TFM

Análisis preliminar para el TFM de Helena Banus.

Objetivo:

En una submuestra de 280 personas del estudio PREDIMED-Plus con datos de isoformas de APOCIII, se busca determinar la asociación entre la concentración de APOCIII y sus isoformas con la incidencia de diabetes.

Objetivos secundarios:

- Asociar la concentración de APOCIII y sus isoformas con el perfil lipídico.
- Asociar la concentración de APOCIII y sus isoformas con la hemoglobina glicosilada, la insulina y la glucosa.
- Asociar la concentración de APOCIII y sus isoformas con la adherencia a la dieta mediterránea medida con el p17 y el perfil de ácidos grasos de los alimentos.

Bases de datos utilizadas:

- Para los datos generales del PREDIMED-Plus: PREDIMEDplus_2024_01_18.dta
- Para la incidencia de diabetes: ev_diab_2023-11-14.sav
- Para la insulina: PPLUS_PRIME_HOMA_1a_01092021.xlsx
- Para la concentración de APOCIII y las isoformas: Copia_Resultats_glicoformes_Tanda1_3.xlsx

Librerías utilizadas:

QC e intergración de las bases de datos:

Apertura de la BBDD:

```
PPlus <- read_dta("BBDD/PREDIMEDplus_2024_01_18.dta")
diab <- read_sav("BBDD/ev_diab_2023-11-14.sav")
insulina <- read_excel("BBDD/PPLUS_PRIME_HOMA_1a_01092021.xlsx")
isoformas <- read_excel("BBDD/Copia_Resultats_glicoformes_Tanda1_3.xlsx")
```

Integración de las bases de datos:

```
PPlus1 = PPlus[c("paciente", "idcluster", "grupo_int_v00", "nodo",
                 "sexo_s1", "escola_v00", "geaf_tot_v00", "fuma_s1", "imc_v00",
                 "glucosa_v00", "coltot_v00", "hdl_v00", "ldl_calc_v00",
                 "trigli_v00", "hba1c_v00", "tto_col_v00",
                 "hc_v00", "prot_v00", "gratot_v00", "mo_v00", "po_v00", "sa_v00",
                 "alcoholg_v00", "energiat_v00", "porc_hc_v00", "porc_pr_v00",
```

```
                  "porc_gr_v00", "porc_mo_v00", "porc_po_v00", "porc_sa_v00",
                  "fibra_v00", "col_v00", "p17_total_v00", "edad_s1")]
```

```
insulina1 = insulina[c("paciente", "insulin_v00", "HOMA.IR")]
```

```
diab1 = diab[c("paciente", "diabetes")]
```

```
colnames(isoformas) = c("IDURLA", "paciente", "Tanda", "ApoC3", "apoC30a_apoC31",
                        "apoC30b_apoC31", "apoC31d_apoC31", "apoC32d_apoC31",
                        "apoC32_apoC31", "apoC30f_apoC31")
```

Efecto Batch de isoformas:

Compruebo efecto batch en las isoformas:

```
pca_iso_pre = isoformas[3:10]
```

Para este análisis no me complico la vida e imputo a los participantes con NAs, pero habría que revisar variable por variable cuántos NAs y dependiendo del %, plantearse si usarla o no, y cómo usarla:

```
registerDoParallel(cores=8)
set.seed(1)

IMP = missForest(as.matrix(pca_iso_pre), verbose = T, parallelize = "forest")
```

```
  parallelizing computation of the random forest model objects
  missForest iteration 1 in progress...done!
    estimated error(s): 0.2952273
    difference(s): 1.889283e-05
    time: 0.25 seconds

  missForest iteration 2 in progress...done!
    estimated error(s): 0.3027388
    difference(s): 9.607022e-05
    time: 0.44 seconds
```

```
IMP$OOBerror
```

```
    NRMSE
0.2952273
```

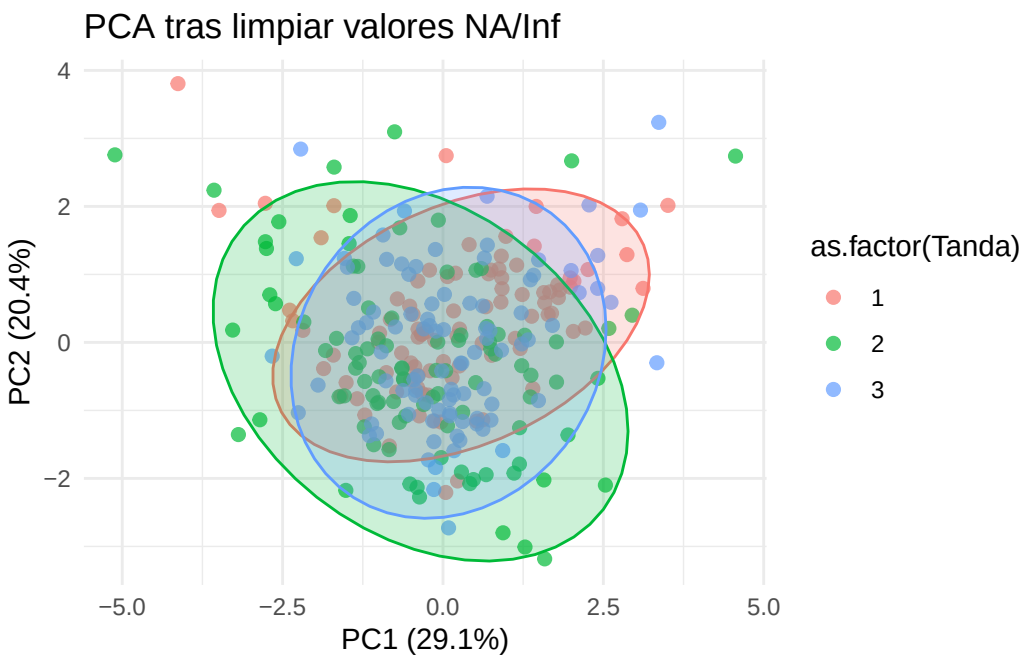
```
IMP = data.frame(IMP$ximp)
```

Genero z-scores y hago PCA:

```
IMP[2:8] = scale(IMP[2:8], center = T, scale = T)
res_pca <- prcomp(IMP[2:8], center = FALSE, scale. = FALSE)
```

```
scores <- data.frame(cbind(res_pca$x, Tanda = IMP$Tanda))

var_exp <- res_pca$sdev^2 / sum(res_pca$sdev^2) * 100

ggplot(scores, aes(PC1, PC2, color = as.factor(Tanda))) +
  geom_point(alpha = 0.7, size = 2) +
  stat_ellipse(aes(fill = as.factor(Tanda)), alpha = 0.2, geom = "polygon", show.legend = FALSE) +
  labs(x = paste0("PC1 (", round(var_exp[1], 1), "%)"),
       y = paste0("PC2 (", round(var_exp[2], 1), "%)"),
       title = "PCA tras limpiar valores NA/Inf") +
```

```
  theme_minimal()
```



No parece haber efecto batch visual (good job)

```
dist_IMP = dist(IMP[2:8])
adonis2(dist_IMP ~ Tanda, data = IMP, permutations = 999)
```

```
Permutation test for adonis under reduced model
Permutation: free
Number of permutations: 999

adonis2(formula = dist_IMP ~ Tanda, data = IMP, permutations = 999)
          Df SumOfSqs      R2      F Pr(>F)
Model      1    11.15 0.00567 1.5958  0.141
Residual 280  1955.85 0.99433
Total    281  1967.00 1.00000
```

```
dist_APOC3 = dist(IMP[2])
adonis2(dist_APOC3 ~ Tanda, data = IMP, permutations = 999)
```

```
Permutation test for adonis under reduced model
Permutation: free
Number of permutations: 999

adonis2(formula = dist_APOC3 ~ Tanda, data = IMP, permutations = 999)
          Df SumOfSqs      R2      F Pr(>F)
Model      1    1.836 0.00653 1.8414   0.17
Residual 280  279.164 0.99347
Total    281  281.000 1.00000
```

Confirmo que estadísticamente no hay efecto batch. Creo la base de datos de trabajo:

```
isoformas1 = data.frame(c(isoformas[2], IMP))
```

```
BBDD = merge(PPlus1, isoformas1, by = "paciente")
BBDD = merge(BBDD, insulina1, by = "paciente", all.x = T)
BBDD = merge(BBDD, diab1, by = "paciente", all.x = T)
```

QC:

```
colnames(BBDD)
```

```
 [1] "paciente"       "idcluster"      "grupo_int_v00"  "nodo"
 [5] "sexo_s1"        "escola_v00"     "geaf_tot_v00"   "fuma_s1"
 [9] "imc_v00"        "glucosa_v00"    "coltot_v00"     "hdl_v00"
[13] "ldl_calc_v00"   "trigli_v00"     "hba1c_v00"      "tto_col_v00"
[17] "hc_v00"         "prot_v00"       "gratot_v00"     "mo_v00"
[21] "po_v00"         "sa_v00"         "alcoholg_v00"   "energiat_v00"
[25] "porc_hc_v00"    "porc_pr_v00"    "porc_gr_v00"    "porc_mo_v00"
[29] "porc_po_v00"    "porc_sa_v00"    "fibra_v00"      "col_v00"
[33] "p17_total_v00"  "edad_s1"        "Tanda"          "ApoC3"
[37] "apoC30a_apoC31" "apoC30b_apoC31" "apoC31d_apoC31" "apoC32d_apoC31"
[41] "apoC32_apoC31"  "apoC30f_apoC31" "insulin_v00"    "HOMA.IR"
[45] "diabetes"
```

```
BBDD$diabetes[is.na(BBDD$diabetes)] <- 0
factores = c(3, 4, 5, 6, 8, 16, 35, 45)
BBDD[, factores] = lapply(BBDD[, factores], factor)
summary(BBDD)
```

```
   paciente       idcluster     grupo_int_v00      nodo    sexo_s1 escola_v00
Min.   : 1024   Min.   : 1024   0:157         4      :67   0:146   1: 39
1st Qu.: 3071   1st Qu.: 3138   1:125         5      :48   1:136   2: 17
Median : 4226   Median : 4265                 3      :47           3: 94
Mean   : 4726   Mean   : 6957                 1      :35           4:130
3rd Qu.: 5294   3rd Qu.: 6074                 6      :31           5:  2
Max.   :19020   Max.   :42303                 2      :29
                                              (Other):25
 geaf_tot_v00     fuma_s1    imc_v00       glucosa_v00      coltot_v00
Min.   :    0.0   1: 35   Min.   :26.82   Min.   : 76.0   Min.   :108.0
1st Qu.:  839.2   2: 10   1st Qu.:30.71   1st Qu.:104.0   1st Qu.:177.0
Median : 2107.2   3: 17   Median :32.71   Median :112.0   Median :200.0
Mean   : 2763.3   4:103   Mean   :33.19   Mean   :113.5   Mean   :202.6
3rd Qu.: 4010.5   5:115   3rd Qu.:35.53   3rd Qu.:121.8   3rd Qu.:224.0
Max.   :14895.1   9:  2   Max.   :41.88   Max.   :149.0   Max.   :311.0
                                                          NA's   :1
   hdl_v00       ldl_calc_v00     trigli_v00      hba1c_v00     tto_col_v00
Min.   :28.00   Min.   : 39.0   Min.   : 54.0   Min.   :4.000   0:145
1st Qu.:39.00   1st Qu.:105.5   1st Qu.:101.0   1st Qu.:5.660   1:136
Median :47.00   Median :123.0   Median :132.5   Median :5.900   9:  1
Mean   :48.45   Mean   :125.2   Mean   :146.4   Mean   :5.917
3rd Qu.:55.00   3rd Qu.:144.0   3rd Qu.:173.0   3rd Qu.:6.200
Max.   :91.00   Max.   :232.0   Max.   :502.0   Max.   :7.600
NA's   :1       NA's   :11                      NA's   :22
    hc_v00          prot_v00        gratot_v00         mo_v00
Min.   : 95.25   Min.   : 29.50   Min.   : 36.64   Min.   : 16.66
1st Qu.:182.55   1st Qu.: 81.16   1st Qu.: 85.75   1st Qu.: 42.44
Median :231.17   Median : 94.86   Median :106.56   Median : 55.60
```

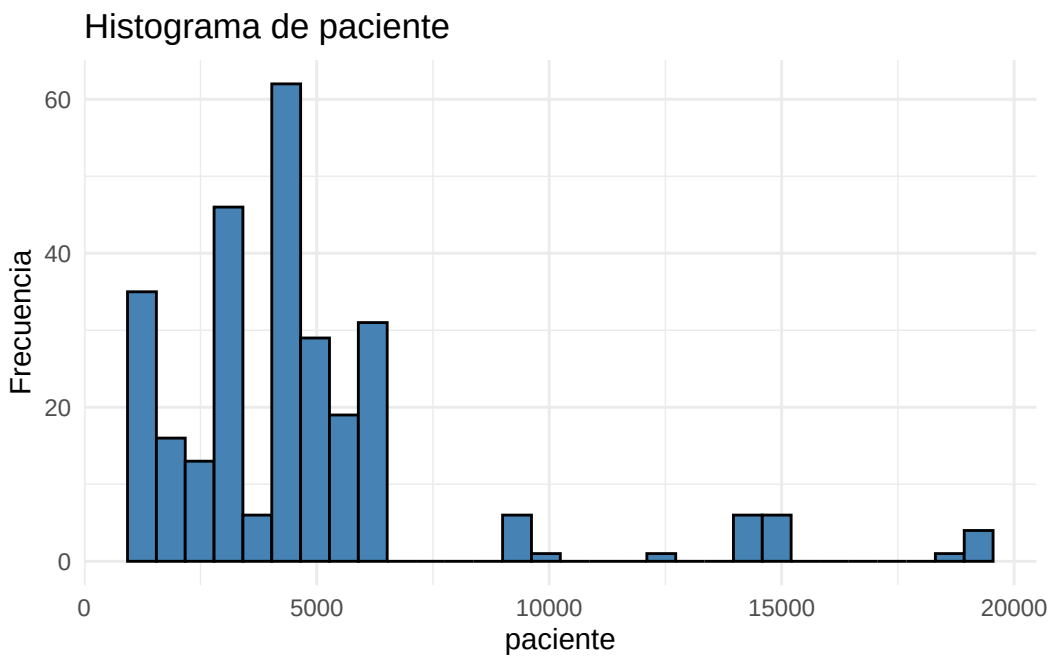
```
Mean   :242.89   Mean   : 96.37   Mean   :107.96   Mean   : 55.33
3rd Qu.:285.75   3rd Qu.:111.86   3rd Qu.:128.15   3rd Qu.: 66.86
Max.   :600.92   Max.   :197.42   Max.   :210.80   Max.   :109.87
NA's   :1        NA's   :1        NA's   :1        NA's   :1
    po_v00           sa_v00        alcoholg_v00      energiat_v00
Min.   : 5.291   Min.   : 9.936   Min.   :  0.000   Min.   :1051
1st Qu.:13.334   1st Qu.:20.275   1st Qu.:  1.292   1st Qu.:1926
Median :17.193   Median :26.032   Median :  5.769   Median :2395
Mean   :18.317   Mean   :27.854   Mean   : 12.507   Mean   :2416
3rd Qu.:22.252   3rd Qu.:34.189   3rd Qu.: 18.565   3rd Qu.:2817
Max.   :48.543   Max.   :82.991   Max.   :103.150   Max.   :5021
NA's   :1        NA's   :1        NA's   :1         NA's   :1
 porc_hc_v00     porc_pr_v00      porc_gr_v00     porc_mo_v00
Min.   :21.27   Min.   : 9.266   Min.   :22.82   Min.   : 8.456
1st Qu.:35.61   1st Qu.:14.260   1st Qu.:35.69   1st Qu.:18.043
Median :40.25   Median :16.068   Median :40.72   Median :20.438
Mean   :40.04   Mean   :16.305   Mean   :40.17   Mean   :20.651
3rd Qu.:44.47   3rd Qu.:18.305   3rd Qu.:44.21   3rd Qu.:23.040
Max.   :55.93   Max.   :27.018   Max.   :56.32   Max.   :32.873
NA's   :1       NA's   :1        NA's   :1       NA's   :1
 porc_po_v00      porc_sa_v00       fibra_v00         col_v00
Min.   : 3.732   Min.   : 4.990   Min.   : 7.154   Min.   :106.4
1st Qu.: 5.513   1st Qu.: 8.756   1st Qu.:18.772   1st Qu.:316.6
Median : 6.465   Median :10.190   Median :23.991   Median :371.0
Mean   : 6.805   Mean   :10.273   Mean   :25.992   Mean   :387.1
3rd Qu.: 7.740   3rd Qu.:11.675   3rd Qu.:32.024   3rd Qu.:464.8
Max.   :16.776   Max.   :16.116   Max.   :79.543   Max.   :782.1
NA's   :1        NA's   :1        NA's   :1        NA's   :1
p17_total_v00      edad_s1      Tanda      ApoC3          apoC30a_apoC31
Min.   : 1.00   Min.   :55.00   1:94   Min.   :-2.1243   Min.   :-1.2861
1st Qu.: 6.00   1st Qu.:60.00   2:94   1st Qu.:-0.6446   1st Qu.:-0.5877
Median : 8.00   Median :65.00   3:94   Median :-0.1518   Median :-0.1516
Mean   : 7.94   Mean   :64.62          Mean   : 0.0000   Mean   : 0.0000
3rd Qu.:10.00   3rd Qu.:68.75          3rd Qu.: 0.5597   3rd Qu.: 0.2729
Max.   :16.00   Max.   :75.00          Max.   : 4.1273   Max.   : 9.5285

apoC30b_apoC31    apoC31d_apoC31    apoC32d_apoC31    apoC32_apoC31
Min.   :-2.3348   Min.   :-2.0690   Min.   :-2.0104   Min.   :-1.7414
1st Qu.:-0.6727   1st Qu.:-0.4713   1st Qu.:-0.7940   1st Qu.:-0.7197
Median :-0.1045   Median :-0.1227   Median :-0.1800   Median :-0.2229
Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000
3rd Qu.: 0.5635   3rd Qu.: 0.3468   3rd Qu.: 0.6735   3rd Qu.: 0.5709
Max.   : 3.9551   Max.   : 6.8186   Max.   : 5.0159   Max.   : 4.0699

apoC30f_apoC31     insulin_v00       HOMA.IR        diabetes
Min.   :-1.6246   Min.   : 2.80   Min.   : 0.6983   0:143
1st Qu.:-0.7984   1st Qu.:14.10   1st Qu.: 3.7956   1:139
Median :-0.2389   Median :19.30   Median : 5.2385
Mean   : 0.0000   Mean   :21.97   Mean   : 6.2566
3rd Qu.: 0.6490   3rd Qu.:26.40   3rd Qu.: 7.6042
Max.   : 3.7184   Max.   :93.60   Max.   :32.9985
                  NA's   :5       NA's   :5
```

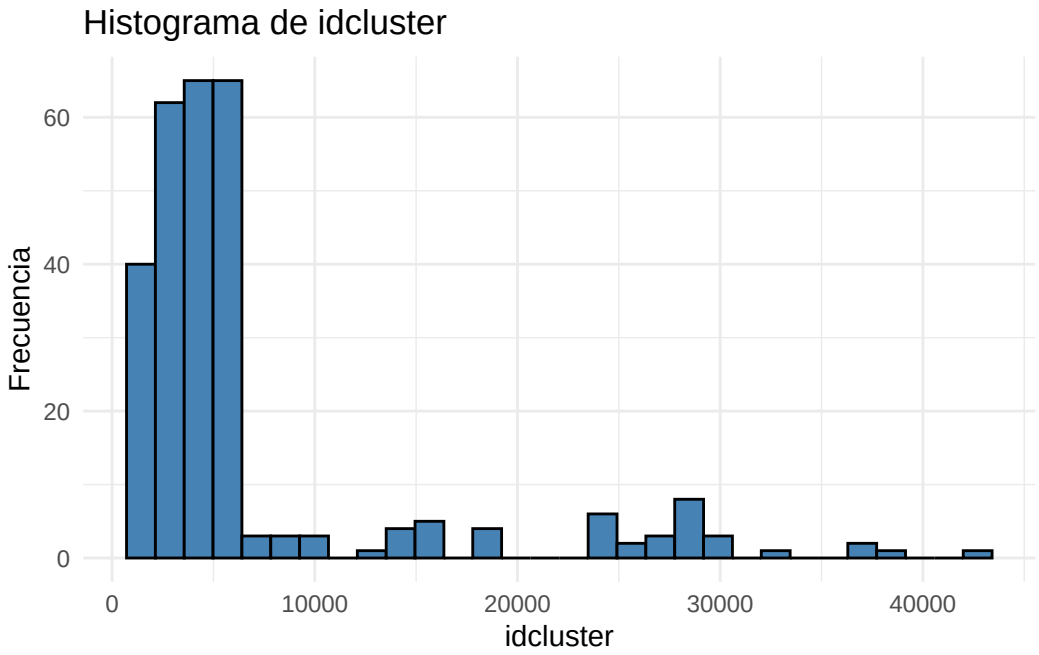
```
vars_numéricas <- BBDD %>%
```

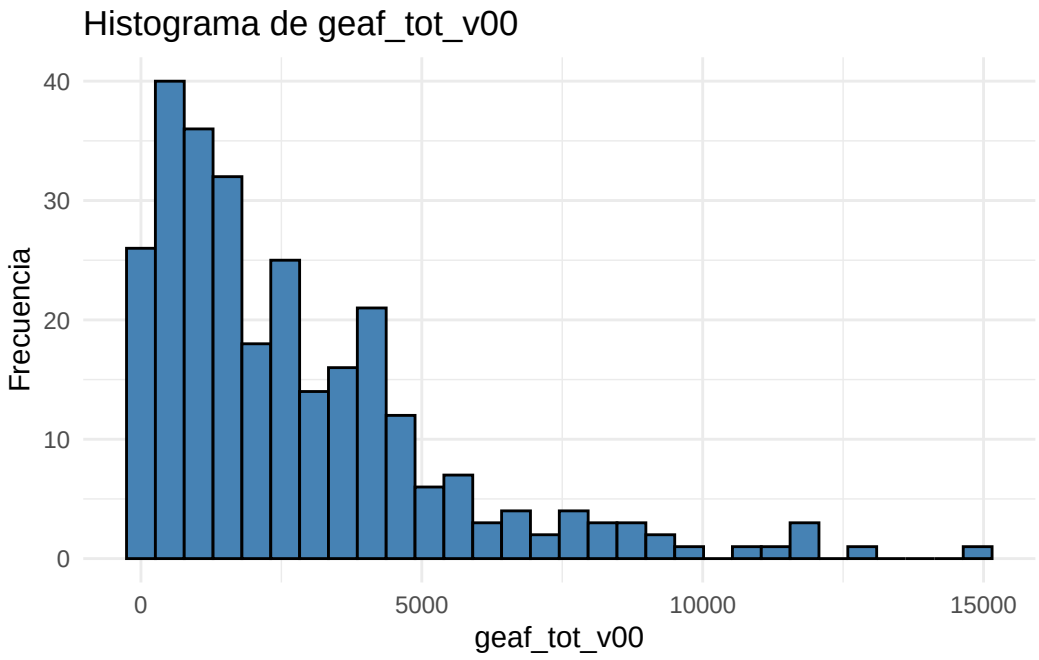
```
  dplyr::select(where(is.numeric)) %>%
  names()

crear_histograma <- function(var) {
  ggplot(BBDD, aes_string(x = var)) +
    geom_histogram(bins = 30, color = "black", fill = "steelblue") +
    labs(title = paste("Histograma de", var), x = var, y = "Frecuencia") +
    theme_minimal()
}

lista_plots <- map(vars_numéricas, crear_histograma)
```

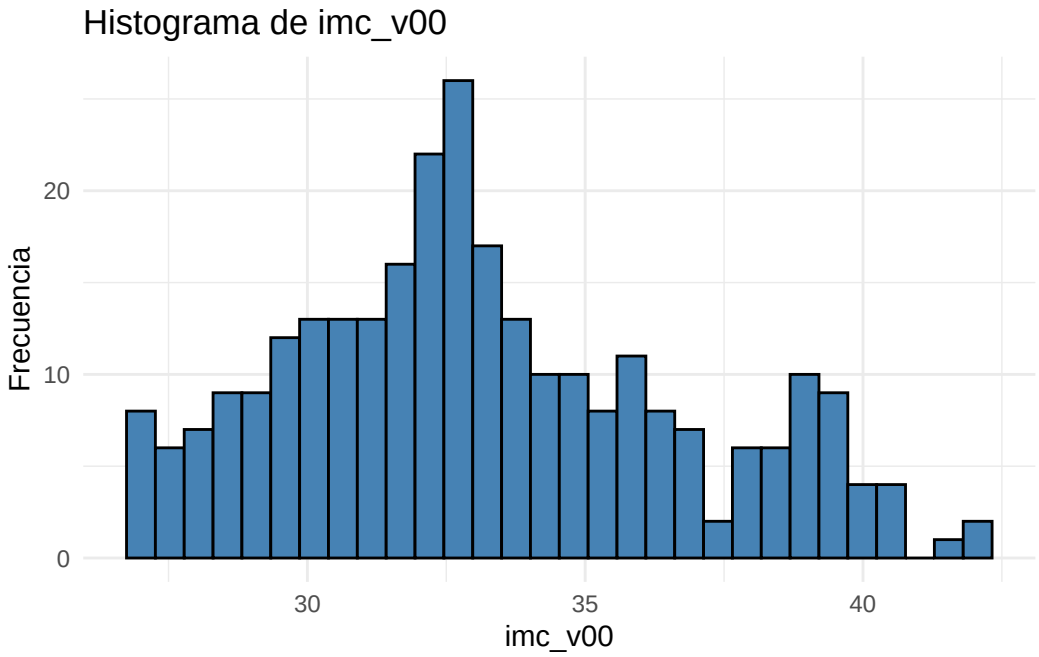
```
Warning: `aes_string()` was deprecated in ggplot2 3.0.0.
i Please use tidy evaluation idioms with `aes()`.
i See also `vignette("ggplot2-in-packages")` for more information.
```

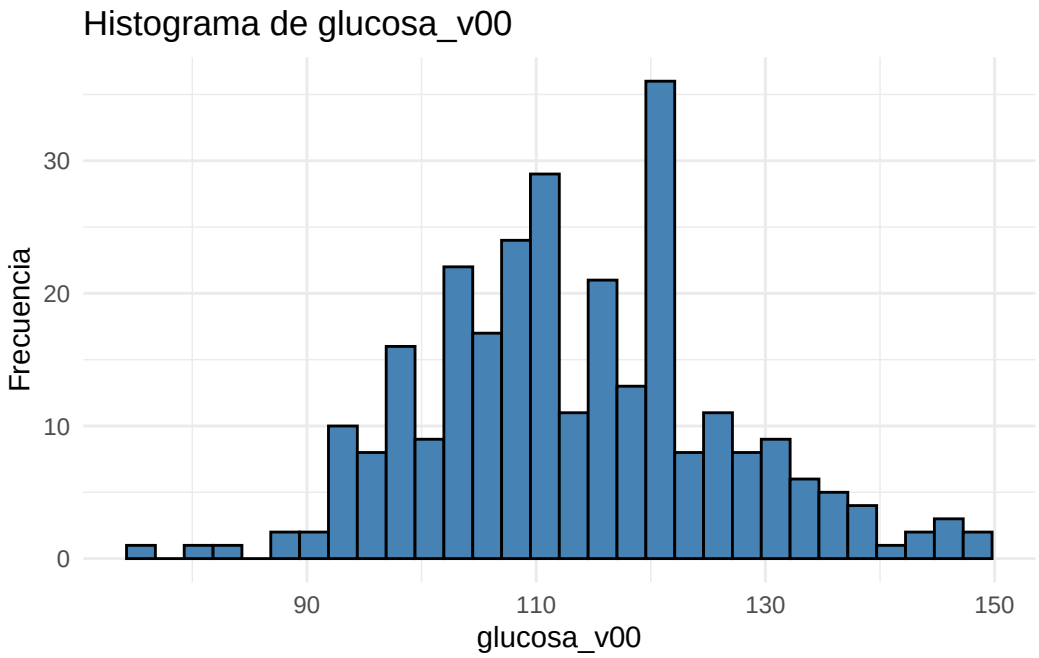
```
walk(lista_plots, print)
```



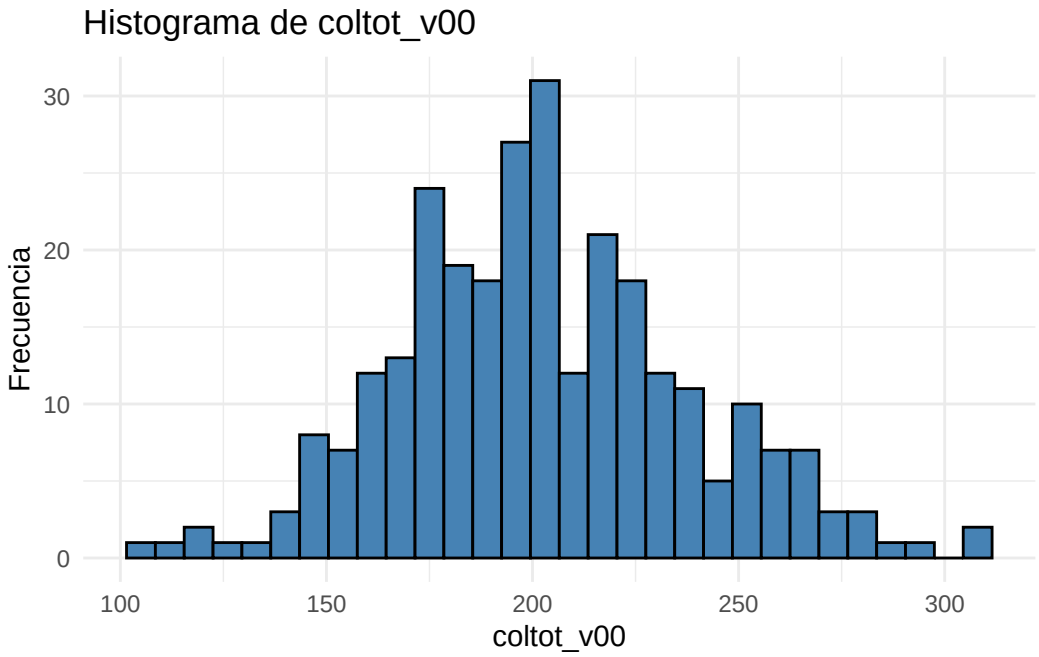

Histograma de idcluster
Frecuencia
60
40
20
0
0
10000
20000
30000
40000
idcluster


Histograma de geaf_tot_v00
Frecuencia
40
30
20
10
0
0
5000
10000
15000
geaf_tot_v00

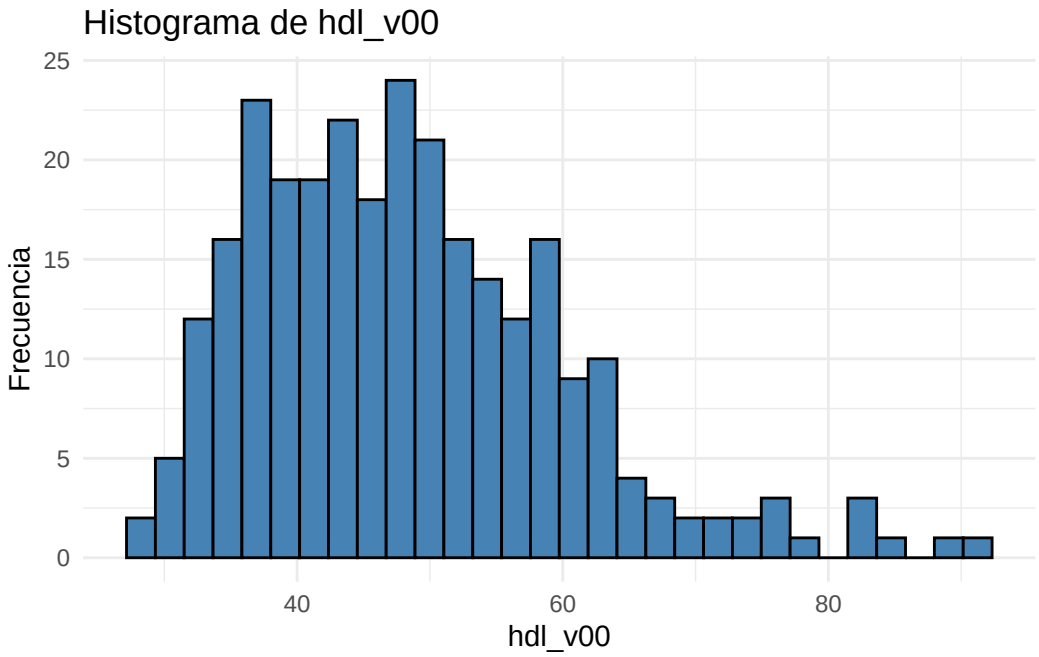




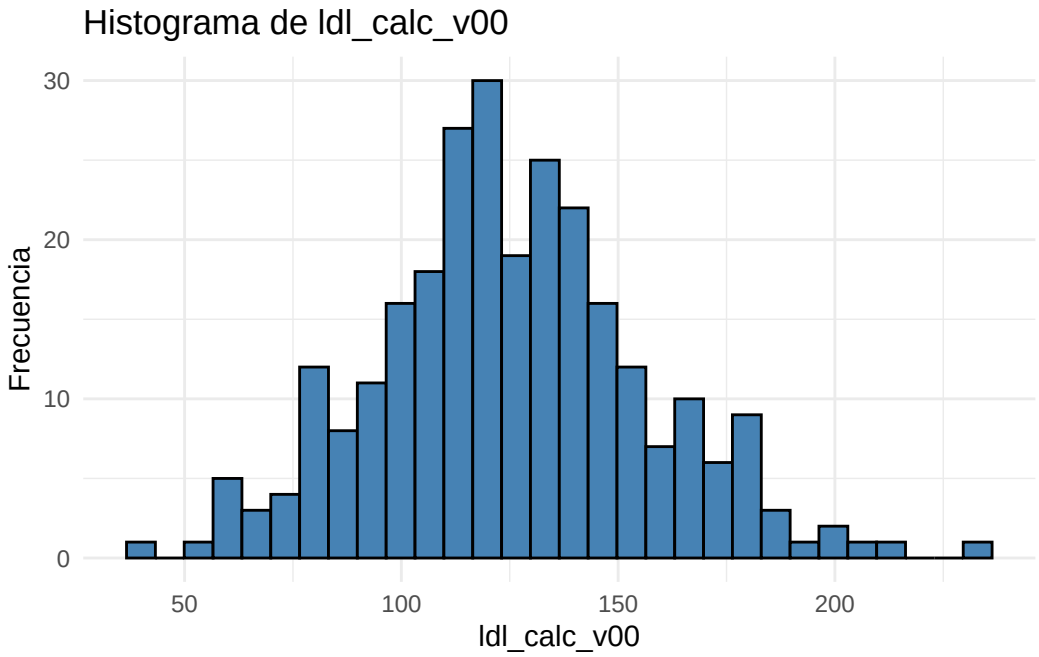

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```

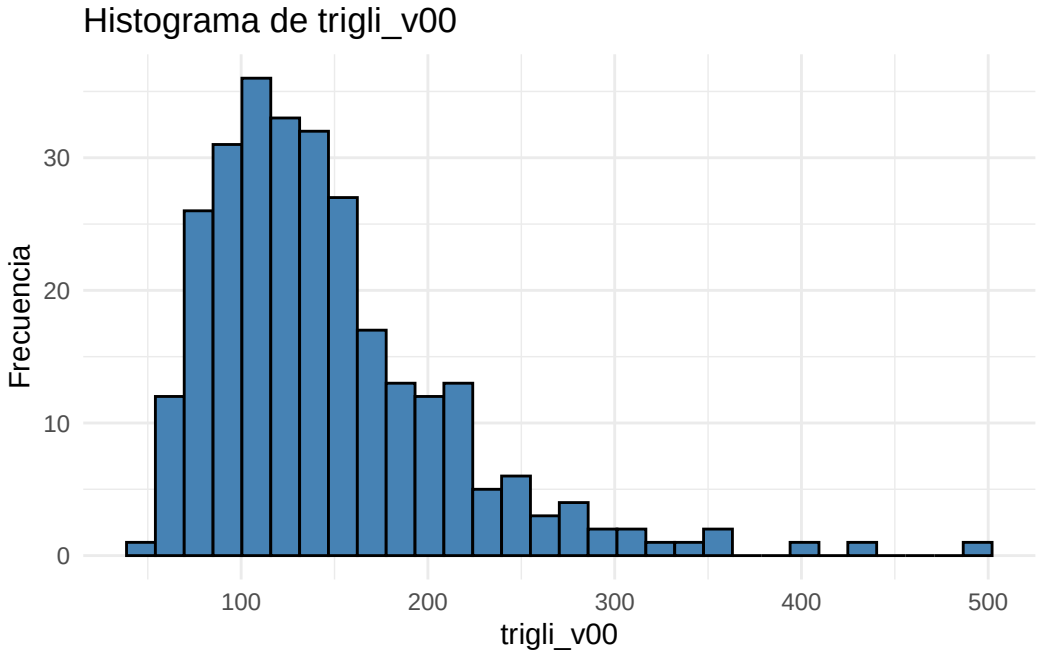



```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```

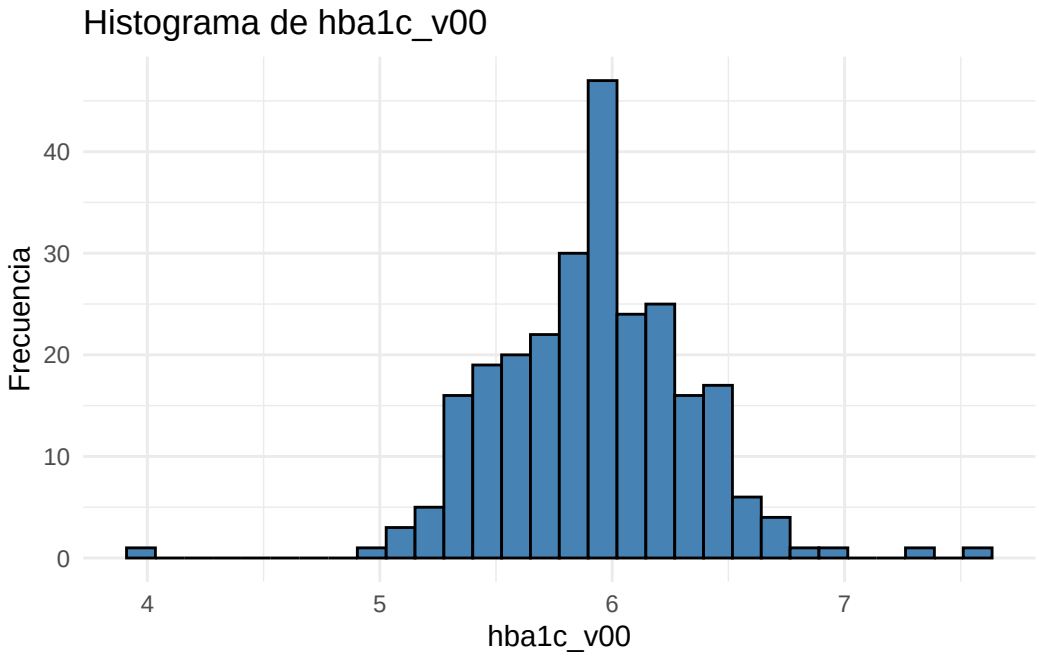



```
Warning: Removed 11 rows containing non-finite outside the scale range
(`stat_bin()`).
```

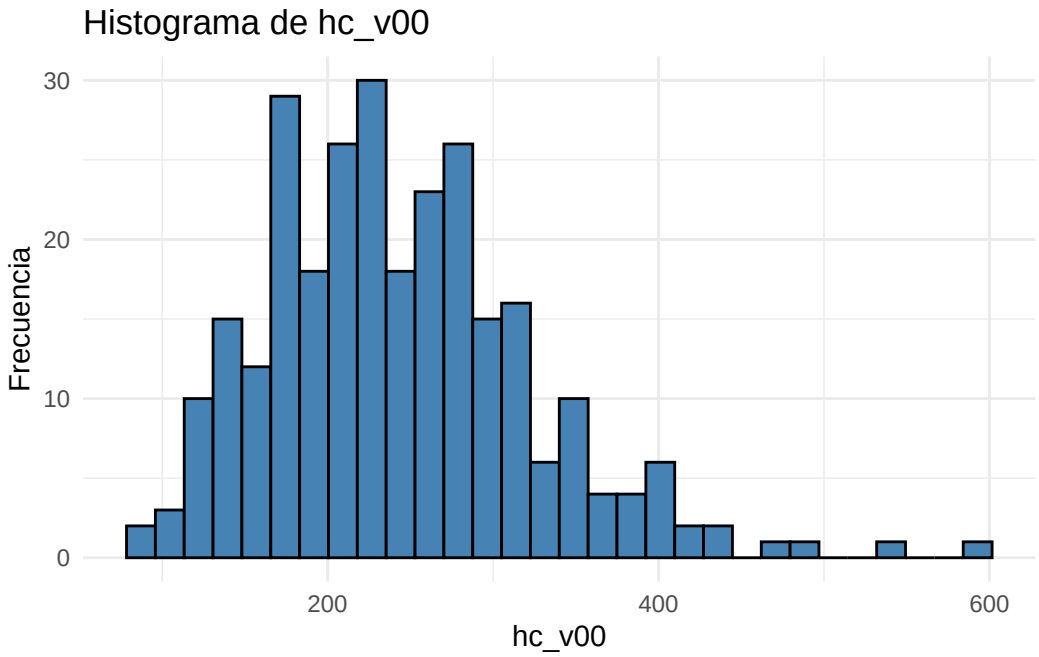




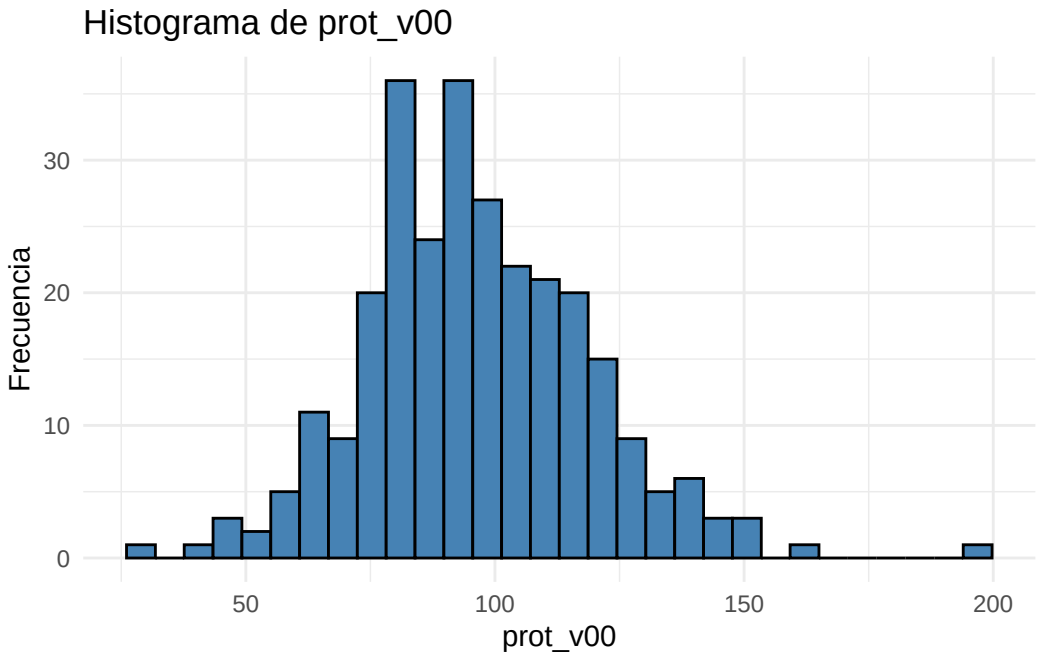

```
Warning: Removed 22 rows containing non-finite outside the scale range
(`stat_bin()`).
```



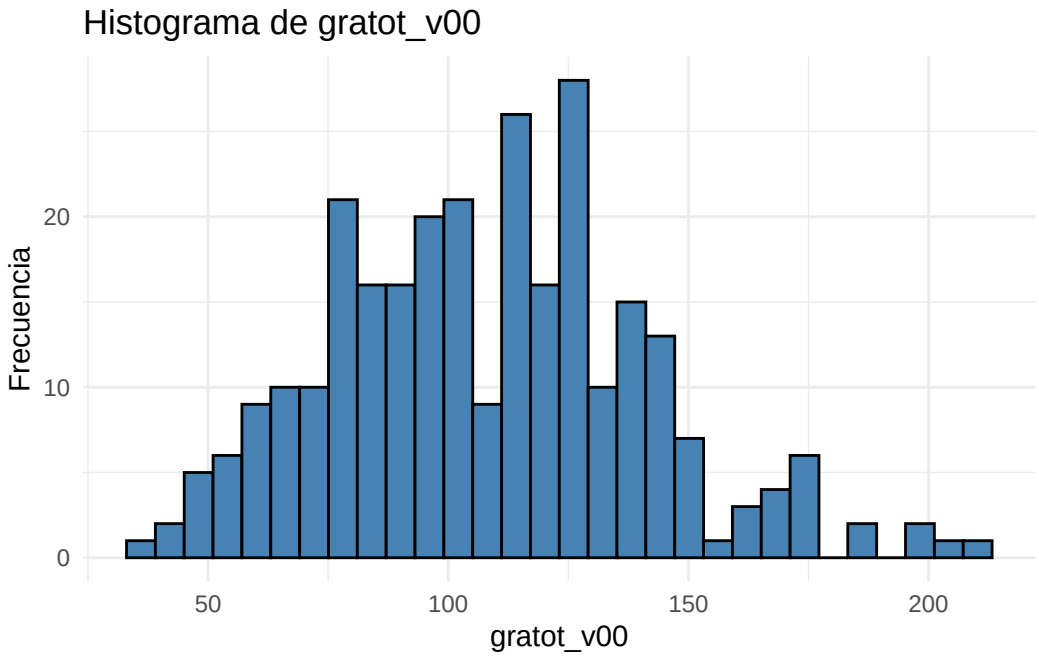

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```



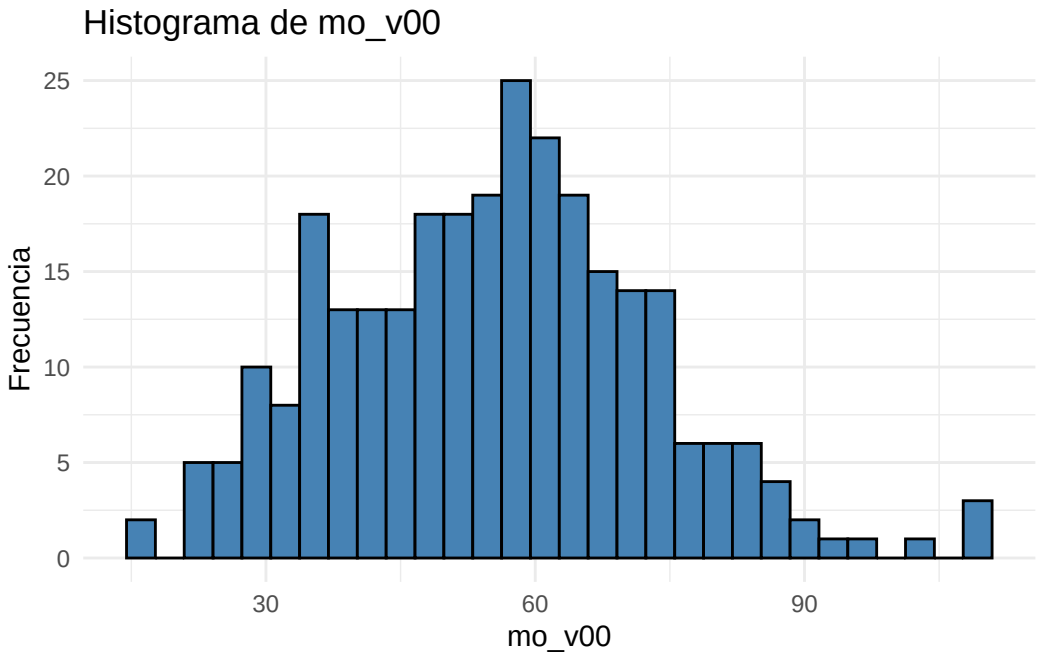

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```



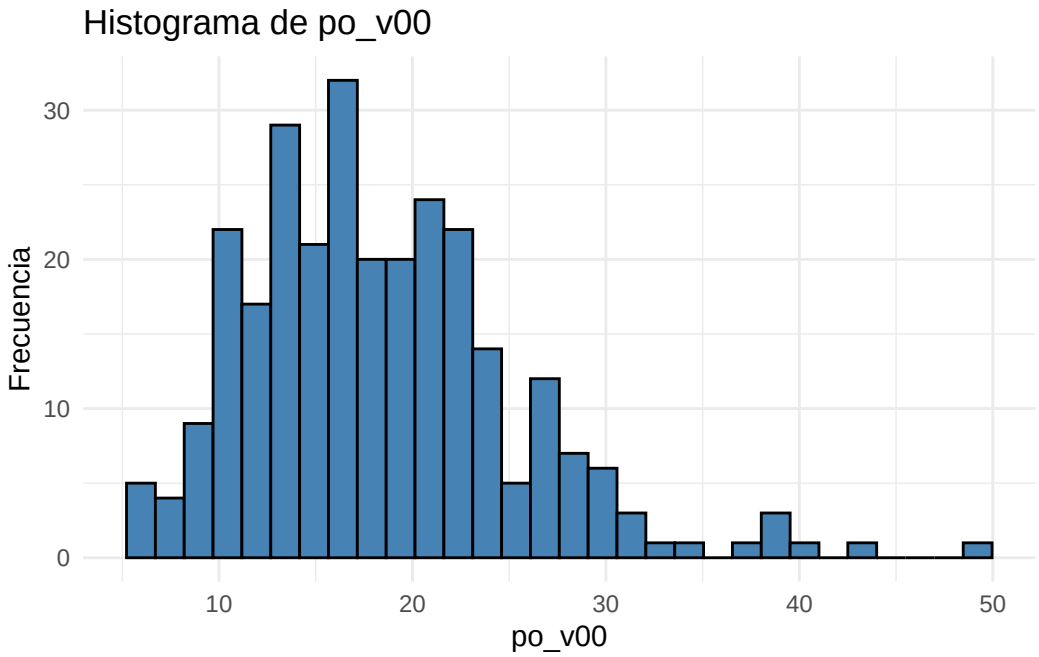

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```



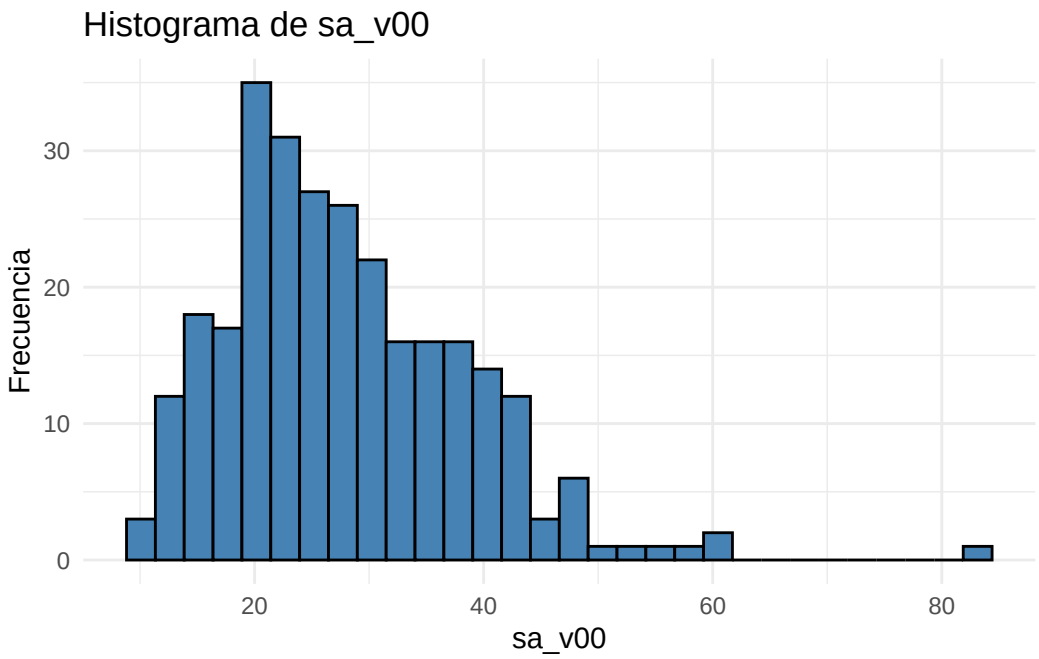

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```



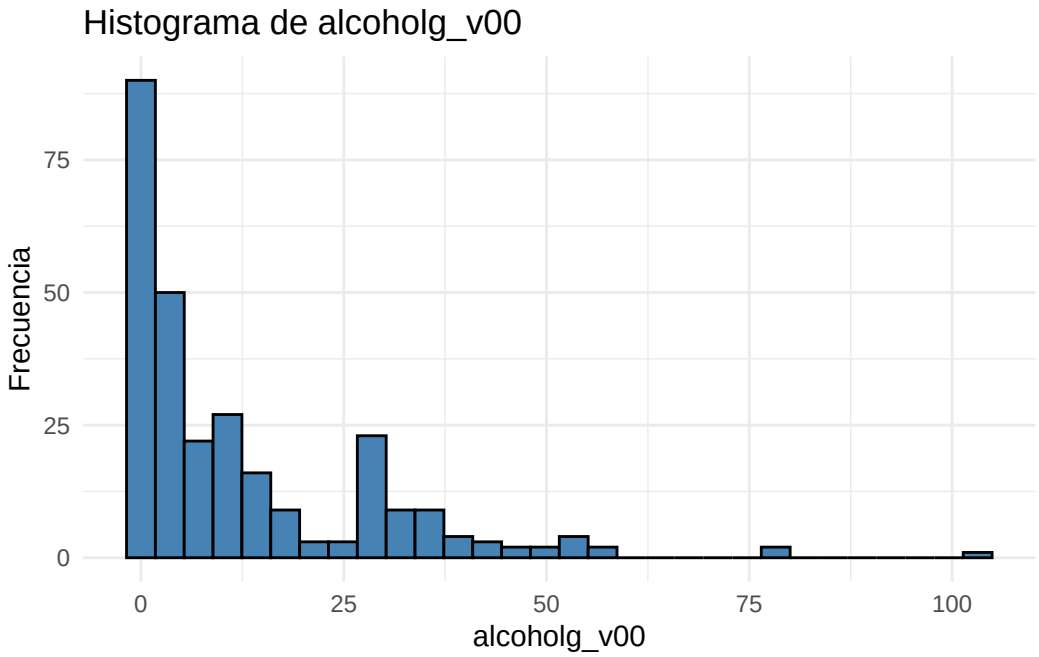

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```



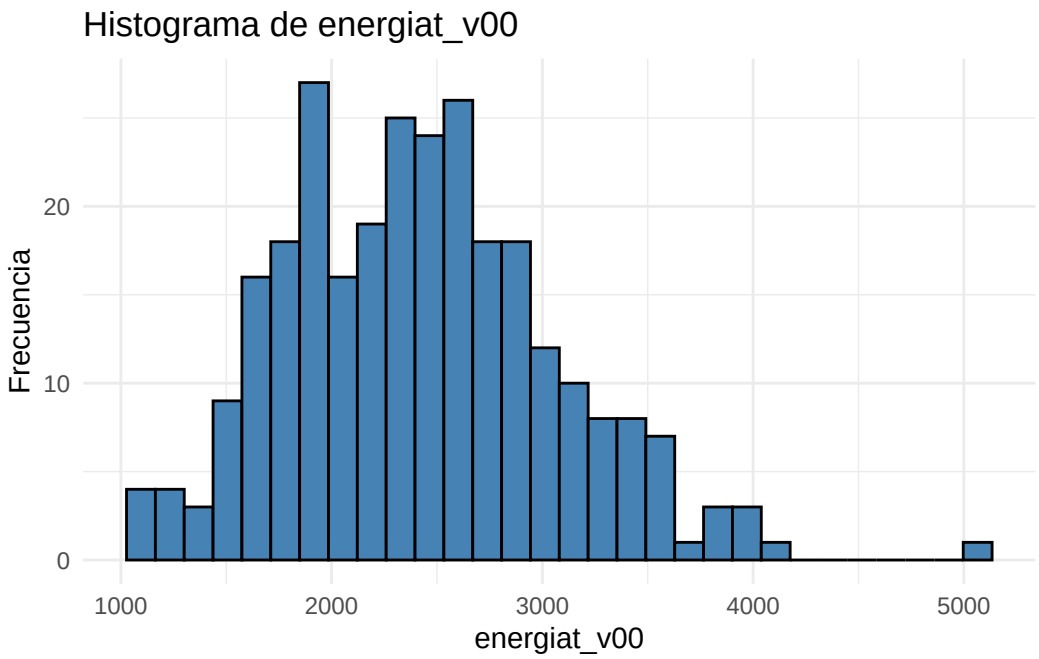

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```



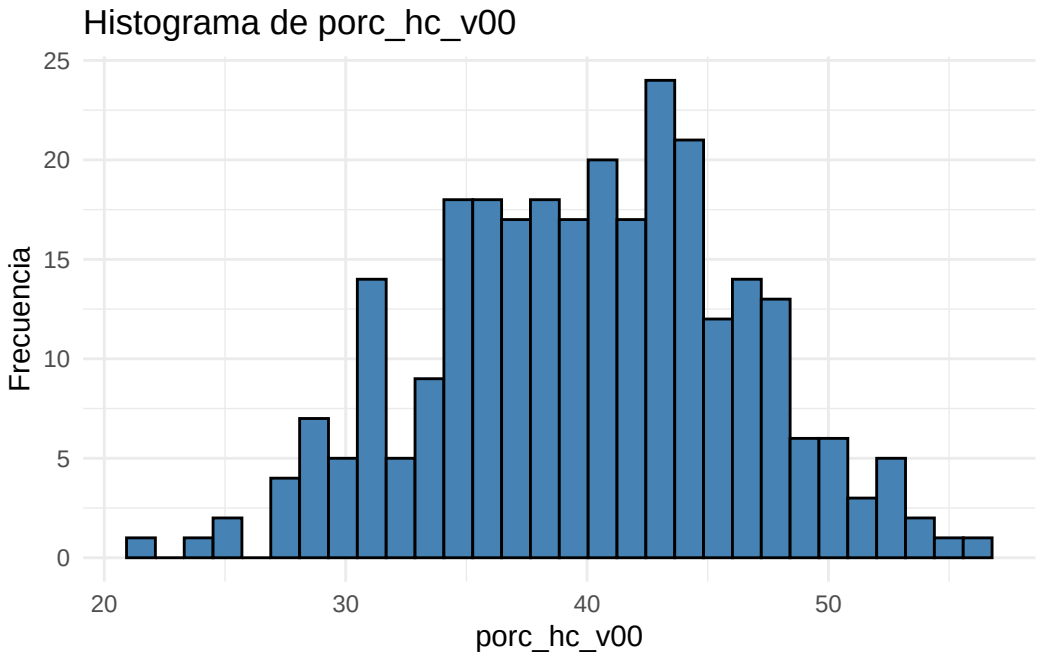

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```



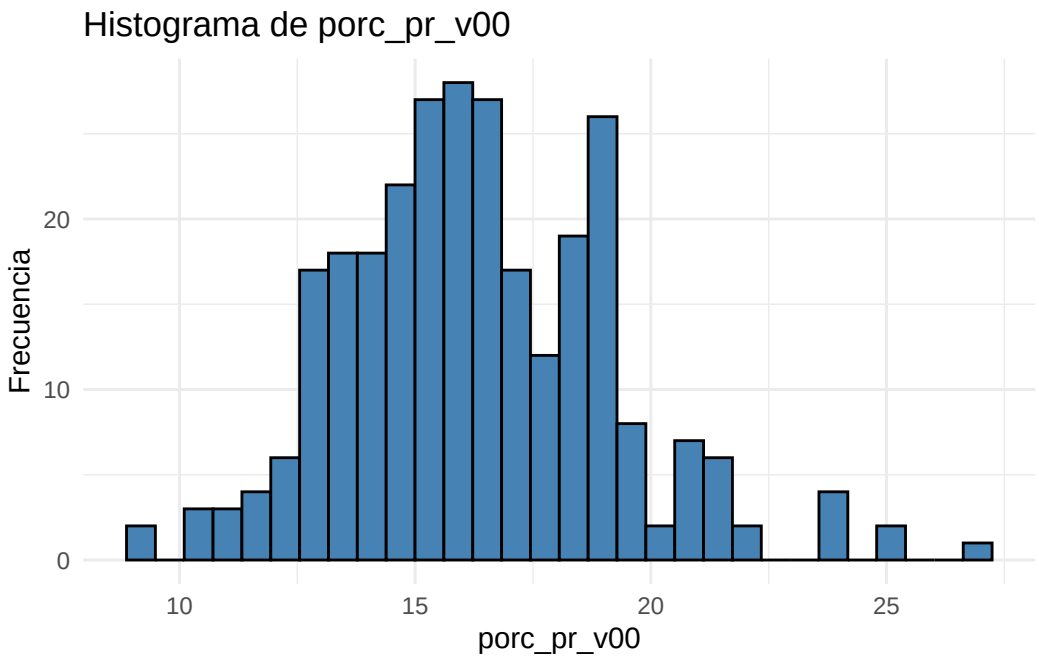

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```



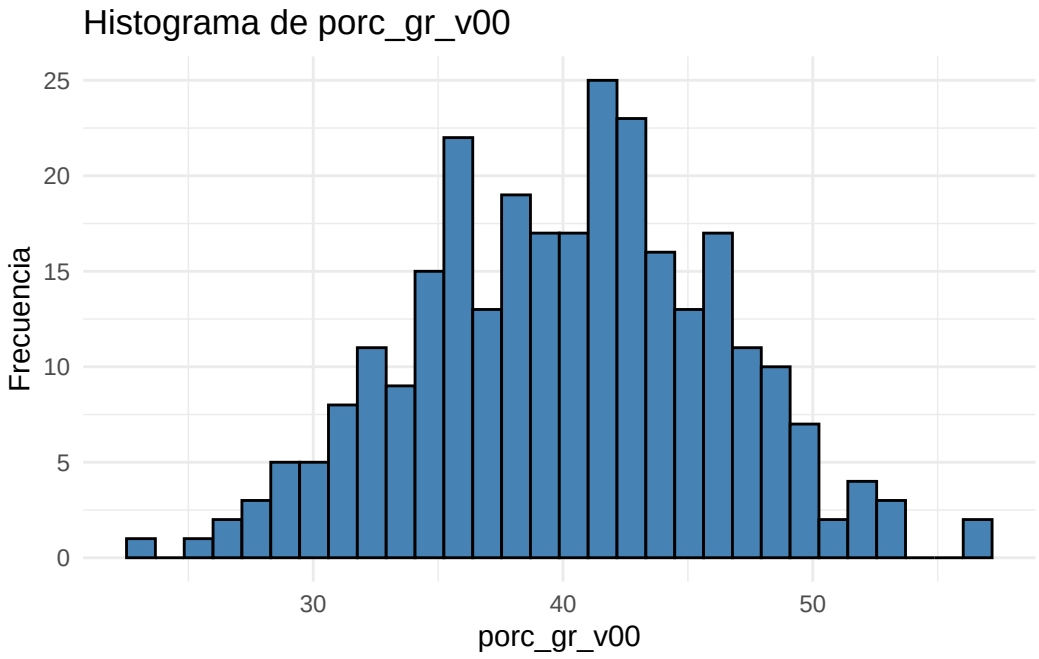

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```



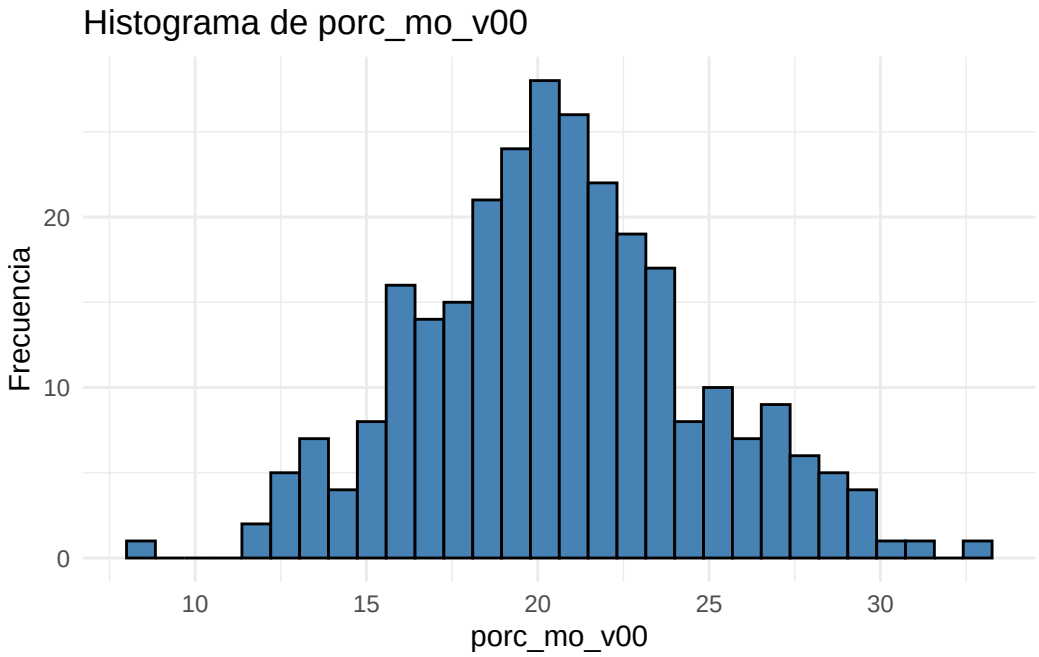

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```



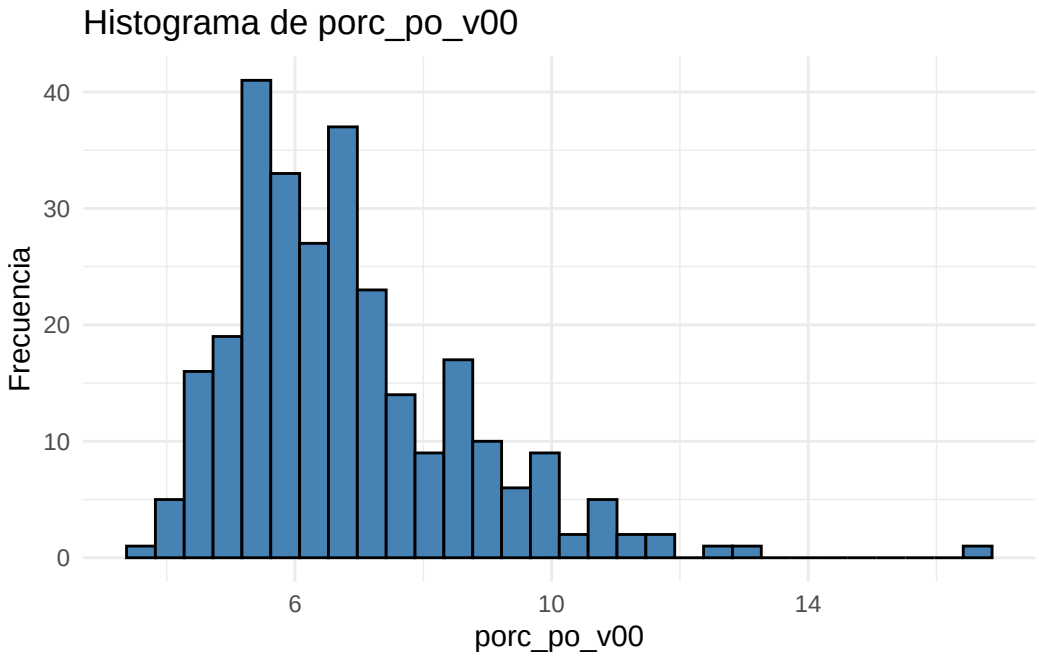

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```



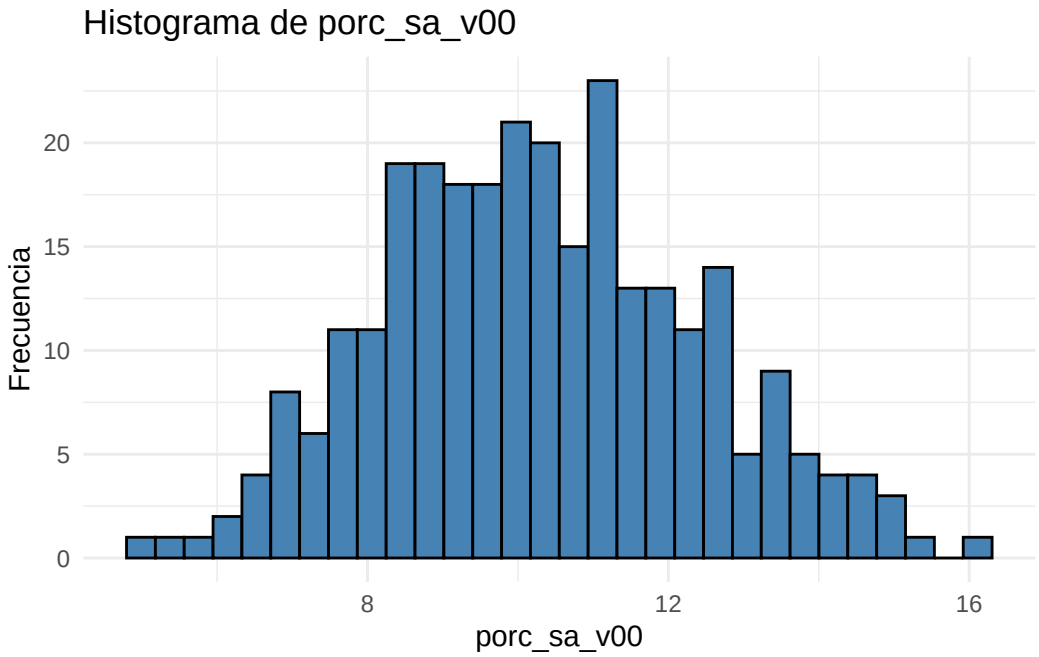

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```



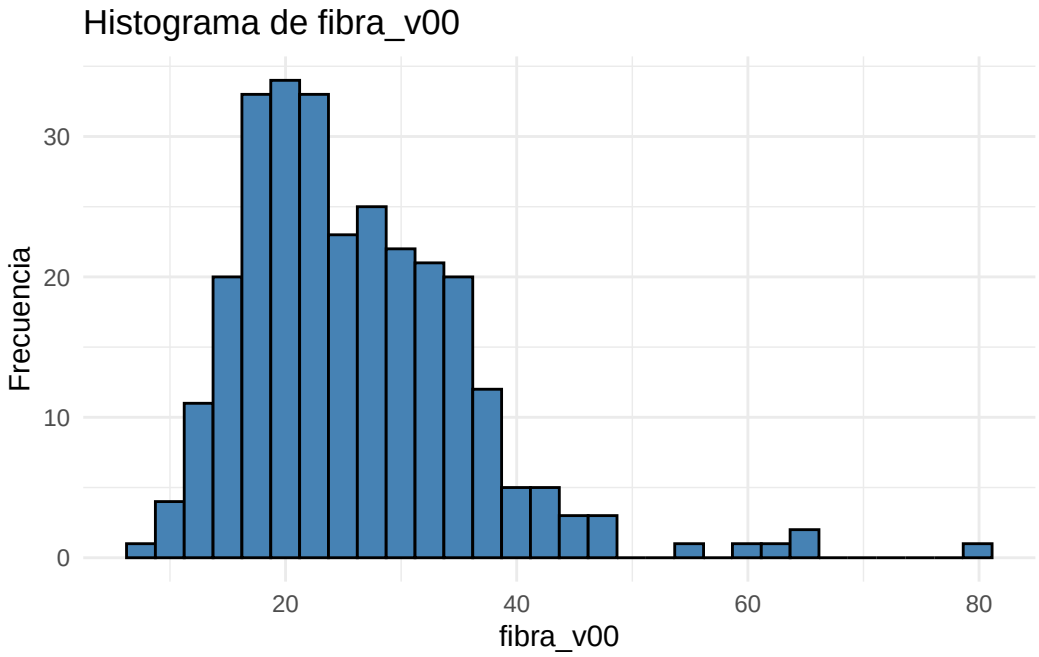

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```



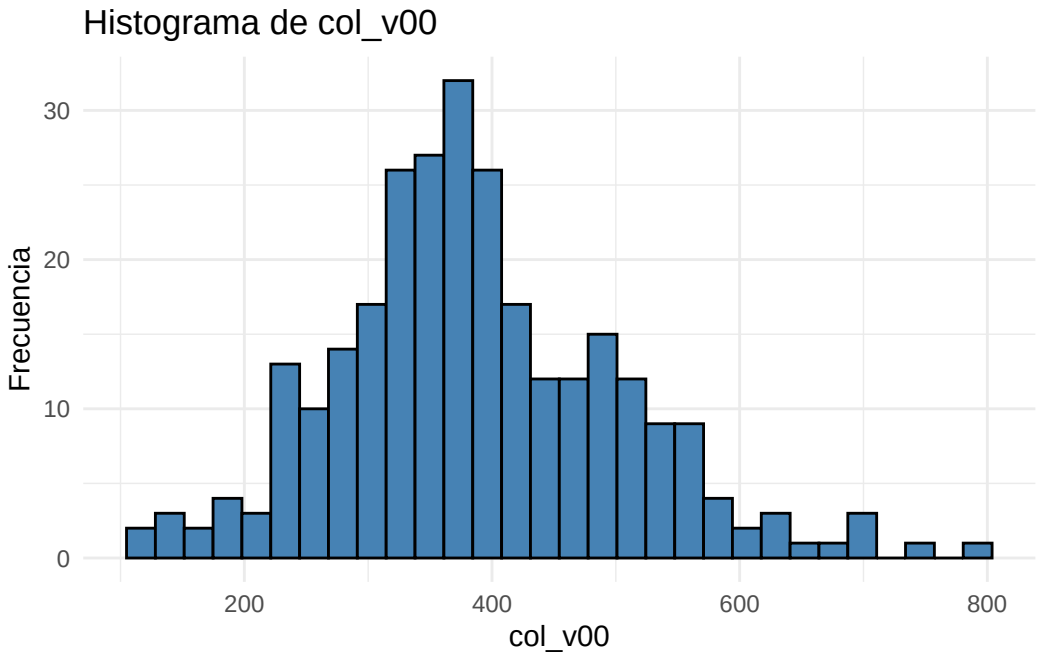

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```

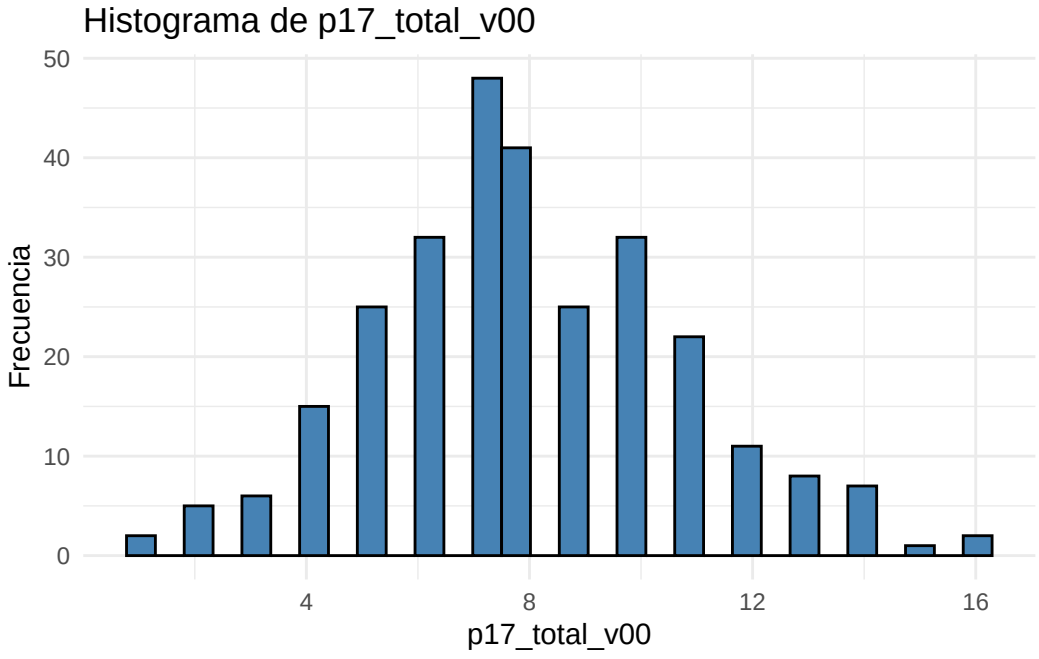


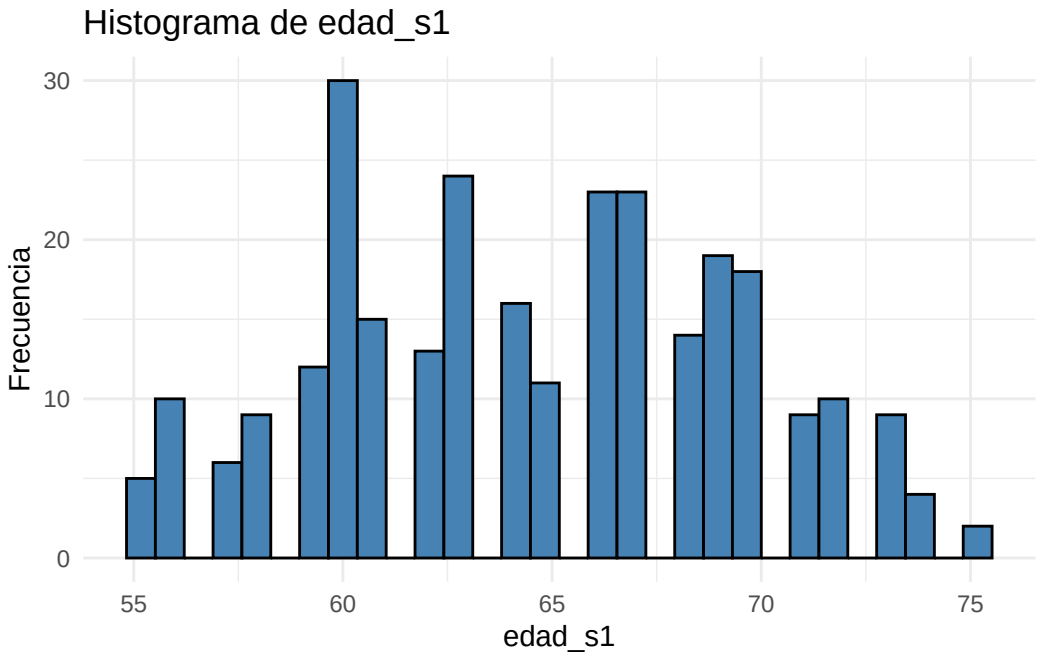

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```

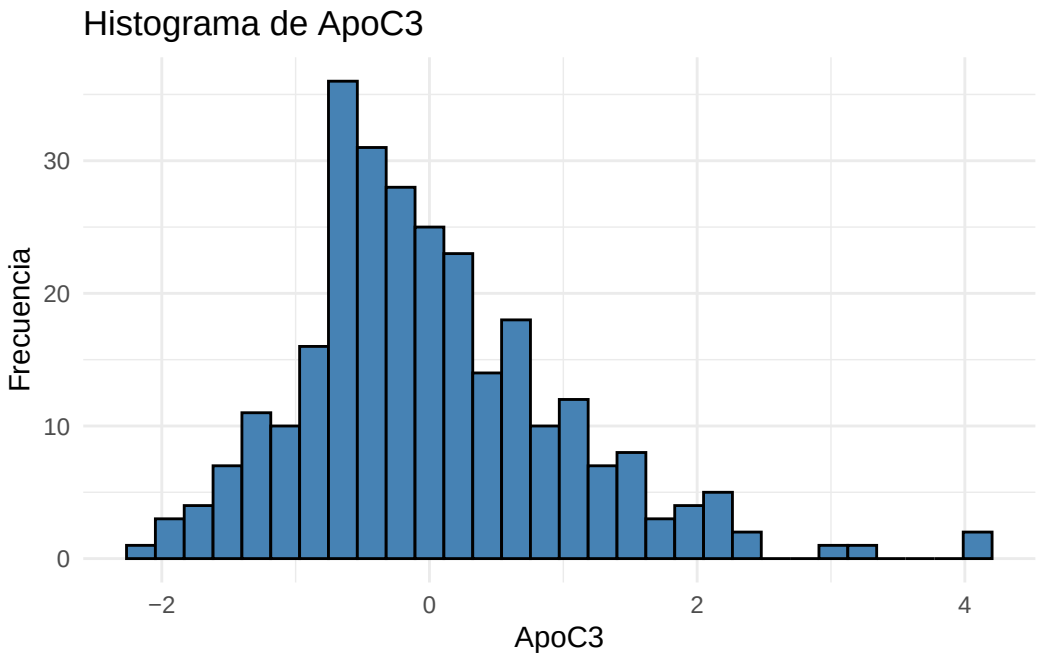


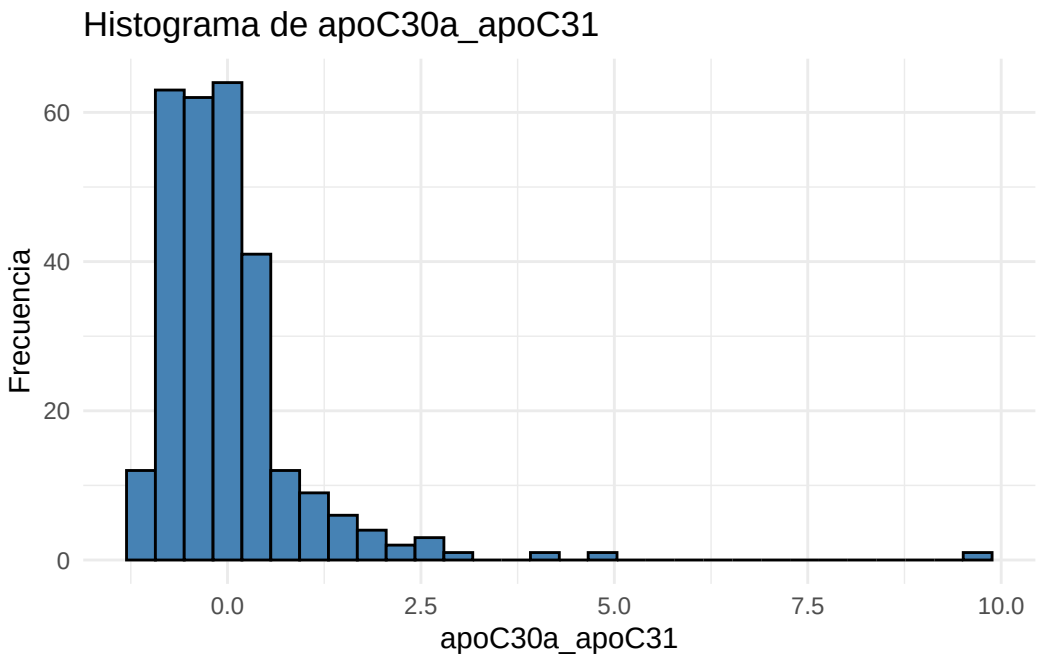

```
Warning: Removed 1 row containing non-finite outside the scale range
(`stat_bin()`).
```

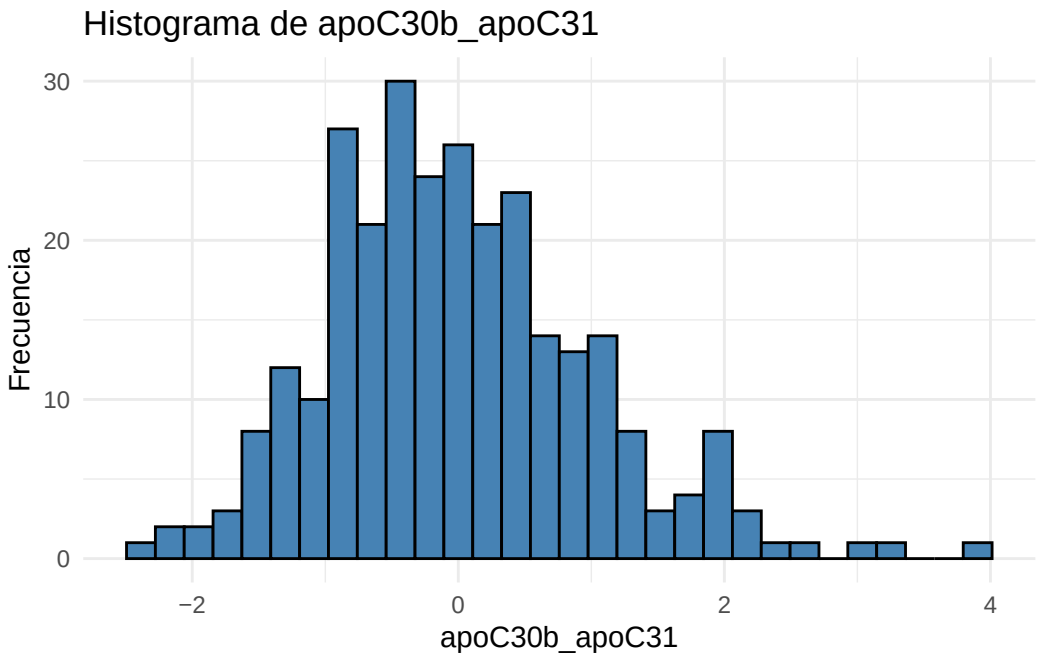

Histograma de col_v00
Frecuencia
30
20
10
0
200
400
600
800
col_v00

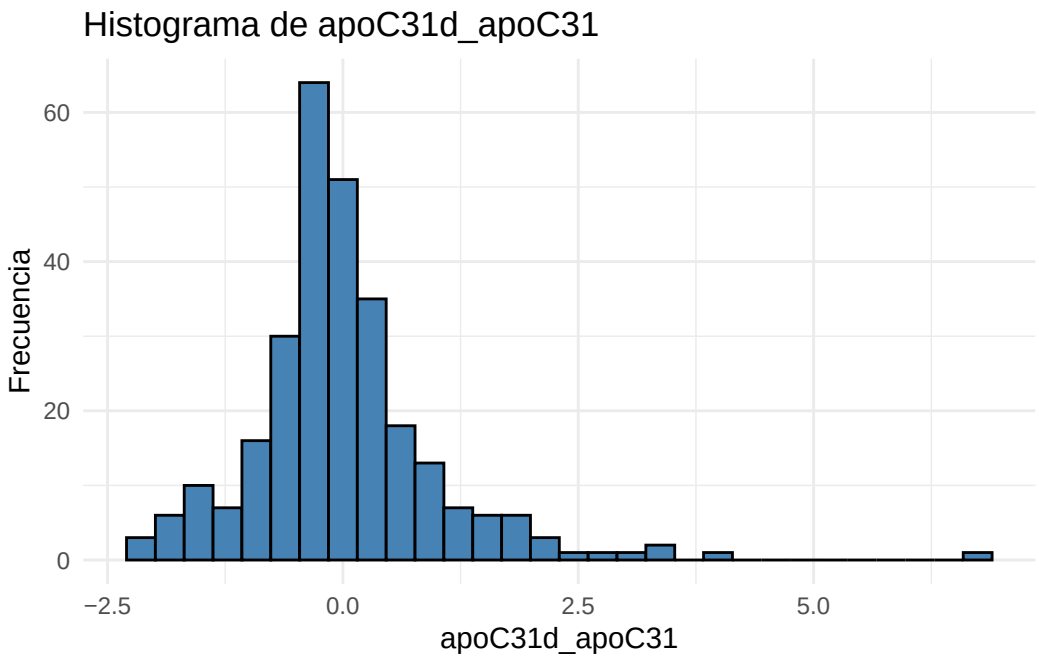

Histograma de p17_total_v00
Frecuencia
50
40
30
20
10
0
4
8
12
16
p17_total_v00

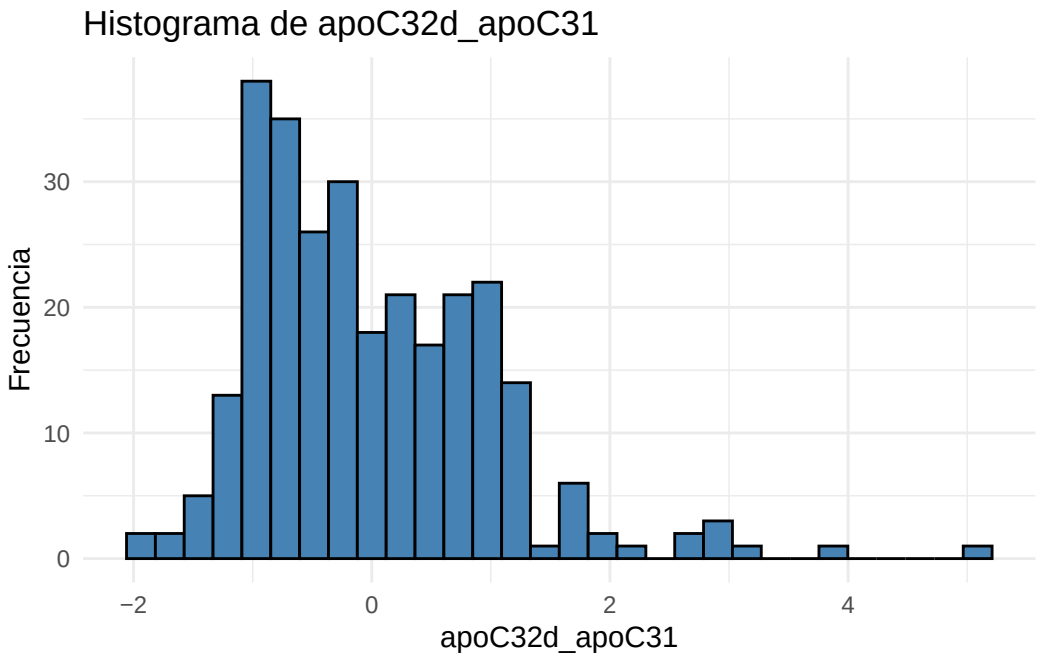

Histograma de edad_s1
Frecuencia
30
20
10
0
55
60
65
70
75
edad_s1

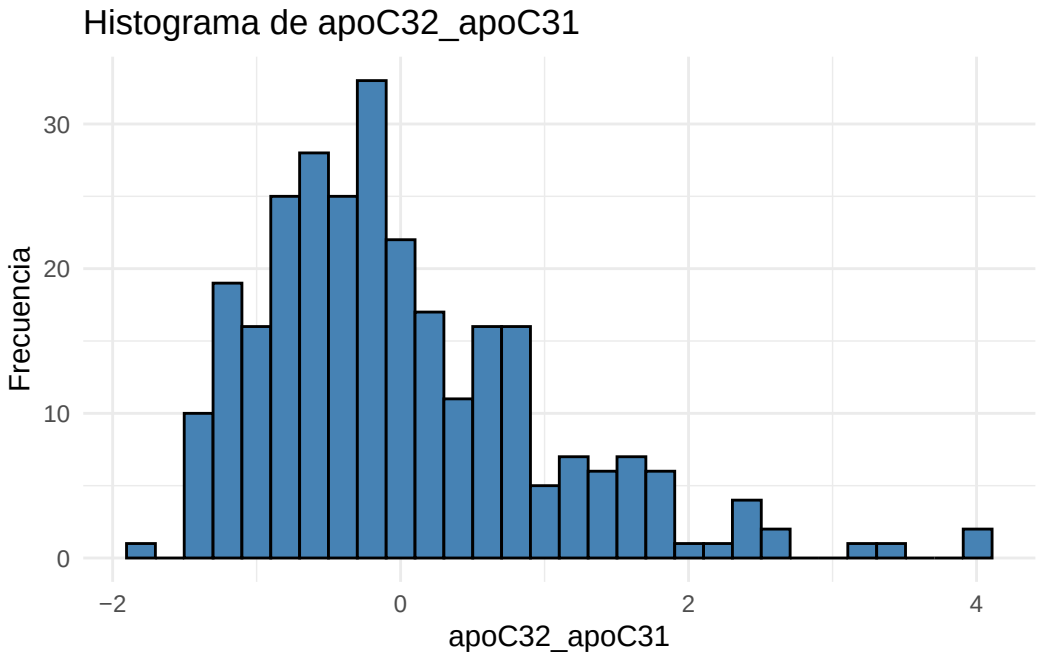

Histograma de ApoC3
Frecuencia
30
20
10
0
−2
0
2
4
ApoC3

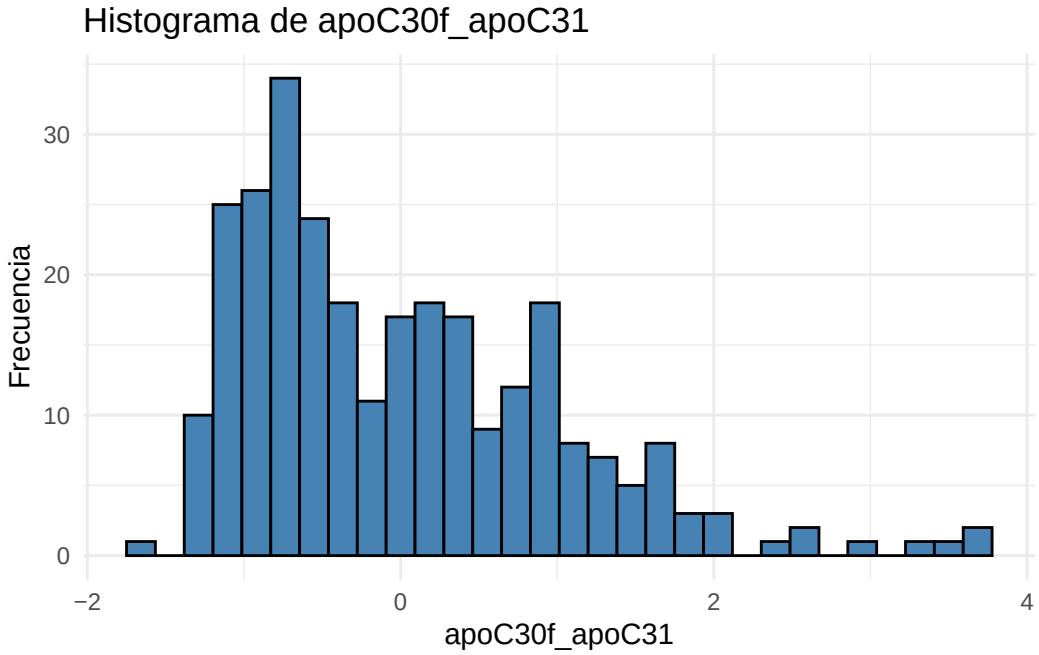

Histograma de apoC30a_apoC31
Frecuencia
60
40
20
0
0.0
2.5
5.0
7.5
10.0
apoC30a_apoC31


Histograma de apoC30b_apoC31
Frecuencia
30
20
10
0
−2
0
2
4
apoC30b_apoC31

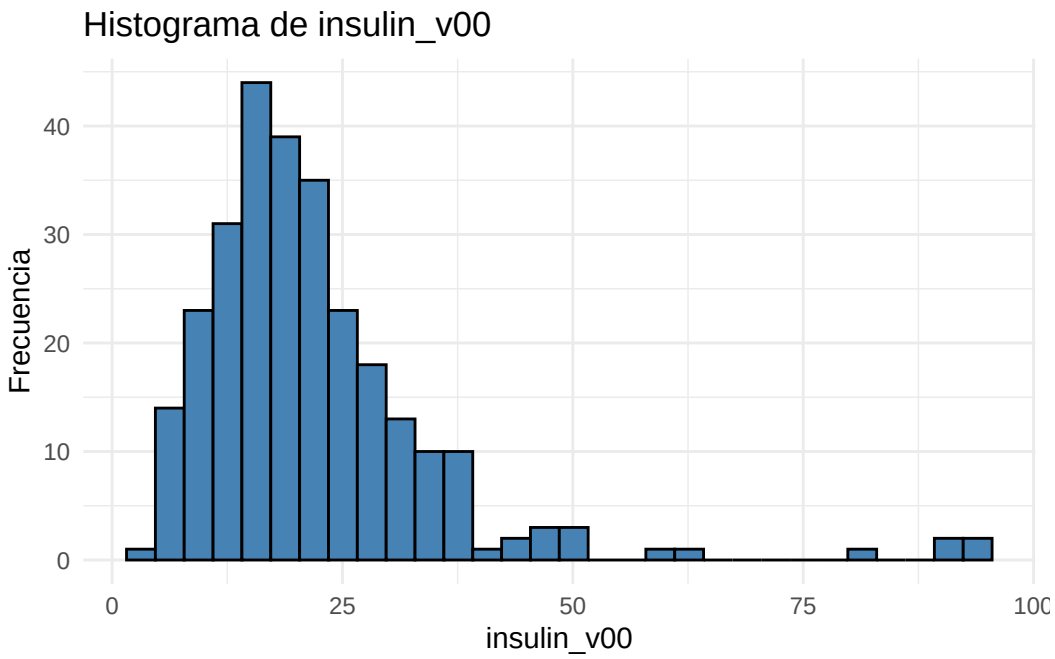

Histograma de apoC31d_apoC31
Frecuencia
60
40
20
0
−2.5
0.0
2.5
5.0
apoC31d_apoC31


Histograma de apoC32d_apoC31
Frecuencia
30
20
10
0
−2
0
2
4
apoC32d_apoC31

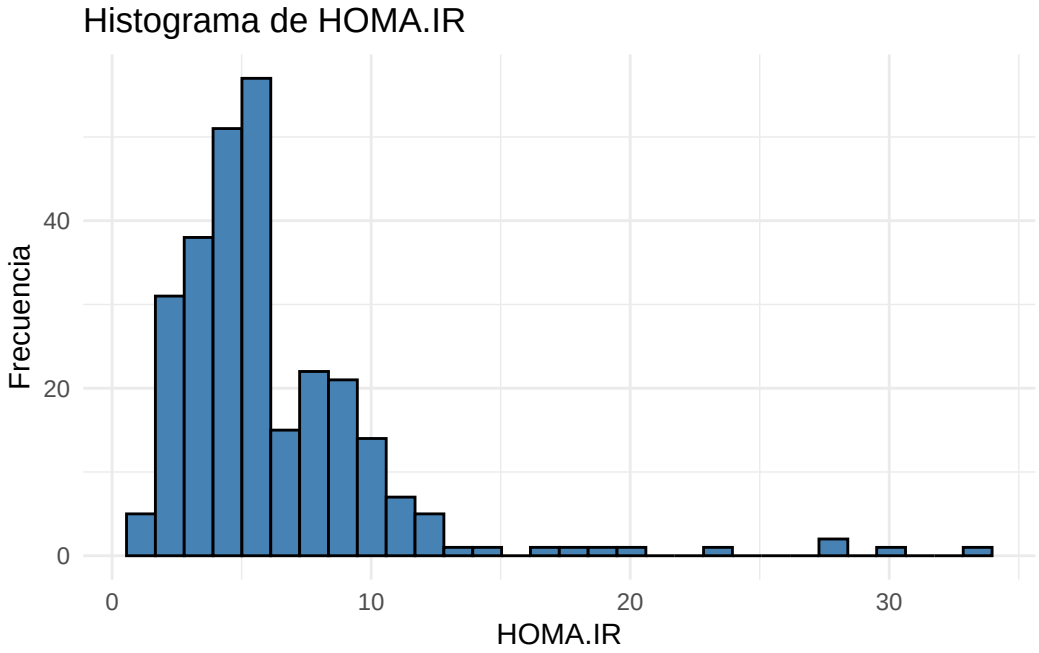





```
Warning: Removed 5 rows containing non-finite outside the scale range
(`stat_bin()`).
```




```
Warning: Removed 5 rows containing non-finite outside the scale range
(`stat_bin()`).
```



Variables a revisar: geaf_tot_v00, imc_v00, hdl_v00, trigli_v00, hc_v00, alcoholg_v00, energiat_v00, porc_po_v00, fibra_v00, apoC3, "apoC30a_apoC31", "apoC30b_apoC31", "apoC31d_apoC31", "apoC32d_apoC31", apoC32_apoC31, apoC30f_apoC31, insulin_v00, HOMA.IR.

Análisis:

Empiezo los modelos, las variables con distribución muy lejos de la normalidad, las transformo con log

Objetivo 1:

Regresiones lineales: modelos ajustados por: idcluster, centro, sexo, educación, actividad física, tabaco, MEDAS, IMC y tratamiendo dislipidémico.

glucosa_v00

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(log(glucosa_v00) ~ BBDD[[i]] + idcluster + edad_s1 +
                  nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                  imc_v00 + p17_total_v00 + tto_col_v00,
                   data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_glucosa.xlsx")
```

insulin_v00

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(log(insulin_v00) ~ BBDD[[i]] + idcluster + edad_s1 +
                  nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                  imc_v00 + p17_total_v00 + tto_col_v00,
                   data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_insulin.xlsx")
```

HOMA.IR

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(log(HOMA.IR) ~ BBDD[[i]] + idcluster + edad_s1 +
                  nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                  imc_v00 + p17_total_v00 + tto_col_v00,
                   data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_HOMA.IR.xlsx")
```

hba1c_v00 (es más o menos normal, no log)

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(hba1c_v00 ~ BBDD[[i]] + idcluster + edad_s1 +
                  nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                  imc_v00 + p17_total_v00 + tto_col_v00,
                   data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_hba1c_v00.xlsx")
```

hba1c_v00 (versión log)

```
r_glm = c()
SE = c()
p_valor = c()
```

```
for (i in 36:42){
   mod_glm <- lm(log(hba1c_v00) ~ BBDD[[i]] + idcluster + edad_s1 +
                   nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                   imc_v00 + p17_total_v00 + tto_col_v00,
                    data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_hba1c_log.xlsx")
```

coltot_v00

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(coltot_v00 ~ BBDD[[i]] + idcluster + edad_s1 +
                   nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                   imc_v00 + p17_total_v00 + tto_col_v00,
                    data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_coltot.xlsx")
```

hdl_v00

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(log(hdl_v00) ~ BBDD[[i]] + idcluster + edad_s1 +
                   nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                   imc_v00 + p17_total_v00 + tto_col_v00,
```

```
                    data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_hdl.xlsx")
```

ldl_calc_v00

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(ldl_calc_v00 ~ BBDD[[i]] + idcluster + edad_s1 +
                  nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                  imc_v00 + p17_total_v00 + tto_col_v00,
                   data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_ldl.xlsx")
```

trigli_v00

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(log(trigli_v00) ~ BBDD[[i]] + idcluster + edad_s1 +
                  nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                  imc_v00 + p17_total_v00 + tto_col_v00,
                   data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}
```

```
df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_TG.xlsx")
```

Objetivo 2:

Diferencias entre grupos:

```
medias = c()
SD = c()
P_v_a = c()

for (i in 36:42) {
  medias[[i]] = round(tapply(BBDD[[i]], BBDD[[45]], mean, na.rm = T),3)
  SD[[i]] = round(tapply(BBDD[[i]], BBDD[[45]], sd, na.rm = T),3)
  modelo = wilcox.test(BBDD[[i]] ~ BBDD[[45]])
  P_v_a[[i]]= round(modelo$p.value,3)
}
df_medias <- bind_rows(map(medias, ~ as.list(.x)))
df_SD <- bind_rows(map(SD, ~ as.list(.x)))
df_p_v_a <- map_dfr(P_v_a, ~ as_tibble(.x))
df_medias = data.frame(cbind(df_medias, df_SD, df_p_v_a))
df_medias <- cbind(variable = colnames(BBDD)[36:42], df_medias)
colnames(df_medias) = c("variables","Media No diab", "Media Diab",
                        "SD No diab", "SD Diab", "P_valor")
export(df_medias, "Resultados/wilcox_diabetes.xlsx")
```

Modelos de regresión logistica: idcluster, centro, sexo, educación, actividad física, tabaco, MEDAS, IMC y tratamiendo dislipidémico. Aquí también añado grupo intervención porque si que puede haber una influencia sobre la incidencia de diabetes. Ojo: añado edad como categórica dicotómica porque parace tener una relación con apoC30a_apoC31. Este modelo, sin edad, sale similar al resultado actual, con edad como continúa, su p es igual a 1 –> punto influyente pero sin interacción.

```
c_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
  mod_glm <- glm(diabetes ~ BBDD[[i]] + idcluster + grupo_int_v00 + cut2(edad_s1, g = 2) +
                   cut2(as.numeric(nodo), g = 3) + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                   imc_v00 + p17_total_v00 + tto_col_v00,
                 family = binomial(link = "logit"), data = BBDD)
  c_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
  SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
  p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(c_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
```

```
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_logisticas_diabetes.xlsx")
```

No hay suficientes participantes para utilizar nodos de forma normal y produce un sesgo. Para evitar la tendencia a infinito de algunos nodos, convierto la variable en numérica y hago terciles que se distribuirán en función a la cantidad de participantes por centro, reequilibrando la distribución.

Objetivo 3:

ANOVAS:

Pruebo primero con terciles de adherencia de MedDiet:

```
BBDD$ter_medas = cut2(BBDD$p17_total_v00, g = 3)
medias = c()
SD = c()
P_v_a = c()

for (i in 36:42) {
  medias[[i]] = round(tapply(BBDD[[i]], BBDD[[46]], mean, na.rm = T),2)
  SD[[i]] = round(tapply(BBDD[[i]], BBDD[[46]], sd, na.rm = T),2)
  modelo = lm(BBDD[[i]] ~ BBDD[[46]])
  P_v_a[[i]]= round(anova(modelo)[1,5],3)
}
df_medias <- bind_rows(map(medias, ~ as.list(.x)))
df_SD <- bind_rows(map(SD, ~ as.list(.x)))
df_p_v_a <- map_dfr(P_v_a, ~ as_tibble(.x))
df_medias = data.frame(cbind(df_medias, df_SD, df_p_v_a))
df_medias <- cbind(variable = colnames(BBDD)[36:42], df_medias)
colnames(df_medias) = c("Variables" ,"Media Ter 1 (1,8)", "Media Ter 2 (8, 10)", "Media Ter 3 (10, 17)",
                        "SD Ter 1 (1,8)", "SD Ter 2 (8, 10)", "SD Ter 3 (10, 17)",
                        "P_valor")
export(df_medias, "Resultados/medias_ter_MEDAS_aov.xlsx")
```

Ancovas: ajustadas por: idcluster, centro, sexo, educación, actividad física, tabaco, IMC y tratamiendo dislipidémico.

```
medias = c()
SD = c()
P_v_a = c()

for (i in 36:42) {
  medias[[i]] = round(tapply(BBDD[[i]], BBDD[[46]], mean, na.rm = T),2)
  SD[[i]] = round(tapply(BBDD[[i]], BBDD[[46]], sd, na.rm = T),2)
  modelo = lm(BBDD[[i]] ~ ter_medas + idcluster + edad_s1 +
                   nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                   imc_v00 + tto_col_v00,
                    data = BBDD)
  P_v_a[[i]]= round(anova(modelo)[1,5],3)
}
df_medias <- bind_rows(map(medias, ~ as.list(.x)))
df_SD <- bind_rows(map(SD, ~ as.list(.x)))
df_p_v_a <- map_dfr(P_v_a, ~ as_tibble(.x))
df_medias = data.frame(cbind(df_medias, df_SD, df_p_v_a))
```

```
df_medias <- cbind(variable = colnames(BBDD)[36:42], df_medias)
colnames(df_medias) = c("Variables" ,"Media Ter 1 (1,8)", "Media Ter 2 (8, 10)", "Media Ter 3 (10, 17)",
                        "SD Ter 1 (1,8)", "SD Ter 2 (8, 10)", "SD Ter 3 (10, 17)",
                        "P_valor")
export(df_medias, "Resultados/medias_ter_MEDAS_aoc_ajust.xlsx")
```

Regresiones lineales:

Terciles de MEDAS:

Regresión lineal asociaciones de terciles: modelos ajustados idcluster, centro, sexo, educación, actividad física, tabaco, IMC y tratamiendo dislipidémico.

```
r_glm1 = c()
r_glm2 = c()
SE1 = c()
SE2 = c()
p_valor1 = c()
p_valor2 = c()

for (i in 36:42){
   mod_glm <- lm(BBDD[[i]] ~ ter_medas + idcluster + edad_s1 +
                  nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                  imc_v00 + tto_col_v00,
                   data = BBDD)
   r_glm1[[i]] = round(summary(mod_glm)$coef[2,1],3)
   r_glm2[[i]] = round(summary(mod_glm)$coef[3,1],3)
   SE1[[i]] = round(summary(mod_glm)$coef[2,2],3)
   SE2[[i]] = round(summary(mod_glm)$coef[3,2],3)
   p_valor1[[i]] = round(summary(mod_glm)$coef[2,4],3)
   p_valor2[[i]] = round(summary(mod_glm)$coef[3,4],3)
}

df_coef1 <- map_dfr(r_glm1, ~ as_tibble(.x))
df_coef2 <- map_dfr(r_glm2, ~ as_tibble(.x))
df_SE1 <- map_dfr(SE1, ~ as_tibble(.x))
df_SE2 <- map_dfr(SE2, ~ as_tibble(.x))
df_p_valor1 <- map_dfr(p_valor1, ~ as_tibble(.x))
df_p_valor2 <- map_dfr(p_valor2, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef1, df_SE1, df_p_valor1, df_coef2, df_SE2,
                           df_p_valor2))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variables" ,"Coef Ter 2 (1,8)", "SE Ter 2 (8,10)", "P_valor Ter2",
                      "Coef Ter 3 (8, 10)", "SE Ter 3 (10, 17)", "P_valor Ter3")
export(df_rlog, "Resultados/regresiones_lineales_ter_Medas.xlsx")
```

Regresiones lineales para MEDAS y macronutrientes ajustadas por: idcluster, centro, sexo, educación, actividad física, tabaco, IMC y tratamiendo dislipidémico.

MEDAS:

```
r_glm = c()
SE = c()
p_valor = c()
```

```
for (i in 36:42){
   mod_glm <- lm(BBDD[[i]] ~ p17_total_v00 + idcluster + edad_s1 +
                  nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                  imc_v00 + tto_col_v00,
                   data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_MEDAS.xlsx")
```

HC:

Porcentaje:

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(BBDD[[i]] ~ porc_hc_v00 + idcluster + edad_s1 +
                  nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                  imc_v00 + tto_col_v00 + porc_pr_v00 + porc_gr_v00 +
                  alcoholg_v00 + fibra_v00,
                   data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_HC_pct.xlsx")
```

Prot:

Porcentaje:

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(BBDD[[i]] ~ porc_pr_v00 + idcluster + edad_s1 +
```

```
                    nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                    imc_v00 + tto_col_v00 + porc_hc_v00 + porc_gr_v00 +
                    alcoholg_v00 + fibra_v00,
                     data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_Prot_pct.xlsx")
```

Grasas totales:

Porcentaje:

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(BBDD[[i]] ~ porc_gr_v00 + idcluster + edad_s1 +
                    nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                    imc_v00 + tto_col_v00 + porc_hc_v00 + porc_pr_v00 +
                    alcoholg_v00 + fibra_v00,
                     data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_Grasas_totales_pct.xlsx")
```

AGrasas Monoinsaturados:

Porcentaje:

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(BBDD[[i]] ~ porc_mo_v00 + idcluster + edad_s1 +
                    nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
```

```
                    imc_v00 + tto_col_v00 + porc_hc_v00 + porc_pr_v00 +
                    alcoholg_v00 + fibra_v00,
                     data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_AGM_pct.xlsx")
```

AGrasas Polinsaturados:

Porcentaje:

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(BBDD[[i]] ~ porc_po_v00 + idcluster + edad_s1 +
                    nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                    imc_v00 + tto_col_v00 + porc_hc_v00 + porc_pr_v00 +
                    alcoholg_v00 + fibra_v00,
                     data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_AGP_pct.xlsx")
```

AGrasas Saturados:

Porcentaje:

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(BBDD[[i]] ~ porc_sa_v00 + idcluster + edad_s1 +
                    nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                    imc_v00 + tto_col_v00 + porc_hc_v00 + porc_pr_v00 +
```

```
                    alcoholg_v00 + fibra_v00,
                     data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_AGS_pct.xlsx")
```

Alcoholg:

Gramos:

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(BBDD[[i]] ~ alcoholg_v00 + idcluster + edad_s1 +
                    nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                    imc_v00 + tto_col_v00 + porc_hc_v00 + porc_pr_v00 + energiat_v00 +
                    porc_sa_v00 + fibra_v00,
                     data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_alcohol_g.xlsx")
```

Fibra:

Gramos:

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(BBDD[[i]] ~ fibra_v00 + idcluster + edad_s1 +
                    nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                    imc_v00 + tto_col_v00 + porc_hc_v00 + porc_pr_v00 + energiat_v00 +
                    porc_sa_v00 + alcoholg_v00,
```

```
                    data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_fibra_g.xlsx")
```

Energiat:

Kcal: la energía tiene un poco de cola a la derecha, utilizo transformación logarítmica para una mejor comprensión de las unidades y neutralizar el efecto de la cola:

```
r_glm = c()
SE = c()
p_valor = c()

for (i in 36:42){
   mod_glm <- lm(BBDD[[i]] ~ log(energiat_v00) + idcluster + edad_s1 +
                  nodo + sexo_s1 + escola_v00 + geaf_tot_v00 + fuma_s1 +
                  imc_v00 + tto_col_v00 + fibra_v00 + alcoholg_v00,
                   data = BBDD)
   r_glm[[i]] = round(summary(mod_glm)$coef[2,1],3)
   SE[[i]] = round(summary(mod_glm)$coef[2,2],3)
   p_valor[[i]] = round(summary(mod_glm)$coef[2,4],3)
}

df_coef <- map_dfr(r_glm, ~ as_tibble(.x))
df_SE <- map_dfr(SE, ~ as_tibble(.x))
df_p_valor <- map_dfr(p_valor, ~ as_tibble(.x))
df_rlog = data.frame(cbind(df_coef, df_SE, df_p_valor))
df_rlog <- cbind(variable = colnames(BBDD)[36:42], df_rlog)
colnames(df_rlog) = c("Variable", "Coeficiente", "Error_Estandar", "P_valor")
export(df_rlog, "Resultados/regresiones_lineales_energia_kcal.xlsx")
```

Características basales:

```
vars = c("grupo_int_v00","edad_s1", "sexo_s1", "escola_v00", "geaf_tot_v00", "fuma_s1",
         "imc_v00", "glucosa_v00", "insulin_v00", "HOMA.IR","hba1c_v00",
         "coltot_v00", "hdl_v00", "ldl_calc_v00", "trigli_v00", "tto_col_v00",
         "hc_v00", "prot_v00", "gratot_v00", "mo_v00", "po_v00", "sa_v00", "alcoholg_v00",
         "energiat_v00", "porc_hc_v00", "porc_pr_v00", "porc_gr_v00", "porc_mo_v00",
         "porc_po_v00", "porc_sa_v00", "fibra_v00", "p17_total_v00",
         "ApoC3", "apoC30a_apoC31", "apoC30b_apoC31",
         "apoC31d_apoC31", "apoC32d_apoC31", "apoC32_apoC31", "apoC30f_apoC31")

table1 = CreateTableOne(vars = vars, strata = "diabetes", data = BBDD, test = TRUE)
```

```r
table1_df = print(table1, quote = FALSE, noSpaces = TRUE, printToggle = FALSE)
table1_df = data.frame(Variable = rownames(table1_df), table1_df, row.names = NULL)

wb = createWorkbook()
addWorksheet(wb, "TableOne")

writeData(wb, "TableOne", table1_df)
saveWorkbook(wb, file = "Resultados/tabla_caract_basals.xlsx", overwrite = TRUE)
```

Conclusiones:

Por objetivos, se puede llegar a las siguientes conclusiones generales: - Objetivo 1: Hay relación entre los ratios de varias isoformas con Colesterol total, HDL, LDL y TG, siendo más fuerte esas asociaciones con Colesterol total y TG. En cuanto al metabolismo de la glucosa, también salen algunas asociaciones interesantes. - Objetivo 2: Un ratio y casi otro parecen asociarse con la diabetes, pero no hay diferencias de medias. - Objetivo 3: La dieta mediterránea no parece relacionarse en exceso con las isoformas salvo un ratio y en diferencia de medias. En modelos continuos no parece hacer relación y, en cuanto a los macronutrientes, en general no parece haber asociaciones salvo alguna asociación puntual.

Otras cuestiones a tener en cuenta: No he aplicado FDR porque probablemente nos quedemos en nada, al menos en los objetivos 2 y 3, por lo que recomiendo presentarlo como un estudio piloto.