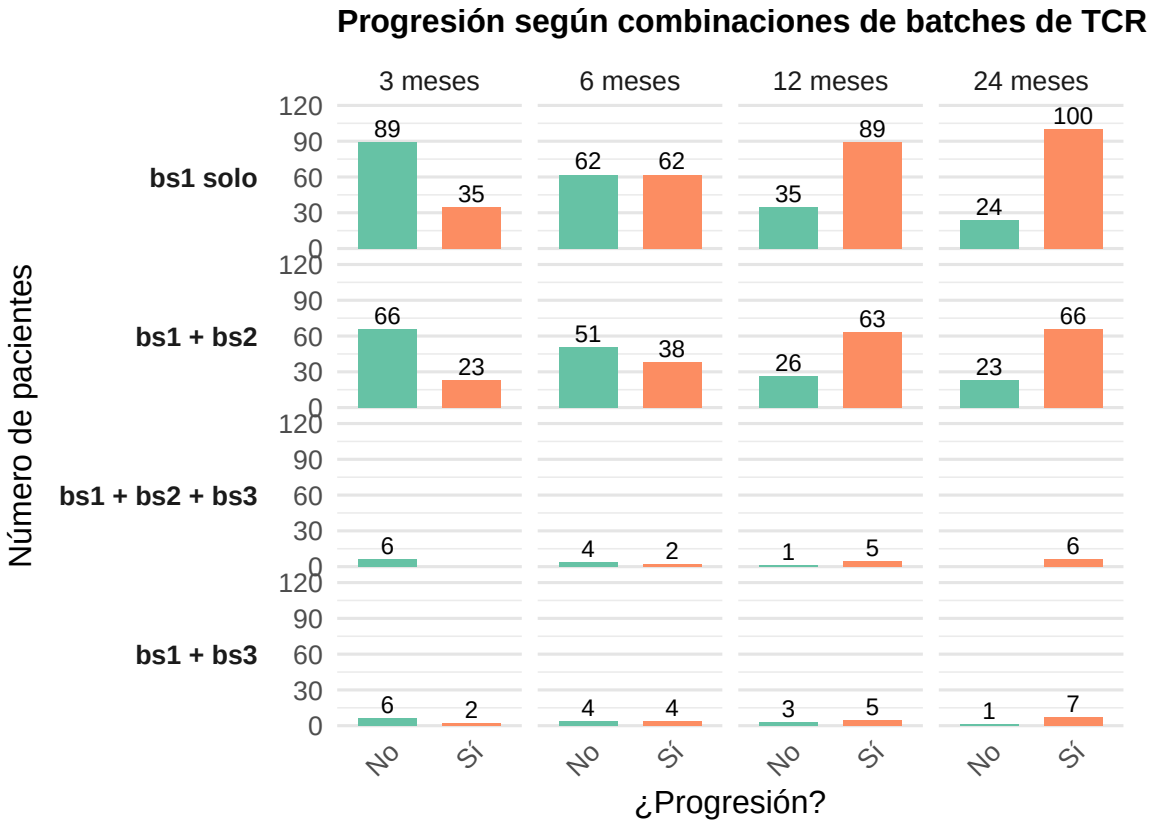


Calcular N de Progresión por datos pareados

```
library(dplyr)
library(tidyr)
library(ggplot2)
library(RColorBrewer)

# 0) Creamos los flags de disponibilidad bs1/bs2/bs3
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.))
  )

# 1) Definimos el grupo de pareado en función de bs1, bs2 y bs3
merged_full <- merged_full %>%
  mutate(
    PareadoGroup = case_when(
      has_bs1 & !has_bs2 & !has_bs3            ~ "bs1 solo",
      has_bs1 &  has_bs2 & !has_bs3            ~ "bs1 + bs2",
      has_bs1 &  has_bs2 &  has_bs3            ~ "bs1 + bs2 + bs3",
      has_bs1 & !has_bs2 &  has_bs3            ~ "bs1 + bs3",
      TRUE                                      ~ NA_character_
    ),
```

```
    PareadoGroup = factor(
      PareadoGroup,
      levels = c("bs1 solo","bs1 + bs2","bs1 + bs2 + bs3","bs1 + bs3")
    )
  ) %>%
  filter(!is.na(PareadoGroup))   # mantenemos solo los con bs1

# 2) Llevamos a formato largo las variables de progresión
plot_data_group <- merged_full %>%
  pivot_longer(
    cols      = c(Prog.3.meses, Prog.6.meses, Prog.12.meses, Progresión),
    names_to  = "Periodo",
    values_to = "Progresion"
  ) %>%
  mutate(
    Periodo = recode(Periodo,
      "Prog.3.meses"  = "3 meses",
      "Prog.6.meses"  = "6 meses",
      "Prog.12.meses" = "12 meses",
      "Progresión"    = "24 meses"
    ),
    Periodo = factor(Periodo, levels = c("3 meses","6 meses","12 meses","24 meses"))
  )

# 3) Resumimos conteos por grupo/periodo/progresión
plot_summary_group <- plot_data_group %>%
  count(PareadoGroup, Periodo, Progresion)

# 4) Dibujamos el gráfico facetado por PareadoGroup ~ Periodo
ggplot(plot_summary_group, aes(x = Progresion, y = n, fill = Progresion)) +
  geom_col(width = 0.7, show.legend = FALSE) +
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(
    PareadoGroup ~ Periodo,
    switch = "y"
  ) +
  scale_y_continuous(expand = expansion(mult = c(0, 0.2))) +
  scale_fill_brewer(
    palette = "Set2",
    name    = "Progresión",
    labels  = c("No","Sí")
  ) +
  labs(
    x     = "¿Progresión?",
    y     = "Número de pacientes",
    title = "Progresión según combinaciones de batches de TCR"
  ) +
  coord_cartesian(clip = "off") +
  theme_minimal(base_size = 12) +
  theme(
    plot.title           = element_text(face = "bold", size = 12),
    panel.grid.major.y   = element_line(color = "grey90"),
    panel.grid.major.x   = element_blank(),
```

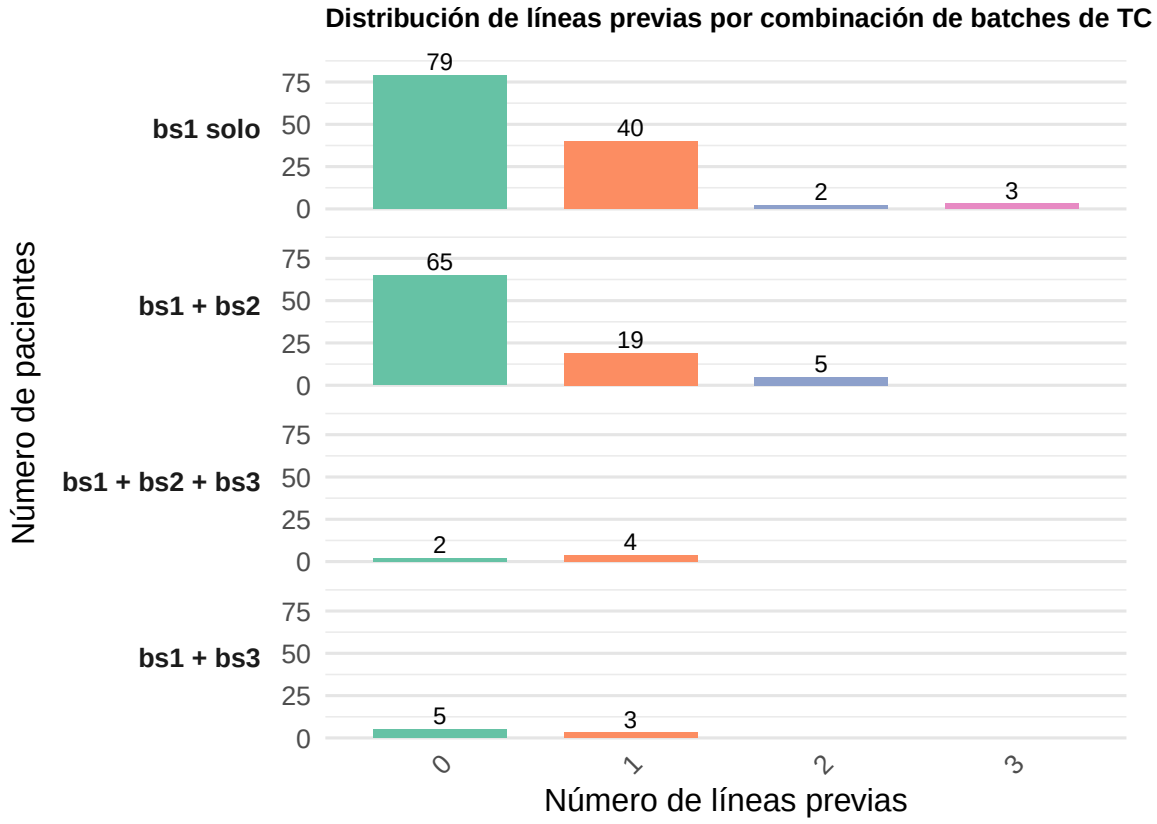
```
    strip.placement      = "outside",
    strip.text.y.left    = element_text(angle = 0, hjust = 1, face = "bold"),
    axis.text.x          = element_text(angle = 45, hjust = 1),
    plot.margin          = margin(t = 10, r = 10, b = 10, l = 40)
  )
```



Calcular N por presencia de lineas previas

```
library(dplyr)
library(ggplot2)
library(RColorBrewer)

# 0) (Re)creamos flags de disponibilidad y el grupo de pareado
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.)),
    PareadoGroup = case_when(
      has_bs1 & !has_bs2 & !has_bs3  ~ "bs1 solo",
      has_bs1 &  has_bs2 & !has_bs3  ~ "bs1 + bs2",
      has_bs1 &  has_bs2 &  has_bs3  ~ "bs1 + bs2 + bs3",
      has_bs1 & !has_bs2 &  has_bs3  ~ "bs1 + bs3",
      TRUE                           ~ NA_character_
    ),
    PareadoGroup = factor(
      PareadoGroup,
```

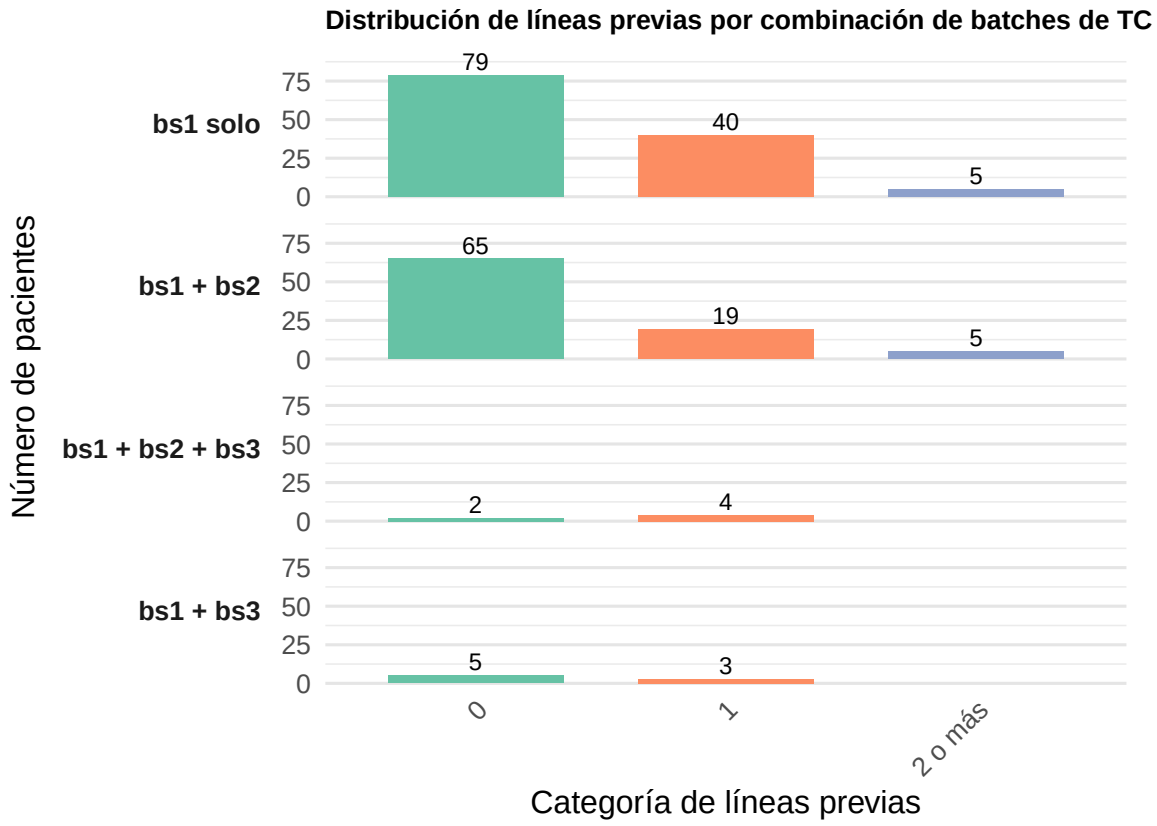
```
      levels = c("bs1 solo","bs1 + bs2","bs1 + bs2 + bs3","bs1 + bs3")
    ),
    LinesPrevias = factor(`Nº.de.líneas.previas`, levels = c("0","1","2","3"))
  ) %>%
  filter(!is.na(PareadoGroup))

# 1) Contamos pacientes por PareadoGroup y LinesPrevias
plot_summary_lp <- merged_full %>%
  count(PareadoGroup, LinesPrevias)

# 2) Gráfico facetado por PareadoGroup
ggplot(plot_summary_lp, aes(x = LinesPrevias, y = n, fill = LinesPrevias)) +
  geom_col(width = 0.7, show.legend = FALSE) +
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(
    PareadoGroup ~ .,
    switch = "y"
  ) +
  scale_fill_brewer(palette = "Set2") +
  scale_y_continuous(expand = expansion(mult = c(0, 0.2))) +
  labs(
    x     = "Número de líneas previas",
    y     = "Número de pacientes",
    title = "Distribución de líneas previas por combinación de batches de TCR"
  ) +
  coord_cartesian(clip = "off") +
  theme_minimal(base_size = 12) +
  theme(
    plot.title           = element_text(face = "bold", size = 10),
    strip.placement      = "outside",
    strip.text.y.left    = element_text(angle = 0, hjust = 1, face = "bold"),
    axis.text.x          = element_text(angle = 45, hjust = 1),
    panel.grid.major.y   = element_line(color = "grey90"),
    panel.grid.major.x   = element_blank(),
    plot.margin          = margin(t = 10, r = 10, b = 10, l = 40)
  )
```



Calcular N por combinación de batches y por numero de lineas previas categorizadas

```
library(dplyr)
library(ggplot2)
library(RColorBrewer)

# 0) (Re)creamos flags de disponibilidad y el grupo de pareado
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.)),
    # definimos el grupo de pareado igual que antes
    PareadoGroup = case_when(
      has_bs1 & !has_bs2 & !has_bs3  ~ "bs1 solo",
      has_bs1 &  has_bs2 & !has_bs3  ~ "bs1 + bs2",
      has_bs1 &  has_bs2 &  has_bs3  ~ "bs1 + bs2 + bs3",
      has_bs1 & !has_bs2 &  has_bs3  ~ "bs1 + bs3",
      TRUE                           ~ NA_character_
    ),
    PareadoGroup = factor(
      PareadoGroup,
      levels = c("bs1 solo","bs1 + bs2","bs1 + bs2 + bs3","bs1 + bs3")
    )
  ) %>%
  filter(!is.na(PareadoGroup))
```

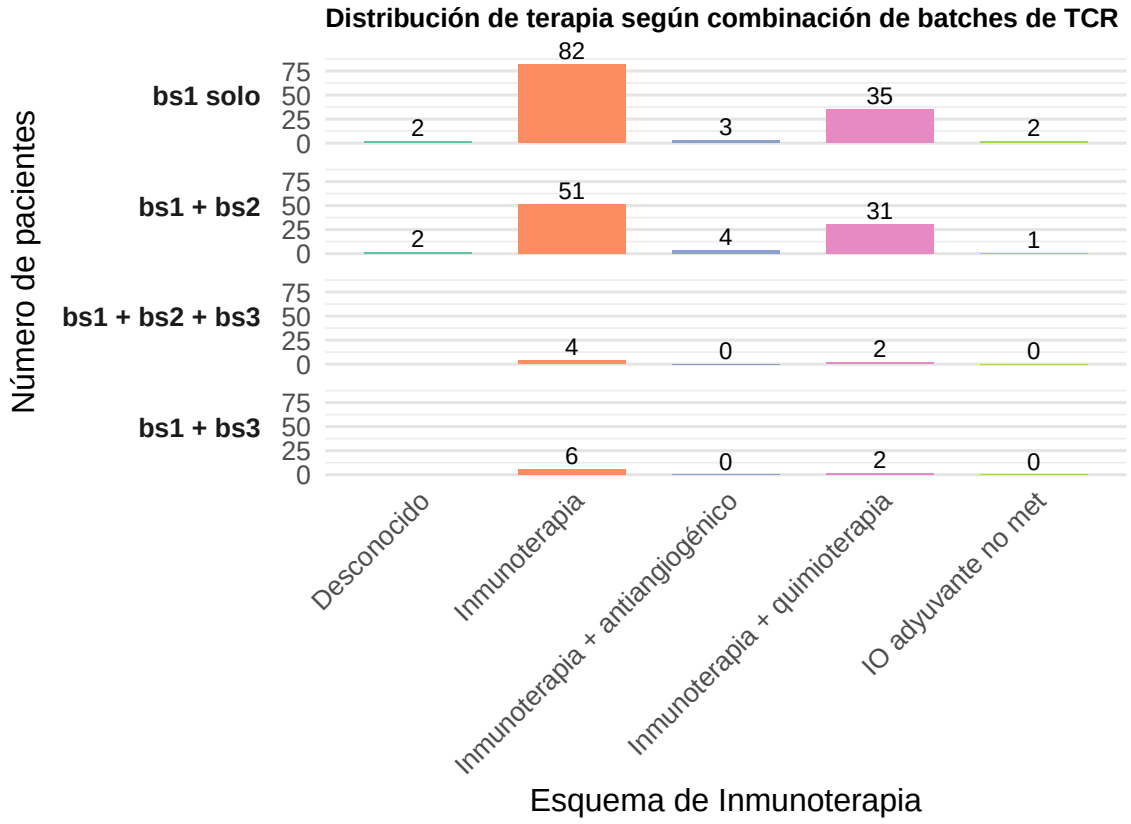
```r
# 1) Contamos el número de pacientes por grupo de pareado y categoría de líneas previas
plot_summary_lp_cat <- merged_full %>%
  # aseguramos que LíneasPrevias_Cat ya exista y esté factorizada
  count(PareadoGroup, LíneasPrevias_Cat)

# 2) Gráfico facetado por PareadoGroup
ggplot(plot_summary_lp_cat, aes(x = LíneasPrevias_Cat, y = n, fill = LíneasPrevias_Cat)) +
  geom_col(width = 0.7, show.legend = FALSE) +
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(
    PareadoGroup ~ .,
    switch = "y"
  ) +
  scale_fill_brewer(palette = "Set2") +
  scale_y_continuous(expand = expansion(mult = c(0, 0.2))) +
  labs(
    x     = "Categoría de líneas previas",
    y     = "Número de pacientes",
    title = "Distribución de líneas previas por combinación de batches de TCR"
  ) +
  coord_cartesian(clip = "off") +
  theme_minimal(base_size = 12) +
  theme(
    plot.title           = element_text(face = "bold", size = 10),
    strip.placement      = "outside",
    strip.text.y.left    = element_text(angle = 0, hjust = 1, face = "bold"),
    axis.text.x          = element_text(angle = 45, hjust = 1),
    panel.grid.major.y   = element_line(color = "grey90"),
    panel.grid.major.x   = element_blank(),
    plot.margin          = margin(t = 10, r = 10, b = 10, l = 40)
  )
```



Calcular N por combinación de batches y por tipo de terapia

```
library(dplyr)
library(tidyr)
library(ggplot2)
library(RColorBrewer)

# 0) Recreamos flags y definimos el grupo de pareado
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.)),
    PareadoGroup = case_when(
      has_bs1 & !has_bs2 & !has_bs3  ~ "bs1 solo",
      has_bs1 &  has_bs2 & !has_bs3  ~ "bs1 + bs2",
      has_bs1 &  has_bs2 &  has_bs3  ~ "bs1 + bs2 + bs3",
      has_bs1 & !has_bs2 &  has_bs3  ~ "bs1 + bs3",
      TRUE                           ~ NA_character_
    ),
    PareadoGroup = factor(
      PareadoGroup,
      levels = c("bs1 solo","bs1 + bs2","bs1 + bs2 + bs3","bs1 + bs3")
    )
  ) %>%
  filter(!is.na(PareadoGroup))
```

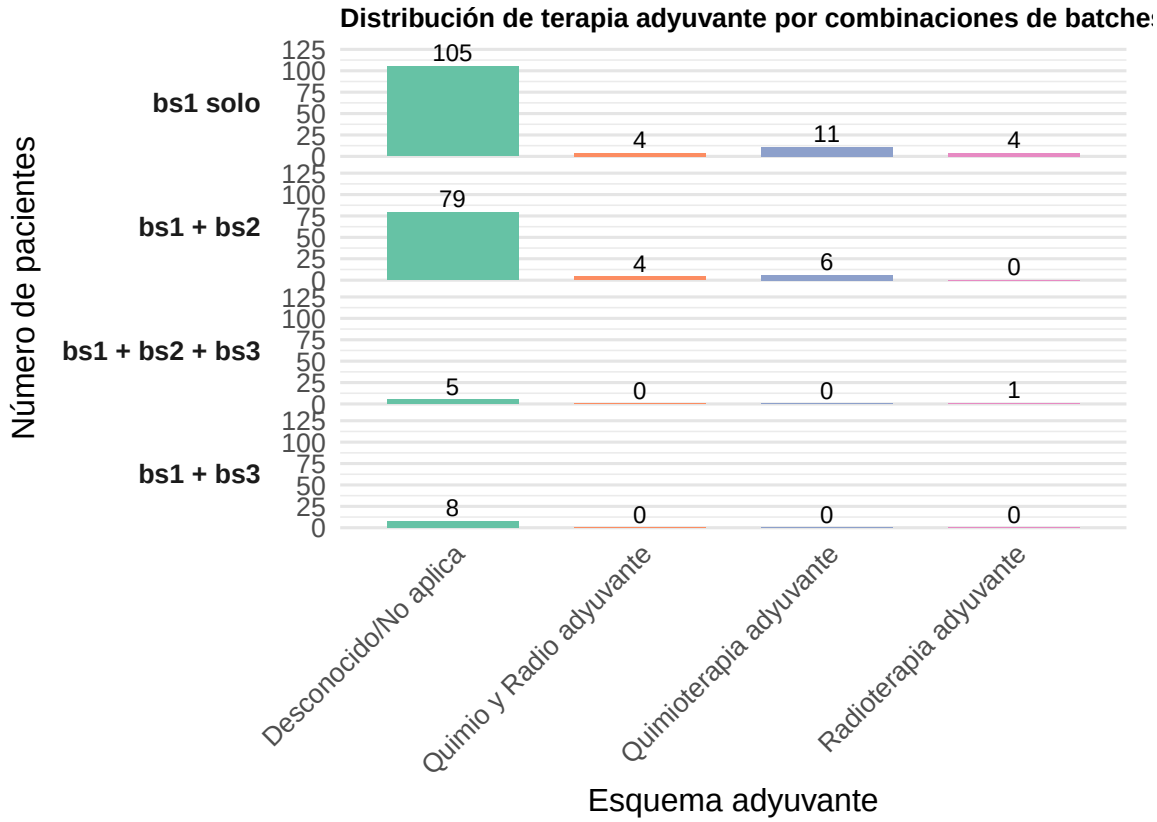
```r
# 1) Preparamos y contamos IO.Tipo.General por grupo de pareado
plot_summary_io_gen_grp <- merged_full %>%
  filter(!is.na(IO.Tipo.General)) %>%
  count(PareadoGroup, IO.Tipo.General) %>%
  complete(
    PareadoGroup,
    IO.Tipo.General = c(
      "Inmunoterapia",
      "Inmunoterapia + quimioterapia",
      "Inmunoterapia + antiangiogénico",
      "IO adyuvante no met"
    ),
    fill = list(n = 0)
  )

# 2) Gráfico facetado PareadoGroup ~
ggplot(plot_summary_io_gen_grp,
       aes(x = IO.Tipo.General, y = n, fill = IO.Tipo.General)) +
  geom_col(width = 0.7, show.legend = FALSE) +
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(
    PareadoGroup ~ .,
    switch = "y"
  ) +
  scale_fill_brewer(
    palette = "Set2",
    name    = "Esquema IO"
  ) +
  scale_y_continuous(
    expand = expansion(mult = c(0, 0.2))
  ) +
  labs(
    x     = "Esquema de Inmunoterapia",
    y     = "Número de pacientes",
    title = "Distribución de terapia según combinación de batches de TCR"
  ) +
  coord_cartesian(clip = "off") +
  theme_minimal(base_size = 12) +
  theme(
    plot.title         = element_text(face = "bold", size = 10),
    strip.placement    = "outside",
    strip.text.y.left  = element_text(angle = 0, hjust = 1, face = "bold"),
    axis.text.x        = element_text(angle = 45, hjust = 1),
    panel.grid.major.y = element_line(color = "grey90"),
    panel.grid.major.x = element_blank(),
    plot.margin        = margin(t = 10, r = 10, b = 10, l = 40)
  )
```



Calcular N por tipo de terapia Adyuvante

```
library(dplyr)
library(ggplot2)
library(tidyr)
library(RColorBrewer)

# 0) Creamos flags de disponibilidad y definimos los grupos pareados
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.)),
    PareadoGroup = case_when(
      has_bs1 & !has_bs2 & !has_bs3 ~ "bs1 solo",
      has_bs1 &  has_bs2 & !has_bs3 ~ "bs1 + bs2",
      has_bs1 &  has_bs2 &  has_bs3 ~ "bs1 + bs2 + bs3",
      has_bs1 & !has_bs2 &  has_bs3 ~ "bs1 + bs3",
      TRUE                          ~ NA_character_
    ),
    PareadoGroup = factor(
      PareadoGroup,
      levels = c("bs1 solo", "bs1 + bs2", "bs1 + bs2 + bs3", "bs1 + bs3")
    )
  ) %>%
  filter(!is.na(PareadoGroup))
```

```r
# 1) Contamos por grupo pareado y Quimio.Radio.Adj
all_qra <- levels(merged_full$Quimio.Radio.Adj)
qradj_summary <- merged_full %>%
  filter(!is.na(Quimio.Radio.Adj)) %>%
  count(PareadoGroup, Quimio.Radio.Adj) %>%
  complete(
    PareadoGroup,
    Quimio.Radio.Adj = all_qra,
    fill = list(n = 0)
  )

# 2) Gráfico facetado por PareadoGroup
ggplot(qradj_summary,
       aes(x = Quimio.Radio.Adj, y = n, fill = Quimio.Radio.Adj)) +
  geom_col(width = 0.7, show.legend = FALSE) +
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(PareadoGroup ~ ., switch = "y", drop = FALSE) +
  scale_fill_brewer(palette = "Set2") +
  scale_y_continuous(expand = expansion(mult = c(0, 0.2))) +
  labs(
    x     = "Esquema adyuvante",
    y     = "Número de pacientes",
    title = "Distribución de terapia adyuvante por combinaciones de batches de TCR"
  ) +
  coord_cartesian(clip = "off") +
  theme_minimal(base_size = 12) +
  theme(
    plot.title         = element_text(face = "bold", size = 10),
    strip.placement    = "outside",
    strip.text.y.left  = element_text(angle = 0, hjust = 1, face = "bold"),
    axis.text.x        = element_text(angle = 45, hjust = 1),
    panel.grid.major.y = element_line(color = "grey90"),
    panel.grid.major.x = element_blank(),
    plot.margin        = margin(t = 10, r = 10, b = 10, l = 40)
  )
```

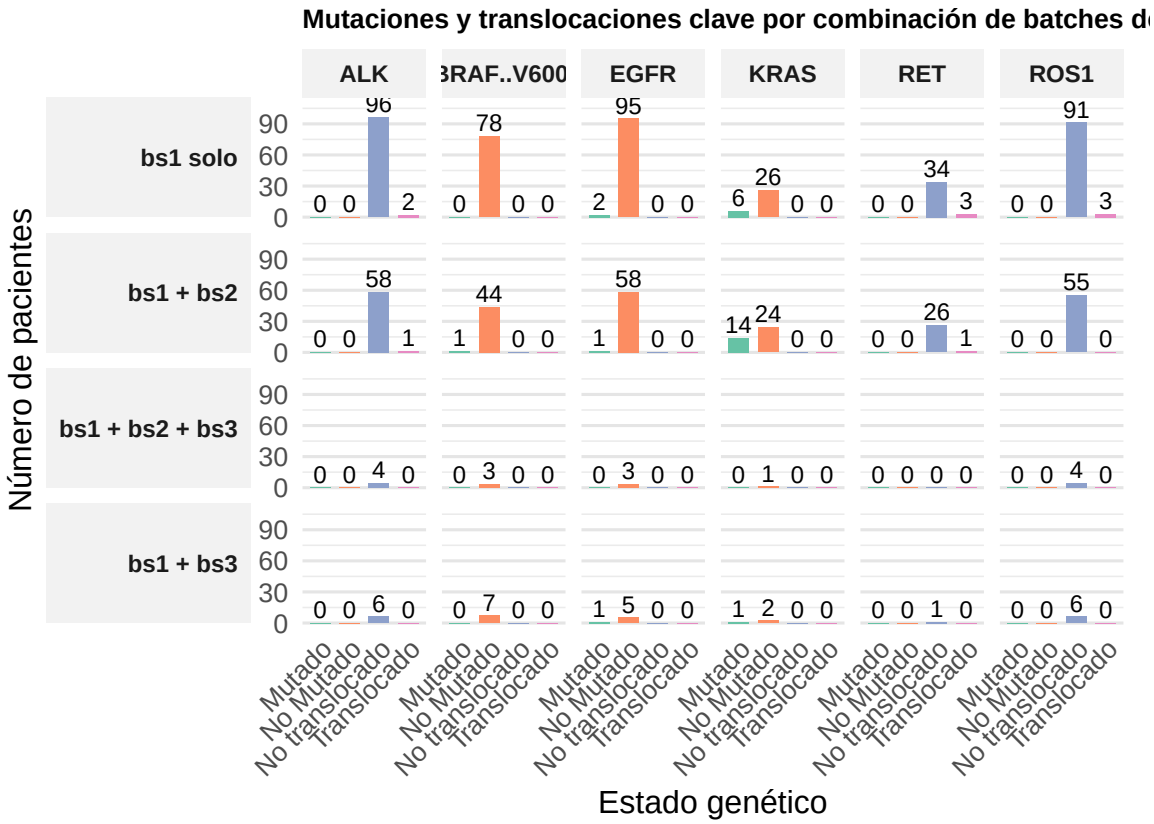


Calculo de N por mutación y combinación de batches de TCR

```
library(dplyr)
library(tidyr)
library(ggplot2)
library(RColorBrewer)

# 0) Recreamos flags y definimos grupos pareados
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.)),
    PareadoGroup = case_when(
      has_bs1 & !has_bs2 & !has_bs3 ~ "bs1 solo",
      has_bs1 &  has_bs2 & !has_bs3 ~ "bs1 + bs2",
      has_bs1 &  has_bs2 &  has_bs3 ~ "bs1 + bs2 + bs3",
      has_bs1 & !has_bs2 &  has_bs3 ~ "bs1 + bs3",
      TRUE                          ~ NA_character_
    ),
    PareadoGroup = factor(
      PareadoGroup,
      levels = c("bs1 solo","bs1 + bs2","bs1 + bs2 + bs3","bs1 + bs3")
    )
  ) %>%
  filter(!is.na(PareadoGroup))
```

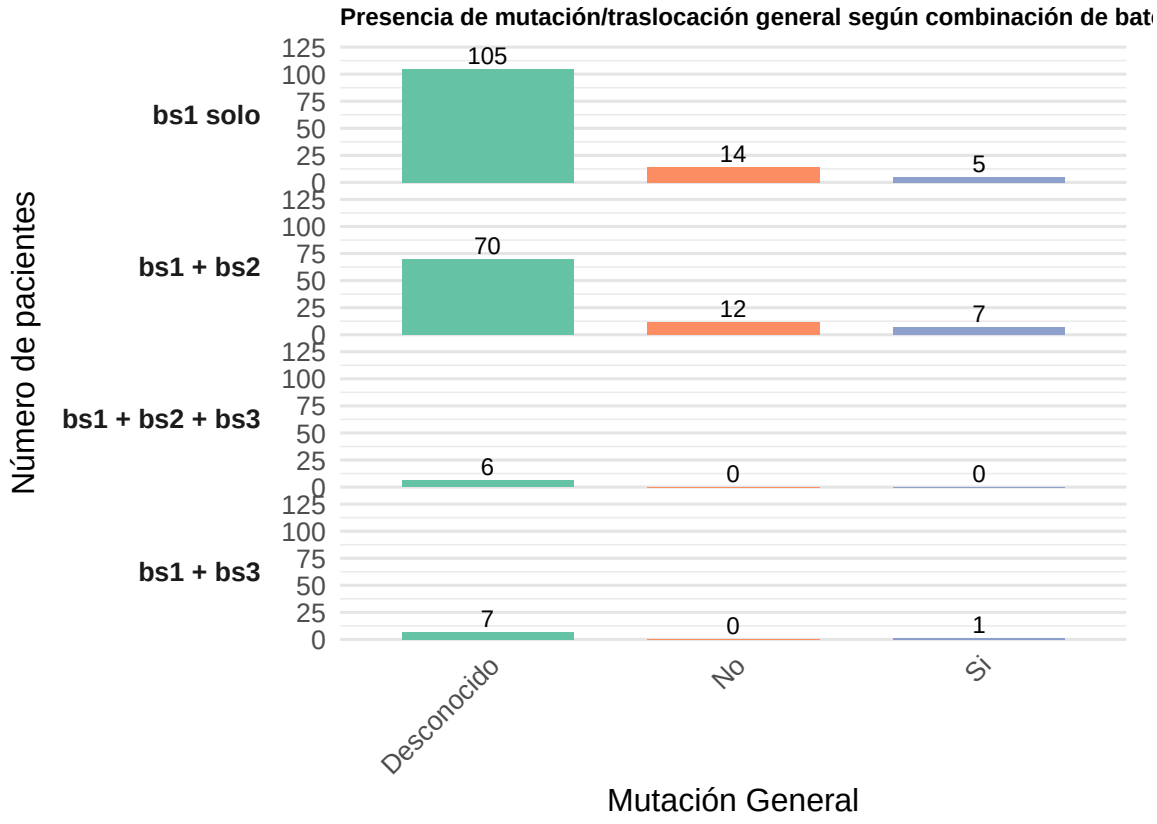
```r
# 1) Genes de interés
genes_want    <- c("EGFR", "ALK", "ROS1", "RET", "BRAF..V600.", "KRAS")
genes_present <- intersect(genes_want, names(merged_full))

# 2) Ponemos en formato largo y ajustamos "Estado" según cada gen
gene_summary_grp <- merged_full %>%
  pivot_longer(
    cols     = all_of(genes_present),
    names_to = "Gen",
    values_to= "Estado"
  ) %>%
  mutate(
    # Para los genes translocados, recogemos tanto "Si"/"No" como "Translocado"/"No translocado"
    Estado = case_when(
      Gen %in% c("ALK","RET","ROS1") &
        Estado %in% c("Si","sí","YES","Translocado")      ~ "Translocado",
      Gen %in% c("ALK","RET","ROS1") &
        Estado %in% c("No","NO","No translocado")         ~ "No translocado",
      # Para los genes mutados, mantenemos "Mutado"/"No Mutado"
      Gen %in% c("EGFR","KRAS","BRAF..V600.")              ~ Estado,
      TRUE                                                 ~ NA_character_
    ),
    # Unificamos niveles y orden
    Estado = factor(
      Estado,
      levels = c("No Mutado","Mutado","No translocado","Translocado")
    )
  ) %>%
  filter(!is.na(Estado)) %>%
  # 3) Contamos y completamos combinaciones faltantes
  count(PareadoGroup, Gen, Estado) %>%
  complete(
    PareadoGroup,
    Gen    = genes_present,
    Estado = levels(Estado),
    fill   = list(n = 0)
  )

# 4) Gráfico facetado PareadoGroup ~ Gen
ggplot(gene_summary_grp, aes(x = Estado, y = n, fill = Estado)) +
  geom_col(width = 0.7, show.legend = FALSE) +
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(
    PareadoGroup ~ Gen,
    switch = "y",
    drop   = FALSE
  ) +
  scale_fill_brewer(palette = "Set2") +
  scale_y_continuous(expand = expansion(mult = c(0, 0.2))) +
  labs(
    x     = "Estado genético",
    y     = "Número de pacientes",
    title = "Mutaciones y translocaciones clave por combinación de batches de TCR"
```

```
  ) +
  coord_cartesian(clip = "off") +
  theme_minimal(base_size = 12) +
  theme(
    plot.title         = element_text(face = "bold", size = 10),
    strip.background   = element_rect(fill = "grey95", color = NA),
    strip.text         = element_text(face = "bold", size = 9),
    strip.placement    = "outside",
    strip.text.y.left  = element_text(angle = 0, hjust = 1),
    axis.text.x        = element_text(angle = 45, hjust = 1),
    panel.grid.major.y = element_line(color = "grey90"),
    panel.grid.major.x = element_blank(),
    plot.margin        = margin(t = 10, r = 10, b = 10, l = 40)
  )
```




```
library(dplyr)
library(tidyr)
library(ggplot2)
library(RColorBrewer)

# 0) Recreamos flags de disponibilidad y definimos el grupo de pareado
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.)),
```

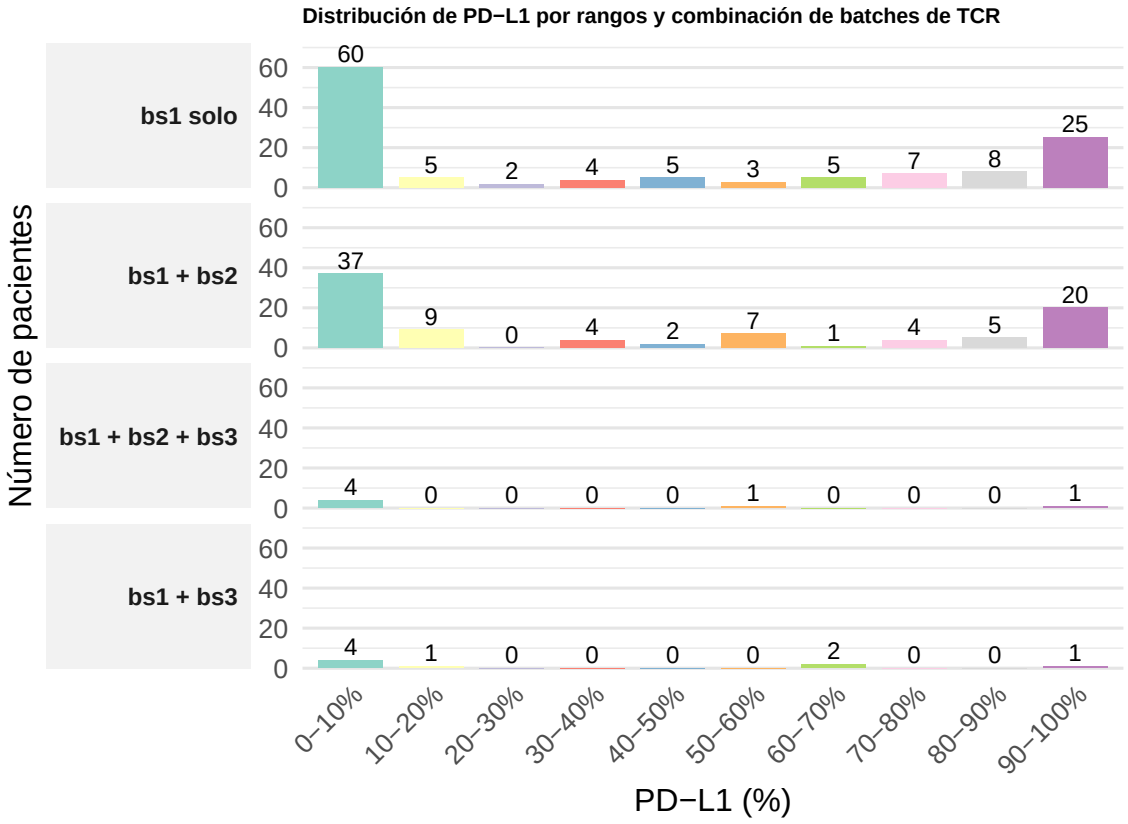
```r
    PareadoGroup = case_when(
      has_bs1 & !has_bs2 & !has_bs3 ~ "bs1 solo",
      has_bs1 &  has_bs2 & !has_bs3 ~ "bs1 + bs2",
      has_bs1 &  has_bs2 &  has_bs3 ~ "bs1 + bs2 + bs3",
      has_bs1 & !has_bs2 &  has_bs3 ~ "bs1 + bs3",
      TRUE                          ~ NA_character_
    ),
    PareadoGroup = factor(
      PareadoGroup,
      levels = c("bs1 solo","bs1 + bs2","bs1 + bs2 + bs3","bs1 + bs3")
    )
  ) %>%
  filter(!is.na(PareadoGroup))  # sólo los que tienen bs1

# 1) Contamos la Mutación.General por grupo de pareado
mut_summary_pg <- merged_full %>%
  filter(!is.na(Mutacion.General)) %>%
  count(PareadoGroup, Mutacion.General) %>%
  complete(
    PareadoGroup,
    Mutacion.General = c("No", "Si", "Desconocido"),
    fill = list(n = 0)
  )

# 2) Dibujamos el gráfico facetado por PareadoGroup
ggplot(mut_summary_pg, aes(x = Mutacion.General, y = n, fill = Mutacion.General)) +
  geom_col(width = 0.7, show.legend = FALSE) +
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(
    PareadoGroup ~ .,
    switch = "y",
    drop   = FALSE
  ) +
  scale_fill_brewer(palette = "Set2", name = "Mutación general") +
  scale_y_continuous(expand = expansion(mult = c(0, 0.2))) +
  labs(
    x     = "Mutación General",
    y     = "Número de pacientes",
    title = "Presencia de mutación/traslocación general según combinación de batches de TCR"
  ) +
  coord_cartesian(clip = "off") +
  theme_minimal(base_size = 12) +
  theme(
    plot.title         = element_text(face = "bold", size = 9),
    strip.placement    = "outside",
    strip.text.y.left  = element_text(angle = 0, hjust = 1, face = "bold"),
    axis.text.x        = element_text(angle = 45, hjust = 1),
    panel.grid.major.y = element_line(color = "grey90"),
    panel.grid.major.x = element_blank(),
    plot.margin        = margin(t = 10, r = 10, b = 10, l = 40)
  )
```



Calculo de N por rangos de PD-L1 y combinación de batches

```
library(dplyr)
library(tidyr)
library(ggplot2)
library(RColorBrewer)

# 0) Recreamos flags y definimos el grupo de pareado
merged_full <- merged_full %>%
  mutate(
    has_bs1 = if_any(ends_with("_bs1_rel"), ~ !is.na(.)),
    has_bs2 = if_any(ends_with("_bs2_rel"), ~ !is.na(.)),
    has_bs3 = if_any(ends_with("_bs3_rel"), ~ !is.na(.)),
    PareadoGroup = case_when(
      has_bs1 & !has_bs2 & !has_bs3 ~ "bs1 solo",
      has_bs1 &  has_bs2 & !has_bs3 ~ "bs1 + bs2",
      has_bs1 &  has_bs2 &  has_bs3 ~ "bs1 + bs2 + bs3",
      has_bs1 & !has_bs2 &  has_bs3 ~ "bs1 + bs3",
      TRUE                          ~ NA_character_
    ),
    PareadoGroup = factor(
      PareadoGroup,
      levels = c("bs1 solo","bs1 + bs2","bs1 + bs2 + bs3","bs1 + bs3")
    )
  ) %>%
  filter(!is.na(PareadoGroup))
```

```r
# 1) Creamos categorías de PD.L1 en intervalos de 10%
breaks <- seq(0, 100, by = 10)
labels <- paste0(head(breaks, -1), "-", tail(breaks, -1), "%")
merged_full <- merged_full %>%
  mutate(
    PD.L1_cat = cut(
      PD.L1,
      breaks = breaks,
      labels = labels,
      include.lowest = TRUE,
      right = FALSE
    )
  )

# 2) Contamos pacientes por PareadoGroup × PD.L1_cat
pd_summary_pg <- merged_full %>%
  filter(!is.na(PD.L1_cat)) %>%
  count(PareadoGroup, PD.L1_cat) %>%
  complete(
    PareadoGroup,
    PD.L1_cat = labels,
    fill = list(n = 0)
  )

# 3) Gráfico facetado por PareadoGroup
ggplot(pd_summary_pg, aes(x = PD.L1_cat, y = n, fill = PD.L1_cat)) +
  geom_col(width = 0.8, show.legend = FALSE) +
  geom_text(aes(label = n), vjust = -0.3, size = 3) +
  facet_grid(
    PareadoGroup ~ .,
    switch = "y",
    drop   = FALSE
  ) +
  scale_fill_brewer(palette = "Set3") +
  scale_y_continuous(expand = expansion(mult = c(0, 0.2))) +
  labs(
    x     = "PD-L1 (%)",
    y     = "Número de pacientes",
    title = "Distribución de PD-L1 por rangos y combinación de batches de TCR"
  ) +
  coord_cartesian(clip = "off") +
  theme_minimal(base_size = 12) +
  theme(
    plot.title         = element_text(face = "bold", size = 8),
    strip.background   = element_rect(fill = "grey95", color = NA),
    strip.text         = element_text(face = "bold", size = 9),
    strip.placement    = "outside",
    strip.text.y.left  = element_text(angle = 0, hjust = 1),
    panel.grid.major.y = element_line(color = "grey90"),
    panel.grid.major.x = element_blank(),
    axis.text.x        = element_text(angle = 45, hjust = 1),
    plot.margin        = margin(t = 10, r = 10, b = 10, l = 40)
  )
```




```
# Eliminamos variables inutiles
merged_full$X <- NULL
merged_full$...1 <- NULL
```

```
write.csv(merged_full, file = "/home/agombau/modelo_pipeline/procesed_data/dataset_full_model.csv")
```