

```

library(tidyverse)
library(haven)
library(survey)

# Cargamos las bases de datos de la ENDES 2017 con la función
read_sav()

SALUD      <- read_sav("data/CSALUD01.sav", encoding = 'UTF-8')
MUJER_OBS  <- read_sav("data/RE223132.SAV", encoding = 'UTF-8')
MUJER_LAC  <- read_sav("data/REC42.SAV", encoding = 'UTF-8')
PERSONA    <- read_sav("data/RECH1.SAV", encoding = 'UTF-8')
VIVIENDA   <- read_sav("data/RECH23.SAV", encoding = 'UTF-8')
HOGAR      <- read_sav("data/RECH0.SAV", encoding = 'UTF-8')

# Transformamos las variables de unión (DESCOMPOSICIÓN DEL CASEID EN
HHID Y QSNUMERO):
# 1. Descartamos últimos 3 caracteres de CASEID y lo convertimos a
HHID
# 2. Extraemos los últimos 2 caracteres de CASEID y lo convertimos a
QSNUMERO
# 3. Convertimos la variable de texto a numérica

MUJER_OBS$HHID      <- str_sub(MUJER_OBS$CASEID,1,
(str_length(MUJER_OBS$CASEID)-3))
MUJER_OBS$QSNUMERO   <- str_sub(MUJER_OBS$CASEID,-2,-1)
MUJER_OBS$QSNUMERO   <- as.numeric(MUJER_OBS$QSNUMERO)
MUJER_LAC$HHID      <- str_sub(MUJER_LAC$CASEID,1,
(str_length(MUJER_LAC$CASEID)-3))
MUJER_LAC$QSNUMERO   <- str_sub(MUJER_LAC$CASEID,-2,-1)
MUJER_LAC$QSNUMERO   <- as.numeric(MUJER_LAC$QSNUMERO)
PERSONA$QSNUMERO     <- PERSONA$HVIDX # Renombra HVIDX a QSNUMERO

# Realizamos la unión mediante los identificadores o llaves HHID y
QSNUMERO

BASE1 <- left_join(PERSONA, SALUD, by = c("HHID","QSNUMERO"))
BASE2 <- left_join(MUJER_OBS, MUJER_LAC, by = c("HHID", "QSNUMERO"))
BASE3 <- left_join(HOGAR, VIVIENDA, by = "HHID")
BASE4 <- left_join(BASE1,BASE2,by = c("HHID","QSNUMERO"))
BASE5 <- left_join(BASE3,BASE4,by = 'HHID')

# Filtramos la base de datos final (BASE5)

BASE_FINAL <- filter(BASE5,V213!= 1 | is.na(V213)) # Descartamos a
las gestantes
BASE_FINAL <- filter(BASE_FINAL,QSRESINF == 1) # Filtramos
entrevistas completas del CSALUD (n = 32 514)

# Descartamos las bases de datos que ya no son necesarias para
ahorrar memoria

rm(SALUD,BASE1,BASE2,BASE3,BASE4,BASE5,PERSONA,HOGAR,VIVIENDA,MUJER_
LAC,MUJER_OBS)

```

```

# Calculamos la variable Hipertensión Arterial y Obesidad

BASE_FINAL$PAMS <-(BASE_FINAL$QS905S+BASE_FINAL$QS903S)/2 #
Obtenemos la PAM sistólica de dos mediciones
BASE_FINAL$PAMD <-(BASE_FINAL$QS905D+BASE_FINAL$QS903D)/2 #
Obtenemos la PAM diastólica de dos mediciones
BASE_FINAL$HIPERTENSION <- (BASE_FINAL$PAMS>=140) |
(BASE_FINAL$PAMD>=90) # Definimos el criterio de HTA
BASE_FINAL$PESO <- ifelse(!
is.na(BASE_FINAL$V437),BASE_FINAL$V437/10,
ifelse(!
is.na(BASE_FINAL$QS900),BASE_FINAL$QS900,NA)) # Extraemos el peso de
dos variables
BASE_FINAL$TALLA <- ifelse(!
is.na(BASE_FINAL$V438),BASE_FINAL$V438/10,
ifelse(!
is.na(BASE_FINAL$QS901),BASE_FINAL$QS901,NA)) # Extraemos la talla
de dos variables
BASE_FINAL$IMC <- BASE_FINAL$PESO/(BASE_FINAL$TALLA/100)^2 #
Calculamos el IMC
BASE_FINAL$OBESIDAD <- ifelse(BASE_FINAL$IMC>=30,TRUE,FALSE) #
Definimos IMC según criterio (IMC mayor o igual a 30)

# Fruta
BASE_FINAL$Frutas_dias_ultima_semana <- ifelse(BASE_FINAL$QS213U ==
1,BASE_FINAL$QS213C,ifelse(BASE_FINAL$QS213U == 3,0,NA))
BASE_FINAL$Frutas_Unidades_dia <- BASE_FINAL$QS214C
BASE_FINAL$Fruta_por_semana <-
as.numeric(as.character(BASE_FINAL$Frutas_dias_ultima_semana))*as.nu
meric(as.character(BASE_FINAL$Frutas_Unidades_dia))
BASE_FINAL$Fruta_por_semana <-
ifelse(is.na(BASE_FINAL$Fruta_por_semana),
0,BASE_FINAL$Fruta_por_semana)

# Jugo de Fruta
BASE_FINAL$Jugo_Frutas_dias_ultima_semana <-
ifelse(BASE_FINAL$QS215U ==
1,BASE_FINAL$QS215C,ifelse(BASE_FINAL$QS215U == 3,0,NA))
BASE_FINAL$Jugo_Frutas_Vasos_dia <- BASE_FINAL$QS216C
BASE_FINAL$Jugo_Fruta_Vasos_por_semana <-
as.numeric(as.character(BASE_FINAL$Jugo_Frutas_dias_ultima_semana))*
as.numeric(as.character(BASE_FINAL$Jugo_Frutas_Vasos_dia))*2
BASE_FINAL$Jugo_Fruta_Vasos_por_semana <-
ifelse(is.na(BASE_FINAL$Jugo_Fruta_Vasos_por_semana),
0,BASE_FINAL$Jugo_Fruta_Vasos_por_semana)

# Ensalada de Fruta
BASE_FINAL$Ensalada_Frutas_dias_ultima_semana <-
ifelse(BASE_FINAL$QS217U ==
1,BASE_FINAL$QS217C,ifelse(BASE_FINAL$QS217U == 3,0,NA))
BASE_FINAL$Ensalada_Frutas_porciones_dia <- BASE_FINAL$QS218C
BASE_FINAL$Ensalada_Fruta_porciones_por_semana <-
as.numeric(as.character(BASE_FINAL$Ensalada_Frutas_dias_ultima_seman

```

```

a))*as.numeric(as.character(BASE_FINAL$Ensalada_Frutas_porciones_dia
))*2
BASE_FINAL$Ensalada_Fruta_porciones_por_semana <-
ifelse(is.na(BASE_FINAL$Ensalada_Fruta_porciones_por_semana),
0,BASE_FINAL$Ensalada_Fruta_porciones_por_semana)

# Ensalada de Verdura
BASE_FINAL$Ensalada_Verduras_dias_ultima_semana <-
ifelse(BASE_FINAL$QS219U ==
1,BASE_FINAL$QS219C,ifelse(BASE_FINAL$QS219U == 3,0,NA))
BASE_FINAL$Ensalada_Verduras_Unidad <- BASE_FINAL$QS220U
BASE_FINAL$Ensalada_Verduras_Unidades_dia <- ifelse(!
is.na(BASE_FINAL$QS220CC),BASE_FINAL$QS220CC,ifelse(!
is.na(BASE_FINAL$QS220CV),BASE_FINAL$QS220CV,NA))
BASE_FINAL$porcion_ensalada_verdura <-
ifelse(BASE_FINAL$Ensalada_Verduras_Unidad == 2,
BASE_FINAL$Ensalada_Verduras_Unidades_dia/4,
ifelse(BASE_FINAL$Ensalada_Verduras_Unidad ==
1,BASE_FINAL$Ensalada_Verduras_Unidades_dia,NA))
BASE_FINAL$Ensalada_Verdura_por_semana <-
as.numeric(as.character(BASE_FINAL$Ensalada_Verduras_dias_ultima_sem
ana))*as.numeric(as.character(BASE_FINAL$porcion_ensalada_verdura))
BASE_FINAL$Ensalada_Verdura_por_semana <-
ifelse(is.na(BASE_FINAL$Ensalada_Verdura_por_semana),
0,BASE_FINAL$Ensalada_Verdura_por_semana)

# Porciones de alimentos sanos
BASE_FINAL$fruta_o_ensalada_porciones_semana <-
BASE_FINAL$Ensalada_Verdura_por_semana+BASE_FINAL$Ensalada_Fruta_por
ciones_por_semana
+BASE_FINAL$Jugo_Fruta_Vasos_por_semana+BASE_FINAL$Fruta_por_semana
BASE_FINAL$DIETA_IDEAL <-
ifelse(BASE_FINAL$fruta_o_ensalada_porciones_semana >=
35,TRUE,FALSE)

# Generamos la base de datos final solo con las variables necesarias

ENDES <- select(BASE_FINAL, HHID, QSNUMERO, CONGLOMERADO = HV001,
ESTRATO = HV022,
PONDERACION = PES015_AMAS, SEX0 = HV104,
AREA_RESIDENCIA = HV025,
QUINTIL_BIENESTAR = HV270, REGION_NATURAL =
SHREGION,HIPERTENSION, PES0,
TALLA, IMC, OBESIDAD,DIETA_IDEAL)

# Especificamos el diseño muestral de la encuesta

diseño <- svydesign(id =~ CONGLOMERADO, strata =~ ESTRATO, weights=~
PONDERACION, data=ENDES)

```