CODER HOUSE

Análisis socioeducativo de los habitantes de la Ciudad de Buenos Aires

Profesor: Damian Dapueto

Tutor: Héctor Alonso

Grupo de Trabajo: Lucia Buzzeo, Lucia Hukovsky, Jose Saint German, Juan Martín Carini

Introdución

Principales hitos:

- En la Ciudad Autónoma de Buenos Aires, se ha encontrado una gran limitación relacionada con el acceso equitativo a la educación.
- Para trabajar esta problemática, se ha recurrido a la Encuesta Anual de Hogares del Gobierno de la Ciudad de Buenos Aires para el año 2019.
- Esta encuesta contiene información demográfica, social, económica, educativa y de salud de 14319 habitantes de la Ciudad.

Objetivos

- Descubrir las principales variables intervinientes en el nivel máximo educativo alcanzado por la población de la Ciudad Autónoma de Buenos Aires (CABA).
- Predecir nuestra variable target "Nivel Máximo Educativo" mediante dos modelos de clasificación:
  - **Árbol de decisión:** que construye un árbol durante el entrenamiento aplicado a la hora de realizar la predicción.
  - **Bosque Aleatorio:** que es un conjunto (ensemble) de árboles de decisión combinados con bagging.

Estructura de los trabajos

Este trabajo se ha dividido en 3 partes:

1. **Introducción a las variables del problema:** Análisis exploratorio del dataset.
2. **Modelos analíticos:** Entrenamiento de modelos analíticos de clasificación.
3. **Conclusión:** Conclusiones finales sobre los hallazgos, discusión de posibles limitaciones y futuras líneas de análisis.

Análisis exploratorio de los datos (EDA)

Introducción a las variables:

- 31 variables
  - 10 numéricas
  - 21 categóricas.

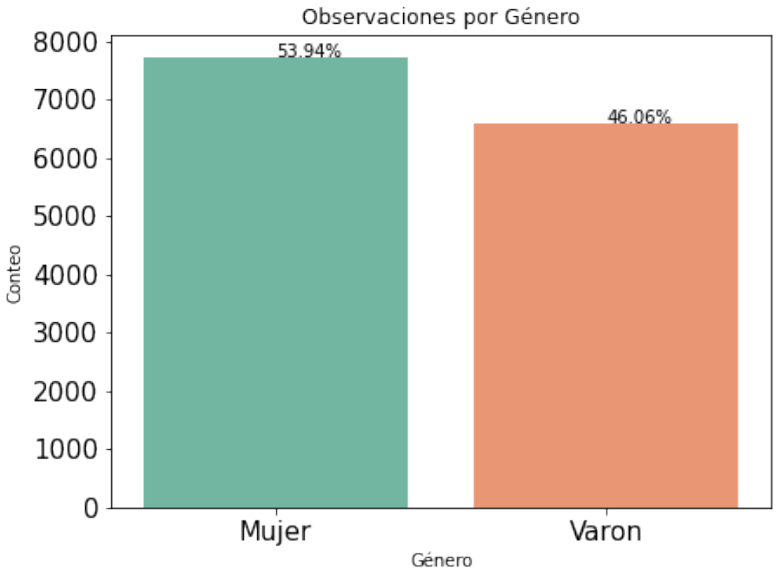
Estas variables describen las siguientes características de los encuestados:

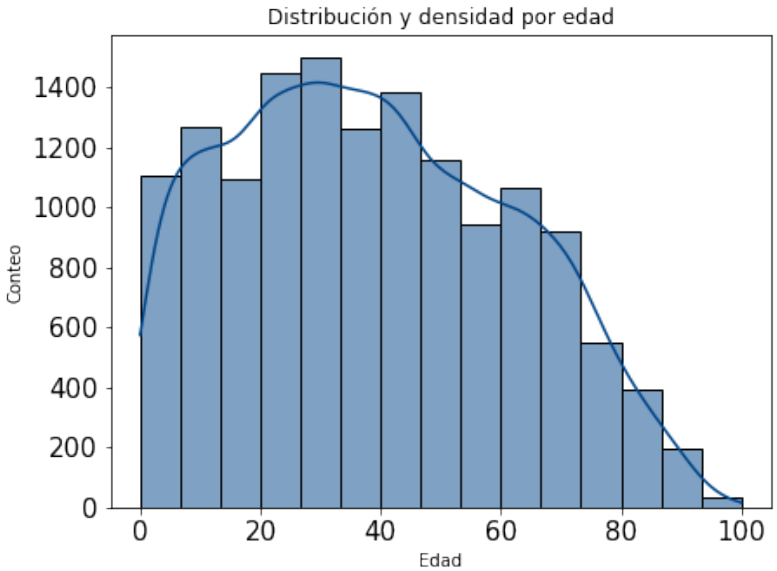
- El nivel de ingresos
- El Sector educativo
- Los Factores geográficos
- Características de la Salud
- Descripción del grupo familiar

Identificación del Target: Seleccionamos nuestra variable Target u objetivo para trabajar. Para reducir su dimensionalidad, la transformamos dejándola con cuatro valores: **Inicial**, **Prim. Completo**, **Sec. Completo** y **Superior**.

Análisis univariado

Análisis del género y la edad de los encuestados:



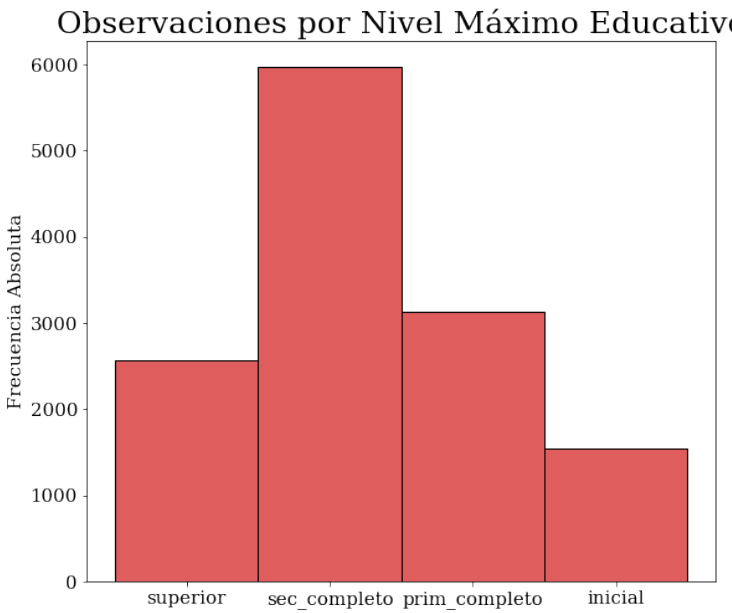


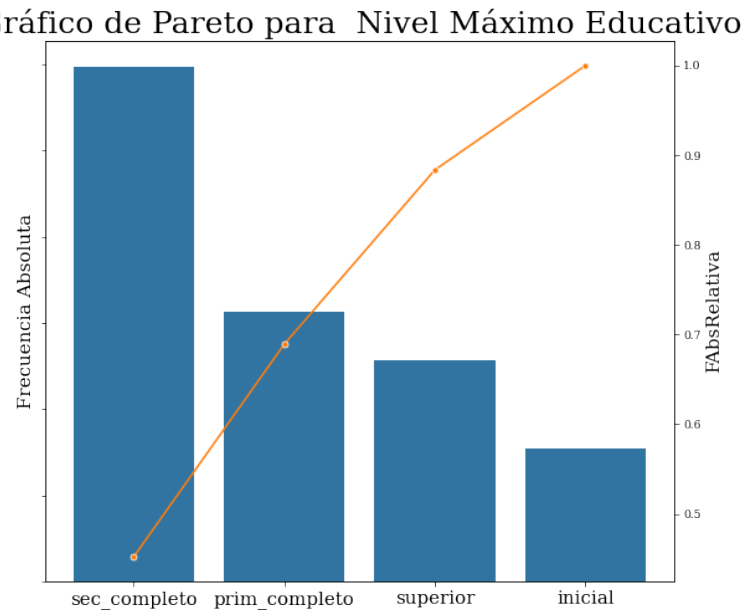
- Género: categorías balanceadas
- Edad: Distribución normal

Análisis univariado

Nivel máximo educativo:

- Moda de la variable: secundario completo
- El nivel secundario y primario explican casi el 77 % de los datos.

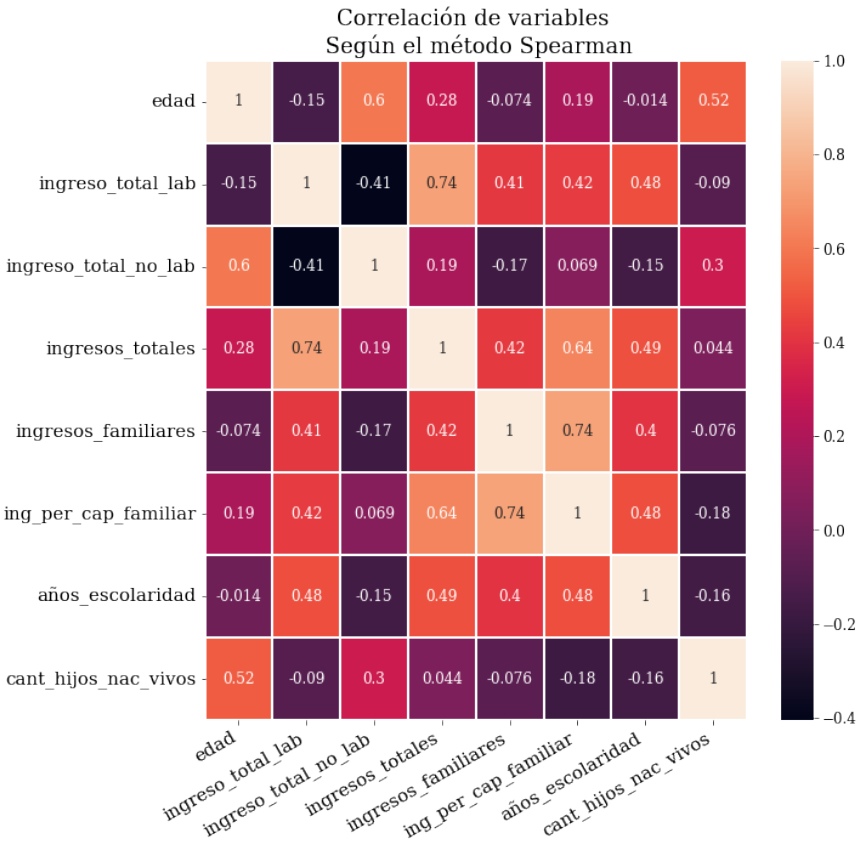




Análisis bivariado

Variable numéricas:

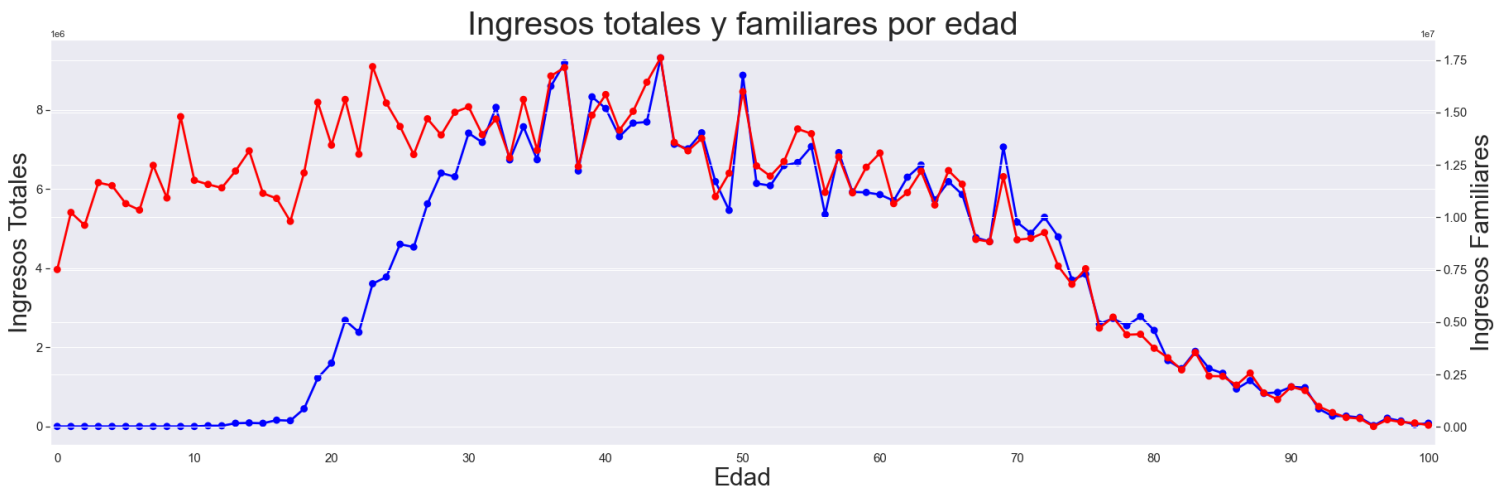
Realizamos un mapa de calor para ver la interacción entre las variables numéricas.



- No se observan fuertes correlaciones.
- "años_escolaridad" correlaciona moderadamente bien con variables relacionadas al ingreso.

Análisis bivariado

Ingresos por edad:

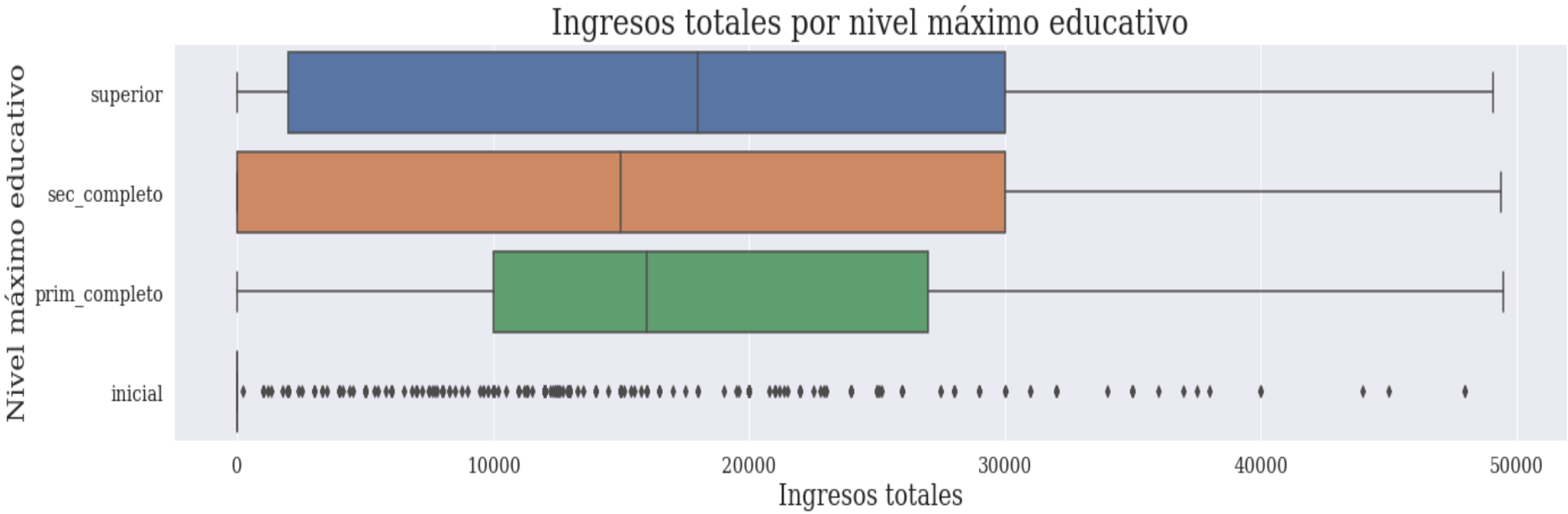


Desde los 30 años en adelante el ingreso total de la persona se corresponde con el ingreso familiar. Por ende, suele haber un único ingreso fuerte por grupo familiar.

Análisis bivariado

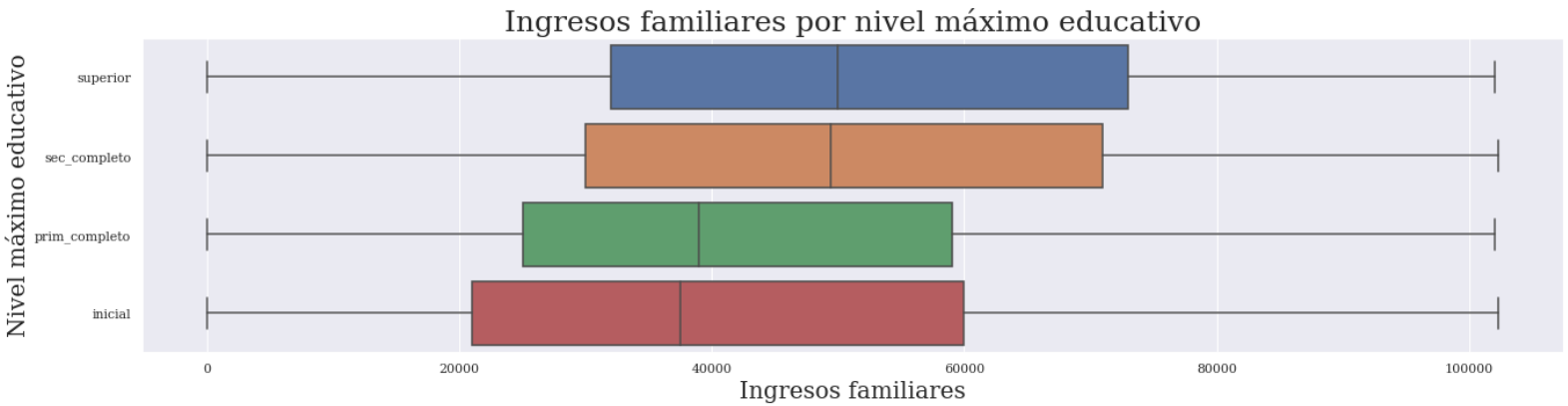
Variable numéricas y categóricas

Comparamos algunas variables con nuestro target, comenzando con los ingresos totales.



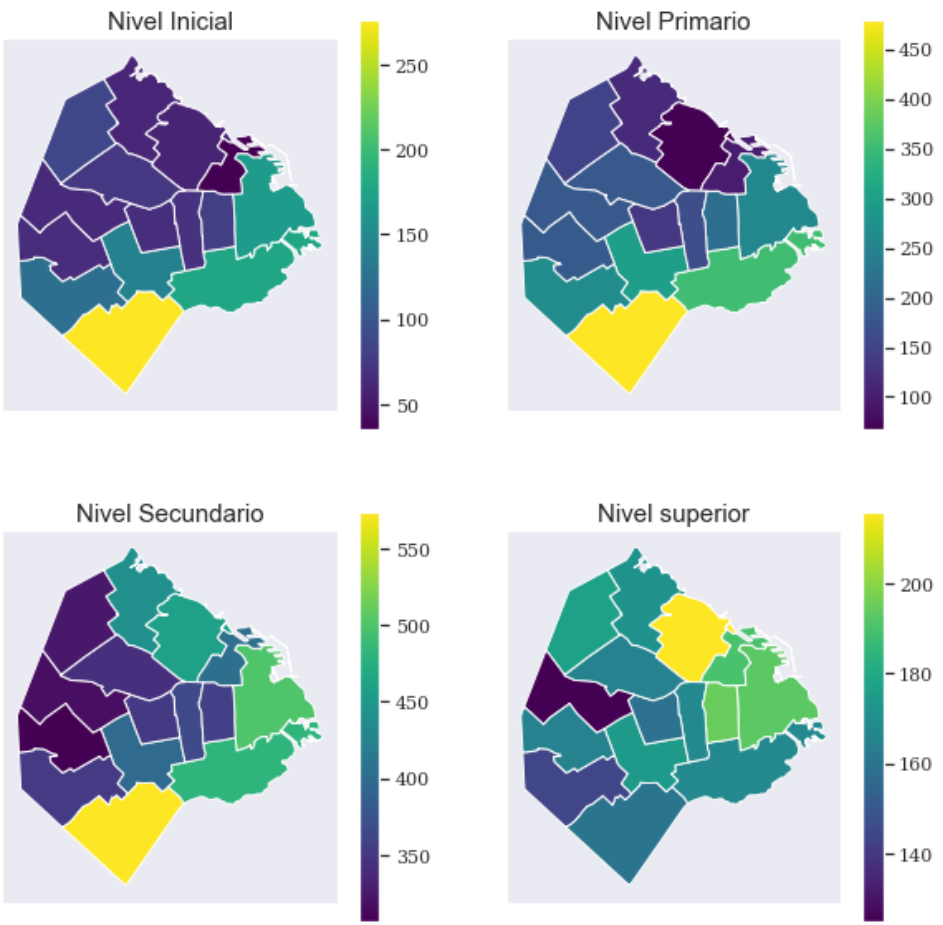
Análisis bivariado

Distribución de los ingresos familiares con respecto a nuestro target:



En definitiva, se observa un desplazamiento de los valores centrales (dentro de la caja) hacia la izquierda a medida que aumenta el nivel educativo.

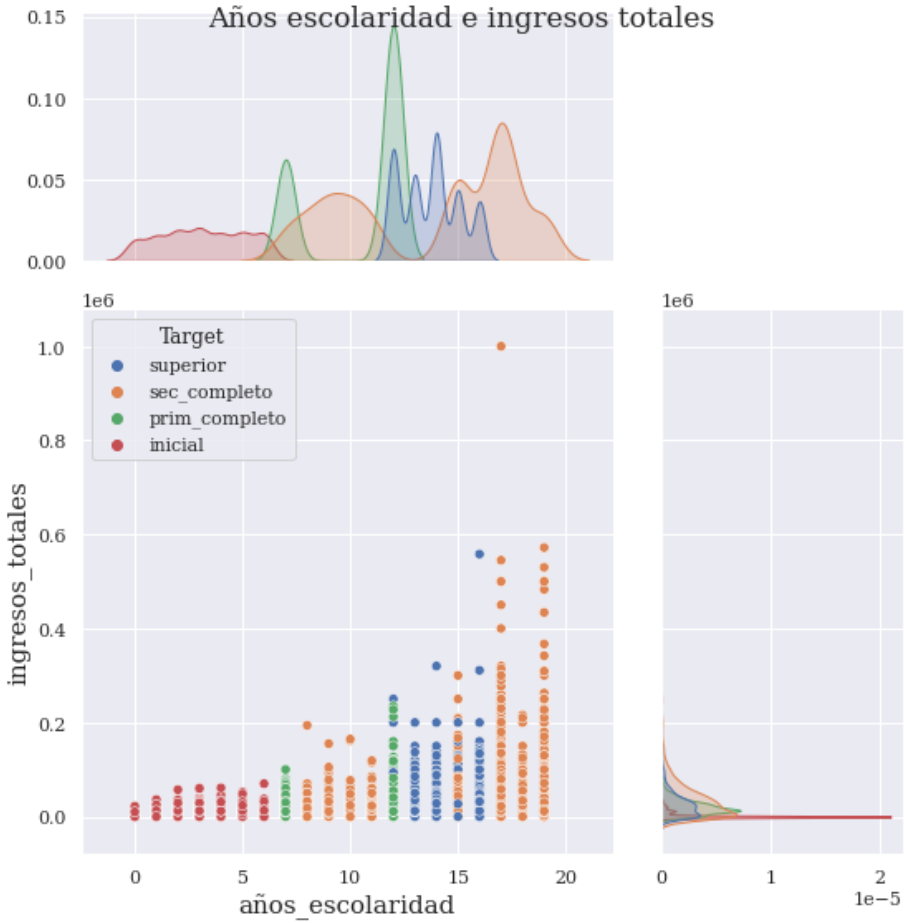
Análisis bivariado



- Sur de la ciudad: mayor cantidad de encuestados con nivel inicial, primario y secundario completo,
- Norte (particularmente el barrio de Palermo): mayor cantidad de personas con estudios superiores,
- Comunas del este ("centro de la ciudad"): Alta cantidad de encuestados con nivel superior.

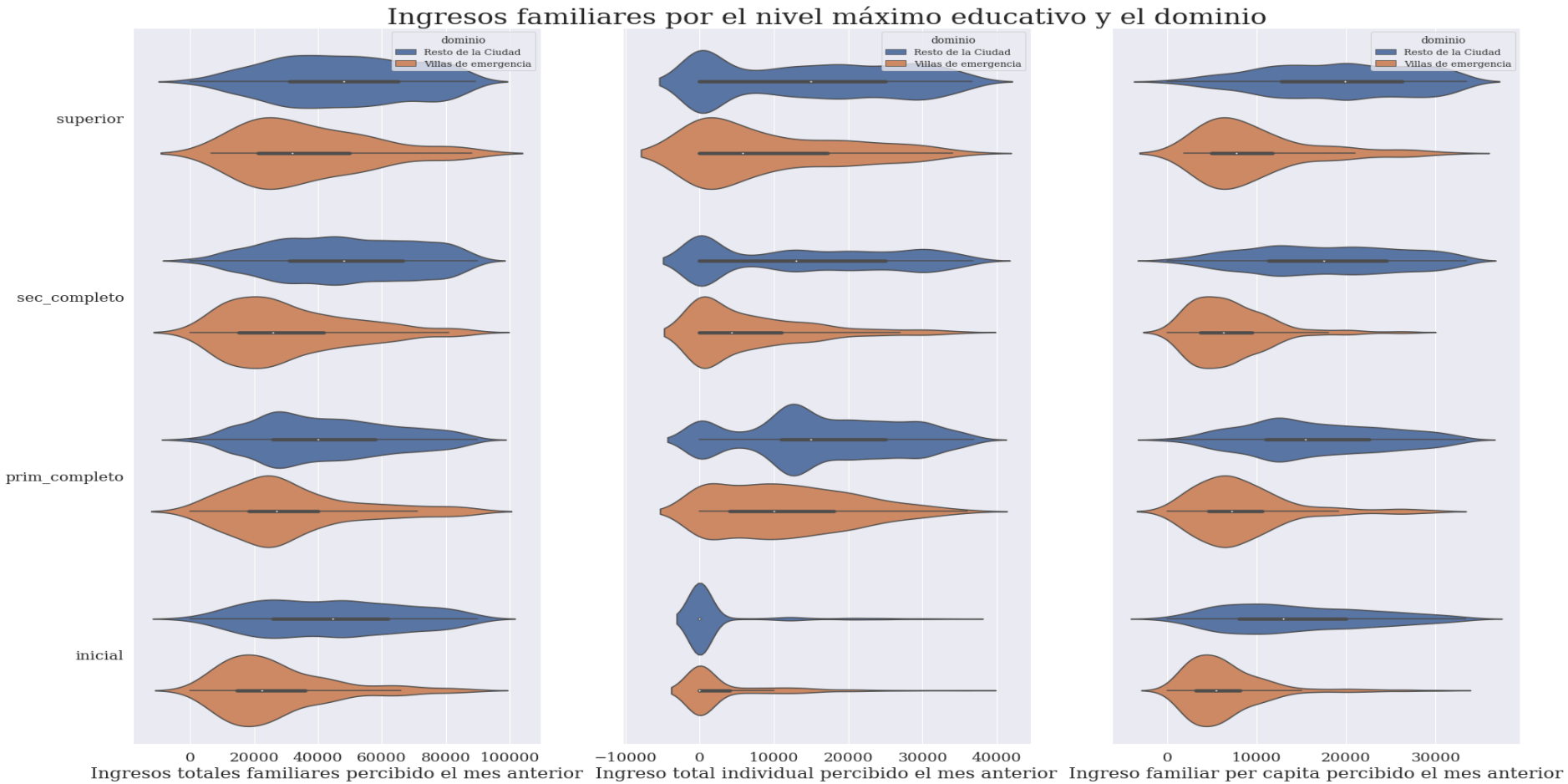
Análisis multivariado

Probamos de cruzar años de escolaridad, nivel máximo educativo y los ingresos totales.



- Hasta los 6 años todos los casos llegan al nivel inicial.
- Vemos dos años en que aparece el primario completo: 7 y 12 años. Estimamos que se debe a la división entre los que comenzaron su educación en la primaria y los que comenzaron en el nivel inicial.
- A partir de los 12 años: un aumento consistente de los ingresos totales.

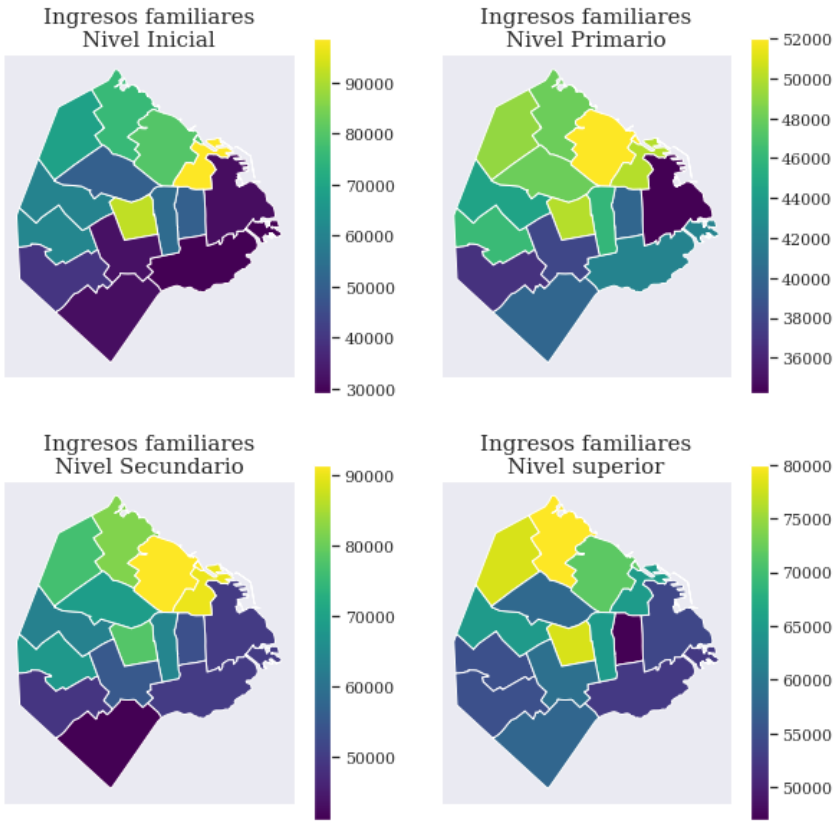
Análisis multivariado



Podemos ver que los casos que no provienen de villas de emergencia obtienen en promedio **ingresos más altos en todos los niveles educativos**. El alcanzar estudios superiores no parece homogeneizar ambos conjuntos.

Análisis multivariado

Ingresos familiares según el máximo nivel educativo alcanzado:



- a mayor nivel educativo, menor varianza de los ingresos familiares entre comunas
- ¿incluir a menores de edad sesga los valores? Queda pendiente realizar ese análisis

Transformación de variables

Transformamos variables para poder trabajar con los algoritmos:

- Recategorización de la variable "Target" en variables numéricas:
  - **inicial=** 1,
  - **prim_completo=** 2,
  - **sec_completo=** 3,
  - **superior=** 4,
- Reagrupación la variable "comuna" por regiones para reducir la dimensionalidad (norte, centro, sur y oeste),
- Eliminamos algunas variables que no resultan relevantes

Tratados de nulos

Encontramos valores nulos en estas variables, por lo que para trabajar con el modelado realizamos las siguientes acciones:

Variable	Nulos	Acción
situacion_conyugal	1	Reemplazamos con la moda
lugar_nacimiento	1	Reemplazamos con la moda
sector_educativo	3	Reemplazamos con la moda
afiliacion_salud	4	Reemplazamos con la moda
años_escolaridad	62	Reemplazamos con la mediana por comuna y sexo
nivel_max_educativo	1054	Eliminamos la variable
Target	1096	Eliminamos sus nulos y transformamos su tipo a entero
hijos_nacidos_vivos	7784	Reemplazamos con la moda

Árbol de decisión

1. Primer modelo:

Parámetros:

- max_depth=8,
- criterion='gini',

El Accuracy score para el test es de: **0.940** y las métricas:

	precision	recall	f1-score	support
Inicial	1.00	1.00	1.0	446
Primario	0.93	0.99	0.95	978
Secundario	0.95	0.96	0.95	1771
Superior	0.90	0.82	0.86	772
accuracy			0.94	3967
macro avg	0.94	0.94	0.94	3967
weighted avg	0.94	0.94	0.94	3967

Resultados

- **Bias o sesgo:** 96.89 % $\Rightarrow$ poco error $\Rightarrow$ sesgo bajo,
- **Variance=Test_Score** $-$ **Bias=** 2.89 % $\Rightarrow$ varianza baja.

Entonces, el modelo tiene una **buena relación** de sesgo y varianza.

Sin embargo, tenemos que la variable "años_escolaridad" tiene una importancia del 84 %, por mucho superior al resto de variables.

Por lo tanto, desarrollamos un nuevo modelo sin esta variable.

2. Segundo modelo:

Esta vez al correr el modelo, utilizaremos el "DecisionTreeClassifier" sin la variable años_escolaridad y con el criterion entropy.

	precision	recall	f1-score	suppo
Inicial	0.83	0.79	0.81	446
Primario	0.45	0.26	0.33	978
Secundario	0.56	0.66	0.60	1771
Superior	0.39	0.45	0.42	772
accuracy			0.53	3967
macro avg	0.56	0.59	0.54	3967
weighted avg	0.53	0.53	0.52	3967

Resultados

- **Bias o sesgo:** 99.78 % $\Rightarrow$ poco error $\Rightarrow$ sesgo bajo,
- **Variance=Test_Score** $-$ **Bias=** 46.39 % $\Rightarrow$ varianza bastante alta,

Lo que nos da como resultado, que este modelo esta haciendo **OVERFITTING**. Por lo que se observa, el árbol performa bastante peor sin esta variable, aumentando especialmente la varianza. Por lo tanto optamos probar mejorar nuestro modelo con un grid search.

Gridsearch con CV

En la grilla de parámetros para el Gridsearch elegimos los siguientes:

- Profundidad máxima del árbol: rango entre 5 y 10 niveles,
- Cantidad máxima de features: rango entre 11 y 13,
- Usamos todos los criterios posibles para el split: gini, entropy y log_loss;

A su vez, realizamos un cross validation partiendo el dataframe en 10 secciones.

Resultado: el mejor árbol de decisión posible obtiene 0.642, con las siguientes características.

- Profundidad de 6
- Utilizar 10 variables
- Usar el método “gini”

Gridsearch con CV

Entonces, entrenamos el modelo bajo estos mismos parámetros y obtenemos el siguiente reporte de clasificación:

	precision	recall	f1-score	suppo
Inicial	0.99	0.74	0.85	446
Primario	0.49	0.21	0.30	978
Secundario	0.55	0.87	0.69	1771
Superior	0.78	0.41	0.54	772
accuracy			0.60	3967
macro avg	0.70	0.56	0.59	3967
weighted avg	0.63	0.60	0.57	3967

Resultados

- **Bias o sesgo:** 65.27 % $\Rightarrow$ bastantes errores $\Rightarrow$ high bias,
- **Variance=Test_Score − Bias:** 5.14 % $\Rightarrow$ low variance.

Por lo tanto, el modelo esta haciendo **UNDERFITTING**

Conclusiones de mejora de modelo:

- Varianza: de 44.97 % a 5.14 % $\Rightarrow$ **MEJORÓ**
- Accuracy: de 100 % a 65.27 % $\Rightarrow$ **EMPEORÓ**

Random Forest Classifier

3. Tercer modelo:

En esta instancia volvemos a incluir la variable años de escolaridad.

En nuestro tercer modelo utilizamos el Random Forest Classifier con los siguientes parámetros:

- n_estimators (cantidad de árboles de decisión generados)=200,
- max_depth=15,
- criterion='gini.

Random Forest Classifier

Que nos da los siguientes resultados en cuanto a las métricas:

	precision	recall	f1-score	suppo
Inicial	1.00	0.96	0.98	446
Primario	0.86	0.95	0.90	978
Secundario	0.91	0.93	0.9	1771
Superior	0.92	0.76	0.83	772
accuracy			0.91	3967
macro avg	0.92	0.90	0.91	3967
weighted avg	0.91	0.91	0.90	3967

El random forest performa bastante bien, es decir, mucho mejor que los modelos anteriores.

Resultados

- **Bias o sesgo:** 97.80 % $\Rightarrow$ pocos errores $\Rightarrow$ sesgo bajo,
- **Variance=Test_Score − Bias:** 7.20 % $\Rightarrow$ la varianza es baja.

Obtuvimos un buen modelo. No obstante, buscamos cuales son las variables más importantes. Encontramos que los años de escolaridad redujo la enorme importancia (a un 43.58 %) que tenía en el random tree.

Sin embargo, sigue correspondiendo quitarla del modelo.

4. Cuarto modelo:

En este caso elegimos los siguientes parámetros, quitando la variable años de escolaridad:

- n_estimators=200,
- max_depth=10,
- criterion='gini'.

Dándonos por resultado los siguientes medidas de desempeño:

	precision	recall	f1-score	suppo
Inicial	0.95	0.78	0.86	446
Primario	0.54	0.23	0.32	978
Secundario	0.56	0.88	0.69	1771
Superior	0.80	0.40	0.53	772
accuracy			0.62	3967
macro avg	0.71	0.57	0.60	3967
weighted avg	0.65	0.62	0.59	3967

Resultados

- **Bias o sesgo:** 89.11 % $\Rightarrow$ pocos errores $\Rightarrow$ sesgo bajo,
- **Variance=Test_Score** $-$ **Bias=** 27.4 % $\Rightarrow$ varianza alta.

El modelo empeora su accuracy pero está muy cercano al mejor modelo de Random Tree, mientras que crece mucho la varianza. Vamos a probar mejorándolo con grid search.

Gridsearch con CV

En la grilla de parámetros para el Gridsearch elegimos los siguientes:

- Profundidad máxima: 5,7,10,15 de profundidad, agregando la opción de que no tenga máximo,
- Cantidad máxima de features: 5,8,10,30,41,
- Número de estimadores: 200,300,500,
- Criterion: [‘gini’,‘entropy’,‘log_loss’],
- También realizamos un cross validation de 10 particiones.

Como resultado, el mejor random forest posible obtiene 0.668. Para eso el árbol debe tener:

- una profundidad de 15,
- utilizar 10 variables,
- tener 300 estimadores
- y utilizar el método “gini”.

Gridsearch con CV

Entonces, entrenamos el modelo bajo estos mismos parámetros y obtenemos el siguiente reporte de clasificación:

	precision	recall	f1-score	support
Inicial	0.95	0.79	0.86	446
Primario	0.56	0.23	0.33	978
Secundario	0.56	0.89	0.69	1771
Superior	0.80	0.40	0.54	772
accuracy			0.62	3967
macro avg	0.72	0.58	0.60	3967
weighted avg	0.65	0.62	0.59	3967

Resultados

- **Bias o sesgo:** 90.65 % $\Rightarrow$ pocos errores $\Rightarrow$ sesgo bajo,
- **Variance=Test_Score − Bias:** 28.54 % $\Rightarrow$ varianza alta.

Lo que nos indica que nuestro modelo esta haciendo **OVERFITING**. Al utilizar random forest hemos podido mejorar el sesgo y disminuir el underfitting en 2 puntos porcentuales aproximadamente.

Sin embargo, se vio afectada la varianza en estos modelos, que pasó de estar alrededor del 5 % en el árbol de decisión mejorado a 28 %.

Conclusiones Finales

Finalmente, tomamos las métricas de cada uno de ellos y hacemos un cuadro comparativo:

modelo	accuracy	sesgo	varianza	f1_inicial	f1_pri	f1_sec	f1_sup
árbol_default	0.53	1.00	0.46	0.81	0.33	0.60	0.42
árbol_mejorado	0.60	0.65	0.05	0.85	0.30	0.69	0.53
bosque_default	0.62	0.89	0.27	0.86	0.32	0.69	0.53
bosque_mejorado	0.62	0.91	0.29	0.86	0.33	0.69	0.54

- El árbol default tiene el mejor resultado con respecto al sesgo, pero su varianza lo deja afuera de la competencia.
- Por el contrario, el árbol mejorado tiene una varianza insuperable de 5 %, aunque con el menor puntaje con respecto al sesgo.
- El bosque default tiene resultados mixtos en ambas categorías.
- El bosque mejorado destaca por bajo sesgo pero su varianza es la segunda peor.

Conclusiones Finales

Los finalistas son **el árbol y el bosque mejorado**. Ambos performan muy bien pero en métricas diferentes.

En nuestra opinión, es **el árbol mejorado el ganador**, ya que:

- Tiene la robustez suficiente para poder generalizar en caso de agregar nuevos datos al modelo.
- Tiene mayor velocidad de entrenamiento
- Tiene mayor capacidad de ser visualizada en un gráfico.

En futuras líneas de investigación se debería investigar en profundidad el desbalanceo de datos propio del Target y mitigar la problemática agregando datos en categorías con deficit y eliminando datos de categorías en exceso.