



Cálculo de los Indicadores Clave con R

Sintaxis para el cálculo de los Indicadores Clave
usando el programa R

Departamento de Metodologías para Estadísticas de Gobierno y
Justicia

Elabora: Martha Aguilar Jiménez

Fecha de actualización: Enero de 2025.

En este documento se presenta la sintaxis para realizar el cálculo de los Indicadores Clave que se encuentran en el Catálogo Nacional de Indicadores, que son responsabilidad de la Dirección General de Estadísticas de Gobierno, Seguridad Pública y Justicia y cuya fuente de información son las Encuestas Nacionales de Gobierno y los Registros administrativos, además, se describe el proceso que se lleva a cabo en la actualización de los Indicadores Clave cuya fuente de información son los Censos Nacionales de Gobierno.

Índice

| | |
|---|----|
| 1. Introducción | 1 |
| 2. Cálculo de los IC cuya fuente de información son las Encuestas Nacionales de Gobierno (ENG). | 2 |
| 2.1 Estimación de los indicadores, variables y sus precisiones estadísticas. | 2 |
| 2.2 Caso especial de los IC cuya fuente de información es la Encuesta Nacional de Victimización de Empresas (ENVE) 2024. | 2 |
| 3. Cálculo de los IC cuya fuente de información es la Encuesta Nacional de Victimización y Percepción sobre Seguridad Pública (ENVIPE) 2024 | 3 |
| 3.1 Limpieza, instalación de paquetes, librerías, lectura de las bases de datos y ajustes de algunas variables de interés. | 3 |
| 3.2 Tasa de prevalencia delictiva por cada cien mil habitantes. | 4 |
| 3.3 Tasa de incidencia delictiva por cada cien mil habitantes | 9 |
| 4. Cálculo de los IC cuya fuente de información es la Encuesta Nacional de Calidad e Impacto Gubernamental (ENCIG) 2023 | 13 |
| 4.1 Limpieza, instalación de paquetes, librerías, lectura de las bases de datos y ajustes de algunas variables de interés. | 13 |
| 4.2 Porcentaje de la población de 18 años y más víctima de corrupción. | 14 |
| 4.3 Tasa de incidencia de corrupción en población de 18 años y más. | 15 |
| 5. Cálculo de los IC cuya fuente de información es la Encuesta Nacional de Población Privada de la Libertad (ENPOL) 2021. | 18 |
| 5.1 Limpieza, instalación de paquetes, librerías, lectura de las bases de datos y ajustes de algunas variables de interés. | 18 |
| 5.2 Victimización en Centros Penitenciarios. | 19 |
| 6. Cálculo de los IC cuya fuente de información es la Encuesta Nacional sobre la Dinámica de las Relaciones en los Hogares (ENDIREH) 2021 | 24 |
| 6.1 Limpieza, instalación de paquetes, librerías, lectura de las bases de datos y ajustes de algunas variables de interés. | 24 |
| 6.2 Prevalencia de la violencia contra las mujeres de 15 años y más a lo largo de su vida. | 25 |
| 7. Cálculo de los IC cuya fuente de información son los registros administrativos de las defunciones registradas. | 30 |
| 7.1 Limpieza, instalación de paquetes, librerías. | 30 |
| 7.2 Tasa bruta anual de defunciones por homicidio por cada cien mil habitantes. | 31 |
| 8. Cálculo de los IC cuya fuente de información son los Censos Nacionales de Gobierno (CNG). | 39 |
| 8.1 Proceso de actualización de los IC cuya fuente de información son los CNG. | 39 |
| 9 ANEXO: Funciones para el cálculo de los IC cuya fuente de información son las ENG. | 42 |

1. Introducción

El presente documento tiene como objetivo explicar cómo se calculan los Indicadores Clave (IC) a cargo del Subsistema Nacional de Información de Gobierno, Seguridad Pública e Impartición de Justicia (SNIGSPIJ), actualmente se cuenta con 83 IC que tienen como fuentes de información: los Censos Nacionales de Gobierno (CNG), las Encuestas Nacionales de Gobierno (ENG) y registros administrativos de defunciones registradas. En relación con los IC que tienen como fuente de información a las ENG, en el Catálogo Nacional de Indicadores (CNI) se presenta solo la estimación del IC, sin embargo, en este trabajo se presenta el código para obtener la estimación y las medidas de precisión (error estándar, coeficiente de variación e intervalos de confianza), con la finalidad de tener una completitud del cálculo.

Las encuestas que se utilizan para el cálculo de los IC son:

- Encuesta Nacional de Victimización de Empresas (ENVE) con 5 IC.
- Encuesta Nacional de Victimización y Percepción sobre Seguridad Pública (ENVIPE) con 21 IC.
- Encuesta Nacional de Calidad e Impacto Gubernamental (ENCIG) con 12 IC.
- Encuesta Nacional de Población Privada de la Libertad (ENPOL) con 4 IC.
- Encuesta Nacional sobre la Dinámica de las Relaciones en los Hogares (ENDIREH) con 8 IC.

Para los IC cuya fuente de información son los registros administrativos de las defunciones generales, el numerador se calcula a partir de los registros, mientras que, para el cálculo del denominador se utiliza la conciliación Demográfica 1950 a 2019 y Proyecciones de la población de México 2020 a 2070 del Consejo Nacional de Población (CONAPO).

- Registros Administrativos sobre las Estadísticas de Mortalidad (Defunciones Registradas) con 2 IC.

También, se presenta una descripción del proceso de actualización de los IC cuya fuente de información son los CNG, mediante el programa interno llamado: Sistema Gestor de Indicadores de Gobierno, Seguridad Pública y Justicia (SIGEI).

Los censos que se utilizan para el cálculo de los IC son:

- Censo Nacional de Gobiernos Estatales (CNGE) con 5 IC.
- Censo Nacional de Gobierno Federal (CNGF) con 1 IC.
- Censo Nacional de Gobiernos Municipales y Demarcaciones Territoriales de la Ciudad de México (CNGMD) con 1 IC.
- Censo Nacional de Procuración de Justicia Estatal (CNPJE) con 5 IC.
- Censo Nacional de Impartición de Justicia Estatal (CNIJE) con 13 IC.
- Censo Nacional de Sistemas Penitenciarios Estatales (CNSIPEE) con 2 IC.
- Censo Nacional de Derechos Humanos Estatal (CNDHE) y Censo Nacional de Derechos Humanos Federal (CNDHF) con 2 IC.
- CNSIPEE y Cuaderno Mensual de Información Estadística Penitenciaria Nacional con 1 IC.

NOTA: En 2021 el Censo Nacional de Gobierno, Seguridad Pública y Sistema Penitenciario Estatales (CNGSPSPE) fue suplido por el Censo Nacional de Gobiernos Estatales; el Censo Nacional de Seguridad Pública Estatal; y el Censo Nacional de Sistemas Penitenciarios Estatales.

Por otro lado, el Cuaderno Mensual de Información Estadística Penitenciaria Nacional es fuente de información de 1 IC a nivel federal.

Por último, al final del documento se muestra un apartado llamado *ANEXO* en donde se muestran las funciones auxiliares para el cálculo de los IC cuya fuente de información son las ENG.

2. Cálculo de los IC cuya fuente de información son las Encuestas Nacionales de Gobierno (ENG).

2.1 Estimación de los indicadores, variables y sus precisiones estadísticas.

En términos generales, el cálculo de un indicador se resume en los siguientes pasos:

1. Se construyen las variables que constituyen el numerador y el denominador para el cálculo del IC.
2. Se realiza la construcción del diseño muestral.
3. Se calcula la estimación del IC a partir de los datos muestrales a nivel nacional y por sus desagregaciones (entidad federativa, sexo y en algunos casos por edad).
4. Al tratarse de una encuesta, también, se calculan las medidas de precisión estadística, a continuación, se describen:
 - a. Error estándar
 - b. Coeficiente de variación
 - c. Límite inferior con nivel de confianza de 90%
 - d. Límite superior con nivel de confianza de 90%

Nota: Para los puntos 3 y 4 antes mencionados, se apoyará con la construcción de funciones que contengan los cálculos de los IC además de las precisiones estadísticas, con la intención de evitar repeticiones de código. Una vez creadas las funciones, éstas se ejecutan cada vez que se calcula un nuevo indicador. Las funciones se muestran en el apartado de *ANEXO* al final de este documento.

5. Se ordena la información de tal manera que cada IC único (desagregado) esté contenido en una tabla con la estimación puntual del IC acompañado de sus medidas de precisión, la(s) variable(s) que fungen como numerador y de sus medidas de precisión, la(s) variable(s) que fungen como denominador y de sus medidas de precisión, asimismo, la primera fila tendrá la estimación nacional y en las siguientes filas las estimaciones ordenadas según su desagregación, generalmente será por entidad federativa.
6. En sintonía con lo que publica el área de encuestas se colorean las estimaciones de acuerdo con su nivel de precisión, en Alto, Moderado y Bajo, tomando como referencia el coeficiente de variación CV (%). Una precisión Baja requiere un uso cauteloso de la estimación en el que se analicen las causas de la alta variabilidad y se consideren otros indicadores de precisión y confiabilidad, como el intervalo de confianza. A continuación, se muestra el nivel de precisión de las estimaciones.

| |
|---|
| Nivel de precisión de las estimaciones: |
| Alto, CV en el rango de (0,15) |
| Moderado, CV en el rango de [15,30) |
| Bajo, CV de 30% en adelante |

7. Se exportan las tablas a un archivo en Excel.

2.2 Caso especial de los IC cuya fuente de información es la Encuesta Nacional de Victimización de Empresas (ENVE) 2024.

Actualmente, en el CNI hay 5 IC familia cuya fuente de información es la ENVE, estos IC no se calculan, sino que se obtienen de los tabulados predefinidos que genera el área de encuestas en el siguiente enlace: **TABULADOS PREDEFINIDOS ENVE 2024**, lo anterior debido a que la información de la ENVE es confidencial.

Los 5 IC se enlistan a continuación.

1. Cifra negra de delitos en unidades económicas **Publicado en el DOF el 03-09-2015.**
2. Porcentaje de unidades económicas que identifica a los Ministerios Públicos y Procuradurías y manifiesta que les generan confianza **Publicado en el DOF el 03-09-2015.**
3. Tasa de prevalencia delictiva por cada diez mil unidades económicas **Publicado en el DOF el 27-09-2016.**
4. Tasa de prevalencia delictiva por cada 10 mil unidades económicas, por sector económico **Publicado en el DOF el 27-09-2016.**
5. Tasa de prevalencia delictiva por cada 10 mil unidades económicas, por tamaño de unidad económica **Publicado en el DOF el 27-09-2016.**

3. Cálculo de los IC cuya fuente de información es la Encuesta Nacional de Victimización y Percepción sobre Seguridad Pública (ENVIPE) 2024

Actualmente, en el CNI hay 21 IC familia cuya fuente de información es la ENVIPE.

3.1 Limpieza, instalación de paquetes, librerías, lectura de las bases de datos y ajustes de algunas variables de interés.

```
# Limpieza del entorno de R
rm(list=ls())
```

Se instalan los paquetes `foreign`, `openxlsx` y `survey` los cuales contienen las librerías de interés.

A continuación, se describen las características de cada paquete:

- a. **foreign** permite realizar la lectura y escritura de datos almacenados por algunos programas externos.
- b. **openxlsx** contiene funciones para simplificar la creación de archivos Excel, su escritura, edición y modificación desde R.
- c. **survey** permite realizar el análisis de muestras de encuestas y realiza el cálculo de las estimaciones, errores estándar, coeficiente de variación e intervalos de confianza.

```
packages<-c("foreign", "openxlsx", "survey")
if (length(setdiff(packages, rownames(installed.packages())) > 0) {
  install.packages(setdiff(packages, rownames(installed.packages())))
}
lapply(packages, require, character.only=TRUE)

# Libreria utilizada
library(foreign) # Lee datos almacenados en un formato externo
library(survey) # Análisis de encuestas complejas
```

Se define el directorio de trabajo con la ubicación de la base de datos de la ENVIPE 2024. Las bases de datos se obtienen en el siguiente enlace: **microdatos ENVIPE 2024**.

```
# Se define el directorio
getwd()
setwd("D:/OneDrive - INEGI/CÓDIGOS_EN GIT/compendio-codigos-ic_v2")
getwd()

# Se leen las bases de datos
```

```

thog <- read.dbf("bd_envipe_2024_dbf/THogar.dbf")
tvi <- read.dbf("bd_envipe_2024_dbf/TVivienda.dbf")
tsd <- read.dbf("bd_envipe_2024_dbf/TSDem.dbf")
tpv1 <- read.dbf("bd_envipe_2024_dbf/TPer_Vic1.dbf")
tpv2 <- read.dbf("bd_envipe_2024_dbf/TPer_Vic2.dbf")
tmv <- read.dbf("bd_envipe_2024_dbf/TMod_Vic.dbf")

```

Se convierten a tipo numérico los factores de expansión, para poder hacer multiplicaciones

```

# Cambiar el tipo de variable a numérico
tpv1$FAC_ELE <-as.numeric(as.character(tpv1$FAC_ELE))
tpv1$FAC_HOG <-as.numeric(as.character(tpv1$FAC_HOG))
tpv2$FAC_ELE <-as.numeric(as.character(tpv2$FAC_ELE))
tpv2$FAC_HOG <-as.numeric(as.character(tpv2$FAC_HOG))
tpv2$FAC_HOG[is.na(tpv2$FAC_HOG)] <- 0
tmv$FAC_DEL <-as.numeric(as.character(tmv$FAC_DEL))
tsd$FAC_HOG <-as.numeric(as.character(tsd$FAC_HOG))

# Opción para tratar los casos de los estratos con una sola una UPM
options(survey.lonely.psu="adjust")

```

3.2 Tasa de prevalencia delictiva por cada cien mil habitantes.

Publicado en el DOF el 28-12-2012.

3.2.1. Tasa de prevalencia delictiva por cada cien mil habitantes de 18 años y más, **total**.

```

#Paso 1: Se ejecutan las funciones: function1_nac y function1_ent.

#Paso 2: Se definen las variables de interés

#Se nombra a la base de datos con la que se trabaja
x<-tpv2

#Se define la variable de los delitos
x$vic18ymas <- ifelse(x$ID_PER%in%
                      tmv[!tmv$BPCOD%in%"03",]$ID_PER, 1, 0)
#Numerador: Población de 18 años y más víctima de algún delito.
x$numerador<- ifelse(x$vic18ymas,1,0)

#Denominador: Población de 18 años y más.
x$denominador <- ifelse(!x$FAC_ELE%in%"0"),1,0)

#Paso 3: Se define el diseño de la encuesta
DIS <- svydesign(id=~UPM_DIS, strata=~EST_DIS, data=x, weights=~FAC_ELE)

#Paso 4: Se mandan a llamar las funciones antes mencionadas

tab_pre_del_18_nac_t<-function1_nac(x,numerador ,denominador,DIS)
tab_pre_del_18_ent_t<-function1_ent(x,numerador ,denominador,DIS)
#Paso 5: Se integran las tablas del nacional y por entidad federativa

#Se asigna el mismo nombre de los encabezados

```

```
names(tab_pre_del_18_nac_t)<-names(tab_pre_del_18_ent_t)
#Unión de tablas
tab_pre_del_18_t<- rbind(tab_pre_del_18_nac_t,tab_pre_del_18_ent_t)
```

3.2.2. Tasa de prevalencia delictiva por cada cien mil habitantes de 18 años y más, **hombres**.

```
#Paso 1: Se ejecutan las funciones: function1_nac y function1_ent.

#Paso 2: Se definen las variables de interés

#Se nombra a la base de datos con la que se trabaja
x<-tpv2
#Se define la variable de los delitos
x$vic18ymas <- ifelse(x$ID_PER%in%
                      tmv[!tmv$BPCOD%in%"03",]$ID_PER, 1, 0)
#Numerador: Población de 18 años y más víctima de algún delito, hombres.
x$numerador<- ifelse ((x$SEXO%in%"1") & (x$vic18ymas),1,0)

#Denominador: Población de 18 años y más, hombres.
x$denominador <- ifelse((x$SEXO%in%"1") & (!x$FAC_ELE%in%"0"),1,0)

#Paso 3: Se define el diseño de la encuesta
DIS <- svydesign(id=~UPM_DIS, strata=~EST_DIS, data=x, weights=~FAC_ELE)

#Paso 4: Se mandan a llamar las funciones antes mencionadas

tab_pre_del_18_nac_h<-function1_nac(x,numerador ,denominador,DIS)
tab_pre_del_18_ent_h<-function1_ent(x,numerador ,denominador,DIS)
#Paso 5: Se integran las tablas del nacional y por entidad federativa

#Se asigna el mismo nombre de los encabezados
names(tab_pre_del_18_nac_h)<-names(tab_pre_del_18_ent_h)
#Unión de tablas
tab_pre_del_18_h<- rbind(tab_pre_del_18_nac_h,tab_pre_del_18_ent_h)
```

3.2.3. Tasa de prevalencia delictiva por cada cien mil habitantes de 18 años y más, **mujeres**.

```
#Paso 1: Se ejecutan las funciones: function1_nac, function1_ent y function1_cv.

#Paso 2: Se definen las variables de interés

#Se nombra a la base de datos con la que se trabaja
x<-tpv2
#Se define la variable de los delitos
x$vic18ymas <- ifelse(x$ID_PER%in%
                      tmv[!tmv$BPCOD%in%"03",]$ID_PER, 1, 0)
#Numerador: Población de 18 años y más víctima de algún delito, mujeres.
x$numerador<- ifelse ((x$SEXO%in%"2") & (x$vic18ymas),1,0)

#Denominador: Población de 18 años y más, mujeres.
x$denominador <- ifelse((x$SEXO%in%"2") & (!x$FAC_ELE%in%"0"),1,0)

#Paso 3: Se define el diseño de la encuesta
```



```

DIS <- svydesign(id=~UPM_DIS, strata=~EST_DIS, data=x, weights=~FAC_ELE)

#Paso 4: Se mandan a llamar las funciones antes mencionadas

tab_pre_del_18_nac_m<-function1_nac(x,numerador ,denominador,DIS)
tab_pre_del_18_ent_m<-function1_ent(x,numerador ,denominador,DIS)
#Paso 5: Se integran las tablas del nacional y por entidad federativa

#Se asigna el mismo nombre de los encabezados
names(tab_pre_del_18_nac_m)<-names(tab_pre_del_18_ent_m)
#Unión de tablas
tab_pre_del_18_m<- rbind(tab_pre_del_18_nac_m,tab_pre_del_18_ent_m)

#Paso 6: Exportar salida a un archivo de Excel
#Se exportará el indicador con sus desagregaciones por sexo.

# Crear un nuevo workbook
wb <- createWorkbook()

#Se agregan las hojas
addWorksheet( wb,"Preval_delic_t")
addWorksheet( wb,"Preval_delic_h")
addWorksheet( wb,"Preval_delic_m")
writeData(wb,"Preval_delic_t",tab_pre_del_18_t)
writeData(wb,"Preval_delic_h",tab_pre_del_18_h)
writeData(wb,"Preval_delic_m",tab_pre_del_18_m)

#Se definen las hojas y filas a aplicar el formato condicional
sheets <- c("Preval_delic_t", "Preval_delic_h", "Preval_delic_m")
rows <- 2:34

# Aplica el formato condicional
wb <- function1_cv(wb, sheets, cols_rules, rows)

#Se elige directorio para exportar los archivos
saveWorkbook(wb, file= "D:/OneDrive - INEGI/CÓDIGOS_EN
GIT/compendio-codigos-ic_v2/Estimaciones/ENVIPE/
Tasa de prevalencia 2023.xlsx",overwrite = TRUE)

```

3.2.4. Tasa de prevalencia delictiva por cada cien mil habitantes de 18 a 29 años, **total**.

```

#Paso 1: Se ejecutan las funciones: function1_nac y function1_ent.

#Paso 2: Se definen las variables de interés

#Se nombra a la base de datos con la que se trabaja
x<-tpv2

#Se define la variable de los delitos
x$vic18ymas <- ifelse(x$ID_PER%in%
                      tmv[!tmv$BPCOD%in%"03",]$ID_PER, 1, 0)
#Numerador: Población de 18 a 29 años víctima de algún delito.
x$numerador<- ifelse((x$vic18ymas%in%"1") &

```

```

      (as.numeric(as.character(x$EDAD)) < 30),1,0)
#Denominador: Población de 18 a 29 años.
x$denominador <- ifelse(!x$FAC_ELE%in%"0") &
      (as.numeric(as.character(x$EDAD)) < 30),1,0)
#Paso 3: Se define el diseño de la encuesta
DIS <- svydesign(id=~UPM_DIS, strata=~EST_DIS, data=x, weights=~FAC_ELE)

#Paso 4: Se mandan a llamar las funciones antes mencionadas

tab_pre_del_18a29_nac_t<-function1_nac(x,numerador ,denominador,DIS)
tab_pre_del_18a29_ent_t<-function1_ent(x,numerador ,denominador,DIS)
#Paso 5: Se integran las tablas del nacional y por entidad federativa

#Se asigna el mismo nombre de los encabezados
names(tab_pre_del_18a29_nac_t)<-names(tab_pre_del_18a29_ent_t)
#Unión de tablas
tab_pre_del_18a29_t<- rbind(tab_pre_del_18a29_nac_t,tab_pre_del_18a29_ent_t)

```

3.2.5. Tasa de prevalencia delictiva por cada cien mil habitantes de 18 a 29 años, **hombres**.

```

#Paso 1: Se ejecutan las funciones: function1_nac y function1_ent.

#Paso 2: Se definen las variables de interés

#Se nombra a la base de datos con la que se trabaja
x<-tpv2
#Se define la variable de los delitos
x$vic18ymas <- ifelse(x$ID_PER%in%
      tmv[!tmv$BPCOD%in%"03",]$ID_PER, 1, 0)
#Numerador: Población de 18 a 29 años víctima de algún delito, hombres.
x$numerador<- ifelse((x$SEXO%in%"1") & (x$vic18ymas) &
      (as.numeric(as.character(x$EDAD)) < 30),1,0)
#Denominador: Población de 18 a 29 años, hombres.
x$denominador <- ifelse((x$SEXO%in%"1") & (!x$FAC_ELE%in%"0") &
      (as.numeric(as.character(x$EDAD)) < 30),1,0)
#Paso 3: Se define el diseño de la encuesta
DIS <- svydesign(id=~UPM_DIS, strata=~EST_DIS, data=x, weights=~FAC_ELE)

#Paso 4: Se mandan a llamar las funciones antes mencionadas

tab_pre_del_18a29_nac_h<-function1_nac(x,numerador ,denominador,DIS)
tab_pre_del_18a29_ent_h<-function1_ent(x,numerador ,denominador,DIS)
#Paso 5: Se integran las tablas del nacional y por entidad federativa

#Se asigna el mismo nombre de los encabezados
names(tab_pre_del_18a29_nac_h)<-names(tab_pre_del_18a29_ent_h)
#Unión de tablas
tab_pre_del_18a29_h<- rbind(tab_pre_del_18a29_nac_h,tab_pre_del_18a29_ent_h)

```

3.2.6. Tasa de prevalencia delictiva por cada cien mil habitantes de 18 a 29 años, **mujeres**.

#Paso 1: Se ejecutan las funciones: function1_nac, function1_ent y function1_cv.

#Paso 2: Se definen las variables de interés

#Se nombra a la base de datos con la que se trabaja

```
x<-tpv2
```

#Se define la variable de los delitos

```
x$vic18ymas <- ifelse(x$ID_PER%in%  
                      tmv[!tmv$BPCOD%in%"03",]$ID_PER, 1, 0)  
#Numerador: Población de 18 a 29 años víctima de algún delito, mujeres.  
x$numerador<- ifelse((x$SEXO%in%"2") & (x$vic18ymas) &  
                     (as.numeric(as.character(x$EDAD)) < 30),1,0)
```

#Denominador: Población de 18 a 29 años, mujeres.

```
x$denominador <- ifelse((x$SEXO%in%"2") & (!x$FAC_ELE%in%"0") &  
                       (as.numeric(as.character(x$EDAD)) < 30),1,0)
```

#Paso 3: Se define el diseño de la encuesta

```
DIS <- svydesign(id=~UPM_DIS, strata=~EST_DIS, data=x, weights=~FAC_ELE)
```

#Paso 4: Se mandan a llamar las funciones antes mencionadas

```
tab_pre_del_18a29_nac_m<-function1_nac(x,numerador ,denominador,DIS)  
tab_pre_del_18a29_ent_m<-function1_ent(x,numerador ,denominador,DIS)
```

#Paso 5: Se integran las tablas del nacional y por entidad federativa

#Se asigna el mismo nombre de los encabezados

```
names(tab_pre_del_18a29_nac_m)<-names(tab_pre_del_18a29_ent_m)
```

#Unión de tablas

```
tab_pre_del_18a29_m<- rbind(tab_pre_del_18a29_nac_m,tab_pre_del_18a29_ent_m)
```

#Paso 6: Exportar salida a un archivo de Excel

#Se exportará el indicador con sus desagregaciones por sexo.

Crear un nuevo workbook

```
wb <- createWorkbook()
```

#Se agregan las hojas

```
addWorksheet( wb,"Preval_delic_18a29_t")  
addWorksheet( wb,"Preval_delic_18a29_h")  
addWorksheet( wb,"Preval_delic_18a29_m")  
writeData(wb,"Preval_delic_18a29_t",tab_pre_del_18a29_t)  
writeData(wb,"Preval_delic_18a29_h",tab_pre_del_18a29_h)  
writeData(wb,"Preval_delic_18a29_m",tab_pre_del_18a29_m)
```

#Se definen las hojas y filas a aplicar el formato condicional

```
sheets <- c("Preval_delic_18a29_t", "Preval_delic_18a29_h",  
           "Preval_delic_18a29_m")  
rows <- 2:34
```

Aplica el formato condicional

```
wb <- function1_cv(wb, sheets, cols_rules, rows)
```

```
#Se elige directorio para exportar los archivos
saveWorkbook(wb, file= "D:/OneDrive - INEGI/CÓDIGOS_EN GIT/
                    compendio-codigos-ic_v2/Estimaciones/ENVIPE/
                    Tasa de prevalencia 18a29 2023.xlsx",overwrite = TRUE)
```

3.3 Tasa de incidencia delictiva por cada cien mil habitantes

Publicado en el DOF el 27-09-2016

```
#Paso 1: Se ejecuta la función: function1_cv.
#Paso 2: Se calcula el numerador
#Numerador: Delitos ocurridos.

tmv[, "FAC_DEL00"] <- ifelse(!tmv$BPCOD%in%"03",tmv[, "FAC_DEL"],0)
for(ent in c(paste0(0,1:9),10:32))
  tmv[,paste0("FAC_DEL",ent)] <- ifelse(!tmv$BPCOD%in%"03" &
                                         tmv$BP1_2C%in%ent,tmv[, "FAC_DEL"],0)

#Se crearon 33 columnas de factores de delito, una para cada entidad,
#incluido el valor nacional en la base tmv

for(ent in c(paste0(0,0:9),10:32)){
  SFD <- data.frame(tapply(tmv[,paste0("FAC_DEL",ent)],tmv$ID_PER,sum))
  #Suma del Factor Delito
  SFD <- data.frame(rownames(SFD),SFD)
  names(SFD) <- c("ID_PER","SUMA_FAC_DEL")
  tpv2[,paste0("TDEO",ent)] <- 0
  Z <-data.frame(1:length(tpv2[tpv2$ID_PER%in%SFD[,1],"ID_PER"]),
                tpv2[tpv2$ID_PER%in%SFD[,1],"ID_PER"])
  names(Z) <- c("NUM","ID_PER")
  AUX0 <- merge(SFD,Z,by = "ID_PER")
  AUX1 <- AUX0[order(AUX0[,3]),]
  tpv2[tpv2$ID_PER%in%SFD[,1],paste0("TDEO",ent)] <- AUX1[,2]
}

#Se crearon 33 variables "TDEO,(ent)" (Total de delitos) en la base tpv2

#Paso 3: Se calcula el denominador

#Denominador: Población de 18 años y más.

for(ent in c(paste0(0,0:9),10:32))
  if(ent%in%"00") tpv2[, "POB00"]<-tpv2$FAC_ELE else tpv2[,paste0("POB",ent)] <-
  ifelse(tpv2$CVE_ENT%in%ent,tpv2$FAC_ELE,0)

#Paso 4: Se define el diseño de la encuesta
DIS <- svydesign(id=~UPM_DIS, strata=~EST_DIS, data=tpv2, weights=~1)

#Paso 5: Se calculan los insumos además de sus medidas de precisión a nivel
#nacional y por entidad federativa

#Cálculo del numerador
Incidencia_N<-list()
for(ent in c(paste0(0,0:9),10:32))
```

```

Incendencia_N[[as.integer(ent)+1]]<-
svytotal(~eval(parse(text = paste0("TDEO",ent))),DIS)

EST_N <- SE_N <- CV_N <- LI_N <- LS_N <- list()
for(h in 1:33){
  EST_N[h] <- Incendencia_N[[h]][[1]]
  SE_N[h] <- SE(Incendencia_N[[h]])
  CV_N[h] <- cv(Incendencia_N[[h]])*100
  LI_N[h] <- confint(Incendencia_N[[h]],level=0.90)[1,1]
  LS_N[h] <- confint(Incendencia_N[[h]],level=0.90)[1,2]
}
EST_TVS_N<-do.call(rbind,EST_N)
SE_TVS_N<-do.call(rbind,SE_N)
CV_TVS_N<-do.call(rbind,CV_N)
LI_TVS_N<-do.call(rbind,LI_N)
LS_TVS_N<-do.call(rbind,LS_N)

#Cálculo del denominador
Incendencia_D<-list()
for(ent in c(paste0(0,0:9),10:32))
  Incendencia_D[[as.integer(ent)+1]]<-
    svytotal(~eval(parse(text = paste0("POB",ent))),DIS)

EST_D <- SE_D <- CV_D <- LI_D <- LS_D <- list()
for(h in 1:33){
  EST_D[h] <- Incendencia_D[[h]][[1]]
  SE_D[h] <- SE(Incendencia_D[[h]])
  CV_D[h] <- cv(Incendencia_D[[h]])*100
  LI_D[h] <- confint(Incendencia_D[[h]],level=0.90)[1,1]
  LS_D[h] <- confint(Incendencia_D[[h]],level=0.90)[1,2]
}
EST_TVS_D<-do.call(rbind,EST_D)
SE_TVS_D<-do.call(rbind,SE_D)
CV_TVS_D<-do.call(rbind,CV_D)
LI_TVS_D<-do.call(rbind,LI_D)
LS_TVS_D<-do.call(rbind,LS_D)

#Paso 6: Se calcula el indicador además de sus medidas de precisión a nivel
#nacional y por entidad federativa
Incendencia<-list()
for(ent in c(paste0(0,0:9),10:32))
  Incendencia[[as.integer(ent)+1]]<-
    svyratio(~eval(parse(text = paste0("TDEO",ent))),
              denominator=~eval(parse(text = paste0("POB",ent))),DIS)

EST <- SE <- CV <- LI <- LS <- list()
for(h in 1:33){
  EST[h] <- Incendencia[[h]][[1]]*100000
  SE[h] <- SE(Incendencia[[h]])*100000
  CV[h] <- cv(Incendencia[[h]])*100
  LI[h] <- confint(Incendencia[[h]],level=0.90)[1,1]*100000
  LS[h] <- confint(Incendencia[[h]],level=0.90)[1,2]*100000
}

```

```

EST_TVSe<-do.call(rbind,EST)
SE_TVSe<-do.call(rbind,SE)
CV_TVSe<-do.call(rbind,CV)
LI_TVSe<-do.call(rbind,LI)
LS_TVSe<-do.call(rbind,LS)

#Paso 7: Creación del tabulado

enti <- c('Estados Unidos Mexicanos', 'Aguascalientes', 'Baja California',
          'Baja California Sur', 'Campeche', 'Coahuila de Zaragoza', 'Colima',
          'Chiapas', 'Chihuahua', 'Ciudad de México', 'Durango', 'Guanajuato',
          'Guerrero', 'Hidalgo', 'Jalisco', 'Estado de México',
          'Michoacán de Ocampo', 'Morelos', 'Nayarit', 'Nuevo León', 'Oaxaca',
          'Puebla', 'Querétaro', 'Quintana Roo', 'San Luis Potosí', 'Sinaloa',
          'Sonora', 'Tabasco', 'Tamaulipas', 'Tlaxcala',
          'Veracruz de Ignacio de la Llave', 'Yucatán', 'Zacatecas')

tvse <- data.frame(EST_TVSe)
tvse_n <- data.frame(EST_TVSe_N)
tvse_d <- data.frame(EST_TVSe_D)
se <- data.frame(SE_TVSe)
se_n <- data.frame(SE_TVSe_N)
se_d <- data.frame(SE_TVSe_D)
cv <- data.frame(CV_TVSe)
cv_n <- data.frame(CV_TVSe_N)
cv_d <- data.frame(CV_TVSe_D)
linf <- data.frame(LI_TVSe)
linf_n <- data.frame(LI_TVSe_N)
linf_d <- data.frame(LI_TVSe_D)
lisup <- data.frame(LS_TVSe)
lisup_n <- data.frame(LS_TVSe_N)
lisup_d <- data.frame(LS_TVSe_D)

tab_incidencia_delic <- cbind(enti,tvse,se,cv,linf,lisup,
                             tvse_n,se_n,cv_n,linf_n,lisup_n,
                             tvse_d,se_d,cv_d,linf_d,lisup_d)

#Paso 8: Exportar salida a un archivo de Excel
#Se exportará el indicador.
# Crear un nuevo workbook
wb<-createWorkbook()

# Se agregan las hojas
addWorksheet( wb,"Tasa_incidencia")
writeData(wb,"Tasa_incidencia",tab_incidencia_delic)

#Se define la hoja y filas a aplicar el formato condicional
sheets <- c("Tasa_incidencia")
rows <- 2:34

# Aplica el formato condicional
wb <- function1_cv(wb, sheets, cols_rules, rows)

```

```
#Se elige directorio para exportar los archivos  
saveWorkbook(wb, file= "D:/OneDrive - INEGI/CÓDIGOS_EN GIT/  
    compendio-codigos-ic_v2/Estimaciones/ENVIPE/  
    Tasa de incidencia_2023.xlsx", overwrite = TRUE)
```

4. Cálculo de los IC cuya fuente de información es la Encuesta Nacional de Calidad e Impacto Gubernamental (ENCIG) 2023

Actualmente, en el CNI hay 12 IC familia cuya fuente de información es la ENCIG.

4.1 Limpieza, instalación de paquetes, librerías, lectura de las bases de datos y ajustes de algunas variables de interés.

```
rm(list=ls())
```

Se instalan los paquetes foreign, openxlsx y survey los cuales contienen las librerías de interés.

```
packages<-c("foreign", "openxlsx", "survey")
if (length(setdiff(packages, rownames(installed.packages()))) > 0) {
  install.packages(setdiff(packages, rownames(installed.packages())))
}
lapply(packages, require, character.only=TRUE)

# Libreria utilizada
library(foreign) # Lee datos almacenados en un formato externo
library(survey) # Análisis de encuestas complejas
```

Se define el directorio de trabajo con la ubicación de la base de datos de la ENCIG 2023. Las bases de datos se obtienen en el siguiente enlace: **microdatos ENCIG 2023**.

```
# Se define el directorio
getwd()
setwd("D:/OneDrive - INEGI/CÓDIGOS_EN GIT/compendio-codigos-ic_v2")
getwd()

# Se leen las bases de datos
Sec1_10<-read.dbf("encig23_base_datos_dbf/encig2023_01_sec1_A_3_4_5_8_9_10.dbf",
  as.is = T)
Sec2<-read.dbf("encig23_base_datos_dbf/encig2023_02_residentes_sec_2.dbf",
  as.is = T)
Sec6<-read.dbf("encig23_base_datos_dbf/encig2023_03_sec_6.dbf",as.is = T)
Sec7<-read.dbf("encig23_base_datos_dbf/encig2023_04_sec_7.dbf",as.is = T)
Sec8<-read.dbf("encig23_base_datos_dbf/encig2023_05_sec_8.dbf",as.is = T)
Sec11<-read.dbf("encig23_base_datos_dbf/encig2023_01_sec_11.dbf",as.is = T)
```

Se convierten a tipo numérico los factores de expansión y las preguntas que se requieran en el cálculo, para poder hacer multiplicaciones

```
Sec1_10$FAC_P18 <- as.numeric(as.character(Sec1_10$FAC_P18))
Sec6$FAC_P18 <- as.numeric(as.character(Sec6$FAC_P18))
Sec8$FAC_P18<- as.numeric(as.character(Sec8$FAC_P18))
Sec8$P8_5<-as.numeric(Sec8$P8_5)
```

Se realizan los cruces en las tablas para el cálculo de los IC.


```
Sec11 <- merge(Sec11,Sec2[, c("SEX0", "ID_PER")],by="ID_PER")
Sec1_10 <- merge(Sec1_10,Sec2[, c("SEX0", "ID_PER")],by="ID_PER")
```

4.2 Porcentaje de la población de 18 años y más víctima de corrupción.

Publicado en el DOF el 24-05-2017

#Paso 1: Se ejecutan las funciones: function2_nac, function2_ent y function1_cv.

#Paso 2: Se definen las variables de interés

#Numerador: Población de 18 años y más víctima de corrupción en al menos un trámite, pago o solicitud de servicio realizado personalmente.

```
Sec8$v0 <- ifelse(!as.numeric(Sec8$P8_5)%in%c(NA,0),1,0)
dfn<-as.data.frame(tapply(Sec8$v0,Sec8$ID_PER,sum))
dfn<-data.frame('ID_PER'=rownames(dfn),dfn)
colnames(dfn)<-c('ID_PER','sum2_8.5')
Sec1_10<-merge(Sec1_10,dfn[,c('ID_PER','sum2_8.5')],by='ID_PER',all.x = T)
Sec1_10$numerador <- ifelse(!Sec1_10$sum2_8.5%in%c(NA,0),1,0)
```

#Denominador: Población de 18 años y más que realizó personalmente al menos un trámite, pago o solicitud de servicios con un servidor público.

```
Sec7$v0 <- ifelse(Sec7$P7_3%in%c("1","3","6") | Sec7$N_TRA%in%"20",1,0)
#Se aplica el tapply
df<-as.data.frame(tapply(Sec7$v0,Sec7$ID_PER,sum))
df<-data.frame('ID_PER'=rownames(df),df)
colnames(df)<-c('ID_PER','sum_7.3yn_tra')
Sec1_10<-merge(Sec1_10,df[,c('ID_PER','sum_7.3yn_tra')],by='ID_PER',all.x = T)
#Se crean dos variables dicótomicas "numerador" y "aux1".
Sec1_10$aux1 <- ifelse(!Sec1_10$sum_7.3yn_tra%in%c(NA,0),1,0)

#Se crea el denominador
Sec1_10$denominador <- ifelse((Sec1_10$aux1%in%1 | Sec1_10$numerador%in%1),1,0)
```

#Se nombra a la base de datos con la que se trabaja

```
x<-Sec1_10
```

#Paso 3: Se define el diseño de la encuesta

```
DIS <- svydesign(id=~ UPM_DIS, strata=~EST_DIS, data=x, weights=~FAC_P18)
```

#Paso 4: Se mandan a llamar las funciones antes mencionadas

```
tab_victima_corrup_nac<-function2_nac(x,numerador ,denominador,DIS)
tab_victima_corrup_ent<-function2_ent(x,numerador ,denominador,DIS)
```

#Paso 5: Se integran las tablas del nacional y por entidad federativa

#Se asigna el mismo nombre de los encabezados

```
names(tab_victima_corrup_nac)<-names(tab_victima_corrup_ent)
```

#Unión de tablas

```
tab_victima_corrup<- rbind(tab_victima_corrup_nac,tab_victima_corrup_ent)
```

```

#Paso 6: Exportar salida a un archivo de Excel
#Se exportará el indicador.

# Crear un nuevo workbook
wb <- createWorkbook()

#Se agregan las hojas
addWorksheet( wb,"Porcentaje_víctima_corrup")
writeData(wb,"Porcentaje_víctima_corrup",tab_victima_corrup)

#Se definen las hojas y filas a aplicar el formato condicional
sheets <- c("Porcentaje_víctima_corrup")
rows <- 2:34

# Aplica el formato condicional
wb <- function1_cv(wb, sheets, cols_rules, rows)

#Se elige directorio para exportar los archivos
saveWorkbook(wb,file="D:/OneDrive - INEGI/CÓDIGOS_EN GIT/
               compendio-codigos-ic_v2/Estimaciones/ENCIG/
               Porce_pob_víctima_corrup_2023.xlsx", overwrite = TRUE)

```

4.3 Tasa de incidencia de corrupción en población de 18 años y más.

Publicado en el DOF el 24-05-2017

```

#Paso 1: Se ejecutan las funciones: function2_nac, function2_ent y function1_cv.

#Paso 2: Se definen las variables de interés

#Numerador: Trámites, pagos o solicitudes de servicios realizados personalmente
#con un servidor público en los que existió corrupción.

Sec8$v1 <- ifelse(!as.numeric(Sec8$P8_5)%in%c(NA,0),as.numeric(Sec8$P8_5),0)
df3<-as.data.frame(tapply(Sec8$v1,Sec8$ID_PER,sum))
df3<-data.frame('ID_PER'=rownames(df3),df3)
colnames(df3)<-c('ID_PER','corrupcion')
Sec1_10<-merge(Sec1_10,df3[,c('ID_PER','corrupcion')],by='ID_PER',all.x = T)
Sec1_10$numerador<-ifelse(!Sec1_10$corrupcion%in%0,Sec1_10$corrupcion,0)

#Denominador: Población de 18 años y más que realizó personalmente al menos un
#trámite, pago o solicitud de servicios con un servidor público.

#Se crea la variable V0 en la sección 7.

Sec7$v0 <- ifelse(Sec7$P7_3%in%c("1","3","6") | Sec7$N_TRA%in%"20",1,0)
df<-as.data.frame(tapply(Sec7$v0,Sec7$ID_PER,sum))
df<-data.frame('ID_PER'=rownames(df),df)
colnames(df)<-c('ID_PER','sum_7.3')
Sec1_10<-merge(Sec1_10,df[,c('ID_PER','sum_7.3')],by='ID_PER',all.x = T)

#Se crea la variable V0 en la sección 8

```

```

Sec8$v0 <- ifelse(!as.numeric(Sec8$P8_5)%in%c(NA,0),1,0)
#APLICAMOS EL tapply
df2<-as.data.frame(tapply(Sec8$v0,Sec8$ID_PER,sum))
df2<-data.frame('ID_PER'=rownames(df2),df2)
colnames(df2)<-c('ID_PER','sum_8.5')
Sec1_10<-merge(Sec1_10,df2[,c('ID_PER','sum_8.5')],by='ID_PER',all.x = T)

#Se crean dos variables auxiliares para filtrar los resultados.
Sec1_10$aux2 <- ifelse(!Sec1_10$sum_7.3%in%c(NA,0),1,0)
Sec1_10$aux3 <- ifelse(!Sec1_10$sum_8.5%in%c(NA,0),1,0)

Sec1_10$denominador <- ifelse((Sec1_10$aux2%in%1 | Sec1_10$aux3%in%1),1,0)

#Se nombra a la base de datos con la que se trabaja
x<-Sec1_10

#Paso 3: Se define el diseño de la encuesta
DIS <- svydesign(id=~UPM_DIS, strata=~EST_DIS, data=x, weights=~FAC_P18)

#Paso 4: Se mandan a llamar las funciones antes mencionadas
tab_tasa_incid_corrup_nac<-function1_nac(x,numerador ,denominador,DIS)
tab_tasa_incid_corrup_ent<-function1_ent(x,numerador ,denominador,DIS)

#Paso 5: Se integran las tablas del nacional y por entidad federativa

#Se asigna el mismo nombre de los encabezados
names(tab_tasa_incid_corrup_nac)<-names(tab_tasa_incid_corrup_ent)
#Unión de tablas
tab_tasa_incid_corrup<-rbind(tab_tasa_incid_corrup_nac,
                             tab_tasa_incid_corrup_ent)

#Paso 6: Exportar salida a un archivo de Excel
#Se exportará el indicador.

# Crear un nuevo workbook
wb <- createWorkbook()

#Se agregan las hojas
addWorksheet( wb,"Tasa_incidencia_corrup")
writeData(wb,"Tasa_incidencia_corrup",tab_tasa_incid_corrup)

#Se definen las hojas y filas a aplicar el formato condicional
sheets <- c("Tasa_incidencia_corrup")
rows <- 2:34

# Aplica el formato condicional
wb <- function1_cv(wb, sheets, cols_rules, rows)

#Se elige directorio para exportar los archivos
saveWorkbook(wb,file="D:/OneDrive - INEGI/CÓDIGOS_ENGIT/compendio-codigos-ic_v2/

```

```
Estimaciones/ENCIG/Tasa_incidencia_corrupcion_2023.xlsx",  
overwrite = TRUE)
```

5. Cálculo de los IC cuya fuente de información es la Encuesta Nacional de Población Privada de la Libertad (ENPOL) 2021.

Actualmente, en el CNI hay 4 IC familia cuya fuente de información es la ENPOL.

5.1 Limpieza, instalación de paquetes, librerías, lectura de las bases de datos y ajustes de algunas variables de interés.

```
# Limpieza del entorno de R  
rm(list=ls())
```

Se instalan los paquetes foreign, openxlsx y survey los cuales contienen las librerías de interés.

```
packages<-c("foreign", "openxlsx", "survey")  
if (length(setdiff(packages, rownames(installed.packages()))) > 0) {  
  install.packages(setdiff(packages, rownames(installed.packages())))  
}  
lapply(packages, require, character.only=TRUE)  
  
# Libreria utilizada  
library(foreign) # Lee datos almacenados en un formato externo  
library(survey) # Análisis de encuestas complejas  
# Opción para tratar los casos de los estratos con una sola una UPM  
options(survey.lonely.psu="adjust")
```

Se define el directorio de trabajo con la ubicación de la base de datos de la ENPOL 2021. Las bases de datos se obtienen en el siguiente enlace: **microdatos ENPOL 2021**.

```
# Se define el directorio  
getwd()  
setwd("D:/OneDrive - INEGI/CÓDIGOS_EN GIT/compendio-codigos-ic_v2")  
getwd()  
  
# Se leen las bases de datos  
Sec_1 <- read.dbf("bd_enpol_2021_dbf/ENPOL2021_SOC.dbf")  
Sec_23 <- read.dbf("bd_enpol_2021_dbf/ENPOL2021_2_3.dbf")  
Sec_4 <- read.dbf("bd_enpol_2021_dbf/ENPOL2021_4.dbf")  
Sec_5 <- read.dbf("bd_enpol_2021_dbf/ENPOL2021_5.dbf")  
Sec_6 <- read.dbf("bd_enpol_2021_dbf/ENPOL2021_6.dbf")  
Sec_7 <- read.dbf("bd_enpol_2021_dbf/ENPOL2021_7.dbf")  
Sec_891011 <- read.dbf("bd_enpol_2021_dbf/ENPOL2021_8_9_10_11.dbf")
```

Se convierten a tipo numérico los factores de expansión, para poder hacer multiplicaciones.

```
# Cambiar el tipo de variable a numérico  
Sec_1$FAC_PER <-as.numeric(as.character(Sec_1$FAC_PER))  
Sec_1$FPC <-as.numeric(as.character(Sec_1$FPC))  
Sec_23$FAC_PER <-as.numeric(as.character(Sec_23$FAC_PER))  
Sec_23$FPC <-as.numeric(as.character(Sec_23$FPC))  
Sec_4$FAC_PER <-as.numeric(as.character(Sec_4$FAC_PER))  
Sec_4$FPC <-as.numeric(as.character(Sec_4$FPC))
```

```

Sec_5$FAC_PER <-as.numeric(as.character(Sec_5$FAC_PER))
Sec_5$FPC <-as.numeric(as.character(Sec_5$FPC))
Sec_6$FAC_PER <-as.numeric(as.character(Sec_6$FAC_PER))
Sec_6$FPC <-as.numeric(as.character(Sec_6$FPC))
Sec_7$FAC_PER <-as.numeric(as.character(Sec_7$FAC_PER))
Sec_7$FPC <-as.numeric(as.character(Sec_7$FPC))
Sec_891011$FAC_PER <-as.numeric(as.character(Sec_891011$FAC_PER))
Sec_891011$FPC <-as.numeric(as.character(Sec_891011$FPC))

```

5.2 Victimización en Centros Penitenciarios.

Publicado en el DOF el 4-11-2022.

5.2.1 Victimización en Centros Penitenciarios Federales, **total**.

```

#Paso 1: Se ejecuta la función: function2_nac.

#Paso 2: Se definen las variables de interés

#Se nombra a la base de datos con la que se trabaja
x<-Sec_7

#Numerador: Personas privadas de la libertad de 18 años y más que fueron
#víctimas de algún delito en el Centro Penitenciario Federal.
x$numerador<- ifelse((x$P7_40_1%in%1 | x$P7_40_2%in%1|x$P7_40_3%in%1
                    |x$P7_40_4%in%1|x$P7_40_5%in%1|x$P7_40_6%in%1)
                    &(x$FUERO%in%1),1,0)

#Denominador: Personas privadas de la libertad de 18 años y más en el
#Centro Penitenciario Federal.
x$denominador <- ifelse(!(x$FAC_PER%in%"0")&(x$FUERO%in%1),1,0)

#Paso 3: Se define el diseño de la encuesta
DIS <- svydesign(id=~1,strata=~EST_DIS, fpc=~FPC, data=x, weights=~FAC_PER)

#Paso 4: Se manda a llamar la función antes mencionada
tab_vic_deli_cfed_t<-function2_nac(x,numerador,denominador,DIS)

#Paso 5: Se integra la tabla a nivel nacional
#Se asigna el nombre de los encabezados
colnames(tab_vic_deli_cfed_t) <- c("nacional","EST_TVSN","SE_TVSN","CV_TVSN",
                                "LI_TVSN","LS_TVSN","EST_TVSN_N",
                                "SE_TVSN_N","CV_TVSN_N","LI_TVSN_N",
                                "LS_TVSN_N","EST_TVSN_D","SE_TVSN_D",
                                "CV_TVSN_D","LI_TVSN_D","LS_TVSN_D")

```

5.2.2 Victimización en Centros Penitenciarios Federales, **hombres**.

```

#Paso 1: Se ejecuta la función: function2_nac.

#Paso 2: Se definen las variables de interés

#Se nombra a la base de datos con la que se trabaja

```

```

x<-Sec_7

#Numerador: Personas privadas de la libertad de 18 años y más que fueron
#víctimas de algún delito en el Centro Penitenciario Federal, hombres.
x$numerador<- ifelse((x$P7_40_1%in%1|x$P7_40_2%in%1|x$P7_40_3%in%1
|x$P7_40_4%in%1|x$P7_40_5%in%1|x$P7_40_6%in%1)
&(x$FUERO%in%1)&(x$SEXO%in%1),1,0)

#Denominador: Personas privadas de la libertad de 18 años y más en el
#Centro Penitenciario Federal, hombres.
x$denominador <- ifelse((!x$FAC_PER%in%"0")&(x$FUERO%in%1)&(x$SEXO%in%1),1,0)

#Paso 3: Se define el diseño de la encuesta
DIS <- svydesign(id=~1,strata=~EST_DIS, fpc=~FPC, data=x, weights=~FAC_PER)

#Paso 4: Se manda a llamar la función antes mencionada
tab_vic_deli_cfed_h<-function2_nac(x,numerador,denominador,DIS)

#Paso 5: Se integra la tabla a nivel nacional
#Se asigna el nombre de los encabezados
colnames(tab_vic_deli_cfed_h) <- c("nacional","EST_TVSN","SE_TVSN","CV_TVSN",
"LI_TVSN","LS_TVSN","EST_TVSN_N",
"SE_TVSN_N","CV_TVSN_N","LI_TVSN_N",
"LS_TVSN_N","EST_TVSN_D","SE_TVSN_D",
"CV_TVSN_D","LI_TVSN_D","LS_TVSN_D")

```

5.2.3 Victimización en Centros Penitenciarios Federales, mujeres.

```

#Paso 1: Se ejecutan las funciones: function2_nac y function1_cv.

#Paso 2: Se definen las variables de interés

#Se nombra a la base de datos con la que se trabaja
x<-Sec_7

#Numerador: Personas privadas de la libertad de 18 años y más que fueron
#víctimas de algún delito en el Centro Penitenciario Federal, mujeres.
x$numerador<- ifelse((x$P7_40_1%in%1|x$P7_40_2%in%1|x$P7_40_3%in%1
|x$P7_40_4%in%1|x$P7_40_5%in%1|x$P7_40_6%in%1)
&(x$FUERO%in%1)&(x$SEXO%in%2),1,0)

#Denominador: Personas privadas de la libertad de 18 años y más en el
#Centro Penitenciario Federal, mujeres.
x$denominador <- ifelse((!x$FAC_PER%in%"0")&(x$FUERO%in%1)&(x$SEXO%in%2),1,0)

#Paso 3: Se define el diseño de la encuesta
DIS <- svydesign(id=~1,strata=~EST_DIS, fpc=~FPC, data=x, weights=~FAC_PER)

#Paso 4: Se manda a llamar la función antes mencionada
tab_vic_deli_cfed_m<-function2_nac(x,numerador,denominador,DIS)

#Paso 5: Se integra la tabla a nivel nacional
#Se asigna el nombre de los encabezados

```

```

colnames(tab_vic_delicfed_m) <- c("nacional", "EST_TVSN", "SE_TVSN", "CV_TVSN",
                                "LI_TVSN", "LS_TVSN", "EST_TVSN_N",
                                "SE_TVSN_N", "CV_TVSN_N", "LI_TVSN_N",
                                "LS_TVSN_N", "EST_TVSN_D", "SE_TVSN_D",
                                "CV_TVSN_D", "LI_TVSN_D", "LS_TVSN_D")

#Paso 6: Exportar salida a un archivo de Excel
#Se exportará el indicador con sus desagregaciones por sexo.

# Crear un nuevo workbook
wb <- createWorkbook()

#Se agregan las hojas
addWorksheet( wb,"vic_delicfed_t")
addWorksheet( wb,"vic_delicfed_h")
addWorksheet( wb,"vic_delicfed_m")
writeData(wb,"vic_delicfed_t",tab_vic_delicfed_t)
writeData(wb,"vic_delicfed_h",tab_vic_delicfed_h)
writeData(wb,"vic_delicfed_m",tab_vic_delicfed_m)

#Se definen las hojas y filas a aplicar el formato condicional
sheets <- c("vic_delicfed_t", "vic_delicfed_h", "vic_delicfed_m")
rows <- 2

# Aplica el formato condicional
wb <- function1_cv(wb, sheets, cols_rules, rows)

#Se elige directorio para exportar los archivos
saveWorkbook(wb, file= "D:/OneDrive - INEGI/CÓDIGOS_EN GIT/
                    compendio-codigos-ic_v2/Estimaciones/ENPOL/
                    indica_vic_delicfed_2021.xlsx",overwrite = TRUE)

```

5.2.4 Victimización en Centros Penitenciarios Estatales, total.

```

#Paso 1: Se ejecutan las funciones: function2_nac y function2_ent.

#Paso 2: Se definen las variables de interés

#Se nombra a la base de datos con la que se trabaja
x<-Sec_7

#Numerador: Personas privadas de la libertad de 18 años y más que fueron
#víctimas de algún delito en Centros Penitenciarios Estatales.
x$numerador<- ifelse((x$P7_40_1%in%1 | x$P7_40_2%in%1|x$P7_40_3%in%1
                    |x$P7_40_4%in%1|x$P7_40_5%in%1|x$P7_40_6%in%1)
                    &(x$FUERO%in%2),1,0)

#Denominador: Personas privadas de la libertad de 18 años y más en los
#Centros Penitenciarios Estatales.
x$denominador <- ifelse((!x$FAC_PER%in%"0")&(x$FUERO%in%2),1,0)

#Paso 3: Se define el diseño de la encuesta
DIS <- svydesign(id=~1,strata=~EST_DIS, fpc=~FPC, data=x, weights=~FAC_PER)

```



```

#Paso 4: Se mandan a llamar las funciones antes mencionadas
tab_vic_deli_cest_nac_t<-function2_nac(x,numerador,denominador,DIS)
tab_vic_deli_cest_ent_t<-function2_ent(x,numerador,denominador,DIS)

#Paso 5: Se integran las tablas del nacional y por entidad federativa
#Se asigna el mismo nombre de los encabezados
names(tab_vic_deli_cest_nac_t)<-names(tab_vic_deli_cest_ent_t)
#Unión de tablas
tab_vic_deli_cest_t<- rbind(tab_vic_deli_cest_nac_t,tab_vic_deli_cest_ent_t)

```

5.2.5 Victimización en Centros Penitenciarios Estatales, **hombres**.

```

#Paso 1: Se ejecutan las funciones: function2_nac y function2_ent.

#Paso 2: Se definen las variables de interés

#Se nombra a la base de datos con la que se trabaja
x<-Sec_7

#Numerador: Personas privadas de la libertad de 18 años y más que fueron
#víctimas de algún delito en Centros Penitenciarios Estatales, hombres.
x$numerador<- ifelse((x$P7_40_1%in%1|x$P7_40_2%in%1|x$P7_40_3%in%1
|x$P7_40_4%in%1|x$P7_40_5%in%1|x$P7_40_6%in%1)
&(x$FUERO%in%2)&(x$SEXO%in%1),1,0)

#Denominador: Personas privadas de la libertad de 18 años y más en los
#Centros Penitenciarios Estatales, hombres.
x$denominador <- ifelse((!x$FAC_PER%in%"0")&(x$FUERO%in%2)&(x$SEXO%in%1),1,0)

#Paso 3: Se define el diseño de la encuesta
DIS <- svydesign(id=~1,strata=~EST_DIS, fpc=~FPC, data=x, weights=~FAC_PER)

#Paso 4: Se mandan a llamar las funciones antes mencionadas
tab_vic_deli_cest_nac_h<-function2_nac(x,numerador,denominador,DIS)
tab_vic_deli_cest_ent_h<-function2_ent(x,numerador,denominador,DIS)
#Paso 5: Se integran las tablas del nacional y por entidad federativa

#Se asigna el mismo nombre de los encabezados
names(tab_vic_deli_cest_nac_h)<-names(tab_vic_deli_cest_ent_h)
#Unión de tablas
tab_vic_deli_cest_h<- rbind(tab_vic_deli_cest_nac_h,tab_vic_deli_cest_ent_h)

```

5.2.6 Victimización en Centros Penitenciarios Estatales, **mujeres**.

```

#Paso 1: Se ejecutan las funciones: function2_nac, function2_ent y function1_cv.

#Paso 2: Se definen las variables de interés

#Se nombra a la base de datos con la que se trabaja
x<-Sec_7

#Numerador: Personas privadas de la libertad de 18 años y más que fueron

```

```

#víctimas de algún delito en Centros Penitenciarios Estatales, mujeres.
x$numerador<- ifelse((x$P7_40_1%in%1|x$P7_40_2%in%1|x$P7_40_3%in%1
|x$P7_40_4%in%1|x$P7_40_5%in%1|x$P7_40_6%in%1)
&(x$FUERO%in%2)&(x$SEXO%in%2),1,0)

#Denominador: Personas privadas de la libertad de 18 años y más en los
#Centros Penitenciarios Estatales, mujeres.
x$denominador <- ifelse((!x$FAC_PER%in%"0")&(x$FUERO%in%2)&(x$SEXO%in%2),1,0)

#Paso 3: Se define el diseño de la encuesta
DIS <- svydesign(id=~1,strata=~EST_DIS, fpc=~FPC, data=x, weights=~FAC_PER)

#Paso 4: Se mandan a llamar las funciones antes mencionadas
tab_vic_deli_cest_nac_m<-function2_nac(x,numerador,denominador,DIS)
tab_vic_deli_cest_ent_m<-function2_ent(x,numerador,denominador,DIS)

#Paso 5: Se integran las tablas del nacional y por entidad federativa
#Se asigna el mismo nombre de los encabezados
names(tab_vic_deli_cest_nac_m)<-names(tab_vic_deli_cest_ent_m)
#Unión de tablas
tab_vic_deli_cest_m<- rbind(tab_vic_deli_cest_nac_m,tab_vic_deli_cest_ent_m)

#Paso 6: Exportar salida a un archivo de Excel
#Se exportará el indicador con sus desagregaciones por sexo.

# Crear un nuevo workbook
wb <- createWorkbook()

#Se agregan las hojas
addWorksheet( wb,"vic_deli_cest_t")
addWorksheet( wb,"vic_deli_cest_h")
addWorksheet( wb,"vic_deli_cest_m")
writeData(wb,"vic_deli_cest_t",tab_vic_deli_cest_t)
writeData(wb,"vic_deli_cest_h",tab_vic_deli_cest_h)
writeData(wb,"vic_deli_cest_m",tab_vic_deli_cest_m)

#Se definen las hojas y filas a aplicar el formato condicional
sheets <- c("vic_deli_cest_t", "vic_deli_cest_h", "vic_deli_cest_m")
rows <- 2:34

# Aplica el formato condicional
wb <- function1_cv(wb, sheets, cols_rules, rows)

#Se elige directorio para exportar los archivos
saveWorkbook(wb, file= "D:/OneDrive - INEGI/CÓDIGOS_EN GIT/
compendio-codigos-ic_v2/Estimaciones/ENPOL/
indica_vic_deli_CE_2021.xlsx", overwrite = TRUE)

```

6. Cálculo de los IC cuya fuente de información es la Encuesta Nacional sobre la Dinámica de las Relaciones en los Hogares (ENDIREH) 2021

Actualmente, en el CNI hay 8 IC familia cuya fuente de información es la ENDIREH.

6.1 Limpieza, instalación de paquetes, librerías, lectura de las bases de datos y ajustes de algunas variables de interés.

```
# Limpieza del entorno de R  
rm(list=ls())
```

Se instalan los paquetes foreign, openxlsx y survey los cuales contienen las librerías de interés.

```
packages<-c("foreign","survey", "openxlsx","dplyr")  
if (length(setdiff(packages, rownames(installed.packages()))) > 0) {  
  install.packages(setdiff(packages, rownames(installed.packages())))  
}  
lapply(packages, require, character.only=TRUE)  
  
# Libreria utilizada  
library(foreign) # Lee datos almacenados en un formato externo  
library(survey) # Análisis de encuestas complejas  
library(dplyr) #Facilita la manipulación y operaciones con data frames
```

Se define el directorio de trabajo con la ubicación de la base de datos de la ENDIREH 2021. Las bases de datos se obtienen en el siguiente enlace: **microdatos ENDIREH 2021**.

```
# Se define el directorio  
getwd()  
setwd("D:/OneDrive - INEGI/CÓDIGOS_EN GIT/compendio-codigos-ic_v2")  
# Se leen las bases de datos  
load("bd_endireh_2021.RData")
```

Se convierten a tipo numérico los factores de expansión, para poder hacer multiplicaciones.

```
# Cambiar el tipo de variable a numérico  
TB_SEC_IVaVD$FAC_MUJ <- as.numeric(as.character(TB_SEC_IVaVD$FAC_MUJ))  
TSDem$EDAD <- as.numeric(as.character(TSDem$EDAD))
```

Para el cálculo de los indicadores por grupo de edad se integra la variable edad en la base de datos principal (TB_SEC_IVaVD), así mismo se crean los grupos de edad y la variable “Condición conyugal”, esto se realiza de la siguiente manera:

```
#Se agrega la variable edad a la tabla "TB_SEC_IVaVD"  
TB_SEC_IVaVD <- merge(TB_SEC_IVaVD,TSDem[,c("ID_PER","EDAD")],by="ID_PER",all.x = T)  
####Se realizan los grupos de edad  
TB_SEC_IVaVD$grp_edad<- cut(TB_SEC_IVaVD$EDAD, breaks = c(15, 17, 19, 24, 29, 34, 39, 44, 49, 54, 59,97)  
#Se construye la variable "Condición conyugal"  
TB_SEC_IVaVD$Condicion_conyugal[TB_SEC_IVaVD$T_INSTRUM == "A1"|TB_SEC_IVaVD$T_INSTRUM == "A2"] <- 1  
TB_SEC_IVaVD$Condicion_conyugal[TB_SEC_IVaVD$T_INSTRUM == "B1"|TB_SEC_IVaVD$T_INSTRUM == "B2"] <- 2  
TB_SEC_IVaVD$Condicion_conyugal[TB_SEC_IVaVD$T_INSTRUM == "C1"] <- 3
```

6.2 Prevalencia de la violencia contra las mujeres de 15 años y más a lo largo de su vida.

Publicado en el DOF el 18-12-2013.

6.2.1. Prevalencia de la violencia contra las mujeres de 15 años y más a lo largo de su vida.

#Paso 1: Se ejecutan las funciones: function2_nac, function2_ent y function1_cv.

#Paso 2: Se definen las variables de interés

#Se nombra a la base de datos con la que se trabaja

`x<-TB_SEC_IVaVD`

*#Numerador: Mujeres de 15 años y más que han sufrido al menos un incidente de
#violencia de cualquier tipo y en cualquier ámbito, a lo largo de su vida,
#ejercida por cualquier agresor.*

```
x$numerador<- ifelse(#ÁMBITO ESCOLAR##
  (x$P7_6_1%in%1)|(x$P7_6_2%in%1)|(x$P7_6_3%in%1)|
  (x$P7_6_4%in%1)|(x$P7_6_5%in%1)|(x$P7_6_6%in%1)|
  (x$P7_6_7%in%1)|(x$P7_6_8%in%1)|(x$P7_6_9%in%1)|
  (x$P7_6_10%in%1)|(x$P7_6_11%in%1)|(x$P7_6_12%in%1)|
  (x$P7_6_13%in%1)|(x$P7_6_14%in%1)|(x$P7_6_15%in%1)|
  (x$P7_6_16%in%1)|(x$P7_6_17%in%1)|(x$P7_6_18%in%1)|
  ##ÁMBITO OBSTETRICO##
  (x$P8_3_1_1%in%1)|(x$P8_3_1_2%in%1)|(x$P8_3_2_1%in%1)|
  (x$P8_3_2_2%in%1)|(x$P8_3_2_3%in%1)|(x$P8_8_1%in%1)|
  (x$P8_8_2%in%1)|(x$P8_8_3%in%1)|(x$P8_8_4%in%1)|
  (x$P8_8_5%in%1)|(x$P8_8_6%in%1)|(x$P8_8_7%in%1)|
  (x$P8_8_8%in%1)|(x$P8_8_9%in%1)|
  #ÁMBITO LABORAL##
  (x$P8_9_1%in%1)|(x$P8_9_2%in%1)|(x$P8_9_3%in%1)|
  (x$P8_9_4%in%1)|(x$P8_9_5%in%1)|(x$P8_9_6%in%1)|
  (x$P8_9_7%in%1)|(x$P8_9_8%in%1)|(x$P8_9_9%in%1)|
  (x$P8_9_10%in%1)|(x$P8_9_11%in%1)|(x$P8_9_12%in%1)|
  (x$P8_9_13%in%1)|(x$P8_9_14%in%1)|(x$P8_9_15%in%1)|
  (x$P8_9_16%in%1)|(x$P8_9_17%in%1)|(x$P8_9_18%in%1)|
  (x$P8_9_19%in%1)|
  ##ÁMBITO COMUNITARIO##
  (x$P9_1_1%in%1)|(x$P9_1_2%in%1)|(x$P9_1_3%in%1)|
  (x$P9_1_4%in%1)|(x$P9_1_5%in%1)|(x$P9_1_6%in%1)|
  (x$P9_1_7%in%1)|(x$P9_1_8%in%1)|(x$P9_1_9%in%1)|
  (x$P9_1_10%in%1)|(x$P9_1_11%in%1)|(x$P9_1_12%in%1)|
  (x$P9_1_13%in%1)|(x$P9_1_14%in%1)|(x$P9_1_15%in%1)|
  (x$P9_1_16%in%1)|
  ##ÁMBITO FAMILIAR#
  (x$P11_1_1%in%c(1,2,3))|(x$P11_1_2%in%c(1,2,3))|
  (x$P11_1_3%in%c(1,2,3))|(x$P11_1_4%in%c(1,2,3))|
  (x$P11_1_5%in%c(1,2,3))|(x$P11_1_6%in%c(1,2,3))|
  (x$P11_1_7%in%c(1,2,3))|(x$P11_1_8%in%c(1,2,3))|
  (x$P11_1_9%in%c(1,2,3))|(x$P11_1_10%in%c(1,2,3))|
  (x$P11_1_11%in%c(1,2,3))|(x$P11_1_12%in%c(1,2,3))|
  (x$P11_1_13%in%c(1,2,3))|(x$P11_1_14%in%c(1,2,3))|
  (x$P11_1_15%in%c(1,2,3))|(x$P11_1_16%in%c(1,2,3))|
  (x$P11_1_17%in%c(1,2,3))|(x$P11_1_18%in%c(1,2,3))|
  (x$P11_1_19%in%c(1,2,3))|(x$P11_1_20%in%c(1,2,3))|
```

```

#ÁMBITO PAREJA###
(x$P14_1_1%in%c(1,2,3))|(x$P14_1_2%in%c(1,2,3))|
(x$P14_1_3%in%c(1,2,3))|(x$P14_1_4%in%c(1,2,3))|
(x$P14_1_5%in%c(1,2,3))|(x$P14_1_6%in%c(1,2,3))|
(x$P14_1_7%in%c(1,2,3))|(x$P14_1_8%in%c(1,2,3))|
(x$P14_1_9%in%c(1,2,3))|(x$P14_1_10%in%c(1,2,3))|
(x$P14_1_11%in%c(1,2,3))|(x$P14_1_12%in%c(1,2,3))|
(x$P14_1_13%in%c(1,2,3))|(x$P14_1_14%in%c(1,2,3))|
(x$P14_1_15%in%c(1,2,3))|(x$P14_1_16%in%c(1,2,3))|
(x$P14_1_17%in%c(1,2,3))|(x$P14_1_18%in%c(1,2,3))|
(x$P14_1_19%in%c(1,2,3))|(x$P14_1_20%in%c(1,2,3))|
(x$P14_1_21%in%c(1,2,3))|(x$P14_1_22%in%c(1,2,3))|
(x$P14_1_23AB%in%c(1,2,3))|(x$P14_1_24AB%in%c(1,2,3))|
(x$P14_1_25%in%c(1,2,3))|(x$P14_1_26%in%c(1,2,3))|
(x$P14_1_27%in%c(1,2,3))|(x$P14_1_28%in%c(1,2,3))|
(x$P14_1_29%in%c(1,2,3))|(x$P14_1_30%in%c(1,2,3))|
(x$P14_1_31%in%c(1,2,3))|(x$P14_1_32%in%c(1,2,3))|
(x$P14_1_33%in%c(1,2,3))|(x$P14_1_34%in%c(1,2,3))|
(x$P14_1_35AB%in%c(1,2,3))|(x$P14_1_36AB%in%c(1,2,3))|
(x$P14_1_37AB%in%c(1,2,3))|(x$P14_1_38AB%in%c(1,2,3)),1,0)

#Denominador: Total de mujeres de 15 años y más.
x$denominador<-ifelse(!x$FAC_MUJ%in%"0",1,0)

#Paso 3: Se define el diseño de la encuesta
DIS <- svydesign(id=~UPM_DIS,strata=~EST_DIS,data=x,weights=~FAC_MUJ,nest=TRUE)

#Paso 4: Se mandan a llamar las funciones antes mencionadas
tab_v_tot_alv_nac<-function2_nac(x,numerador ,denominador,DIS)
tab_v_tot_alv_ent<-function2_ent(x,numerador ,denominador,DIS)

#Paso 5: Se integran las tablas del nacional y por entidad federativa
#Se asigna el mismo nombre de los encabezados
names(tab_v_tot_alv_nac)<-names(tab_v_tot_alv_ent)
#Unión de tablas
tab_v_tot_alv<- rbind(tab_v_tot_alv_nac,tab_v_tot_alv_ent)

#Paso 6: Exportar salida a un archivo de Excel
#Se exportará el indicador con sus desagregaciones.

# Crear un nuevo workbook
wb <- createWorkbook()

#Se agregan las hojas
addWorksheet( wb,"v_tot_alv")
writeData(wb,"v_tot_alv",tab_v_tot_alv)

#Se definen las hojas y filas a aplicar el formato condicional
sheets <- c("v_tot_alv")
rows <- 2:34

# Aplica el formato condicional
wb <- function1_cv(wb, sheets, cols_rules, rows)

```

```
#Se elige directorio para exportar los archivos
saveWorkbook(wb, file= "D:/OneDrive - INEGI/CÓDIGOS_EN GIT/
compendio-codigos-ic_v2/Estimaciones/ENDIREH/
Preval_viol_total_alv_2021.xlsx", overwrite = TRUE)
```

6.2.2. Prevalencia de la violencia contra las mujeres de 15 años y más a lo largo de su vida, por grupo de edad.

#Paso 1: Se ejecutan las funciones: function2_nac, function7_edad y function1_cv.

#Paso 2: Se definen las variables de interés

#Se nombra a la base de datos con la que se trabaja

```
x<-TB_SEC_IVaVD
```

*#Numerador: Mujeres de 15 años y más que han sufrido al menos un incidente de
#violencia de cualquier tipo y en cualquier ámbito, a lo largo de su vida,
#ejercida por cualquier agresor, por grupo de edad.*

```
x$numerador<- ifelse(#ÁMBITO ESCOLAR##
(x$P7_6_1%in%1)|(x$P7_6_2%in%1)|(x$P7_6_3%in%1)|
(x$P7_6_4%in%1)|(x$P7_6_5%in%1)|(x$P7_6_6%in%1)|
(x$P7_6_7%in%1)|(x$P7_6_8%in%1)|(x$P7_6_9%in%1)|
(x$P7_6_10%in%1)|(x$P7_6_11%in%1)|(x$P7_6_12%in%1)|
(x$P7_6_13%in%1)|(x$P7_6_14%in%1)|(x$P7_6_15%in%1)|
(x$P7_6_16%in%1)|(x$P7_6_17%in%1)|(x$P7_6_18%in%1)|
##ÁMBITO OBSTETRICO##
(x$P8_3_1_1%in%1)|(x$P8_3_1_2%in%1)|(x$P8_3_2_1%in%1)|
(x$P8_3_2_2%in%1)|(x$P8_3_2_3%in%1)|(x$P8_8_1%in%1)|
(x$P8_8_2%in%1)|(x$P8_8_3%in%1)|(x$P8_8_4%in%1)|
(x$P8_8_5%in%1)|(x$P8_8_6%in%1)|(x$P8_8_7%in%1)|
(x$P8_8_8%in%1)|(x$P8_8_9%in%1)|
#ÁMBITO LABORAL##
(x$P8_9_1%in%1)|(x$P8_9_2%in%1)|(x$P8_9_3%in%1)|
(x$P8_9_4%in%1)|(x$P8_9_5%in%1)|(x$P8_9_6%in%1)|
(x$P8_9_7%in%1)|(x$P8_9_8%in%1)|(x$P8_9_9%in%1)|
(x$P8_9_10%in%1)|(x$P8_9_11%in%1)|(x$P8_9_12%in%1)|
(x$P8_9_13%in%1)|(x$P8_9_14%in%1)|(x$P8_9_15%in%1)|
(x$P8_9_16%in%1)|(x$P8_9_17%in%1)|(x$P8_9_18%in%1)|
(x$P8_9_19%in%1)|
##ÁMBITO COMUNITARIO##
(x$P9_1_1%in%1)|(x$P9_1_2%in%1)|(x$P9_1_3%in%1)|
(x$P9_1_4%in%1)|(x$P9_1_5%in%1)|(x$P9_1_6%in%1)|
(x$P9_1_7%in%1)|(x$P9_1_8%in%1)|(x$P9_1_9%in%1)|
(x$P9_1_10%in%1)|(x$P9_1_11%in%1)|(x$P9_1_12%in%1)|
(x$P9_1_13%in%1)|(x$P9_1_14%in%1)|(x$P9_1_15%in%1)|
(x$P9_1_16%in%1)|
##ÁMBITO FAMILIAR#
(x$P11_1_1%in%c(1,2,3))|(x$P11_1_2%in%c(1,2,3))|
(x$P11_1_3%in%c(1,2,3))|(x$P11_1_4%in%c(1,2,3))|
(x$P11_1_5%in%c(1,2,3))|(x$P11_1_6%in%c(1,2,3))|
(x$P11_1_7%in%c(1,2,3))|(x$P11_1_8%in%c(1,2,3))|
(x$P11_1_9%in%c(1,2,3))|(x$P11_1_10%in%c(1,2,3))|
(x$P11_1_11%in%c(1,2,3))|(x$P11_1_12%in%c(1,2,3))|
```

```

(x$P11_1_13%in%c(1,2,3))|(x$P11_1_14%in%c(1,2,3))|
(x$P11_1_15%in%c(1,2,3))|(x$P11_1_16%in%c(1,2,3))|
(x$P11_1_17%in%c(1,2,3))|(x$P11_1_18%in%c(1,2,3))|
(x$P11_1_19%in%c(1,2,3))|(x$P11_1_20%in%c(1,2,3))|
#ÁMBITO PAREJA###
(x$P14_1_1%in%c(1,2,3))|(x$P14_1_2%in%c(1,2,3))|
(x$P14_1_3%in%c(1,2,3))|(x$P14_1_4%in%c(1,2,3))|
(x$P14_1_5%in%c(1,2,3))|(x$P14_1_6%in%c(1,2,3))|
(x$P14_1_7%in%c(1,2,3))|(x$P14_1_8%in%c(1,2,3))|
(x$P14_1_9%in%c(1,2,3))|(x$P14_1_10%in%c(1,2,3))|
(x$P14_1_11%in%c(1,2,3))|(x$P14_1_12%in%c(1,2,3))|
(x$P14_1_13%in%c(1,2,3))|(x$P14_1_14%in%c(1,2,3))|
(x$P14_1_15%in%c(1,2,3))|(x$P14_1_16%in%c(1,2,3))|
(x$P14_1_17%in%c(1,2,3))|(x$P14_1_18%in%c(1,2,3))|
(x$P14_1_19%in%c(1,2,3))|(x$P14_1_20%in%c(1,2,3))|
(x$P14_1_21%in%c(1,2,3))|(x$P14_1_22%in%c(1,2,3))|
(x$P14_1_23AB%in%c(1,2,3))|(x$P14_1_24AB%in%c(1,2,3))|
(x$P14_1_25%in%c(1,2,3))|(x$P14_1_26%in%c(1,2,3))|
(x$P14_1_27%in%c(1,2,3))|(x$P14_1_28%in%c(1,2,3))|
(x$P14_1_29%in%c(1,2,3))|(x$P14_1_30%in%c(1,2,3))|
(x$P14_1_31%in%c(1,2,3))|(x$P14_1_32%in%c(1,2,3))|
(x$P14_1_33%in%c(1,2,3))|(x$P14_1_34%in%c(1,2,3))|
(x$P14_1_35AB%in%c(1,2,3))|(x$P14_1_36AB%in%c(1,2,3))|
(x$P14_1_37AB%in%c(1,2,3))|(x$P14_1_38AB%in%c(1,2,3)),1,0)

```

#Denominador: Total de mujeres de 15 años y más, por grupo de edad.

```
x$denominador<-ifelse(!x$FAC_MUJ%in%"0",1,0)
```

#Paso 3: Se define el diseño de la encuesta

```
DIS <- svydesign(id=~UPM_DIS,strata=~EST_DIS,data=x,weights=~FAC_MUJ,nest=TRUE)
```

#Paso 4: Se mandan a llamar las funciones antes mencionadas

```
tab_v_tot_alv_nac<-function2_nac(x,numerador ,denominador,DIS)
```

```
tab_v_tot_alv_edad<-function7_edad(x,numerador ,denominador,DIS)
```

#Paso 5: Se integran las tablas del nacional y por grupo de edad

#Se asigna el mismo nombre de los encabezados

```
names(tab_v_tot_alv_nac)<-names(tab_v_tot_alv_edad)
```

#Unión de tablas

```
tab_v_tot_alv_grp_edad<- rbind(tab_v_tot_alv_nac,tab_v_tot_alv_edad)
```

#Paso 6: Exportar salida a un archivo de Excel

#Se exportará el indicador con sus desagregaciones.

Crear un nuevo workbook

```
wb <- createWorkbook()
```

#Se agregan las hojas

```
addWorksheet( wb,"v_tot_alv_grp_edad")
```

```
writeData(wb,"v_tot_alv_grp_edad",tab_v_tot_alv_grp_edad)
```

#Se definen las hojas y filas a aplicar el formato condicional

```
sheets <- c("v_tot_alv_grp_edad")
```

```
rows <- 2:14

# Aplica el formato condicional
wb <- function1_cv(wb, sheets, cols_rules, rows)

#Se elige directorio para exportar los archivos
saveWorkbook(wb, file= "D:/OneDrive - INEGI/CÓDIGOS_EN GIT/
    compendio-codigos-ic_v2/Estimaciones/ENDIREH/
    Preval_viol_total_alv_edad_2021.xlsx", overwrite = TRUE)
```


7. Cálculo de los IC cuya fuente de información son los registros administrativos de las defunciones registradas.

Actualmente, en el CNI hay 2 IC familia que tienen como fuentes de información a los registros administrativos sobre las estadísticas de mortalidad (defunciones registradas) y la Conciliación Demográfica de 1950 a 2019 y Proyecciones de la población de México y de las entidades federativas 2020 a 2070 de CONAPO.

En términos generales, el cálculo de un indicador se resume en los siguientes pasos:

1. Se construyen las variables que constituyen el numerador y el denominador para el cálculo del IC.
2. Se calcula el IC a nivel nacional y por sus desagregaciones (entidad federativa, sexo y en algunos casos por edad).
3. Se ordena la información de tal manera que cada IC único (desagregado) esté contenido en una tabla con la(s) variable(s) que fungen como numerador y la(s) variable(s) que fungen como denominador, asimismo, la primera fila tendrá la estimación nacional y en las siguientes filas las estimaciones ordenadas según su desagregación.
4. Se exportan las tablas a un archivo de Excel.

7.1 Limpieza, instalación de paquetes, librerías.

```
# Limpieza del entorno de R
rm(list=ls())
```

Se instalan los paquetes pacman, tidyverse, readr, foreign, openxlsx, readxl y data.table los cuales contienen las librerías de interés.

A continuación se describen las características de cada paquete:

- a. **Pacman** es una herramienta de gestión de paquetes R que combina la funcionalidad de las funciones relacionadas con la biblioteca base en funciones con nombres intuitivos.
- b. **Tidyverse** es una colección de paquetes de R, entre los que se encuentran dplyr, tidyr, stringr y readr, estos permiten un mejor procesamiento de los datos.
- c. **Readr** funciona como un lector de datos en formato csv y valores separados por tabulaciones.
- d. **foreign** permite realizar la lectura y escritura de datos almacenados por algunos programas externos.
- e. **openxlsx** contiene funciones para simplificar la creación de archivos Excel, su escritura, edición y modificación desde R.
- f. **readxl** permite importar archivos de excel a R.
- g. **data.table** permite el trabajo con datos de gran tamaño, uniones, modificaciones y eliminación.

```
packages<-c("pacman", "foreign", "openxlsx", "tidyverse", "readr", "readxl", "data.table")
if (length(setdiff(packages, rownames(installed.packages()))) > 0) {
  install.packages(setdiff(packages, rownames(installed.packages())))
}
lapply(packages, require, character.only=TRUE)

# Libreria utilizada
library(foreign) # Lee datos almacenados en un formato externo
library(openxlsx)
library(tidyverse)
library(readr)
library(readxl)
library(data.table)
```

Se define el directorio de trabajo con la ubicación de la base de datos de las defunciones registradas 2023 y las proyecciones de población a mitad de año de CONAPO. Las bases de datos se obtienen en los siguientes enlaces: **DEFUNCIONES REGISTRADAS 2023** y **POBLACIÓN**

7.2 Tasa bruta anual de defunciones por homicidio por cada cien mil habitantes.

Publicado en el DOF el 18-12-2013 y modificado el 15-02-2022.

7.2.1. Tasa bruta anual de defunciones por homicidio por cada cien mil habitantes.

```
# Se define el directorio
getwd()
setwd("C:/DEFUNCIONES/BD")

# Se leen las bases de datos
#Leemos la base de datos de CONAPO
pob <- read_excel("CONAPO/0_Pob_Mitad_1950_2070.xlsx")

#Se realiza el filtro para trabajar solo con las entidades
pob1 <- subset(pob,(!pob$CVE_GEO==0))

#Convertimos a data frame la base de datos
setDT(pob1)

#Preparamos la base de datos de población

names(pob1) <- tolower(names(pob1))

pob1[sexo == "Hombres", sexo_cod := 1]
pob1[sexo != "Hombres", sexo_cod := 2]

pob1[, .N, by = .(sexo, sexo_cod)]

pob1[, sum(poblacion), by = .(sexo, año)]

#Cálculo de la Tasa bruta anual de defunciones por homicidio por
#cada cien mil habitantes

preg_int <- "causa_def"

#Se definen los años
anios <- 2023
res_entidad <- list()
View()

cont <- 0

for(anio in anios){

  names3 <- paste0("X",850:999)
  names4 <- paste0("Y", formatC(0:99, width = 3, flag = "0"))
  val_int <- c(names3, names4)
```

```

}

cont <- cont + 1
nom_arch_dbf <- paste0("DEFUN", substr(as.character(anio), 3, 4), ".dbf")
arch_base <- file.path(getwd(), "Defunciones", anio, nom_arch_dbf)
base <- read.dbf(arch_base)
names(base) <- tolower(names(base))
setDT(base)
base$ent_ocurr <- as.numeric(as.character(base$ent_ocurr))
base[[preg_int]] <- as.character(base[[preg_int]])

base[, .N, by = causa_def%in%val_int]

base[, .N, by = .(base[[preg_int]]%in%val_int)]

base[, V1 := ifelse(base[[preg_int]]%in%val_int, 1, 0)]

#Cálculo del numerador

base[, entidad := case_when(base$ent_ocurr==1 ~ 'Aguascalientes',#1
                             base$ent_ocurr==2 ~ 'Baja California',#2
                             base$ent_ocurr==3 ~ 'Baja California Sur',#3
                             base$ent_ocurr==4 ~ 'Campeche',#4
                             base$ent_ocurr==5 ~ 'Coahuila',#5
                             base$ent_ocurr==6 ~ 'Colima',#6
                             base$ent_ocurr==7 ~ 'Chiapas',#7
                             base$ent_ocurr==8 ~ 'Chihuahua',#8
                             base$ent_ocurr==9 ~ 'Ciudad de México',#9
                             base$ent_ocurr==10 ~ 'Durango',#10
                             base$ent_ocurr==11 ~ 'Guanajuato',#11
                             base$ent_ocurr==12 ~ 'Guerrero',#12
                             base$ent_ocurr==13 ~ 'Hidalgo',#13
                             base$ent_ocurr==14 ~ 'Jalisco',#14
                             base$ent_ocurr==15 ~ 'México',#15
                             base$ent_ocurr==16 ~ 'Michoacán',#16
                             base$ent_ocurr==17 ~ 'Morelos',#17
                             base$ent_ocurr==18 ~ 'Nayarit',#18
                             base$ent_ocurr==19 ~ 'Nuevo León',#19
                             base$ent_ocurr==20 ~ 'Oaxaca',#20
                             base$ent_ocurr==21 ~ 'Puebla',#21
                             base$ent_ocurr==22 ~ 'Querétaro',#22
                             base$ent_ocurr==23 ~ 'Quintana Roo',#23
                             base$ent_ocurr==24 ~ 'San Luis Potosí',#24
                             base$ent_ocurr==25 ~ 'Sinaloa',#25
                             base$ent_ocurr==26 ~ 'Sonora',#26
                             base$ent_ocurr==27 ~ 'Tabasco',#27
                             base$ent_ocurr==28 ~ 'Tamaulipas',#28
                             base$ent_ocurr==29 ~ 'Tlaxcala',#29
                             base$ent_ocurr==30 ~ 'Veracruz',#30
                             base$ent_ocurr==31 ~ 'Yucatán',#31
                             base$ent_ocurr==32 ~ 'Zacatecas',#32
                             base$ent_ocurr == 33 | base$ent_ocurr == 34 |
                             base$ent_ocurr == 35 | base$ent_ocurr == 99 ~ 'NE'')]

```

```

tbl_sexo1<-base[V1==1&sexo==1,.(H_num=.N),
               by=(entidad)][order(entidad)][!is.na(entidad)&entidad!='NE']
tbl_sexo2<-base[V1==1&sexo==2,.(M_num=.N),
               by=(entidad)][order(entidad)][!is.na(entidad)&entidad!='NE']

tbl_total<-base[V1==1,.(T_num=.N),
               by=(entidad)][order(entidad)][!is.na(entidad)&entidad!='NE']

tbl_final <- Reduce(function(...){merge(..., all = T, by = "entidad")},
                    list(tbl_total,tbl_sexo1,tbl_sexo2))

tbl_nac <- data.table(entidad = 'Nacional',
                      T_num = base[V1 == 1, .N],
                      H_num = base[V1 == 1 & sexo == 1, .N],
                      M_num = base[V1 == 1 & sexo == 2, .N])

tbl_con_total <- rbindlist(list(tbl_nac, tbl_final))

tbl_con_total[str_detect(entidad, regex("^NAC", ignore_case = T)), num_ent := 0]
tbl_con_total[str_detect(entidad, regex("^AG", ignore_case = T)), num_ent := 1]
tbl_con_total[str_detect(entidad, regex("^B.*C.*", ignore_case = T)), num_ent := 2]
tbl_con_total[str_detect(entidad, regex("^B.*C.*S.*", ignore_case = T)), num_ent := 3]
tbl_con_total[str_detect(entidad, regex("^CAM", ignore_case = T)), num_ent := 4]
tbl_con_total[str_detect(entidad, regex("^COA", ignore_case = T)), num_ent := 5]
tbl_con_total[str_detect(entidad, regex("^COL", ignore_case = T)), num_ent := 6]
tbl_con_total[str_detect(entidad, regex("^CHIA", ignore_case = T)), num_ent := 7]
tbl_con_total[str_detect(entidad, regex("^CHIH", ignore_case = T)), num_ent := 8]
tbl_con_total[str_detect(entidad, regex("^Ciudad*", ignore_case = T)), num_ent := 9]
tbl_con_total[str_detect(entidad, regex("^DUR", ignore_case = T)), num_ent := 10]
tbl_con_total[str_detect(entidad, regex("^GUA", ignore_case = T)), num_ent := 11]
tbl_con_total[str_detect(entidad, regex("^GUE", ignore_case = T)), num_ent := 12]
tbl_con_total[str_detect(entidad, regex("^HID", ignore_case = T)), num_ent := 13]
tbl_con_total[str_detect(entidad, regex("^JAL", ignore_case = T)), num_ent := 14]
tbl_con_total[str_detect(entidad, regex("^México", ignore_case = T)), num_ent := 15]
tbl_con_total[str_detect(entidad, regex("^MIC", ignore_case = T)), num_ent := 16]
tbl_con_total[str_detect(entidad, regex("^MOR", ignore_case = T)), num_ent := 17]
tbl_con_total[str_detect(entidad, regex("^NAY", ignore_case = T)), num_ent := 18]
tbl_con_total[str_detect(entidad, regex("^NU", ignore_case = T)), num_ent := 19]
tbl_con_total[str_detect(entidad, regex("^OA", ignore_case = T)), num_ent := 20]
tbl_con_total[str_detect(entidad, regex("^PU", ignore_case = T)), num_ent := 21]
tbl_con_total[str_detect(entidad, regex("^QUE", ignore_case = T)), num_ent := 22]
tbl_con_total[str_detect(entidad, regex("^QUI", ignore_case = T)), num_ent := 23]
tbl_con_total[str_detect(entidad, regex("^SAN", ignore_case = T)), num_ent := 24]
tbl_con_total[str_detect(entidad, regex("^SIN", ignore_case = T)), num_ent := 25]
tbl_con_total[str_detect(entidad, regex("^SON", ignore_case = T)), num_ent := 26]
tbl_con_total[str_detect(entidad, regex("^TAB", ignore_case = T)), num_ent := 27]
tbl_con_total[str_detect(entidad, regex("^TAM", ignore_case = T)), num_ent := 28]
tbl_con_total[str_detect(entidad, regex("^TLA", ignore_case = T)), num_ent := 29]
tbl_con_total[str_detect(entidad, regex("^VER", ignore_case = T)), num_ent := 30]
tbl_con_total[str_detect(entidad, regex("^YUC", ignore_case = T)), num_ent := 31]
tbl_con_total[str_detect(entidad, regex("^ZAC", ignore_case = T)), num_ent := 32]

```

```

tbl_con_total <- tbl_con_total[order(num_ent)]

tbl_con_total <- tbl_con_total[,.(orden=num_ent, entidad, T_num, H_num, M_num)]
tbl_con_total<-replace(tbl_con_total, is.na(tbl_con_total), 0)

#Cálculo del denominador(bases Conapo)

pob_sex01<-pob1[año==anio&sexo_cod==1,.(H_den=sum(poblacion)),
               by=.entidad][order(entidad)][!is.na(entidad)&entidad!='NE']

pob_sex02<-pob1[año==anio&sexo_cod==2,.(M_den=sum(poblacion)),
               by=.entidad][order(entidad)][!is.na(entidad)&entidad!='NE']

pob_total<-pob1[año==anio&sexo_cod!=9,.(T_den=sum(poblacion)),
               by=.entidad][order(entidad)][!is.na(entidad)&entidad!='NE']

pob_final <-
  Reduce(function(...) {merge(..., all = T, by = "entidad")}, list(pob_total,
    pob_sex01, pob_sex02))

pob_nac <- data.table(entidad = 'Nacional',
                     T_den = pob1[año == anio, sum(poblacion)],
                     H_den = pob1[año == anio & sexo_cod == 1, sum(poblacion)],
                     M_den = pob1[año == anio & sexo_cod == 2, sum(poblacion)])

pob_con_total <- rbindlist(list(pob_nac, pob_final))

#Integración del numerador y denominador

uni_tab <- merge(tbl_con_total,pob_con_total,by = "entidad")

#Cálculo de la tasa
class(uni_tab)
uni_tab[,c("tasa_T","tasa_H","tasa_M"):=.(T_num/T_den*100000,
    H_num/H_den*100000,M_num/M_den*100000)]

setcolorder(
  uni_tab,c("orden","entidad","tasa_T","tasa_H","tasa_M","T_num","H_num",
    "M_num","T_den","H_den","M_den"))

uni_tab[order(orden)]

#Se genera el tabulado#
res_entidad[[cont]] <- uni_tab[order(orden)]
names(res_entidad)[[cont]] <- as.character(anio)

rm(base)
}

lapply(seq_along(res_entidad), function(i){res_entidad[[i]][, anio :=

```

```

names(res_entidad)[i]}})
res_entidad_junto <- rbindlist(res_entidad)
class(res_entidad_junto)

#Se elige directorio para exportar los archivos
write.xlsx(list(res_entidad_junto),"D:/OneDrive - INEGI/CÓDIGOS_EN GIT/
compendio-codigos-ic_v2/Estimaciones/
HOMICySUICI/tbl_ent_Hom_2023.xlsx")

```

7.2.2. Tasa bruta anual de defunciones por homicidio por cada cien mil habitantes, por grupo de edad.

```

#Definir el directorio de trabajo
getwd()
setwd("C:/DEFUNCIONES/BD")

#Leemos la base de datos de CONAPO#
pob <- read_excel("CONAPO/0_Pob_Mitad_1950_2070.xlsx")

#Se realiza el filtro para trabajar solo con las entidades
pob1 <- subset(pob,(!pob$CVE_GEO==0))

#Convertimos a data frame la base de datos
setDT(pob1)

#Preparamos la base de datos de población

names(pob1) <- tolower(names(pob1))

pob1[sexo == "Hombres", sexo_cod := 1]
pob1[sexo != "Hombres", sexo_cod := 2]

pob1[, .N, by = .(sexo, sexo_cod)]

pob1[, sum(poblacion), by = .(sexo, año)]

#Grupos de edad
pob1[, grupo_edad := case_when(pob1$edad >= 0 & pob1$edad <= 4 ~ 'De 0 a 04 años',#1
pob1$edad >= 5 & pob1$edad <= 9 ~ 'De 05 a 09 años',#2
pob1$edad >= 10 & pob1$edad <= 14 ~ 'De 10 a 14 años',#3
pob1$edad >= 15 & pob1$edad <= 19 ~ 'De 15 a 19 años',#4
pob1$edad >= 20 & pob1$edad <= 24 ~ 'De 20 a 24 años',#5
pob1$edad >= 25 & pob1$edad <= 29 ~ 'De 25 a 29 años',#6
pob1$edad >= 30 & pob1$edad <= 34 ~ 'De 30 a 34 años',#7
pob1$edad >= 35 & pob1$edad <= 39 ~ 'De 35 a 39 años',#8
pob1$edad >= 40 & pob1$edad <= 44 ~ 'De 40 a 44 años',#9
pob1$edad >= 45 & pob1$edad <= 49 ~ 'De 45 a 49 años',#10
pob1$edad >= 50 & pob1$edad <= 54 ~ 'De 50 a 54 años',#11
pob1$edad >= 55 & pob1$edad <= 59 ~ 'De 55 a 59 años',#12
pob1$edad >= 60 & pob1$edad <= 64 ~ 'De 60 a 64 años',#13
pob1$edad >= 65 & pob1$edad <= 69 ~ 'De 65 a 69 años',#14
pob1$edad >= 70 & pob1$edad <= 74 ~ 'De 70 a 74 años',#15
pob1$edad >= 75 & pob1$edad <= 79 ~ 'De 75 a 79 años',#16
pob1$edad >= 80 & pob1$edad <= 84 ~ 'De 80 a 84 años',#17

```

```

pobl$edad >= 85 ~ 'Mas de 85 años')]]#18

#Cálculo de la Tasa bruta anual de defunciones por homicidio por cada cien mil
#habitantes, por grupo de edad

preg_int <- "causa_def"

#Se definen los años
anios <- 2023
res_edad <- list()
View()

cont <- 0

for(anio in anios){
  names3 <- paste0("X",850:999)
  names4 <- paste0("Y", formatC(0:99, width = 3, flag = "0"))
  val_int <- c(names3, names4)
}
cont <- cont + 1
nom_arch_dbf <- paste0("DEFUN", substr(as.character(anio), 3, 4), ".dbf")
arch_base <- file.path(getwd(), "Defunciones", anio, nom_arch_dbf)
base <- read.dbf(arch_base)
names(base) <- tolower(names(base))
setDT(base)

base[[preg_int]] <- as.character(base[[preg_int]])
base[, .N, by = causa_def%in%val_int]
base[, .N, by = .(base[[preg_int]]%in%val_int)]
base[, V1 := ifelse(base[[preg_int]]%in%val_int, 1, 0)]

#Cálculo del numerador#
base[, grupo_edad := case_when(base$edad<=4004~'De 0 a 04 años',#1
                                base$edad>=4005&base$edad<=4009~'De 05 a 09 años',#2
                                base$edad>=4010&base$edad<=4014~'De 10 a 14 años',#3
                                base$edad>=4015&base$edad<=4019~'De 15 a 19 años',#4
                                base$edad>=4020&base$edad<=4024~'De 20 a 24 años',#5
                                base$edad>=4025&base$edad<=4029~'De 25 a 29 años',#6
                                base$edad>=4030&base$edad<=4034~'De 30 a 34 años',#7
                                base$edad>=4035&base$edad<=4039~'De 35 a 39 años',#8
                                base$edad>=4040&base$edad<=4044~'De 40 a 44 años',#9
                                base$edad>=4045&base$edad<=4049~'De 45 a 49 años',#10
                                base$edad>=4050&base$edad<=4054~'De 50 a 54 años',#11
                                base$edad>=4055&base$edad<=4059~'De 55 a 59 años',#12
                                base$edad>=4060&base$edad<=4064~'De 60 a 64 años',#13
                                base$edad>=4065&base$edad<=4069~'De 65 a 69 años',#14
                                base$edad>=4070&base$edad<=4074~'De 70 a 74 años',#15
                                base$edad>=4075&base$edad<=4079~'De 75 a 79 años',#16
                                base$edad>=4080&base$edad<=4084~'De 80 a 84 años',#17
                                base$edad>=4085&!base$edad==4998~'Mas de 85 años',#18
                                base$edad==1098|base$edad==2098|base$edad==3098
                                |base$edad==4998~'NE')]]

```

```

tbl_sexo1<-base[V1==1&sexo==1,.(H_num=.N),
               by=(grupo_edad)][order(grupo_edad)][!is.na(grupo_edad)&
               grupo_edad!='NE']
tbl_sexo2<-base[V1==1&sexo==2,.(M_num=.N),
               by=(grupo_edad)][order(grupo_edad)][!is.na(grupo_edad)&
               grupo_edad!='NE']
tbl_total<-base[V1==1,.(T_num=.N),
               by=(grupo_edad)][order(grupo_edad)][!is.na(grupo_edad)&
               grupo_edad!='NE']

tbl_final <-Reduce(function(...) {merge(..., all=T,by="grupo_edad")},
                  list(tbl_total,tbl_sexo1,tbl_sexo2))

tbl_final<-replace(tbl_final, is.na(tbl_final), 0)

#Cálculo del denominador (bases Conapo)
pob_sexo1<-pob1[año==anio&sexo_cod==1,.(H_den=sum(poblacion)),
               by=(grupo_edad)][order(grupo_edad)][!is.na(grupo_edad)&
               grupo_edad!='NE']
pob_sexo2<-pob1[año==anio&sexo_cod==2,.(M_den=sum(poblacion)),
               by=(grupo_edad)][order(grupo_edad)][!is.na(grupo_edad)&
               grupo_edad!='NE']
pob_total<-pob1[año==anio&sexo_cod!=9,.(T_den=sum(poblacion)),
               by=(grupo_edad)][order(grupo_edad)][!is.na(grupo_edad)&
               grupo_edad!='NE']

pob_final <- Reduce(function(...) {merge(...,all=T,by="grupo_edad")},
                  list(pob_total,pob_sexo1,pob_sexo2))

#Integración del numerador y denominador

uni_tab <- merge(tbl_final,pob_final,by = "grupo_edad")

#Cálculo de la tasa
class(uni_tab)
uni_tab[,c("tasa_T","tasa_H","tasa_M"):=.(T_num/T_den*100000,
               H_num/H_den*100000,M_num/M_den*100000)]

setcolorder(uni_tab,c("grupo_edad","tasa_T","tasa_H","tasa_M","T_num","H_num",
               "M_num","T_den","H_den","M_den"))

uni_tab[, orden := c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18)]
uni_tab[order(orden)]

#Se genera el tabulado
res_edad[[cont]] <- uni_tab[order(orden)]
names(res_edad)[[cont]] <- as.character(anio)

rm(base)
}

lapply(seq_along(res_edad), function(i){res_edad[[i]][, anio := names(res_edad)[i]]})

```



```
res_edad_junto <- rbindlist(res_edad)
class(res_edad_junto)

#Se elige directorio para exportar los archivos
write.xlsx(list(res_edad_junto), "D:/OneDrive - INEGI/CÓDIGOS_EN GIT/
    compendio-codigos-ic_v2/Estimaciones/
    HOMICySUICI/tbl_edad_Hom_2023.xlsx")
```

8. Cálculo de los IC cuya fuente de información son los Censos Nacionales de Gobierno (CNG).

Debido al compromiso con la publicación en tiempo y forma de los IC en el CNI cuya fuente de información son los CNG no es posible trabajar con los datos abiertos (microdatos) provenientes de los mismos, dado que éstos se liberan tiempo después a las fechas de publicación de los IC, por tanto, es necesario acceder directamente a las bases de datos del área de censos para realizar el cálculo y la actualización de los IC. Por tal situación los cálculos y actualizaciones se realizan en el programa llamado: Sistema Gestor de Indicadores de Gobierno, Seguridad Pública y Justicia (SIGEI). SIGEI es un programa de uso interno de la Subdirección de Indicadores e Información Gubernamental (SIIG) que tiene tres principales funciones las cuales son: Actualizar los IC, Incorporar los IC y sirve como un repositorio de los IC independientemente del estatus que tengan en el CNI (eliminado, vigente o en proceso de incorporación). El personal del Departamento de Metodologías para Estadísticas de Gobierno y Justicia (DMEGJ) perteneciente a la SIIG es el usuario del SIGEI, sin embargo, la dirección de informática de la Dirección General de Estadísticas de Gobierno Seguridad Pública y Justicia (DGEESPJ) es quién le da mantenimiento y realiza ajustes en caso de que se requiera.

A continuación, se describe cómo es el proceso de actualización de un IC cuya fuente de información son los CNG.

8.1 Proceso de actualización de los IC cuya fuente de información son los CNG.

El proceso de actualización de los IC se lleva a cabo en colaboración con el área de censos y en particular con la Subdirección de Administración de Información de Censos Nacionales de Gobierno (SAICNG). En seguida, se muestra el esquema sobre el proceso que se lleva a cabo, con posterioridad, se hace una descripción de cada actividad del proceso.



8.1.1. Construcción de lista de necesidades.

Para poder llevar a cabo la construcción de la lista de necesidades, como paso previo la Dirección de Indicadores y Análisis de Información Gubernamental (DIAIG) solicita las versiones de los cuestionarios etiquetados de los CNG a la Dirección de Censos Nacionales de Gobierno (DCNG) estas versiones pueden tener estatus definitivo o preliminar, sin embargo, a lo largo del proceso se va dando seguimiento al estatus. Por otro lado, el equipo de la DIAIG realiza el documento de identificación de reactivos con base en los cuestionarios etiquetados de los CNG (lista de necesidades); clasificados según el instrumento de captación. Después, la DCNG revisa y comenta el documento de identificación de reactivos. Finalmente, personal de la DIAIG, DCNG y la Dirección de Políticas de Información Gubernamental (DPIG) revisan el documento de identificación de reactivos y acuerdan la versión definitiva para el procesamiento de insumos de los IC que estarán disponibles en el Banco de Información Sociodemográfica y Económica (BISE).

8.1.2. Extracción y procesamiento de variables.

La DIAIG solicita las claves de acceso a las bases de datos de los CNG y la DCNG envía dichas claves, en este proceso, también se indica el estatus de la información, éste puede ser definitivo, preliminar o reservado. En algunas ocasiones, el acceso a las bases de datos se envía en conjunto con los cuestionarios etiquetados. En este proceso la DIAIG y la DCNG realizan la extracción e integración de los insumos de los IC cuya fuente de información son los CNG. La extracción de insumos se realiza con base en secuencias SQL y la DIAIG se apoya de SIGEI para realizar esta actividad.

8.1.3. Validación de información con el área generadora.

La DIAIG solicita el archivo a la DCNG (considerando claves BIINEGI, denominación y resultado) con los resultados de cálculo de los insumos de los IC. Por su parte, la DCNG envía el archivo en formato Excel. Paso seguido, la DIAIG compara los resultados concentrados por ambas áreas. La DIAIG envía el resultado de la validación en el documento en donde se identifican posibles diferencias en los insumos de los IC. Finalmente, la DIAIG y la DCNG realizan la revisión y acuerdan sobre los resultados de los insumos principalmente en los casos en donde haya diferencias.

8.1.4. Procesamiento y cálculo de los IC

La DIAIG realiza el cálculo de los IC, este proceso se desarrolla en SIGEI, y consiste en que una vez que los insumos ya fueron procesados, se procede con el cálculo de los IC. Es importante mencionar que en esta etapa SIGEI considera las reglas de procesamiento de los IC, que refieren a situaciones en donde se reportan valores de excepción (valores no numéricos).

8.1.5. Generación de tabulados.

Una vez que se llevó a cabo el cálculo de los IC, lo siguiente es elaborar los tabulados que contenga la información de los insumos e IC, los cuales se trabajan en formato Excel.

8.1.6. Validación de resultados.

Se realiza la validación de los resultados, en este proceso se comparan los resultados obtenidos por integrantes del DMEGJ.

8.1.7. Análisis de los resultados obtenidos.

Se realiza el análisis de la calidad de los resultados de los insumos e IC, en caso de haber algún comportamiento atípico en los datos, por ejemplo; variaciones importantes entre los resultados de un año a otro, gran número de valores de excepción en las series de los datos, poca sensibilidad del IC, etc., éstos se reportan en un documento detallando cada situación en particular.

8.1.8. Reconsulta de datos atípicos.

En caso de existir comportamientos atípicos la DIAIG informa a la DCNG sobre este comportamiento y solicita hacer consultas a los informantes con la finalidad de saber qué puede estar pasando. Por su parte, la DCNG realiza las consultas e informa a la DIAIG el estatus de éstas.

8.1.9. Construcción de formatos cortos.

Una vez que se da respuestas a las consultas, se considera que se tienen las versiones finales de los insumos e IC, por lo que, la DIAIG genera los formatos cortos, que son archivos en Excel que continen la información del metadato del IC, los valores de los insumos e IC.

8.1.10. Envío de formatos al área encargada del CNI.

La DIAIG envía los formatos cortos y la documentación requerida a la Dirección de Monitoreo y Evaluación de los Programas del SNIEG (DMESNIEG) para la publicación de los IC en el CNI.

8.1.11. Seguimiento al procedimiento de carga de datos.

En este proceso hay actividades importantes, por un lado, la DCNG realiza la integración de las versiones finales de insumos y realiza el envío los resultados de éstos al BISE con copia a la DIAIG, después la Dirección de Integración de la Información en Bases de Datos (DIIBD) se encarga de realizar la integración

de los resultados de los insumos en BISE. Finalmente, la DCNG realiza la validación de la información de los insumos con BISE y atención a posibles inconsistencias con DIIBD. Las actividades 10 y 11 son paralelas, es decir, se llevan al mismo tiempo, sin embargo, las llevan a cabo diferentes áreas ya que en el proceso de actualización de los insumos de los IC se toman de BISE a cargo de la DCNG y los resultados de los IC se toman de los formatos cortos responsabilidad de la DIAIG. La DIAIG al ser la responsable del proceso de actualización de los IC está informada o involucrada de las actividades 10 y 11.

8.1.12. Validación en etapa de desarrollo y preproducción.

La DMESNIEG realiza la revisión de la información para la actualización de los IC y coordinación con DIIBD para su integración a BISE. Por su parte, la DIAIG realiza la revisión y validación del proceso de publicación de la actualización de los IC en las páginas de desarrollo y preproducción. Si hay observaciones la DIAIG realiza un documento con las observaciones encontradas y se envían a la DMESNIEG para su atención. En caso contrario, se informa a la DMESNIEG para que se continúe con la publicación en la página pública del CNI.

8.1.13. Publicación de resultados en el CNI.

Se realiza la publicación de la actualización de los IC en el CNI.

8.1.14. Validación en la etapa de producción

Una vez publicados los IC en la página pública del CNI, la DIAIG lleva a cabo la última revisión, en caso de haber observaciones, se envían a la DMESNIEG para su atención. En caso contrario, se informa a la DMESNIEG que la actualización ha sido concluida.

9 ANEXO: Funciones para el cálculo de los IC cuya fuente de información son las ENG.

A continuación, se presentan las funciones que serán utilizadas para el cálculo de los IC cuya fuente de información son las ENG.

a. Función: **function1_nac**

```
# Se crea la función llamada function1_nac, que contiene el cálculo del  
# indicador, sus insumos y sus medidas de precisión a nivel nacional.  
#Esta función se utiliza para el cálculo de las tasas.
```

```
# FUNCIÓN CÁLCULO NACIONAL
```

```
# Se crea la función
```

```
function1_nac<-function(x,numerador,denominador,DIS){
```

```
# Cálculo del numerador
```

```
TVSN_N <- svytotal(~numerador,DIS)
```

```
# Estimaciones, Error estándar, Coeficiente de variación e
```

```
# Intervalos de confianza
```

```
EST_TVSN_N <-TVSN_N[[1]]
```

```
SE_TVSN_N <-SE(TVSN_N)
```

```
CV_TVSN_N <-cv(TVSN_N)*100
```

```
LI_TVSN_N <-confint(TVSN_N,level = 0.90)[1,1]
```

```
LS_TVSN_N <-confint(TVSN_N,level = 0.90)[1,2]
```

```
# Cálculo del denominador
```

```
TVSN_D <- svytotal(~denominador,DIS)
```

```
# Estimaciones, Error estándar, Coeficiente de variación e
```

```
# Intervalos de confianza
```

```
EST_TVSN_D <-TVSN_D[[1]]
```

```
SE_TVSN_D <-SE(TVSN_D)
```

```
CV_TVSN_D <-cv(TVSN_D)*100
```

```
LI_TVSN_D <-confint(TVSN_D,level = 0.90)[1,1]
```

```
LS_TVSN_D <-confint(TVSN_D,level = 0.90)[1,2]
```

```
# Cálculo del indicador
```

```
TVSN <- svyratio(~numerador, denominator=~denominador, DIS)
```

```
# Estimaciones, Error estándar, Coeficiente de variación e
```

```
# Intervalos de confianza
```

```
EST_TVSN <-TVSN[[1]]*100000
```

```
SE_TVSN <-SE(TVSN)*100000
```

```
CV_TVSN <-cv(TVSN)*100
```

```
LI_TVSN <-confint(TVSN,level = 0.90)[1,1]*100000
```

```
LS_TVSN <-confint(TVSN,level = 0.90)[1,2]*100000
```

```
# Creación del tabulado
```

```
estimac <- data.frame(EST_TVSN[1,1])
```

```
estimac_n <- data.frame(EST_TVSN_N )
```

```
estimac_d <- data.frame(EST_TVSN_D)
```

```

serror <- data.frame(SE_TVSN)
serror_n <-data.frame(SE_TVSN_N[1,1])
serror_d <- data.frame(SE_TVSN_D[1,1])
coefvar <- data.frame(CV_TVSN[1,1])
coefvar_n <- data.frame(CV_TVSN_N[1,1])
coefvar_d <- data.frame(CV_TVSN_D[1,1])
lim_inf <- data.frame(LI_TVSN)
lim_inf_n <- data.frame(LI_TVSN_N)
lim_inf_d <- data.frame(LI_TVSN_D)
lim_sup <- data.frame(LS_TVSN)
lim_sup_n <- data.frame(LS_TVSN_N)
lim_sup_d <-data.frame(LS_TVSN_D)
nacional <- c('Estados Unidos Mexicanos')
tab_nacional <- cbind(nacional,estimac,serror,coefvar,lim_inf,lim_sup,
                      estimac_n,serror_n,coefvar_n,lim_inf_n,lim_sup_n,
                      estimac_d,serror_d,coefvar_d,lim_inf_d,lim_sup_d)

# Resultado que la función retorna.
return(tab_nacional)
}

```

b. Función: **function1__ent**

```

# Se crea la función llamada function1_ent, que contiene el cálculo del
# indicador, sus insumos y sus medidas de precisión por entidad federativa.
# Esta función se utiliza para el cálculo de las tasas.

# FUNCIÓN CÁLCULO ENTIDAD FEDERATIVA
# Se crea la función
function1_ent<-function(x,numerador,denominador,DIS){

  # Cálculo del numerador
  TVSE_N <- svyby(~numerador,by=~CVE_ENT, DIS, svytotal)

  # Estimaciones, Error estándar, Coeficiente de variación e
  # Intervalos de confianza
  EST_TVSE_N <-TVSE_N[[2]]
  SE_TVSE_N<-SE(TVSE_N)
  CV_TVSE_N<-cv(TVSE_N)*100
  LI_TVSE_N<-confint(TVSE_N,level=0.90)[,1]
  LS_TVSE_N<-confint(TVSE_N,level=0.90)[,2]

  # Cálculo del denominador
  TVSE_D <- svyby(~denominador,by=~CVE_ENT, DIS, svytotal)

  # Estimaciones, Error estándar, Coeficiente de variación e
  # Intervalos de confianza
  EST_TVSE_D <-TVSE_D[[2]]
  SE_TVSE_D<-SE(TVSE_D)
  CV_TVSE_D<-cv(TVSE_D)*100
  LI_TVSE_D<-confint(TVSE_D,level=0.90)[,1]
  LS_TVSE_D<-confint(TVSE_D,level=0.90)[,2]

  # Cálculo del indicador

```

```

TVSE <- svyby(~numerador, denominator=~denominador, by=~CVE_ENT, DIS, svyratio)

# Estimaciones, Error estándar, Coeficiente de variación e
# Intervalos de confianza
EST_TVSE <-TVSE[[2]]*100000
SE_TVSE<-SE(TVSE)*100000
CV_TVSE<-cv(TVSE)*100
LI_TVSE<-confint(TVSE,level=0.90)[,1]*100000
LS_TVSE<-confint(TVSE,level=0.90)[,2]*100000

# Creación del tabulado
estimac <- data.frame(EST_TVSE)
estimac_n <- data.frame(EST_TVSE_N)
estimac_d <- data.frame(EST_TVSE_D)
error<- data.frame(SE_TVSE)
error_n <- data.frame(SE_TVSE_N)
error_d<- data.frame(SE_TVSE_D)
coefvar <- data.frame(CV_TVSE)
coefvar_n <- data.frame(CV_TVSE_N)
coefvar_d <- data.frame(CV_TVSE_D)
lim_inf <- data.frame(LI_TVSE)
lim_inf_n <- data.frame(LI_TVSE_N)
lim_inf_d <- data.frame(LI_TVSE_D)
lim_sup <- data.frame(LS_TVSE)
lim_sup_n <- data.frame(LS_TVSE_N)
lim_sup_d <- data.frame(LS_TVSE_D)

entidad <- c('Aguascalientes', 'Baja California',
             'Baja California Sur', 'Campeche', 'Coahuila de Zaragoza', 'Colima',
             'Chiapas', 'Chihuahua', 'Ciudad de México', 'Durango', 'Guanajuato',
             'Guerrero', 'Hidalgo', 'Jalisco', 'Estado de México',
             'Michoacán de Ocampo', 'Morelos', 'Nayarit', 'Nuevo León', 'Oaxaca',
             'Puebla', 'Querétaro', 'Quintana Roo', 'San Luis Potosí', 'Sinaloa',
             'Sonora', 'Tabasco', 'Tamaulipas', 'Tlaxcala',
             'Veracruz de Ignacio de la Llave', 'Yucatán', 'Zacatecas')

tab_entidad <- cbind(entidad,estimac,error,coefvar,lim_inf,lim_sup,
                    estimac_n,error_n,coefvar_n,lim_inf_n,lim_sup_n,
                    estimac_d,error_d,coefvar_d,lim_inf_d,lim_sup_d)
# Resultado que la función retorna.
return(tab_entidad)
}

```

c. Función: **function2__nac**

```

# Se crea la función llamada function2_nac, que contiene el cálculo del
# indicador, sus insumos y sus medidas de precisión a nivel nacional.
# Esta función se utiliza para el cálculo de los porcentajes

# FUNCIÓN CÁLCULO NACIONAL
# Se crea la función
function2_nac<-function(x,numerador,denominador,DIS){

```

```

# Cálculo del numerador
TVSN_N <- svytotal(~numerador,DIS)

# Estimaciones, Error estándar, Coeficiente de variación e
# Intervalos de confianza
EST_TVSN_N <-TVSN_N[[1]]
SE_TVSN_N <-SE(TVSN_N)
CV_TVSN_N <-cv(TVSN_N)*100
LI_TVSN_N <-confint(TVSN_N,level = 0.90)[1,1]
LS_TVSN_N <-confint(TVSN_N,level = 0.90)[1,2]

# Cálculo del denominador
TVSN_D <- svytotal(~denominador,DIS)

# Estimaciones, Error estándar, Coeficiente de variación e
# Intervalos de confianza
EST_TVSN_D <-TVSN_D[[1]]
SE_TVSN_D <-SE(TVSN_D)
CV_TVSN_D <-cv(TVSN_D)*100
LI_TVSN_D <-confint(TVSN_D,level = 0.90)[1,1]
LS_TVSN_D <-confint(TVSN_D,level = 0.90)[1,2]

# Cálculo del indicador
TVSN <- svyratio(~numerador, denominator=~denominador, DIS)

# Estimaciones, Error estándar, Coeficiente de variación e
# Intervalos de confianza
EST_TVSN <-TVSN[[1]]*100
SE_TVSN <-SE(TVSN)*100
CV_TVSN <-cv(TVSN)*100
LI_TVSN <-confint(TVSN,level = 0.90)[1,1]*100
LS_TVSN <-confint(TVSN,level = 0.90)[1,2]*100

# Creación del tabulado
estimac <- data.frame(EST_TVSN[1,1])
estimac_n <- data.frame(EST_TVSN_N )
estimac_d <- data.frame(EST_TVSN_D)
serror <- data.frame(SE_TVSN)
serror_n <-data.frame(SE_TVSN_N[1,1])
serror_d <- data.frame(SE_TVSN_D[1,1])
coefvar <- data.frame(CV_TVSN[1,1])
coefvar_n <- data.frame(CV_TVSN_N[1,1])
coefvar_d <- data.frame(CV_TVSN_D[1,1])
lim_inf <- data.frame(LI_TVSN)
lim_inf_n <- data.frame(LI_TVSN_N)
lim_inf_d <- data.frame(LI_TVSN_D)
lim_sup <- data.frame(LS_TVSN)
lim_sup_n <- data.frame(LS_TVSN_N)
lim_sup_d <-data.frame(LS_TVSN_D)
nacional <- c('Estados Unidos Mexicanos')
tab_nacional <- cbind(nacional,estimac,serror,coefvar,lim_inf,lim_sup,
                      estimac_n,serror_n,coefvar_n,lim_inf_n,lim_sup_n,
                      estimac_d,serror_d,coefvar_d,lim_inf_d,lim_sup_d)

```



```

# Resultado que la función retorna.
return(tab_nacional)
}

```

d. Función: **function2__ent**

```

# Se crea la función llamada function2_ent, que contiene el cálculo del
# indicador, sus insumos y sus medidas de precisión por entidad federativa.
# Esta función se utiliza para el cálculo de los porcentajes

# FUNCIÓN CÁLCULO ENTIDAD FEDERATIVA
# Se crea la función

function2_ent<-function(x,numerador,denominador,DIS){

  # Cálculo del numerador
  TVSE_N <- svyby(~numerador,by=~CVE_ENT, DIS, svytotal)

  # Estimaciones, Error estándar, Coeficiente de variación e
  # Intervalos de confianza
  EST_TVSE_N <-TVSE_N[[2]]
  SE_TVSE_N<-SE(TVSE_N)
  CV_TVSE_N<-cv(TVSE_N)*100
  LI_TVSE_N<-confint(TVSE_N,level=0.90)[,1]
  LS_TVSE_N<-confint(TVSE_N,level=0.90)[,2]

  # Cálculo del denominador
  TVSE_D <- svyby(~denominador,by=~CVE_ENT, DIS, svytotal)

  # Estimaciones, Error estándar, Coeficiente de variación e
  # Intervalos de confianza
  EST_TVSE_D <-TVSE_D[[2]]
  SE_TVSE_D<-SE(TVSE_D)
  CV_TVSE_D<-cv(TVSE_D)*100
  LI_TVSE_D<-confint(TVSE_D,level=0.90)[,1]
  LS_TVSE_D<-confint(TVSE_D,level=0.90)[,2]

  # Cálculo del indicador
  TVSE <- svyby(~numerador, denominator=~denominador, by=~CVE_ENT, DIS, svyratio)

  # Estimaciones, Error estándar, Coeficiente de variación e
  # Intervalos de confianza
  EST_TVSE <-TVSE[[2]]*100
  SE_TVSE<-SE(TVSE)*100
  CV_TVSE<-cv(TVSE)*100
  LI_TVSE<-confint(TVSE,level=0.90)[,1]*100
  LS_TVSE<-confint(TVSE,level=0.90)[,2]*100

  # Creación del tabulado
  estimac <- data.frame(EST_TVSE)
  estimac_n <- data.frame(EST_TVSE_N)
  estimac_d <- data.frame(EST_TVSE_D)
  serror<- data.frame(SE_TVSE)

```

```

serror_n <- data.frame(SE_TVSE_N)
serror_d<- data.frame(SE_TVSE_D)
coefvar <- data.frame(CV_TVSE)
coefvar_n <- data.frame(CV_TVSE_N)
coefvar_d <- data.frame(CV_TVSE_D)
lim_inf <- data.frame(LI_TVSE)
lim_inf_n <- data.frame(LI_TVSE_N)
lim_inf_d <- data.frame(LI_TVSE_D)
lim_sup <- data.frame(LS_TVSE)
lim_sup_n <- data.frame(LS_TVSE_N)
lim_sup_d <- data.frame(LS_TVSE_D)

entidad <- c('Aguascalientes', 'Baja California',
             'Baja California Sur', 'Campeche', 'Coahuila de Zaragoza', 'Colima',
             'Chiapas', 'Chihuahua', 'Ciudad de México', 'Durango', 'Guanajuato',
             'Guerrero', 'Hidalgo', 'Jalisco', 'Estado de México',
             'Michoacán de Ocampo', 'Morelos', 'Nayarit', 'Nuevo León', 'Oaxaca',
             'Puebla', 'Querétaro', 'Quintana Roo', 'San Luis Potosí', 'Sinaloa',
             'Sonora', 'Tabasco', 'Tamaulipas', 'Tlaxcala',
             'Veracruz de Ignacio de la Llave', 'Yucatán', 'Zacatecas')

tab_entidad <- cbind(entidad,estimac,serror,coefvar,lim_inf,lim_sup,
                    estimac_n,serror_n,coefvar_n,lim_inf_n,lim_sup_n,
                    estimac_d,serror_d,coefvar_d,lim_inf_d,lim_sup_d)
# Resultado que la función retorna.
return(tab_entidad)
}

```

e. Función: `function7_edad`

```

# Se crea la función llamada function7_edad, que contiene el cálculo del
# indicador, sus insumos y sus medidas de precisión por grupo de edad.
# Esta función se utiliza para el cálculo de los porcentajes.

# FUNCIÓN CÁLCULO POR GRUPO DE EDAD
# Se crea la función

function7_edad<-function(x,numerador,denominador,DIS){

  # Cálculo del numerador
  TVSE_N <- svyby(~numerador,by=~grp_edad, DIS, svytotal, na.rm = TRUE)

  # Estimaciones, Error estándar, Coeficiente de variación e
  # Intervalos de confianza
  EST_TVSE_N <-TVSE_N[[2]]
  SE_TVSE_N<-SE(TVSE_N)
  CV_TVSE_N<-cv(TVSE_N)*100
  LI_TVSE_N<-confint(TVSE_N,level=0.90)[,1]
  LS_TVSE_N<-confint(TVSE_N,level=0.90)[,2]

  # Cálculo del denominador
  TVSE_D <- svyby(~denominador,by=~grp_edad, DIS, svytotal, na.rm = TRUE)

```

```

# Estimaciones, Error estándar, Coeficiente de variación e
# Intervalos de confianza
EST_TVSE_D <-TVSE_D[[2]]
SE_TVSE_D<-SE(TVSE_D)
CV_TVSE_D<-cv(TVSE_D)*100
LI_TVSE_D<-confint(TVSE_D,level=0.90)[,1]
LS_TVSE_D<-confint(TVSE_D,level=0.90)[,2]

# Cálculo del indicador
TVSE <- svyby(~numerador,denominator=~denominator,by=~grp_edad,DIS,
              svyratio, na.rm = TRUE)

# Estimaciones, Error estándar, Coeficiente de variación e
# Intervalos de confianza
EST_TVSE <-TVSE[[2]]*100
SE_TVSE<-SE(TVSE)*100
CV_TVSE<-cv(TVSE)*100
LI_TVSE<-confint(TVSE,level=0.90)[,1]*100
LS_TVSE<-confint(TVSE,level=0.90)[,2]*100

# Creación del tabulado
estimac <- data.frame(EST_TVSE)
estimac_n <- data.frame(EST_TVSE_N)
estimac_d <- data.frame(EST_TVSE_D)
serror<- data.frame(SE_TVSE)
serror_n <- data.frame(SE_TVSE_N)
serror_d<- data.frame(SE_TVSE_D)
coefvar <- data.frame(CV_TVSE)
coefvar_n <- data.frame(CV_TVSE_N)
coefvar_d <- data.frame(CV_TVSE_D)
lim_inf <- data.frame(LI_TVSE)
lim_inf_n <- data.frame(LI_TVSE_N)
lim_inf_d <- data.frame(LI_TVSE_D)
lim_sup <- data.frame(LS_TVSE)
lim_sup_n <- data.frame(LS_TVSE_N)
lim_sup_d <- data.frame(LS_TVSE_D)

edad <- c('15-17 años', '18-19 años', '20-24 años', '25-29 años',
          '30-34 años', '35-39 años', '40-44 años','45-49 años',
          '50-54 años', '55-59 años', '60 y mas años', 'No especificado')

tab_edad <- cbind(edad,estimac,serror,coefvar,lim_inf,lim_sup,
                  estimac_n,serror_n,coefvar_n,lim_inf_n,lim_sup_n,
                  estimac_d,serror_d,coefvar_d,lim_inf_d,lim_sup_d)

# Resultado que la función retorna.
return(tab_edad)
}

```

f. Función: `applyConditionalFormatting`

```

# Se crea la función llamada applyConditionalFormatting,
#que contiene aplica el formato condicional a los insumos con base en el

```

```

#coeficiente de variación.

# FUNCIÓN CÁLCULO
# Se crea la función
applyConditionalFormatting <- function(wb, sheet, cols, rows, rules) {
  # Define los estilos
  styles <- list(
    style_30 = createStyle(fontColour = "#000000", bgFill = "#FF5400"),
    style_30_15 = createStyle(fontColour = "#000000", bgFill = "#FFEA00")
  )

  # Aplicar reglas de formato condicional para cada columna
  for (rule in rules) {
    for (col in cols) {
      conditionalFormatting(
        wb, sheet,
        cols = col,
        rows = rows,
        rule = rule$condition,
        style = styles[[rule$style]]
      )
    }
  }
}

```

Se crea una lista para almacenar los tabulados

```

#Define reglas y columnas para cada conjunto de columnas
#Formato condicional
cols_rules <- list(
  list(
    cols = 2, # Columna B en Excel
    rules = list(
      list(condition = "D2 >= 30", style = "style_30"),
      list(condition = "AND(D2>=15, D2<30)", style = "style_30_15")
    )
  ),
  list(
    cols = 7, # Columna G en Excel
    rules = list(
      list(condition = "I2 >= 30", style = "style_30"),
      list(condition = "AND(I2>=15, I2<30)", style = "style_30_15")
    )
  ),
  list(
    cols = 12, # Columna L en Excel
    rules = list(
      list(condition = "N2 >= 30", style = "style_30"),
      list(condition = "AND(N2>=15, N2<30)", style = "style_30_15")
    )
  ),
  list(
    cols = 17, # Columna Q en Excel
    rules = list(

```

```

    list(condition = "S2 >= 30", style = "style_30"),
    list(condition = "AND(S2>=15, S2<30)", style = "style_30_15")
  )
),
list(
  cols = 22, # Columna V en Excel
  rules = list(
    list(condition = "X2 >= 30", style = "style_30"),
    list(condition = "AND(X2>=15, X2<30)", style = "style_30_15")
  )
)
)

```

g. Función: **function1_cv**

```

# Se crea la función llamada function1_cv, para aplicar
#formato condicional a múltiples hojas

# FUNCIÓN CÁLCULO
# Se crea la función
function1_cv <- function(wb, sheets, cols_rules, rows) {
  # Aplica el formato condicional a cada hoja
  for (sheet in sheets) {
    for (rule_set in cols_rules) {
      cols <- rule_set$cols
      rules <- rule_set$rules

      # Aplicar formato condicional usando las reglas y columnas definidas
      applyConditionalFormatting(wb, sheet, cols, rows, rules)
    }
  }

  return(wb)
}

```