**COMENATARIOS DEL TP2**

1\_

Según la Encuesta Permanente de Hogares (EPH) del INDEC, se considera desocupada a toda persona que no tiene trabajo, está disponible para trabajar y, además, buscó activamente empleo durante las últimas 4 semanas previas a la encuesta.

Es decir, para que una persona sea clasificada como desocupada, deben cumplirse tres condiciones simultáneas:

1. No tener trabajo.
2. Estar disponible para trabajar.
3. Haber buscado activamente empleo en las últimas 4 semanas.

También se incluyen en este grupo a quienes:

* Están esperando volver a un trabajo del que fueron suspendidos.
* Ya consiguieron un empleo, pero aún no comenzaron a trabajar.

2\_

Lo que hicimos fue unificar los tipos de datos de ambas bases en formato float, es decir, valores numéricos, desde el principio para evitar inconvenientes posteriores. En el caso de la base de 2004, los valores estaban en formato string ("str"), por lo que fue necesario convertirlos y asignarles el valor numérico correspondiente según lo informado en el registro de la EPH de ese año (por ejemplo:

df\_2004["pp07a"] = df\_2004["pp07a"].replace({

"0": 0,

"Menos de un mes": 1,

"1 a 3 meses": 2,

"Más de 3 a 6 meses": 3,

"Más de 6 a 12 meses": 4,

"Más de 1 a 5 años": 5,

"Más de 5 años": 6,

"Ns./Nr.": 99})

Por otra parte, en la base de 2024 dejamos el valor 0 para los casos faltantes utilizando “valores\_faltantes = [0, np.nan]”. De esta manera, nos resulta más fácil identificar los datos incompletos mediante un filtrado. Tras realizar estas modificaciones, no encontramos errores en las variables seleccionadas.

Para filtrar por región tuvimos que primero poner los títulos de ambsa bases en mayúsculas ya que en la de 2004 estaban en minúscula y no podíamos unir el mismo dato creando una nueva variable por cada año:

df\_2004\_noa = df\_2004[df\_2004["REGION"] == 40]

df\_2024\_noa = df\_2024[df\_2024["REGION"] == 40]

Luego unimos las bases con la nueva variable creada “de\_unido” en base a df\_2004\_noa y df\_2024\_noa. Definimos las variables de interés y buscamos los datos faltantes con la función “.isin” para detectar los Nan y la “.sum” para luego sumarlos como valores:

faltantes\_2004 = df\_2004[variables\_interes].isin(valores\_faltantes).sum() + df\_2004[variables\_interes].isna().sum()

Seleccionamos las siguientes variables:

CH04: Sexo

CH06: Edad

ESTADO: Condición de actividad

CAT\_OCUP: Categoría ocupacional

AGLOMERADO: Código de Aglomerado

NIVEL\_ED: Nivel educativo

P47T: Ingreso total individual

P21: Monto de ingreso de la ocupación principal

CH03: Relación de parentesco

CH07: Estado civil

CH08: ¿Tiene algún tipo de cobertura médica por la que paga o le descuentan?

CH10: ¿Asiste o asistió a algún establecimiento educativo (colegio, escuela, universidad)?

CH11: Ese establecimiento es público, privado o Ns. /Nr.?

CH12: Nivel cursado

CH13: ¿Lo finalizó ese nivel?

De las 15 variables las que tienen mayores datos faltantes ordenados de mayor a menor son:

Para 2004:

* P21: Monto de ingreso de la ocupación principal: 6578 datos faltantes.
* CH11: Ese establecimiento es público, privado o Ns. /Nr. ?: 6135 datos faltantes.
* CAT\_OCUP: Categoría ocupacional: 5624 datos faltantes.
* P47T: Ingreso total individual: 5331 datos faltantes.
* CH12: Nivel cursado: 930 datos faltantes
* CH10: ¿Asiste o asistió a algún establecimiento educativo (colegio, escuela, universidad) ?: 328 datos faltantes

Para 2024:

* CH11: Ese establecimiento es público, privado o Ns. /Nr. ?: 6580 datos faltantes.
* P21: Monto de ingreso de la ocupación principal: 5527 datos faltantes.
* CAT\_OCUP: Categoría ocupacional: 5249 datos faltantes.
* P47T: Ingreso total individual: 3371 datos faltantes.
* CH12: Nivel cursado: 353 datos faltantes.
* CH11: Ese establecimiento es público, privado o Ns. /Nr. ?: 353 datos faltantes
* CH10: ¿Asiste o asistió a algún establecimiento educativo (colegio, escuela, universidad) ?: 159 datos faltantes.